MODEL METÓDY (3): INŠTRUKCIA A METÓDA[1]

LUKÁŠ BIELIK, MILOŠ KOSTEREC, MARIÁN ZOUHAR, Katedra logiky a metodológie vied Filozofickej fakulty UK, Bratislava, SR

BIELIK, L., KOSTEREC, M. ZOUHAR, M.: The Model of Method (3): Instruction and Method
FILOZOFIA 69, 2014, No 8, pp. 637-652

The present article is the third part of a longer paper in which we outline a model of (scientific) method as a system of instructions aimed at a certain kind of (cognitively interesting) goal. Building on the results of the previous part concerning the notions of instruction and its occurrence, the present article specifies the ways of chaining the occurrences. The occurrences of instructions constitute linear chains if involving only the occurrences of categorical or simple hypothetical instructions; a chain is nonlinear provided there is at least one complex hypothetical instruction in it. Every chain of occurrences can be represented as a sequence of postulate and derivate transitions. The method is represented as an oriented graph consisting of the chains of occurrences of instructions. We specify various formal and informal constraints that are to be met by a graph if it is to be taken as a representation of a method. Finally, we describe a link between the model of method proposed in this part and our intuitive specification of method as a kind of problem solving activity given in the first part of our paper.

Keywords: Chain of occurrences of instructions – Compound instruction – Derivate transition – Method – Occurrence of an instruction – Oriented graph – Postulate transition

V predchádzajúcich pokračovaniach tejto state (pozri Bielik, Kosterec, Zouhar 2014a; 2014b)[2] sme charakterizovali niektoré dôležité intuície, teoretické pojmy a ďalšie prostriedky potrebné na modelovanie metódy. Úvahy o inštrukciách a vzťahoch medzi ich výskytmi zrejme dávajú tušiť, akým smerom sa vyberieme pri modelovaní (vedeckých) metód v tomto pokračovaní. Využijeme na to niektoré prostriedky teórie grafov.

13. Reťazenie výskytov inštrukcií. Ako sme videli, inštrukcie modelujeme ako relácie pomocou vstupných a výstupných stavov a výskyty inštrukcií ako konkrétne dvojice takýchto stavov. Teraz sa pozrieme na to, ako možno vytvárať reťazce výskytov inštrukcií, t. j. komplexnejšie sústavy obsahujúce viac ako jeden výskyt. Reťazenie výsky-

[1] Chceme poďakovať Pavlovi Cmorejovi, Marii Duží, Františkovi Gahérovi, Daniele Glavaničovej, Jurajovi Halasovi, Igorovi Hanzelovi, Vladimírovi Markovi, Martinovi Vacekovi a Marekovi Vicianovi za pripomienky k predchádzajúcim verziám state a za podnetné diskusie.

[2] Do druhého pokračovania sa nám vkradla jedna chyba, ktorú sme odstránili v errátach uverejnených v časopise *Filozofia* 69, 2014, č. 6, s. 547.

tov inštrukcií opíšeme pomocou pojmov nadväznosti a nezávislosti z 11. podkapitoly. Špecifikácia jednotlivých výskytov pomocou vstupných a výstupných stavov ponúka jednoduchý návod na tvorbu komplexnejších sústav výskytov.

Opäť začnime jednoduchým príkladom. V 11. podkapitole sme uvažovali o úlohe vypočítať ($a + b$), kde a a b boli prirodzené čísla. Splniť túto úlohu znamená vykonať inštrukciu „Sčítaj čísla a a b!“. Teraz si vezmime inú úlohu: vypočítať ($a + b$) + a. Splniť ju znamená vykonať dve inštrukcie: (*i*) „Sčítaj čísla a a b!“; (*ii*) „Pripočítaj k výsledku číslo a!“. Nech vstupným stavom výskytu prvej inštrukcie je S_1 a výstupným stavom je S_2, pričom ich vymedzenie môžeme prevziať z 11. podkapitoly; ($On(S)$ je ontológia stavu, $Op(S)$ je množina operácií stavu a $Pr(S)$ je množina propozícií stavu):

$On(S_1) = \{a, b\}$	$On(S_2) = \{a, b\} \cup \{k\}$
$Op(S_1) = \{+\}$	$Op(S_2) = \{+\}$
$Pr(S_1) = \emptyset$	$Pr(S_2) = \emptyset \cup \{a + b = k\}$

Vykonanie druhej inštrukcie (v danom výskyte) predpokladá, že sa najprv vykoná prvá inštrukcia (v predchádzajúcom výskyte). Preto vstupný stav výskytu druhej inštrukcie, ktorý označíme S_3, by mal byť totožný s výstupným stavom výskytu prvej inštrukcie, t. j. s S_2. Vidíme, že druhá inštrukcia pracuje len s tými entitami (objektmi a operáciami), ktoré sú už obsiahnuté v S_2 (a teda aj v S_3), a preto nepotrebujeme nové postuláty.[3] Vykonaním tejto inštrukcie (v danom výskyte) získame nový stav S_4, ktorý je výstupným stavom daného výskytu a bude bohatší o nový objekt z ontológie, číslo l, ktoré dostaneme pripočítaním čísla a k číslu k, a o novú informáciu zachytávajúcu tento vzťah. Stavy S_3 a S_4 teda môžeme vymedziť takto:

$On(S_3) = \{a, b\} \cup \{k\}$	$On(S_4) = \{a, b\} \cup \{k\} \cup \{l\}$
$Op(S_3) = \{+\}$	$Op(S_4) = \{+\}$
$Pr(S_3) = \emptyset \cup \{a + b = k\}$	$Pr(S_4) = \emptyset \cup \{a + b = k\} \cup \{k + a = l\}$

Stav S_4 sa od stavu S_3 líši novými derivátmi, teda odvodenými entitami. Tento spôsob reťazenia výskytov inštrukcií spočíva v tom, že výstupný stav výskytu jednej inštrukcie sa stotožní so vstupným stavom výskytu nadväzujúcej inštrukcie. Ide o jednoduchý spôsob, ktorý však má svoje limity, keďže nedokážeme pomocou neho zachytiť všetky prípady.[4]

Vezmime si iný príklad: vypočítať ($a + b$) – c. Splniť túto úlohu znamená vykonať dve inštrukcie: (*i*) „Sčítaj čísla a a b!“; (*ii*) „Odpočítaj od výsledku číslo c!“. Vstupný aj výstupný stav výskytu prvej inštrukcie možno stotožniť so stavmi S_1, resp. S_2 z predchádzajúceho príkladu. Označme ich ako S'_1, resp. S'_2. Vstupný stav výskytu druhej inštrukcie však už nemôže byť totožný s S'_2 (ako to bolo v predchádzajúcom príklade). Druhá inštrukcia sa totiž zmieňuje o entitách (objektoch a operáciách), ktoré nemáme

[3] Pripomíname, že postuláty sú také entity zo vstupného, resp. výstupného stavu, ktoré sa predpokladajú. Zvyčajne sa postuláty do stavov dostanú vďaka tomu, že sa o nich zmieňujú inštrukcie.

[4] Pripomíname, že postulátový prechod môže byť triviálny, teda vstupný stav môže byť totožný s výstupným stavom, ako je to v prípade prechodu medzi stavmi S_2 a S_3. Výstupný stav výskytu inštrukcie však vždy musí byť netriviálnym rozšírením zodpovedajúceho vstupného stavu.

v množinách $On(S'_2)$ a $Op(S'_2)$. Vstupný stav jej výskytu teda musí byť postulátovým rozšírením výstupného stavu výskytu prvej inštrukcie. Vďaka postulátovému prechodu rozšírime množiny $On(S'_2)$ a $Op(S'_2)$ na množiny $On(S'_3) = \{a, b\} \cup \{k\} \cup \{c\}$, resp. $Op(S'_3) = \{+\} \cup \{-\}$, teda doplníme ontológiu a operácie potrebnými entitami. Takto získame vstupný stav výskytu druhej inštrukcie, t. j. stav S'_3; vykonaním inštrukcie v danom výskyte dostaneme výstupný stav, ktorý sa od S'_3 líši novými derivátmi, a to konkrétne číslom l, ktoré dostaneme odpočítaním c od k, a novou propozíciou zachytávajúcou tento vzťah. Keď to skompletizujeme, získame nasledujúce štyri stavy, kde S'_1 a S'_2 predstavujú vstupný, resp. výstupný stav výskytu prvej inštrukcie a S'_3 a S'_4 predstavujú vstupný, resp. výstupný stav výskytu druhej inštrukcie:

$$
\begin{array}{ll}
On(S'_1) = \{a, b\} & On(S'_2) = \{a, b\} \cup \{k\} \\
Op(S'_1) = \{+\} & Op(S'_2) = \{+\} \\
Pr(S'_1) = \emptyset & Pr(S'_2) = \emptyset \cup \{a + b = k\} \\
& \\
On(S'_3) = \{a, b\} \cup \{k\} \cup \{c\} & On(S'_4) = \{a, b\} \cup \{k\} \cup \{c\} \cup \{l\} \\
Op(S'_3) = \{+\} \cup \{-\} & Op(S'_4) = \{+\} \cup \{-\} \\
Pr(S'_3) = \emptyset \cup \{a + b = k\} & Pr(S'_4) = \emptyset \cup \{a + b = k\} \cup \{k - c = l\}
\end{array}
$$

Vo všeobecnosti možno povedať, že dva výskyty inštrukcií sú zreťazené, ak existuje postulátový prechod medzi výstupným stavom jedného z nich a vstupným stavom druhého z nich. Ak výstupný stav prvého výskytu je totožný so vstupným stavom druhého výskytu, postulátový prechod je triviálny, resp. ide o reláciu identity, do ktorej vstupuje stav so sebou samým. Triviálny postulátový prechod je limitným prípadom postulátového prechodu. Takéto reťazenie inštrukcií je *lineárne*, utvára teda postupnosť dvoch či viacerých výskytov, v ktorej neexistujú žiadne dva vzájomne nezávislé výskyty. To znamená, že pre výstupný stav výskytu inštrukcie existuje najviac jedno jeho postulátové rozšírenie na vstupný stav výskytu ďalšej inštrukcie. Zreťazenie výskytov inštrukcií môžeme zachytiť pomocou jednoduchého obrázku (prerušované šípky znázorňujú postulátové prechody, plné šípky zase derivátové prechody, kruhy predstavujú stavy; pripomíname, že zápis $[i_x]_y$ znamená y-tý výskyt inštrukcie i_x):

$$\circ \xrightarrow{[i_a]_j} \circ \dashrightarrow \circ \xrightarrow{[i_b]_k} \circ \dashrightarrow \circ \xrightarrow{[i_c]_l} \circ$$

Vidíme, že reťazenie výskytov inštrukcií spočíva v striedaní postulátových a derivátových prechodov: Výskyt inštrukcie je derivátovým prechodom a postulátový prechod má zase za úlohu upraviť určitý stav tak, aby sa z neho stal vstupný stav ďalšieho výskytu inštrukcie. Postulátový prechod môže spočívať v obohatení ontológie a/alebo množiny operácií (no niekedy môže byť triviálny) a derivátový prechod spočíva v obohatení množiny propozícií a (niekedy aj) ontológie. V nasledujúcej tabuľke sú v schematickej podobe zachytené prechody medzi stavmi v prípade reťazca výskytov inštrukcií (symbol „+“ znamená obohatenie v porovnaní s predchádzajúcim stavom, „0“ znamená v porovnaní s predchádzajúcim stavom absenciu zmeny a „+/0“ zachytáva možnosť, že v porovnaní s predchádzajúcim stavom mohlo, no nemuselo dôjsť k obohateniu):

S_1	S_2	S_3	S_4	S_5	…
$On(S_1)$	+/0	+/0	+/0	+/0	…
$Op(S_1)$	+/0	0	+/0	0	…
$Pr(S_1)$	0	+	0	+	…

S_2 je postulátovým rozšírením S_1, keďže množiny objektov a operácií z S_1 sa mohli (no nemuseli) vhodným spôsobom rozšíriť a množina propozícií zostáva nezmenená. S_3 je derivátovým rozšírením S_2, keďže sa obohacuje množina propozícií a môže (no nemusí) sa obohatiť ontológia. To isté striedanie zaznamenáme aj v prípade ďalších stavov. Táto tabuľka zachytáva akýsi všeobecný „vzorec" striedania stavov pri reťazení výskytov inštrukcií. Vidno, že pri derivátovom prechode sa nahrádza množina propozícií bohatšou množinou, pričom množina operácií sa nemení; pri postulátovom prechode sa zase nemení množina propozícií, no ostatné množiny môžu byť nahradené bohatšími množinami. Dôležitou skutočnosťou je to, že striedanie postulátových a derivátových prechodov môžeme jednoznačne identifikovať na základe striedania symbolov „0" a „+" v riadku reprezentujúcom množinu propozícií daných stavov.

14. Zložené inštrukcie. Ako kategorické (t. j. konjunktívne a disjunktívne) a hypotetické zložené inštrukcie ovplyvnia naše rozlíšenie inštrukcií a výskytov inštrukcií a modelovanie výskytov ako derivátových prechodov? Tvrdíme, že ak chceme do doterajšieho systému zakomponovať aj zložené inštrukcie, netreba ho nijako osobitne obohacovať.

Najprv sa pozrime na konjunktívne a disjunktívne inštrukcie (i_b & i_c), resp. ($i_b \vee i_c$). Môžeme ich považovať za celok, a to znamená, že výskytu konjunktívnej, resp. disjunktívnej inštrukcie zodpovedá na obrázku jedna plná šípka. Treba doplniť len určité špecifické dodatky týkajúce sa vstupných a výstupných stavov výskytov takýchto inštrukcií. Vstupný stav výskytu konjunktívnej inštrukcie $[(i_b \,\&\, i_c)]_k$ aj výskytu disjunktívnej inštrukcie $[(i_b \vee i_c)]_l$ získame z výstupného stavu výskytu nejakej inštrukcie $[i_a]_j$ tak, že ho postulátovo rozšírime o *všetky* nové entity, o ktorých sa zmieňujú obidve konštitutívne inštrukcie i_b a i_c z výskytov $[(i_b \,\&\, i_c)]_k$, resp. $[(i_b \vee i_c)]_l$. Výstupný stav výskytu konjunktívnej inštrukcie bude obsahovať ako nové deriváty (*i*) objekty, ktoré by sme dostali, keby sme inštrukcie i_b a i_c vykonali samostatne, a (*ii*) konjunkciu propozícií, ktoré by sme dostali, keby sme inštrukcie i_b a i_c vykonali samostatne, takže množina propozícií výstupného stavu je bohatšia o jednu konjunktívnu propozíciu. Výstupný stav výskytu disjunktívnej inštrukcie bude obsahovať ako nové deriváty (*i*) objekty, ktoré by sme dostali, keby sme inštrukcie i_b a i_c vykonali samostatne, a (*ii*) disjunkciu propozícií, ktoré by sme dostali, keby sme inštrukcie i_b a i_c vykonali samostatne, takže množina propozícií výstupného stavu je bohatšia o jednu disjunktívnu propozíciu. Konjunktívnu ani disjunktívnu inštrukciu nechápeme ako dvojice nezávislých inštrukcií i_b a i_c, pričom v prípade výskytu konjunktívnej inštrukcie by platilo, že sa majú vykonať oba konjunkty, a v prípade výskytu disjunktívnej inštrukcie by zase platilo, že sa má vykonať aspoň jeden z disjunktov.

Rozdiel medzi nimi spočíva len v tom, že výstupný stav výskytu takejto inštrukcie je bohatší o určitú špecifickú propozíciu.[5]

Hypotetické inštrukcie sú problematickejšie. Kvôli jednoduchosti si vezmime inštrukciu formy $(\pi \rightarrow i_a)$, kde π je atomárna propozícia a i_a atomárny imperatív.[6] Ide síce o inštrukciu, no povahu imperatívu má len konzekvent i_a, a teda len konzekvent je inštrukciou v pravom zmysle slova. Z tohto dôvodu nejaký vstupný stav, resp. výstupný stav obohatený vykonaním danej inštrukcie má len výskyt imperatívneho konzekventu. Podľa nášho chápania kondicionálne pripojenie propozičného antecedenta π k inštrukcii i_a špecifikuje *test* určitého druhu. Konkrétne, treba testovať, či množina propozícií zo vstupného stavu výskytu inštrukcie $[i_a]_j$ obsahuje propozíciu π; inštrukcia i_a z výskytu $[i_a]_j$ sa má vykonať len v prípade, že jeho vstupný stav obsahuje π. Pri hypotetických inštrukciách teda testujeme, či sme vykonaním predchádzajúcich inštrukcií dostali takú množinu propozícií, ktorá obsahuje určitú relevantnú propozíciu. V prípade, že sa daná propozícia vo vstupnom stave nachádza, môžeme vykonať inštrukciu z konzekventa; ak sa v ňom nenachádza, inštrukciu vykonať nemôžeme.[7] Obrázok, ktorý by mal reprezentovať výskyt hypotetickej inštrukcie, bude preto obsahovať len výskyt jej konzekventa:[8]

$$\circ \dashrightarrow \circ \xrightarrow{[i_a]_j} \circ \dashrightarrow \circ$$

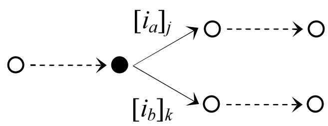
[5] Aby sa inštrukcie i_b a i_c vôbec mohli vyskytovať v zložených inštrukciách $(i_b \,\&\, i_c)$ a $(i_b \cup i_c)$, musia spĺňať určité obmedzenia. Dôležité je to, aby sa ani i_b, ani i_c neodvolávala na *výsledok* vykonania druhej z nich. Takáto situácia nastáva napríklad v súvislosti s inštrukciami „Sčítaj čísla *a* a *b*!“ a „Od výsledku odpočítaj číslo *c*!“, kde druhá inštrukcia predpokladá vykonanie prvej inštrukcie. Tieto inštrukcie nemožno vykonať súbežne. Nemôžeme z nich teda utvoriť zloženú inštrukciu „Sčítaj čísla *a* a *b* a od výsledku odpočítaj číslo *c*!“, no musí ísť o dve samostatné, na seba nadväzujúce inštrukcie. Pripúšťame teda len také zložené inštrukcie, ktorých zložky možno vykonať súbežne, teda bez toho, že by jedna z nich predpokladala výsledok vykonania druhej z nich.

[6] Pojem atomárnej inštrukcie je analogický pojmu atomárnej propozície. Ide o inštrukciu, ktorá neobsahuje ako svoje zložky ďalšie inštrukcie. Inštrukcie formy $(i_a \,\&\, i_b)$, $(i_a \cup i_b)$ a $(\pi \rightarrow i_a)$ atomárne nie sú.

[7] Test môžeme chápať ako položenie otázky a hľadanie odpovede. V prvom pokračovaní sme otázku modelovali ako funkciu určitého druhu. V tomto prípade ide o funkciu, ktorú aplikujeme na množinu propozícií zo vstupného stavu výskytu inštrukcie; ide o zisťovaciu otázku, takže odpoveďou je buď „Áno“, alebo „Nie“. Konkrétnejšie, pýtame sa „π?“ a odpoveďou je buď „π“, alebo „$\sim\pi$“. (Pripomíname, že v predchádzajúcom pokračovaní sme pomocou symbolu „$\sim$“ v „$\sim\pi$“ zachytávali neprítomnosť propozície π v danom stave, pričom nepravdivosť propozície sme explikovali na základe jej neprítomnosti.) Vykonať inštrukciu v konzekvente môžeme len v prípade, že odpoveďou je „π“.

[8] Podobne sa modelujú podmienené prechody v tzv. Petriho sieťach. Petriho sieť je modelovací nástroj na analýzu dynamických systémov. Tie sa môžu nachádzať v rôznych stavoch, medzi ktorými sa premiestňujú pomocou prechodov. Petriho sieť je orientovaný graf. Vrcholy grafu sa delia na *miesta* (ktorým sa v našom návrhu podobajú stavy) a *prechody*. Hrany v grafe Petriho siete sú orientované a buď spájajú miesto s prechodom, alebo spájajú prechod s miestom. Miesta v Petriho sieti môžu obsahovať tzv. *tokeny*, pričom prechod medzi miestami je podmienený určitým počtom tokenov na daných miestach. Prechody sú takto vo všeobecnosti podmienené určitou vlastnosťou miesta, z ktorého vychádzajú. Viac k Petriho sieťam pozri napríklad (Murata 1989).

Zložitejší prípad hypotetickej inštrukcie predstavujú vety formy „Ak π, tak i_a, inak i_b“.[9] Táto inštrukcia fakticky hovorí, že i_a máme vykonať v prípade, že π, a i_b máme vykonať v prípade, že ~π. Explicitnejšie môžeme túto skutočnosť zachytiť preformulovaním pôvodnej vety na vetu „Ak π, tak i_a, a ak ~π, tak i_b“. Ide o kombináciu dvoch hypotetických inštrukcií predchádzajúceho druhu, teda o inštrukciu formy $(\pi \rightarrow i_a)$ & $(\sim\pi \rightarrow i_b)$. Na rozdiel od inštrukcie formy $(\pi \rightarrow i_a)$, na základe ktorej sme nemohli vykonať nič, ak sme najprv nezistili, že množina propozícií zo vstupného stavu jej výskytu obsahuje určitú propozíciu, v prípade inštrukcie formy $(\pi \rightarrow i_a)$ & $(\sim\pi \rightarrow i_b)$ môžeme niečo vykonať aj v prípade, že množina propozícií zo vstupného stavu jej výskytu neobsahuje určitú propozíciu: Máme vykonať alternatívnu inštrukciu. Opäť teda testujeme množinu propozícií vstupného stavu a podľa výsledku testu máme vykonať buď jednu, alebo druhú inštrukciu. Testovaním v podstate zisťujeme, či ide o vstupný stav výskytu $[i_a]_j$, alebo o vstupný stav výskytu $[i_b]_k$. Ponúka sa nasledujúca grafická reprezentácia:



Na tomto obrázku sa objavuje stav (znázornený vyplneným kruhom), ktorý je bodom vetvenia, keďže z neho vychádzajú dve samostatné vetvy, pričom jedna začína výskytom $[i_a]_j$ a druhá výskytom $[i_b]_k$. Testovaním vstupného stavu (vyplnený kruh) zistíme, ktorou vetvou máme ďalej pokračovať. Výskyty $[i_a]_j$ a $[i_b]_k$ sú navzájom nezávislé (v zmysle definície z 12. podkapitoly), keďže vykonanie inštrukcie z jedného výskytu nie je podmienkou vykonania inštrukcie z druhého výskytu. Prirodzene, ak v postupnosti existuje taký výskyt $[i_c]_l$, pre ktorý platí, že nadväzuje na $[i_a]_j$ (a nie na $[i_b]_k$), tak $[i_c]_l$ je nezávislý od $[i_b]_k$. V jednotlivých vetvách teda budú nasledovať vzájomne nezávislé výskyty inštrukcií. Nezávislosť výskytov je príznakom toho, že ich zreťazenie je *nelineárne*.[10] Budeme uvažovať len o takomto druhu nelineárneho zreťazenia výskytov inštrukcií, ktoré je dané inštrukciami formy $(\pi \rightarrow i_a)$ & $(\sim\pi \rightarrow i_b)$.

Doterajšie poznámky o inštrukciách a ich výskytoch môžeme zhrnúť v podobe nasledujúcich tvrdení:

---

[9] V Transparentnej intenzionálnej logike (ale aj v informatike) sa analyzuje *výroková* spojka „ak…, tak…, inak…“ (pozri napríklad Duží, Jespersen, Materna 2010, 263; Duží 2010). V našom prípade však „ak…, tak…, inak…“ nie je výrokovou spojkou (hoci k nej má veľmi blízko), keďže pomocou nej možno utvoriť vety obsahujúce rozkazovacie vety ako svoje súčasti.

[10] Ľahko pochopíme, že jednotlivé vetvy predstavujúce postupnosti nezávislých výskytov inštrukcií sa nespoja; t. j. neexistuje taký stav, ktorý by bol bodom spojenia jednotlivých vetiev. Podľa povahy inštrukcie formy $(\pi \rightarrow i_a)$ & $(\sim\pi \rightarrow i_b)$ totiž pokračujeme buď vykonaním inštrukcie i_a, alebo vykonaním inštrukcie i_b, no nie vykonaním obidvoch inštrukcií. Keby existoval bod spojenia vetiev, znamenalo by to, že sa vyžaduje vykonanie obidvoch inštrukcií. Lenže v takom prípade by vo vstupnom stave museli byť pravdivé obidve propozície π a ~π, čo je nemožné.

1. Vstupným stavom a výstupným stavom výskytu inštrukcie sú množiny, ktoré obsahujú ako svoje prvky (*i*) množinu objektov (ontológiu stavu), (*ii*) množinu operácií aplikovateľných na prvky ontológie a (*iii*) množinu propozícií opisujúcich vzťahy medzi prvkami ontológie.

2. Prvkami množín objektov, operácií a propozícií sú postuláty a/alebo deriváty, pričom platí, že postulátmi môžu byť objekty a operácie a derivátmi zase objekty a propozície.

3. Výskyty inštrukcií možno reťaziť na základe postulátového prechodu, ak sa výstupný stav jedného výskytu inštrukcie postulátovo rozšíri na vstupný stav iného výskytu inštrukcie.

4. Nasledujúci stav sa od predchádzajúceho stavu líši postulátovými alebo derivátovými prechodmi, pričom *a*) nasledujúci vstupný stav sa od predchádzajúceho výstupného stavu odlišuje len postulátovými prechodmi a *b*) nasledujúci výstupný stav sa od predchádzajúceho vstupného stavu odlišuje len derivátovými prechodmi.

5. Zloženosť (kategorickej) inštrukcie netreba modelovať pomocou zreťazenia výskytov dvoch alebo viacerých atomárnych inštrukcií, ale odráža sa len v tom, že súčasťou množiny propozícií výstupného stavu jej výskytu je zložená konjunktívna alebo disjunktívna propozícia.

6. Hypotetickosť inštrukcie spočíva v tom, že inštrukcia v konzekvente sa vykonáva až na základe výsledku testu aplikovaného na množinu propozícií vstupného stavu výskytu danej inštrukcie. Niektoré hypotetické inštrukcie spôsobujú vetvenie reťazca inštrukcií.

Veríme, že uvedené príklady a poznámky poskytujú aspoň rudimentárnu predstavu o tom, čo je vstupný stav, čo je výstupný stav a v čom spočíva prechod medzi nimi. Úvahy týkajúce sa inštrukcií, ich sémantickej povahy, vzájomných vzťahov a reťazenia môžeme uzavrieť a prejsť k modelu metódy pomocou zavedeného aparátu.

15. Na ceste k metóde. Neformálne poznámky. Metódami budú niektoré postupnosti postulátových a derivátových prechodov. Prv, než prejdeme k formálnejšiemu vymedzeniu metódy, uveďme niektoré neformálne poznámky, ktoré pomôžu bližšie špecifikovať, aké druhy postulátových a derivátových prechodov možno považovať za metódy.

V prvom pokračovaní state sme povedali, že (vedecké) metódy môžeme (aspoň čiastočne) charakterizovať podľa druhu cieľa. Metóda definovania je charakteristická tým, že jej produktom je definícia; metóda merania je charakteristická tým, že jej produktom je priradenie číselnej hodnoty nejakému objektu; metóda falzifikovania hypotézy je charakteristická tým, že jej produktom je falzifikácia danej hypotézy; metóda priameho dokazovania je zase charakteristická tým, že jej produktom je priamy dôkaz teorémy. Podobným spôsobom môžeme pokračovať ďalej. Metódami teda budú určité postupnosti postulátových a derivátových prechodov, pre ktoré napríklad platí: V prípade metódy definovania pôjde o postup, v ktorom sa nedefinovanému pojmu (či entite iného druhu) napokon priradí definujúci pojem (či entita iného druhu); v prípade metódy merania pôjde o postup, v ktorom sa objektu bez priradenej číselnej hodnoty určitého druhu napokon priradí číselná hodnota vyjadrujúca mieru vykazovania vlastnosti (veličiny); v prípade metódy falzifikovania ide o postup, v ktorom sa hypotéza testuje empirickou evidenciou, a ak jej evidencia (resp. výrok, ktorý ju vyjadruje) protirečí, hypotéza nadobudne štatút falzifikovanej hypotézy; v prípade metódy priameho dokazovania ide zase o postup, v ktorom sa formule, ktorá zatiaľ nie je dokázaná v danom systéme, napokon priradí postupnosť formúl predstavujúca jej priamy dôkaz. Predpokladáme teda, že každá metóda súvisí s aspoň jedným druhom cieľa, ktorý je jej pomocou v princípe *dosiahnuteľný*.[11]

Dôležitou súčasťou vymedzenia metódy nie je len druh cieľa, ktorý sa má dosiahnuť, ale aj inštrukcie, ktoré sa pri tom používajú. Existujú rôzne metodologické či pragmatické dôvody, ktoré vylučujú ako metódy také postupnosti derivátových a postulátových prechodov, v ktorých sa objavujú výskyty inštrukcií určitých druhov. Napríklad ak máme sústavu inštrukcií, ktoré nemožno z principiálnych (logických či nomologických) dôvodov nikdy vykonať, a dosiahnuť tak druh cieľa, ktorého sa týkajú, alebo ktoré nemajú žiadny druh cieľa, nebudeme takúto sústavu inštrukcií považovať za metódu. Zoberme si napríklad inštrukcie „Nakresli okrúhly štvorec!“ a „Skonštruuj perpetuum mobile!“ vyskytujúce sa v nejakých sústavách inštrukcií. Hoci je v prípade prvej inštrukcie otázne, či má vôbec druh cieľa (teda či druhom cieľa môže byť aj niečo, čo je logicky protirečivé), isté je to, že ju nemožno nikdy vykonať, keďže sa požaduje nakreslenie logicky nemožného objektu. Druhá inštrukcia je nevykonateľná zase preto, že fyzikálne zákony nášho sveta nomologicky neumožňujú existenciu objektu, ktorý by bol perpetuom mobile. Sústavy pozostávajúce z týchto inštrukcií teda určite nebudeme považovať za metódy. Metódami budú pre nás len tie sústavy inštrukcií, ktorých vykonanie je principiálne možné a ktoré vedie k nejakému druhu cieľa. Analogicky metódami nebudú také sústavy inštrukcií, ktoré obsahujú inštrukcie adresované jedinému adresátovi a nedajú sa vykonávať opakovane. Rozkazovacia veta „Alfonz Delikvent, dostavte sa dňa 11.11.2014 o 9,30 hod. na súdne pojednávanie Okresného súdu …!“ síce vyjadruje imperatív, a teda aj inštrukciu, no môže ju vykonať len jedna osoba, a navyše len raz.

Tieto dve obmedzujúce podmienky nám pomáhajú určiť, aké inštrukcie nebudú tvoriť metódu. Nehovoria nám však nič o tom, ktoré inštrukcie sú v nejakom zmysle potrebné na dosiahnutie určitého druhu cieľa. Vezmime si napríklad *metódu výberu vzorky z populácie* objektov určitého druhu a uvažujme nad nasledujúcim scenárom. Ak by súčasťou nejakej postupnosti výskytov inštrukcií nebol napríklad výskyt inštrukcie, ktorá od používateľa metódy požaduje identifikáciu populácie, z ktorej sa vzorka má vyberať, tak by takáto postupnosť fakticky nebola metódou výberu vzorky z populácie. Chýbal by totiž ten krok metódy – výskyt inštrukcie –, ktorý je nevyhnutný na realizáciu ostatných krokov (výskytov inštrukcií) metódy vedúcich k dosiahnutiu daného druhu cieľa. V 18. podkapitole zavedieme niektoré pojmy, ktoré sú v tejto súvislosti relevantné a v konečnom dôsledku sa dajú použiť pri rozhodovaní o tom, ktoré inštrukcie sú, resp. nie sú nevyhnutné na dosiahnutie príslušného druhu cieľa.

[11] Jeden zo súčasných metodologických prístupov prepája (vedecké) metódy (a metodologické pravidlá) s určitými epistemickými cieľmi vo forme metodologických princípov. Ich všeobecnú schému možno vyjadriť nasledovne: „Ak chceš dosiahnuť epistemickú hodnotu e_i, použi metodologické pravidlo (metódu) r_j!“ (pozri napríklad Laudan 1990, 1996; Nola, Sankey 2000). Principiálna dosiahnuteľnosť (realizovateľnosť) cieľa konania sa tu považuje za predpoklad racionality konania (pozri napríklad Laudan 1984).

16. Na ceste k metóde. Formálne poznámky. Neformálne obmedzenia špecifikujúce, ktoré sústavy inštrukcií možno charakterizovať ako metódy, doplníme formálnejším vymedzením.

Nech $S = \{S_0, \ldots, S_n\}$ je množina stavov, pričom ľubovoľný stav S_i ($0 \leq i \leq n$) možno reprezentovať ako trojicu ($On(S_i)$, $Op(S_i)$, $Pr(S_i)$), kde $On(S_i)$ je množina objektov (ontológia) stavu S_i, $Op(S_i)$ je množina operácií stavu S_i, ktoré možno aplikovať na prvky $On(S_i)$, a $Pr(S_i)$ je množina propozícií stavu S_i, ktoré opisujú prvky z $On(S_i)$. Na množine S možno definovať binárnu reláciu $R \subseteq S \times S$, t. j. množinu usporiadaných dvojíc prvkov $\{S_0, \ldots, S_n\}$, a to podľa nasledujúcich pravidiel (pričom $0 \leq i, j \leq n$ a $1 \leq k < n$):[12]

1. Vzájomné poradie ľubovoľných prvkov S_i a S_j je jednoznačne určené reláciou $\subseteq$ medzi množinami *a*) $On(S_i)$ a $On(S_j)$, *b*) $Op(S_i)$ a $Op(S_j)$, *c*) $Pr(S_i)$ a $Pr(S_j)$, pričom hovoríme, že stav S_i sa vyskytuje pred stavom S_j (zapíšeme: $S_i \prec S_j$), ak (*i*) $On(S_i) \subseteq On(S_j)$, (*ii*) $Op(S_i) \subseteq Op(S_j)$ a (*iii*) $Pr(S_i) \subseteq Pr(S_j)$.[13]
2. Dva stavy S_i a S_j, pre ktoré platí, že $S_i \prec S_j$, tvoria usporiadanú dvojicu (S_i, S_j) len vtedy, keď neexistuje taký stav S_k, pre ktorý platí, že $S_i \prec S_k$ a $S_k \prec S_j$.
3. Ak pre prvky S_i a S_j z dvojice (S_i, S_j) ďalej platí, že $Pr(S_i) = Pr(S_j)$, tak (S_i, S_j) je postulátový prechod.
4. Ak pre prvky S_i a S_j z dvojice (S_i, S_j) ďalej platí, že $Pr(S_i) \subset Pr(S_j)$, tak (S_i, S_j) je derivátový prechod.
5. Pre ktorúkoľvek usporiadanú dvojicu platí, že buď spĺňa 2. a 3. bod, alebo spĺňa 2. a 4. bod.

Tieto pravidlá ukazujú, že stavy tvoria len také usporiadané dvojice, v ktorých sa prvý člen v postupnosti vymedzenej v 1. bode nachádza pred druhým členom, pričom – podľa 2. bodu – neexistuje žiadny stav, ktorý by sa v danej postupnosti nachádzal medzi nimi. Ak pre stavy v nejakej usporiadanej dvojici platí, že druhý z nich má bohatšiu množinu propozícií ako prvý (t. j. spĺňa 4. bod), tak druhý stav je derivátovým rozšírením prvého stavu v tom zmysle, v akom sme tento pojem zaviedli v predchádzajúcom pokračovaní (11. podkapitola).[14] Ak pre stavy v nejakej usporiadanej dvojici platí, že druhý z nich nemá bohatšiu množinu propozícií ako prvý z nich (t. j. spĺňa 3. bod), tak druhý stav je postulátovým rozšírením prvého stavu; ak totiž predpokladáme, že ide o rôzne stavy, tak sa musia líšiť len množinami objektov (ontológiami) a/alebo množinami operácií, pričom tento rozdiel môže byť daný len tým, že bohatšia množina obsahuje v po-

[12] Kvôli jednoduchosti v tejto podkapitole predpokladáme, že medzi inštrukciami tvoriacimi metódu sa nevyskytuje inštrukcia formy $(\pi \rightarrow i_a)$ & $(\sim\pi \rightarrow i_b)$.

[13] Stavy S_i a S_j sú rôzne, ak platí aspoň jedna z týchto možností: (*i*) $On(S_i) \subset On(S_j)$, (*ii*) $Op(S_i) \subset Op(S_j)$, (*iii*) $Pr(S_i) \subset Pr(S_j)$. Ak platí, že $On(S_i) = On(S_j)$ a $Op(S_i) = Op(S_j)$ a $Pr(S_i) = Pr(S_j)$, tak $S_i = S_j$.

[14] Hoci sme v predchádzajúcej časti pripustili, že druhý stav môže byť derivátovým rozšírením prvého stavu aj v prípade, že ontológia prvého stavu je (len) vlastnou podmnožinou ontológie druhého stavu, nemusíme túto črtu využívať, keďže podmienka formulovaná v 2. bode je všeobecnejšia: Akékoľvek derivátové rozšírenie sa odrazí v tom, že množina propozícií druhého stavu je bohatšia ako množina propozícií prvého stavu, no nie každé derivátové rozšírenie musí viesť k tomu, že druhý stav bude mať bohatšiu ontológiu ako prvý stav.

rovnaní s chudobnejšou množinou iné postuláty. Keby sa mali líšiť aj novými derivátmi, museli by sa obdobným spôsobom líšiť aj množiny propozícií, čo sme vylúčili. V množine usporiadaných dvojíc utvorených podľa 3. bodu sa môžu vyskytovať prvky, pre ktoré je postulátové rozšírenie nulové. To znamená, že druhý stav sa od prvého stavu nelíši žiadnymi postulátmi (a keďže podľa predpokladu sa nelíšia ani derivátmi, tak ide o ten istý stav).

Celá postupnosť stavov má svoj počiatok, ktorý predstavuje stav S_0. Ak táto postupnosť nenadväzuje na nejakú predchádzajúcu postupnosť (napríklad na nejakú predchádzajúcu metódu), tak môžeme predpokladať, že $On(S_0) = \emptyset$, $Op(S_0) = \emptyset$ a $Pr(S_0) = \emptyset$. Ontológia, množina operácií a množina propozícií daného stavu neobsahuje žiadne prvky, čo je prirodzené, keďže tento stav nie je výsledkom žiadneho postulátového ani derivátového prechodu. Pravda, S_0 môže obsahovať neprázdne množiny za predpokladu, že daná metóda nadväzuje na nejakú inú metódu. Koncovým prvkom celej postupnosti je zase stav S_n, pri ktorom predpokladáme, že $On(S_n) \neq \emptyset$, $Op(S_n) \neq \emptyset$ a $Pr(S_n) \neq \emptyset$. Pre ľubovoľný stav S_i (kde $0 \leq i < n$) platí, že (*i*) $On(S_i) \subseteq On(S_n)$, (*ii*) $Op(S_i) \subseteq Op(S_n)$ a (*iii*) $Pr(S_i) \subseteq Pr(S_n)$. Je zrejmé, že pre ľubovoľný stav S_i okrem počiatočného stavu a koncového stavu platí, že existuje usporiadaná dvojica tvaru (S_i, S_j) a usporiadaná dvojica tvaru (S_k, S_i) (kde S_j a S_k sú nejaké stavy). V prípade počiatočného stavu S_0 existuje len dvojica tvaru (S_0, S_j) a v prípade koncového stavu S_n existuje len dvojica tvaru (S_k, S_n).

Keď to zhrnieme, máme k dispozícii množinu stavov $S = \{S_0, \ldots, S_n\}$ a binárnu reláciu $R \subseteq S \times S$, ktorá je určená uvedenými pravidlami. Metódu môžeme modelovať ako dvojicu (S, R), pričom reláciu R môžeme považovať za zjednotenie dvoch vzájomne disjunktných množín R_{der} a R_{pos}, kde R_{der} je množina derivátových prechodov a R_{pos} je množina postulátových prechodov.

17. Metóda a graf. Vhodným prostriedkom na modelovanie metódy je orientovaný graf.[15] Na grafickú reprezentáciu grafov budeme používať diagramy, v ktorých budeme navyše rozlišovať prechody oboch druhov, čo budeme aj naďalej znázorňovať šípkami rôznych druhov. Diagram môže vyzerať napríklad takto:[16]

$$
\begin{tikzcd}
 & & & & S_4 \arrow[r, dashed] & S_6 \arrow[r, "{[i_d]_4}"] & S_8 \arrow[r, dashed] & S_{11} \arrow[r, "{[i_g]_7}"] & S_{14} \\
S_0 \arrow[r, dashed] & S_1 \arrow[r, "{[i_a]_1}"] & S_2 \arrow[r, dashed] & S_3 \arrow[ur, "{[i_b]_2}"] \arrow[ddr, "{[i_c]_3}"'] & & & & & \\
 & & & & & & S_9 \arrow[r, dashed] & S_{12} \arrow[r, "{[i_h]_8}"] & S_{15} \\
 & & & & S_5 \arrow[r, dashed] & S_7 \arrow[ur, "{[i_e]_5}"] \arrow[dr, "{[i_f]_6}"'] & & & \\
 & & & & & & S_{10} \arrow[r, dashed] & S_{13} \arrow[r, "{[i_i]_9}"] & S_{16}
\end{tikzcd}
$$

[15] Pomocou grafov modeloval svoje chápanie metódy aj V. Filkorn v prácach (Filkorn 1972; 1973; 1998).

[16] Uvedený diagram je grafickou reprezentáciou len ilustratívneho grafu, ktorý má zachytiť niektoré relevantné črty metódy, no nemáme v úmysle stotožniť ho s nejakou konkrétnou metódou. Príklady konkrétnych metód budeme rozoberať v nasledujúcom pokračovaní state.

Stavy sú *vrcholmi* grafu a derivátové a postulátové prechody, t. j. usporiadané dvojice stavov, sú *hranami* grafu. Ak G je graf, tak $V(G)$ je množina vrcholov G a $H(G)$ je množina hrán G. Graf G reprezentovaný uvedeným diagramom je daný množinou vrcholov $V(G)$ a množinou hrán $H(G)$:

$V(G) = \{S_0, S_1, S_2, S_3, S_4, S_5, S_6, S_7, S_8, S_9, S_{10}, S_{11}, S_{12}, S_{13}, S_{14}, S_{15}, S_{16}\}$
$H(G) = \{(S_0, S_1), (S_1, S_2), (S_2, S_3), (S_3, S_4), (S_3, S_5), (S_4, S_6), (S_5, S_7), (S_6, S_8), (S_7, S_9), (S_7, S_{10}), (S_8, S_{11}), (S_9, S_{12}), (S_{10}, S_{13}), (S_{11}, S_{14}), (S_{12}, S_{15}), (S_{13}, S_{16})\}$

Definujme niektoré potrebné pojmy (R_{pos} je množina postulátových prechodov a R_{der} je množina derivátových prechodov, *vtt* je skratka za „vtedy a len vtedy, keď“):

1. *Stupeň vrcholu* $S_i \in V(G)$ – píšeme $d(S_i)$ – je počet hrán, v ktorých sa vrchol S_i v grafe G nachádza.
2. Vrchol $S_i \in V(G)$ je *počiatočný vrchol vtt* (*i*) $d(S_i) = 1$; (*ii*) existuje taký vrchol $S_j \in V(G)$, že *a*) $(S_i, S_j) \in H(G)$; *b*) $(S_i, S_j) \in R_{pos}$; a (*iii*) pre ľubovoľné vrcholy $S_k, S_l \in V(G)$, pre ktoré platí, že $d(S_k) = 1$, $(S_k, S_l) \in H(G)$ a $(S_k, S_l) \in R_{pos}$, platí, že $(S_i, S_j) = (S_k, S_l)$.
3. Vrchol $S_i \in V(G)$ je *koncový vrchol vtt* (*i*) $d(S_i) = 1$ a (*ii*) existuje taký vrchol $S_j \in V(G)$, že *a*) $(S_j, S_i) \in H(G)$ a *b*) $(S_j, S_i) \in R_{der}$.
4. Postupnosť vrcholov $(S_0, \ldots, S_k, S_l, \ldots, S_n)$ je *trasa* z vrcholu $S_i \in V(G)$ do vrcholu $S_j \in V(G)$ *vtt* (*i*) $S_0, \ldots, S_k, S_l, \ldots, S_n \in V(G)$; (*ii*) $S_i = S_0$; (*iii*) S_0 je počiatočný vrchol G; (*iv*) $S_j = S_n$; (*v*) S_n je koncový vrchol G; a (*vi*) pre ľubovoľné vrcholy $S_k, S_l \in V(G)$ (kde $0 \leq k < n$, $0 < l \leq n$ a neexistuje také číslo m, že $k < m < l$) platí, že $(S_k, S_l) \in H(G)$.[17]
5. Vrchol $S_i \in V(G)$ je *bod vetvenia vtt* (*i*) $d(S_i) \geq 3$; (*ii*) pre ľubovoľný vrchol $S_j \in V(G)$ platí, že $(S_i, S_j) \in H(G)$ *vtt* $(S_i, S_j) \in R_{der}$; (*iii*) pre ľubovoľný vrchol $S_k \in V(G)$ platí, že $(S_k, S_i) \in H(G)$ *vtt* $(S_k, S_i) \in R_{pos}$; a (*iv*) pre ľubovoľné vrcholy $S_l, S_m \in V(G)$ platí, že $(S_l, S_i) \in H(G)$ a $(S_m, S_i) \in H(G)$ *vtt* $(S_l, S_i) = (S_m, S_i)$.
6. Vrchol $S_i \in V(G)$ je *predkom* ľubovoľného vrcholu $S_j \in V(G)$ *vtt* (*i*) S_i a S_j ležia na tej istej trase a (*ii*) platí, že buď *a*) $(S_i, S_j) \in H(G)$, alebo *b*) existuje taká postupnosť vrcholov $(S_k, \ldots, S_m)$, že pre ľubovoľné dva stavy S_l a S_{l+1} (kde $k \leq l < m$) platí, že $(S_i, S_k) \in H(G)$, $(S_l, S_{l+1}) \in H(G)$ a $(S_m, S_j) \in H(G)$.
7. Vrchol $S_i \in V(G)$ je *potomkom* ľubovoľného vrcholu $S_j \in V(G)$ *vtt* S_j je predkom S_i.
8. Vrcholy $S_i, S_j \in V(G)$ sú *nezávislé vtt* S_i nie je ani predkom, ani potomkom S_j.

Na základe týchto definícií môžeme o uvedenom grafe povedať:

1. Graf obsahuje jeden počiatočný vrchol (t. j. S_0) a tri koncové vrcholy (t. j. S_{14}, S_{15} a S_{16}). Z definície počiatočného vrcholu vyplýva, že v jednom grafe sa nachádza práve

[17] Trasou grafu je teda ľubovoľná lineárna postupnosť vrcholov grafu, ktorá zahŕňa počiatočný aj koncový vrchol. V teórii grafov sa definujú pojmy *cesty* a *sledu*, ktoré sa v niektorých aspektoch podobajú nášmu pojmu *trasy*, no nie sú s ním totožné. Pojmy *cesty* a *sledu* v našom grafovom modeli metódy nebudeme využívať.

jeden počiatočný vrchol. Koncových vrcholov však môže byť viac, nielen jeden, čo sa ilustruje aj v našom príklade. Stupeň ktoréhokoľvek počiatočného či koncového vrcholu však musí byť 1.

2. Graf obsahuje tri trasy: (S_0, S_1, S_2, S_3, S_4, S_6, S_8, S_{11}, S_{14}), (S_0, S_1, S_2, S_3, S_5, S_7, S_9, S_{12}, S_{15}), (S_0, S_1, S_2, S_3, S_5, S_7, S_{10}, S_{13}, S_{16}). Ak aktér bude konať podľa danej metódy, nemôže postupovať po všetkých troch trasách, ale musí si vybrať jednu z nich. Body vetvenia totiž signalizujú prítomnosť inštrukcie formy $(\pi \rightarrow i_a)$ & $(\sim\pi \rightarrow i_b)$, pre ktorú je charakteristické to, že určitá inštrukcia sa môže vykonať len v prípade, že vstupný stav jej výskytu obsahuje určitú propozíciu; ak ju neobsahuje, má sa vykonať alternatívna inštrukcia, no vykonať obidve inštrukcie nie je možné.

3. V grafe sú dva vrcholy, ktoré sú bodmi vetvenia – S_3 a S_7. (Takéto vrcholy sú križovatkami najmenej dvoch trás.) Napríklad vrchol S_3 má stupeň 3, keďže sa vyskytuje v troch hranách (S_2, S_3), (S_3, S_4) a (S_3, S_5), pričom dve z nich – konkrétne (S_3, S_4) a (S_3, S_5) – ležia na rôznych trasách. Je zjavné, že počiatočné ani koncové vrcholy nemôžu byť bodmi vetvenia.

4. Vrchol môže ležať na viacerých trasách aj v prípade, že nie je bodom vetvenia. Príkladom je vrchol S_1, ktorý nie je bodom vetvenia, no nachádza sa na všetkých troch trasách, alebo vrchol S_5, ktorý leží na dvoch trasách. Samozrejme, niektoré vrcholy ležia len na jednej trase; príkladmi sú S_8, S_9 a S_{10}.

5. Vrchol je potomkom všetkých vrcholov, ktoré sa pred ním vyskytujú na tých istých trasách ako daný vrchol. Napríklad vrchol S_{11} je potomkom vrcholov S_0 – S_4, S_6 a S_8. Vrchol S_5 je síce potomkom napríklad vrcholov S_0 a S_1, no nie vrcholu S_4, keďže leží na trase, na ktorej sa nenachádza S_4. Analogicky to platí o predkoch. Napríklad vrchol S_1 je predkom všetkých vrcholov S_2 – S_{16}. Na druhej strane napríklad vrchol S_5 nie je predkom vrcholu S_6, keďže ležia na rôznych trasách. Vrcholy S_5 a S_6 sú nezávislé. Je zrejmé, že nezávislé vrcholy sa môžu vyskytovať len v takých grafoch, ktoré obsahujú viac ako jednu trasu, a teda obsahujú body vetvenia; graf pozostávajúci len z jednej trasy neobsahuje vetvenie, takže neobsahuje ani nezávislé vrcholy (a zodpovedajúca metóda neobsahuje inštrukciu formy $(\pi \rightarrow i_a)$ & $(\sim\pi \rightarrow i_b)$).

Zhrňme: Náš model metódy dobre korešponduje s tým, ako sa špecifikujú niektoré orientované grafy,[18] a teda niektoré orientované grafy môžu predstavovať metódy. Metódu modelujeme ako dvojicu (S, R), kde $S = \{S_0, \ldots, S_n\}$ je množina stavov a $R \subseteq S \times S$ je binárna relácia, pričom ďalej platí ($0 \leq i, j, m \leq n$):

1. $S_i = (On(S_i), Op(S_i), Pr(S_i))$;
2. $R = R_{der} \cup R_{pos}$, pričom $R_{der} \cap R_{pos} = \emptyset$;
3. $(S_i, S_j) \in R_{pos}$ *vtt* $Pr(S_i) = Pr(S_j)$;
4. $(S_i, S_j) \in R_{der}$ *vtt* $Pr(S_i) \subset Pr(S_j)$;
5. ak (*i*) $On(S_0) \subseteq On(S_i)$, (*ii*) $Op(S_0) \subseteq Op(S_i)$ a (*iii*) $Pr(S_0) \subseteq Pr(S_i)$, tak S_0 je počiatočný vrchol grafu;

[18] Ide o také orientované grafy, v ktorých sa medzi ľubovoľnými dvomi stavmi nachádza najviac jedna hrana.

6. ak S_i, S_j sú počiatočné vrcholy grafu, tak $S_i = S_j$;
7. ak neexistuje taký stav S_m, pre ktorý platí, že (*i*) $On(S_n) \subseteq On(S_m)$, (*ii*) $Op(S_n) \subseteq Op(S_m)$ a (*iii*) $Pr(S_n) \subseteq Pr(S_m)$, tak S_n je koncový vrchol grafu.

Treba dodať, že akýkoľvek graf, ktorý by nespĺňal aspoň jednu z uvedených podmienok, nebude spĺňať nevyhnutné podmienky na to, aby mohol byť grafom metódy (hoci môže byť grafom niečoho iného).

18. Metóda. Intuície a model. V závere tejto časti sa budeme venovať téme, ktorá zatiaľ zostáva otvorená a ku ktorej pozorný čitateľ určite očakáva nejaké vyjadrenie. Ide o zachytenie súvislosti medzi naším pôvodným intuitívnym vymedzením metódy z prvej časti state a modelom metódy, ktorý sme práve navrhli.

V prvej časti sme uviedli, že metódu možno neformálne považovať za návod na riešenie nejakého problému. Pojem problému sme vymedzili pomocou pojmu otázky a pojmu odpovede: Odpoveď sme reprezentovali ako propozíciu a otázku sme modelovali ako funkciu priraďujúcu predmetným oblastiam odpovede. Problém chápeme ako otázku, na ktorú nemáme vzhľadom na danú predmetnú oblasť odpoveď. Presnejšie, keďže otázku modelujeme ako funkciu, ktorá predmetnej oblasti priraďuje propozície (t. j. odpovede), tak problém (týkajúci sa určitej predmetnej oblasti) možno v našom modeli reprezentovať ako funkciu, ktorá nepriraďuje žiadnu hodnotu (propozíciu, odpoveď) danej predmetnej oblasti. Takúto predmetnú oblasť, ktorú sme nazvali bázou problému, reprezentujeme ako usporiadanú trojicu (**U**, **K**, **R**), kde **U** je univerzum, t. j. množina objektov, **K** je konceptuálny systém a **R** je množina propozícií s priradeným epistemickým štatútom (presnejšie, **R** je binárna relácia, množina usporiadaných dvojíc, pre ktorú platí, že $\mathbf{R} \subseteq \mathbf{P} \times \mathbf{E}$, kde **P** je množina propozícií a **E** množina epistemických štatútov). Vyriešiť problém znamená nájsť takú predmetnú oblasť, ktorá už poskytuje odpoveď na danú otázku, t. j. to, čo pôvodne bolo problémom (otázkou bez odpovede), sa vzhľadom na novú predmetnú oblasť stáva bezproblémovou otázkou. Presnejšie, vyriešiť problém znamená nahradiť pôvodnú bázu problému inou vhodnou trojicou (**U***, **K***, **R***), pre ktorú platí, že $\mathbf{U} \neq \mathbf{U}^*$ alebo $\mathbf{K} \neq \mathbf{K}^*$ alebo $\mathbf{R} \neq \mathbf{R}^*$. Takáto usporiadaná trojica je bázou riešenia.

Metódu sme zase modelovali ako určitú sústavu postulátových a derivátových prechodov medzi stavmi. Každú metódu možno v podstate charakterizovať ako postup od vstupného stavu prvého prechodu k výstupnému stavu posledného prechodu. Teraz sa môžeme spýtať: Ako možno prepojiť uvedené neformálne vymedzenie metódy ako návodu na riešenie problémov s naším modelom metódy ako sústavy derivátových a postulátových prechodov medzi stavmi? Inými slovami: Ako možno návod na riešenie problémov reprezentovať modelom tvoreným derivátovými a postulátovými prechodmi medzi stavmi? Odpoveď na tieto otázky je pomerne jednoduchá a priamočiara.

Problém vyriešime, ak bázu problému upravíme tak, že sa z nej stane báza riešenia, teda umožní odpovedať na pôvodne problémovú otázku. Sústavu derivátových a postulátových prechodov môžeme chápať ako špecifikujúcu postup, ktorý vedie k vhodnej báze riešenia (tou sa má nahradiť pôvodná báza problému). Podľa povahy problému to znamená, že musíme nahradiť aspoň jednu zo zložiek **U**, **K**, **R** inou zložkou. Ak sa problém týka objektov v univerze, vyriešime ho tak, že množinu **U** nahradíme inou množinou

U*; ak sa týka pojmov, vyriešime ho nahradením systému **K** iným systémom **K***; a ak sa týka propozícií, resp. ich epistemických štatútov, riešenie problému bude spočívať v nahradení relácie **R** reláciou **R***. Vymedzenie metódy ako návodu na riešenie problémov prepojíme s modelom metódy ako určitej sústavy prechodov tak, že *ontológiu* (t. j. množinu objektov) vstupných aj výstupných stavov daných prechodov naviažeme na množiny entít z bázy problému.

Kvôli jednoduchosti sa tu budeme zaoberať len takými problémami, ktoré sa týkajú jednej zo zložiek bázy problému, t. j. buď množiny **U**, alebo systému **K**, alebo relácie **R**. Nevylučujeme existenciu aj takých problémov, ktoré sa týkajú dvoch či všetkých troch zložiek bázy problému. V takom prípade sa môže stať, že nahradenie jednej zložky (napríklad **U**) inou zodpovedajúcou zložkou (**U***) môže viesť k tomu, že aj ostatné zložky (napríklad **K**, resp. **R**) bude treba nahradiť inými zodpovedajúcimi zložkami (t. j. **K***, resp. **R***). Prípady tohto druhu by si vyžiadali len trochu komplikovanejšie formulácie na niektorých miestach, no na podstate nášho návrhu nič nemenia, takže budeme od nich abstrahovať.

Keďže bázu problému tvoria množiny **U**, **K** a **R**, tak ontológiou stavu môže byť buď množina objektov (množina entít toho istého typu ako entity z **U**), alebo množina pojmov a vzťahov medzi pojmami (množina entít toho istého typu ako entity z **K**), alebo množina propozícií s epistemickým štatútom (množina entít toho istého typu ako entity z **R**). Presnejšie, pokiaľ ide o prvý stav prvého prechodu danej metódy, navrhujeme, aby jeho ontológia bola podmnožinou príslušnej množiny **U**, **K**, resp. **R**.[19] Každý ďalší stav sa môže svojou ontológiou líšiť od prvého stavu a nové prvky už síce nemusia patriť aj do danej množiny **U**, **K**, resp. **R**, no musí ísť o entity toho istého typu. V prípade, že riešime problém týkajúci sa univerza bázy problému, tak ontológiu jednotlivých stavov v derivátových a postulátových prechodov budú tvoriť množiny objektov takého druhu, aký tvorí univerzum; ak riešime problém týkajúci sa konceptuálneho systému bázy problému, tak ontológiou stavov bude množina pojmov (a vzťahov medzi nimi); ak napokon riešime problém týkajúci sa množiny propozícií s epistemickým štatútom, tak v ontológii stavov budeme mať množinu propozícií s priradeným epistemickým štatútom.[20]

[19] Ako sme uviedli (pozri 16. podkapitolu), ontológiou prvého stavu môže byť prázdna množina, no nie je to nevyhnutné, ako to bude zrejmé z existencie komplexných metód a reťazcov metód, ktoré opíšeme v záverečnej časti našej state.

[20] Mimochodom, tento postup korešponduje s jednou našou klasifikáciou inštrukcií. Inštrukcie sme rozdelili podľa ich predmetu na objektové, konceptuálne a propozičné (pozri Bielik, Kosterec, Zouhar 2014b, 10. podkapitola). Objektové inštrukcie majú vo svojom vstupnom a výstupnom stave ako ontológiu množinu objektov, konceptuálne inštrukcie zase množiny pojmov (a vzťahov medzi pojmami) a propozičné inštrukcie množiny propozícií (s epistemickým štatútom). (Určitú komplikáciu predstavujú hybridné inštrukcie, no tieto komplikácie sa eliminujú, ak si uvedomíme, že ide o zložené inštrukcie, pričom jednoduché inštrukcie, ktoré sú ich konštitutívnymi zložkami, už hybridné byť nemôžu.) Do vstupného aj výstupného stavu inštrukcie takisto zavádzame operácie a propozície, a to v závislosti od toho, čo je ontológiou stavu. Operácie nám umožňujú manipulovať s prvkami ontológie a propozície charakterizujú ontológiu, t. j. špecifikujú vlastnosti prvkov ontológie, resp. vzťahy medzi prvkami ontológie. V každom prípade však platí, že ak riešime problém týkajúci sa univerza, (spravidla) používame

Ak teda vykonáme všetky kroky špecifikované v príslušnej metóde, ktorú sme si zvolili na vyriešenie určitého problému, získame výstupný stav obsahujúci (okrem iného) určitú ontológiu, t. j. množinu objektov určitého druhu. Ak sme riešili problém týkajúci sa univerza pôvodnej bázy problému, v báze riešenia treba nahradiť pôvodné univerzum takou množinou objektov, ktorá bude obsahovať ako svoju podmnožinu ontológiu posledného výstupného stavu. Ak sme zase riešili problém týkajúci sa konceptuálneho systému pôvodnej bázy, tak v báze riešenia treba nahradiť pôvodný konceptuálny systém takým konceptuálnym systémom, ktorý bude obsahovať ako svoju súčasť ontológiu – ktorou je množina pojmov a vzťahov medzi nimi – posledného výstupného stavu (keďže v množine objektov jednotlivých stavov danej metódy sa vyskytujú pojmy a vzťahy medzi nimi). Ak sme riešili problém týkajúci sa množiny propozícií s epistemickým štatútom z pôvodnej bázy, tak v báze riešenia treba nahradiť pôvodnú množinu propozícií s epistemickým štatútom takou množinou, ktorá bude ako svoju podmnožinu obsahovať ontológiu – t. j. množinu propozícií s epistemickým štatútom – posledného výstupného stavu. Ak takýmto spôsobom nahradíme pôvodnú bázu problému novou bázou riešenia, otázka, ktorá bola problémom vzhľadom na pôvodnú bázu, sa stane vzhľadom na novú bázu neproblémovou otázkou. To znamená, že sme problém vyriešili. Ak sa tak nestalo, problém sme nevyriešili a musíme hľadať iné riešenie.

Treba však upozorniť na jednu komplikáciu, skrytú v predchádzajúcom odseku. Ak vykonáme všetky kroky špecifikované v určitej metóde, spravidla nenájdeme len jednu konkrétnu bázu riešenia, ale celú množinu vhodných báz riešenia. Postup, ktorý sme špecifikovali v predchádzajúcom odseku, totiž vyžaduje len to, aby sa v báze riešenia vyskytovala také univerzum, konceptuálny systém, resp. množina propozícií s epistemickým štatútom, pre ktoré platí, že obsahujú ako svoju súčasť ontológiu posledného výstupného stavu použitej metódy. Lenže túto podmienku môže v princípe spĺňať množstvo množín objektov, množstvo konceptuálnych systémov, resp. množstvo množín propozícií s epistemickým štatútom. Keď teda vyriešime problém, dostaneme celú množinu vhodných báz riešenia, z ktorých si môžeme vybrať tú, ktorá nám vzhľadom na rôzne epistemologické, metodologické, pragmatické a ďalšie dôležité faktory riadiace (vedecké, ale aj mimovedecké či nevedecké) skúmanie vyhovuje najlepšie.

Je teda zrejmé, že náš model metódy využívajúci postulátové a derivátové prechody medzi stavmi vyhovuje pôvodnému intuitívnemu vymedzeniu metódy ako návodu na riešenie problémov. Náš model totiž zachytáva spôsob, ako nájsť vhodnú bázu riešenia daného problému.

---

objektové inštrukcie, ak riešime problém týkajúci sa konceptuálneho systému, (spravidla) používame konceptuálne inštrukcie, a ak riešime problém týkajúci sa propozícií s epistemickým štatútom, (spravidla) používame propozičné inštrukcie.

Literatúra

BIELIK, L., KOSTEREC, M., ZOUHAR, M. (2014a): Model metódy (1): Metóda a problém. *Filozofia*, 69 (2), 105-118.

BIELIK, L., KOSTEREC, M., ZOUHAR, M. (2014b): Model metódy (2): Inštrukcia a imperatív. *Filozofia*, 69 (3), 197-211.

DUŽÍ, M. (2010): Tenses and Truth-Conditions: A Plea for if-then-else. In: Peliš, M. (ed.): *The Logica Yearbook 2009*. London: College Publications, 63-80.

DUŽÍ, M., JESPERSEN, B., MATERNA, P. (2010): *Procedural Semantics for Hyperintensional Logic: Foundations and Applications of Transparent Intensional Logic*. Dordrecht – Heidelberg – London – New York: Springer.

FILKORN, V. (1972): Pojem metódy. *Filozofia*, 27 (3), 225-244.

FILKORN, V. (1973): Cyklický aspekt vedeckej metódy. *Filozofia*, 28 (1), 37-53.

FILKORN, V. (1998): *Povaha súčasnej vedy a jej metódy.* Bratislava: Veda.

MURATA, T. (1989): Petri Nets: Properties, Analysis and Applications. *Proceedings of the IEEE*, 77 (4), 541-580.

LAUDAN, L. (1984): *Science and Values.* University of California Press.

LAUDAN, L. (1990): Normative Naturalism. *Philosophy of Science*, 57 (1), 44-59.

LAUDAN, L. (1996): *Beyond Positivism and Relativism: Theory, Method and Evidence*. Boulder, CO: Westview Press.

NOLA, R., SANKEY, H. (2000): A Selective Survey of Theories of Scientific Method. In: Nola, R. – Sankey, H. (eds.): *After Popper, Kuhn and Feyerabend*. Kluwer Academic Publishers, 1-65.

Táto práca bola podporovaná Agentúrou na podporu výskumu a vývoja na základe Zmluvy č. APVV-0149-12.

Lukáš Bielik
Katedra logiky a metodológie vied FiF UK
Šafárikovo nám. 6
814 99 Bratislava 1
Slovenská republika
e-mail: bielikluc@yahoo.com

Miloš Kosterec
Katedra logiky a metodológie vied FiF UK
Šafárikovo nám. 6
814 99 Bratislava 1
Slovenská republika
e-mail: milos.kosterec@gmail.com

Marián Zouhar
Katedra logiky a metodológie vied FiF UK
Šafárikovo nám. 6
814 99 Bratislava 1
Slovenská republika
e-mail: marian.zouhar@gmail.com