

# Dvě aplikace bayesovských sítí\*

Jiří Vomlel

vomlel@utia.cas.cz

Laboratoř inteligentních systémů,  
Vysoká škola ekonomická,  
Praha

**Abstrakt** V článku představujeme dvě moderní aplikace bayesovských sítí - adaptivní testování a technickou diagnostiku (angl. troubleshooting). Nejprve stručně shrneme teoretické základy a navrhujeme obecný rámec použitý při vytváření strategií pomocí bayesovských sítí. Následně popíšeme aplikaci teorie při adaptivním testování a v technické diagnostice. Tento článek vychází ze zkušeností, které jsme získali při řešení dvou projektů: - studentském semestrálním projektu během něhož byl navržen adaptivní test znalostí žáků řešících základní úlohy se zlomky - a při projektu SACSO, společném projektu firmy Hewlett-Packard a Aalborgské univerzity v Dánsku, zaměřeném na diagnostiku složitých elektro-mechanických zařízení.

## 1 Úvod

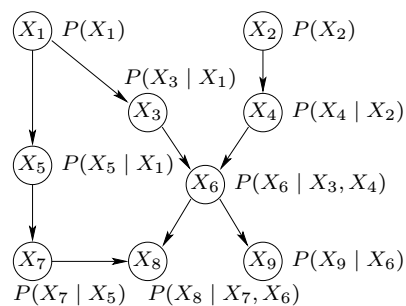
V této kapitole představíme bayesovské sítě a popíšeme dvě základní úlohy společné pro obě aplikace popsané v tomto článku: (1) vytváření modelu bayesovské sítě a (2) použití modelu pro hledání řešící strategie.

### 1.1 Bayesovské sítě

Bayesovské sítě patří do skupiny pravděpodobnostních grafových modelů které jsou schopny modelovat problémy, ve kterých je třeba pracovat s nejistými (vágními) informacemi. Návrh použít bayesovské sítě v oblasti expertních systémů se poprvé objevuje v pracích [18] a [21]. První skutečnou aplikací byl expertní systém pro elektromyografii Munin [2] a Pathfinder system [7]. Od té doby byly bayesovské sítě úspěšně použity v mnoha oblastech. Síla těchto modelů je založena nejen na jejich schopnosti provádět efektivně výpočty se stovkami veličin zároveň, např. použitím metody popsané v pracích [14]) a v [11], ale také na jejich schopnosti pomoci člověku lépe porozumět modelované oblasti. Toho je dosaženo hlavně díky velmi dobře srozumitelné reprezentaci nezávislostí mezi veličinami pomocí acyklických orientovaných grafů.

*Bayesovská síť* je tvořena acyklických orientovaným grafem  $G = (V, E)$ , ke každému uzlu  $i \in V$  je přiřazena jedna náhodná veličina  $X_i$  s konečnou množinou  $\mathbb{X}_i$  navzájem disjunktních stavů a tabulka podmíněné pravděpodobnosti  $P(X_i | (X_j)_{j \in pa(i)})$ , kde  $pa(i)$  označuje množinu rodičů uzlu  $i$  v grafu  $G$ . Viz obrázek 1, kde je uveden příklad bayesovské sítě.

\* Tato práce je českou verzí článku publikovaného na konferenci Znalosti 2003. Autor byl podpořen grantem Grantové agentury České republiky číslo 201/02/1269.



Obrázek 1. Příklad bayesovské sítě

Bayesovská síť reprezentuje kvalitativní i kvantitativní znalosti. Kvantitativní znalosti jsou reprezentovány pomocí tabulek podmíněné pravděpodobnosti, zatímco kvalitativní znalosti pomocí acyklického orientovaného grafu. Tento graf vyjadřuje vztahy podmíněné nezávislosti mezi veličinami  $(X_i)_{i \in V}$ . Koncept nazývaný *d-separace*, zavedený v práci [19], může být využit pro získání podmíněných nezávislostí z grafu. Jestliže  $P(X_i | \mathcal{Y}) = P(X_i | \mathcal{Y}, X_j)$  říkáme, že veličiny  $X_i, X_j$  jsou *podmíněně nezávislé*, je-li dána množina veličin  $\mathcal{Y}$ . Veličiny  $X_i, X_j$  jsou *d-separovány* množinou  $\mathcal{Y}$  jestliže, pro všechny cesty<sup>1</sup> spojující uzly  $i$  a  $j$  existuje mezilehlý uzel  $k$  takový, že buď

- (1) hrany se nesetkávají “head-to-head” v uzlu  $k$  a  $X_k \in \mathcal{Y}$ , nebo
- (2) hrany se setkávají “head-to-head” v uzlu  $k$  a ani  $X_k$ , ani žádný jeho potomek v grafu  $G$ , nepatří do  $\mathcal{Y}$ .

Základní požadavek na pravděpodobnostní rozdělení  $P$  odpovídající bayesovské síti je, aby všechny veličiny  $X_i, X_j$ , které jsou d-separovány množinou  $\mathcal{Y}$ , byly podmíněně nezávislé dáno  $\mathcal{Y}$  v  $P$ . Poměrně snadno se dá ukázat, že pravděpodobnostní rozdělení splňující uvedený požadavek a zároveň mající podmíněné pravděpodobnosti rovny  $P(X_i | (X_j)_{j \in pa(i)})$ ,  $i \in V$  je právě jedno a rovná se součinu podmíněných pravděpodobností, t.j.

$$P((X_i)_{i \in V}) = \prod_{i \in V} P(X_i | (X_j)_{j \in pa(i)}) .$$

Čtenáře, které by zajímal detailní úvod do teorie bayesovských sítí, odkazujeme na knihu [9].

## 1.2 Vytváření modelů

Pro vytváření modelů bayesovských sítí se obvykle používají tři základní přístupy:

<sup>1</sup> Cestou v orientovaném grafu  $G$  rozumíme posloupnost uzlů  $v_1, \dots, v_k$  takovou, že pro  $i = 2, \dots, k$  buď  $(v_{i-1}, v_i)$  nebo  $(v_i, v_{i-1})$  je hranou grafu  $G$  a pro  $i \neq j$  platí, že  $v_i \neq v_j$ .

- Návrhář modelu konzultuje experta oboru, který mu poskytne expertní znalosti o modelovaném problému, oblasti, či zařízení. Tyto znalosti návrhář použije při vytváření struktury modelu a pro specifikaci číselných hodnot v tabulkách podmíněné pravděpodobnosti.
- Shromáždí se data obsahující záznamy o chování modelovaného problému, oblasti, či zařízení. Následně se použije některá metoda strojového učení, pomocí které získáme jak strukturu modelu, tak odhady číselných hodnot v tabulkách podmíněné pravděpodobnosti.
- Třetí základní přístup je kombinace předchozích dvou. Například struktura modelu je navržena při konzultacích s expertem, ale číselné parametry jsou odhadovány ze shromážděných dat pomocí některé ze statistických metod.

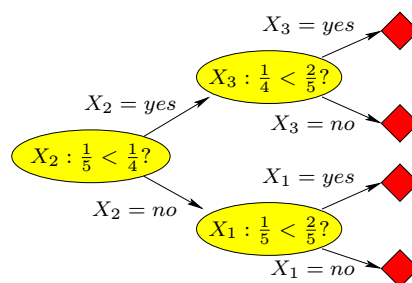
I stručný přehled různých metod, které je možné použít pro vytváření modelů bayesovských sítí by značně překročil rozsah tohoto článku. Více informací může čtenář nalézt například v knize [5], kde je několik kapitol věnováno učení modelů. V kapitole 2 popíšeme, jak byla vytvořena bayesovská síť použitá při adaptivním testování znalostí.

### 1.3 Vytváření strategií pomocí modelů

*Strategie* popisuje kroky, které má uživatel systému učinit, aby dosáhl požadovaný cíl. Příklady kroků jsou: (1) akce na zařízení, kterou uživatel udělá, nebo (2) pozorování, které uživatel učiní, nebo (3) odpověď uživatele na položenou otázku. Protože není dopředu zřejmé, jaký bude výsledek provedení kroku, každá strategie musí specifikovat, co má uživatel učinit pro všechny možné kombinace výsledků všech předchozích kroků. Takže strategie může být reprezentována pomocí orientovaného stromu. Je výhodné definovat dva druhy uzlů stromu - *náhodné uzly* a *terminální uzly*. Každý náhodný uzel odpovídá jednomu kroku strategie. Terminální uzly jsou listy stromu, kde strategie končí. Jedno použití strategie, které budeme nazývat *sezení*, odpovídá cestě ve stromu začínající v kořenu stromu a končící v některém z uzlů stromu. Obrázek 2 představuje příklad strategie adaptivního testu složené ze dvou otázek. Elipsy představují náhodné uzly a kosočtverce terminální uzly. Každý náhodný uzel je označen názvem kroku, který mu odpovídá. Každá hrana vycházející z náhodného uzlu je označena výsledkem kroku odpovídajícího dotyčnému uzlu. Strategie reprezentovaná tímto stromem je následující: *Jestliže odpověď na první otázku  $X_2$  je správně pak druhá otázka je  $X_3$  jinak druhá otázka je  $X_1$ .*

Nechť  $\mathcal{S}$  označuje množinu všech přípustných strategií řešení daného problému a necht'  $\mathcal{L}(s)$  označuje množinu všech terminálních uzlů strategie  $s \in \mathcal{S}$ . Definujeme hodnotící funkci  $f : \cup_{s \in \mathcal{S}} \mathcal{L}(s) \mapsto \mathbb{R}$ . Cílem je minimalizovat hodnotu této funkce na konci sezení. Výstupy navržených kroků nejsou dopředu známy, pouze pravděpodobnost  $P(\mathbf{e}_\ell)$  ukončení sezení v uzlu  $\ell \in \mathcal{L}$  může být spočtena pomocí modelu reprezentovaného bayesovskou sítí. Pro strategii  $s \in \mathcal{S}$  definujeme očekávanou hodnotu hodnotící funkce

$$E_f(s) = \sum_{\ell \in \mathcal{L}(s)} P(\mathbf{e}_\ell) \cdot f(\mathbf{e}_\ell) . \quad (1)$$



**Obrázek 2.** Příklad strategie adaptivního testu

Cíl tak může být formalizován jako hledání strategie  $s^* \in \mathcal{S}$  mající minimální hodnotu  $E_f(s)$  ze všech přípustných strategií  $s \in \mathcal{S}$ . Často se stává, že není možné, kvůli kombinatorické složitosti problému, najít optimální strategii. Místo toho se používají různé heuristické metody poskytující suboptimální řešení.

V kapitole 2 je tento přístup použit v adaptivním testování a v kapitole 3 v technické diagnostice. Tyto aplikace mají různé typy modelů bayesovských sítí, různě definované množiny přípustných strategií a rozdílné hodnotící funkce  $f$ . Uvidíme, že také metody použité při hledání suboptimální strategií jsou rozdílné.

## 2 Adaptivní testování

Testy, které se automaticky přizpůsobují zjištěné úrovni znalostí zkoušeného, se nazývají *adaptivní testy*. Po získání odpovědi na položenou otázku, automatický systém vybere následující otázku s využitím znalosti získané na základě předchozích odpovědí zkoušeného. Jednoduchý příklad adaptivního testu byl ukázán na obrázku 2. Protože realizace adaptivního testu vyžaduje použití počítačů, je tento přístup často nazýván *počítačové adaptivní testování*, anglicky *computerized adaptive testing* (CAT). Více informací je možné získat v publikacích [25] a [15].

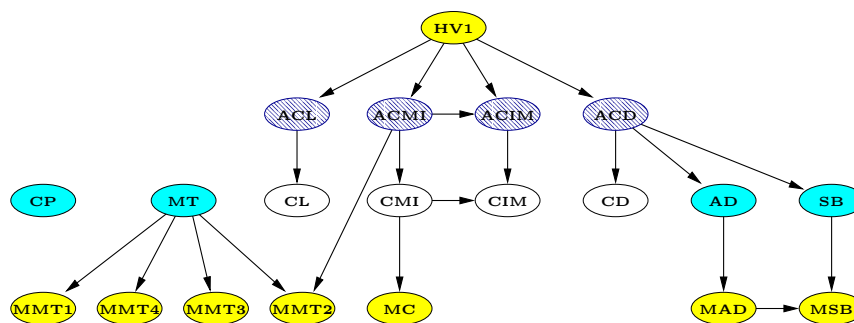
### 2.1 Bayesovská síť pro adaptivní testování

Autoři práce [1] navrhli použití grafových pravděpodobnostních modelů pro počítačové adaptivní testování. Jejich model se skládá z *modelu studenta* a několika *pozorovacích modelů*. Každý pozorovací model odpovídá jedné otázce. Nejprve návrhář testu specifikuje testované znalosti a dovednosti  $\mathcal{Y} = \{Y_1, \dots, Y_k\}$  a připraví soubor otázek  $\mathcal{X} = \{X_1, \dots, X_m\}$ . Model studenta popisuje vztahy mezi znalostmi a dovednostmi studenta. Expertní znalost o studentech je tak reprezentována sdruženým pravděpodobnostním rozdělením  $P(Y_1, \dots, Y_k)$  definovaném pro veličiny z modelu studenta. Nyní se budeme věnovat konkrétnímu případu vytváření modelu studenta řešícího základní úlohy se zlomky [23].

Nejprve, skupina studentů Aalborgské univerzity připravila papírové testy. Tyto testy řešili studenti prvního ročníku střední školy v obci Brønderslev. Byly testovány čtyři základní dovednosti - sčítání, odčítání, násobení a porovnávání zlomků, čtyři operační dovednosti - krácení zlomků, převod nepravých zlomků na složené zlomky a obráceně, hledání společného dělitele a schopnosti aplikovat operační dovednosti v komplexních úlohách. Studenti Aalborgské univerzity shrnuli výsledky jako seznam datových záznamů o jednotlivých studentech. Bylo objeveno několik typických špatných přístupů k řešení některých operací, které byly také zahrnuty jako veličiny do modelu studenta.

Jako druhý krok byla pomocí PC-algoritmu [22], implementovaného v systému Hugin [8], vytvořena struktura modelu studenta. Tato automatická metoda učení poskytla první náhled na vztahy mezi veličinami. Následně byly některé vztahy mezi veličinami vysvětleny pomocí takzvaných latentních veličin<sup>2</sup>. Byly také vynuceny některé závislosti mezi veličinami pomocí vynucených hran v modelu. Výsledný model byl opět naučen pomocí PC-algoritmu avšak při použití výše zmiňovaných požadavků na výsledný model. Číselné parametry konečného modelu byly odhadnuty z dat pomocí EM-algoritmu [13].

Výsledný model studenta je zobrazen na obrázku 3. Uzly v první řadě odspoda odpovídají pozorovaným špatným postupům, šedé uzly ve druhé řadě základním dovednostem, nevyplněné uzly v druhé řadě odpovídají operačním dovednostem a šedé uzly ve třetí řadě aplikačním dovednostem. Uzel umístěný nejvýše odpovídá jedné latentní veličině. Detailnější popis modelu je možné nalézt v článku [23].

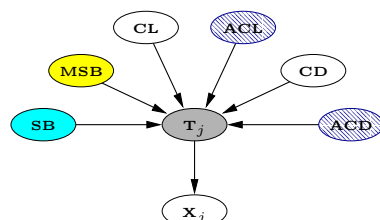


Obrázek 3. Model studenta.

Jak bylo uvedeno na začátku této kapitoly, pro každou otázku  $X_j \in \mathcal{X}$  je vytvořen jeden pozorovací model. Tento model popisuje závislost mezi danou otázkou  $X_j$  a relevantními dovednostmi, případně špatnými postupy v modelu studenta. Příklad úlohy je  $\frac{1}{3} - \frac{1}{12} = \frac{4}{12} - \frac{1}{12} = \frac{3}{12} = \frac{1}{4}$ . Aby student byl schopen vyřešit tuto úlohu měl by mít dovednosti  $SB$  (odčítání),  $CL$  (krácení),  $ACL$  (aplikace krácení),  $CD$  (nalezení společného dělitele),  $ACD$  (aplikace nalezení společného dělitele), a neměl by mít  $MSB$

<sup>2</sup> Latentní veličina je veličina, jejíž hodnota není v datech nikdy pozorována. Může však například modelovat společnou nepozorovanou příčinu několika pozorovaných veličin.

špatný postup při odčítání. Na obrázku 4 je zobrazena struktura pozorovacího modelu pro tuto úlohu. Vztah mezi veličinou  $T_j$  a souvisejícími dovednostmi a špatným postupem



**Obrázek 4.** Pozorovací model úlohy  $X_j$ .

může být popsán boolovskou funkcí. Ve skutečnosti ale student může udělat chybu, ikdyž má všechny potřebné předpoklady pro správné řešení úlohy. Na druhou stranu, správná odpověď nemusí vždy znamenat, že student má všechny potřebné předpoklady, například když správnou odpověď pouze uhodl. Tato nejistota je modelována pomocí podmíněného pravděpodobnostního rozdělení  $P(X_j | T_j)$  odhadnutého ze shromážděných dat.

## 2.2 Vytváření adaptivních testů

Obvykle adaptivní test skončí, když byl položen předem určený počet otázek, nebo když zkoušející získal postačující informace o zkoušeném. Takové podmínky definují množinu přípustných strategií  $\mathcal{S}$ .

Každý zkoušející chce maximalizovat informace, které má o zkoušeném na konci testu. Jednou z možností formalizace tohoto požadavku je usilovat o pravděpodobnostní rozdělení  $P(Y_1, \dots, Y_k)$  minimalizující Shannonovu entropii na konci testu [3]. Shannonova entropie (zkráceně entropie)  $H(P)$  pravděpodobnostního rozdělení  $P(Y_1, \dots, Y_k)$  je definována následovně:

$$H(P) = - \sum_{y_1, \dots, y_k} P(Y_1 = y_1, \dots, Y_k = y_k) \cdot \log P(Y_1 = y_1, \dots, Y_k = y_k) .$$

V každém terminálním uzlu  $\ell$  testovací strategie  $s$  spočteme entropii podmíněného pravděpodobnostního rozdělení podmíněného získanou evidencí, t.j.

$$H(\mathbf{e}_\ell) = H(P(Y_1, \dots, Y_k | \mathbf{e}_\ell)) .$$

Použijeme-li substituci  $f(\mathbf{e}_\ell) = H(\mathbf{e}_\ell)$  vzorec 1 můžeme přepsat jako

$$E_H(\mathbf{s}) = \sum_{\ell \in \mathcal{L}(\mathbf{s})} P(\mathbf{e}_\ell) \cdot H(\mathbf{e}_\ell) . \quad (2)$$

Cílem je tedy najít testovací strategie  $\mathbf{s} \in \mathcal{S}$  minimalizující očekávanou entropii  $E_H(\mathbf{s})$ .

V praxi se často používají tzv. hladové heuristiky, pomocí který nalezneme suboptimální test. Test, který se nazývá *krátkozrace optimální* je test složený z otázek vybraných tak, že každá otázka minimalizuje očekávanou entropii po odpovězení jedné otázky. Strategie nalezené tímto způsobem nemusí být optimální, ale v praxi se často ukazuje, že se příliš neliší od optimálních strategií.

### 2.3 Výsledky a porovnání s jinými přístupy

Klasickým přístupem používaným v testování znalostí od 60. let 20. století je *teorie odezvy* (angl. item response theory - IRT) [16, 20]. Tato metoda modeluje studenta pomocí jedné veličiny  $\Theta$ . Tento jednoduchý model je vhodný například pro hodnocení studentů, ale není schopen poskytnout detailnější diagnostiku. Rozšířením je vícerozměrná teorie odezvy, kde se používá několik veličin reprezentujících různé aspekty související se znalostmi a schopnostmi zkoušeného. Aplikace bayesovských sítí je možné považovat za zobecnění vícerozměrná teorie odezvy přinášející dvě základní výhody.

- Bayesovská síť je schopna lépe modelovat usuzování studenta a umožňuje hlubší pochopení modelovaného problému.
- Protože model studenta reprezentovaný pomocí bayesovské sítě je schopen zachytit vztahy mezi testovanými dovednostmi a schopnostmi, je možné podstatně zkrátit adaptivní test při zachování spolehlivosti výsledků.

Výsledky experimentů s adaptivními testy využívajícími bayesovské sítě, prezentované v článku [23], ukazují, že bylo možné dobře odhadovat skutečné dovednosti studenta. V průměru více než 90% dovedností bylo správně odhadnuto již po sedmi položených otázkách. V papírových (neadaptivních testech) bylo třeba položit 20 otázek pro dosažení stejné kvality predikce.

Další experimenty s bayesovskými sítěmi v oblasti adaptivního testování a výuky (angl. tutoring) jsou popsány v článku [17], respektive v článku [4].

## 3 Technická diagnostika

Technická diagnostika je často velmi složitá úloha. Proto počítačové systémy využívající postupně získané informace o diagnostikovaném zařízení mohou podstatně zrychlit diagnostický proces [6].

### 3.1 Bayesovské sítě v technické diagnostice

V článku [10] je popsáno využití bayesovské sítě v technické diagnostice. Model bayesovské sítě zachycuje vztahy mezi třemi typy veličin: *poruchami* zařízení  $F \in \mathcal{F}$ , *akcemi*  $A \in \mathcal{A}$ , t.j. opravnými kroky které mohou odstranit poruchu a *pozorováními*  $Q \in \mathcal{Q}$  - kroky které nemohou odstranit poruchu, ale mohou ji pomoci odhalit. Každé akci a pozorování je přiřazena cena  $c$ , která může například představovat čas potřebný pro provedení dotyčné akce či pozorování, nebo cenu, kterou je třeba za provedení akce či pozorování

zaplatit. Použijeme zjednodušený příklad z článku [24], abychom ilustrovali, jak může probíhat technická diagnostika laserové tiskárny.

*Example 1 (Světlý tisk).* Předpokládejme, že tiskárna vytiskne stránku, která je příliš světlá. Tento problém může mít mnoho příčin. Uvažujme zjednodušený model obsahující pouze 4 možné příčiny světlého tisku:  $F_1$  distribuční problém toneru,  $F_2$  vadný toner,  $F_3$  narušený tok dat, a  $F_4$  špatné nastavení ovladače. Necht' akce, které mohou vyřešit tento problém jsou:  $A_1$  "vyjměte toner, zatřeste s ním a uložte ho zpět" s cenou  $c_1 = 5$ ,  $A_2$  "zkuste jiný toner" s cenou  $c_2 = 15$  a  $A_3$  "vypněte a zase zapněte tiskárnu" s cenou  $c_3 = 1$ . Pro každou akci expert zadal podmíněnou pravděpodobnost  $P(A_i = \text{yes} \mid F_j)$ . Například, akce  $A_2$  "zkuste jiný toner" vyřeší distribuční problém toneru a vadný toner s pravděpodobností 0.9, t.j.  $P(A_2 = \text{yes} \mid F_i) = 0.9, i = 1, 2$ , ale nevyřeší špatné nastavení ovladače, t.j.  $P(A_2 = \text{yes} \mid F_4) = 0$ .

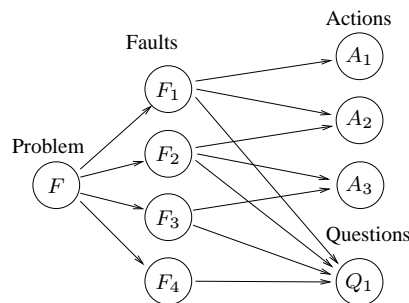
Často je výhodné položit během diagnostiky několik otázek o stavu zařízení. Odpovědi mohou pomoci identifikovat příčinu problému zařízení. Například, jestliže odpověď na otázku  $Q_1$  "Je konfigurační stránka vytištěna světle?" je negativní pak příčiny  $F_1$  distribuční problém toneru a  $F_2$  vadný toner je možné vyloučit. Pro každou otázku  $Q_i$  a každou příčinu  $F_j$  expert definuje podmíněnou pravděpodobnost  $P(Q_i = \text{yes} \mid F_j)$ .

U většiny technických zařízení je možné předpokládat, že akce a otázky jsou navzájem podmíněně nezávislé je-li známa příčina. T.j. pro každé  $A_i \in \mathcal{A}, Q_k \in \mathcal{Q}$

$$P(A_i \mid \mathcal{F}) = P(A_i \mid \mathcal{F}, \mathcal{V}) \text{ pro každé } \mathcal{V} \subseteq (\mathcal{A} \cup \mathcal{Q}) \setminus \{A_i\}$$

$$P(Q_k \mid \mathcal{F}) = P(Q_k \mid \mathcal{F}, \mathcal{U}) \text{ pro každé } \mathcal{U} \subseteq (\mathcal{A} \cup \mathcal{Q}) \setminus \{Q_k\} .$$

Dále je často možné předpokládat, že v jednom okamžiku způsobuje špatnou funkci zařízení pouze jedna příčina. Bayesovská síť na obrázku 5 zachycuje oba předpoklady.



**Obrázek 5.** Bayesovská síť pro zjednodušený problém světlého tisku

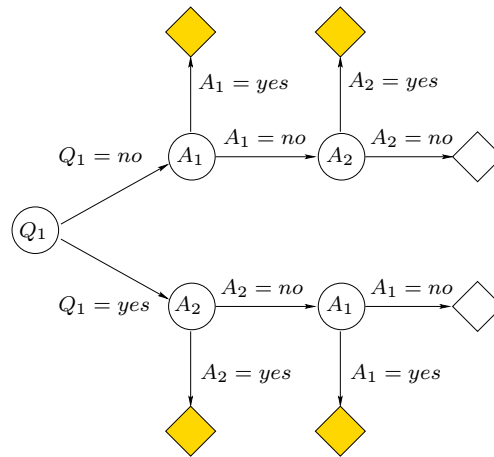


### 3.2 Diagnostická strategie

Každá diagnostická strategie může skončit dvěma způsoby, buď úspěšným vyřešením problému nebo neúspěšně - problém zůstane nevyřešen. Proto definujeme dva typy *terminálních uzlů*:

- (1) *úspěšné terminální uzly* odpovídají vyřešení problému,
- (2) *neúspěšné terminální uzly* odpovídají situaci, kdy diagnostika skončí dříve, než je problém vyřešen.

Obrázek 6 zobrazuje příklad diagnostické strategie. Úspěšné terminální uzly jsou šedé, zatímco neúspěšné terminální uzly jsou světlé.



Obrázek 6. Diagnostická strategie

Definujeme hodnotící funkci  $CR(\mathbf{e}_\ell)$ , která odpovídá ceně opravy zařízení. Skládá se ze dvou složek. První složka  $t(\mathbf{e}_\ell)$  je celková cena provedených akcí a položených otázek, které jsou na cestě do stavu  $\mathbf{e}_\ell$  odpovídajícímu uzlu  $\ell$  ve stromu strategie  $\mathbf{s}$ . Druhá složka je penalizační funkce  $c(\mathbf{e}_\ell)$ , která se aplikuje v každém terminálním uzlu. Jestliže je problém úspěšně vyřešen (v úspěšném terminálním uzlu), pak penalizační funkce je rovna nule. V neúspěšném terminálním uzlu má penalizační funkce kladnou hodnotu, která může být například interpretována jako cena za zavolání servisních techniků, kteří zařízení zaručeně opraví (například tím, že ho celé vymění). Takže  $CR(\mathbf{e}_\ell) = t(\mathbf{e}_\ell) + c(\mathbf{e}_\ell)$ . Provedeme-li substituci  $f(\mathbf{e}_\ell) = CR(\mathbf{e}_\ell)$  do vzorce 1 získáme kritérium nazývané *očekávaná cena opravy*.

$$E_{CR}(\mathbf{s}) = \sum_{\ell \in \mathcal{L}(\mathbf{s})} P(\mathbf{e}_\ell) \cdot ( t(\mathbf{e}_\ell) + c(\mathbf{e}_\ell) ) . \quad (3)$$

Základní úlohou technické diagnostiky je najít diagnostickou strategii  $\mathbf{s} \in \mathcal{S}$  minimalizující  $E_{CR}(\mathbf{s})$ .

Řešení je možné snadno najít v případě, když (1) každá akce může odstranit právě jednu příčinu, když (2) všechny akce jsou navzájem nezávislé, (3) v jednom okamžiku způsobuje špatnou funkci zařízení pouze jedna příčina a (4) není možné použít žádné otázky. V takovém případě, postačuje seřadit otázky sestupně podle poměru  $P(A = yes)/c_A$ , viz [12]. V práci [24] bylo ukázáno, že když některé akce mohou odstranit více než dvě příčiny, úloha nalezení optimální diagnostické strategie se stává  $NP$ -těžkou. Proto je třeba použít metody, které naleznou dostatečně dobré (ikdyž neoptimální) diagnostické strategie v polynomiálním čase.

### 3.3 Výsledky

Diagnostický systém vyvinutý v rámci projektu SACSO využívá bayesovskou síť pro reprezentaci diagnostického problému. Protože struktura modelu je jednoduchá (odpovídá tzv. naivnímu bayesovskému modelu), je možné při výběru dalšího kroku strategie provést velký počet výpočtů podmíněných pravděpodobností. Přístup použitý v projektu SACSO je založený na heuristikách využívajících poměr  $P(A = yes)/c_A$  a zároveň rozšiřuje přístup popsany v [6], v rámci kterého je možné vybírat i otázky. V článku [10] jsou diagnostické strategie získané pomocí přístupu projektu SACSO porovnány s optimálními strategiemi získanými výpočetně náročným prohledáváním stavového prostoru všech možných strategií. Hodnoty  $E_{CR}$  strategií získaných diagnostickým systémem SACSO byly velmi blízko optimálním hodnotám, průměrná odchylka byla méně než 2%. Tento diagnostický systém je nyní komerčně nabízen pod jménem DezisionWorks dánskou společností Dezide. Více informací o systému je možné nalézt na webovské stránce společnosti: <http://www.dezide.com/>.

### Reference

1. Russell G. Almond and Robert J. Mislevy. Graphical models and computerized adaptive testing. *Applied Psychological Measurement*, 23(3):223–237, 1999.
2. S. Andreassen, F. V. Jensen, S. K. Andersen, B. Falck, U. Kjærulff, M. Woldbye, A. R. Sørensen, A. Rosenfalck, and F. Jensen. MUNIN — an expert EMG assistant. In John E. Desmedt, editor, *Computer-Aided Electromyography and Expert Systems*, chapter 21. Elsevier Science Publishers, Amsterdam, 1989.
3. Moshe Ben-Bassat. Myopic policies in sequential classification. *IEEE Transactions on Computers*, 27(2):170–174, 1978.
4. Cristina Conati, Abigail S. Gertner, Kurt VanLehn, and Marek J. Druzdzel. On-line student modeling for coached problem solving using Bayesian networks. In Anthony Jameson, Cecile Paris, and Carlo Tasso, editors, *Proc. of the Sixth Int. Conf. on User Modeling (UM97)*, Chia Laguna, Sardinia, Italy. Springer Verlag, 1997.
5. Robert G. Cowell, A. Philip Dawid, Steffen L. Lauritzen, and David J. Spiegelhalter. *Probabilistic Networks and Expert Systems*. Statistics for Engineering and Information Science. Springer Verlag, 1999.
6. D. Heckerman, J. S. Breese, and K. Rommelse. Decision-theoretic troubleshooting. *Communications of the ACM*, 38(3):49–57, 1995.
7. David Heckerman, E. Horwitz, and B. Nathwani. Towards normative expert systems: Part I, the Pathfinder project. *Methods of Information in Medicine*, 31:90–105, 1992.
8. Hugin Explorer, ver. 6.0. Computer software, 2002. <http://www.hugin.com>.

9. Finn V. Jensen. *Bayesian networks and decision graphs*. Statistics for Engineering and Information Science. Springer Verlag, New York, Berlin, Heidelberg, 2001.
10. Finn V. Jensen, Uffe Kjærulff, Brian Kristiansen, Helge Langseth, Claus Skaanning, Jiří Vomlel, and Marta Vomlelová. The SACSO methodology for troubleshooting complex systems. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 15:321–333, 2001.
11. Finn V. Jensen, Steffen L. Lauritzen, and Kristian G. Olesen. Bayesian updating in recursive graphical models by local computation. *Computational Statistics Quarterly*, 4:269–282, 1990.
12. J. Kalagnanam and M. Henrion. A comparison of decision analysis and expert rules for sequential analysis. In P. Besnard and S. Hanks, editors, *The Fourth Conference on Uncertainty in Artificial Intelligence*, pages 271–281, New York, 1988.
13. Steffen L. Lauritzen. The EM-algorithm for graphical association models with missing data. *Computational Statistics and Data Analysis*, 1:191–201, 1995.
14. Steffen L. Lauritzen and David J. Spiegelhalter. Local computations with probabilities on graphical structures and their application to expert systems (with discussion). *Journal of the Royal Statistical Society, Series B*, 50:157–224, 1988.
15. Wim J. Van Der Linden and Cees A. W. Glas, editors. *Computerized Adaptive Testing: Theory and Practice*. Kluwer Academic Publishers, 2000.
16. F. M. Lord. *A theory of test scores*. Number 7 in Psychometric Monographs. Psychometric Society, 1952.
17. Eva Millán and José Luis Pérez-de-la-Cruz. A Bayesian diagnostic algorithm for student modeling and its evaluation. *User modeling and User-Adapted Interaction*, 12(2–3):281–330, 2002.
18. Judea Pearl. Reverend Bayes on inference engines: a distributed hierarchical approach. In *Proceedings, AAAI National Conference on AI, Pittsburgh, PA*, pages 133–136, August 1982.
19. Judea Pearl. Fusion, propagation and structuring in belief networks. *Artificial Intelligence*, 29(3):241–288, 1986.
20. G. Rasch. Probabilistic models for some intelligence and attainment tests. Technical report, Danish Institute for Educational Research, Copenhagen, 1960.
21. D. J. Spiegelhalter and R. P. Knill-Jones. Statistical and knowledge-based approaches to clinical decision-support systems. *Journal of the Royal Statistical Society, Series A*, (147):35–77, 1984.
22. Peter Spirtes, Clark Glymour, and Richard Scheines. *Causation, Prediction, and Search*. Number 81 in Lecture Notes in Statistics. Springer Verlag, 1993.
23. Jiří Vomlel. Bayesian networks in educational testing. In *Proceedings of First European Workshop on Probabilistic Graphical Models (PGM'02)*, Cuenca, Spain, November 6-8 2002.
24. Marta Vomlelová and Jiří Vomlel. Troubleshooting: NP-hardness and solution methods. *Soft Computing Journal*, 7(5):357–368, April 2003.
25. Howard Wainer, David Thissen, and Robert J. Mislevy. *Computerized adaptive testing : a primer*. Mahwah, N.J. : Lawrence Erlbaum Associates, second edition, 2000.