# Samoopravné kódy

*Jiří Koula*

## Co to je a k čemu je to dobré?

Představte si, že chcete někomu poslat určitou informaci, ale máte k dispozici nespolehlivý kanál v tom smyslu, že když posíláte např. nuly a jedničky, občas se po cestě z nuly stane jednička a naopak. Příjemce vaší zprávy tedy nemusí obdržet přesně to, co jste odeslali, ale obdrží pravděpodobně něco velmi podobného (protože chyb nenastane mnoho). To může být ovšem občas fatální – pošlete-li třeba někomu zprávu „Sejdeme se v 14:00“ a dojde mu „Sejdeme se v 18:00“, potom je přijatá zpráva téměř stejná jako odeslaná, ovšem je vidět, že nás to nijak neuspokojí.

Jak tedy posílat zprávy po nespolehlivém kanálu? Odpovědí jsou samoopravné kódy – nepošleme samotnou zprávu, ale něco, co bude delší, bude mít v sobě původní zprávu nějak zakódovánu a v případě, že nenastane příliš mnoho chyb, budeme schopni odeslanou zprávu rekonstruovat ze zprávy přijaté, případně aspoň zjistit, že při přenosu došlo k chybě.

## A co praxe, kde se to dá použít?

Pokud vám předešlé řádky přijdou trošku odtržené od reality (kde najít nespolehlivý kanál?), můžeme si uvést pár příkladů. Jedním z hlavních vlivů pro rozvoj teorie samoopravných kódů bylo zkoumání vesmíru. Totiž když např. dorazí vesmírná sonda na Mars a začne posílat údaje na Zemi, jsme v situaci, kdy máme nespolehlivý kanál. Kdybychom data nijak nechránili, pak to, co bychom odchytili na Zemi jako vysílání sondy, by bylo značně zkreslené a možná nepoužitelné. Mohli bychom si sice data vyžádat opakovaně (což je myšlenka nejjednoduššího samoopravného kódu), ovšem takové vysílání stojí nemalé peníze a vyplatí se přemýšlet, jak si zajistit spolehlivost za cenu co nejmenšího vysílání nad rámec zprávy.

Jiným příkladem, který vám asi bude bližší, jsou obyčejná CDčka. Máte-li datové CD se svými oblíbenými písničkami, filmy, programy, . . . , pak věřte, že za to, že tato data můžete i po několika měsících používat, vděčíte právě samoopravným kódům. Takové CDčko, i když je v klidu v obalu, totiž podléhá působení okolí a po pár měsících poklidného ležení obsahuje desítky až stovky tisíc špatných bitů. Kdybychom proti tomu neměli žádnou ochranu, bylo by CD dost špatné médium pro archivaci. Naštěstí máme samoopravné kódy, které poskytují záchrannou síť.

## A co takhle být trochu konkrétnější?

Obecného povídání už bylo dost, podívejme se proto na konkrétní příklady samoopravných kódů. Aby se ovšem nejednalo o takové povídání o čemsi, udělejme si trochu pořádek, zavedeme si několik pojmů.

**Definice.** *Abecedou $\Sigma$ rozumějme konečnou množinu symbolů $\{s_0, \ldots, s_m\}$. Slovo délky $l$ nad abecedou $\Sigma$ bude uspořádaná $l$-tice symbolů ze $\Sigma$. Množinu všech slov délky $l$ označíme $\Sigma^l$.*

**Definice.** *Od této chvíle dál bude naší abecedou množina $\{0, 1\}$. Nyní si popíšeme, co je symetrický binární kanál. Je to černá skříňka, na jejímž jednom konci stojí odesílatel, na druhém příjemce. Odesílatel posílá kanálem nuly a jedničky. Každý symbol se při přenosu kanálem s pravděpodobností $p$ změní na symbol opačný (nula na jedničku a naopak), s pravděpobností $1 - p$ se pak nezmění. Přenosy jednotlivých bitů (bit=jeden symbol) jsou na sobě nezávislé.*

**Definice.** *Mějme slova $u$ a $v$ délky $l$ nad abecedou $\Sigma$. Hammingovou vzdáleností slov $u$ a $v$ rozumíme počet pozic, v nichž se tato slova liší (např. slova* 011011 *a* 111000 *mají Hammingovu vzdálenost* 3 *– liší se na pozicích jedna, pět a šest).*

**Definice.** *Uvažujme nyní, že odesílatel bude odesílat slova z množiny $W \subseteq \Sigma^l$. Potom kód $C$ délky $n$ je množina $C \subseteq \Sigma^n$ a bijekce $\pi : W \to C$. Kód $C$ i zobrazení $\pi$ znají obě zúčastněné strany.*

*Chce-li nyní odesílatel poslat slovo $w \in W$, pošle místo něj slovo $c = \pi(w)$. Příjemce obdrží slovo $\overline{c}$. Nyní se snaží zrekonstruovat $c$ z $\overline{c}$, neboť z $c$ už snadno získá $w = \pi^{-1}(c)$ – provádí dekódování. Obvyklé je dekódovat $\overline{c}$ na nejbližší kódové slovo ve smyslu Hammingovy vzdálenosti – pokud je pravděpodobnost, že dojde při přenosu k chybě, vcelku malá, potom přijaté slovo asi nebude příliš daleko od odeslaného.*

**Definice.** *Velikostí kódu myslíme počet prvků množiny $C$. Občas je vhodné uvažovat místo $|C|$ spíše $\log_q |C|$, kde $q$ je velikost abecedy, v našem případě tedy* 2.

*Minimální vzdáleností kódu $C$ rozumíme nejmenší možnou (Hammingovu) vzdálenost dvou různých slov z $C$.*

**Definice.** *O kódu $C$ délky $n$ nad abecedou velikosti $q$ s velikostí $q^k$ a minimální vzdáleností $d$ mluvíme jako o $(n, k, d)$-kódu (je-li to třeba, potom použijeme značení $(n, k, d)_q$-kód).*

**Definice.** *Hustota kódu $C$ je podíl*

$$\alpha(C) = \frac{k}{n}.$$

## A teď už konečně příklady!

Dobře, uvedeme si pár příkladů, přesněji jakési extrémy. Na jednu stranu za kód délky $n$ můžeme vzít celou množinu $\Sigma^n$, jedná se o $(n, n, 1)$-kód.

Opačným extrémem je tzv. opakovací kód délky $n$, který posílá nulu na $n$ nul a jedničku na $n$ jedniček. Tento kód má parametry $(n, 1, n)$.

Vidíme, že uvedené příklady jsou extrémy, co se hodnot $k$ a $d$ týče.

Abychom viděli i trošku zajímavější kód, zmiňme tzv. paritní kód. Ten sestává ze všech slov délky $n$, která obsahují sudý počet jedniček. Na přednášce si dokážeme, že tento kód má parametry $(n, n-1, 2)$, dále si také uvedeme ještě zajímavější kód.

## Singeltonův odhad

Zajímavou otázkou je, když máme dáno $n$ a $d$, jaké může být maximální $k$ – tedy kolik slov může mít binární kód, když je dána jeho délka a minimální vzdálenost. Označme $A(n, d)$ maximum z logaritmů (o základu 2) velikostí všech kódů délky $n$.

**Pozorování.** Pro každé $d \leq n$ je

$$A(n, d) \leq A(n-1, d-1).$$

**Tvrzení.** Pro každé sudé $d \leq n$ je

$$A(n, d) = A(n-1, d-1).$$

**Věta.** (Singeltonův odhad) *Pro každé $d \leq n$ je*

$$A(n, d) \leq n - d + 1.$$