0. Úvod – obecná představa

Téměř bez výjimky je základním metodologickým východiskem všech formalismů vytvořených pro zpracování syntaxe přirozeného jazyka na počítači myšlenka, že struktura věty má být prostředky formalismu popisována staticky – popis syntaktické struktury má být prostřednictvím formalismu formulován jako popis pevných, neměnných vlastností, které lze na větě a její struktuře jakožto na objektech lingvistického pozorování spatřit. Tato zásada přitom platí všeobecně, ať už jde o řešení takových úloh počítačového zpracování jazyka jako je analýza vět, generování vět, nebo o jiné aplikace. To je patrný posun od transformačních popisů let šedesátých a sedmdesátých minulého století, ve kterých k popisu struktury věty nedílně patřil popis vzniku této struktury (úkolem syntaxe vlastně bylo především popsat postup vzniku struktury, nikoliv samotnou strukturu, ta byla jen jakýmsi „vedlejším produktem“ syntaktického popisu). Tato změna pohledu na metodologii formalizace syntaxe, tj. přechod od popisů tzv. procedurálních (tranformačních) k popisům tzv. deklarativním (netransformačním), ovšem nenastala pouze v aplikační oblasti, ale je do značné míry posuvem i v oblasti čistě teoretické, o čemž svědčí rozšíření řady lingvistických teorií založených na stejné základní myšlence za posledních zhruba dvacet let (Functional Unification Grammar, Kay 1978; Lexical Functional Grammar, Bresnan (ed.) 1982; Generalized Phrase Structure Grammar, Gazdar, Klein, Pullum & Sag 1985, Head-driven Phrase Structure Grammar, Pollard & Sag 1987, 1994; jakožto i mnohé jejich varianty a další, méně rozšířené teorie).

1. Popis vlastností objektů – jednoduché sestavy rysů

Základem těchto formalismů (a deklarativního přístupu k popisu syntaxe obecně) je tedy popis (statické) syntaktické struktury pomocí informace, kterou o ní na základě pozorování máme. Celková informace se přitom skládá z informací dílčích, a základním kamenem takové informace je popis jedné pozorované vlastnosti – v dalším jej budeme nazývat *rys* (angl. *feature*). Formalizace rysu – (popisu) znalosti jedné takové základní vlastnosti – je přitom rozložena do dvou částí: do (popisu) názvu vlastnosti a do (popisu) hodnoty vlastnosti. Název vlastnosti přitom udává typ informace, hodnota tento typ konkretizuje (viz příklady bezprostředně níže). Jednotlivé rysy, tj. jednotlivou informaci o vlastnosti nějakého objektu (nejtypičtěji pro nás samozřejmě objektu syntaktického, jako je slovo nebo konstrukce), potom schematicky zapisujeme jako

<název_vlastnosti> : <hodnota_vlastnosti>

kde jak **<*název_vlastnosti*>** tak **<*hodnota_vlastnosti*>** jsou metasymboly, které v konkrétním zápise budou nahrazeny konkrétnímí hodnotami. V dalším textu budeme předpokládat, že **<*název_vlastnosti*>** je vždy identifikátor (posloupnost sestávající z písmen, číslic a podtržítek, začínající malým písmenem), **<*hodnota_vlastnosti*>** může být obecně vzata z pestřejšího repertoáru, ale v této chvíli předpokládejme, že je zde možný pouze identifikátor nebo číslo[1] (dále v textu potom možnosti postupně rozšíříme). Kvůli tomuto omezení budeme takové rysy nazývat *jednoduché*.

Příklady rysů – formálně vyjádřených vlastností objektů:

barva	:	žlutá
cena_v_korunách	:	100
slovní_druh	:	zájmeno

[1] dekadický zápis reálného čísla, abychom byli opravdu přesní ...

(Syntaktický) objekt je v takovémto přístupu popsán (reprezentován) jako souhrn rysů, tj. vlastností, které tento objekt má (přesněji řečeno: souhrn informací, které jsou o vlastnostech tohoto objektu známy). Popis objektu budeme tedy formalizovat jako množinu vlastností tohoto objektu. Abychom dali najevo, že se nejedná o "obyčejnou" množinu, ale o popis objektu, budeme psát každou vlastnost na nový řádek a celý zápis uzavřeme do velkých hranatých závorek. Podobně jako při zápisu množiny nebude přitom pořadí zápisu jednotlivých vlastností objektu hrát žádnou roli, tj. dva zápisy, které se liší pouze pořadím zápisu rysů, budeme pokládat za identické. Takový zápis budeme nazývat *sestava rysů* (angl. *feature structure*), v našem případě, kdy jsme zatím zavedli jen jednoduché rysy, jej budeme nazývat *jednoduchá sestava rysů*.

Příklad: morfologické vlastnosti slovního tvar "*knihou*" popíšeme následující jednoduchou sestavou rysů:

$$\begin{bmatrix} \text{grafématický\_zápis : knihou} \\ \text{slovní\_druh : podstatné\_jméno} \\ \text{rod : ženský} \\ \text{číslo : jednotné} \\ \text{pád : 7} \end{bmatrix}$$

Stanovíme si přitom formální omezení, že žádný ***<název_vlastnosti>*** se v sestavě rysů nesmí vyskytnout více než jednou. Takové omezení odpovídá intuitivní představě, že dvojí zápis stejné vlastnosti je zbytečný, pokud by šlo o zápis, který by přiřazoval stejnému názvu vlastnosti stejnou hodnotu této vlastnosti, a že by vedl ke sporu, pokud by stejnému názvu vlastnosti přiřazoval hodnoty různé. Zvláště ve druhém případě je toto omezení velmi důležité, neboť automaticky zabraňuje nekonsistenci popisu vlastností objektu.

Příklad: následující zápis morfologické informace o slovním tvaru *"knize"* není zápisem platné (správně tvořené) sestavy rysů, protože se tomto zápise objevuje dvakrát vlastnost s názvem *"pád",* což není přípustné (odporuje to definici sestavy rysů).

$$\begin{bmatrix} \text{grafématický\_zápis : knize} \\ \text{slovní\_druh : podstatné\_jméno} \\ \text{pád : 3} \\ \text{rod : ženský} \\ \text{pád : 6} \\ \text{číslo : jednotné} \end{bmatrix}$$

Vyloučení zápisů obsahujících stejné jméno vlastnosti vícekrát s sebou ovšem přináší problém, jak zachycovat situace, kdy jeden rys může potenciálně mít několik hodnot (např. v případě víceznačnosti popisovaného objektu, viz minulý příklad). Tento problém vyřešíme níže.

2. Unifikace jednoduchých sestav rysů

Kromě bezprostřední informace o vlastnostech objektů a o objektech jako takových je důležité pracovat i s informací o informaci (rozumí se "s informací o informaci o objektech a jejich vlastnostech"). Nejdůležitější mezi nimi je informace o tom, že dvě informace (tj. dva popisy objektů, dvě sestavy rysů) popisují tentýž objekt. V tom případě je zřejmě rozumné oba příslušné popisy (sestavy rysů) zkombinovat do popisu jediného, který ponese najednou informaci předtím rozdělenou. Takovou operaci budeme nazývat *unifikace* (plným termínem *unifikace sestav rysů*).

Příklad: mějme dvě sestavy rysů

$$\begin{bmatrix} \text{slovní\_druh} : \text{sloveso} \\ \text{osoba} : 3 \\ \text{číslo} : \text{množné} \end{bmatrix} \qquad \begin{bmatrix} \text{rod} : \text{mužský\_životný} \\ \text{číslo} : \text{množné} \end{bmatrix}$$

a současně informaci, že obě tyto sestavy rysů popisují tentýž objekt. Pokud výše zmíněnou operaci unifikace označíme symbolem Y (tj. stejným symbolem jakým značíme sjednocení množin, což není podobnost vůbec náhodná), můžeme pro sestavy rysů v tomto příkladě psát následující rovnost (která by čtenáři měla být zřejmá z analogie mezi unifikací sestav rysů a sjednocením množin).

$$\begin{bmatrix} \text{slovní\_druh} : \text{sloveso} \\ \text{osoba} : 3 \\ \text{číslo} : \text{množné} \end{bmatrix} \mathrm{Y} \begin{bmatrix} \text{rod} : \text{mužský\_životný} \\ \text{číslo} : \text{množné} \end{bmatrix} = \begin{bmatrix} \text{slovní\_druh} : \text{sloveso} \\ \text{osoba} : 3 \\ \text{rod} : \text{mužský\_životný} \\ \text{číslo} : \text{množné} \end{bmatrix}$$

Výsledný objekt může být popisem nějakého slovesného tvaru ve třetí osobě času minulého, čísla množného, rodu mužského životného (např. *“přišli, pracovali, …”*)

Zajímavá (a relevantní) je otázka, jaký má být výsledek unifikace v případě, že sestavy rysů, které do operace vstupují, nesou informace, jež spolu nejsou kompatibilní, tj. v případě, že by unifikací vznikla “formálně nesprávná sestava rysů”[2]. Jedním triviálním příkladem takové dvojice sestav rysů je pár

$$\left[\textit{číslo} : \textit{jednotné} \right] \quad \text{a} \quad \left[\textit{číslo} : \textit{množné} \right]$$

Je ovšem zřejmé, že příklady nekompatibilní informace mohou být i mnohem rozsáhlejší, v podstatě k nekompatibilitě dvou sestav rysů stačí, když se popis jedné jediné vlastnosti vyskytuje v obou sestavách, v každé s jinou hodnotou (bez ohledu na počet a kompatibilitu všech ostatních rysů v sestavách)[3].
V takovém případě by na první pohled výsledek operace zřejmě vůbec neměl být definován (podobně jako např. v aritmetice není definován výsledek dělení nulou), což ostatně odpovídá našemu omezení týkajícímu se počtu jmen vlastností obsažených v jedné sestavě rysů (obecně bychom pro zachycení kombinace informace z obou sestav rysů potřebovali, aby v sestavě výsledné bylo jméno vlastnosti, ve které jsou původní sestavy nekompatibilní, dvakrát, pokaždé s jinou hodnotou).
Běžně se však přijímá poněkud odlišný pohled, který vychází z toho, že každá sestava rysů obsahuje určité množství informace o objektu, který popisuje. Z tohoto hlediska je výsledkem unifikace dvou nekompatibilních sestav rysů (ať už jejich nekompatibilita spočívá v jakémkoliv rysu či v jakýchkoliv rysech a bez ohledu na počet takových nekompatibilních rysů) sestava speciální, která má tu vlastnost, že obsahuje (intuitivně řečeno) “příliš mnoho” informace – totiž tak mnoho, že je to sestava vnitřně sporná (a tím nesprávná). Taková sestava se značí $\perp$ a definitoricky se pokládá za výsledek unifikace ve všech těchto případech.

---

[2] Jinak lze tuto otázku zformulovat takto: jak má vypadat výsledek unifikace v případě, že buď "informace o informaci" nebo původní informace jsou nesprávné ?
[3] Z toho by bylo možné odvodit i definici unifikace rysů: dva rysy lze unifikovat právě tehdy, jsou-li identické (tj. mají-li stejný nazev a stejnou hodnotu). Výsledkem takové unifikace je právě tato hodnota. Běžně se však unifikace rysů nedefinuje, protože definice unifikace sestav rysů je postačující.

Příklad: platí tedy například následující rovnosti

$$[\text{číslo} : \text{jednotné}] \ \mathrm{Y} \ [\text{číslo} : \text{množné}] \ = \begin{bmatrix} \text{slovní\_druh} : \text{sloveso} \\ \text{osoba} : 3 \\ \text{číslo} : \text{jednotné} \end{bmatrix} \mathrm{Y} \begin{bmatrix} \text{rod} : \text{mužský\_životný} \\ \text{číslo} : \text{množné} \end{bmatrix} =$$

$$= \begin{bmatrix} \text{osoba} : 3 \\ \text{číslo} : \text{množné} \end{bmatrix} \mathrm{Y} \begin{bmatrix} \text{slovní\_druh} : \text{zájmeno} \\ \text{osoba} : 2 \\ \text{číslo} : \text{množné} \end{bmatrix} = [\text{barva:černá}] \ \mathrm{Y} \ [\text{barva:bílá}] \ = \ \perp$$

Formálně tedy můžeme definovat unifikaci dvou jednoduchých sestav rysů SR1 a SR2 následujícím způsobem:

nechť SR1 obsahuje rysy r11,r12, …, r1m, po řadě s hodnotami h11, h12,…, h1m,

nechť SR2 obsahuje rysy r21,r22, …, r2n, po řadě s hodnotami h21, h22,…, h2n.

Pak SR1 Y SR2 je definováno jako

a) sestava obsahující rysy r11,r12, …, r1m,r21,r22, …, r2n po odstranění duplicit právě tehdy, když pro všechna $i \in \{1,...,m\}$, $j \in \{1,...,n\}$ platí právě jeden ze vztahů

a1. rys s názvem r1i není obsažen mezi rysy sestavy SR2

a2. rys s názvem r2j není obsažen mezi rysy sestavy SR1

a3. rys r1i je identický (co do názvu i hodnoty) s rysem r2j pro právě jedno j

b) sestava $\perp$ ve ostatních případech (tj. v případech, kdy existují $i,j \in N$ taková, že název rysu r1i je shodný s názvem rysu r2j, ale hodnoty r1i a r2j jsou rozdílné).

Je zřejmé, že tato definice je vlastně jen jednoduchým rozšířením běžné definice sjednocení množin – jediný rozdíl je v tom, že jsou (oproti definici množinového sjednocení) ještě přidány podmínky, které zaručují, že výsledek operace bude platnou sestavou rysů.

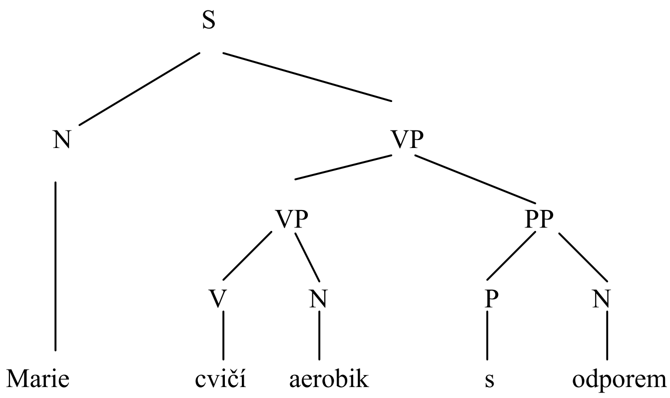
3. Jednoduché unifikační gramatiky

V předchozím textu jsme v podobě sestav rysů a operace unifikace na těchto sestavách vybudovali základní formální aparát pro zápis jednoduchých unifikačních gramatik. Předveďme si nyní na krátkém příkladě, jak unifikační gramatiky vypadají a především to, jaký je jejich vztah k "normálním" bezkontextovým gramatikám.

Mějme jednoduchou bezkontextovou gramatiku, jejíž jazyk obsahuje větu *"Marie cvičí aerobik s odporem"*. Taková gramatika může vypadat následujícím způsobem:

Pravidla:	S → N VP	N → Marie
	VP → V N	V → cvičí
	VP → VP PP	N → aerobik
	PP → P N	P → s
		N → odporem

Taková gramatika přiřadí příkladové větě strukturu

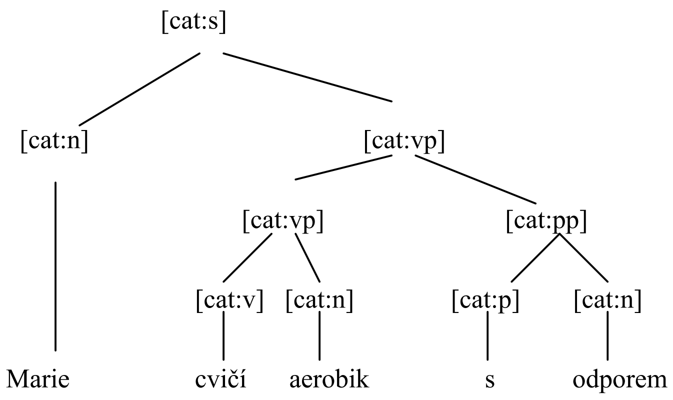


(Je zřejmé, že tato gramatika slouží pouze pro ilustrační účely a rozhodně si nečiní nárok na to, aby byla lingvisticky adekvátním popisem nějaké české věty – kromě jiného proto, že by byla i popisem ne-vět *“Odporem cvičí Marie s aerobik”* atd.)

V pravidlech takové gramatiky se vyskytují dva druhy symbolů: terminály, tj. slova jazyka, a neterminály, které budeme pro účely tohoto příkladu pokládat za “jediné skutečné” syntaktické objekty. Veškerá informace, kterou o těchto objektech máme, je ale velmi skromná – víme pouze, o jaký druh objektu se jedná (že jde o větu, podstatné jméno, slovesnou frázi atd.). Přestože je z tohoto pohledu vše velmi triviální, přepišme nyní tuto gramatiku tak, že místo jednoduchých neterminálních symbolů použijeme sestavy rysů vyjadřující příslušnou informaci o kategorii objektu (kategorii – tj. název vlastnosti – budeme zkracovat obvyklým způsobem jako *cat,* hodnotu kategorie – původní neterminál – budeme psát malým písmenem). Dostaneme tak gramatiku s pravidly v následujícím tvaru:

Pravidla:	[cat:s] →	[cat:n] [cat:vp]	[cat:n] →	Marie
	[cat:vp] →	[cat:v] [cat:n]	[cat:v] →	cvičí
	[cat:vp] →	[cat:vp] [cat:pp]	[cat:n] →	aerobik
	[cat:pp] →	[cat:p] [cat:n]	[cat:p] →	s
			[cat:n] →	odporem

Taková gramatika sice používá sestavy rysů jako neterminálních symbolů, stále však ještě není gramatikou unifikační – operaci unifikace vneseme až do její aplikace, tím, že při přepisování symbolů (rozvíjení neterminálů) nebudeme požadovat striktní identitu přepisovaného a přepisujícího neterminálu (jako je to obvyklé u standardních gramatik), ale pouze stanovíme podmínku, že výsledek unifikace přepisovaného a přepisujícího neterminálu je různý od ⊥ (tj. že tyto dva neterminály jsou unifikovatelné). Všimněmě si, že takovový požadavek je rozumným rozšířením požadavku striktní formální identity neterminálů, neboť unifikace dvou popisů objektů znamená vlastně znalost faktu, že oba tyto popisy se týkají identického objektu (tj. identita, od které bylo formálně, vnějškově ustoupeno, zůstává zachována na intuitivním pozadí). V souvislosti s nahrazením identity neterminálů jejich unifikovatelností zavedeme navíc konvenci, že do stromu odvození budeme vždy zapisovat tu nejpodrobnější informaci, kterou jsme na daném místě schopni zapsat – konkrétně tedy budeme do uzlů stromu odvození vždy zapisovat výsledek unifikace přepisovaného a přepisujícího neterminálu (z hlediska stromové geometrie unifikaci informace přicházející “shora” a informace přicházející “zdola”). Strom odvození příslušný příkladové větě "*Marie cvičí aerobik s odporem"* v takovéto gramatice vypadá takto:



V předchozím triviálním příkladě znamenají přitom identita a unifikace totéž: sestavy rysů tvořící neterminály jsou totiž velmi jednoduché. Tento příklad (tj. gramatiku) nyní rozšíříme tak, aby se rozdíl mezi identitou a unifikací stal zřejmým – rozdíl v tom, že identitu již nebude možné pro odvozování či analýzu řetězců pomocí takto rozšířené gramatiky použít, zatímco s unifikací bude vše "fungovat" správně. Použijeme při tom – opět jen pro ilustrační účely, bez valné lingvistické motivace – následujících faktů:

- podstatné jméno v podmětu a sloveso v přísudku věty se musejí shodovat v čísle – buď musí být jak podmět tak přísudek v singuláru, nebo musejí být oba tyto členy v plurálu
- v oznamovacím způsobu přítomného času (a předpokládejme, že naše gramatika platí jen pro věty, jejichž sloveso splňuje tyto charakteristiky) není potřeba vyjadřovat požadavky na shodu podmětu a přísudku ve jmenném rodě (jako je to např nutné v čase minulém – *"Jan přišeL"* vs. *"Marie přišLA"*)
- tvar *"Marie"* je tvarem prvního pádu podstatného jména ženského rodu, avšak informaci o tom, zda jde o singulár nebo plurál, nám tento tvar nepodává (tj. v zápisu informace o slově *"Marie"* musí chybět rys vyjadřující číslo)[4]
- tvar *"cvičí"* je slovesným tvarem, na němž je patrna pouze osoba (třetí), čas (přítomný) a slovesný rod (činný), nikoliv však číslo ("*cvičí*" je shodně tvarem singuláru i plurálu) či jmenný rod (jako by tomu bylo např. u tvaru *"cvičilo"*)

Když znalost těchto faktů promítneme do neterminálů naší příkladové gramatiky, změní se její tvar na gramatiku s následujícími pravidly. Je velmi důležité si všimnout, že se počet pravidel zvýšil, a to proto, že je potřeba, aby se pravidla popisující strukturu slovesné fráze vyskytovala v gramatice dvakrát – tato pravidla totiž slouží (viz obrázek struktury níže) také k "přenosu" informace o shodě mezi podmětem a přísudkem, je tedy třeba je formulovat separátně pro shodu singulárovou a pro shodu plurálovou[5].

$$[\text{cat:s}] \rightarrow \begin{bmatrix} \text{cat : n} \\ \text{num : sg} \\ \text{case : nom} \end{bmatrix} \begin{bmatrix} \text{cat : vp} \\ \text{num : sg} \end{bmatrix} \qquad\qquad \begin{bmatrix} \text{cat : n} \\ \text{gend : f} \\ \text{case : nom} \end{bmatrix} \rightarrow \text{Marie}$$

$$[\text{cat:s}] \rightarrow \begin{bmatrix} \text{cat : n} \\ \text{num : pl} \\ \text{case : nom} \end{bmatrix} \begin{bmatrix} \text{cat : vp} \\ \text{num : pl} \end{bmatrix} \qquad\qquad \begin{bmatrix} \text{cat : v} \\ \text{per : 3} \\ \text{tns : prs} \\ \text{vce : act} \end{bmatrix} \rightarrow \text{cvičí}$$

[4] Fakt, že tvar *"Marie"* odpovídá i jiným pádům (2.pád jednotného čísla, 4.pád množného čísla, 5.pád obou čísel) zde pro jednoduchost výkladu zanedbáme.

[5] Níže si ukážeme, jak je pomocí pokročilejších technik možné dosáhnout toho, aby byla shoda vyjádřena korektně a přitom počet pravidel nevzrostl. Zatím však takové prostředky nemáme zavedeny a nezbývá tedy, než se pro tuto chvíli smířit s nárůstem počtu pravidel.

$$\begin{bmatrix} \text{cat : vp} \\ \text{num : sg} \end{bmatrix} \rightarrow \begin{bmatrix} \text{cat : v} \\ \text{num : sg} \end{bmatrix} \begin{bmatrix} cat : n \end{bmatrix}$$

$$\begin{bmatrix} \text{cat : vp} \\ \text{num : pl} \end{bmatrix} \rightarrow \begin{bmatrix} \text{cat : v} \\ \text{num : pl} \end{bmatrix} \begin{bmatrix} cat : n \end{bmatrix}$$

[cat:pp] $\rightarrow$ [cat: p] [cat:n]

$$\begin{bmatrix} \text{cat : n} \\ \text{gend : m \_ i} \\ \text{case : acc} \end{bmatrix} \rightarrow \text{aerobik}$$

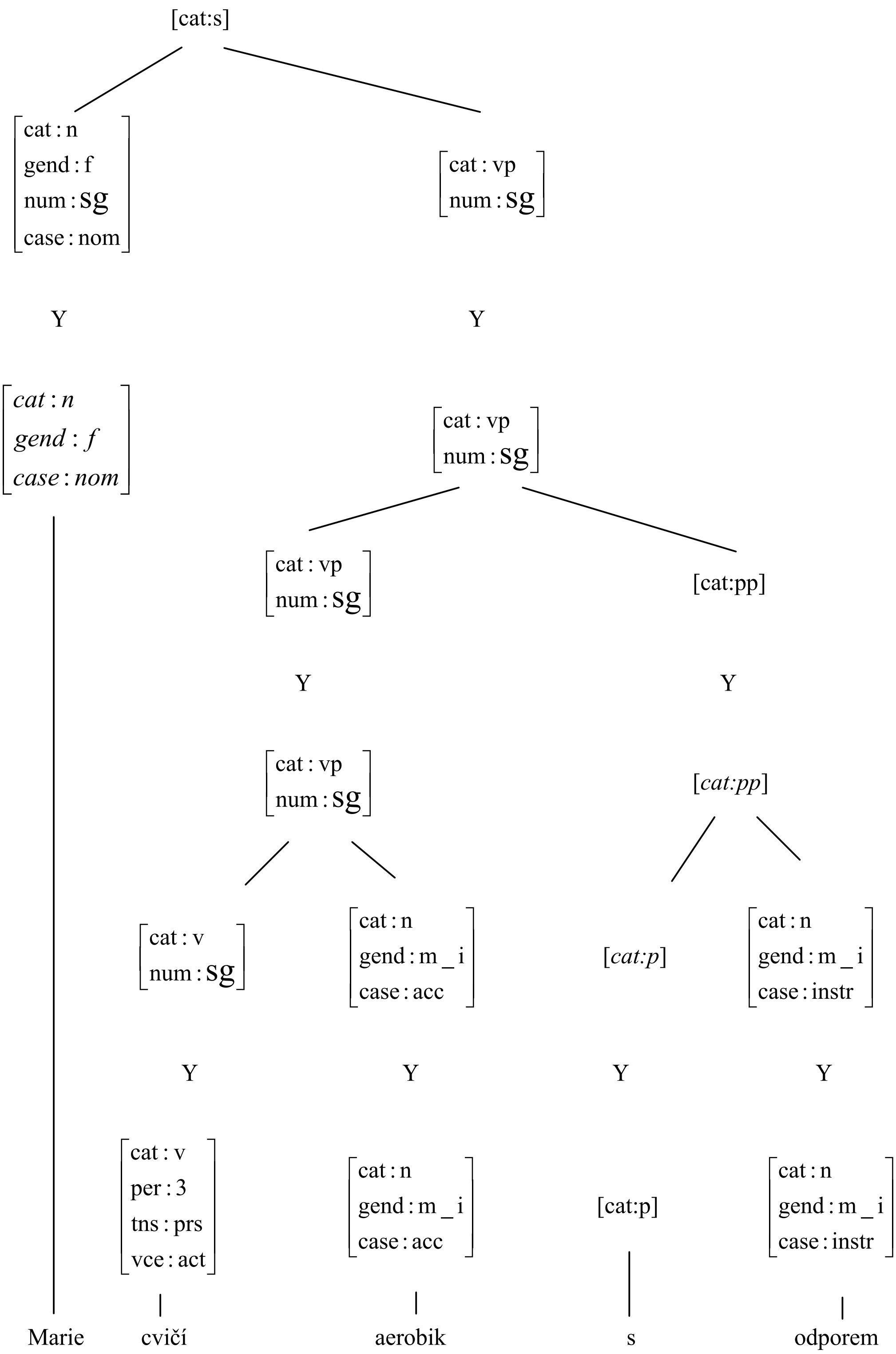
[cat:p] $\rightarrow$ s

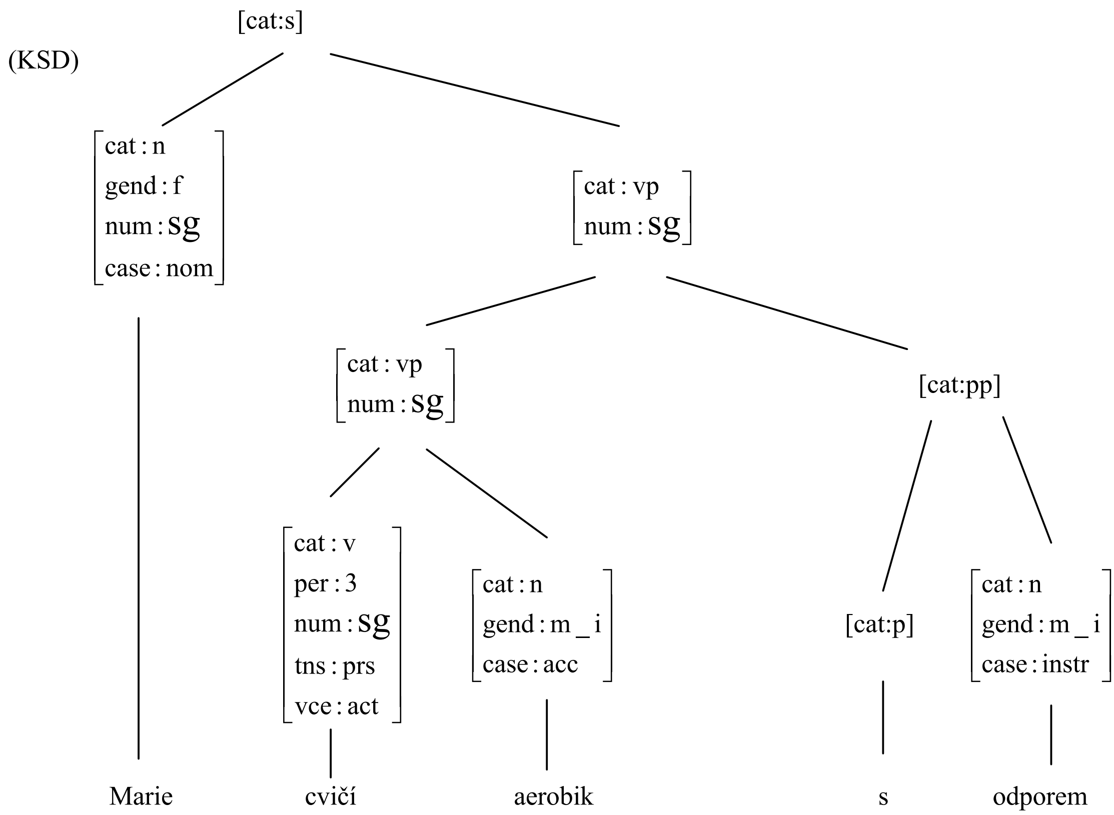
$$\begin{bmatrix} \text{cat : n} \\ \text{gend : m \_ i} \\ \text{case : instr} \end{bmatrix} \rightarrow \text{odporem}$$

Pokud chceme tuto gramatiku aplikovat na příkladovou větu, je okamžitě jasné, že při aplikaci pravidel nevystačíme s identitou symbolů a že je nutné užít unifikaci. Ta nám ovšem do výsledné struktury přinese neterminály, které se v takové formě v původní gramatice nevyskytovaly – totiž neterminály, které jsou co do informace, kterou obsahují, rozšířením (pomocí unifikace) neterminálů z gramatiky. Situace struktury věty *"Marie cvičí aerobik s odporem"* před explicitní unifikací je zachycena v (HER), výsledná struktura (tj. po unifikaci) je vidět v (KSD). V (KSD) je důležité si všimnout preteminálu příslušnému slovu *"cvičí"*, což je právě příklad neterminálu, který se v takové formě nikde v gramatice nevyskytuje (vzniká v této podobě až unifikací).

Za zmínku stojí i fakt, že (HER) a (KSD) představují pouze jednu ze dvou struktur, které je podle gramatiky (GHZ) možno přiřadit příkladové větě. Druhá možná struktura by se od ní lišila v tom, že slovům *"Marie"* a *"cvičí"* by v ní byly jednotně přiřazeny struktury obsahující rys *num:pl* (jde tedy o větu, kde se mluví o více Mariích, které všechny cvičí aerobik s odporem). Je samozřejmě důležité si všimnout toho, že unifikace (spolu s tvarem neterminálů v pravidlech gramatiky) zajišťuje, že nemůže dojít k "překřížení" – k tomu, aby jedno z těchto slov bylo v singuláru a druhé v plurálu; konkrétně je to zajištěno tím, že díky unifikaci užité pří aplikaci pravidel jsou slova *"Marie"* a *"cvičí"* (přesněji řečeno: příslušné preterminály) ve výsledné struktuře spojeny řetězcem neterminálů s identickou hodnotou rysu *num*. (Tato technika zajištění shody, obecně zajištění vztahů mezi "vzdálenými" neterminály, se anglicky nazývá *"feature threading",* český termín zatím neexistuje.)

(HER)


[cat:s]
cat : n
gend : f
num : sg
case : nom
cat : vp
num : sg
Y
Y
cat : n
gend : f
case : nom
cat : vp
num : sg
cat : vp
num : sg
[cat:pp]
Y
Y
cat : vp
num : sg
[cat:pp]
cat : v
num : sg
cat : n
gend : m _ i
case : acc
[cat:p]
cat : n
gend : m _ i
case : instr
Y
Y
Y
Y
cat : v
per : 3
tns : prs
vce : act
cat : n
gend : m _ i
case : acc
[cat:p]
cat : n
gend : m _ i
case : instr
Marie
cvičí
aerobik
s
odporem



4. Zavedení sestavy rysů a proměnné jako možných hodnot rysu

Jako možný typ hodnoty rysu v popisovaném formalismu jsme zatím připustili pouze identifikátory a čísla (viz začátek první kapitoly). Takovým hodnotám říkame atomické; k jejich výhodám patří, že operace se sestavami rysů, jejichž hodnoty jsou takto jednoduché, jsou přehledné a že je možné je také velmi jednoduše implementovat. Z takové jednoduchosti však vyplývá i jeden velmi zásadní nedostatek: pomocí atomických hodnot nelze přehledně popsat takové objekty, jejichž vlastnosti mají vlastní vnitřní strukturu.
Uveďme si nejprve jako odlehčující motivaci triviální příklad zcela mimo oblast lingvistiky. Představme si, že máme sestavou rysů popsat jídelní lístek slavnostního oběda[6]. Takový oběd se typicky skládá z více chodů – předpokládejme, že v našem případě to budou předkrm, polévka, hlavní chod a zákusek. Předkrm bude chřestový salát s bylinkovou zálivkou a obzvláště křupavými rohlíčky z pekárny *"U císařovy pochoutky"*, polévka bude sýrová, jako hlavní jídlo zvolíme toufu à la bažant s rýží a jako desert krém z mascarpone šlehaného s borůvkami. Pokud budeme chtít takový jídelní lístek popsat "přirozeným" způsobem pomocí sestavy rysů, zřejmě nevystačíme s atomickými hodnotami. Na základní úrovni totiž bude zřejmě vhodné uvést jen rysy *předkrm, polévka, hlavní_jídlo* a *zákusek*, nikoliv už ale jakékoliv další "vlastnosti" jídelního lístku. Přitom ovšem např. rys *předkrm* má (v našem příkladě) hodnotu, která by měla být strukturována – předkrm se skládá ze salátu a přílohy, a dokonce i tento salát je "dále vnitřně strukturován" – skládá se z chřestu a zálivky. Prostředky na vnitřní strukturaci hodnot nám přitom dosavadní formalismus, připouštějící pouze hodnoty atomické, nedává. Proto je vhodné tento formalismus rozšířit, konkrétně (v tuto chvíli) tak, že jako hodnotu rysu (tj. výraz stojící napravo od dvojtečky v zápisu ***<název_vlastnosti>*** *:* ***<hodnota_vlastnosti>***) povolíme kromě identifikátorů (posloupností písmen, číslic a podtržítek, začínajících malým písmenem) a čísel také sestavy rysů. Zápis jídelního lístku z příkladu tedy bude nyní moci vypadat jako následující (správně tvořená) sestava rysů:

[6] Například pohoštění pro autory skript, z nichž jsme studovali na úspěšnou zkoušku.

$$\left[\begin{array}{l} \text{predkrm}: \left[\begin{array}{l} \text{salát}: \left[\begin{array}{l} \text{základ}:\text{chrest} \\ \text{zálivka}:\text{bylinná} \end{array}\right] \\ \text{príloha}: \left[\begin{array}{l} \text{typ}:\text{rohlícky} \\ \text{jakost}:\text{obzvláště\_krupavé} \\ \text{dodavatel}:\text{pekárna\_U\_císarovy\_pochoutky} \end{array}\right] \end{array}\right] \\ \text{polévka}:\text{sýrová} \\ \text{hlavní\_chod}: \left[\begin{array}{l} \text{pokrm}: \left[\begin{array}{l} \text{základ}:\text{toufu} \\ \text{úprava}:\text{a\_la\_bažant} \end{array}\right] \\ \text{príloha}:\text{rýže} \end{array}\right] \\ \text{desert}: \left[\begin{array}{l} \text{základ}:\text{mascarpone} \\ \text{úprava}:\text{šlehané} \\ \text{dodatková\_surovina}:\text{boruvky} \end{array}\right] \end{array}\right]$$

Je samozřejmé, že takové rozšíření sestav rysů nebylo zavedeno kvůli popisu jídelních lístků, zároveň je však právě na takovém – jinak profánním – příkladě dobře patrné, jaký je skutečný účel toho, aby byly jako hodnoty připuštěny i sestavy rysů: tímto účelem je těsně spolu sdružit ty rysy, které k sobě z nějakého důvodu patří, a moci takovouto související skupinu rysů i v popisu pojímat jako samostanou jednotku, oddělenou od rysů ostatních.

Konkrétním důvodem, proč sdružovat rysy v syntaktickém popisu, je nejčastěji ten fakt, že několik rysů spolu popisuje určitý jazykový jev. Typickým lingvistickým jevem, který téměř volá po takové sdružením rysů, je tak např. shoda mezi podmětem a přísudkem – takovou shodu na jedné straně pokládáme za jediný jazykový jev, který ale na straně druhé zasahuje více rysů: podmět a přísudek se (v češtině) shodují v osobě, čísle a (jmenném) rodě[7]. Např. ve větě "*Já, děvenka moje starostlivá, jsem se vypravila do města pro trochu toho šafránu*" se zájmeno *"já"* stojící v (holém) podmětu a slovesný tvar *"jsem se vypravila"* tvořící (holý) přísudek shodují v osobě (ta je u obou *první*), čísle (*singulár*) a jmenném rodě (*femininum*).

Nazveme-li sdružený rys pro shodu podmětu s přísudkem *s_v_agr* (z anglického *subject-verb agreement*), mohla by gramatika pro věty *"Josef cvičil aerobik s nadšením"* a *"Josefové cvičili aerobik s nadšením"* (které jsou, jak patrno, variacemi na příkladovou větu minulou, zvolenými tak, aby "důležitá" slova v nich obsažená nesla morfologicky pokud možno jednoznačné rysy) vypadat následujícím způsobem[8]:

$$[\text{cat:s}] \rightarrow \left[\begin{array}{l} \text{cat}:\text{n} \\ \text{s\_v\_agr}: \left[\begin{array}{l} \text{num}:\text{sg} \\ \text{gend}:\text{m\_a} \\ \text{per}:3 \end{array}\right] \\ \text{case}:\text{nom} \end{array}\right] \left[\begin{array}{l} \text{cat}:\text{vp} \\ \text{s\_v\_agr}: \left[\begin{array}{l} \text{num}:\text{sg} \\ \text{gend}:\text{m\_a} \\ \text{per}:3 \end{array}\right] \end{array}\right]$$

[7] Že se jedná o rod jmenný, je potřeba zdůraznit proto, aby byl odlišen od rodu slovesného (aktivum vs. pasívum).

[8] Jak vidno, není jen tato věta variací na větu předchozí, ale je – a to především – tato gramatika variací na gramatiku předchozí.

$$[\text{cat:s}] \rightarrow \begin{bmatrix} \text{cat : n} & \\ \text{s\_v\_agr :} & \begin{bmatrix} \text{num : pl} \\ \text{gend : m\_a} \\ \text{per : 3} \end{bmatrix} \\ \text{case : nom} & \end{bmatrix} \begin{bmatrix} \text{cat : vp} & \\ \text{s\_v\_agr :} & \begin{bmatrix} \text{num : pl} \\ \text{gend : m\_a} \\ \text{per : 3} \end{bmatrix} \end{bmatrix}$$

$$\begin{bmatrix} \text{cat : vp} & \\ \text{s\_v\_agr :} & \begin{bmatrix} \text{num : sg} \\ \text{gend : m\_a} \\ \text{per : 3} \end{bmatrix} \end{bmatrix} \rightarrow \begin{bmatrix} \text{cat : v} & \\ \text{s\_v\_agr :} & \begin{bmatrix} \text{num : sg} \\ \text{gend : m\_a} \\ \text{per : 3} \end{bmatrix} \end{bmatrix} [\text{cat : n}]$$

$$\begin{bmatrix} \text{cat : vp} & \\ \text{s\_v\_agr :} & \begin{bmatrix} \text{num : pl} \\ \text{gend : m\_a} \\ \text{per : 3} \end{bmatrix} \end{bmatrix} \rightarrow \begin{bmatrix} \text{cat : v} & \\ \text{s\_v\_agr :} & \begin{bmatrix} \text{num : pl} \\ \text{gend : m\_a} \\ \text{per : 3} \end{bmatrix} \end{bmatrix} [\text{cat : n}]$$

$$[\text{cat:pp}] \rightarrow [\text{cat: p}] \quad [\text{cat:n}]$$

$$\begin{bmatrix} \text{cat : n} & \\ \text{s\_v\_agr :} & \begin{bmatrix} \text{num : sg} \\ \text{gend : m\_a} \\ \text{per : 3} \end{bmatrix} \\ \text{case : nom} & \end{bmatrix} \rightarrow \text{Josef} \qquad \begin{bmatrix} \text{cat : n} & \\ \text{s\_v\_agr :} & \begin{bmatrix} \text{num : pl} \\ \text{gend : m\_a} \\ \text{per : 3} \end{bmatrix} \\ \text{case : nom} & \end{bmatrix} \rightarrow \text{Josefové}$$

$$\begin{bmatrix} \text{cat : v} & \\ \text{s\_v\_agr :} & \begin{bmatrix} \text{num : sg} \\ \text{gend : m\_a} \\ \text{per : 3} \end{bmatrix} \\ \text{tns : past} & \\ \text{vce : act} & \end{bmatrix} \rightarrow \text{cvičil} \qquad \begin{bmatrix} \text{cat : v} & \\ \text{s\_v\_agr :} & \begin{bmatrix} \text{num : pl} \\ \text{gend : m\_a} \\ \text{per : 3} \end{bmatrix} \\ \text{tns : past} & \\ \text{vce : act} & \end{bmatrix} \rightarrow \text{cvičili}$$

$$\begin{bmatrix} \text{cat : n} \\ \text{gend : m\_i} \\ \text{case : acc} \end{bmatrix} \rightarrow \text{aerobik} \qquad [\text{cat:p}] \rightarrow \text{s} \qquad \begin{bmatrix} \text{cat : n} \\ \text{gend : m\_i} \\ \text{case : instr} \end{bmatrix} \rightarrow \text{odporem}$$

Tato gramatika je zapsána sice tak, že rysy a atomickými hodnotami relevantními pro shodu podmětu s přísudkem jsou v ní sdruženy do rysu s komplexní hodnotou, avšak přínos takového zápisu je – zdá se alespoň – nulový či přímo záporný. Neušetřili jsme si totiž žádné psaní, naopak, museli jsme psát ještě více. Především se ale nepovedlo vyjádřit lingvistický fakt, že podmět a přísudek se shodují v patřičných kategoriích bez toho, abychom tyto hodnoty museli vždy explicitně vypisovat, což má za následek, že oproti původní gramatice s jednoduchými symboly (tj. gramatice bez použití sestav rysů) vzrostl počet pravidel.

Pokud si předchozí odstavec promyslíme, zjistíme, že problém je zřejmě v tom, že formalismus nám umožňuje používat pouze konstanty – proto musíme hodnoty vždy vypisovat (bez ohledu na to, zda jde o hodnoty atomické nebo komplexní). Např. pokud chceme, aby byly hodnoty čísla na dvou místech pravidla stejné (tj. aby se např. podmět a přísudek spolu v čísle shodovaly), musíme napsat pravidla dvě, jedno, kde na obou místech uvedeme singulár, a druhé, kde na obou místech uvedeme plurál. Proto, abychom si tuto práci mohli ušetřit, a především proto, abychom byli schopni vyjádřit relevantní lingvistické zobecnění, že podmět a přísudek se (prostě) shodují v osobě, čísle a jemnném rodě, ať už jsou hodnoty těchto rysů jakékoliv, zavedeme další možný typ hodnoty rysu: proměnnou. Značit budeme takovou hodnotu jako (malé) přirozené číslo uzavřené mezi svislé čáry zleva a zprava, např. tedy |*1*|, |*2*|, |*15*| jsou proměnné[9].

Dva výskyty (příp. více výskytů) téže proměnné na místě hodnot dvou rysů v jedné sestavě rysů nebo v jednom pravidle nazýváme koindexací těchto rysů (přesněji koindexací jejich hodnot). Koindexaci interpretujeme tak, že (možná) neznáme, jaké konkrétní hodnoty tyto dva rysy mají, ale víme, že jejich hodnoty jsou shodné (používáme proměnnou tedy zcela standardně jako kdekoliv jinde v matematice). Důležité je si uvědomit, že právě uvedená koindexace je zřejmým příkladem "informace o informaci" (možná nevíme, jaká konkrétní informace na těch dvou místech stojí, víme ale, že na obou místech je to informace identická). V tomto smyslu je dvojí (či vícenásobný) výskyt proměnné ve struktuře či pravidle "naznačenou" či "skrytou" unifikací[10] informací stojících jako hodnoty na příslušných místech.

Důležité přitom je, že zavedení proměnných jako hodnot rysů umožňuje vyjadřovat identitu rysů bez toho, že bychom museli vypisovat konkrétní hodnoty rysů (konstanty). To potom dovoluje následující zápis gramatiky, kterou lze vytvořit všechny naše předchozí věty o *"Josefovi/Josefech"* a *"Marii/Mariích"*. Tato gramatika má stejný počet pravidel (pokud v gramatikách nepočítáme pravidla přepisující preterminály na neterminály) jako původní gramatika s jednoduchými neterminálními symboly, na rozdíl od této původní gramatiky však vyjadřuje i shodu podmětu s přísudkem (uvádíme jen pravidla neobsahující terminální symboly – slova). Důležité je uvědomit si při čtení této gramatiky fakt, že každá proměnná je lokální v jednom pravidle, tj. že |1| kondexující hodnoty shody v (XXX)a je "zcela jiná" než |1| kondexující hodnoty shody v (XXX)b.

(XXX) a. $$[\mathrm{cat{:}s}] \rightarrow \begin{bmatrix} \mathrm{cat: n} \\ \mathrm{s\_v\_agr}: |1| \\ \mathrm{case: nom} \end{bmatrix} \begin{bmatrix} \mathrm{cat: vp} \\ \mathrm{s\_v\_agr}: |1| \end{bmatrix}$$

b. $$\begin{bmatrix} \mathrm{cat: vp} \\ \mathrm{s\_v\_agr}: |1| \end{bmatrix} \rightarrow \begin{bmatrix} \mathrm{cat: v} \\ \mathrm{s\_v\_agr}: |1| \end{bmatrix} [\mathrm{cat: n}]$$

c. [cat:pp] → [cat: p] [cat:n]

Výšeuvedené rozšíření typů hodnot rysů si automaticky vynutí i některé změny (rozšíření) další.

První z nich se týká případu, kdy pro to, abychom vyjádřili identitu rysů, chceme jako hodnotu určitého rysu použít proměnnou, a současně přitom ale víme, o jakou hodnotu se konkrétně jedná. Situaci můžeme ilustrovat na příkladu rozlišení mezi tvary českých sloves v aktivu minulého času. Představme si, že chceme vytvořit gramatiku pro popis následujících tří vět:

(ZZZ) a. Já jsem cvičil aerobik s odporem.
b. Ty jsi cvičil aerobik s odporem.
c. Josef cvičil aerobik s odporem.

[9] V literatuře se běžně používá notace, kdy je číslo uzavřeno v "krabičce". My jsme tuto notaci nezavedli jen proto, že je typograficky náročnější.

[10] Dvojí výskyt proměnné je tedy – v tomto ohledu – analogický psaní zlomků: dva výskyty jedné proměnné je "naznačená unifikace", zlomek je (jak známo ze základní školy) "naznačené dělení".

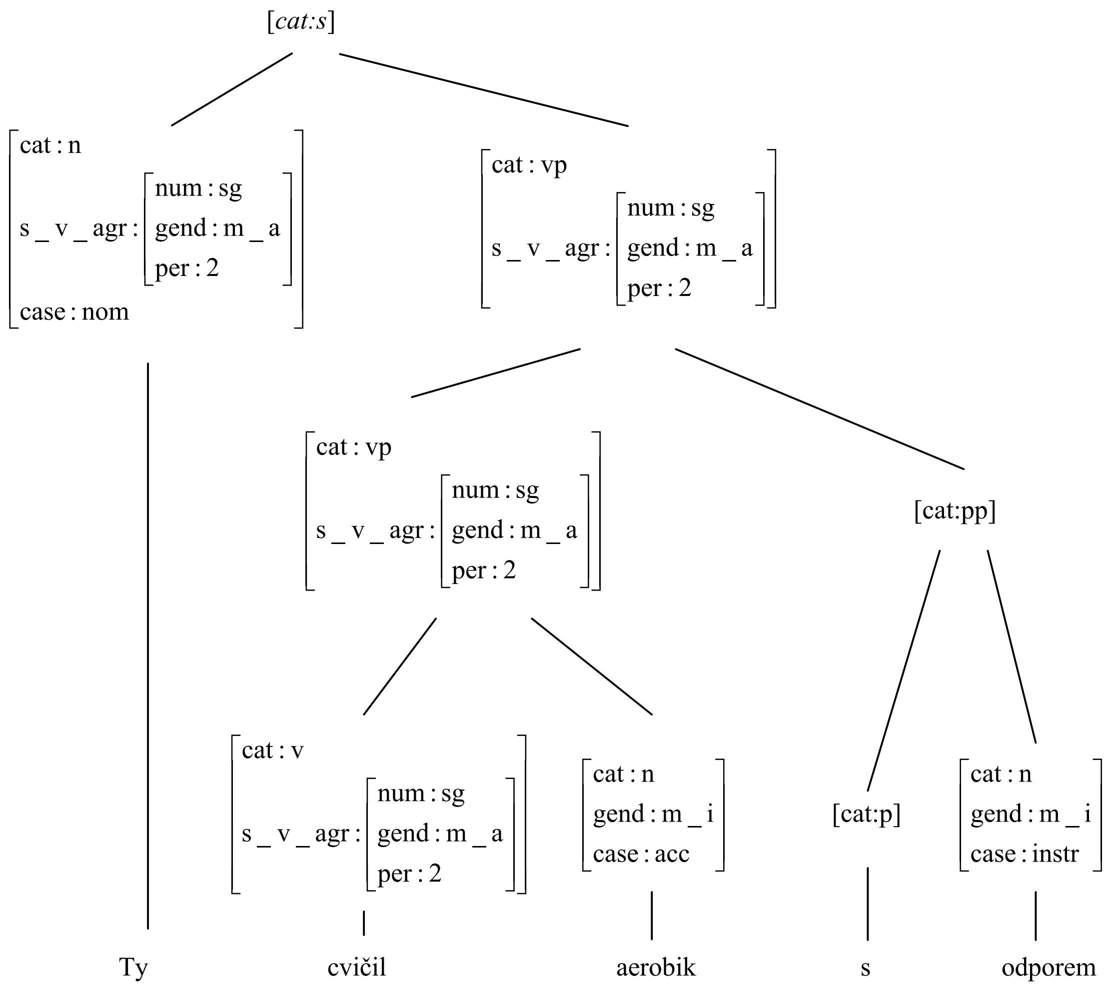
Zřejmě nemůžeme pro tento účel převzít beze změn gramatiku předchozí. Zejména je problematické preterminální pravidlo, které definuje slovo *cvičil* jako třetí osobu. Pokud bychom měli pro slovo *cvičil* jen toto preterminální pravidlo, mohli bychom popsat vlastně jen třetí z těchto vět – větu, kde je slovesný tvar ve třetí osobě. Musíme tedy nutně do gramatiky přidat i následující pravidla, která definují slovo *cvičil* jako součást tvaru první a druhé osoby.

$$\begin{bmatrix} cat : v \\ s\_v\_agr : \begin{bmatrix} num : sg \\ gend : m\_a \\ per : 1 \end{bmatrix} \\ tns : past \\ vce : act \end{bmatrix} \rightarrow \text{cvičil} \qquad \begin{bmatrix} cat : v \\ s\_v\_agr : \begin{bmatrix} num : sg \\ gend : m\_a \\ per : 2 \end{bmatrix} \\ tns : past \\ vce : act \end{bmatrix} \rightarrow \text{cvičil}$$

To však samozřejmě není vše – pro popis slovesných tvarů osoby první a druhé, které obsahují i pomocné sloveso, musíme do takové gramatiky ještě přidat pravidlo (např.) ve tvaru

$$\text{(YYY)} \quad \begin{bmatrix} \text{cat : vp} \\ \text{s\_v\_agr} : |1| \end{bmatrix} \rightarrow \begin{bmatrix} \text{cat : aux} \\ \text{s\_v\_agr} : |1| \end{bmatrix} \begin{bmatrix} \text{cat : v} \\ \text{s\_v\_agr} : |1| \end{bmatrix} \quad \begin{bmatrix} \text{cat : n} \end{bmatrix}$$

Zavedením takového pravidla do gramatiky nastává situace, kdy máme dvě pravidla se stejnou levou stranou (VP) – pravidlo (YYY) a druhé pravidlo (XXX)b. To není nic neobvyklého samo o sobě, důležitý však je v tomto konkrétním případě fakt, že (alespoň pro popis vět ZZZ) u pravidla (XXX)b vlastně máme informaci navíc, totiž informaci o tom, že je to pravidlo popisující věty s podmětem ve třetí osobě (pravidlo neobsahuje pomocné sloveso, které ve třetí osobě minulého času sloves v aktivu indikativu není součástí slovesného tvaru). Tuto informaci jsme přitom nevyjádřili explicitně, ale jen skrytě, nepřítomností preterminálu pomocného slovesa na pravé straně pravidla. Tím jsme se "provinili proti lingvistice", což má za následek, že bychom takovou gramatikou připisovali strukturu i nesprávně tvořeným řetězcům (nesprávným větám, ne-větám), např. řetězci *Ty cvičil aerobik s odporem* bychom podle takové gramatiky přiřadili následující strukturu.



Taková struktura je ovšem zřejmě nesprávná, neboť tvar druhé osoby minulého času slovesa "*cvičit"* musí obsahovat i příslušný tvar pomocného slovesa "*být"* – problém je tedy v tom, že naše gramatika v současné podobě takovou podobu druhé osoby nevynucuje. Je tedy potřeba tuto gramatiku příslušně upravit. Jednoduchou možností by bylo postulovat, že v pravidle (XXX)b, ve kterém se pomocné sloveso nevyskytuje, se jedná o tvar slovesa ve třetí osobě, zatímco v pravidle (YYY) musí vždy jít o osobu první nebo druhou. V současné podobě formalismu však nemáme prostředky, jak vyjádřit současně koindexaci dvou hodnot a znalost (alespoň částečnou) jedné z těchto hodnot. Proto je nyní zavedeme.

Jako hodnotu rysu budeme od nynějška připouštět i dvojici tvořenou proměnnou a atomickou hodnotou nebo proměnnou a sestavou rysů. Následující dva příklady rysů jsou tedy správně tvořeny:

$$\text{rod : |1| femininum} \qquad\qquad \text{agr: |258|}\begin{bmatrix}\text{num : sg}\\ \text{per : 2}\end{bmatrix}$$

Kromě těchto "běžných" případů připustíme i možnost, aby proměnných stojících před atomickou nebo komplexní hodnotou bylo více, připustíme tedy jako správně tvořený např. i rys

$$\text{agr: |1| |2| |3|}\begin{bmatrix}\text{num : sg}\\ \text{gend : m \_ a}\\ \text{per : 2}\end{bmatrix}$$

Výsledný tvar pravidla (XXX)b po takové úpravě vypadá takto:

$$\text{(XXX)} \qquad \text{b.} \begin{bmatrix} \text{cat : vp} \\ \text{s\_v\_agr} : \boxed{1} \end{bmatrix} \rightarrow \begin{bmatrix} \text{cat : v} \\ \text{s\_v\_agr} : \boxed{1}\,[\text{osoba : 3}] \end{bmatrix} \ [\text{cat : n}]$$

Samozřejmě bychom měli analogicky upravit i pravidlo (YYY), avšak vzhledem k tomu, že nemáme prozatím k dispozici formální prostředky, jak vyjádřit, že v tomto případě se může jednat buď o první nebo o druhou osobu, nebudeme upravenou formu pravidla (YYY) uvádět (také proto, že pro ilustraci zápisu proměnných s hodnotou postačuje již upravená verze pravidla (XXX)b, spolu s ostaními příklady). Úmyslně, pro jednoduchost zde pomíjíme jinak důležitý jazykový fakt, že shoda mezi podmětem a přísudkem je v těchto případech vlastně poněkud složitější: podmět se shoduje v osobě a čísle s pomocným slovesem – zde je značíme kategorií *aux* – zatímco s minulým příčestím se shoduje v čísle a jmenném rodě.

Po takovémto rozšíření oboru hodnot rysů (tj. množiny, jejíž prvky se mohou stát ***<hodnotou_vlastnosti>***) musíme samozřejmě rozšířit i definici unifikace tak, aby pokryla i případy, kdy hodnotou rysu je sestava rysů nebo proměnná (zatím jsme definovali unifikaci pouze pro případy, kdy hodnotami rysů byly konstanty – atomy nebo čísla). Takové rozšíření je sice "nasnadě", ale přesto je potřeba je fomálně vyjádřit.
Unifikaci dvou sestav rysů SR1 a SR2 budeme nyní definovat takto:

- nechť SR1 obsahuje rysy r11,r12, …, r1m, po řadě s hodnotami h11, h12,…, h1m,
- nechť SR2 obsahuje rysy r21,r22, …, r2n, po řadě s hodnotami h21, h22,…, h2n.

Pak SR1 Y SR2 je definováno jako

a) sestava obsahující rysy r11,r12, …, r1m,r21,r22, …, r2n po odstranění duplicit právě tehdy, když pro všechna $i \in \{1,...,m\}$, $j \in \{1,...,n\}$ platí právě jeden ze vztahů

a1. rys s názvem r1i není obsažen mezi rysy sestavy SR2

a2. rys s názvem r2j není obsažen mezi rysy sestavy SR1

a3. název rysu r1i je identický s názvem rysu r2j pro právě jedno j, přičemž hodnota rysu r1i je h1i a hodnota rysu r2j je h2j a platí

a31. h1i je atomická hodnota AH nebo dvojice tvořená proměnnou P1 a atomickou hodnotou AH, a zároveň h2j je atomická hodnota AH nebo dvojice tvořená proměnnou P2 a atomickou hodnotou AH; v takovém případě je hodnota příslušného rysu ve výsledné struktuře rovna:

- AH v případě bez proměnných
- P1 AH v případě s proměnnou pouze v h1i
- P2 AH v případě s proměnnou pouze v h2j
- P1 P2 AH v případě přítomnosti proměnných v h1i i h2j

a32. h1i je sestava rysů HSR1 nebo dvojice tvořená proměnnou P1 a sestavou rysů HSR1, a zároveň h2j je sestava rysů HSR2 nebo dvojice tvořená proměnnou P2 a sestavou rysů HSR2; v takovém případě je hodnota příslušného rysu ve výsledné struktuře rovna[11]:

[11] Uzávorkováním je naznačeno, že se proměnné vztahují na výsledek unifikace; formálně vzato je však takové uzávorkování zbytečné, protože by se proměnná/proměnné na výsledek nakonec vztahovaly i v případě, že by byla/byly zpočátku chápány jen jako proměnná/proměnné pro první (levou) z unifikovaných sestav.

- HSR1 Y HSR2 v případě bez proměnných
- P1 (HSR1 Y HSR2) v případě s proměnnou pouze v h1i
- P2 (HSR1 Y HSR2) v případě s proměnnou pouze v h2j
- P1 P2 (HSR1 Y HSR2) v případě přítomnosti proměnných v h1i i h2j

pokud je unifikace SR1 Y SR2 různá od sestavy ⊥

a33. h1i je volná proměnná a h2j je volná proměnná; v tomto případě budou h1i a h2j nadále pokládány za jedinou proměnnou (h1i = h2j)

(Je potřeba přitom vzít v úvahu, že vázání proměnné nemusí být vyznačeno přímo v místě, kde je zapsán příslušný rys – proměnná je vázána na hodnotu libovolným výskytem ve struktuře rysů, přesněji tedy všemi svými výskyty ve struktuře rysů. To může konkrétně znamenat, že před vlastní unifikací s jinou sestavou je potřeba provést unifikaci všech hodnot, k nimž je vázána jedna proměnná !)

b) sestava ⊥ ve ostatních případech (tj. v případech, kdy existují i,j ∈ N taková, že název rysu r1i je shodný s názvem rysu r2j, ale hodnoty r1i a r2j nejsou unifikovatelné podle bodu a)).

Je zřejmé, že tato definice je stále jen rozšířením definice sjednocení množin, i když už je rozšířením netriviálním: vlastně se jedná o "rekurzivní sjednocení" rozšířené o podmínky, které zaručují, že výsledek operace bude platnou sestavou rysů.

Dále je potřeba zmínit, že i nadále sice samozřejmě zůstává platit omezení, že **<*název_vlastnosti*>** musí být identifikátor, ale je nutné nyní přesněji formulovat omezení na výskyt identických názvů vlastnosti. Obecně totiž nyní může nastat, že jeden název vlastnosti se může vyskytnout v platné komplexní sestavě rysů víckrát, vždy jako nejvíce vnější název vlastnosti v určité (pod)sestavě rysů. Jako ještě jeden nelingvistický příklad si uveďme ve formalismu sestav rysů zapsanou personálně-organizační strukturu nějakého koncernu.

$$\begin{bmatrix} \text{ředitel} : \text{jan\_novák} \\ \text{první\_závod} : \begin{bmatrix} \text{ředitel} : \text{honza\_novák} \\ \text{první\_továrna} : \begin{bmatrix} \text{ředitel} : \text{honzík\_novák} \end{bmatrix} \\ \text{druhá\_továrna} : \begin{bmatrix} \text{ředitel} : \text{honcek\_nováků} \end{bmatrix} \end{bmatrix} \\ \text{druhý\_závod} : \begin{bmatrix} \text{ředitel} : \text{jenda\_novák} \\ \text{první\_továrna} : \begin{bmatrix} \text{ředitel} : \text{jeník\_novák} \end{bmatrix} \\ \text{druhá\_továrna} : \begin{bmatrix} \text{ředitel} : \text{janek\_novákojc} \end{bmatrix} \end{bmatrix} \\ \text{samostatný\_zahr\_dodavatel} : \begin{bmatrix} \text{ředitel} : \text{john\_newman} \end{bmatrix} \end{bmatrix}$$

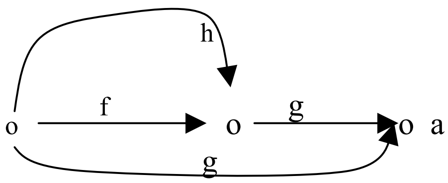
Jak je z příkladu zřejmé, vyskytují se názvy vlastností *ředitel, první_továrna* a *druhá_továrna* v této komplexní struktuře vícekrát, ke konfliktu však zřejmě nedochází, a to proto, že každý takový název vlastnosti je jediný (unikátní) ve své doméně platnosti, kterou je nejbližší sestava rysů názvu vlastnosti bezprostředně nadřazená ("nejbližší nadřazené hranaté závorky"). Omezení výskytu názvů vlastností je tedy potřeba přeformulovat tak, že žádný **<*název_vlastnosti*>** se v sestavě rysů nesmí vyskytnout více než jednou ve stejné doméně platnosti. (Toto znění je přitom zřejmě konzervativním rozšířením formulace minulé, která byla speciálním případem pro situaci, kdy existuje pouze jediná doména platnosti).

Pro zachycení informací o objektech, které jsme formalizovaně zapisovali jako sestavy rysů, je možné použít i notaci odlišnou[12] – notaci v podobě orientovaných acyklických grafů (Directed Acyclic Graphs, odtud zkratka

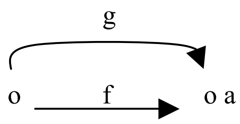
[12] O případech, kdy takovou notaci naopak ("výjimečně") použít nelze, se zmíníme bezprostředně níže.

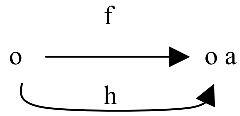
DAG, kterou budeme dále používat). DAG je definován jako konečný souvislý orientovaný graf neobsahující orientovanou kružnici (tj. orientovanou (pod)cestu složenou z hran, která by se vracela do svého výchozího bodu). Z této definice mj. plyne důležitá vlastnost, že každý neprázdný DAG má právě jeden význačný uzel takový, že do něj žádná hrana nevstupuje – takový uzel budeme nazývat kořen DAGu. Každý uzel DAGu přitom reprezentuje hodnotu nějakého rysu, každá hrana reprezentuje název nějakého rysu.
Je-li hodnota rysu atom, je tento atom připsán přímo tomuto uzlu (uzel je touto hodnotou ohodnocen); v tomto případě z takového uzlu zřejmě nemohou vycházet žádné hrany. Je-li hodnota rysu sestava rysů, pokládáme za příslušnou hodnotu (pod)DAG s kořenem v tomto uzlu (a uzel sám žádné ohodnocení konstantou tedy nenese). Např. hodnota kořenu DAGu je tedy celý DAG.
Je-li hodnota rysu volná proměnná (tj. o vlastnostech hodnoty rysu nic nevíme), nenese příslušný koncový uzel žádné ohodnocení[13]. Koindexace pomocí proměnných v DAGu však nejsou potřeba, protože koindexaci dvou (či více) hodnot lze vyjádřit tak, že příslušné orientované hrany (reprezentující jména koindexovaných rysů) vcházejí do stejného uzlu.

Příklad: sestavu rysů

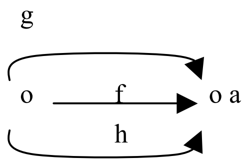
$\begin{bmatrix} f : \boxed{2}\begin{bmatrix} g : \boxed{1}\, a \end{bmatrix} \\ g : \boxed{1} \\ h : \boxed{2} \end{bmatrix}$ lze vyjádřit alternativně jako DAG 

Velmi názorná (a konceptuálně jednoduchá) je operace unifikace DAGů: vzhledem k tomu, že DAGy neleží v centru naší pozornosti, ji nebudeme definovat formálně, ale přiblížíme ji tím, že ji připodobníme k "položení dvou DAGů na sebe" (např. na průhledné folii). Velmi jednoduchý příklad je uveden na následující sérii obrázků.

První DAG: 

druhý DAG: 

Výsledný DAG (po unifikaci ~ překrytí):



Platí dále, že každý DAG lze vyjádřit jako sestavu rysů jednoduchým postupem – každý uzel bude odpovídat buď právě jednomu atomu (tomu, kterým byl ohodnocen) nebo právě jedné "hranaté závorce" (představující sestavu rysů odpovídající DAGu, kterým byl tento uzel ohodnocen), každá hrana bude odpovídat tomu názvu rysu, kterým byla ohodnocena. Pokud do jednoho uzlu vstupuje více hran, je takovou situaci potřeba vyřešit pomocí proměnných (koindexací). Právě kvůli možným alternativním koindexacím není takové vyjádření DAGu pomocí sestavy rysů jednoznačné, což je zřetelná výhoda DAGů oproti sestavám rysů – DAGy jednoznačné jsou, tj. popis infromace pomocí DAGu je jednoznačný, popis téže informace pomocí sestavy rysů v těch případech, kdy jsou koindexovány částečně známé "podsestavy", jednoznačný není (všechny možné popisy však jsou co do zachycené informace ekvivalentní). Příklad na "jednotný" DAG a tři různé sestavy, jež mu odpovídají, je uveden v následujícím.

[13] Je ovšem otázka, proč se takový rys v sestavě rysů vůbec vyskytuje – nenese totiž naprosto žádnou informaci.

Příklad: DAG: $\begin{tikzcd} o \arrow[r, bend left, "f"] \arrow[r, bend right, "g"'] & o\ a \end{tikzcd}$ sestavy: $\begin{bmatrix} f: |1|\, a \\ g: |1| \end{bmatrix}$ $\begin{bmatrix} f: |1| \\ g: |1|\, a \end{bmatrix}$ $\begin{bmatrix} f: |1|\, a \\ g: |1|\, a \end{bmatrix}$

Na druhé straně ale (bohužel !) neplatí, že každá sestava rysů je vyjádřitelná jako DAG. Na rozdíl od DAGu neplatí pro sestavy rysů acykličnost – proměnné mohou koindexací odkazovat i na sestavy, v nichž se samy vyskytují.

Příklad: $\left[f: |1| \begin{bmatrix} g: a \\ h: \left[i: |1| \right] \end{bmatrix} \right]$

Je zřejmé, že takovéto (tj. cyklické) sestavy nelze přepsat do podoby DAGů[14] – z toho důvodu, že při definici sestavy rysů jsme nestanovili omezení, že sestava musí být acyklická.

Občas proto bývá skutečně sestava rysů definována méně obecně, než jsme to učinili my, totiž jako sestava, která nemá cykly. Takové řešení má ale zásadní nevýhodu v tom, že množina takových objektů (tj. DAGů či acyklických sestav rysů) není uzavřena vůči operaci unifikace – jinými slovy, platí, že unifikací dvou acyklických sestav rysů může obecně vzniknout sestava cyklická, viz následující příklad.

Příklad: $\begin{bmatrix} f: |1| \left[j: |2| \right] \\ g: \left[i: |1| \right] \\ h: |2| \end{bmatrix} \ \mathrm{Y} \ \begin{bmatrix} f: |1| \\ g: |3| \left[i: |1| \right] \\ h: \left[k: |3| \right] \end{bmatrix} = \begin{bmatrix} f: |1| \left[j: |2| \right] \\ g: |3| \left[i: |1| \right] \\ h: |2| \left[k: |3| \right] \end{bmatrix}$

Že je výsledná sestava v předchozím příkladu skutečně cyklická (zatímco dvě původní nejsou), uvidíme dobře tehdy, když si uvědomíme, že sestava je cyklická právě tehdy, když nějaká její část "obsahuje sama sebe".
Ve výsledné struktuře v tomto příkladu je vidět, že (např.) sestava označená indexem |*1*| obsahuje sestavu s indexem |*2*| (první řádek, hodnota rysu *f*), sestava s indexem |*2*| obsahuje sestavu s indexem |*3*| (třetí řádek, hodnota rysu *h*), a pomyslný kruh se uzavře tím, že (dle řádku druhého, hodnota rysu *g*), sestava s indexem |*3*| obsahuje sestavu s indexem |*1*| – neboli sestava |*1*| obsahuje sestavu |*1*| (byť zprostředkovaně, přes sestavy |*2*| a |*3*|).

Fakt, že unifikací dvou sestav acyklických může vzniknout sestava cyklická, není jen teoretickou záležitostí, ale má značný – a bohužel negativní – praktický dopad při implementaci. Datová struktura, která odpovídá cyklickým sestavám, je totiž podstatně náročnější na zpracování než struktura, při jejímž zpracování se dá využít implicitní informace, že žádný cyklus neobsahuje. Jednoduchý příklad takového rozdílu je třeba procedura na vytištění sestavy – jde o to, jak zabezpečit, aby taková procedura na jedné straně vytiskla opravdu veškerou informaci a na druhé straně se nezacyklila (netiskla "věčně"), tedy o problém, který pro necyklické sestavy vůbec nenastává.

[14] Je dobré si to explicitně uvědomit a zdůvodnit – pokud tedy čtenáři tento fakt zřejmý nepřipadá, doporučujeme, aby se o takovou transformaci sestavy rysů na DAG pokusil, což je nejlepší cesta, jak zjistit, proč to nejde: v žádném případě se totiž nepovede vytvořit *acyklický* graf.

5. Několik algebraických vlastností

V minulých kapitolách jsme zavedli sestavy rysů a operaci na nich (unifikaci) a ukázali jsme si, jak je možné s pomocí takového aparátu vytvářet gramatiky. Takové gramatiky je však nejen potřeba vytvářet, ale také (pro praxi) implementovat a (pro teorii) studovat co do formálních vlastností. Jak k jednomu tak k druhému je ovšem potřeba především postavit vše – a zejména unifikaci – na formálnější základ. To je obsahem této kapitoly.
Představme si, že nyní pomocí sestav rysů popisujeme množinu Ω objektů, z nichž každý má konečný počet vlastností, z nichž každá nabývá konečného počtu hodnot (příp. konečného počtu rozlišitelných stupňů atp.). Představme si, že v libovolné sestavě rysů nepoužijeme vlastnost, která by nebyla přítomna alespoň v jednom objektu z množiny Ω (tj. omezíme se jen na ty vlastnosti, které jsou skutečně přítomny na nějakých objektech z Ω (alespoň na jednom); takové omezení je zřejmě rozumné, např. i proto, že jedině s tímto omezením jsou sestavy rysů schopny efektivně rozlišovat jednotlivé objekty). V takovém případě zřejmě platí, že i počet všech možných (a navzájem různých) sestav rysů, které s různou mírou informace správně (tj. pravdivě, odpovídajícím způsobem) popisují nějaký objekt z množiny Ω, je konečný počet. Přidejme nyní k množině všech takových sestav ještě dvě sestavy speciální, totiž:

- sestavu ⊥ zavedenou výše
- sestavu, která nenese žádnou informaci ("prázdnou" sestavu – dosud jsme se o ní nezmínili, je však zřejmé, že její zavedení dává smysl, je totiž formálním odrazem situace, kdy o popisovaném objektu nevíme nic); takovou sestavu bychom mohli značit např. [] (tj. jen "sestavové závorky", bez jakékoliv vlastnosti v nich uzavřené), avšak zavedeme pro ni značku ⊤ (důvod pro takové značení – jakož i pro značení ⊥ – vyplyne níže).

Je zřejmé, že sestavy ⊥ a ⊤ jsou přidány "rozumně" či "organicky", že rozšiřují základní množinu sestav přirozeným způsobem, jako speciální případy sestav popisujících informaci o vlastnostech objektů z množiny: ⊤ je sestavou, která o objektu dává nulovou informaci, ⊥ je sestavou, která poskytuje přespříliš mnoho informace (tj. konfliktní informaci). Označme tuto množinu sestav rysů (tj. po rozšíření o ⊥ a ⊤) nad množinou objektů Ω jako M.
V následujícím přehledu vlastností algebraické struktury $(M, \curlyvee)$, tj. struktury tvořené množinou M a operací $\curlyvee$ na M definovanou, budeme všechna tvrzení uvádět bez formálních důkazů, půjde však vždy o tvrzení z názoru naprosto zřejmá. Formálněji orientovaného čtenáře odkazujeme na práci (Carpenter, 1992).

Jako první bod bude důležité si uvědomit, jaký je vztah mezi sestavou rysů jakožto prvkem množiny M a množinami popisovaných objektů (tj. množinami prvků z Ω). Platí totiž, že každá sestava rysů $\sigma \in M$ je popisem nějaké (určité) podmnožiny množiny Ω – takovou podmnožinu nazýváme *denotací sestavy rysů σ* a budeme ji v následujících odstavcích značit Den(σ). Den(σ) je tedy taková množina objektů, které (z logického hlediska) splňují podmínky kladené σ na vlastnosti těchto objektů. Platí přitom následující vztahy:

$$\mathrm{Den}(\top) = \Omega$$

Denotace ⊤, tj. sestavy rysů, která nenese žádnou informaci, je celá množina Ω. Plyne to z toho, že sestava rysů ⊤ nijak nedefinuje (tj. ani neomezuje) vlastnosti objektu/objektů, jejichž „popisem" by měla být – ⊤ je tedy popisem libovolného objektu z Ω.

$$\mathrm{Den}(\perp) = \emptyset$$

Denotace ⊥, tj. sestavy rysů, která nese "příliš mnoho" informace, je prázdná množina. Plyne to z toho, že sestava rysů ⊥ je "vnitřně sporná", tj. popisuje vlastnosti, které jsou samy se sebou ve sporu. Je samozřejmé, že takové vlastnosti nemůže mít žádný objekt z Ω, tj. že množina objektů majících takové vlastnosti je prázdná.

Dále platí následující vztah:

$$\forall\ \sigma,\tau \in M : \mathrm{Den}(\sigma\ \mathrm{Y}\ \tau) = \mathrm{Den}(\sigma) \cap \mathrm{Den}(\tau)$$

Toto tvrzení říká, že množina všech objektů, které vyhovují popisu vzniklému unifikací dvou popisů σ a τ, je rovna průniku množin objektů, které odpovídají každému z popisů σ,τ vzatých zvlášť. V podstatě jde tedy jen o formální vyjádření zcela zřejmého faktu, že objekty popsané unifikací dvou popisů jsou právě ty objekty, které odpovídají oběma popisům. Toto tvrzení je také zřejmě v dobré shodě s intuitivní představou, že "čím podrobnější je popis (tj. čím více informací), tím méně je objektů, které tomuto popisu (těmto informacím vzatým najednou) odpovídají".

Uvědomme si dále na tomto místě, že výše uvedenou formulaci, že každá sestava rysů σ je popisem nějaké (určité) podmnožiny Ω, nelze obrátit – tj. neplatí, že každá podmnožina množiny Ω je popsána nějakou (určitou) sestavou rysů. Ukažme si to na (proti)příkladě. Vezměme za Ω množinu, jejímiž prvky jsou (právě) tři objekty O1, O2 a O3, o nichž máme jen tu informaci, že O1 má bílou barvu, O2 ma zelenou barvu a O3 má černou barvu. Každému objektu O1, O2 a O3 pak (po řadě) odpovídá jedna sestava rysů

$$[\text{barva} : \text{bílá}] \qquad [\text{barva} : \text{zelená}] \qquad [\text{barva} : \text{černá}]$$

Neexistuje však sestava rysů, která by popisovala libovolnou množinu obsahující právě dva objekty (např. bílý a zelený). Všimněmě si, že se jedná o situaci velmi podobnou té, kdy se nám nepovedlo popsat morfologické kategorie slovního tvaru slovo *"knize"* kvůli tomu, že jsme neuměli přesně určit jeho pád, současně jsme jej však nechtěli popsat tak, jako by šlo o slovo, jehož pád vůbec neznáme. Tento problém tedy – jak slíbeno – vyřešíme níže.

Podívejme se nyní na další vlastnosti algebraické struktury dané množinou M a operací Y na M definovanou (unifikací). Platí totiž následující řada tvrzení.

Operace Y je uzavřená na M, tj. platí

$$\forall\ \sigma,\tau \in M : \sigma\ \mathrm{Y}\ \tau \in M.$$

Tvrzení říká, že unifikace dvou popisů je také popis, platí tedy triviálně z definice unifikace (příp. po rozšíření o unifikaci, jejímž výsledkem je $\perp$).

Operace Y je na M asociativní, tj. platí

$$\forall\ \sigma,\tau,\upsilon \in M : (\sigma\ \mathrm{Y}\ \tau)\ \mathrm{Y}\ \upsilon = \sigma\ \mathrm{Y}\ (\tau\ \mathrm{Y}\ \upsilon)$$

Toto tvrzení je zřejmé z názoru, a to proto, že existují-li tři popisy objektů (nikoliv nutně navzájem kompatibilní), pak je zřejmě výsledek jejich kombinace stejný, ať již jsou zkombinovány tak, že nejprve jsou zkombinovány první dva popisy a pak teprve k nim přidán popis třetí, nebo pokud je první popis zkombinován s výsledkem kombinace popisu druhého a třetího – a to vše i v případě, že výsledkem (či libovolným mezivýsledkem) je popis vnitřně sporný (v zavedené notaci značený $\perp$).

Operace Y je na M komutativní, tj. platí

$$\forall \sigma,\tau \in M : \sigma \ Y \ \tau = \tau \ Y \ \sigma$$

Z názoru je toto tvrzení zřejmé proto, že existují-li dva popisy téhož objektu (nikoliv nutně navzájem kompatibilní), pak je zřejmě výsledek jejich kombinace stejný, ať již jsou rysy v nich obsažené zkombinovány v libovolném pořadí, opět i v případě, že výsledkem je popis vnitřně sporný (v zavedené notaci značený $\perp$).

Operace Y je na M idempotentní, tj. platí

$$\forall \sigma \in M : \sigma \ Y \ \sigma = \sigma$$

Z názoru je toto tvrzení zřejmé proto, že pokud k informaci σ přidáme tutéž informaci (tj. opět σ), nese výsledná kombinace přesně stejné množství informace, jako je obsaženo v σ (přidali jsme informaci identickou, již známou, tj. stávající informaci jsme o nic nerozšířili).

Další zajímavou a významnou vlastností struktury (M, Y) je existence „jednotkového" prvku (kterým je T), což plyne z toho, že platí

$$\forall \sigma \in M : \sigma \ Y \ \mathsf{T} = \mathsf{T} \ Y \ \sigma = \sigma$$

(prvek T se nazývá „jednotkový" kvůli analogii s operací násobení čísel, kde také platí, že libovolné číslo po znásobení číslem 1 zůstane samo sebou). Z názoru je zřejmé, že uvedené tvrzení platí proto, že unifikace s T vlastně znamená přidání prázdné informace (tj. nepřidání žádné informace), čímž se tedy evidentně množství informace nemůže změnit.

Analogická existenci „jednotkového" prvku je existence „nulového" prvku (kterým je $\perp$), což plyne z toho, že platí

$$\forall \sigma \in M : \sigma \ Y \ \perp = \perp \ Y \ \sigma = \perp$$

(prvek $\perp$ se nazývá „nulový" opět kvůli analogii s operací násobení čísel – výsledek násobení libovolného čísla nulou je nula). Z názoru je zřejmé, že uvedené tvrzení platí proto, že unifikací $\perp$ s libovolnou sestavou rysů nelze dosáhnout toho, aby ve výsledné struktuře nebylo „příliš mnoho" informace – tak „mnoho", že tato sestava je vnitřně nekonsistentní. (Takového výsledku by šlo dosáhnout „ubráním" informace, ne však – jakýmkoliv – jejím přidáním.)

Dále je vhodné (a intuitivní) si uvědomit, že v určitých případech je možné srovnávat dvě sestavy rysů z množiny M podle množství informace, kterou každá z nich obsahuje. Tak např. je asi rozumné (tj. odpovídající naší intuici), že ze sestav σ a τ definovaných následujícím způsobem

$$\sigma = \begin{bmatrix} \text{číslo : jednotné} \\ \text{pád : 4} \end{bmatrix} \qquad \tau = \begin{bmatrix} \text{číslo : jednotné} \\ \text{rod : ženský} \\ \text{slovní\_druh : adjektivum} \\ \text{stupeň : 3} \\ \text{pád : 4} \end{bmatrix}$$

obsahuje sestava σ striktně méně informace a sestava τ obsahuje informace více. Na druhé straně je ale zřejmé, že provádět taková srovnání množství informace obsažené ve dvou sestavách není možné vždy, např. není (rozumně) možné srovnávat množství informace obsažené v následujících dvou sestavách.

$$\begin{bmatrix} \text{číslo} : \text{jednotné} \\ \text{pád} : 4 \end{bmatrix} \qquad \begin{bmatrix} \text{rod} : \text{ženský} \\ \text{slovní\_druh} : \text{adjektivum} \\ \text{stupeň} : 3 \\ \text{pád} : 4 \end{bmatrix}$$

Důležité přitom je, že množství informace v sestavách rysů nelze (lépe: není rozumné) srovnávat proto, že každá z nich obsahuje alespoň částečně jinou informaci – sestava na levé straně obsahuje informaci o čísle, která není obsažena v sestavě na straně pravé, a ta zase obsahuje infomaci o rodu, slovním druhu a stupni, která se nevskytuje v sestavě na straně levé (z čehož je mj. vidět, že porovnávání informačního obsahu sestav se řídí kvalitou, nikoliv kvantitou – alespoň ne kvantitou měřenou počtem rysů).
Další situace, v níž sestavy nelze porovnávat podle množství informace v nich obsažené, nastává tehdy, když sestavy obsahují informaci navzájem konfliktní. Např tedy sestavy

$$\begin{bmatrix} \text{číslo} : \text{množné} \\ \text{pád} : 1 \end{bmatrix} \qquad \begin{bmatrix} \text{číslo} : \text{množné} \\ \text{pád} : 5 \end{bmatrix}$$

zřejmě nelze (co do množství obsažené informace) porovnávát.

Jinými slovy, dvě sestavy lze porovnávat jenom tehdy, když se informace obsažené v každé z nich „plně překrývají" – množina rysů (jak co do jmen vlastností, tak co do jejich hodnot) obsažená v jedné ze sestav je plně obsažena také v sestavě druhé.
Názorně lze takovou "úplnou inkluzi informace" vidět, pokud jsou příslušné sestavy rysů zobrazeny jako DAGy (za předpokladu, že je to možné). Pokud jsou DAGy opravdu srovnatelné, lze jeden z porovnávaných DAGů (ten informativně méně obsáhlý) beze zbytku přiložit na druhý (informativně obsáhlejší) DAG ("přiložením" se míní identita hran, identita ohodnocení hran a identita ohodnocení uzlů). Pokud takové přiložení není možné (vždy nějaká část každého z DAGů "vyčuhuje"), jsou DAGy, tj. i jim příslušné sestavy rysů, neporovnatelné.

Důležitým příkladem dvou sestav rysů, které lze porovnat co do obsahu informace, je dvojice sestav rysů

$$\begin{bmatrix} \text{podmet} : \begin{bmatrix} \text{osoba} : 2 \\ \text{rod} : \text{fem} \end{bmatrix} \\ \text{prísudek} : \begin{bmatrix} \text{osoba} : 2 \\ \text{rod} : \text{fem} \end{bmatrix} \end{bmatrix} \qquad \begin{bmatrix} \text{podmet} : |1| \begin{bmatrix} \text{osoba} : 2 \\ \text{rod} : \text{fem} \end{bmatrix} \\ \text{prísudek} : |1| \end{bmatrix}$$

Je potřeba si uvědomit, že sestava nalevo obsahuje striktně méně informace než sestava napravo: levá sestava říká, že hodnotou rysu *podmet* je sestava $\begin{bmatrix} \text{osoba} : 2 \\ \text{rod} : \text{fem} \end{bmatrix}$ a že hodnotou rysu *prísudek* je sestava $\begin{bmatrix} \text{osoba} : 2 \\ \text{rod} : \text{fem} \end{bmatrix}$, sestava napravo obsahuje tutéž informaci a navíc informaci o tom, že sestavy, které jsou hodnotami rysů *podmet* a *prísudek,* jsou identické (že je to tedy vlastně sestava jediná). Tento rozdíl můžeme samozřejmě vyjádřit i formálně, totiž unifikační rovností

$$\begin{bmatrix} \text{podmet}: \begin{bmatrix} \text{osoba}: 2 \\ \text{rod}: \text{fem} \end{bmatrix} \\ \text{prísudek}: \begin{bmatrix} \text{osoba}: 2 \\ \text{rod}: \text{fem} \end{bmatrix} \end{bmatrix} \;\text{Y}\; \begin{bmatrix} \text{podmet}: \boxed{1} \\ \text{prísudek}: \boxed{1} \end{bmatrix} = \begin{bmatrix} \text{podmet}: \boxed{1} \begin{bmatrix} \text{osoba}: 2 \\ \text{rod}: \text{fem} \end{bmatrix} \\ \text{prísudek}: \boxed{1} \end{bmatrix}$$

ze které fakt, že v levé z výše uvedených sestav je striktně méně informace než v sestavě pravé, vyplývá zvláště zřetelně.

Rozdílnost mezi těmito dvěma sestavami rysů si dále lze dobře uvědomit, pokud si promyslíme, jak se každá z nich chová při unifikaci např. se sestavou $[\text{podmet}: [\text{císlo}: \text{pl}]]$. Výsledky této unifikace jsou totiž rozdílné, v prvním případě dojde jen ke změně hodnoty rysu *podmět*, ve druhém se samozřejmě změní hodnota obou rysů *podmět i přísudek* (prostě proto, že oba tyto rysy mají identickou hodnotu). Konkrétně vypadají tyto unifikace následujícím způsobem.

$$\begin{bmatrix} \text{podmet}: \begin{bmatrix} \text{osoba}: 2 \\ \text{rod}: \text{fem} \end{bmatrix} \\ \text{prísudek}: \begin{bmatrix} \text{osoba}: 2 \\ \text{rod}: \text{fem} \end{bmatrix} \end{bmatrix} \;\text{Y}\; [\text{podmet}: [\text{císlo}: \text{pl}]] = \begin{bmatrix} \text{podmet}: \begin{bmatrix} \text{osoba}: 2 \\ \text{rod}: \text{fem} \\ \text{císlo}: \text{pl} \end{bmatrix} \\ \text{prísudek}: \begin{bmatrix} \text{osoba}: 2 \\ \text{rod}: \text{fem} \end{bmatrix} \end{bmatrix}$$

$$\begin{bmatrix} \text{podmet}: \boxed{1} \begin{bmatrix} \text{osoba}: 2 \\ \text{rod}: \text{fem} \end{bmatrix} \\ \text{prísudek}: \boxed{1} \end{bmatrix} \;\text{Y}\; [\text{podmet}: [\text{císlo}: \text{pl}]] = \begin{bmatrix} \text{podmet}: \boxed{1} \begin{bmatrix} \text{osoba}: 2 \\ \text{rod}: \text{fem} \\ \text{císlo}: \text{pl} \end{bmatrix} \\ \text{prísudek}: \boxed{1} \end{bmatrix}$$

Důležité zde je, že v prvním z případů se nyní hodnoty rysů *podmět* a *přísudek* od sebe liší, zatímco v druhém jsou (samozřejmě) i nadále identické.

Vzhledem k tomu všemu je zřejmě vhodné na množině M zavést reflexivní částečné uspořádání odrážející množství informace v jednotlivých sestavách rysů , tj. relaci, kterou budeme značit symbolem $\subseteq$ (který si – z dobrých důvodů – opět vypůjčíme ze symboliky množinové) a která má následující vlastnosti:

$\subseteq$ je reflexivní, tj. platí	$\forall\, \sigma \in M$:	$\sigma \subseteq \sigma$
$\subseteq$ je antisymetrická, tj. platí	$\forall\, \sigma,\tau \in M$:	$\sigma \subseteq \tau \;\&\; \tau \subseteq \sigma \Rightarrow \sigma = \tau$
$\subseteq$ je tranzitivní, tj. platí	$\forall\, \sigma,\tau,\upsilon \in M$:	$\sigma \subseteq \tau \;\&\; \tau \subseteq \upsilon \Rightarrow \sigma \subseteq \upsilon$

Pokud pro dvě sestavy rysů σ,τ platí $\sigma \subseteq \tau$, budeme říkat, že σ je méně informativní (menší)[15] než τ.

Je dobré si na tomto místě uvědomit, že operace Y (unifikace) a relace $\subseteq$ (uspořádání podle informativity) spolu souvisejí: uspořádání podle informativity lze definovat na základě unifikace. Platí totiž následující vztah:

$$\forall\, \sigma,\tau \in M : \qquad \sigma \subseteq \tau \iff \tau \text{ Y } \sigma = \tau$$

[15] V literatuře se často pro inkluzi informace zavádí termín ***subsumpce***, říká se, že jedna sestava rysů subsumuje druhou. Vzhledem k tomu, že úzus je však nejednotný (pro $\sigma \subseteq \tau$ se někdy říká, že σ subsumuje τ, a jindy, že τ subsumuje σ), budeme se tomuto termínu raději vyhýbat.

Tento vztah opět uvádíme bez (formálního) důkazu (ten lze nalézt v literatuře), i on však patří ke tvrzením zřejmým z názoru. Pokud platí pravá strana ekvivalence, totiž to, že přidáním informace σ k informaci τ se informace τ nijak nerozšíří, znamená to, že informace v σ je již zcela obsažena v informaci τ, neboli že σ je méně informativní (obsahuje méně informace) než τ, což formálně zapisujeme právě jako $\sigma \sqsubseteq \tau$, tj. jako levou stranu ekvivalence. Pokud naopak platí levá strana ekvivalence, tj. σ je méně informativní než τ, pak platí její pravá strana téměř triviálně – pokud přidáme k informaci τ informaci σ, která je v τ již obsažena, pak se tím množství informace v τ zřejmě nezmění, což je přesně obsah fomálního zápisu na pravé straně. Jako speciální případy tohoto tvrzení pak platí zřejmá tvrzení o minimálním a maximálním prvku vzhledem k uspořádání $\sqsubseteq$

$\perp$ je maximální prvek vzhledem k $\sqsubseteq$, formálně $\forall\ \sigma \in M : \sigma \sqsubseteq \perp$

Toto tvrzení, které říká, že $\perp$ obsahuje více informace než kterákoliv jiná sestava rysů, je triviální – sestava $\perp$ je definována právě touto vlastností (viz též výše).

$\top$ je minimální prvek vzhledem k $\sqsubseteq$, formálně $\forall\ \sigma \in M : \top \sqsubseteq \sigma$

Toto tvrzení je opět triviální, neboť sestava $\top$ je definována právě tou vlastností, že je v ní obsaženo méně informace než v jakékoliv jiné sestavě rysů (zcela přesně: v $\top$ není obsažena informace vůbec žádná).

Na samý závěr této kapitoly ještě podotkněme dvě věci:

1. že je rozumné zavést i speciální značení pro negaci relace $\sqsubseteq$. Pokud pro dvě sestavy rysů σ,τ neplatí relace $\sigma \sqsubseteq \tau$, a to ať už proto, že platí $\tau \sqsubseteq \sigma$ (přičemž $\tau \neq \sigma$), nebo proto, že σ a τ jsou co do obsaženého množství informace v našem smyslu neporovnatelné, budeme tento fakt zapisovat jako $\sigma \not\sqsubseteq \tau$. Například tedy platí následující vztahy:

$$\begin{bmatrix} \text{slovní\_druh : zájmeno} \\ \text{osoba : 2} \\ \text{číslo : množné} \end{bmatrix} \not\sqsubseteq \begin{bmatrix} \text{osoba : 2} \\ \text{číslo : množné} \end{bmatrix}$$

$$\begin{bmatrix} \text{pád : 1} \end{bmatrix} \not\sqsubseteq \begin{bmatrix} \text{slovní\_druh : zájmeno} \\ \text{osoba : 2} \\ \text{číslo : množné} \end{bmatrix}$$

$$\begin{bmatrix} \text{osoba : 2} \\ \text{číslo : množné} \end{bmatrix} \not\sqsubseteq \begin{bmatrix} \text{slovní\_druh : zájmeno} \\ \text{osoba : 3} \\ \text{číslo : množné} \end{bmatrix}$$

2. Že jsme kvůli jednoduchosti v celé kapitole používali jen takové příklady sestav rysů, v nichž hodnoty rysů byly vždy atomické. Formální tvrzení, která jsme uvedli, však platí samozřejmě i obecně, což by se ostatně dalo ukázat i na složitějších příkladech – takových, kde hodnotami rysů jsou také sestavy rysů nebo proměnné.

6. Rozšíření formalismu o negaci a disjunkci

Vraťme se nyní k příkladu s barevnými objekty ze začátku kap. 5. Jádrem problému, který tento příklad ilustroval, byla nemožnost popsat pomocí formalismu sestav rysů libovolnou podmnožinu určité množiny objektů. Konkrétně šlo o to, že základní množina Ω měla jako své prvky (právě) tři objekty O1, O2 a O3, o nichž byla k dispozici jen ta informace, že O1 má bílou barvu, O2 má zelenou barvu a O3 má černou barvu, tj. tyto objekty bylo možno (po řadě) popsat sestavami rysů

$$[\text{barva} : \text{bílá}] \qquad [\text{barva} : \text{zelená}] \qquad [\text{barva} : \text{černá}]$$

Potíž pak tkvěla v tom, že nebylo možno sestrojit takovou sestavu rysů, která by popisovala libovolnou množinu obsahující právě dva objekty (např. bílý a zelený).
V podstatě identický problém byl zmíněn i v Příkladu XXX (pád slovního tvaru "*knize*").

Je zřejmé, že oba tyto problémy lze vyřešit zavedením disjunkce hodnot jako možné hodnoty rysu. To ostatně odpovídá i situaci, kdy je k dispozici pouze částečná (nejednoznačná) informace o určité vlastnosti objektu. Například tedy pokud ve výše uvedeném příkladu víme, že objekt je buď bílý nebo černý (tj. nevíme, kterou z těchto barev má, víme však, že není zelený), rádi bychom tuto znalost zapsali např. následujícím způsobem (disjunkci budeme značit symbolem $\vee$).

$$[\text{barva} : \text{bílá} \vee \text{černá}]$$

Denotace takovéto sestavy rysů je zřejmě (v našem případě) dvouprvková množina obsahující bílý objekt a černý objekt, v obecném případě množina všech objektů, jejichž barva je buď černá nebo bílá[16].
Abychom skutečně mohli takové disjunktivní hodnoty používat, musíme ovšem dále rozšířit definici přípustných hodnot rysu – naštěstí je to v tomto případě jednoduché, je potřeba jen prostě říci, že pokud jsou *h1* a *h2* možné (tj. již definované) hodnoty rysu, pak je i *h1* $\vee$ *h2* možná (tj. správně definovaná) hodnota rysu, přičemž sémantika sestavy rysů tvaru $[\text{f} : \text{h1} \vee \text{h2}]$ je identická výsledku operace disjunkce sestav rysů $[\text{f} : \text{h1}]$ a $[\text{f} : \text{h2}]$, tj. platí definitorická rovnost

$$[\text{f} : \text{h1} \vee \text{h2}] = [\text{f} : \text{h1}] \vee [\text{f} : \text{h2}]$$

Takovou definicí jsme ovšem (momentálně) nijak nepokročili, protože operace $\vee$ (disjunkce) není na sestavách rysů definována. Proto ji níže v kapitole zavedeme.

Nejdříve ale ještě pro lepší porozumění uvedeme následující rovnosti, které dávají návod na to, jak se dívat na méně triviální sestavy rysů obsahující disjunkci (totiž tak, že je na případ triviálních sestav převedeme):

$$\begin{bmatrix} \text{f} : \text{h1} \vee \text{h2} \\ \text{g} : \text{a} \end{bmatrix} = [\text{g} : \text{a}] \, \text{Y} \, [\text{f} : \text{h1} \vee \text{h2}] = [\text{g} : \text{a}] \; \text{Y} \; ([\text{f} : \text{h1}] \vee [\text{f} : \text{h2}])$$

$$\begin{bmatrix} \text{f} : \text{h1} \vee \text{h2} \\ \text{g} : \text{a} \\ \text{k} : \text{m1} \vee \text{m2} \end{bmatrix} = [\text{g} : \text{a}] \, \text{Y} \, [\text{f} : \text{h1} \vee \text{h2}] \, \text{Y} \, [\text{k} : \text{m1} \vee \text{m2}] = [\text{g} : \text{a}] \; \text{Y} \; ([\text{f} : \text{h1}] \vee [\text{f} : \text{h2}]) \; \text{Y} \; ([\text{k} : \text{m1}] \vee [\text{k} : \text{m2}])$$

Dále se v rámci neformálního úvodu do problematiky disjunkce ještě podívejme na disjunkci jako na operaci s informací. Byla-li unifikace sestav rysů z tohoto pohledu sjednocením informace pocházející ze dvou různých zdrojů, je (asi) přirozené požadovat, aby disjunkce byla průnikem informačních obsahů sestav rysů, tj.

[16] Uvažujeme ovšem jen popis množiny jednobarevných objektů – nebudeme se zde pouštět do úvah, co by mělo platit ve "světě", kde objekty mohou být i vícebarevné (např. bílé s černými tečkami, fialové s černými kolečky a bílými hvězdičkami atd.)

aby výsledkem disjunkce dvou sestav rysů byla sestava obsahující právě tu informaci, která je oběma původním sestavám společná.
Takový požadavek zřejmě triviálně platí v některých speciálních případech (či – při jiném pohledu – motivuje v těchto případech definici výsledku):

- zřejmě by mělo platit, že disjunkce libovolné sestavy ***S*** s informačně prázdnou sestavou T je opět T (za srovnání zde, tak jako všude jinde, stojí i analogie s disjunkcí ve výrokovém počtu: výsledkem disjunkce libovolného výroku ***a*** s konstantou *pravda* je vždy *pravda*)
- podobně by disjunkce libovolné sestavy ***S*** s informačně "přeplněnou" sestavou ⊥ měla být sestava ***S*** (disjunkce libovolného výroku ***a*** s konstantou *nepravda* je výrok ***a***).

Není však na první pohled jasné, že pro dvě obecné sestavy rysů ***S1, S2*** platí, že jejich disjunkce je právě informace obsažená v obou z nich. Proto se to pokusíme v dalším odstavci neformálně osvětlit.
Vezměme si nejdříve již dvakrát uvedený příklad jednoduchého světa barevných objektů a zopakujme si, že jsou v něm k dispozici právě tři barvy: *bílá*, *zelená* a *černá*. V takovém světě mohou tedy existovat tři "základní" ("jednoduché") popisy objektů:

$$[\text{barva}:\text{bílá}] \qquad [\text{barva}:\text{zelená}] \qquad [\text{barva}:\text{černá}]$$

Úkolem bude ukázat, že vskutku platí, že (např.) informace obsažená v disjunkci popisů (sestav rysů) $[barva:bílá]$ a $[barva:zelená]$ je skutečně právě tou informací, která je jim (jako jednotlivým popisům) společná. Centrálním bodem celé úvahy je všimnout si, že dojem, že tyto dvě sestavy vůbec žádnou společnou informaci neobsahují, není správný – přesto, že "na první pohled" to tak možná vypadá.
Ve "světě", jehož objekty tyto sestavy rysů popisují, totiž platí, že

a. být bílý je totéž co nebýt zelený a nebýt černý
b. být zelený je totéž co nebýt bílý a nebýt černý

a dále ještě, že

c. být bílý nebo být zelený je totéž co nebýt černý.

Pomiňme nyní formální nedostatek, totiž to, že nemáme formálně zavedenou negaci (napravíme jej vzápětí níže), a uvědomme si, že:

- informace obsažená v sestavě $[\text{barva}:\text{bílá}]$ je tedy z tohoto hlediska právě infomace, že objekt není ani zelený ani černý
- informace obsažená v sestavě $[\text{barva}:\text{zelená}]$ je tedy z tohoto hlediska právě infomace, že objekt není ani bílý ani černý.

Z toho už lze vskutku snadno nahlédnout, že ona ***společná*** informace je, že objekt není černý. A to podle bodu c. výše zase dále znamená, že objekt je bílý nebo zelený. Tedy platí, že

$$[\text{barva}:\text{bílá}] \;\vee\; [\text{barva}:\text{zelená}] = [\text{barva}:\text{bílá} \vee \text{zelená}]$$

přičemž sestava na pravé straně vskutku obsahuje informaci, která je "průnikem informací" obsažených v jednotlivých sestavách na straně levé.

Instruktivní může být i uvést alespoň příklad toho, jak vypadá denotace disjunkce dvou sestav rysů – byť asi nepůjde o nic velmi překvapivého. Mějme tedy např. množinu G geometrických objektů, jejímiž prvky jsou různé krychle a koule v bílé a černé barvě (v množině existuje alespoň jedna bílá krychle, alespoň jedna bílá koule, alespoň jedna černá krychle a alespoň jedna černá koule).

Zřejmě (z názoru) platí následující rovnosti:

$$\mathrm{Den}(\left[\mathrm{barva: bílá}\right] \vee \left[\mathrm{barva: černá}\right]) = \mathrm{Den}(\mathrm{T}) = \mathrm{Den}(\left[\mathrm{barva: bílá}\right]) \;\; \mathrm{Y} \;\; \mathrm{Den}(\left[\mathrm{barva: černá}\right]) = \mathrm{G}$$

$$\mathrm{Den}(\begin{bmatrix}\mathrm{barva: bílá}\\ \mathrm{t\,var: krychle}\end{bmatrix} \vee \begin{bmatrix}\mathrm{barva: černá}\\ \mathrm{t\,var: krychle}\end{bmatrix}) = \mathrm{Den}(\left[\mathrm{t\,var: krychle}\right]) =$$

$$= \mathrm{Den}(\begin{bmatrix}\mathrm{barva: bílá}\\ \mathrm{t\,var: krychle}\end{bmatrix}) \;\; \mathrm{Y} \;\; \mathrm{Den}(\begin{bmatrix}\mathrm{barva: černá}\\ \mathrm{t\,var: krychle}\end{bmatrix})$$

Z těchto rovností je možno usoudit ("uhádnout" – ve smyslu "generalizovat bez důkazu), že mezi libovolnými dvěma sestavami rysů SR1 a SR2 obecně platí vztah

$$\mathrm{Den}(\mathrm{SR1} \vee \mathrm{SR2}) = \mathrm{Den}(\mathrm{SR1}) \;\; \mathrm{Y} \;\; \mathrm{Den}(\mathrm{SR2}).$$

Takový vztah vyplývá kromě výše uvedených příkladů jednak z představy o disjunkci jako "společné informaci" (tj. informaci méně obsáhlé, a tedy méně specifické, než jaká se nachází v jednotlivých sestavách – a je zřejmé, že méně specifické informaci denotačně vyhovuje větší množina objektů) a jednak i z korespondence mezi unifikací a disjunkcí, konkrétně tedy z předpokládané korespondence vztahů mezi denotacemi unifikace a disjunkce.

Po takovémto neformálním zavedení disjunkce se nyní podívejme na některé základní vlastnosti operace disjunkce, tedy na operace, které jsou pro disjunkci natolik charakteristické, že je možné ji definovat jako operaci tyto vlastnosti splňující. Kromě toho je pak již triviální odvodit z těchto vlatností i vlastnosti sestav rysů obsahujících disjunkci jako hodnotu nějakého svého rysu.
Mějme tedy dánu množinu Ω určitých objektů, množinu M sestav rysů (obsahujících disjunkci) popisujících nějaké podmnožiny množiny Ω, a operaci disjunkce na množině M.
Operace $\vee$ je uzavřená na M, tj. platí

$$\forall\, \sigma,\tau \in M : \sigma \vee \tau \in M.$$

Platnost tohoto tvrzení je zřejmá z názoru, opírajícího se o denotaci popisů: pokud σ a τ jsou popisy nějakých pomnožin množiny Ω, pak jistě i disjunkce těchto popisů, tj. popis množiny, jejíž prvky patří buď do $\mathrm{Den}(\sigma)$ nebo do $\mathrm{Den}(\tau)$, jsou popisem podmnožiny množiny Ω. Každou podmnožinu množiny Ω přitom již nyní (po zavedení disjunkce) umíme zapsat jako disjunkci několika popisů, tj. jako prvek M.

Operace $\vee$ je na M asociativní, tj. platí

$$\forall\, \sigma,\tau,\upsilon \in M : (\sigma \vee \tau) \vee \upsilon = \sigma \vee (\tau \vee \upsilon)$$

Z názoru je toto tvrzení zřejmé proto, že existují-li tři popisy objektů, pak je zřejmě výsledek jejich disjunkce je stejný, ať již jsou zkombinovány tak, že nejprve jsou zkombinovány první dva popisy a pak teprve k nim přidán popis třetí, nebo pokud je první popis zkombinován s výsledkem kombinace popisu druhého a třetího.

Operace $\vee$ je na M komutativní, tj. platí

$$\forall\, \sigma,\tau \in M : \sigma \vee \tau = \tau \vee \sigma$$

Z názoru je toto tvrzení zřejmé proto, že existují-li dva popisy objektů, pak je zřejmě výsledek jejich disjunkce stejný, ať již jsou disjunktivně zkombinovány v libovolném pořadí.

Dále zřejmě platí, že operace $\vee$ je na M idempotentní, tj. platí

$$\forall\ \sigma \in M : \sigma \vee \sigma = \sigma$$

Další vlastností struktury (M, $\vee$) je existence „jednotkového“ prvku (kterým je $\perp$). Spíše než abychom to dokazovali (platí to však opět zcela zřejmě z názoru), nadefinujeme, že platí vztah

$$\forall\ \sigma \in M : \sigma \vee \perp = \perp \vee\, \sigma = \sigma$$

Taková definice je přitom "rozumná" – tedy minimálně stejně rozumná jako tvrzení z výrokové logiky, že disjunkce libovolné pravdivostní hodnoty ***a*** s hodnotou *nepravda* je právě hodnota ***a***: jde v obou případech o to, že se vždy snažíme "vyhnout" hodnotám jako je $\perp$ nebo *nepravda*, tj. v disjunkci dáváme systematicky přednost hodnotě alternativní.

Jak lze očekávat, je existenci „jednotkového“ prvku analogická i existence „nulového“ prvku (kterým je $\top$). Platí totiž

$$\forall\ \sigma \in M : \sigma \vee \top = \top \vee \sigma = \top$$

Znovu (tak jako v případě "jednotkového" prvku) je vše zřejmé z názoru a také z analogie s vlastnostmi disjunkce ve výrokovém počtu (prvek $\top$ zde hraje roli hodnoty *pravda*).

Po zavedení disjunkce jako hodnoty rysu a disjunkce jako operace na množině M všech sestav rysů je potřeba patřičně rozšířit i operaci unifikace, přesněji řečeno provázat operace unifikace a disjunkce. Platí totiž jejich vzájemná distributivita, tj. platí následující vztahy:

$$\forall\ \sigma,\tau,\upsilon \in M : (\sigma \vee \tau)\ \ Y\ \ \upsilon = (\sigma\ \ Y\ \ \upsilon) \vee (\tau\ \ Y\ \ \upsilon)$$
$$\forall\ \sigma,\tau,\upsilon \in M : (\sigma\ \ Y\ \ \tau) \vee \upsilon = (\sigma \vee \upsilon)\ \ Y\ \ (\tau \vee \upsilon)$$

Zároveň platí, že algebraická struktura (M, Y, $\vee$, $\subseteq$) je svaz, tj. že pro každé dva prvky σ, τ je jednoznačně definováno:

- jejich společné minimum (totiž jako $\sigma \vee \tau$ – tj. sestava rysů, která je informačně nejobsažnější ze všech sestav, které obsahují méně (nebo nejvýše stejně) informace než σ a současně méně (nebo nejvýše stejně) informace než τ)
- jejich společné maximum (jako $\sigma\ \ Y\ \ \tau$ – tj. sestava rysů, obsahující nejmenší množství společné informace),

a dále platí, že

- maximální prvek svazu vzhledem k uspořádání $\subseteq$ je $\perp$
- minimální prvek svazu vzhledem k uspořádání $\subseteq$ je $\top$,

což vše velmi dobře odpovídá intuitivní představě o sestavách rysů jako "zásobnících na informaci" a představám o unifikaci jako sjednocní informace a disjunkci jako průniku informace.

Pojem maxima a minima dvou sestav je samozřejmě možno velmi přirozeně rozšířit na maximum a minimum konečné nebo spočetné množiny sestav. Formálně to je možno provést následujícími induktivními definicemi, které se opřou o komutativitu a asociativitu operací unifikace a disjunkce[17].

$$\min(\emptyset) = \max(\emptyset) = \mathsf{T}$$

$$\max(\{\sigma_1, \sigma_2, \sigma_3, ...\}) = \max(\{\sigma_1\} \ \mathrm{Y} \ \{\sigma_2, \sigma_3, ...\}) = \sigma_1 \ \mathrm{Y} \ \max(\{\sigma_2, \sigma_3, ...\}) =$$
$$= \sigma_1 \ \mathrm{Y} \ \sigma_2 \ \mathrm{Y} \ \sigma_3 \ \mathrm{Y} \ ...$$

$$\min(\{\sigma_1, \sigma_2, \sigma_3, ...\}) = \min(\{\sigma_1\} \ \mathrm{Y} \ \{\sigma_2, \sigma_3, ...\}) = \sigma_1 \vee \min(\{\sigma_2, \sigma_3, ...\}) =$$
$$= \sigma_1 \vee \sigma_2 \vee \sigma_3 \vee ...$$

Taková definice říká:

- ve své první části, že maximum i minimum prázdné množiny sestav rysů je T (sestava rysů neobsahující žádnou informaci − což lze pro prázdnou množinu sestav rysů jistě pokládat za nejpřirozenější možnosti),
- v druhé části, že maximem množiny sestav rysů je jejich unifikace[18],
- ve třetí části, že minimem množiny sestav rysů je jejich disjunkce.

Po zavedení disjunkce a zjištění, že vzniklá struktura je svaz, tj. že zejména ke každé množině je samozřejmé možné (a rozumné) vést paralelu s výrokovým počtem dále a definovat na množině sestav rysů negaci. Ta může být nejjednodušeji (a přitom i nejpřirozeněji) zavedena následujícím předpisem:

$$\forall \sigma \in M : \neg \sigma = \min(\{\tau \in M; \sigma \ \mathrm{Y} \ \tau = \perp\})$$

Tento na první pohled možná ne úplně srozumitelný zápis říká, že negace sestavy rysů σ je nejmenší sestava rysů, která obsahuje informaci nekompatibilní se σ. Jinými slovy, negace σ je sestava rysů taková, že jak ona sama, tak všechny sestavy, které obsahují více informace než ona sama, dávají při unifikaci se σ výsledek $\perp$. (Opět může být pro porozumění užitečné vést paralelu s výrokovým počtem, i když ten je v tomto případě daleko jednodušší: konjunkce výroku ***a*** s jeho negací je vždy *nepravda*, což tedy, převedeno do oblasti sestav rysů, znamená, že hledání negace sestavy rysů je hledání nejobecnější, tj. informativně nejméně obsažné, sestavy takové, že při unifikaci se σ dává $\perp$).
Je asi vhodné poukázat na tomto místě na dva speciální případy negace, totiž na rovnosti

$$\neg \mathsf{T} = \perp \qquad\qquad \neg \perp = \mathsf{T}$$

Po zavedení unifikace (vlastně ovšem: konjunkce), disjunkce a negace sestav rysů je zřejmé, že by bylo možné zavést i další operace na sestavách rysů: implikaci, ekvivalenci atd. Záležitost je ale natolik triviální, že s ní nebudeme čtenáře nudit, raději připomeneme jinou (ale skoro stejně zřejmou) věc, že na množině M všech sestav rysů popisujících určité univerzum objektů platí i de Morganovy vztahy mezi třemi již definovanými operacemi, tj, že platí

[17] Při čtení těchto definic je potřeba dát pozor na to, že symbol Y je v nich použit ve dvojím významu: jednak jako symbol pro operaci unifikace (stojí-li mezi sestavami rysů), jednak jako symbol pro sjednocení množin (stojí-li mezi dvěma množinami).
[18] Z hlediska "čisté matematiky" je druhá část definice vlastně zmatečná – je to současně definice a tvrzení. Pro účely tohoto skripta však se spokojíme i s takto nedokonalou formulací. Totéž provedeme ostatně i bezprostředně níže, s definicí minima množiny sestav rysů.

$$\forall\, \sigma, \tau \in \mathrm{M} : \neg(\sigma \;\mathrm{Y}\; \tau) = \neg\, \sigma \vee \neg\, \tau$$
$$\forall\, \sigma, \tau \in \mathrm{M} : \neg(\sigma \vee \tau) = \neg\, \sigma \;\mathrm{Y}\; \neg\, \tau$$

Na závěr této kapitoly zmiňme již jen jednou větou fakt, že disjunkce a v míře ještě daleko větší negace jsou implementačně náročné, a proto se (alespoň v době psaní těchto skript) používají spíše jen v teoretických popisech syntaxe – nebývají implementovány v počítačových formalismech na zpracování přirozeného jazyka, alespoň ne ve své plné podobě (relativně často se tak lze setkat např. s disjunkcí a/nebo negací omezenými pouze na atomické hodnoty apod.)

7. Rozšíření formalismu o typování

Formalismus sestav rysů tak, jak jsme se s ním setkali doposud, není schopen zabránit tomu, aby unifikací vznikaly sestavy, které jsou sice formálně bezchybné, ale přitom intuitivně zcela nesmyslné – sestavy, které nemohou odpovídat žádnému "rozumnému" objektu. Uveďme si pro takovou unifikaci (a její výsledek) po jednom příkladu ze světa geometrických těles a z lingvistiky.

$$\left[\mathrm{t\,var : krychle}\right] \;\mathrm{Y}\; \left[\mathrm{polomer : 10}\right] = \begin{bmatrix} \mathrm{t\,var : krychle} \\ \mathrm{polomer : 10} \end{bmatrix}$$

$$\left[\text{pád : acc}\right] \;\mathrm{Y}\; \left[\mathrm{zpusob : ind}\right] = \begin{bmatrix} \text{pád : acc} \\ \mathrm{zpusob : ind} \end{bmatrix}$$

Je zřejmé, že ač byly "původní" sestavy rysů zcela v pořádku a dobře popisovaly určité třídy objektů, jsou v obou případech výsledky unifikace naprosto nesmyslné a žádný objekt popisovat nemohou – krychle nemají poloměr a jakýkoliv lingvistický objekt, který vykazuje jako svůj rys vykazuje *pád*, už nemůže vykazovat (slovesný) *způsob*. Z takového inuitivního pohledu by tedy vlastně měly platit rovnosti

(HJZ) a. $\left[\mathrm{t\,var : krychle}\right] \;\mathrm{Y}\; \left[\mathrm{polomer : 10}\right] = \perp$

b. $\left[\text{pád : acc}\right] \;\mathrm{Y}\; \left[\mathrm{zpusob : ind}\right] = \perp$

V té formalizaci, kterou jsme až doposud zavedli, je však v podstatě nemožné požadovat, aby takové rovnosti platily. Zásadním problémem je to, jak zařídit, aby v určitých případech platily rovnosti (HJZ), avšak aby obecně neplatilo pro libovolné dva od sebe různé rysy *f,g*, a jejich hodnoty *a,b*, že

$$\left[\mathrm{f : a}\right] \mathrm{Y} \left[\mathrm{g : b}\right] = \perp$$

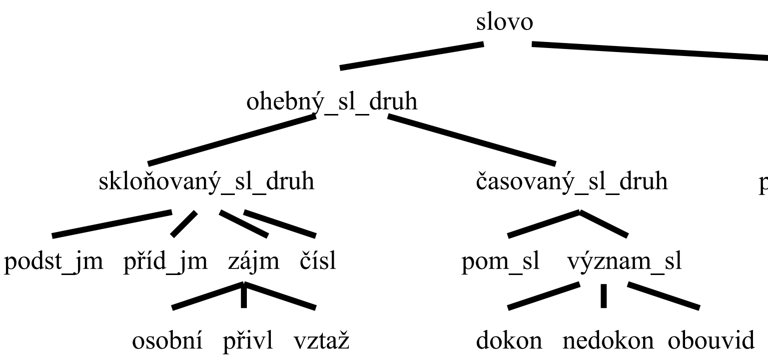
Důvodem, proč by měly rovnosti v (HJZ) platit, je totiž informace, kterou zatím neumíme vyjádřit: informace o možném (či naopak nemožném) souvýskytu rysů v jedné struktuře. K výsledku $\perp$ při unifikaci v těchto rovnostech vede nekompatibilita výskytu jmen rysů (zatímco ve všech předchozích případech k takovému výsledku vždy vedla nekompatibilita hodnot rysů). Tak např. v (HJZ)b není ve sporu informace o akusativu a indikativu, ale současná přítomnost (v jedné struktuře) rysů *pád* a *způsob*.

Je proto potřeba zavést do formalismu sestav rysů prostředek, který vyjádření a uplatnění takové informace umožní: je jím ***typování*** sestav rysů. Typování přiřazuje každé sestavě rysů její typ, který přitom není formálně vyjádřen zápisem rysu (tj. jako dvojice ***<název> : <hodnota*** >), ale jiným způsobem: v tomto skriptu bude použit prostý zápis jména typu zcela nahoře mezi hranatými závorkami, viz příklady v (KGT).

(KGT) $$\begin{bmatrix} {}^{\text{pravidelny\_mnohosten}} \\ \text{delka\_hrany} : 150 \end{bmatrix} \qquad \begin{bmatrix} {}^{\text{substantivum}} \\ \text{pád} : \text{acc} \\ \text{num} : \text{pl} \end{bmatrix} \qquad \begin{bmatrix} {}^{\text{sloveso}} \\ \text{osoba} : \text{druhá} \\ \text{číslo} : \text{jednotné} \\ \text{zpusob} : \text{rozkazovací} \end{bmatrix}$$

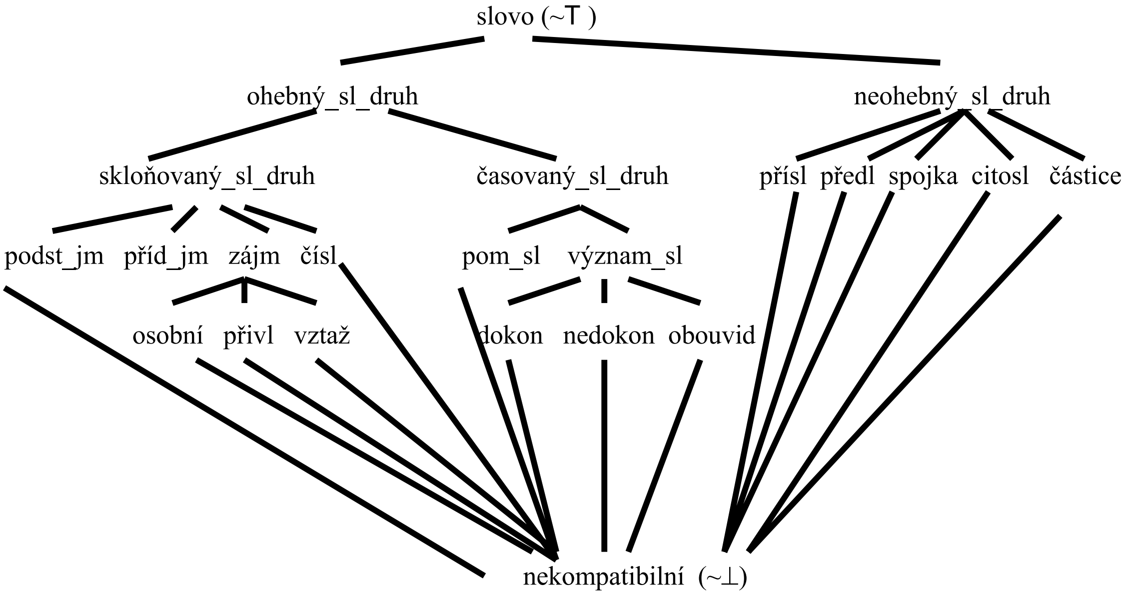
Nejjednodušším požadavkem by potom mohlo být, aby dvě struktury bylo možné unifikovat pouze tehdy, pokud mají stejný typ (tj. aby platilo, že pokud mají dvě struktury různý typ, je jejich unifikace rovna $\perp$). Fakticky by však takový požadavek nic nepřinesl, protože by se totéž dalo vyjádřit i tím, že by se typ struktury zavedl jako rys (např: *typ:sloveso*)– unifikace dvou struktur by pak byla automaticky rovna $\perp$ vždy, když by jejich typy (tj. hodnoty rysu *typ*) nebyly identické (a samozřejmě případně i jindy). Skutečně novou kvalitu přináší typování až teprve tehdy, když je zavedena (jednoduchá) hierarchie typů. Příkladem jedné linie takové (jednoduché) hierarchie je posloupnost (od nejobecnějšího k nejspecifičtějšímu typu) *objekt - mnohostěn - pravidlený_mnohostěn - kvádr - krychle*, jiným příkladem by mohla být posloupnost *morfologická_kategorie - flektivní_kategorie - zájmeno - osobní_zájmeno*. V množině sestav rysů, z nichž každý nese konkrétní typ z konkrétní linie hierarchie, jsou dvě sestavy typově kompatibilní (tj. unifikovatelné) právě tehdy, když oba typy jsou prvky téže linie. Typem sestavy, která je výsledkem unifikace, je potom typ, jenž je (v rámci hierarchie) specifičtější.

Příklad: pokud hierarchie bude "definována" graficky (obecnější typ výše, méně obecný níže), je možné představit si např. následující hierarchii morfologických kategorií:

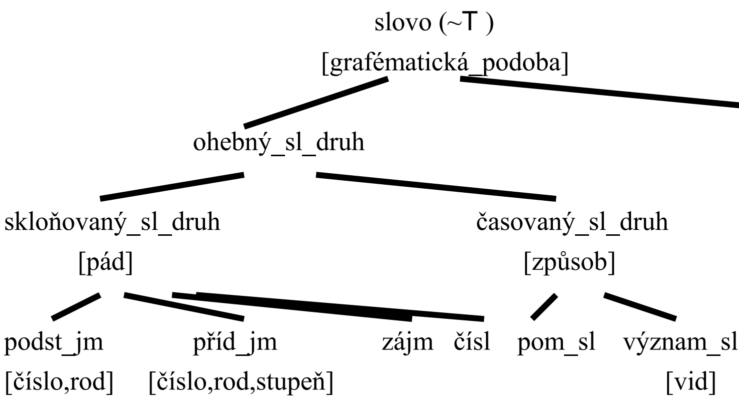


V takové hierarchii je možné (a intuitivně "rozumné") unifikovat dvě sestavy rysů, z nichž první je typu *ohebný_sl_druh* a druhá typu *obouvid*, nebo první typu *zájm* a druhá typu *přivl*. Ze zcela stejných důvodů pak naopak není možné unifikovat[19] dvě sestavy, z nichž první je typu *ohebný_sl_druh* a druhá *předl* a podobně. Hierarchie typů tak vlastně vytváří triviální horní polosvaz, jehož minimálním prvkem (v grafickém zobrazení) je typ odpovídající nejobecnějšímu (tj. libovolnému) objektu – zároveň je to ovšem typ, který o objektu nevypovídá žádnou konkrétní informaci (vlastně je to tedy "typová" obdoba $\top$). Tento polosvaz lze velmi jednoduše, totiž přidáním umělé hodnoty "nekompatibilní" (obdoba $\perp$), doplnit na svaz, viz příklad[20].

[19] "není možné unifikovat" samozřejmě ve smyslu "výsledkem unifikace je vždy hodnota $\perp$"
[20] Pouze z grafických důvodů chybí v obrázku spojnice mezi *příd_jm* a *nekompatibilní* ...



Smysl zavedení typů však není jen v tom, aby bylo zabráněno vzniku "nesmyslných" sestav rysů při unifikaci, ale také (a možná především) v tom, aby byla možná typová kontrola[21] jak během výpočtu, tak již i na vstupních datech. Jde o to, že svou plnou roli hrají typy teprve tehdy, když se ještě doplní údaji o tom, jaké rysy (jména vlastností) každý z typů připouští na sestavách rysů, které k tomuto typu příslušejí. Platí přitom samozřejmě konvence, že ty rysy, které jsou připuštěny "nadtypem" (tj. obecnějším typem, typem umístěným výše v hierarchické linii) se u "podtypu" již neuvádějí, tj. předpokládá se jejich hierarchické dědění. Pokud bychom přípustnost připisovali přímo do hierarchie, mohli bychom horní část výše uvedeného obrázku doplnit např. následujícím způsobem.



Výše uvedené příklady ukazovaly ten nejjednodušší příklad typování, který odpovídá dělení množiny všech objektů podle jednoho kritéria (přesněji: podle jednoho stále zjemňovaného kritéria, příp. několika navzájem propojených kritérií – viz příklady s dělením na slovní druhy). Obecně je ovšem možné (a prakticky to i velmi často nastává) typování provést podle několika nezávislých kritérií: např. množinu nějakých stereometrických modelů klasifikovat podle tvaru (kvádr, koule, válec, kužel, jehlan, ...), podle barvy (bílá, žlutá, modrá, červená, ...) a podle materiálu, ze kterého jsou vyrobeny (dřevo, plech, plastik, ...). Tato tři typování vytvářejí tři nezávislé hierarchie (obecně je takto vytvořeno ***n*** nezávislých hierarchií typů), které ovšem fakticky mají společný

[21] "Zabránění vzniku nesmyslných sestav" a "typová kontrola" jsou ovšem do značné míry totéž, tj. jde jen o rozdílný pohled na stejnou věc.

minimální prvek T a společný maximální prvek ⊥. Proto je možné všechny spojit do hierarchie jediné, což ovšem obrázek velmi zesložití: aby výsledek byl více než pouhou mechanickou kombinací několika hierarchií, je potřeba zavést operace analogické unifikaci (tj. operaci "sjednocení" dvou typů) a disjunkci (operaci "průniku" dvou typů). V uvedeném příkladu tak vzniknou např. typy *koule ∪ žlutá, jehlan ∪ modrá ∪ dřevo, kvádr ∨ plastik, bílá ∨ červená, dřevo ∨ plech ∨ plastik ∨ válec* atd. Samozřejmě přitom zůstanou v platnosti rovnosti typu *kvádr ∪ koule = nekompatibilní (⊥)*, vyplývající z původních (jednoduchých, jednokritériových) hierarchií. Obdobně lze samozřejmě zavést i negaci, případně další operace, o nichž všech platí stejné zákony, které byly uvedeny výše pro unifikaci, disjunkci, negaci, ... netypovaných sestav rysů.

Poté, co byl takto vybudován svaz nad všemi možnými typy (tj. základními typy a jejich možnými kombinacemi), vyvstává otázka, jaký je vlastně rozdíl např. mezi zápisy (AWR)a,b nebo (PKN)a,b (míní se rozdíl v intuitivním chápání, množství informace atd., nikoliv rozdíl ve formálním zápisu).

(AWR) a. [barva: červená] b. $[^{\text{červená}}]$

(PKN) a. $\begin{bmatrix} \text{t var : koule} \\ \text{materiál : plastik} \end{bmatrix}$ b. $\begin{bmatrix} ^{\text{koule Y plastik}} \\ \\ \end{bmatrix}$

V daném případě je situace taková, že zápisy jsou z těchto hledisek zcela rovnocenné, což by platilo pro (PKN)a,b dokonce i v případě, že bychom k nim přidali ještě zápisy (OLJ)a,b.

(OLJ) a. $\begin{bmatrix} ^{\text{koule}} \\ \text{materiál : plastik} \end{bmatrix}$ b. $\begin{bmatrix} ^{\text{plastik}} \\ \text{t var : koule} \end{bmatrix}$

Je tedy v zcela na uživateli (v případě unifikačních gramatik: na lingvistovi), zda se rozhodne určitou vlastnost (lingvistických) objektů popisovat pomocí rysu nebo pomocí typu sestavy rysů[22].
Důležitou výjimku z této "svobody rozhodování" tvoří případy, kdy je potřeba určitou vlastnost sdílet (koindexovat hodnoty pomocí proměnné) – to je možné pouze v případě, že vlastnost je vyjádřena rysem (pro typy nebyly proměnné zavedeny, a zejmě by to ani nedávalo dobrý smysl). Nejtypičtějším příkladem, na kterém si tuto situaci můžeme ukázat, je v lingvistice koindexace vyjadřující shodu podmětu s přísudkem v osobě, čísle a rodu (např. ve větě *Chlapec přišel* se jedná o shodu ve třetí osobě, jednotném čísle a rodu mužském životném). Pokud chceme vyjádřit pouze shodu právě v těchto třech kategoriích (bez toho, že by byly udány jejich konkrétní hodnoty), je možné to vyjádřit různými způsoby, ale v každém způsobu vyjádření se musí vyskytnout koindexace příslušných hodnot, viz příklady v (UZG)a,b (v (UZG)b stojí za povšimnutí trik spočívající ve "sbalení" shodových kategorií do speciální "shodové" struktury rysů, která ale žádné rysy nemá, nese jenom unifikaci typů).

[22] Z takové možné libovůle je vidět, že formalismus zřejmě zdaleka není ideální a měl by projít dalším vývojem. První návrhy takových úprav lze najít v (Moshier 1997a,b) nebo (Oliva 2002), obecně se ale neujaly.

$$\text{(UZG)} \quad \text{a.} \quad \begin{bmatrix} \textit{veta} & \\ \text{podmet}: & \begin{bmatrix} \textit{podst\_jm} \vee \textit{zájmeno} \\ \text{osoba}: \boxed{1} \\ \text{císlo}: \boxed{2} \\ \text{rod}: \boxed{3} \end{bmatrix} \\ \text{prísudek}: & \begin{bmatrix} \textit{sloveso} \\ \text{osoba}: \boxed{1} \\ \text{císlo}: \boxed{2} \\ \text{rod}: \boxed{3} \end{bmatrix} \end{bmatrix} \qquad \text{b.} \quad \begin{bmatrix} \textit{veta} & \\ \text{podmet}: & \begin{bmatrix} \textit{podst\_jm} \vee \textit{zájmeno} \\ \text{shoda}: \boxed{1} \begin{bmatrix} \text{os3 Y jedn Y muz\_ziv} \end{bmatrix} \end{bmatrix} \\ \text{prísudek}: & \begin{bmatrix} \textit{sloveso} \\ \text{shoda}: \boxed{1} \end{bmatrix} \end{bmatrix}$$

Na závěr této kapitoly ještě dodejme, že pokud má typování sloužit také ke kontrole správnosti (vstupních) dat, je rozumné typovat nejenom struktury, ale i stanovit typy (tj. přípustné hodnoty) jednotlivých rysů. Tak např. (TTT) zřejmě obsahuje nesmyslnou hodnotu rysu *barva* – zřejmě je rozumné omezit hodnoty rysů na určité množiny (zde by mohlo jít např. o množinu *{bílá,žlutá,modrá,červená,zelená,černá}*). Toto je však zřejmě příklad jiného typování, než o kterém jsme mluvili dosud – jde vlastně o něco jako o deklarace oboru hodnot (jak je to běžné např. v některých programovacích jazycích).

(TTT) [barva : jehlan]

8. Funkční/relační závislosti mezi hodnotami rysů v sestavě

Pokud budeme na rysy a jejich hodnoty pohlížet tak, jak je naznačeno v posledních řádcích minulé kapitoly, nemělo by nám ujít, že se vlastně jedná o pohled jakoby na funkce (mají např. obor hodnot). Z tohoto hlediska pak může začít být nápadné, že prozatím jsme využívali jen ty "nejjednodušší" funkce: konstantu (při přiřazení nějakého pevného objektu jako hodnoty rysu) a identitu (v případě sdílení – tj. právě identity – hodnoty jednoho rysu s hodnotou rysu jiného). To vede k myšlence rozšíření možných hodnot o hodnoty funkcionálně definované a ještě obecněji k rozšíření sestav rysů na sestavy rysů s relačními závislostmi mezi hodnotami rysů.
Velmi přímou a jednoduchou motivaci pro funkcionálně definované hodnoty můžeme nalézt v již dobře známém geometrickém světě. Představme, si že v něm popisujeme nějakou množinu krychlí, mezi jejichž vlastnosti (tj. rysy) patří mj. *délka_hrany* a *objem*. Zatímco každý jednotlivý popis konkrétní krychle má jako hodnoty těchto rysů pevná čísla (jako např. v (LMB) – je třeba si uvědomit, že π je konstanta, tedy π^3 je také konstanta), nelze dosavadními prostředky obecný vztah mezi délkou hrany krychle a jejím objemem (jakožto třetí mocninou proměnné) vyjádřit.

$$\text{(LMB)} \quad \text{a.} \begin{bmatrix} \textit{krychle} \\ \text{délka\_hrany}: 1 \\ \text{objem}: 1 \end{bmatrix} \quad \text{b.} \begin{bmatrix} \textit{krychle} \\ \text{délka\_hrany}: 2{,}5 \\ \text{objem}: 15{,}625 \end{bmatrix} \quad \text{c.} \begin{bmatrix} \textit{krychle} \\ \text{délka\_hrany}: \pi \\ \text{objem}: \pi^3 \end{bmatrix}$$

Je ovšem zřejmé, že tuto situaci lze snadno napravit tím, že připustíme, aby se platnými hodnotami rysů mohly stát i funkcionální výrazy (tj. výrazy, které definují hodnotu jako funkci hodnot jiných rysů v sestavě). V uvedeném případě vypadá pak popis "obecné" krychle pomocí sestavy rysů tak, jak je uvedeno v (WKM).

(WKM)
$$\begin{bmatrix} \textit{krychle} \\ \text{délka\_hrany} : \boxed{1} \\ \text{objem} : \boxed{1}^3 \end{bmatrix}$$

Obecně však samozřejmě nejsou funkční hodnoty omezeny na funkce jedné reálné proměnné: může jít o funkce více argumentů, kterými mohou být čísla, atomy, sestavy rysů ...

Je ovšem potřeba si uvědomit, že takovéto rozšíření přináší do formalismu novou kvalitu: kromě vlastních sestav rysů je (alespoň v případě funkcí, které nejsou "samozřejmé") nutné definovat i předpisy funkcí – i ty se totiž stávají součástí popisu objektů (jinak by nebylo možné funkce použít). Jako lingvistický příklad je možné uvést koordinační strukturu dvou podstatných jmen rodu středního. Výsledný rod takové koordinace (např. pro účely shody se slovesem nebo se vztažným zájmenem) je závislý na číslech původních podstatných jmen – je-li alespoň jedno z nich v čísle množném, je výsledkem také rod střední, pokud jsou však obě v jednotném čísle, je výsledný rod ženský[23], srv. "*štěně a kotě běhaly/*běhala po dvoře"*. Příslušná struktura by tedy mohla vypadat např. jako (KEN), definice funkce (ve formátu odpovídající definicím funkcí v Prologu) je pak uvedena v (KBN) (proměnná |0| je zde použita jako "anonymní", tj. nemající jinou funkci než zaplnit argumentovou pozici, aniž by přitom na její hodnotě jakkoliv záleželo).

(KEN)
$$\begin{bmatrix} \textit{koordinace Y podst\_jm} \\ \text{rod} : \text{výsl\_rod\_koordinace\_neuter}(\boxed{1}, \boxed{2}) \\ \text{číslo} : \text{pl} \\ \text{levý\_člen} : \begin{bmatrix} \textit{podst\_jm} \\ \text{rod} : \text{neut} \\ \text{číslo} : \boxed{1} \end{bmatrix} \\ \text{pravý\_člen} : \begin{bmatrix} \textit{podst\_jm} \\ \text{rod} : \text{neut} \\ \text{číslo} : \boxed{2} \end{bmatrix} \end{bmatrix}$$

(KBN)
výsl_rod_koordinace_neuter(pl, |0|) = neut.
výsl_rod_koordinace_neuter(|0|, pl) = neut.
výsl_rod_koordinace_neuter(sg, sg) = fem.

Předchozí příklady na funkční hodnoty rysů mají dvě zásadní vady:

- z hlediska intuitivního porozumění je problematické to, že takové funkcionální pojetí působí dojmem, že jedna z hodnot je jaksi "základní" (např. *délka_hrany* krychle) a druhá je jen "odvozená" (v uvedeném příkladu *objem* krychle); tak to ale samozřejmě v realitě není, obě hodnoty jsou pouze v určitém vztahu, který ale nemá žádný "směr" (v příkladu s krychlí je to zřetelné z toho, že délku hrany

[23] Nebo mužský neživotný – v plurálové shodě totiž mezi nimi není rozdíl, takže volba je zde libovolná.

lze jednoznačně zjistit z objemu prostým výpočtem třetí odmocniny, tj. "směr" závislosti by musel být vlastně přesně opačný – z čehož tedy plyne, že zřejmě není vůbec žádný)

- z hlediska formálního je nevyhovující, že pojem funkcionálního vztahu s sebou automaticky nese nutnou jednoznačnost; vztahy v jazyce však nejsou vždy takto jednoznačné (např. z informace, že základním tvarem skloňovaného substantiva je "*rybník*" a že jeho pádem je *lokál* a číslem je *singulár*, nelze výsledný tvar jednoznačně odvodit: spravné jsou oba tvary, "*rybníku*" i "*rybníce*"; podobné, i když složitější příklady se ovšem celkem snadno dají najít i v syntaxi).

Proto se jako adekvátnější alternativa k funkcionálním hodnotám rysů zavadějí relační omezení (angl. *"relational constraints"*). Každé takové omezení je vlastně predikátový výraz s argumenty, kterými jsou nějaké hodnoty rysů (nejčastěji samozřejmě proměnné). Aby byla sestava, se kterou je takové relační omezení asociováno (což bude vyznačeno symbolem konjunkce $\wedge$), platná (správná, správně tvořená), musí příslušný predikátový výraz nabývat hodnotu *pravda*. Triviální příklad relačního omezení, popisujícího vztahy mezi rozměry a objemem kvádru, je uveden v (LSR). Příklad má dvě části: první je sestava rysů s asociovaným omezením (asociace je vyznačena znaménkem $\wedge$), druhou je definice tohoto omezení[24].

(LSR)

$$\begin{bmatrix} \text{kvádr} \\ \text{délka\_přední\_strany} : |1| \\ \text{délka\_boční\_strany} : |2| \\ \text{výška} : |3| \\ \text{objem} : |4| \end{bmatrix} \wedge \text{objem\_kvádru}(|1|,|2|,|3|,|4|)$$

objem_kvádru(|1| , |2| , |3| , |1|*|2|*|3|).

Lingvistický příklad je uveden v (ORT), kde je relačním omezením vyjádřena skutečnost, že grafématický zápis celé věty je sřetězením (konkatenací) grafématických zápisů jejích jednotlivých větných členů (rozumí se v příslušném pořadí). V definici konkatenace (concatenation) je použita opět "prologovská" notace, řetězec prvků je uzavřen mezi "špičaté" závorky "<" a ">", prázdný řetězec je tedy značen jako "< >", první prvek řetězce (hlava) a zbytek řetězce jsou od sebe odděleny tečkou "." .

(ORT)

$$\begin{bmatrix} \text{veta} \\ \text{grafematický\_zápis} : |1| \\ \text{větné\_členy} : \begin{bmatrix} \text{podmetova\_cast} : [\text{grafematicky\_zapis} : |2|] \\ \text{prisudkova\_cast} : [\text{grafematicky\_zapis} : |3|] \end{bmatrix} \end{bmatrix} \wedge \text{concatenation}(|2|,|3|,|1|)$$

concatenation(< > , < > , < >).
concatenation(<|1|.|2|> , |3| , <|1|.|4|>) if concatenation(|2| , |3| , |4|).

Je potřeba si uvědomit, že je opravdu důležité, aby vztah mezi grafématickými zápisy zde byl definován relacionálně, nikoliv funkcionálně. Zaprvé totiž není zřejmé, co je argumentem a co výsledkem (popis může být

[24] V této stati se nebudeme zabývat otázkou, jak by se zjišťovalo v nějakém konkrétním případě, zda je omezení splněno. Standardní postup by asi byl použít metod CLP (constraint logic programming), myslitelné jsou jistě i jiné metody, každopádně však tato otázka zde není relevantní.

použit jak pro generování, tak pro analýzu, přesněji pro analýzu zdola), jednak výsledek není dán jednoznačně (srv. např. dva možné výsledky rozdělení věty *Účastníci soutěží odpovědí na třetí otázku.* na podmětnou a přísudkovou část).

9. Závěr

V předchozím textu byly představeny základní vlastnosti formalismu sestav rysů, který je používán v moderních teoriích syntaktického popisu. Důraz byl přitom kladen na vysvětlení intuitivního základu tohoto formalismu a jeho jednotlivých komponent, formální hledisko (přesné definice, věty, důkazy atd.) bylo záměrně ponecháno v pozadí či dokonce zcela stranou. Podstatné přitom je, že zatímco v oblasti přesné formalizace existuje relativně bohatá literatura (nejvýznamnější je zřejmě kniha (Carpenter, 1992)), je prvotní (a proto nejdůležitější) porozumění sestavám rysů v podstatě jen součástí jakéhosi folklóru matematických lingvistů a všeobecně se předpokládá, že "vše je přeci jasné ..." Žádné pojednání či kniha, která by sloužila jako skutečný úvod do problematiky, proto zřejmě neexistuje (a zcela jistě neexistuje v češtině). Tuto mezeru se snaží zaplnit tato stať, která si ale samozřejmě nedělá nárok na úplnost (ani v oblasti intuitivních základů). Nejsou popsána další možná rozšíření sestav rysů, chybí mnohé oblasti použití předvedených formálních prostředků (např. vyjádření lingvistických principů jako implikací mezi sestavami rysů), zcela byly vynechány možnosti a metody rozšíření hierarchie sestav rysů pro účely popisu hierarchického slovníku atd. Všechny tyto další, speciální aspekty, lze však již dohledat v odborné literatuře.

Literatura:

Carpenter Bob: The Logic of Typed Feature Structures, Cambridge Tracts in Theoretical Computer Science 32, Cambridge University Press 1992

King Paul J.: Typed Feature Structures As Descriptions, Proceedings of Coling 1994, Kyoto

Moshier M.A.: Featureless HPSG, in: P. Blackburn and M. de Rijke (eds.): Logic, Structures and Syntax, Reidel, Dordrecht 1997a

Moshier M.A.: Is HPSG Featureless or Unprincipled ? in: Linguistics and Philosophy vol. 20 Nr. 6, 1997b

Oliva Habilitace

Pollard C. and I.A. Sag: Head-driven Phrase Structure Grammar, Chicago University Press 1994

http://www.colloquial.com/carp/Publications/

http://www.sfs.nphil.uni-tuebingen.de/~dg