Katedra informatiky
Přírodovědecká fakulta
Univerzita Palackého v Olomouci

BAKALÁŘSKÁ PRÁCE

Topologická teorie datových typů

2024

Martin Holub

Vedoucí práce:
doc. RNDr. Michal Krupka, Ph.D.

Studijní program: Informatika,
Specializace: Obecná informatika

Bibliografické údaje

Autor:	Martin Holub
Název práce:	Topologická teorie datových typů
Typ práce:	bakalářská práce
Pracoviště:	Katedra informatiky, Přírodovědecká fakulta, Univerzita Palackého v Olomouci
Rok obhajoby:	2024
Studijní program:	Informatika, Specializace: Obecná informatika
Vedoucí práce:	doc. RNDr. Michal Krupka, Ph.D.
Počet stran:	39
Přílohy:	elektronická data v úložišti katedry informatiky
Jazyk práce:	český

Bibliographic info

Author:	Martin Holub
Title:	Topological theory of data types
Thesis type:	bachelor thesis
Department:	Department of Computer Science, Faculty of Science, Palacký University Olomouc
Year of defense:	2024
Study program:	Computer Science, Specialization: General Computer Science
Supervisor:	doc. RNDr. Michal Krupka, Ph.D.
Page count:	39
Supplements:	electronic data in the storage of department of computer science
Thesis language:	Czech

Anotace

Topologická teorie datových typů dává do souvislosti problémy z teorie vyčíslitelnosti s problémy z topologie, a přináší tak na ně nový úhel pohledu. V této práci tuto teorii zformulujeme s pomocí výpočetního modelu částečně rekurzivních funkcí. Topologickou teorii datových typů stavíme na nově vytvořené definici datového typu, která zohledňuje způsob jeho reprezentace v počítači. Na základě této teorie následně zkoumáme vyčíslitelnost zobrazení a její podmínky. Rozebíráme možnost definovat základní aritmetické operace pro vybrané reprezentace reálných čísel a jejich případnou implementaci. Na závěr si ukazujeme konkrétní případy, ve kterých tyto datové typy narážejí na hranice vyčíslitelnosti, a tedy je naprogramovat nelze.

Synopsis

The topological theory of data types links problems from the computability theory to problems from topology, and thus brings a new perspective to them. In this thesis, we formulate this theory using a computational model of partially recursive functions. We build the topological theory of data types on a newly developed definition of a data type that takes into account the method of its representation in the computer. Based on this theory, we then investigate the computability of functions and its conditions. We analyze the possibility of defining basic arithmetic operations for selected representations of real numbers and their possible implementation. In the end, we show specific cases in which these data types hit the bounds of computability and therefore cannot be programmed.

Klíčová slova: topologie datových typů; reprezentace reálných čísel; vyčíslitelnost

Keywords: topology of data types; representations of real numbers; computability;

Děkuji doc. RNDr. Michalu Krupkovi, Ph.D. za cenné rady, věcné připomínky a vstřícnost při vedení této práce.

Odevzdáním tohoto textu jeho autor/ka místopřísežně prohlašuje, že celou práci včetně příloh vypracoval/a samostatně a za použití pouze zdrojů citovaných v textu práce a uvedených v seznamu literatury.

Obsah

Seznam obrázků

1 Úvod

První programovací jazyky vznikly v polovině minulého století. Za tu dobu se při práci s nimi podařilo překonat spoustu překážek. V dnešní době jsou některé z hlavních problémů spojené s hranicí vyčíslitelnosti. Jedním z hlavních cílů této práce je na tuto problematiku nahlédnout a zformulovat základy teorie, která se podílí na zkoumání tohoto fenoménu.

S vývojem programovacích jazyků se objevila potřeba pracovat na počítači s nejrůznějšími druhy dat, a dala tím tak za vznik datovým typům. Ačkoliv máme k dispozici pouze konečné množství paměti, můžeme se kromě běžných konečných datových typů, jako jsou `char` a `float`, také setkat s nekonečnými, mezi které patří třeba reálná čísla nebo funkční datové typy. Dosáhneme toho například tím, že hodnoty reprezentujeme jako funkce, které pro vstup n vrátí n-tou cifru nebo líným vyhodnocováním nekonečných seznamů.

Právě při práci s nekonečnými datovými typy narazíme na to, že některé jejich operace nelze naprogramovat. Klasickým příkladem takové operace je rovnost reálných čísel. Zároveň existují různé způsoby, jak hodnoty datového typu reprezentovat, proto se může stát, že jednotlivé způsoby se liší v tom, které operace nám umožní naprogramovat. Z hlediska vyčíslitelnosti je tedy datový typ od své reprezentace neodlučitelný.

Topologická teorie datových typů představuje jednu z mnoha aplikací topologie v informatice. Za pomoci Sierpińského prostoru, který je tvořen z prvku představujícího konečný výpočet a prvku představujícího zacyklení, propojíme pojem spojitost a vyčíslitelnost. Stanovíme na datových typech topologii pomocí množin rozpoznatelných výpočtem, a dosáhneme tím toho, že každé naprogramovatelné zobrazení je spojité. Topologie datového typu nám tedy bude určovat, které operace nejsme schopni naprogramovat a bude nám tak poskytovat pomyslnou hranici vyčíslitelnosti.

V existující literatuře definice datového typu nezohledňují způsob reprezentace a nejsou přímo napojeny na nějaký výpočetní model. V této práci pro upřesnění definujeme datový typ tak, aby na způsobu reprezentace závisel a za pomoci výpočetního modelu částečně rekurzivních funkcí na něm tuto teorii zformulujeme.

V poslední části textu se zaměříme na datový typ reálných čísel, konkrétně na reprezentaci nekonečným binárním rozvojem, reprezentaci vloženými intervaly a vyváženou trojkovou soustavou. U těchto reprezentací rozebereme, jestli lze naprogramovat základní aritmetické operace a případně jakým způsobem tyto operace implementovat.

V Kapitole 2 si tedy představíme základní definice, které nás budou po zbytek práce provázet. V následující Kapitole 3 zformulujeme základní prvky topologické teorie datových typů, jako je datový typ, pozorovatelná množina či vyčíslitelná funkce. V poslední, Kapitole 4, se na základě této teorie budeme věnovat už zmíněným datovým typům reálných čísel.

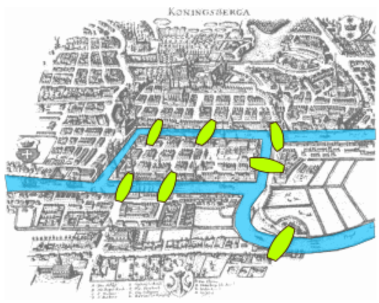
2 Základní definice

V této kapitole stanovíme základní definice, na kterých je topologická teorie datových typů postavena. Zároveň poslouží k představení použité terminologie a symbolů. Předpokládá se základní znalost matematické analýzy, teorie vyčíslitelnosti a diskrétní matematiky.

2.1 Topologie

Topologie je matematický obor, který se zabývá především vlastnostmi geometrických objektů, které se nemění při spojitých zobrazeních.

Za jednu z prvních zmínek o topologii se považuje Eulerův slavný problém Sedmi mostů města Královce z roku 1735 [1]. Cílem je zjistit, jestli lze přejít všechny mosty ve městě z Obrázku 1 tak, abychom každý z nich navštívili právě jednou.[1]



Obrázek 1: Mapa sedmi mostů města Královec.

Ačkoliv se jedná spíše o problém z teorie grafů, jeho podstata je úzce spjatá s topologií. Nejedná se o klasický geometrický problém, protože pro jeho vyřešení nepotřebujeme znát velikost jednotlivých mostů ani vzdálenosti mezi nimi.[2] Řešení závisí pouze na způsobu rozestavění mostů a jejich počtu. Spojitá geometrická zobrazení, jako jsou roztahování a ohýbání, zachovávají jakousi topologickou strukturu, a proto se po jejich aplikaci řešení nemění.

2.1.1 Obecná topologie

Definice z této kapitoly vycházejí především z poznatků [2] a [3].

Ačkoliv existují různá odvětví topologie, v této práci se budeme zabývat pouze obecnou topologií. Obecná topologie zkoumá vlastnosti topologických prostorů, jako jsou kompaktnost, souvislost, separabilita a mnoho dalších. Jak název napovídá, jedná se o obecnou teorii, nejsme tedy limitováni pouze na problémy

[1] Giuşcă Bogdan, 2005, The Problem of the Seven Bridges of Königsberg. Dostupné z https://commons.wikimedia.org/w/index.php?curid=112920.

[2] Vlastnostmi geometrických objektů souvisejícími se vzdáleností se zabývají metrické prostory.

spojené s nějakou geometrickou interpretací. Abychom mohli hovořit o topologickém prostoru, musíme nejdříve definovat topologickou strukturu.

Topologickou strukturou (topologií) na množině X nazveme systém τ podmnožin X splňující:

1. $\emptyset \in \tau, X \in \tau$,
2. jestliže $U, V \in \tau$, pak i $U \cap V \in \tau$,
3. jestliže $\sigma \subseteq \tau$, pak i $\bigcup \sigma \in \tau$.

Topologická struktura je tedy systém množin uzavřený na konečné průniky a libovolná sjednocení. Jednotlivé prvky τ nazýváme *otevřené množiny. Topologickým prostorem* tedy nazveme množinu X spolu s topologií τ, budeme ho zjednodušeně značit X.

Mějme na množině X topologie τ_1 a τ_2, jestliže $\tau_1 \subseteq \tau_2$, řekneme, že topologie τ_2 je *silnější* než τ_1 a naopak topologie τ_1 je *slabší* než τ_2.

V důkazech některých tvrzení o všech otevřených množinách, jako je například spojitost zobrazení, nám bude stačit ověřit, že tvrzení platí pouze pro bázi topologie. Bází topologie rozumíme systém množin, ze kterého jsme schopni topologii vygenerovat.

Báze topologie na množině X nazveme systém σ podmnožin X, pro který platí:

1. $\bigcup_{U \in \sigma} U = X$,
2. Jestliže $U_1, U_2 \in \sigma$ a $x \in U_1 \cap U_2$, pak existuje $V \in \sigma$ takové, že $x \in V \subseteq U_1 \cap U_2$.

Řekneme, že topologie τ je *generovaná bází* σ, jestliže pro každou otevřenou množinu $U \in \tau$ platí, že je sjednocením nějakých množin z báze σ.

Mezi příklady základních topologických prostorů na množině X patří *diskrétní topologie*, její bází je množina $\sigma = \{\{x\} \mid x \in X\}$ a *triviální topologie*, její bází je množina $\sigma = \{X\}$. Na kartézském součinu $X \times Y$ topologických prostorů je příkladem *součinová topologie*, její bází je množina $\sigma = \{U \times V \mid U \in \sigma_1, V \in \sigma_2\}$, kde σ_1 a σ_2 jsou popořadě báze topologie na X a Y.

Pokud není řečeno jinak, při práci s reálnými čísly se používá *standardní* (též *přirozená*) topologie. Bází standardní topologie je množina všech otevřených intervalů $(x, y) = \{a \in \mathbb{R} \mid x < a < y\}$.

Dále si definujeme důležité pojmy, které budeme používat v kontextu zobrazení mezi topologickými prostory.

O zobrazení f topologických prostorů X a Y řekneme, že je *spojité*, jestliže pro každou otevřenou množinu V v Y je množina $f^{-1}(V)$ otevřená v X.

Která zobrazení jsou spojitá, tedy závisí na tom, jaké uvažujeme na množinách topologie. Jak už jsme zmínili, spojitost zobrazení lze také určit jen na základě báze topologie.

Věta 1

Nechť $f : X \to Y$ je zobrazení topologických prostorů, σ je báze topologie na Y a pro každý prvek báze $B \in \sigma$ platí, že $f^{-1}(B)$ je otevřená v X. Pak zobrazení f je spojité.

Důkaz

Označme τ topologii na množině X. Ověříme, že pro každou otevřenou množinu $O \subseteq Y$ platí $f^{-1}(O) \in \tau$. Víme, že pro každou otevřenou množinu v Y, tedy i O existuje množina prvků báze σ, jejichž sjednocením tato množina vznikne. Označme množinu těchto bázových prvků $\theta \subseteq \sigma$. Protože θ obsahuje pouze prvky báze, tak z předpokladu plyne, že $\beta = \{f^{-1}(B) \mid B \in \theta\}$ je systém otevřených množin v X. Platí, že $\beta \subseteq \tau$ a tak z třetího bodu definice topologie plyne, že $\bigcup \beta \in \tau$. Takže množina O je skutečně otevřená, a zobrazení f je spojité. □

Další vlastnost, kterou budeme určovat u zobrazení topologických prostorů, je otevřenost. Zobrazení $f : X \to Y$ topologických prostorů nazveme *otevřené*, jestliže pro každou otevřenou množinu $U \in X$ platí, že její obraz $f(U)$ je otevřená množina v Y.

Otevřená zobrazení jsou tedy zjednodušeně právě ta, která zachovávají strukturu otevřených množin.

2.2 Parciální funkce

Mějme množiny X a Y, pak *parciální funkcí* $f : X \rightharpoonup Y$ nazveme funkci mezi množinami $X_1 \subseteq X$ a Y. Jestliže množina $X_1 = X$, tedy funkce f je definovaná pro každou hodnotu z X, pak řekneme, že parciální funkce f je *totální.*

Jestliže $x \in X$ a hodnota $f(x)$ je definována, nazveme hodnotu $f(x)$ *obrazem* prvku x. Množinu všech prvků $x \in X$, pro které existuje hodnota $f(x)$, nazveme *definičním oborem* parciální funkce f a budeme ji označovat jako $dom(f)$.

2.3 Vyčíslitelnost

V teorii vyčíslitelnosti se zabýváme algoritmickou řešitelností problémů. Podle známé Church–Turingovy teze lze ke každému algoritmu sestrojit ekvivalentní Turingův stroj. Turingovy stroje tedy chápeme jako jednu z možností, jak formálně popsat algoritmus. Nejedná se ale o jediný výpočetní model ztotožnitelný s pojmem algoritmus, mezi další patří například RAM stroje, Lambda kalkul nebo také částečně rekurzivní funkce.

V rámci této práce budeme pracovat s posledním zmíněným výpočetním modelem, a to s částečně rekurzivními funkcemi. V následující kapitole si tento model definujeme. V kontextu částečně rekurzivních funkcí budeme vždy uvažovat přirozená čísla včetně nuly a budeme tuto množinu označovat $\mathbb{N}_0$.

2.3.1 Částečně rekurzivní funkce

Poznatky z této kapitoly vycházejí z textů [4] a [5]. Množinu částečně rekurzivních funkcí bychom mohli charakterizovat, jako množinu těch funkcí na přirozených číslech, které jsme schopni definovat algoritmem. Výpočetní model částečně rekurzivních funkcí získáme rozšířením modelu primitivně rekurzivních funkcí.

Pro $n \geq 1$ funkci $f : \mathbb{N}_0^n \to \mathbb{N}_0$ nazveme *primitivně rekurzivní*, zkráceně *PRF*, jestliže platí některé z následujících:

1. f je nulová funkce: $f(x_1, \ldots, x_n) = 0$,
2. f je funkce projekce: pro nějaké $i \leq n$ platí, že $f(x_1, \ldots, x_n) = x_i$,
3. f je funkce následníka: pro nějaké $i \leq n$ platí, že $f(x_1, \ldots, x_n) = x_i + 1$,
4. f je složení primitivně rekurzivních funkcí: $g_1, \ldots, g_m$ jsou n-ární PRF a h je m-ární PRF a tedy $f(x_1, \ldots, x_n) = h(g_1(x_1, \ldots, x_n), \ldots, g_m(x_1, \ldots, x_n))$,
5. f je primitivní rekurzí: $\phi^{-1}(x_1, \ldots, x_{n-1})$ a $h(x_1, \ldots, x_{n+1})$ jsou PRF funkce a platí, že $f(x_1, \ldots, x_{n-1}, 0) = g(x_1, \ldots, x_{n-1})$ a zároveň pro $m \in \mathbb{N}_0$ platí, že $f(x_1, \ldots, x_{n-1}, m + 1) = h(x_1, \ldots, x_{n-1}, m, f(x_1, \ldots, x_{n-1}, m))$.

Jsou to ty funkce na přirozených číslech, které lze definovat algoritmem používajícím pouze konečné *for* cykly. Víme, že každá primitivně rekurzivní funkce pro každý svůj vstup vždy po konečném počtu kroků skončí a vrátí výsledek. Jde pomocí nich definovat většinu základních aritmetických operací na přirozených číslech, například ale také funkci signum, či absolutní hodnotu. Jelikož nelze používat záporná čísla, pak pro $n_1, n_2 \in \mathbb{N}_0$, takové že $n_1 < n_2$ odčítání definujeme jako $n_1 - n_2 = 0$.

Poznámka 2

Mějme funkce projekce $\pi_1(x) = x$ a $\pi_3(x_1, x_2, x_3) = x_3$, funkci následníka $S(x) = x + 1$ a primitivně rekurzivní funkci $h(x_1, x_2, x_3) = S(\pi_3(x_1, x_2, x_3))$. Funkci pro součet dvou čísel f definujeme pomocí primitivní rekurze následovně:

$$f(x, 0) = \pi_1(x) = x$$

$$f(x, n + 1) = h(x, n, f(x, n)) = S(\pi_3(x, n, f(x, n))) = f(x, n) + 1$$

Pro čitelnost budeme u základních operací, jako je sčítání, odčítání, násobení a jiné používat notaci, která popisuje, co funkce počítá místo toho, jak daná funkce výpočet provádí. Místo prefixové notace v některých situacích budeme používat infixovou notaci, například součet dvou čísel x, y označíme $x + y$.

Často se v důkazech využívá funkce, která nám vrátí předchůdce dané hodnoty, tuto funkci budeme značit P a definujeme ji pomocí primitivní rekurze následovně:

$$f(0) = 0$$

$$f(n + 1) = \pi_1(n, f(n)) = n$$

Stále ale lze algoritmicky sestrojit funkce, na které nám tento výpočetní model nestačí. Příkladem je třeba Ackermannova funkce [7]. Abychom ale měli výpočetní model ekvivalentní Turingovým strojům, musíme přidat operátor neomezené minimalizace.

Funkci $f(x_1, \ldots, x_n)$ nazveme *částečně rekurzivní*, zkráceně *ČRF*, jestliže je primitivně rekurzivní, nebo vznikne minimalizací $(n+1)$-ární částečně rekurzivní funkce g, což znamená, že platí $f(x_1, \ldots, x_n) = \mu y[g(x_1, \ldots, x_n, y) = 0]$, kde zápis $\mu y[g(x_1, \ldots, x_n, y) = 0]$ představuje nejmenší přirozené číslo y takové, že $\phi^{-1}(x_1, \ldots, x_n, y) = 0$.

Může se stát, že pro nějaký vstup $x_1, \ldots, x_n$ neexistuje přirozené číslo y splňující danou podmínku. Pomocí neomezené minimalizace můžeme tedy vytvořit funkce, které jsou parciální. Z hlediska vyčíslitelnosti to znamená, že pro ty částečně rekurzivní funkce, které jsou parciální, existují vstupy, jejichž výpočet se zacyklí, protože hledaná hodnota neexistuje. Použití neomezené minimalizace na funkci si ukážeme v následujícím příkladu.

Příklad 3

Mějme částečně rekurzivní funkci $g(x, y) = (x + y) - 4$, vytvoříme z ní pomocí neomezené minimalizace funkci $f(x) = \mu y[g(x, y) = 0]$. Hodnota funkce f existuje pouze pro některá přirozená čísla. Například $f(1) = 3$, protože $y = 3$ představuje nejmenší hodnotu, pro kterou platí rovnost $(1 + y) - 4 = 0$.

Pokud bychom chtěli znát například hodnotu $f(5)$, tak se její výpočet zacyklí, protože hodnota funkce f pro tento vstup neexistuje.

Jedním z hlavních poznatků teorie vyčíslitelnosti je to, že existují problémy, které nejsou algoritmicky řešitelné. U některých množin proto nelze konečným výpočtem určit, jestli zadaná hodnota do této množiny patří, nebo ne. My tuto vlastnost budeme určovat pomocí charakteristické funkce.

Charakteristickou funkcí množiny $A \subseteq X$ rozumíme parciální funkci $c : X \rightharpoonup \mathbb{N}_0$ takovou, že $c(x) = 0$ právě když $x \in A$.

Množinu $X \subseteq \mathbb{N}_0^n$ nazveme *částečně rekurzivní*, jestliže existuje částečně rekurzivní charakteristická funkce této množiny. Jestliže existuje totální částečně rekurzivní charakteristická funkce, pak říkáme, že X je *rekurzivní*.

Z definic plyne, že rekurzivní množina je rozhodnutelná, protože její charakteristická funkce se pro každý vstup zastaví a rozhodne, zda do této množiny patří. Částečně rekurzivní množina je už dle svého názvu pouze částečná rozhodnutelná, to znamená, že pro vstupy, které do množiny nepatří, se může výpočet charakteristické funkce zacyklit.

Částečně rekurzivní funkci $U : \mathbb{N}_0^2 \rightharpoonup \mathbb{N}_0$ nazveme *univerzální*, jestliže přijímá kód částečně rekurzivní funkce a vstup, přičemž výsledkem této funkce je hodnota, kterou by kódovaná funkce vrátila pro daný vstup. Existence této funkce plyne z ekvivalence částečně rekurzivních funkcí a výpočetního modelu Turingových strojů, ve kterém se s univerzálním strojem často setkáváme. Konstrukci této funkce najdeme mimo jiné v literatuře [6].

2.4 Číslování

2.4.1 Párovací funkce

Párovací funkcí rozumíme bijektivní zobrazení $f : \mathbb{N}_0 \times \mathbb{N}_0 \to \mathbb{N}_0$, které každé dvojici přirozených čísel vrátí jedinečné přirozené číslo. Jelikož se jedná o bijekci, z každého přirozeného čísla naopak získáme dvojici, kterou toto číslo reprezentuje. Jednotlivé složky tohoto čísla určíme pomocí projekcí p_1 a p_2. Jako příklad si uvedeme Cantorovu párovací funkci [8]. Tuto funkci sestrojíme následovně:

$$f(a, b) = \frac{(a + b)^2 + 3a + b}{2}$$

Označme $w(x) = \left\lfloor \frac{\sqrt{8x+1}-1}{2} \right\rfloor$, projekce p_1, p_2 párovací funkce f definujeme následovně:

$$b = p_2(x) = x - \frac{w(x)(w(x) + 1)}{2}$$
$$a = p_1(x) = w(x) - b$$

2.4.2 Gödelovo číslování

Tato kapitola vychází z [9].

Gödelovo číslování (též kódování) je zobrazení $\phi : \mathbb{N}^n \to \mathbb{N}$ pro $n \geq 1$, dané následujícím předpisem:

$$\phi(x_1, \ldots, x_n) = \prod_{i=1}^{n} p_i^{x_i},$$

kde p_i představuje i-té nejmenší prvočíslo.

Toto kódování nám umožňuje přiřadit unikátní čísla matematickým objektům, jako jsou například formule, důkazy nebo také funkce. Jelikož poskytuje způsob, jak zakódovat n-tici kladných přirozených čísel, můžeme pomocí něj jednoznačně zakódovat řetězce libovolné konečné abecedy. Dosáhneme toho tak, že všechny symboly dané abecedy seřadíme a popořadě očíslujeme. Danou n-tici přirozených čísel můžeme ze součinu prvočísel zpětně dekódovat tak, že činitele seřadíme podle velikosti vzestupně a jejich mocnina bude určovat kód symbolu na dané pozici.

Princip tohoto číslování staví na základní větě aritmetiky, tedy že každé přirozené číslo $n > 1$ lze jednoznačně rozložit na součin prvočísel. Jelikož pro tento rozklad existují různé algoritmy, získaná prvočísla se mohou lišit pořadím, což nevadí, protože při součinu na pořadí nezáleží.

Příklad 4

Mějme abecedu $\Sigma = \{a, +, =\}$, očíslujme prvky funkcí f tak, že $f(a) = 1, f(+) = 2, f(=) = 3$, pak pomocí Gödelova číslování je řetězec $a + a = aa$ reprezentován šesticí $(1, 2, 1, 3, 1, 1)$ a ta má hodnotu $4414410 = 2^1 3^2 5^1 7^3 11^1 13^1$.

Jelikož máme součin prvočísel a jednotlivá prvočísla jsou seřazená, n-tici přirozených čísel získáme z exponentů jednotlivých mocnin, tedy $(1, 2, 1, 3, 1, 1)$. Dosazením symbolů za hodnoty získáme zpět řetězec $a + a = aa$.

3 Topologická teorie datových typů

Tato teorie byla vytvořena na základě poznatků z [2] a [10].

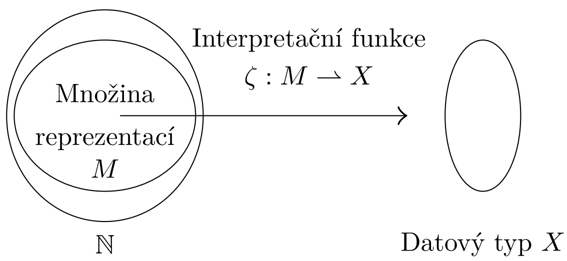
3.1 Datový typ

V programování si pod pojmem datový typ většinou představíme nějakou množinu hodnot spolu s operacemi, které s nimi můžeme provádět a způsobem jejich uložení v počítači. Kromě klasických datových typů, jako jsou například `char`, `int` nebo `float` v programovacím jazyce C, se můžeme ale setkat i s nekonečnými datovými typy. Některé programovací jazyky, jako jsou Common Lisp a Python, například umožňují pracovat s libovolně velkými přirozenými čísly. Jedinou limitací jejich velikosti je paměť použitého počítače. Dalším příkladem nekonečného datového typu jsou funkční datové typy, můžeme se s nimi setkat například v programovacím jazyce Haskell, umožňují nám reprezentovat data funkcemi.

V této práci budeme na datový typ nahlížet hlavně z pohledu vyčíslitelnosti. Při práci s nekonečnými datovými typy může způsob reprezentace ovlivnit, které operace jsme schopni naprogramovat. Proto definujeme datový typ tak, aby způsob reprezentace zohledňoval.

Abychom ale mohli obecně pracovat s datovým typem, musíme nejdříve stanovit nějaký jednotný jazyk, ve kterém dané reprezentace budeme definovat. Ať vybereme jakýkoliv programovací jazyk, stejně každá hodnota skončí v paměti počítače zapsaná pomocí nul a jedniček jako přirozené číslo. Přirozená čísla se proto jeví jako ideální způsob pro jednotný popis reprezentací.

Definice 5 (Datový typ)
Datovým typem nazveme množinu X spolu s podmnožinou $M \subseteq \mathbb{N}$ a surjektivním parciálním zobrazením $\zeta : M \rightharpoonup X$ nazvaným *interpretační funkce*. Množinu M nazveme *množina reprezentací* datového typu.



Obrázek 2: Vizualizace datového typu.

Množina reprezentací M vlastně představuje všechny správně syntakticky zapsané reprezentace daného datového typu. Datové typy, které zmíníme v rámci této práce, budou mít tuto množinu rekurzivní.

Příklad 6

Uvedeme příklad datového typu nezáporných racionálních čísel $\mathbb{Q}^+$. Jednou z možností by byla reprezentace s použitím párovací funkce z Kapitoly 2.4.1, vzhledem k čitelnosti interpretační funkce ale definujeme vlastní reprezentaci.

Jednotlivá racionální čísla reprezentujeme zřetězením binárních zápisů čitatele a jmenovatele tak, že menší z nich je doplněn nulami zleva, aby měly stejnou délku. V situacích, kdy bude doplněn nulami zleva čitatel, bychom neměli validní přirozené číslo, tudíž vždy před vzniklé binární číslo vložíme číslici 1.

Označme operaci celočíselného dělení $\lfloor a/b \rfloor = a//b$ a operaci zbytek po celočíselném dělení symbolem %. Pro $r \in \mathbb{N}$ položme $l = \lfloor \log_2 r \rfloor$, l tedy označuje počet binárních číslic zmenšený o jedna. Definujme množinu reprezentací $M = \{r \in \mathbb{N} \mid r - 2^l \text{ není dělitelné } 2^{l//2}\}$, interpretační funkci ζ pak pro $m \in M$ definujeme následovně:

$$\zeta(m) = \frac{(m - 2^l)//2^{l//2}}{(m - 2^l)\%2^{l//2}}$$

Uvažujme například číslo $453_{10} = 111000101_2$, zde $l = 8$ a číslo $197 = 453 - 2^8$ není dělitelné číslem $16 = 2^4$. Víme tedy, že $453 \in M$. Přirozené číslo 453 v tomto datovém typu reprezentuje racionální číslo $\zeta(453) = \frac{197//16}{197\%16} = \frac{12}{5}$.

V uvedeném příkladu není interpretační funkce prostá. Pokud bychom ale použili dodatečné podmínky na množinu reprezentací M, bylo by možné jednoznačnosti dosáhnout. Vytvořili bychom tím ale odlišný datový typ, ve kterém nezkrácené podíly přirozených čísel neinterpretují hodnoty z $\mathbb{Q}^+$. Obecně u reprezentace prvků datového typu už z jejich podstaty nemusí jít jednoznačnosti dosáhnout.

Další důležitou poznámkou je, že v tomto příkladu je interpretační funkce totální, to ale také není podmínkou. Interpretační funkce je dle definice parciální, protože existují datové typy, u nichž je ověřit správnost reprezentace nerozhodnutelným problémem. Jako příklad poslouží příklad datového typu reálných čísel s intervalovou reprezentací z Kapitoly 4.3.

3.2 Pozorovatelná množina

Cílem této části bude definovat na datovém typu topologii. Topologie nám poskytne důležitý úhel pohledu, který využijeme mimo jiné při zkoumání zobrazení datových typů. Konkrétně nám poslouží jako nástroj pro určování hranic vyčíslitelnosti.

Zvolíme-li totiž vhodně otevřené množiny, dosáhneme toho, že funkce, které se dají naprogramovat, budou spojité. Touto volbou budou právě pozorovatelné množiny. Pozorovatelné množiny jsou zjednodušeně takové množiny, ve kterých lze existence prvku ověřit nějakým výpočtem v konečném čase.

Definice 7 (Pozorovatelná množina)

Mějme datový typ X, množinu $U \subseteq X$ nazveme *pozorovatelná* (též *rozpoznatelná*), jestliže existuje částečně rekurzivní funkce f taková, že pro každé m z definičního oboru interpretační funkce ζ platí $\zeta(m) \in U$ právě když $f(m) = 0$. Funkci f nazveme *pozorovací.*

Věta 8

V datovém typu tvoří systém pozorovatelných množin bázi topologie.

Důkaz

Mějme datový typ X se systémem pozorovatelných množin σ. Nejdříve dokážeme, že $\bigcup_{U\in\sigma} U = X$. Tuto podmínku ověříme důkazem, že množina X je pozorovatelná. Poté ověříme, že pro každé dvě pozorovatelné množiny U_1 a U_2 platí, že $V = U_1 \cap U_2$ je také pozorovatelná množina.

Chceme dokázat, že existuje částečně rekurzivní funkce f, která pro každé $m \in dom(\zeta)$ splňuje $f(m) = 0$ právě když $\zeta(m) \in X$. Hledanou funkcí je konstantní funkce $f(x) = 0$. Zřejmě $f(m) = 0$ pro každé $m \in dom(\zeta)$ jelikož konstantní funkce vrací nulu vždy. Jestliže naopak $\zeta(m) \in X$, pak ze stejného důvodu také platí $f(m) = 0$. Je důležité si uvědomit, že výsledek funkce f pro čísla $m \notin dom(\zeta)$ neřešíme. Jelikož nulová funkce f je z definice částečně rekurzivní, tak existuje pozorovací funkce množiny X, a tedy X je pozorovatelná.

Abychom dokázali druhou podmínku, musíme najít pozorovací funkci g množiny V. Jelikož U_1 a U_2 jsou pozorovatelné množiny, existují jejich pozorovací funkce, označíme je popořadě f_1 a f_2. Pomocí těchto funkcí vytvoříme hledanou funkci g následovně:

$$g(x) = f_1(x) + f_2(x)$$

Operace sčítání je částečně rekurzivní funkce jak víme z Poznámky 2. Předpokládejme tedy, že pro $x \in dom(\zeta)$ platí $g(x) = 0$. Pro jednotlivé pozorovací funkce musí platit $f_1(x) = 0$ a $f_2(x) = 0$. Z toho plyne, že $\zeta(x) \in U_1$ a zároveň $\zeta(x) \in U_2$, takže $\zeta(x) \in V$. Také musí platit, že pro každý prvek $x \in dom(\zeta)$ takový, že $\zeta(x) \in V$ je $g(x) = 0$. Protože $\zeta(x) \in V$ a $V = U_1 \cap U_2$, pak $\zeta(x) \in U_1$, a tedy $f_1(x) = 0$ a také $\zeta(x) \in U_2$, a proto $f_2(x) = 0$, takže platí $g(x) = 0+0 = 0$.
□

Přirozeně se naskytne otázka, jestli pozorovatelné množiny tvoří také topologii. Už na základě mohutnosti množiny všech ČRF zjistíme, že tomu tak být nemusí. Víme, že každá částečně rekurzivní funkce vznikne použitím konečného počtu operací minimalizace,[3] složení a primitivní rekurze se základními rekurzivními funkcemi. Částečně rekurzivních funkcí je tedy spočetně mnoho. Aby byly splněny podmínky topologie, muselo by platit, že libovolné sjednocení pozorovatelných množin je pozorovatelná množina. To ale neplatí vždy, máme-li nekonečně prvků, dokážeme použitím Cantorovy diagonalizační metody, že existuje

[3] Každá částečně rekurzivní funkce lze zapsat v Kleenově normální formě, která dokonce používá minimalizaci nejvýše jednou.

nespočetně mnoho různých sjednocení těchto prvků. Jelikož částečně rekurzivních funkcí je spočetně mnoho a topologie musí být uzavřena na sjednocení, tak pozorovatelné množiny nemusí vždy tvořit topologii.

Pro demonstraci poslouží příklad datového typu $\mathbb{N}$ s interpretační funkcí definovanou jako identita. Pozorovatelné množiny zde tvoří bázi diskrétní topologie, nikoliv však topologii. Příkladem množiny, která není pozorovatelná, je $H = \{M \in \mathbb{N} \mid$ M představuje kód Turingova stroje, který se pro alespoň jeden svůj vstup nezastaví$\}$.

Je důležité si uvědomit, že pozorovatelné množiny tvoří sice jen bázi topologie, ale zase jsou uzavřené na spočetná sjednocení.

Když budeme hovořit o topologii datových typů, pokud nebude uvedeno jinak, budeme uvažovat topologii generovanou pozorovatelnými množinami.

Příklad 9

Uvažujme datový typ $X = \{0, 1\}$ s množinou reprezentací $M = \{1, 2, 3, 4\}$ a interpretační funkcí $\zeta(x) = x \,\%\, 2$. V tomto datovém typu je každá podmnožina X pozorovatelná. Pozorovací funkce množiny X je nulová funkce a pozorovací funkce $\emptyset$ je například $f(x) = 1$, která nevrátí nulu nikdy. Definujeme-li funkci pro absolutní hodnotu rozdílu jako $|x - y| = (x - y) + (y - x)$, pak pozorovací funkci f_1 množiny $\{1\}$ vytvoříme jako $f_1(x) = |x - 1| \times |x - 3|$, pro množinu $\{0\}$ pozorovací funkci f_2 vytvoříme jako $f_2(x) = |x - 2| \times |x - 4|$. Jelikož absolutní hodnota rozdílu, sčítání i odčítání jsou primitivně rekurzivní funkce, pak funkce f_1, f_2 jsou jistě částečně rekurzivní. V tomto příkladu pozorovatelné množiny tvoří diskrétní topologii.

Věta 10

V datovém typu X s konečnou množinou reprezentací M tvoří pozorovatelné množiny diskrétní topologii.

Důkaz

Toto tvrzení dokážeme zobecněním myšlenky z Příkladu 9. Díky Větě 8 stačí ověřit, že každá jednoprvková množina je pozorovatelná a že pozorovatelné množiny jsou uzavřeny na libovolná sjednocení.

Jelikož množina reprezentací M v daném datovém typu je konečná, tak každá hodnota je reprezentována konečným počtem reprezentací. Označme interpretační funkci ζ a pro $x \in X$ mějme množinu přirozených čísel $N = \{m \in M \mid \zeta(m) = x\}$, pak existuje pozorovací funkce f této množiny, kterou vytvoříme jako $f(x) = \prod_{n \in N} |x - n|$.

Jelikož interpretační funkce je surjektivní a M je konečná množina, tak X je také konečná množina a má $2^{|X|}$ podmnožin. Pozorovací funkci libovolného sjednocení tedy vytvoříme jako součin konečného počtu pozorovacích funkcí dílčích množin. Jestliže pro $m \in dom(\zeta)$ náleží $\zeta(m)$ do sjednocení, pak musí také náležet alespoň do jedné ze sjednocovaných množin, pozorovací funkce této množiny pro vstup m vrátí hodnotu 0, a tedy i výsledkem součinu bude 0. Jestliže naopak pozorovací funkce libovolného sjednocení vrátí hodnotu 0 pro $m \in dom(\zeta)$, tak jeden z činitelů součinu musí být nulový, tedy některá z dílčích pozorovacích funkcí vrátila výsledek 0, což znamená, že prvek m do této množiny patří a musí být také součástí sjednocení. $\square$

Víme, že systém množin, který tvoří topologii je zároveň i bází topologie. Pokud pozorovatelné množiny v datovém typu tvoří bázi diskrétní topologie, jedná se z hlediska vyčíslitelnosti o ideální situaci.

Věta 11
Mějme zobrazení datových typů $f : X \to Y$, jestliže pozorovatelné množiny v datovém typu X tvoří bázi diskrétní topologie, pak f je spojité zobrazení.

Důkaz
Na množině X uvažujeme topologii generovanou pozorovatelnými množinami, a tedy v tomto případě diskrétní topologii. Pro každou pozorovatelnou množinu $O \subseteq Y$ tedy platí, že $f^{-1}(O)$ je otevřená, protože v diskrétní topologii je každá množina otevřená, tedy i množina $f^{-1}(O)$. Zobrazení f je tedy spojité. $\square$

Nyní, když máme definovanou topologii datových typů, zbývá nám definovat, co to znamená pro zobrazení datových typů, že jsou naprogramovatelná.

3.3 Vyčíslitelná funkce

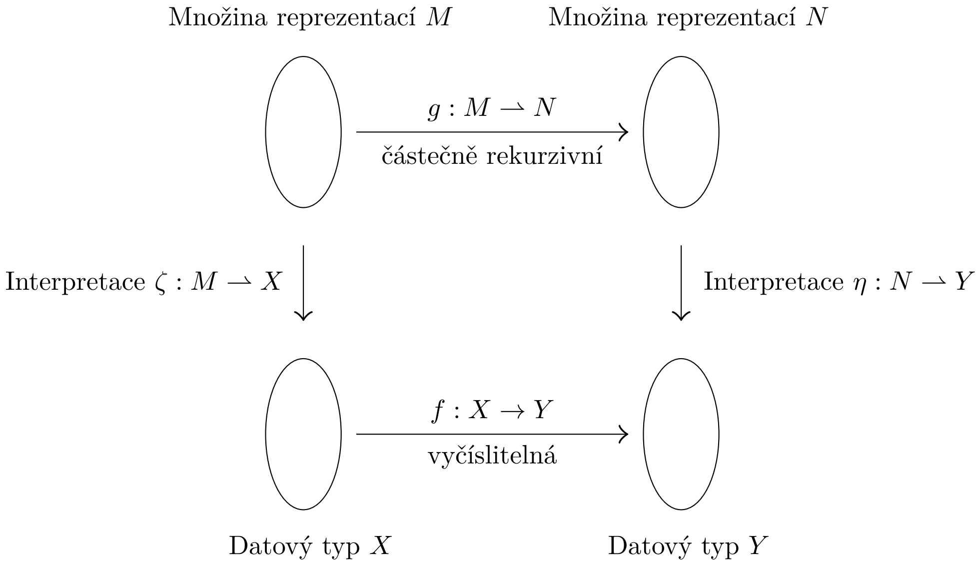
Zobrazení datových typů nás budou zajímat především ta, která lze ve skutečnosti naprogramovat. Takovým zobrazením se říká vyčíslitelná. Jestliže je zobrazení vyčíslitelné, pak dle intuice musí existovat algoritmus, který správně převede prvky mezi jednotlivými množinami reprezentací.

V této kapitole si také ukážeme, že v kontextu konečných datových typů skutečně na hranice vyčíslitelnosti nenarazíme, a tedy všechna jejich zobrazení lze naprogramovat.

Definice 12 (Vyčíslitelná funkce)
Mějme datové typy X a Y s interpretacemi $\eta : N \rightharpoonup Y$ a $\zeta : M \rightharpoonup X$. Řekneme, že funkce $f : X \to Y$ je *vyčíslitelná*, jestliže existuje parciální částečně rekurzivní funkce $g : M \rightharpoonup N$ taková, že pro každé $m \in dom(\zeta)$ platí $\eta(g(m)) = f(\zeta(m))$.

Funkce f tedy bude vyčíslitelná, jestliže diagram v Obrázku 3 komutuje.

Funkce g je parciální ze stejného důvodu jako interpretační funkce, u některých datových typů množiny reprezentací nejsou částečně rekurzivní a ověřit pro zadaný prvek, jestli do této množiny patří, může být nerozhodnutelným problémem. Dle definice také platí, že $dom(\zeta) = dom(g)$, protože vyčíslitelná funkce f



Obrázek 3: Diagram komutace vyčíslitelné funkce.

je definována pro všechny své hodnoty.

Věta 13

Mějme datové typy X a Y, jestliže zobrazení $f : X \to Y$ je vyčíslitelné, pak je spojité.

Důkaz

Díky Větě 8 víme, že pozorovatelné množiny tvoří bázi topologie a podle Věty 1 nám stačí ukázat, že pro každou pozorovatelnou množinu $O \subseteq Y$ platí, že $f^{-1}(O)$ je pozorovatelná množina v X. Musíme tedy dokázat, že existuje pozorovací funkce r množiny $f^{-1}(O) \subseteq X$.

Nechť ζ je interpretační funkce datového typu X a M je množina reprezentací. Interpretační funkci pro Y označme η. Jelikož funkce f je vyčíslitelná, existuje parciální částečně rekurzivní funkce g taková, že pro každé $m \in dom(\zeta)$ platí $\eta(g(m)) = f(\zeta(m))$. Jelikož O je pozorovatelná množina, pak existuje její pozorovací funkce, tu označíme h. Hledanou pozorovací funkci r vytvoříme složením těchto funkcí $r(x) = h(g(x))$. Ověříme nyní, že $r(x) = 0$ právě když $\zeta(m) \in f^{-1}(O)$ pro každé $m \in dom(\zeta)$.

1. $\zeta(m) \in f^{-1}(O) \Rightarrow r(x) = 0$

   Tento důkaz provedeme sporem. Kdyby pro nějaké $m \in dom(\zeta)$ platilo $\zeta(m) \in f^{-1}(O)$, ale $r(m) \neq 0$, pak dojdeme ke sporu, že funkce g splňuje rovnost $\eta(g(m)) = f(\zeta(m))$, nebo že h je pozorovací funkce množiny $O \subseteq Y$.

   Rozeberme nejdříve první zmíněnou možnost. Jestliže $r(m) = h(g(m)) \neq 0$, pak tedy $\eta(g(m)) \notin O$. Dle předpokladu $\zeta(m) \in f^{-1}(O)$ ale víme, že

$f(\zeta(m)) \in O$, dojdeme tedy ke sporu s tím, že $\eta(g(m)) = f(\zeta(m))$.

Nyní rozebereme druhou možnost. Víme, že $f(\zeta(m)) \in O$ a také $\eta(g(m)) \in O$. Zároveň platí, že $g(m) \in dom(\eta)$, ale $r(m) = h(g(m)) \neq 0$. To znamená, že $h(g(m))$ nevrátí nulu, ačkoliv $\eta(g(m)) \in O$, to je spor s tím, že h je pozorovací funkce množiny O.

2. $\zeta(m) \in f^{-1}(O) \Leftarrow r(x) = 0$

   Důkaz opět provedeme sporem. Dokážeme, že pro $m \in dom(\zeta)$ takové, že $\zeta(m) \notin f^{-1}(O)$ a $r(m) = 0$ buď platí, že funkce g nesplňuje rovnost $\eta(g(m)) = f(\zeta(m))$, nebo h není pozorovací funkce množiny $O \subseteq Y$.

   V prvním případě z $\zeta(m) \notin f^{-1}(O)$ plyne, že $f(\zeta(m)) \notin O$, ale dle $r(m) = h(g(m)) = 0$ víme, že $\eta(g(m)) \in O$. Vznikne nám tedy spor s tím, že $\eta(g(m)) = f(\zeta(m))$.

   V druhém případě, pokud by rovnost $r(m) = 0$ platila, tak by to znamenalo, že pro $n \in dom(\eta)$ neplatí, že $h(n) = 0$ právě když $\eta(n) \in O$, protože $g(m) \in dom(\eta)$ a $\eta(g(m)) \notin O$, ale $h(g(m)) = 0$ což je spor s tím, že h je pozorovací funkce množiny O, když vrátí hodnotu 0 pro prvek, který O nenáleží.

Ověřili jsme, že platí obě strany ekvivalence, takže r je pozorovací funkce množiny $f^{-1}(O)$. Dokázali jsme tedy, že každé vyčíslitelné zobrazení je spojité. $\square$

Opačný směr implikace Věty 13 neplatí. Některá spojitá zobrazení nemusí být vyčíslitelná. Toto tvrzení lze dokázat opět pomocí argumentu mohutnosti. V datovém typu je vyčíslitelných zobrazení pouze spočetně mnoho, zatímco spojitých zobrazení může být nespočetně mnoho.

Důsledek 14

Jestliže zobrazení datových typů f není spojité, není vyčíslitelné.

Ačkoliv tento důsledek přímo vyplývá z Věty 13, poskytuje nám užitečný nástroj při určování vyčíslitelnosti u některých zobrazení. Jestliže není splněn požadavek na spojitost, nelze vůbec uvažovat o vyčíslitelnosti tohoto zobrazení. Abychom si mohli přiblížit intuici za tímto tvrzením, představíme si Sierpińského datový typ.

Sierpińského datový typ S nabývá hodnot $\perp$ a $\top$ a je základním prvkem při studiu datových typů z hlediska topologie. Hodnota $\perp$ označuje nekonečný výpočet a $\top$ označuje konečný výpočet. Dle naší teorie tento datový typ formálně nelze definovat, požadujeme totiž, aby interpretační funkce byla surjektivní, zatímco hodnota $\perp$ je dosažena pouze ve chvíli, kdy hodnota interpretační funkce neexistuje.

Pokud bychom pro Sierpińského datový typ uvažovali topologii pozorovatelných množin, otevřenými množinami budou $S = \{\top, \perp\}$, $\emptyset$ a zřejmě $\{\top\}$, protože konečný výpočet je pozorovatelný, stejnou úvahou dojdeme k tomu, že množina $\{\perp\}$ nebude pozorovatelná.

Pro tento datový typ lze definovat čtyři různá zobrazení $f : S \to S$, z nichž pouze jedno není spojité, a je to právě to zobrazení, které nelze naprogramovat. Uvažujeme-li funkcionální programovací jazyk s líným vyhodnocováním, naprogramovatelná zobrazení jsou následující:

1. $f_1(x) = x$ představující identitu, pro konečný výpočet vrátí konečný výpočet, pro nekonečný výpočet se zacyklí,

2. $f_2(x) = \perp$ představující konstantní funkci, která se vždy zacyklí a svůj argument nevyhodnocuje,

3. $f_3(x) = \top$ představující konstantní funkci, která svůj argument také nevyhodnocuje, ale vždy vrátí konečný výpočet.

Zbývá nám takové zobrazení, které se pro konečný výpočet na vstupu zacyklí $f_4(\top) = \perp$, ale pokud na vstupu dostane nekonečný výpočet, vrátí nám konečný výpočet $f_4(\perp) = \top$. Taková funkce zřejmě naprogramovat nejde, jelikož neexistuje algoritmus, který by obecně určil, jestli hodnota na vstupu představuje konečný, nebo nekonečný výpočet.

Věta 15

Mějme datový typ X s konečnou množinou reprezentací M, každé zobrazení $f : X \to X$ je vyčíslitelné.

Důkaz

Z Věty 10 víme, že X má diskrétní topologii a z Věty 11 víme, že f je spojité. Označme interpretační funkci ζ, jelikož předpoklad na spojitost je splněn, dokážeme, že existuje funkce g taková, že pro každé $m \in dom(\zeta)$ platí $\zeta(g(m)) = f(\zeta(m))$.

Jelikož množina reprezentací M je konečná, je konečná také $dom(\zeta) \subseteq M$, její velikost označíme $n = |dom(\zeta)|$. Pro každé $a \in dom(\zeta)$ existuje nějaké $b \in dom(\zeta)$ takové, že $\zeta(b) = f(\zeta(a))$. Tento prvek skutečně existuje, protože interpretační funkce je surjektivní a vyčíslitelná funkce f je totální. Stačí tedy vytvořit funkci g tak, že pro každé a vrátí příslušnou hodnotu b.

Pro každé $a_i \in dom(\zeta)$ označme tuto hodnotu jako b_i a mějme primitivně rekurzivní funkci $f_i(x) = 1 - sgn(|x - a_i|)$, která pro $f_i(a_i) = 1$, jinak $f(x) = 0$. Hledanou funkci g poté vytvoříme následovně:

$$g(x) = b_1 f_1(x) + b_2 f_2(x) + \ldots + b_n f_n(x)$$

Pro každé $i \leq n$ platí $g(a_i) = b_i$, protože pro každé $j \leq n$ takové, že $i \neq j$ je součin $b_j f_j(x) = 0$. To víme, jelikož $f_j(a_i) = 0$, zbyde nám tedy součin $b_i f_i(a_i)$, zde víme, že $f_i(a_i) = 1$ a proto $g(a_i) = b_i$. $\square$

3.4 Součinový datový typ

Abychom nezůstali pouze u unárních operací, a mohli definovat na datových typech i operace s vyšší aritou, definujeme součinový datový typ. Součinový datový typ představuje kartézský součin dílčích datových typů.

Definice 16 (Součinový datový typ)

Mějme datové typy X_1 a X_2 s interpretačními funkcemi $\zeta_1 : M_1 \rightharpoonup X_1$ a $\zeta_2 : M_2 \rightharpoonup X_2$. Nechť $d : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ je párovací funkce s projekcemi p_1, p_2. *Součinový datový typ* nazveme množinu $D = X_1 \times X_2$ s interpretační funkcí $\theta(x) = (\zeta_1(p_1(x)), \zeta_2(p_2(x)))$ a množinou reprezentací $M = d(M_1, M_2)$.

Ověřme nejdříve, že součinový datový typ splňuje definici datového typu. Musí platit, že interpretační funkce θ je surjektivní, a tedy pro každou dvojici $(x_1, x_2) \in D$ existuje $m \in M$ tak, že $\theta(m) = (x_1, x_2)$. To jistě platí, protože párovací funkce d je bijekce, dílčí interpretační funkce ζ_1 a ζ_2 jsou surjektivní a kompozice surjektivních zobrazení je také surjektivní zobrazení.

Na základě intuice by pozorovatelné množiny měly tvořit bázi topologie součinu. Toto tvrzení se nám ale nepodařilo dokázat, proto pouze dokážeme, že topologie generovaná pozorovatelnými množinami je silnější, než topologie součinu. Stále bude platit, že každá funkce, která je spojitá v topologii součinu, je také spojitá v topologii generované pozorovatelnými množinami, ale přibudou další zobrazení, která jsou spojitá, ale nelze je naprogramovat.

Věta 17

Topologie generovaná pozorovatelnými množinami součinového datového typu je silnější než topologie součinu.

Důkaz

Máme datový typ $D = X_1 \times X_2$, označme interpretační funkci $\theta(x) = (\zeta_1(p_1(x)), \zeta_2(p_2(x)))$ a množinu reprezentací $M = d(M_1, M_2)$, kde d je párovací funkce s projekcemi p_1, p_2 a ζ_1, ζ_2 jsou popořadě interpretační funkce datových typů X_1, X_2. Systém pozorovatelných množin označíme σ.

Dle věty 8 víme, že systém pozorovatelných množin σ tvoří bázi topologie. Zbývá dokázat, že generovaná topologie je silnější než topologie součinu. To dokážeme ověřením, že kartézské projekce π_1 a π_2 jsou spojitá zobrazení.

Mějme pozorovatelné množiny $A_1 \subseteq X_1$ a $A_2 \subseteq X_2$ s pozorovacími funkcemi popořadě f_1 a f_2. Dokážeme, že množiny $\pi_1^{-1}(A_1)$ a $\pi_2^{-1}(A_2)$ jsou pozorovatelné. Pozorovací funkci g_1 pro množinu $\pi_1^{-1}(A_1)$ vytvoříme jako $g_1(x) = f_1(p_1(x))$. Pro každé $m \in dom(\theta)$ takové, že $\theta(m) \in \pi_1^{-1}(A_1)$ skutečně platí $g_1(m) = 0$, protože $p_1(m) \in dom(\zeta_1)$ a jelikož $\zeta_1(p_1(m)) \in A_1$, tak dle definice pozorovací funkce f_1 platí $f_1(p_1(m)) = 0$.

Naopak jestliže $g_1(x) = 0$ a $x \in dom(\theta)$, ověříme, že $\theta(x) \in \pi_1^{-1}(A_1)$. Jelikož z $x \in dom(\theta)$ vyplývá, že $p_1(x) \in dom(\zeta_1)$, tak potom platí $f_1(p_1(x)) = 0$ právě když $\zeta_1(p_1(x)) \in A_1$. Jelikož funkce g_1, neboli $f_1(p_1(x))$ vrátila hodnotu 0, tak

$\zeta_1(p_1(x)) \in A_1$, a tudíž $\theta(x) \in \pi_1^{-1}(A_1)$.

Pro množinu $\pi_2^{-1}(A_2)$ pozorovací funkci g_2 definujeme jako $g_2(x) = f_2(p_2(x))$ a důkaz provedeme obdobně. $\square$

Poznámka 18

Abychom pozorovací funkce f_1, f_2 mohli považovat za částečně rekurzivní, museli bychom nejdříve dokázat, že samotné projekce p_1 a p_2 párovací funkce jsou částečně rekurzivní. Ačkoliv tomu tak je, v rámci této práce pro složitost toto ověření provádět nebudeme a budeme předpokládat, že p_1 a p_2 definované v Kapitole 2.4.1 jsou skutečně částečně rekurzivní funkce.

Příklad 19

Mějme součinový datový typ $D = X \times X$, kde X je příklad datového typu kladných racionálních čísel s interpretační funkcí $\zeta(x) = \dfrac{p_1(x)}{p_2(x)}$, kde $p_1(x)$ a $p_2(x)$ jsou projekce párovací funkce d. Mějme zobrazení $f : D \to X$ definované jako $f(x) = |\pi_1(x) - \pi_2(x)|$, ukážeme si že toto zobrazení je vyčíslitelné. Hledanou ČRF g vytvoříme podle následující rovnosti:

$$\frac{a}{b} - \frac{c}{d} = \frac{a \cdot d - b \cdot c}{b \cdot d}$$

$$g(x) = d(p_1(p_1(x)) \cdot p_2(p_2(x)) - p_1(p_2(x)) \cdot p_2(p_1(x)), p_2(p_2(x)) \cdot p_1(p_2(x)))$$

kde operace odčítání je pomocí funkce předchůdce P definována následovně:

$$f(x,0) = x \qquad\qquad f(x,n+1) = P(f(x,n))$$

V situacích, kdy výsledkem by mělo být záporné číslo, funkce vrátí hodnotu 0, protože $P(0) = 0$.

Mějme čísla $\frac{3}{4}$ a $\frac{1}{4}$ reprezentovaná čísly $d(3,4) = \dfrac{7^2+9+4}{2} = 31$ a $d(1,4) = \dfrac{5^2+3+4}{2} = 16$. Prvek součinového datového typu $\left(\frac{3}{4}, \frac{1}{4}\right) \in D$ je reprezentovaný hodnotou $d(31,16) = \dfrac{47^2+48+31}{2} = 1144$. Definovanou operaci odčítání vypočítáme jako $g(1144) = d(3 \cdot 4 - 4 \cdot 1, 4 \cdot 4) = d(8,16) = 308$.

4 Datový typ reálných čísel

Reálná čísla značíme symbolem $\mathbb{R}$, jsou sjednocením množiny racionálních čísel $\mathbb{Q}$ s množinou iracionálních čísel $\mathbb{I}$. Motivací za vznikem tohoto číselného oboru a důvodů, proč racionální čísla nestačí, je nespočet:

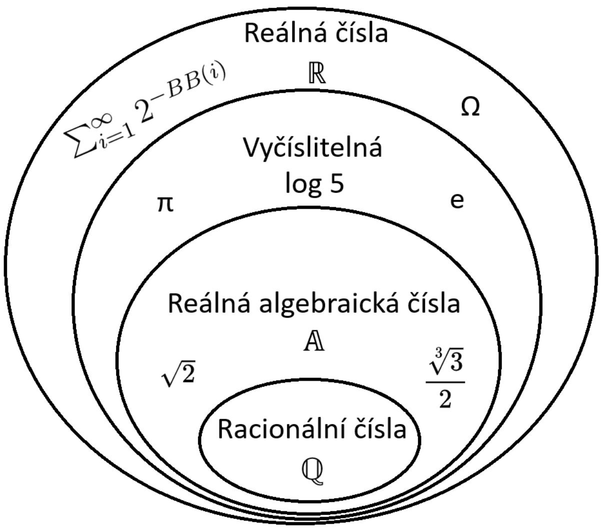
- Nejsme schopni spočítat délku úhlopříčky ve čtverci.
- Nelze určit poměr mezi obvodem kružnice a jejím poloměrem.

- Existují konvergentní posloupnosti racionálních čísel, které nemají limitu.
- Číselná osa racionálních čísel obsahuje „díry“.[4]

Jedním z hlavních cílů této práce je prozkoumat reprezentace reálných čísel a jejich vlastnosti, především tedy schopnost definovat základní aritmetické operace. Jelikož v počítači nemáme k dispozici nekonečné množství paměti, musíme být schopni popsat jednotlivá čísla konečným množstvím informací. Dosáhneme toho například tím, že je budeme chápat jako programy, které pro zadaný index vrátí cifru na dané pozici.

Existují různé reprezentace, které tento způsob umožňují. Ať už ale zvolíme jakoukoliv reprezentaci, žádná nám neumožní reprezentovat všechna reálná čísla. Mohli bychom použít argument o nespočetnosti reálných čísel a spočetnosti množiny všech algoritmů. Reálná čísla, která naopak tímto způsobem lze reprezentovat, se nazývají vyčíslitelná.

Turing ve své práci [12] definoval vyčíslitelná čísla jako taková, pro která existuje Turingův stroj, jenž se pro vstup n zastaví a na pásce bude mít vypsánu n-tou cifru tohoto čísla. My se budeme touto definicí inspirovat a reálné číslo nazveme *vyčíslitelné*, jestliže existuje částečně rekurzivní funkce, jejíž výpočet pro každý vstup n skončí a vrátí n-tou cifru tohoto čísla. Množina vyčíslitelných reálných čísel je zachycena na Obrázku 4.



Obrázek 4: Vennův diagram podmnožin reálných čísel.

Symbolem $\mathbb{A}$ značíme reálná algebraická čísla. Algebraická čísla jsou komplexní čísla, která získáme jako kořeny polynomiálních rovnic jedné proměnné

[4]Rozdělíme-li racionální čísla do dvou neprázdných množin A, B tak, že každý prvek z B je větší než prvky v A, pak B nemusí mít minimum a A nemusí mít maximum. Například $A = \{x \mid x \in \mathbb{Q} \text{ a platí } x^2 < 2\}$ a $B = \{x \mid x \in \mathbb{Q} \text{ a platí } x^2 > 2\}$. Dírou je myšleno, že neexistuje racionální číslo, které by tvořilo hranici mezi těmito množinami. Tyto díry jsou právě iracionální čísla [11].

s celočíselnými koeficienty. Těchto čísel je spočetně mnoho [13], použitím algoritmů na výpočet kořenů polynomu [14] je skutečně můžeme výpočtem v konečném čase aproximovat na libovolnou přesnost. Doplňkem algebraických čísel je množina *transcendentních* čísel, která je nespočetná.

Některá transcendentní čísla jsou vyčíslitelná, je jich spočetně mnoho a patří mezi ně například π, e, $log_{10}5$. Většina transcendentních čísel ale vyčíslitelná není. Příkladem takového čísla je *Chaitinova konstanta* Ω [15] nebo třeba $\sum_{i=1}^{\infty} 2^{-BB(i)}$, kde BB označuje funkci známou jako *BusyBeaver* [16]. Hodnota $BB(n)$ označuje maximální možný počet kroků, který provede Turingův stroj o velikosti n předtím než se zastaví. V obou případech platí, že kdybychom byli schopni vypočítat tato čísla s libovolnou přesností, tak vyřešíme nerozhodnutelný Problém zastavení. Proto, když budeme hovořit o reálných číslech v kontextu datového typu, budeme mít na mysli množinu všech vyčíslitelných reálných čísel.

V následujících kapitolách si představíme některé datové typy reálných čísel a rozebereme jejich hranice vyčíslitelnosti především v kontextu základních aritmetických operací.

4.1 Reprezentace nekonečným binárním rozvojem

Nejdříve si představíme tu nejintuitivnější reprezentaci reálných čísel, jednotlivá čísla budeme reprezentovat pomocí nekonečného číselného rozvoje s přirozeným základem.

Každé reálné číslo se skládá z celočíselné a desetinné části. Můžeme si jej tedy představit jako jejich součet. Nyní budeme pro srozumitelnost demonstrace pracovat pouze s reálnými čísly z uzavřeného intervalu $[0, 1]$, neboli pouze s desetinnou částí.

Obecně při reprezentaci reálných čísel nekonečným číselným rozvojem o základu b zapisujeme čísla z intervalu $[0, 1]$ ve formě:[5]

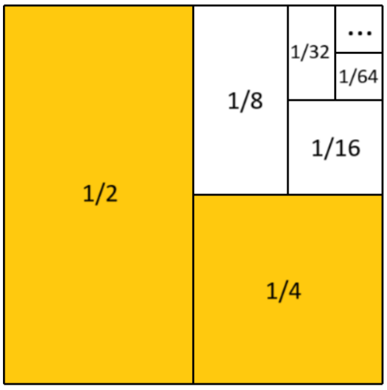
$$(0, a_1a_2a_3\ldots)_b = \sum_{k=1}^{+\infty} a_k b^{-k}$$

V této práci pro přehlednost zvolíme základ $b = 2$, tedy každá cifra $a_k \in \{0, 1\}$. Pokud bychom zvolili jiný přirozený základ, nebude to mít vliv na to, které funkce budou vyčíslitelné, protože základy mezi sebou můžeme jednoduše převádět.

Uvažujme jednotkový čtverec, který je rozdělen na obdélníky o obsahu 2^{-k} pro každé $k \in \mathbb{N}$ a ty obdélníky, jejichž cifra $a_k = 1$, jsou vybarveny. Výsledné reálné číslo si potom můžeme představit jako součet obsahů, které zabírají právě vybarvené obdélníky. Jako příklad si uvedeme číslo $(0, 11000\ldots)_2 = 2^{-1} + 2^{-2} = (\frac{3}{4})_{10}$, které je zobrazeno na Obrázku 5.

Reálná čísla s touto reprezentací skutečně tvoří datový typ. Definičním oborem interpretační funkce jsou všechna přirozená čísla, která použitím Gödelova kódovní představují zápis úplné částečně rekurzivní funkce.

[5] Číslo 1 lze také zapsat jako 1,000...

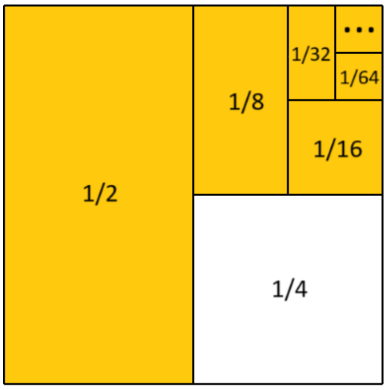


Obrázek 5: Reálné číslo zapsané v jednotkovém čtverci.

Mějme univerzální částečně rekurzivní funkci U, tato funkce pro Gödelovo číslo x, reprezentující částečně rekurzivní funkci f a vstup y, vrátí hodnotu této funkce pro zadaný argument, tedy $f(y)$. Interpretační funkci ζ poté definujeme jako:

$$\zeta(x) = \sum_{k=1}^{+\infty} U(x,k) \cdot 2^{-k}$$

V tomto datovém typu tato funkce nebude prostá, protože existují čísla, která nemají pouze jednu možnost reprezentace. Konkrétně jsou to všechna čísla, která vzniknou jako součet konečného počtu záporných mocnin svého základu, říká se jim *dyadická*. Například druhým způsobem reprezentace čísla $(\frac{3}{4})_{10}$ je $(0,1011\ldots)_2$, jeho zápis v jednotkovém čtverci vidíme na Obrázku 6. U čísel, která nejsou dya-



Obrázek 6: Alternativní reprezentace dyadického čísla.

dická, tedy u iracionálních čísel nebo například čísla $(\frac{1}{3})_{10} = (0,01010\ldots)_2$, tuto dvojitou reprezentaci nemáme.

Pokud bychom chtěli bezpodmínečně dosáhnout unikátní reprezentace reálných čísel, mohlo by se zdát, že řešením je zakázat reprezentaci končící nekonečnou sekvencí jedniček. Interpretační funkce ale ani v tomto případě nebude prostá, protože existuje více způsobů, jak vytvořit částečně rekurzivní funkci, která pro stejné vstupy vrací stejné výstupy.

Nyní charakterizujeme topologii generovanou pozorovatelnými množnami.

Věta 20

Pozorovatelné množiny v datovém typu reálných čísel s reprezentací nekonečným binárním rozvojem generují topologii silnější, než je standardní topologie.

Důkaz

Nejdříve dokážeme, že každý otevřený interval je pozorovatelný a topologie generovaná pozorovatelnými množinami je tedy silnější, než standardní topologie. V druhé části si naznačíme jak bychom postupovali při důkazu vedoucímu k tomu, že každá pozorovatelná množina je spočetným sjednocením otevřených intervalů. Potom by platilo, že topologie generovaná pozorovatelnými množinami je standardní topologie.

1. Ověříme, že každý otevřený interval je pozorovatelný.

   Označme tento datový typ X, interpretační funkci ζ a mějme otevřený interval $(\zeta(a), \zeta(b))$. Abychom ověřili, že prvek $\zeta(x) \in (\zeta(a), \zeta(b))$ stačí dokázat, že $\zeta(x) \in (\zeta(a), +\infty)$ a zároveň $\zeta(x) \in (-\infty, \zeta(b))$. Nechť x označuje Gödelovo číslo, jestliže $\zeta(x) \in (\zeta(a), \zeta(b))$, pak $\zeta(x) > \zeta(a)$, a tedy existuje $n \in \mathbb{N}$ takové, že $\sum_{i=1}^{n} U(x,i) > \zeta(a)$, kde U představuje univerzální částečně rekurzivní funkci. Zároveň jestliže platí $\zeta(x) \in (\zeta(a), \zeta(b))$, pak $\zeta(x) < \zeta(b)$, a tedy existuje $n \in \mathbb{N}$ takové, že $\sum_{i=1}^{n} U(b,i) > \zeta(x)$.

   Mějme otevřený interval $(\zeta(a), +\infty)$ pro $a \in dom(\zeta)$, za pomoci $\overline{sgn}(x) = 1 - sgn(x)$, $\overline{sgnr}(x) = \overline{sgn}(p_1(x))$ a rozšířené univerzální funkce U', která nám vrátí racionální číslo $U'(x,i) = d(U(x,i),1)$ vytvoříme jeho pozorovací funkci f_1 jako:

   $$f_1(x) = \overline{sgn}\left(\mu y \left[\overline{sgnr}\left(\sum_{i=1}^{y} U'(x,i)\cdot 2^{-i} - \sum_{i=1}^{y}\Big(U'(a,i)\cdot 2^{-i}\Big) - 2^{-y}\right) = 0\right]\right)$$

   Pro otevřený interval $(-\infty, \zeta(b))$ a $b \in dom(\zeta)$, pozorovací funkci f_2 vytvoříme jako:

   $$f_2(x) = \overline{sgn}\left(\mu y \left[\overline{sgnr}\left(\sum_{i=1}^{y} U'(b,i)\cdot 2^{-i} - \sum_{i=1}^{y}\Big(U'(x,i)\cdot 2^{-i}\Big) - 2^{-y}\right) = 0\right]\right)$$

   Výsledná pozorovací funkce f množiny $(\zeta(a), \zeta(b))$ bude definována následovně:

   $$f(x) = f_1(x) + f_2(x)$$

   Zřejmě funkce f vrátí 0 pouze pokud prokazatelně platí $\zeta(a) < \zeta(x) < \zeta(b)$ neboli pokud obě funkce f_1, f_2 vrátí 0. To se stane právě když existuje $y \in \mathbb{N}_0$ takové, že rozdíl sum, představující hodnotu jednotlivých reálných čísel po prozkoumání y cifer, bude větší 2^{-y}. Toto ověření je podmíněno existencí dyadických čísel, která se mohou lišit v jedné cifře, přičemž zdánlivě vyšší číslo je zakončeno nekonečnou posloupností nul a zdánlivě nižší číslo končí nekonečnou posloupností jedniček, a tedy jsou si rovny.

Víme, že použitá suma je ČRF, násobení racionálních čísel také a z Příkladu 19 víme, že i odčítání racionálních čísel, pak f_1 je také ČRF.

2. Pokusíme se ověřit, že každá pozorovatelná množina je spočetným sjednocením otevřených intervalů.

   Mějme pozorovatelnou množinu O a její pozorovací funkci f. Pro $x \in dom(\zeta)$, takové že $\zeta(x) \in O$ funkce f po konečném počtu kroků skončí. Znamená to, že skončí po přečtení konečného množství cifer. Označíme-li tyto cifry $a_1, \ldots, a_n \in \{0, 1\}$, pak platí:

$$\sum_{i=1}^{n} a_i \cdot 2^{-i} \leq \zeta(x) \leq \ 2^n + \sum_{i=1}^{n} a_i \cdot 2^{-i}$$

   Tato nerovnost nám říká, že pokud pozorovací funkce f vrátí hodnotu 0 po přečtení n cifer čísla $\zeta(x)$, musí vrátit hodnotu 0 také pro všechna čísla, která náleží uzavřenému intervalu $Z = [\sum_{i=1}^{n} a_i \cdot 2^{-i}, 2^n + \sum_{i=1}^{n} a_i \cdot 2^{-i}]$. Platí, že $Z \subseteq O$, jinak by f nebyla pozorovací funkce, protože by existoval prvek $z \in dom(\zeta)$, pro který by platilo $f(z) = 0$, ale $f(z) \notin O$.

   Nechť $\zeta(x)$ je číslo, které není dyadické, například iracionální, víme, že $\zeta(z) \neq \sum_{i=1}^{n} a_i \cdot 2^{-i}$ a stejně tak, že $\zeta(z) \neq 2^n + \sum_{i=1}^{n} a_i \cdot 2^{-i}$. To znamená, že $\zeta(z) \in (\sum_{i=1}^{n} a_i \cdot 2^{-i}, 2^n + \sum_{i=1}^{n} a_i \cdot 2^{-i})$, a tedy patří do O spolu s otevřeným intervalem.

   Pokud by se nám tato skutečnost podařila dokázat i pro dyadická čísla, pak by pro každý prvek platilo, že do množiny O patří spolu s otevřeným intervalem. Jelikož prvků v množině O je nevýše spočetně mnoho, znamenalo by to, že O se skládá ze sjednocení nejvýše spočetně mnoha otevřených intervalů, a tedy že pozorovatelné množiny generují právě standardní topologii.

□

Kdybychom se chtěli pokusit definovat operaci sčítání, provedeme to tak, že sečteme cifry obou čísel na daných pozicích. Pokud obě cifry budou rovny jedné, musíme ale přenést jedničku do vyššího řádu, proto se při sčítání začíná od poslední cifry. Zde ale máme nekonečný rozvoj, narážíme tedy na situaci, kdy výpočet cifry by obnášel kontrolu nekonečného množství cifer sčítaných čísel.

Mějme součet $(\frac{1}{3})_{10} + (\frac{2}{3})_{10} = (0,01010\ldots)_2 + (0,10101\ldots)_2$, víme, že binární rozvoje těchto čísel se budou periodicky opakovat do nekonečna, a tedy číslo $1,000\ldots = 0,111\ldots$ bude výsledkem tohoto součtu. Pokud si ale uvědomíme, že tato čísla máme reprezentována jako funkce, které pro daný index vrátí cifru na dané pozici, není možné zjistit, jestli někde na i-té pozici místo nuly nedostaneme jedničku či naopak.

Z nemožnosti definovat sčítání v této reprezentaci také vyplývá, že nelze definovat operaci násobení, a tedy ani mocnění. V každé z těchto operací narazíme na stejný problém, že porušení zdánlivě nekonečné posloupnosti čísel může vést ke změně první cifry výsledného čísla.

Ještě než si představíme přívětivější reprezentace, rozebereme si vlastnosti nového datového typu, který by vznikl zakázáním reprezentace reálných čísel pomocí nekonečných jedniček. Nejen, že jednotlivé operace by stále nešlo definovat, ale tento krok by vedl také ke vzniku nových pozorovatelných množin.

Příkladem nové pozorovatelné množiny by byla $[\frac{1}{2}, 1)$. Jelikož validní reprezentace čísla $\frac{1}{2}$ je pouze $(0,100\ldots)_2$, už podle první cifry bychom byli schopni rozhodnout, zda zadané číslo do této množiny patří, či nikoliv. Pokud by hodnota první cifry byla jedna, je jisté, že zadané číslo je větší, nebo rovno číslu $\frac{1}{2}$, a tedy do množiny patří. Zatímco čísla s první cifrou nula do množiny pak nepatří, protože $(0,0111\ldots)_2$ není v tomto kontextu validní reálné číslo. Jedna z věcí, kterou bychom tím narušili, je například věta o limitě součtu posloupností.

Příklad 21

Mějme konstantní posloupnost $x_n = \frac{1}{6}$ a konvergentní posloupnost $y_n = \frac{1}{12} - \frac{1}{n}$, která se limitně blíží číslu $\frac{1}{12}$. Dle věty o limitě součtu posloupností platí následující rovnost:

$$\lim_{n\to\infty}(x_n + y_n) = \lim_{n\to\infty} x_n + \lim_{n\to\infty} y_n$$

Jelikož čísla $\frac{1}{6}$ a $\frac{1}{12}$ nejsou dyadická, pravá strana rovnice se nezmění a výsledek vyjde podle očekávání:

$$\lim_{n\to\infty}\frac{1}{6} + \lim_{n\to\infty}(\frac{1}{12} - \frac{1}{n}) = \frac{1}{6} + \frac{1}{12} = \frac{1}{4}$$

Vložením podmínky ale vznikla nová okolí, například $[\frac{1}{4}, x)$ pro libovolné $x > \frac{1}{4}$. Znamená to, že existuje okolí U, pro které neexistuje $n_0 \in \mathbb{N}$ takové, že pro všechna $n \in \mathbb{N}$, pro která platí $n \geq n_0$, platí také $x_n \in U$. Jinými slovy pro danou posloupnost zde limita neexistuje, a tedy základní očekávaná vlastnost limit je porušena.

$$\lim_{n\to\infty}(\frac{1}{6} + \frac{1}{12} - \frac{1}{n}) = \lim_{n\to\infty}(\frac{1}{4} - \frac{1}{n}) \neq \frac{1}{4}$$

4.2 Reprezentace vnořenými intervaly

Další z možností, jak reprezentovat reálná čísla, je pomocí posloupnosti vnořených uzavřených intervalů s racionálními koncovými body. Tento způsob staví na Cantorově principu vložených intervalů, větu která z něj vyplývá si dokážeme.

Věta 22

Mějme posloupnost omezených uzavřených intervalů I_n, pro kterou platí:

1. $I_{i+1} \subset I_i$ *pro každé* $i \in \mathbb{N}$*,*

2. pro každé $\epsilon \in \mathbb{R}$ *takové, že* $\epsilon > 0$ *existuje* $j \in \mathbb{N}$ *takové, že* $|I_j| < \epsilon$*.*

Existuje právě jeden bod x*, který leží ve všech intervalech, značíme ho jako průnik intervalů* $x = \bigcap_{n=1}^{\infty} I_n$*.*

Důkaz

Dokážeme, že takový bod skutečně existuje a že je právě jeden. Pro posloupnost intervalů $I_n = [a_n, b_n]$ mějme množinu všech levých krajních bodů A a množinu všech pravých krajních bodů B. Víme, že pro každé $i, j \in \mathbb{N}$ platí, že $a_i < b_j$. Také víme, že každá neprázdná shora omezená množina reálných čísel má supremum, tedy i množina A, označme tuto hodnotu $sup(A)$. Každá neprázdná zdola omezená množina má naopak infimum, platí to tedy i pro B, označíme ji $inf(B)$. Z definice suprema a infima víme, že pro každé $k \in \mathbb{N}$ platí, že $a_k \leq sup(A)$ a $inf(B) \leq b_k$. Existují tedy hodnoty, které jsou v každém intervalu posloupnosti I_n, jsou to právě hodnoty z intervalu $[sup(A), inf(B)]$. S využitím druhé podmínky víme, že pro každé ϵ musí platit, že $inf(B) - sup(A) < \epsilon$, tato podmínka je splněna pouze pokud $inf(B) = sup(A)$, a tedy v uzavřeném intervalu $[sup(A), inf(B)]$ je pouze jediný bod. $\square$

Pro otevřené intervaly obdobné tvrzení neplatí. Uvažujme posloupnost intervalů $I_n = (0, \frac{1}{n})$, zřejmě platí $(0, 1) \supset (0, \frac{1}{2}) \supset (0, \frac{1}{3}) \supset ...$ a $\lim_{n\to\infty} |I_n| = 0$, průnikem takovéto posloupnosti intervalů je ale prázdný interval, protože posloupnost konverguje k nule, ta ale nepatří do žádného z intervalů, protože jsou otevřené.

Uvažujme tedy datový typ reálných čísel s touto reprezentací. Nejdříve si rozebereme množinu reprezentací M. Každý prvek $m \in M$ představuje posloupnost intervalů racionálních čísel. Posloupnost intervalů reprezentujeme jako dvojici posloupností racionálních čísel zakódovanou pomocí párovací funkce d s projekcemi p_1, p_2. Abychom mohli dvojici posloupností zakódovat pomocí párovací funkce, nejdříve je musíme pomocí Gödelova číslování ϕ převést na přirozená čísla. Jednotlivé posloupnosti budou částečně rekurzivní funkce splňující podmínky zmíněné výše. Interpretační funkci tohoto datového typu vytvoříme za pomoci univerzální částečně rekurzivní funkce U následovně:

$$\zeta(x) = \bigcap_{i=1}^{\infty} [U(p_1(x), i), U(p_2(x), i)]$$

Jelikož ověřit, jestli dané posloupnosti racionálních čísel jsou konvergentní, nebo jestli intervaly, které vytvářejí, jsou vložené, je nerozhodnutelným problémem, tak interpretační funkce tohoto datového typu nebude totální. Interpretační funkce nebude ani injektivní, protože existují různé posloupnosti intervalů racionálních čísel konvergující ke stejnému reálnému číslu.

Ukážeme si, že pozorovatelné množiny v této reprezentaci generují standardní topologii. Zobrazení, která jsou spojitá dle definic z matematické analýzy budou tedy v tomto kontextu spojitá také.

Věta 23

Pozorovatelné množiny datového typu reálných čísel s reprezentací vnořenými intervaly generují standardní topologii.

Důkaz

Ověření bude probíhat obdobným způsobem jako v reprezentaci nekonečným binárním rozvojem.

1. Ověříme, že každý otevřený interval je pozorovatelný.

   Máme-li reálné číslo x reprezentované posloupností I_n a otevřený interval (a, b) pro $a, b \in \mathbb{R}$, pak, jestliže $x \in (a, b)$, tak tuto skutečnost po konečném počtu kroků lze ověřit. Idea důkazu opět spočívá v tom, že nejdříve ověříme, že $x \in (a, +\infty)$, a poté, že $x \in (-\infty, b)$. Jelikož $a < b$, tak plyne $x \in (a, b) = (-\infty, b) \cap (a, +\infty)$.

   Pokud tedy $x \in (a, b)$, pak víme z podmínek reprezentace, že existuje $j \in \mathbb{N}$ takové, že $|I_j| < |x - a|$. Jelikož pro tento interval $I_j = [a_j, b_j]$ platí, že $a_j > a$, musí pak platit $x > a$, což znamená, že $x \in (a, +\infty)$. Také víme, že existuje $k \in \mathbb{N}$ takové, že $|I_k| < |b - x|$ a pro interval $I_k = [a_k, b_k]$ platí, že $b_k < b$ tedy musí platit $x < b$, což znamená, že $x \in (-\infty, b)$.

   Pozorovací funkci f intervalu $(\zeta(a), \zeta(b))$ vytvoříme jako $f(x) = f_1(x) + f_2(x)$, kde f_1 a f_2 jsou popořadě pozorovací funkce intervalů $(\zeta(a), +\infty)$ a $(-\infty, \zeta(b))$. Pro sestrojení f_1 využijeme následující funkci e:

   $$e(x, a) = \mu y \left[\overline{sgn}\left(U(x, y) - U(a, y)\right) = 0\right],$$

   kde U představuje univerzální částečně rekurzivní funkci. Tato funkce vrátí 0 právě když existuje cifra y v posloupnostech kódovanými čísly a, x taková, že racionální číslo $U(x, y) > U(a, y)$.

   $$f_1(x) = \overline{sgn}\left(e(p_1(x), p_2(a))\right)$$

   Pozorovací funkci f_2 vytvoříme jako:

   $$f_2(x) = \overline{sgn}\left(e(p_1(a), p_2(x))\right)$$

   Z Příkladu 19 víme, že odčítání racionálních čísel lze realizovat částečně rekurzivní funkcí, platí tedy, že f je pozorovací funkce intervalu $(\zeta(a), \zeta(b))$.

2. Ověříme, že každá pozorovatelná množina je spočetným sjednocením otevřených intervalů.

   Mějme pozorovatelnou množinu O a její pozorovací funkci f. Pro každé $x \in dom(\zeta)$, takové, že $\zeta(x) \in O$ po konečném počtu kroků f skončí a vrátí 0. Víme tedy, že výpočet musí skončit pro každý vstup po přečtení konečného množství prvků z jednotlivých posloupností racionálních čísel. Označme

n_1 počet přečtených cifer posloupnosti $p_1(x)$ a n_2 počet přečtených cifer posloupnosti $p_2(x)$, pak platí:

$$p_1(n_1) \leq \zeta(x) \leq \ p_2(n_2)$$

Tato nerovnost znamená, že pokud pozorovací funkce f vrátí hodnotu 0 na základě $p_1(n_1)$ a $p_2(n_2)$, musí vrátit hodnotu 0 také pro všechna čísla, která náleží uzavřenému intervalu $Z = [p_1(n_1), p_2(n_2)]$. Stejně jako v důkazu u předešlé reprezentace platí, že $Z \subseteq O$. Mějme prvek z množiny Z, f vrátí 0 pro každou reprezentaci daného prvku, zvolíme tedy reprezentaci $z \in dom(\zeta)$ tak, že $\zeta(z)$ nebude krajním bodem. Platí tedy, že $\zeta(z) \neq p_1(n_1)$ a také $\zeta(z) \neq p_2(n_2)$, díky tomu víme, že $\zeta(z) \in (p_1(n_1), p_2(n_2))$, a tedy patří do množiny O spolu s otevřeným intervalem. Jelikož prvků v O je nejvýše spočetně mnoho, množina je sjednocením nejvýše spočetného množství otevřených intervalů.

Pozorovatelné množiny v reprezentaci vnořenými intervaly tedy generují standardní topologii. □

Znamená to tedy, že spojitá zobrazení budou právě ta, která jsou spojitá dle definic z matematické analýzy.

Mějme tedy reálná čísla definovaná posloupnostmi vložených uzavřených intervalů $x = [x_1, x_2], y = [y_1, y_2]$. Základní operace definujeme pomocí intervalové aritmetiky následovně:

$$x + y = [x_1, x_2] + [y_1, y_2] = [x_1 + y_1, x_2 + y_2] \tag{1}$$

$$x - y = [x_1, x_2] - [y_1, y_2] = [x_1 - y_2, x_2 - y_1] \tag{2}$$

$$x \cdot y = [\min\{x_1y_1, x_1y_2, x_2y_1, x_2y_2\}, \max\{x_1y_1, x_1y_2, x_2y_1, x_2y_2\}] \tag{3}$$

$$\frac{x}{y} = [x_1, x_2] \cdot \frac{1}{[y_1, y_2]}, \tag{4}$$

$$\frac{1}{[y_1, y_2]} = \begin{cases} \left[\frac{1}{y_2}, \frac{1}{y_1}\right] & \text{jestliže } 0 \notin [y_1, y_2], \\ \left(-\infty, \frac{1}{y_1}\right] & \text{jestliže } y = [y_1, 0], \\ \left[\frac{1}{y_2}, \infty\right) & \text{jestliže } y = [0, y_2], \\ \left(-\infty, \frac{1}{y_1}\right] \cup \left[\frac{1}{y_2}, \infty\right) & \text{jestliže } 0 \in [y_1, y_2]. \end{cases} \tag{5}$$

Z těchto operací si rozebereme podrobně pouze sčítání. Ukážeme si, že v tomto datovém typu se jedná skutečně o vyčíslitelné zobrazení. Jelikož víme, že sčítání je v přirozené topologii spojité zobrazení, náš předpoklad pro vyčíslitelnost je splněn.

Věta 24

Nechť X označuje datový typ reálných čísel s reprezentací vloženými intervaly a $D = X \times X$ označuje součinový datový typ, pak zobrazení $f : D \to X$ definované jako $f(x) = \pi_1(x) + \pi_2(x)$, je vyčíslitelné.

Důkaz

Označme θ interpretační funkci datového typu D a ζ interpretační funkci datového typu X. Dokážeme, že existuje ČRF g taková, že pro každé $x \in dom(\theta)$ platí $\zeta(g(x)) = f(\theta(x))$. Ověříme nejdříve ale, že s použitím sčítání z intervalové aritmetiky skutečně platí, že funkce g pro $x \in dom(\theta)$ vrátí hodnotu splňující podmínky pro datový typ X, neboli $g(x) \in dom(\zeta)$.

Mějme dvě posloupnosti intervalů $I_n^1 = [a_n, b_n]$ a $I_n^2 = [c_n, d_n]$ splňující podmínky pro datový typ X. Posloupnost intervalů $I_n = [a_n + c_n, b_n + d_n]$ je vnořená a pro každé $\epsilon \in \mathbb{R}$ takové, že $\epsilon > 0$ existuje $j \in \mathbb{N}$ takové, že $|I_j| < \epsilon$. To, že krajní body jednotlivých intervalů výsledné posloupnosti budou racionální čísla je zřejmé. Vnořenost, neboli že $a_{n+1} + c_{n+1} > a_n + c_n$ nebo $b_{n+1} + d_{n+1} < b_n + d_n$ plyne z vnořenosti jednotlivých posloupností intervalů, pro každé $j \in \mathbb{N}$ totiž platí $a_{j+1} > a_j$ nebo $c_{j+1} > c_j$ a $b_{j+1} < b_j$ nebo $d_{j+1} < d_j$.

Jelikož dle předpokladu $b_n - a_n$ i $d_n - c_n$ konvergují k nule, jejich součet také konverguje k nule. Potom pro každé $\epsilon > 0$ existuje přirozené číslo N_1 takové, že pro všechna $n > N_1$ platí $|b_n - a_n| < \frac{\epsilon}{2}$. Obdobně existuje N_2, takové že pro každé $n > N_2$ a platí $|d_n - c_n| < \frac{\epsilon}{2}$. Označme $N = \max(N_1, N_2)$, pro každé $n > N$ platí:

$$|(b_n - a_n) + (d_n - c_n)| < \frac{\epsilon}{2} + \frac{\epsilon}{2} = \epsilon$$

Tedy pro všechna $n > N$ platí $|I_n| < \epsilon$ a pro každé $\epsilon > 0$ existuje $|I_n| < \epsilon$.

Díky těmto důkazům víme, že $A = \bigcap_{i=1}^{n=\infty} [a_i, b_i] = [inf(a_n), sup(b_n)]$, takže také $B = [inf(c_n), sup(d_n)]$. Platí tedy $A + B = [inf(a_n) + inf(c_n), sup(b_n) + sup(d_n)]$ což můžeme zapsat jako $A + B = \bigcap_{i=1}^{n=\infty} [a_i + c_i, b_i + d_i]$.

Hledanou funkci g tedy vytvoříme jako:

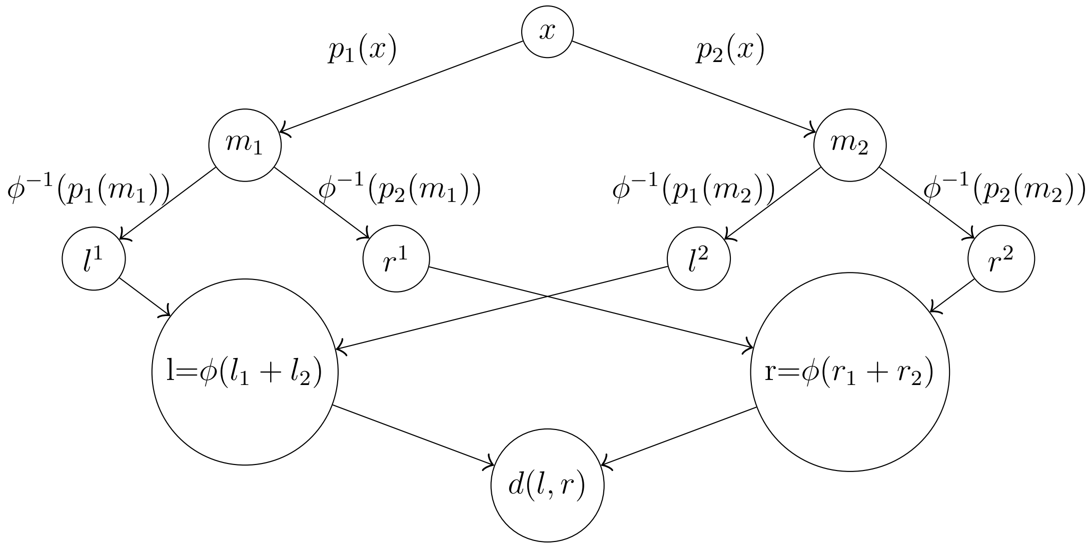
$$g(x) = d(\phi(\phi^{-1}(p_1(p_1(x))) + \phi^{-1}(p_2(p_1(x)))), \phi(\phi^{-1}(p_1(p_2(x))) + \phi^{-1}(p_2(p_2(x)))))$$

Jelikož sčítání i ϕ^{-1} jsou částečně rekurzivní funkce, pak i g je částečně rekurzivní funkce.

Pro lepší vizualizaci označme $p_1(x) = m_1, p_2(x) = m_2$ jednotlivé částečně rekurzivní funkce reprezentující posloupnosti racionálních čísel pro m_1 označíme $l^1 = \phi^{-1}(p_1(m_1))$ a $r^1 = \phi^{-1}(p_2(m_1))$, pro m_2 je označíme $l^2 = \phi^{-1}(p_1(m_2))$ a $r^2 = \phi^{-1}(p_2(m_2))$, kde ϕ^{-1} představuje dekódování částečně rekurzivní funkce z Gödelova čísla. Rozklad na jednotlivé posloupnosti racionálních čísel součinového datového typu je vyobrazen na Obrázku 7.

□

Nyní využijeme Důsledku 14 a ukážeme si příklad zobrazení, které není vyčíslitelné, a tedy nelze naprogramovat, protože není spojité.



Obrázek 7: Součet reálných čísel s intervalovu reprezentací.

Příklad 25

Uvažujme X jako datový typ reálných čísel s intervalovou reprezentací a mějme zobrazení $f : X \to X$ definované následovně:

$$f(x) = \begin{cases} 1/x & \text{jestliže } x \neq 0, \\ 0 & \text{pro } x = 0. \end{cases} \tag{6}$$

Abychom ukázali, že zobrazení není spojité, stačí ověřit, že existuje ve standardní topologii otevřená množina V, pro kterou platí, že $f^{-1}(V)$ není otevřená množina.

Zvolme množinu V například jako interval $(-2, 2)$. Interval $(-2, 0)$ se zobrazí na interval $(-\infty, -\frac{1}{2})$ a interval $(0, 2)$ na interval $(\frac{1}{2}, +\infty)$. Hodnota 0 se zobrazí na uzavřený degenerovaný interval $[0]$. Výsledná množina $f^{-1}(V)$ bude sjednocením $(-\infty, -\frac{1}{2}) \cup [0] \cup (\frac{1}{2}, +\infty)$. Jelikož toto sjednocení nelze zapsat formou spočetného sjednocení otevřených intervalů, tak množina reprezentovaná těmito intervaly není ve standardní topologii otevřená. V kontextu datového typu to znamená, že neexistuje pozorovací funkce této množiny.

Pokud bychom tuto funkci chtěli sestrojit, obnášelo by to mimo jiné vytvoření pozorovací funkce pro množinu $\{0\}$, a tedy vyřešení nerozhodnutelného problému rovnosti reálných čísel.

Mezi další důležitá zobrazení, která ve zmíněných datových typech reálných čísel nejsou spojitá, patří *sgn*, pro které je bodem nespojitosti číslo 0, *floor* a *ceil*, která jsou nespojitá v každém celém čísle ze $\mathbb{Z}$. Nevyčíslitelnou funkcí je také *Dirichletova funkce* χ, tato funkce rozhoduje, jestli zadané reálné číslo je racionální, nebo iracionální, není tedy spojitá v žádném svém bodě.

4.3 Reprezentace vyváženou trojkovou soustavou

Když upustíme od požadavku, že nekonečný číselný rozvoj musí mít pouze kladné koeficienty a přidáme možnost záporných koeficientů, dosáhneme reprezentace, ve které operace sčítání lze naprogramovat. O číselné soustavě, která umožňuje i záporné koeficienty se říká, že je *vyvážená*.

Místo klasických binárních hodnot 0 a 1 budeme tedy používat hodnoty z množiny $\{-1, 0, 1\}$. Tato číselná soustava se nazývá trojková, protože množina koeficientů je tvořena třemi hodnotami. Možnost používat třetí, dodatečnou zápornou hodnotu, nám přidá do množiny reprezentací další redundanci, která ale bude pro definici aritmetických operací zásadní.

V kontextu tohoto datového typu budeme uvažovat čísla z uzavřeného intervalu $[-1, 1]$. Reálné číslo pro přehlednost zapíšeme ve tvaru:

$$(0, a_1 a_2 a_3 \ldots)_T = \sum_{k=1}^{+\infty} a_k 2^{-k},$$

kde každá cifra $a_k \in \{-1, 0, 1\}$. Lze ověřit, že tímto způsobem můžeme reprezentovat stejnou množinu reálných čísel jako pomocí klasického nekonečného binárního rozvoje. Například číslo $(\frac{1}{2})_{10}$ má nekonečno různých reprezentací.

$$0,100\ldots = 0,1(-1)11\ldots = 1,10(-1)11\ldots = 0,100(-1)11\ldots$$

Rozeberme si nyní příklad sčítání, který se nám nepodařilo v datovém typu s reprezentací nekonečným binárním rozvojem vyřešit. Sčítání nešlo provést, jelikož možnost nekonečného rozvoje jedniček nám v některých situacích znemožnila rozhodnout v konečném čase o cifře na první pozici.

Mějme čísla $a = (\frac{1}{3})_{10} = (0,01010\ldots)_T$ a $b = (\frac{2}{3})_{10} = (0,10101\ldots)_T$ a výsledné číslo označme x. Může se zdát, že při jejich součtu opět čelíme nerozhodnutelnému problému rovnosti reálných čísel. Ve vyvážené trojkové soustavě ale po přečtení prvních tří cifer můžeme díky možnosti záporné cifry, bez váhání zvolit jedničku jako první cifru. Rozebereme si nyní dva krajní případy, které mohou nastat.

Prvním krajní případ nastane, pokud obě hodnoty jsou nejnižší možné, tedy $a = 0,01(-1)(-1)\ldots$ a hodnota $b = 0,10(-1)(-1)\ldots$. Protože $b = (\frac{1}{2})_{10}$ a $a = (0)_{10}$, tak jsme stále schopni obsáhnout výsledné číslo, jednou z možností je $x = (\frac{1}{2})_{10} = (1,(-1)00\ldots)_T$.

Druhým krajním případem je, že hodnoty a, b jsou nejvyšší možné, tedy $a = 0,011\ldots$ a $b = 0,1011\ldots$. Protože $a = (\frac{1}{2})_{10}$ a $b = (\frac{3}{4})_{10}$, tak jsme stále schopni obsáhnout výsledné číslo, jednou z možností je $x = (\frac{5}{4})_{10} = (1,0100\ldots)_T$.

Samotný algoritmus pro operaci sčítání i násobení je komplexní, takže ho uvádět nebudeme. V literatuře se objevují nejrůznější reprezentace reálných čísel, například reprezentace konečným rozvojem, přičemž základ je tvořen iracionálním číslem [17]. Další známý způsob reprezentace reálných čísel je pomocí řetězových zlomků.

Závěr

Cílem této práce bylo zformulovat hlavní prvky topologické teorie datových typů úzce propojené s teorií vyčíslitelnosti a rozebrat základní vlastnosti reprezentací reálných čísel. Výsledky této práce jsou postaveny na tom, že jednotlivé způsoby reprezentace ovlivňují, které operace jsou naprogramovatelné, a tudíž z pohledu vyčíslitelnosti vytvářejí odlišné datové typy.

Jedním z hlavních poznatků, který jsme při formulaci této teorie zjistili, je to, že pozorovatelné množiny v datovém typu nemusejí tvořit topologii, ale pouze její bázi. V existující literatuře [10] tato skutečnost není zohledněna, a jedná se tedy o upřesnění dosavadních výsledků topologické teorie datových typů.

Dalším důležitým dokázaným tvrzením je, že zobrazení, která v topologii generované pozorovatelnými množinami nejsou spojitá, nemohou být vyčíslitelná. V kontextu konečných datových typů jsme naopak dokázali, že veškeré operace jsou vyčíslitelné, a lze je tedy naprogramovat.

U studovaných datových typů reálných čísel jsme ukázali, že pro reprezentaci nekonečným binárním rozvojem sčítání nelze definovat. Poté jsme se zaměřili na reprezentaci vnořenými intervaly a ukázali jsme, že všechna zobrazení spojitá dle definic z matematické analýzy jsou spojitá i v této reprezentaci. Následně jsme pro tuto reprezentaci dokázali vyčíslitelnost operace sčítání.

Na závěr jsme si představili zobrazení, která v těchto datových typech na základě nespojitosti naprogramovat nelze. Přehledově jsme si představili reprezentaci vyváženou trojkovou soustavou, která je rozšíření reprezentace nekonečným binárním rozvojem a umožňuje definovat sčítání.

Pro úplnost práce by bylo vhodné doplnit konstruktivní důkazy ověřující, že inverzní funkce Gödelova číslování ϕ a projekce p_1, p_2 Cantorovy párovací funkce d jsou částečně rekurzivní funkce. Mezi kroky vedoucí k rozšíření práce by patřilo například prozkoumání existence datového typu reálných čísel, který umožňuje definovat základní aritmetické operace, přičemž topologie generovaná jeho pozorovatelnými množinami není standardní.

Conclusions

The goal of this thesis was to formulate the main elements of the topological theory of data types closely linked to the theory of computability and to analyze the basic properties of the real numbers representations. The results of this work are based on the fact that different methods of representation affect which operations are programmable, and thus create different data types from the viewpoint of computability.

One of the main insights we have found in formulating this theory is that the observable sets in a data type does not need to form a topology, but only its base. The existing literature [10] does not take this fact into account, and is therefore a clarification of existing results in topological theory of data types.

Another important proved statement is that functions which are not continuous in the topology generated by observable sets cannot be computable. On the contrary, we proved that in the context of finite data types, all operations are computable, and therefore can be programmed.

Within the studied data types of real numbers, we have shown that for the representation by infinite binary expansion addition cannot be defined. We then focused on the representation by nested intervals, and showed that all representations continuous under definitions from mathematical analysis are also continuous in this representation. We then proved the computability of the addition operation for this representation.

In the end, we have introduced functions that cannot be programmed in these data types based on discontinuity. We overviewed the balanced triple system representation, which is an extension of the representation by infinite binary expansion and allows to define addition.

For completeness of the thesis, it would be appropriate to add constructive proofs verifying that the inverse function of the Gödel numbering ϕ and the projections p_1, p_2 of the Cantor pairing function d are partially recursive functions. Steps leading to an extension of the work would include, for example, investigating the existence of a real number data type that allows to define basic arithmetic operations while the topology generated by its observable sets is not standard.

A Obsah elektronických dat

`text/`

Adresář s textem práce ve formátu PDF, vytvořený s použitím závazného stylu KI PřF UP v Olomouci pro závěrečné práce, včetně všech příloh, a všechny soubory potřebné pro bezproblémové vygenerování PDF dokumentu textu (v ZIP archivu), tj. zdrojový text textu, vložené obrázky, a podobně.

Literatura

[1] Rob Shields . Cultural topology: The seven bridges of Königsburg, 1736. *Theory, Culture & Society*, 29(4-5):43–57, 2012. Dostupné z: https://doi.org/10.1177/0263276412451161.

[2] Michal Krupka. Topologie pro informatiky: Poznámky k přednášce, 2022.

[3] James R. Munkres. *Topology*. Featured Titles for Topology. Prentice Hall, Incorporated, 2000. ISBN 978-0-13-181629-9.

[4] George S Boolos, John P Burgess, and Richard C Jeffrey. *Computability and logic*. Cambridge university press, 2002. Dostupné z: https://doi.org/10.1017/CBO9781139164931.

[5] William Gasarch. Primitive Rec, Ackermans Function, Decidable, Undeciable, and Beyond Exposition, 2014. Dostupné z: https://www.cs.umd.edu/users/gasarch/COURSES/858/S20/notes/dec.pdf.

[6] Mario Carneiro. Formalizing computability theory via partial recursive functions, 2019. Dostupné z: https://doi.org/10.48550/arXiv.1810.08380.

[7] Cristian Calude, Solomon Marcus, and Ionel Tevy. The first example of a recursive function which is not primitive recursive. *Historia Mathematica*, 6(4):380–384, 1979. Dostupné z: https://doi.org/10.1016/0315-0860(79)90024-7.

[8] Meri Lisi. Some remarks on the cantor pairing function. *Le Matematiche*, 62(1):55–65, 2007. Dostupné z: https://lematematiche.dmi.unict.it/index.php/lematematiche/article/download/14/13.

[9] Kurt Gödel. Über formal unentscheidbare sätze der principia mathematica und verwandter systeme i. *Monatshefte für mathematik und physik*, 38:173–198, 1931. Dostupné z: https://doi.org/10.1007/BF01700692.

[10] Martín Escardó. Synthetic topology: of Data Types and Classical Spaces. *Electronic Notes in Theoretical Computer Science*, 87:40–41, 2004. Dostupné z: https://doi.org/10.1016/j.entcs.2004.09.017.

[11] Walter Rudin et al. *Principles of mathematical analysis*, volume 3. McGraw-hill New York, 1976. ISBN 9780070856134.

[12] Alan Mathison Turing et al. On computable numbers, with an application to the Entscheidungsproblem. Proceedings of the London Mathematical Society 42 (2):230-265, 1937. Dostupné z: https://doi.org/10.1112/plms/s2-42.1.230.

[13] Georg Cantor. On a Property of the Class of all Real Algebraic Numbers. *Crelle's Journal for Mathematics*, 77(1874):258–262, 1874. Dostupné z: https://srjcstaff.santarosa.edu/~jomartin/IrratFiles/Cantors1874Paper.pdf.

[14] Aberth Oliver. Iteration methods for finding all zeros of a polynomial simultaneously. *Mathematics of computation*, 27(122):339–344, 1973. Dostupné z: https://api.semanticscholar.org/CorpusID:120655175.

[15] Gregory J Chaitin. A theory of program size formally identical to information theory. *Journal of the Association for Computing Machinery*, 22(3):329–340, 1975. Dostupné z: https://dl.acm.org/doi/pdf/10.1145/321892.321894.

[16] Chaitin Gregory J Open Problems in Communication and Computation. *Springer*, 108–112, 1987. ISBN 978-1-4612-4808-8. Dostupné z: https://doi.org/10.1007/978-1-4612-4808-8.

[17] Di Gianantonio Pietro. A golden ratio notation for the real numbers. *Centrum voor Wiskunde en Informatic*, 1996. Dostupné z: https://www.researchgate.net/publication/2254312_A_Golden_Ratio_Notation_for_the_Real_Numbers.