Cyclic Redundancy Checks and Error Detection

Bc. Michal Šedý
Waylon Jepsen
Systémy odolné proti poruchám – téma č. 5

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

- Motivace
- Separovatelný CRC
- Galoisovo pole GF_2
- Konstrukce kódového slova
- Vlastností generátorů
- Typy chyb
- Experimentální data
- Výsledky experimentů

- Je potřeba detekce poškozených dat.
- Poškozená data vedou na chybné výsledky (genetická data, ML).
- Mimo blokových a náhodných chyb, se v praxi objevují také výpadky nebo vložení části dat.
- Výpadky a vložení části dat se CRC hůře detekují.
- Je potřebné znát úspěšnost detekce chyb pro jednotlivé polynomy.
- Cílem této práce bylo vyhodnotit úspěšnost detekce chyb na reálných datech pro různé polynomy 16. stupně.

Separovatelný CRC (Cyclic Redundancy Check)

- CRC kód (**n**, **k**)
  - **n** – délka celého kódového slova
  - **k** – délka kódové informace
  - vytvořen generujícím polynomem $g(x)$ stupně $n - k$
- Úspěšnost detekce chyb závisí na zvoleném polynomu $g(x)$.
- $b_i b_{i-1} \dots b_1 b_0$ je ekvivalentní polynomu: $b_i x^i + b_{i-1} x^{i-1} + \cdots + b_1 x + b_0$
- Kódové slovo **c** se skládá ze dvou částí **c = [m, r]**
  - **m** – původní zpráva s ekvivalentním polynomem $M(x)$.
  - **r** – zbytek po dělení $\frac{M(x) \cdot x^{n-k}}{g(x)}$
  - **c** – musí být dělitelné bezezbytku polynomem $g(x)$ (jinak obsahuje chybu)
- Pro operace je použito Galoisovo pole GF_2.

Galoisovo pole GF_2

- Obsahuje dva elementy {0,1}.
- Pro GF_2 platí: $\forall n \in \mathbb{N}: n \,\%\, 2 \in GF_2$, kde % je operace modulo.
- Dělení polynomů lze provést pomocí posuvných registrů a hradel XOR.
- Generující polynomy, které jsou primitivními dosahují nejlepších detekčních výsledků.

Příklad separovatelného CRC (9,6)

- Mějme data d = **100100** s ekvivalentním polynomem $\mathrm{D}(x)$.
- Generující polynom $g(x) = x^3 + x^2 + 1$
- $g(x)$ je ekvivalentní zápisu **1101**
- $\mathrm{D}(x) \cdot x^3 \sim 100100000$
- *Násobení polynomu x^3 je ekvivalentní násobení 2^3 (posuv).*
- Zbytek po dělení $\frac{10010000}{1101}$ je 001.
- Výsledné zpráva M je 100100 001.

$$\underline{100100}000 = \mathbf{d} \cdot 2^3$$

```
 10000000
  1010000
   111000
    01100
     1100
      001 = r
```

$$\mathbf{c} = \underline{100100}\ 001$$

- Pokud není polynom zakódované zprávy $\mathrm{M}(x)$ dělitelný generujícím polynomem $g(x)$ bezezbytku, obsahuje zpráva chybu.
- Je nutné minimalizovat případy, kdy je chybný polynom $M_E(x)$ dělitelný bezezbytku (chyba se nepozná).
- Vhodně navržený generující polynom je základem.

- **Jednoduchá chyba**
  - Stačí, aby měl $g(x)$ dva nenulové členy.
- **Lichý počet chyb**
  - $g(x)$ musí mít sudý počet členů.
  - Volí se například $(x+1)g'(x)$, kde $g'(x)$ je ireducibilní polynom s lichým počtem členů.

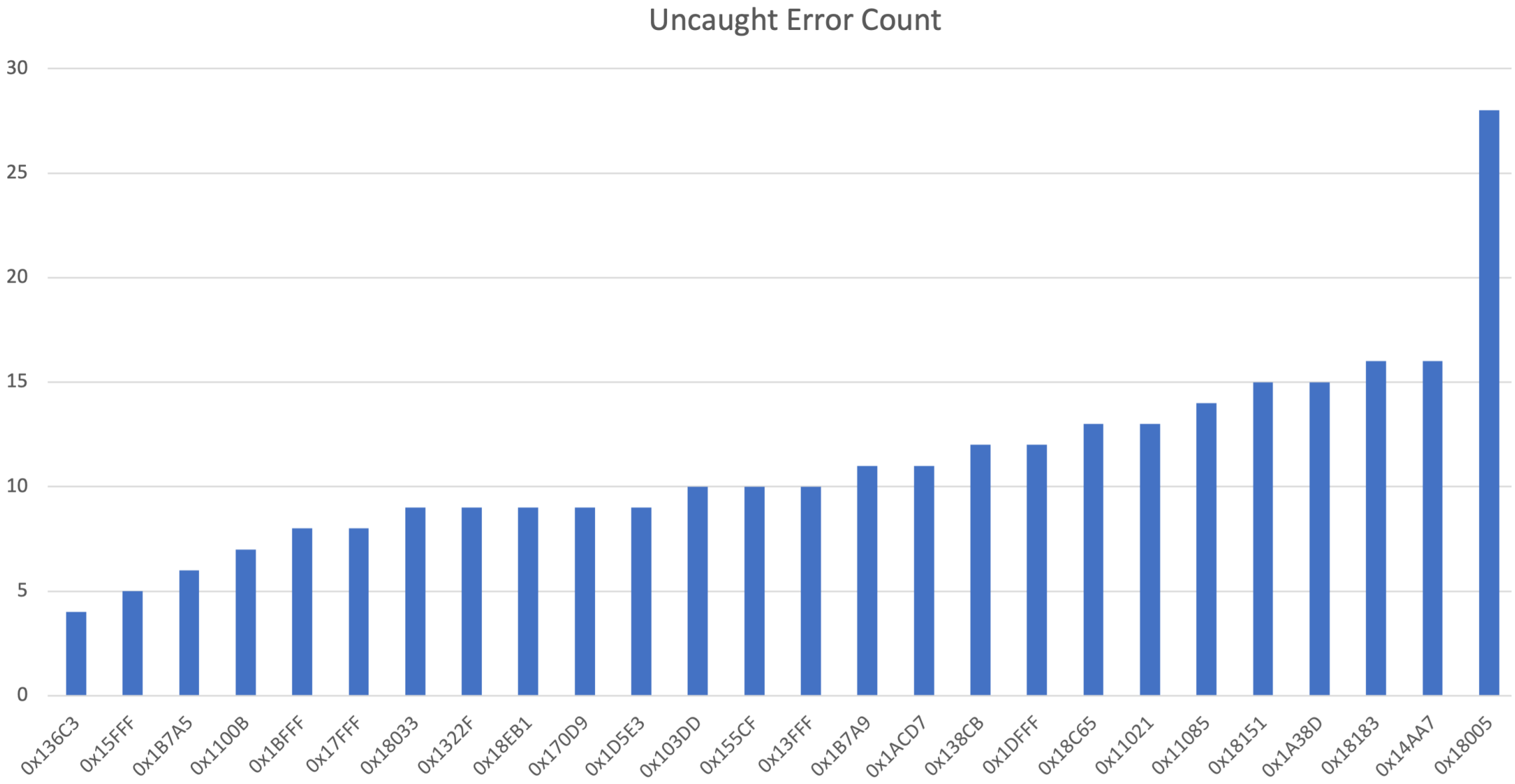
- **Dvojchyba**
  - Pro každé $g(x)$ existuje $N \in \mathbb{N}$ takové, že $g(x)$ dělí $(x^N + 1)$ bezezbytku.
  - Vždy existuje nedetekovatelná dvojchyba.
  - Délka zprávy M musí být menší než $2n - k - 1$.
  - Pro CRC (n, k) volíme $g(x)$ tak, aby bylo $N = 2^{n-k} - 1$ dostatečně velké.
  - Délka polynomu zprávy $M(x)$ musí být menší, než $2^{n-k} - 1$.

- Jedná se o shluk délky *b*, který obsahuje chybné hodnoty.
  - $M(x) = 00001010000$
  - $M_E(x) = 000\underline{10011}000$

- Pro všechny ireducibilní generující polynomy stupně **p** platí:
  - Detekuje všechny shlukové chyby do délky **p**.
  - Detekuje $1 - 0{,}5^p$ všech shlukových chyb délky více než $p + 1$.
  - Pravděpodobnost nedekování shlukové chyby větší délky než $n - k$ je $2^{-(n-k)}$.

- CRC (65552, 65536)
- 727 552 poškozených paketů
- 27 různých generujících polynomů stupně 16
  - **Primitive in GF2**
    - 5 primitivních polynomů stupně 16
  - **Primitive 15 in GF2(x+1)**
    - 5 polynomů stupně 15 vynásobených $(x + 1)$
  - **Ireducible GF2 Orbiter**
    - 10 ireducibilních polynomů stupně 16 zvolených nástrojem Orbiter
  - **From Program**
    - 5 nejlepších polynomů pro detekci dvoubitové chyby
    - polynom CCITT využívaný v X.25, V:41, Bluetooth, SD
    - polynom CRC-16 od IBM využívaný v USB, ANSI X3.28, Bisinc, Modbus

Nezachycené chyby podle generátorů



- 0x136C3
  - Primitivní polynom stupně 16.
  - Pouze 4x neodhalil chybu.
  - Očekávalo se, že bude dávat dobré výsledky.

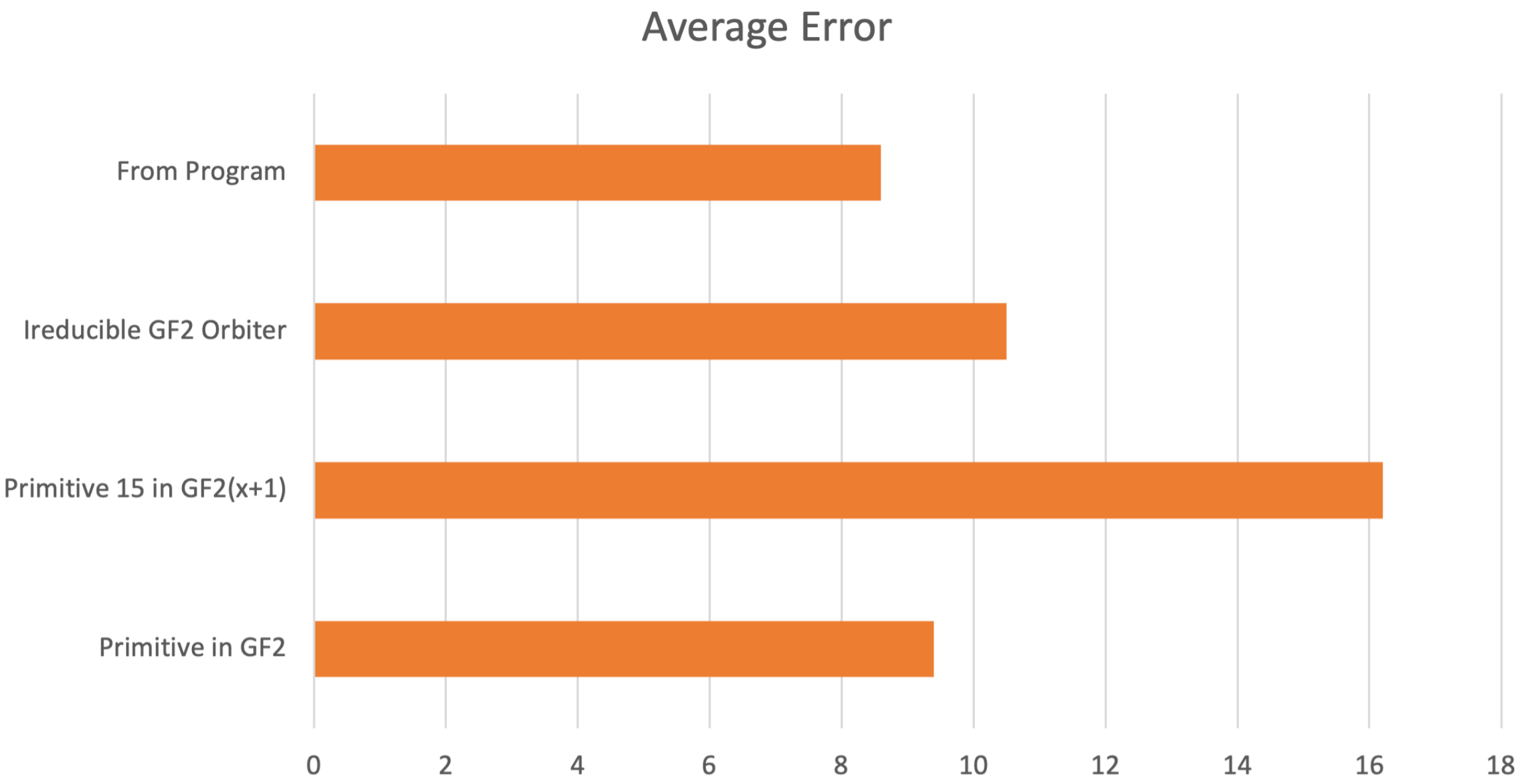
- 0x15FFF
  - Polynom stupně 16 vybraný z nejlepších pro detekci dvoubitových chyb.
  - Pouze 5-krát neodhalil chybu.
  - Není ireducibilní.
  - Neočekávaly se tak dobré výsledky.

Nejhorší generátory

- 0x18005
  - Polynom CRC-16 používaný IBM v mnoha protokolech (zarážející).
  - Nejhorší testovaný generátor.
  - Neodhalil 28 chybných paketů.

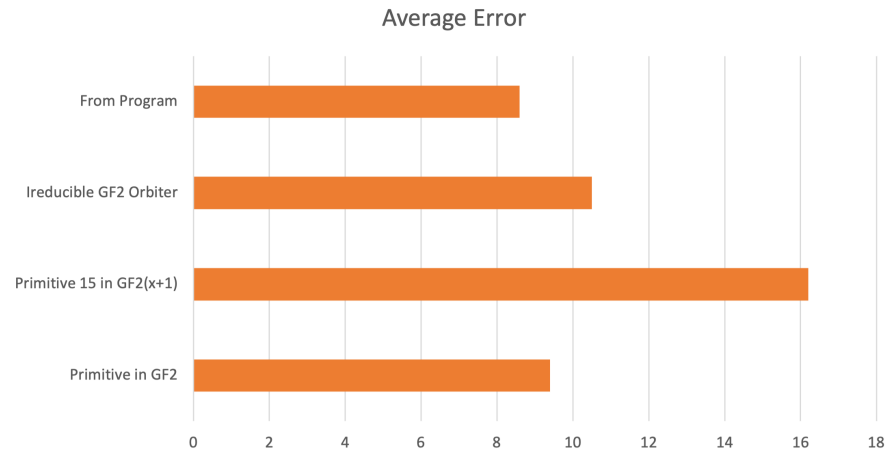
- 0x14AA7
  - Ireducibilní polynom stupně 16 vybraný nástrojem Orbiter.
  - Neodhalil 16 chybných paketů.
  - Ne všechny ireducibilní polynomy reagují dobře na náhodné shlukové chyby.

Průměrná chyba skupiny generátorů



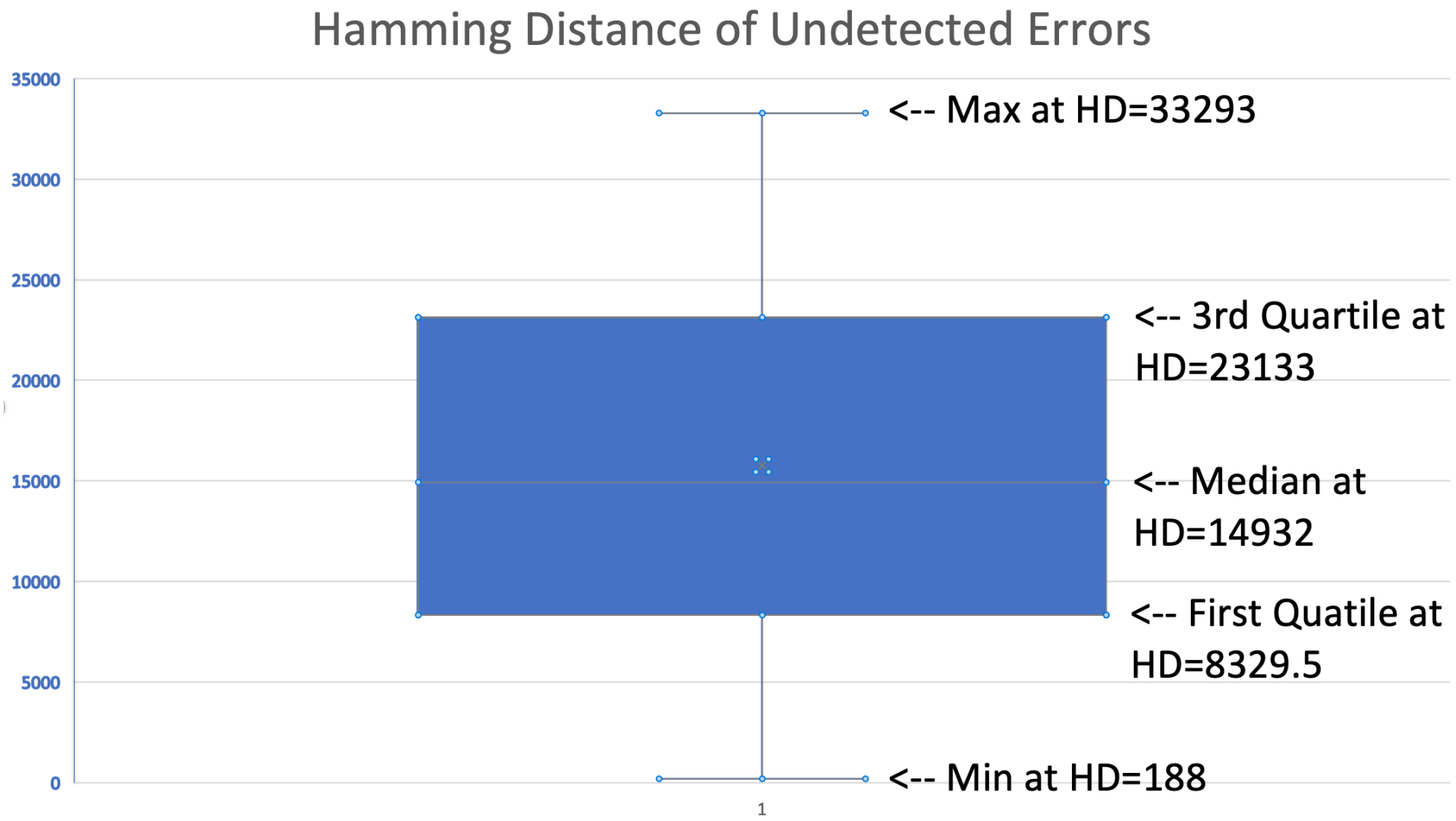
Průměrná chyba skupiny generátorů – komentář

- Nejhorších výsledků dosahovaly generující polynomy stupně 15 vynásobeny (x+1).
  - Toto není překvapující, protože žádný z nich nebyl ireducibilní.



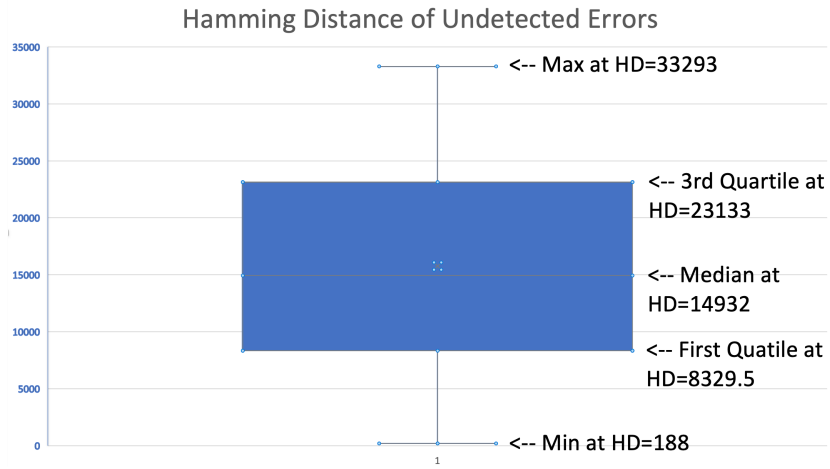
- Průměrně dosáhly nejlepších výsledků polynomy ze skupiny „From Program“, které současně obsahují polynomy nejlépe detekujícími dvoubitové chyby.
  - Tento výsledek byl nečekaný, protože v této skupině nejsou pouze primitivní polynomy.

Hammingová vzdálenost chyb



Hammingová vzdálenost chyb – komentář

- Na základě výsledku Koopmana[1] můžeme zhodnotit hammingovou vzdálenost (HD) nedetekovaných chyb.
- Koopman uvedl tabulku generujících polynomů různých řádů a jejich HD.
- Nejsou uvedeny polynomy pro větší HD, než 19.



- Z grafu lze vidět, že HD reálně se vyskytujících chyb je mnohem větší, než se kterou počítá Koopman.
- Většina nedetekovaných chyb měla HD > 8329,5.

1: http://users.ece.cmu.edu/~koopman/crc/index.html

Budoucí práce

- Optimalizace programu na vyhledávání generujících polynomů.
- Vyzkoušet generátory řádu 32 a 64.
- Prozkoumat rozdílné chybové vzory (mohou se lišit pro různé přenosové technologie).
- Otestovat BCH kódy, opravné kódy založené na CRC

- Jepsen, W. (2022). Cyclic Redundancy Checks and Error Detection. arXiv. https://doi.org/10.48550/ARXIV.2205.11344
- Koopman, P. http://users.ece.cmu.edu/~koopman/crc/index.html
- Drábek, V., Bidlo, M. (2023). Systémy odolné proti poruchám: SSP 6. Cyklické kódy
- https://github.com/0xJepsen/CRC_Research/tree/master/crclists

Děkuji za pozornost