

# Pojem *problému* z hlediska teorie konstrukcí

Pavel Materna

Akademie věd České republiky, Praha

**Abstract:** Transparent Intensional Logic (TIL) explicates objective abstract procedures as so-called *constructions*. Constructions that do not contain free variables and are in a well-defined sense ‘normalized’ are called *concepts* in TIL. An argument is given for the claim that *every concept defines a problem*. The paper treats just mathematical concepts, and so mathematical problems, and tries to show that this view makes it possible to take into account some links between conceptual systems and the ways how to replace a non-effective formulation of a problem by an effective one. To show this *in concreto* a well-known Kleene’s idea from his (1952) is exemplified and explained in terms of conceptual systems so that a threatening paradox is avoided.

**Keywords:** Transparent Intensional Logic (TIL), constructions, problem, concept, conceptual system.

## 1 Problém a pojem

Při čtení slavné Kolmogorovovy stati (1932) musí logického analytika přirozeného jazyka napadnout, že pojem *problém* si zaslouží explikaci z hlediska logické analýzy přirozeného jazyka, zejména když sám Kolmogorov problémy formuluje v přirozeném jazyce (obsahujícím ovšem matematické výrazy).

Zadání i řešení problému je nepochybně spjato s *procedurami*. Není divu, že mi jako nejvhodnější nástroj explikace připadá Tichého Transparentní Intenzionální Logika (TIL), která vyvinula *procedurální sémantiku*.<sup>1</sup>

---

<sup>1</sup> Viz zejména Tichý (1988) a (systematický výklad současné podoby v) Duží – Jespersen – Materna (2010).

Podobně jako Montague (1974) vyšel i Tichý z předpokladu, že logická analýza přirozeného jazyka musí být po formální stránce inspirována nikoli predikátovou logikou (jakéhokoli řádu), nýbrž (typovaným)  $\lambda$ -kalkulem: ten totiž realizuje Churchovu geniální intuici, že logické operace lze v podstatě redukovat na *vytvoření funkce abstrakcí a aplikaci funkce na argumenty*.

TIL je spojena s typovou hierarchií, nejprve 1. řádu (kdy tzv. *konstrukce* jsou pouze užity, aby vytvořily množinové objekty, konkrétně funkce), posléze vyšších řádů, kdy konstrukce jsou také *zmiňovány*, takže se stávají objektem *sui generis*. Konstrukce jsou definovány a pojmenovány (proměnné, Trivializace, Provedení, Dvojí provedení, Kompozice, Uzávěr), což zde nemůžeme reprodukovat, takže odkazujeme hlavně na Duží – Jespersen – Materna (2010). Typy 1. řádu jsou množiny parciálních funkcí nad bázi  $\{0, \iota, \tau, \omega\}$ , tj. po řadě *pravdivostní hodnoty, Univerzum, tj. individua, časové momenty, popř. reálná čísla, a možné světy*.

Rozvětvená hierarchie typů pak určuje typy vyšších řádů. Zhruba můžeme ideu této hierarchie vyjádřit takto: Konstrukce konstruuji objekty typu řádu  $n$ . Jsou to *konstrukce řádu  $n$* . Množina všech konstrukcí řádu  $n$  tvoří nový typ: *typ řádu  $n + 1$* , značený  $*_n$ . Tak například proměnná  $x$   $v$ -konstruuující čísla, tedy objekty typu řádu 1, je konstrukce řádu 1 a je typu  $*_1$ . Je to tedy objekt typu řádu 2. Když v TIL zmiňujeme konstrukce, nemusíme kvůli tomu potřebovat metajazyk.

Mějme konstrukci  $C$ , která neobsahuje volnou proměnnou<sup>2</sup>.  $C$  je *pojem*.

*Příklad:* S A. Churchem tvrdíme, že každý smysluplný (ne výhradně syntaktický) výraz vyjadřuje pojem (viz Church 1956), tj. zvolíme jakýkoli výraz (zde: češtiny), včetně věty. Například:

$3 > 2$  vyjadřuje pojem:  $[^0 > ^0 3 ^0 2]$ , kde [...] je instrukce k vytvoření Kompozice,<sup>0</sup> vytváří Trivializaci a typy jsou  $3, 2 / \tau, > / (\sigma\tau)$ . Tento pojem konstruuje pravdivostní hodnotu T.

*Čísla (reálná) větší než 0* vyjadřuje pojem  $\lambda x [^0 > x ^0 0]$ , konstruuje třídu kladných reálných čísel, apod.

Budeme se zabývat matematickými pojmy, tj. naše báze obsahuje jen  $0$  a  $\tau$ , kde  $\tau$  bude množina *přirozených čísel*. Následující tvrzení se týká i empirických pojmů, dále půjde už jen o pojmy matematické.

<sup>2</sup> Pro přesnost dodejme „a je jistým dobře definovaným způsobem normalizována“.

**Tvrzení 1:** Každý pojem definuje určitý problém. –

*Argument:* Pojem jakožto konstrukce je možno chápat jako návod na proceduru vedoucí k určitému výsledku. Když formulujeme problém, tak nás zajímá výsledek nějaké procedury.

*Poznámka 1:* Tichý při analýze tázacích vět (1978) rozlišuje sémantické jádro (*topic*) a pragmatický postoj. Tak jako můžeme mít k téže propozici postoj tvrzení, tázání nebo například chtění, tak i v našem případě můžeme mít k danému pojmu postoj spočívající v provádění dané konstrukce, ale i postoj navádění k proceduře. Ta procedura (pojem) je v obou případech stejná.

*Poznámka 2:* V případě otevřené konstrukce, tj. konstrukce obsahující aspoň jednu volnou proměnnou, je situace následující: Nechť konstrukce  $C$  obsahuje jednu volnou proměnnou, řekněme  $x$ . Pak pojem určený jednoznačně příslušným pojmem je konstrukce  $\lambda x C$ . Při  $n > 1$  je více příslušných pojmů, a tedy i problémů. Například k otevřené konstrukci

$$[{}^0 > x y]$$

existují čtyři příslušné pojmy a tedy problémy:

$$\lambda x y [{}^0 > x y]$$

$$\lambda y x [{}^0 > x y]$$

$$\lambda x \lambda y [{}^0 > x y]$$

$$\lambda y \lambda x [{}^0 > x y]$$

Podle některých vykladačů pojmu *problém* by explikace měla zahrnovat jen případy, v nichž je obsažena nějaká netriviální obtíž. Např. právě uvedené konstrukce by pro ně jistě nereprezentovaly problémy. Je však velice vágní i nejednoznačné definovat stupeň obtížnosti, od něhož bychom teprve mluvili o problémech. Krom toho by takové kritérium znamenalo zavádění do explikace relativismu, resp. subjektivismu: Shora uvedené příklady představují jistě určitý stupeň obtížnosti (a vypočítat  $3 + 5$ , umí to každý v každém věku?).

## 2 Algoritmické a nealgoritmické pojmy

Pojem konstrukce v TIL není totožný s pojmem algoritmu. (Srovnej Tichý 1986.) Jednak může jít pouze o algoritmický výpočet jako např.  $(3 + 5)$ , jednak konstrukce jsou definovány i v těch případech, kdy není zaručena efektivní metoda (například se uznává jako legitimní aplika-

ce funkce na argument, aniž je vyžadován finitní charakter argumentu nebo efektivnost funkce). Protože pojmy jsou konstrukce, můžeme tedy rozlišit *pojmy algoritmické*, definující efektivní (rekurzivní, algoritmickou) proceduru, a *pojmy nealgoritmické*, kde příslušná procedura není efektivní.

Nealgoritmické pojmy jsou však dvojího druhu:

- a) pojmy, které jsou nealgoritmické a nejsou ekvivalentní žádnému algoritmickému pojmu, jako např. pojem množiny teorémů predikátové logiky 2. řádu, a
- b) pojmy, které jsou ekvivalentní určitému algoritmickému pojmu (a samy nejsou).

Jako příklad pojmu sub b) můžeme uvést (zapsáno matematickým jazykem)<sup>3</sup>:

$$\forall \lambda abc n (n > 2 \supset \neg [a^n + b^n = c^n]) \quad (\text{Fermatův poslední teorém})$$

Můžeme tedy definovat:

### **Definice 1**

*Problém indukovaný pojmem C*: algoritmická konstrukce C' ekvivalentní C. –

**Tvrzení 2:** *Je-li pojem C nealgoritmický, pak problém logicky ekvivalentní je rovněž nealgoritmický.* –

*Důkaz:* Necht' konstrukce C' je logicky ekvivalentní konstrukci C. To znamená, že logická forma C je logicky ekvivalentní C'. Tyto logické formy se tedy liší pouze logickými konstantami. Je-li jedna z konstrukcí C, C' nealgoritmická, pak druhá bude rovněž nealgoritmická, protože logicky ekvivalentní transformací se z neefektivní kombinace logických konstant nestane kombinace efektivní.

*Příklad:* Je-li konstrukce  $\forall \lambda x C$  nealgoritmická, pak logicky ekvivalentní konstrukce

$$\neg \exists \lambda x \neg C$$

je rovněž nealgoritmická.

<sup>3</sup> Výrazy jazyka matematiky jsou blízké způsobu, jakým zapisujeme konstrukce. Např. srovnej výraz  $(3 + 5)$  a zápis konstrukce  $[^0 + ^03 ^05]$ .

Vezměme konkrétně problém *prvočíselných čtveřic* (*prime quadruplets*): chceme zjistit, zda je nekonečně mnoho za sebou jdoucích čtveřic prvočísel s intervalem 0, 2, 6, 8. Nechť tedy *Kard(inalita)* je počet (v našem případě tříd o čtyřech prvcích), *c* je proměnná tříd čísel, *n, p* nechť jsou proměnné přirozených čísel. Náš problém lze napsat jako konstrukci (pojem), jejíž zápis bude opět proveden (kvůli jednoduchosti) běžným matematickým výrazem:

$$\exists n (n = \text{Kard } \lambda c (\exists p [ [\text{Prvočíslo } p] \wedge [c = \lambda x [x = p] \vee [x = p + 2] \vee [x = p + 6] \vee [x = p + 8]] ] ] )$$

(Jde o dosud nevyřešený problém. V kladném případě je těchto čtveřic konečně mnoho.)

Takto zadaný problém je nealgoritmický. Po provedení jakékoli logicky ekvivalentní transformace se na této skutečnosti nic nezmění. (Například  $\neg \forall n \neg (\dots)$ .)

Tvrzení 2 však může vést ke sporu! Lze totiž snadno dokázat:

**Fakt:** Třída těch pojmů *C*, které nejsou algoritmické a ke kterým byl nalezen problém *C'* tímto pojmem indukovaný není prázdná.

(Paradigmatickým příkladem je Fermatův poslední teorém.) Znamená to, že nalezený pojem/problém je ekvivalentní, ale nikoli logicky ekvivalentní.

### Definice 2

Nechť  $= / (o*_n*_n)$  je relace mezi dvěma pojmy, které konstruují stejný objekt nebo jsou oba prázdné.

Problém *C* je ekvivalentní problému *D*, jestliže existuje konstrukce *E* tvaru  $[C = D]$  konstruující *T*. –

(Je-li konstrukce *E* algoritmická, pak je problémem indukovaným pojmem *C*. Viz **Definici 1**.)

## 3 Pojmové systémy

Kleene ve své klasické knize (1952, 317) upozorňuje na jeden vysoce důležitý fakt. Nechť *R* je efektivně vyčíslitelná relace. Definujme (schematicky) následující funkci:

$$\mu y R(x, y) \begin{cases} \text{Nejmenší číslo takové, že platí } R(x, y), \text{ jestliže } \exists y R(x, y) \\ 0 \text{ v opačném případě} \end{cases}$$

Tato definice neurčuje ovšem efektivní proceduru. Jak však ukazuje Kleene, existují takové instance relace  $R$ , pro něž efektivní procedura existuje, ale k níž se nedostaneme na základě shora uvedené definice.

Nyní si rozvedeme příslušný příklad, který Kleene zmiňuje. Necht'  $a, b, c$  jsou určitá celá čísla a  $R$  je definováno takto:

$$\lambda xy [ax + by = c]$$

Problém, který má Kleene na mysli, je<sup>4</sup>

$$\mathbf{P} \quad \lambda x [\mu \lambda y [ax + by = c]]$$

Tento problém je nealgoritmický: nemáme záruku, že  $\exists y [ax + by = c]$  pro všechna  $x$ .

Mějme nyní  $gcd$  jako Euklidův algoritmus pro nalezení největšího společného dělitele dvou čísel. Problém

$$\mathbf{P}' \quad \lambda x [\mu \lambda y [gcd(a, b) = ax + by]]$$

je ovšem algoritmický. Lze nyní dokázat, že funkce, kterou konstruuje  $\mathbf{P}'$ , má pro

$$c = gcd(a, b)$$

stejně hodnoty jako funkce, kterou konstruuje  $\mathbf{P}$ , a pro

$$c \neq gcd(a, b)$$

nemají obě funkce hodnotu (resp. mají hodnotu 0 podle Kleeneovy definice).  $\mathbf{P}$  a  $\mathbf{P}'$  jsou tedy různé ekvivalentní konstrukce. (V TIL máme  $\mathbf{P} = \mathbf{P}'$ , ale  ${}^0\mathbf{P} \neq {}^0\mathbf{P}'$ .)  $\mathbf{P}'$  je tedy problém indukovaný problémem (pojmem)  $\mathbf{P}$ .

Předpokládejme, že chceme najít pojem indukovaný pojmem  $\mathbf{P}$  a ,nemáme k dispozici' pojem, který nazýváme Euklidovým algoritmem. Zřejmě nám nepomůže žádná logicky ekvivalentní úprava (**Tvrzení 2**) a k cíli nedojdeme. V okamžiku, kdy mezi použitelné pojmy zařadíme Euklidův algoritmus, je náš úkol splněn. Tento příklad ilustruje užitečnost zavedení dynamického měřítka naší pojmové kompetence: budeme definovat *pojmový systém*.<sup>5</sup>

<sup>4</sup> Opět: příslušná konstrukce, která je analýzou uvedeného matematického zápisu.

<sup>5</sup> Viz Materna (1998, 2004) a ovšem Duží – Jespersen – Materna (2010).

**Definice 3**

Nechť  $X$  je objekt, který není konstrukce. Pak  ${}^0X$  je *jednoduchý pojem*. Jednoduché pojmy nerozkládáme na složky a předpokládáme, že jsou dány.

**Definice 4**

Mějme konečnou množinu *Prim*  $\{P_1, \dots, P_n\}$  jednoduchých pojmů a rozvětvenou hierarchii typů. Pojmy z *Prim* nazýváme *primitivní pojmy* a pojmy z nekonečné množiny *Der* pojmů odvozených z *Prim* na základě rozvětvené hierarchie typů pojmy *odvozenými (derived)*. Sjednocení  $Prim \cup Der$  nazveme *pojmový systém*. –

Pojmový systém je jednoznačně určen množinou *Prim* a určuje, o kterých objektech můžeme mluvit. Je to ‚pojmová výbava‘, jakou máme díky tomuto systému k dispozici. Zde uvažujeme pouze matematické pojmy.

Je zřejmé, že nalézání efektivních postupů (algoritmických pojmů, které jsou indukovány daným pojmem) je závislé na tom, jaký pojmový systém máme k dispozici. Náš příklad ilustruje tento fakt. Zároveň můžeme fakt, že pojmové systémy se mění a celkově rozšiřují, chápat i jako podklad oprávněného matematického optimismu, který vyjádřil Gödel ve své filozofické kritice Turingova díla (1990, 306).<sup>6</sup>

To zřejmě neznamená, že vznikne pojmový systém, který umožní najít problém indukovaný pojmem *množina teorémů predikátové logiky 2. řádu* apod.<sup>7</sup>

Filozofický ústav  
Akademie věd České republiky  
Jilská 1  
110 00 Praha 1  
Česká republika  
maternapavel@seznam.cz

<sup>6</sup> „What Turing disregards completely is the fact that mind, in its use, is not static, but constantly developing, ...and that more and more abstract terms enter the sphere of our understanding.“

<sup>7</sup> Tato práce vznikla s podporou Grantové agentury České republiky, projekt P401/10/0792.

**Literatura**

- CHURCH, A. (1956): *Introduction to Mathematical Logic*. Princeton: Princeton University Press.
- DUŽI, M. – JESPERSEN, B. – MATERNA, P. (2010): *Procedural Semantics for Hyperintensional Logic*. Berlin: Springer Verlag.
- GÖDEL, K. (1972): Some remarks on the Undecidability Results. In: Gödel, K. (2004): *Collected works, Vol. II*. New York – Oxford: Oxford University Press.
- KLEENE, S. C. (1952): *Introduction to Metamathematics*. New York, Toronto: D. van Nostrand Company.
- KOLMOGOROV, A. (1932): Zur Deutung der intuitionistischen Logik. *Mathematische Zeitschrift* 35, 58-65.
- MATERNA, P. (1998): *Concepts and Objects*. *Acta Philosophica Fennica* 63. Helsinki: Societas Philosophica Fennica.
- MATERNA, P. (2004): *Conceptual Systems*. Berlin: Logos Verlag.
- MONTAGUE, R. (1974): *Formal Philosophy: Selected Papers of Richard Montague*. Edited by R. Thomason. New Haven: Yale University Press.
- TICHÝ, P. (1988): *The Foundations of Frege's Logic*. Berlin – New York: De Gruyter.