# Obsah

## Poděkování

Tímto bych rád poděkoval vedoucímu mé bakalářské práce, Ing. Janu Kacálkovi, za pomoc a veškerý čas, který mi věnoval.

## Prohlášení

Prohlašuji, že svou bakalářskou práci na téma „Možnosti výpočtu vzájemné informace z časové řady“ jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této bakalářské práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

V Brně dne 01.06.2010 .....

podpis autora

## Anotace

Vzájemná informace je jedním z faktorů, využívaných při analýze síťového provozu a sestavení fázového prostoru. V úvodu práce se zabývám teorií informace se zaměřením na teoretický výpočet vzájemné informace. K výpočtu tohoto parametru je k dispozici již řada algoritmů, které ve své závěrečné práci podrobně rozebírám. Dva z algoritmů (Fraser-Swinneyho a výpočet vzájemné informace pomocí adaptivního XY dělení) jsou aplikovány na vstupní data Rösslerova atraktoru, jak je znázorněno výstupními tabulkami a grafy. Třetí uvažovanou výpočetní metodou je Dinh-Tuan-Phamův algoritmus. Hlavním cílem mé práce tedy je srovnání efektivity, rychlost výpočtu a přesnost zmíněných algoritmů.

### Abstrakt - anglicky

Mutual information is one of the factors used in traffic analysis and preparation phase space. At the beginning of this work I deal with information theory, focusing on the theoretical calculation of mutual information. To calculate this parameter has been available for many algorithms which I analyze in my final work. Two of the algorithms (Fraser-Swinney and calculation of mutual information using adaptive XY subdivision) are applied to the input data Rössler' attractor, as shown in the output tables and graphs. The third consideration method is the computational Dinh-Tuan-Pham algorithm. The main goal of my work is a comparison of efficiency, speed and accuracy of the calculation of these algorithms.

# Úvod

Ve své bakalářské práci se podrobně zabývám způsobem výpočtu vzájemné informace. Nejprve se ovšem stručně zmíním o charakteristice pojmu vzájemné informace v několika bodech bez potřeby její matematické definice. Teorii informace je ovšem zapotřebí nejprve pochopit, proto se jí zaobírám v následující kapitole a dále odkáži k literárnímu prameni [1], ze které jsem ostatně sám čerpal cenné poznatky.

Charakteristika pojmu „vzájemná informace“ není vždy tak zcela jednoznačná, jelikož různé zdroje ji pojímají z rozdílných úhlů pohledu.

**Vzájemná informace je definována jako:**

- míra skutečně přeneseného množství informace od generátoru zprávy až k příjemci.
- závislost mezi dvěma náhodnými veličinami. Zde platí přímá úměrnost, tedy čím vyšší je hodnota vzájemné informace, tím větší je závislost mezi dvěma náhodnými veličinami.

Množství vzájemné informace je udáváno v binárních jednotkách, tedy v bitech. V praktickém využití je vzájemná informace důležitým parametrem například pro rekonstrukci fázového prostoru. To je jeden z požadavků metod z nelineární analýzy, kdy je vyžadováno, aby data byla zobrazována jako body v dimenzionálním fázovém prostoru. Po vložení dat z chaotické časové řady do fázového prostoru je pak možné určit chaotický atraktor.[1]

K výpočtu vzájemné informace byla vyvinuta již řada vhodných algoritmů a z těchto jsem vybral právě následující tři výpočetní postupy.

**Vybrané algoritmy výpočtu vzájemné informace:**

- Fraser-Swinneyho algoritmus (angl. Fraser-Swinney algorithm)
- Výpočet vzájemné informace pomocí adaptivního XY dělení
- Dinh-Tuan-Phamův algoritmus

V této práci se zaměřuji na aplikaci prvních dvou ze zmíněných algoritmů na vstupní data, generovaná rovnicí Rösslerova atraktoru, s cílem porovnávání jejich rychlosti,

[1] Citace převzata z: Jan Kacálek, Ivan Míča, *Nelineární analýza a predikce síťového provozu.* VUT v Brně: Elektrorevue 2009, s.2.

efektivity (přesnosti) a výpočetní náročnosti. Třetí algoritmus je rozebírán především pro srovnání teoretické, jelikož jeho implementace je poměrně náročná.

# 1. Úvod do problematiky

Pro kvalitativnější porozumění „teorii vzájemné informace“ je zapotřebí nejprve pochopit základy, které se vůbec k teorii informace váží. V této kapitole si ujasníme, jakého významu má pro nás informace nabývat a vysvětlíme si její matematické parametry.

## 1.1. Teorie informace[1]

**Charakteristika pojmů:**

Zpráva - jakákoliv posloupnost rozlišitelných znaků

Symboly - rozlišitelné prvky ve zprávě, v grafickém znázornění jde o znaky

Abeceda - množina všech symbolů, případně znaků

Příklad.:

e b c e a b d a b e d c b a c - zpráva

D = 15 - délka zprávy

A={a,b,c,d,e} - abeceda

S=4 - počet symbolů abecedy

Signál - materiální nositel zprávy

Kódování - transformace zprávy vyjádřené symboly jedné abecedy na zprávu vyjádřenou symboly druhé abecedy

Informace - strukturní vztahy mezi symboly

- vztahy mezi symboly zprávy a okolním světem - omezené na vztahy:
  - = mezi označením a významem
  - = mezi významem a jejich překladem
- bývá dělena do tří odvětví
  - = syntaktická neboli skladební

= sémantická (sémantika – nauka o významu slov)

= pragmatická neboli dbá na příčinnou souvislost

### 1.2. Vlastnosti informace

Signál je tvořen posloupností n kódových slov o celkové délce $n_I$ informačních prvků. Nechť tedy syntaktická abeceda obsahuje N rozlišitelných informačních prvků. Nazveme-li celkový počet všech povolených kódových slov $N_K$, pak celkový počet $N_Z$ všech zpráv, které je možné signálem vyjádřit, je dán vztahem 1.

$$N_Z = N_K^n \quad (1)$$

V tomto případě tedy mluvíme o jakési formě informační kapacity daného signálu. Obdobou tohoto zápisu je také Informační kapacita soustavy, kterou se v roce 1928 zabýval muž jménem Hartley. Jeho studie zahrnují do pojmu „soustava“ vše z množiny diskrétních stavů, kromě signálů v informačním pojetí tedy i sdělovací kanály a zprávy. Informační kapacita soustavy je tedy dle něj dána následujícím vztahem 2.

$$C = \log_2 N_S \quad (2)$$

Zde $N_S$ je počet všech možných stavů soustavy, přičemž původní jednotkou informační kapacity byl „bit“, nyní je jí Shannon [Sh – čti Šenon]. Uvážíme-li dvě vzájemně oddělené soustavy o informačních kapacitách $C_1$ a $C_2$, pak jejich sloučením vznikde soustava o informační kapacitě rovné jejich součtu, tedy vztah 3.

$$C = \log_2 (N_{S1} N_{S2}) = \log_2 N_{S1} + \log_2 N_{S2} = C_1 + C_2 \quad (3)$$

Informační kapacita sama o sobě může sice kvalifikovat množství informace dané soustavy, pomíjí ovšem ale pohled příjemce informace z hlediska důležitosti. Tento „parametr“ nebo také „kvalitu“ informace není možné popsat vzorcem, proto užíváme aparát výpočtu pravděpodobnosti a náhodných jevů.

### 1.2.1. Kvalitativní vyjádření množství informace na základě náhodných jevů

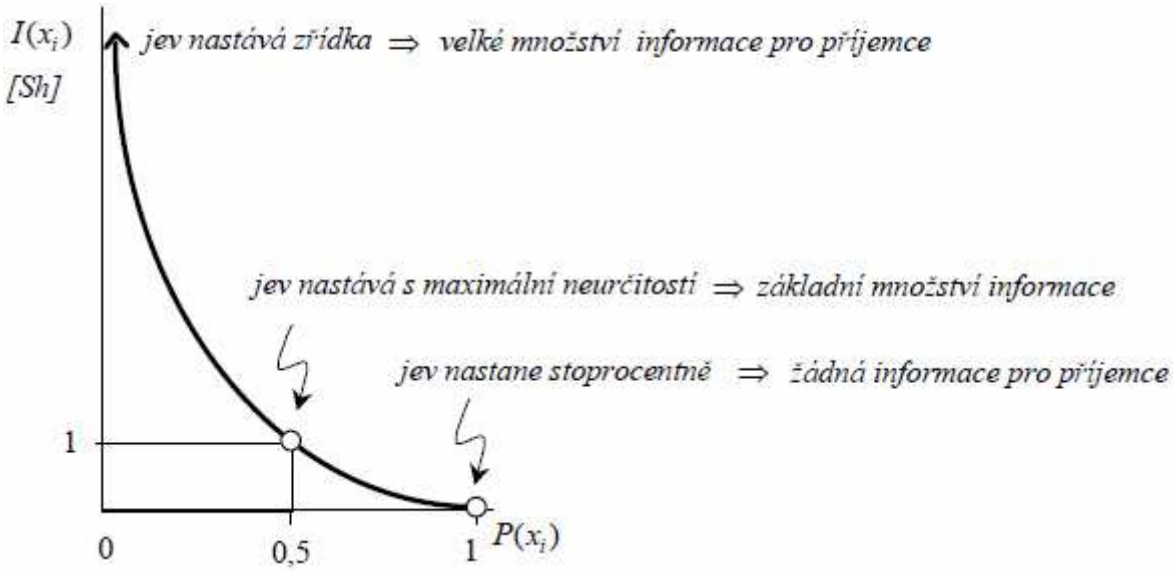
Je obecně známo, že fyzikální jevy sledujeme pomocí signálů, které jsou jimi generovány. Pokud ovšem nejsme schopni předem určit hodnoty těchto signálů v daných časových okamžicích, říkáme, že jde o jevy náhodné. Tyto jevy se vzájemně vylučují, jelikož v daný časový okamžik může být ze souboru náhodných jevů platný pouze jeden. Adekvátním příkladem pro toto tvrzení je hod šestistěnnou kostkou, kde pravděpodobnost jevu, kdy padne jakékoli číslo je právě 1/6. Úplný soubor jevů společně s pravděpodobnostmi jejich výskytu se někdy zapisují do tzv. Konečného schématu a pro tyto platí 4.

$$\{X\}=\{x_1,x_2,...,x_N\}\quad\begin{pmatrix}x_1 & x_2 & x_3 & ... & x_i & ... & x_N\\ P(x_1) & P(x_2) & P(x_3) & ... & P(x_i) & ... & P(x_N)\end{pmatrix}\qquad \sum_{i=1}^{N}P(x_i) \quad (4)$$

### 1.2.2. Kvalitativní vyjádření množství informace na základě pravděpodobnosti

Množství informace získané příjemcem po přijetí zprávy, že v daný okamžik došlo k výskytu jevu $x_i$ z úplného souboru vzájemně se vylučujících jevů $\{X\}$, je dáno vztahem

$$I(x_i)=\log_2\frac{1}{P(x_i)}=-\log_2 P(x_i)\qquad [Sh] \quad (5)$$




Obr.1: Grafické znázornění vzorce pro výpočet množství informace získané příjemcem

Z pohledu užitečnosti tedy chápeme vzorec 5 následovně. Pokud nastal jev, který je zákonitě nevyhnutelný, např. že jablko ze stromu vždy spadne dolů, přiřadíme nulovou informační hodnotu, podle Obr.1 tedy pravý krajní bod. Oproti tomu informace o udání jevu, který nastává velmi zřídka ($I \to \infty$ pro $P \to 0$) je pro nás zajímavá. Těmto úvahám zcela vyhovuje vzorec 5, který navíc zajišťuje výše zmiňovanou aditivitu (nalezení hodnoty proměnné sečítáním jejích dílčích hodnot).

Oproti tomuto pohledu můžeme stejně tak výše zmiňovaný vzorec chápat i z pohledu návrháře digitálního přenosu zprávy. Uvažujeme-li například o zakódování přenášené zprávy, úvahu zobecníme na n znaků $a_1$ až $a_n$, kde n je celá mocnina dvou, které lze vyjádřit kódovými slovy o délce $\log_2 n$. K zakódování každého z n znaků tedy potřebujeme $\log_2 n$ bitů. Každému znaku přísluší vždy P hodnota pravděpodobnosti, kterou je nutno zakódovat $\log_2(1/P)$ bity. Přitom musí být množství informace spojené s výskytem tohoto znaku být úměrné výrazu 6.

$$I = k \cdot \log_2 \frac{1}{P} \quad (6)$$

Konstantu k lze odvodit volbou základu logaritmu. Jednotku množství informace odvodíme pomocí Tab.1.

| **Základ logaritmu** | **Jednotka** | **Hodnota v shannonech** |
|---|---|---|
| 2 | 1 Sh (shannon) | 1 Sh |
| 10 | 1 Hartley | 3,32 Sh |
| e | 1 nat | 1,44 Sh |

Tab. 1: Modifikace jednotky množství informace

Jak zjistíme od následujícího příkladu dále, množství informace I je možné definovat nejen pro jednotlivé znaky, ale i pro zprávy složené z těchto znaků.

Vezměme v úvahu například jednovýstupový logický člen, na jehož se nezávisle na vstupních hodnotách objevují binární hodnoty, tedy jedničky a nuly

s pravděpodobností P(1)=0,9 a P(0)=0,1. Jaké množství informace získáme přijetím zprávy 1101?

Za předpokladu, že pravděpodobnosti P(1) a P(0) jsou vzájemně nezávislé, vypočteme pravděpodobnost přijetí zprávy podle vztahu 7.

$$P(1)\cdot P(1)\cdot P(0)\cdot P(1) = P^3(1)\cdot P(0) = 0{,}9^3\cdot 0{,}1 = 0{,}0729$$
$$I(1) = -\log_2 0{,}0729 \cong \underline{\underline{0{,}152\,Sh}} \quad (7)$$

Pokud by pravděpodobnosti přijetí zmiňovaných znaků byly stejné, tzn. P(1)=P(0)=0,5, pak by pravděpodobnost přijetí jakékoli čtyřbitové zprávy byla rovna $P^4(1)=0{,}0625$, čemuž odpovídá množství informace $I(1)=-\log_2 0{,}0625=4Sh$.

### 1.2.3. Entropie úplného souboru nahodných jevů

Entropie je zjednodušeně charakterizována jako míra neurčitosti náhodného procesu, v našem případě ovšem půjde o charakter úplného souboru, jako celku. Stejně jako v předešlých případech, uvažujeme i nyní úplný soubor N jevů s konečným schématem 8.

$$\begin{pmatrix} x_1 & x_2 & x_3 & \dots & x_i & \dots & x_N \\ P(x_1) & P(x_2) & P(x_3) & \dots & P(x_i) & \dots & P(x_N) \end{pmatrix} \quad (8)$$

Rozložení pravděposobností ve spodním řádku odpovídá neurčitosti, který z jevů nastane. V případě rovnoměrného rozložení je například tato neurčitost maximální, jelikož každý člen nabývá stejné hodnoty pravděpodobnosti. V důsledku růstu počtu jevů N dále poroste i neurčitost. Dalším úplným souborem jevů může být kupříkladu případ, kdy jedna z pravděpodobností bude jernotková a ostatní tudíž nulové. Takový soubor již ovšem není náhodný, nýbrž deterministický, jelikož bude pravidelně docházet pouze k výskytu jevu s pravděpodobnosti hodnoty 1. Neurčitost příjemce v tomto případě je tedy nulová.

Jak bylo naznačeno již v úvodu této podkapitoly, zmiňovaná neurčitost je nazývána entropií a je poměrně snadno vyčíslitelná. Jelikož příjemce současně získává

informaci o tom, nastane-li daný jev, lze tedy říci, že se entropie úplného souboru jevů číselně rovná množství informace připadající na výskyt jednoho jevu.

Entropie musí jako funkce pochopitelně splňovat i několik požadavků:

a) Být funkcí všech pravděpodobností $P(x_i)$, kdy i=1.N.
b) Musí nabývat při rovnoměrném rozložení pravděpodobností maximální hodnoty a tato hodnota musí růst při rostoucím počtu jevů N.
c) Musí být nulová při deterministickém rozložení, kdy jen jedna z pravděpodobností je rovna jedné.
d) Musí být zachována kompatifility s definicí množství informace, jelikož se entropie rovná množství informace připadající na výskyt jednoho jevu.

### 1.2.3.1. Průměrná entropie

Požadavkům uvedeným ve skupině 1.2.3. Entropie úplného souboru nahodných jevů vyhovuje definice průměrného množství informace z úplného souboru náhodných jevů {X}. Jednotkou takto definované entropie je Sh/jev a za předpokladu, že je jevem výskyt jednoho z možných symbolů zprávy (prvků signálu), pak jednotkou je Sh/symbol (Sh/prvek).

$$H(X) = -\sum_{i=1}^{N} P(x_i)\log_2 P(x_i) \ \ [Sh/\ jev] \quad (9)$$

### 1.2.3.2. Maximální entropie

Maximální entropie při rovnoměrném rozložení pravděpodobnosti je platná pro soubory s N jevy. Pro její výpočet je obecně dán vztah 10.

$$H_{\max} = \log_2 N \quad (10)$$

Srovnáme-li jej se vzorcem 2 docházíme k závěru, že maximální možná entropie úplného souboru jevů je číselně srovnatelná s informační kapacitou podle Hartleye.

Číselná hodnota maximální entropie ovšem zpravidla neodpovídá skutečnosti. Počítá se zde totiž pouze s počtem prvků, nikoli s pravděpodobností jejich výskytu. Stejně tak může být výskyt daného prvku závislý na znaku předešlém a to ať už výpočetně, gramaticky, či na základě nějakého algoritmu.

#### 1.2.3.3. Relativní entropie

Relativní entropie je pojem vyjadřující poměr entropie a její maximání hodnoty.

$$h = \frac{H}{H_{\max}} \in \langle 0,1 \rangle \quad (11)$$

#### 1.2.3.4. Redundance

Na základě relativní entropie je zaváděn pojem redundance, neboli nadbytečnost.

$$r = 1 - h = \frac{H_{\max} - H}{H_{\max}} \in \langle 0,1 \rangle \quad (12)$$

### 1.2.4. Entropie zprávy

Jelikož byl pojem entropie v rámci teorie informace původné zaveden pro úplný soubor vzájemně se vylučujících jevů, je i příkladné hovořit o entropii abecedy. Tímto se myslí entropie souboru jevů spojených s náhodným výskytem jednoho z prvků abecedy na sledované pozici ve zprávě. V tomto případě je entropie průměrným množstvím informace, která je nesena jedním prvkem zprávy. Rozlišujeme zde následující dva případy.

#### 1.2.4.1. Případ výskytů nezávislých

Za předpokladu, že je výskyt prvku zprávy zcela nezávislý na výskytu některého z prvků předcházejících, pak jsou náhodné jevy spojené s výskytem dalších prvků (v oblasti nezávislých výskytů) pokládány za nezávislé. Pokud tedy vynásobíme entropii

$H_{abc}$ abecedy délkou zprávy L, neboli počtem jejích prvků, získáme “průměrné” množství informace nesené celou zprávou $I_{zpr}$ dle vztahu 13.

$$I_{zpr} = L \cdot H_{abc} \quad [Sh] \quad (13)$$

#### 1.2.4.2. Případ výskytů závislých

Pokud je výskyt prvků v daném místě abecedy silně závislý na významu předešlého textu, pak je „průměrné“ množství informace nesené celou zprávou menší než součin její délky a entropie její abecedy.

$$I_{zpr} < L \cdot H_{abc} \quad [Sh] \quad (14)$$

Vzorec 13 lze mimo jiné interpretovat také tak, že informační hodnota zprávy klesla na základě vazeb mezi znaky, jelikož klesla „efektivní“ entropie abecedy oproti případu bez úvahy těchto vazeb.

### 1.2.5. Entropie a redundance zdroje zpráv

V počátku sdělovacího řetězce je obvykle generátor, vysílač nebo jiný zdroj zprávy. Vždy existuje určitá množina možných sdělení spolu s pravděpodobnostmi jejich výskytu, což je společný parametr zpráv bez ohledu na jejich character. Díky tomuto společnému rysu lze využít entropii k ohodnocení informačního obsahu generovaných zpráv. V případě diskrétních zdrojů je přitom entropie zdroje rovna entropii použité abecedy.

Jednotkou entropie zdroje může být Sh/znak a stejně tak přepočtená hodnota v jednotkách Sh/s. Samotný přepočet je realizován při entropii v Sh/znak násobkem počtu generovaných znaků za sekundu (např. modulační rychlost). Z entropie zdroje lze pomocí vzorce 11 vypočíst nadbytečnost, která vyjadřuje jeho “informační rezervu”.

### 1.2.6. Kódování zpráv na rozhraní

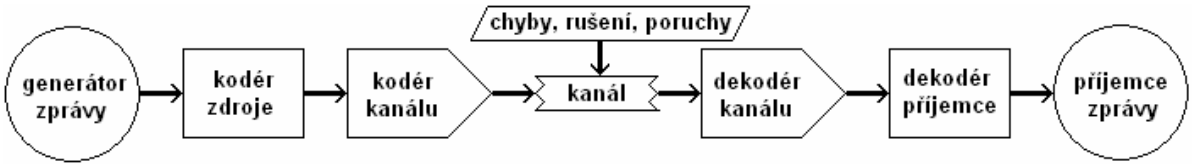
Celkem rozlišujeme dvě rozhraní, přičemž jde konkrétně o rozhraní fyzikální a informační rozhraní. Zatímco v rámci fyzického rozhraní dochází k fyzické změně formy signálu, na rozhraní informačním dochází ke změnám informačního modelu signálu. V příklad fyzikálního rozhraní lze uvést například převod optického signálu na odpovídající elektrické impulsy. Oproti tomu na fyzickém rozhraní je diditální signál překódován za účelem jeho komprese. Nyní jsou pro nás ovšem podstatné děje probíhající na informačním rozhraní.

Kódováním je myšlen proces změny syntaktické abecedy nebo smluvené transformace celých informačních slov. Účel tohoto procesu je změna entropie i redundance abecedy. Lze tak totiž hledat kód s nejvyšší entropií zprávy, neboli s nejmenším počtem znaků na dané množství generované informace. Takový kód je pak nejekonomičtější pro informační přenos, jelikož údělem této procedury je dosáhnout co nejmenšího počtu znaků a nejkratší doby přenosu.

Vezměme například v úvahu příklad, kdy zprávu dvakrát překódujeme dvěma rozdílnými kódy, založenými na binární soustavě. Tímto se postupně zvyšuje entropie zprávy, tedy její neurčitost.

### 1.2.6.1. Shannonovo schéma komunikačního systému

Muž jménem Claude Elwood Shannon (30.4.1916 - 24.2.2001) v 50. letech dokázal fakt, že takřka všechny komunikační systémy užívané od minulosti až do dnešní doby jsou pouze obecné případy komunikačního systému. Tento je definován na Obr. 2.




Obr.2: Shannonovo schéma obecného komunikačního systému

Úděl jednotlivých prvků v koncepci:

**Kodér zdroje** provádí kódování zprávy tak, aby její entropie byla co nejvyšší a redundance minimalizována. Jinak řečeno, aby byl pro přenos zprávy použit co

nejmenší počet znaků. Jeho častým úkolem je převod původního signálu na signál elektrický (případně do digitální podoby.

**Kodér kanálu** zabezpečuje spolehlivost přenosu doplněním zprávy o přídavné znaky. Pomocí těchto je pak příjemce schopen určit buď, že při přenosu došlo k chybě (detekční) nebo je navíc schopen lokalizovat místo výskytu chyby a opravit ji (korekční).

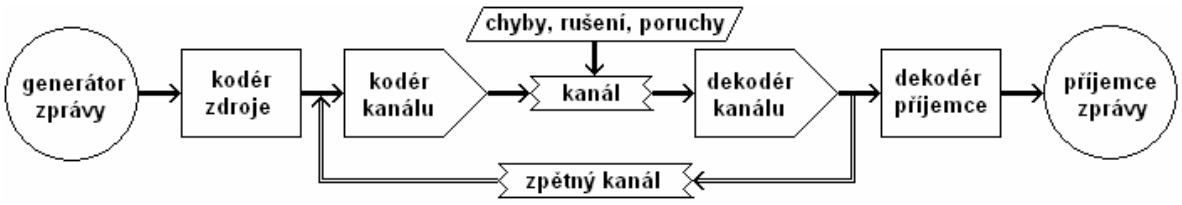
**Kanál** obsahuje další transformace signálu, jako jsou modulace a demodulace, vliv přenosového média a možný výskyt chyby vlivem rušení.

**Dekodér kanálu** je schopen detekovat nebo i opravit nalezené chyby při přenosu a především rekonstruovat signál tak, aby odpovídal vstupu kodéru zdroje.

**Dekodér příjemce** upravuje dekódovanou zprávu na tvar vhodný pro příjemce.

Není-li kladen příliš velký důraz na spolehlivost přenosu zprávy nebo je-li úroveň rušení při přenosu relativně malá, pak si vystačíme s koncepcí na Obr.2. Takové systémy nazýváme FEC (Forward Error Correction) neboli dopředná korekce chyb. Takové systémy jsou údpornější z hlediska přenosové rychlosti (šířky pásma dopředného kanálu), účinnost zabezpečení je ovšem menší.

Požadavky na vysoce spolehlivý přenost dat vedly k doplnění schémata obecného komunikačního systému o člen zpětný kanál (Obr.3). Data jsou totiž obecně zabezpečena pouze detekčním kódem a zpětný kanál je schopen na základě tohoto výsletku vyslat povel k opakování přenosu. Zpětnovazební systémy jsou zkráceně nazývány ARQ (Automatic Request for Repetition) neboli automatická žádost o opakování přenosu.




Obr.3: Shannonovo schéma obecného komunikačního systému se zpětnovazebním kanálem

Rozlišujeme zpětnovazební koncepce dvojího druhu.

#### 1.2.6.1.1. Systémy s rozhodovací zpětnou vazbou DFB

*DFB (Decision Feedback)* - rozhodovací zpětná vazba

Dekodér kanálu v tomto případě vyhodnocuje věrnost jednotlivých slov ve zprávě s využitím detekčního kódu. Není-li zjištěna chyba, vyšle přijímač zpětným kanálem vysílači potvrzení ACK (Acknowledgment). V opačném případě, tedy je-li zjištěna chyba, zažádá přijímač skrze zpětný kanál zasláním příkazu NACK (Negative Acknowledgement) o opakování přenosu daného slova. Zpětný kanál zde tedy slouží pouze k přenosu jednoduchých řídících signálů a rozhodnutí o opakování přenosu je údělem příjemce. Nevýhodou tohoto způsobu je ovšem neschopnost opravy všech chyb, které není daný kód schopen detekovat. Proto je třeba volik druh kódu pečlivě s přihlédnutím k charakteru rozložení chyb.

#### 1.2.6.1.2. Systémy s informační zpětnou vazbou IFB

*IFB (Information Feedback)* - informační zpětná vazba

Přímým kanálem jsou vysílána jen nezabezpečená slova zprávy a zabezpečující část je vepsána v paměti vysílače. Podle přijatého slova je dále přijímacím zařízením vypočtena zabezpečující část. Ta je vyslána zpětným kanálem k vysílači. Zde je výpočet porovnán s údajem v paměti a pokud je výsledek negativní, dochází k opětovnému vyslání daného slova. V případě, že údaj v paměti souhlasí s údajem vypočteným, vyšle vysílač pokyn k uvolnění dat v paměti přijímače a vysílá další slovo. V případě této zpětnovazební koncepce tedy dochází k rozhodnutí o opakování přenosu slova na strané vysílače. Nevýhoda tkví ovšem v zabezpečení srovnatelných přenosových rychlostí na dopředném i na zpětném kanálu. Výhodou je ovšem výrazná spolehlivost v porovnání atributů vyslaného a přijatého slova.

### 1.2.6.2. Druhy sdělovacích kanálů

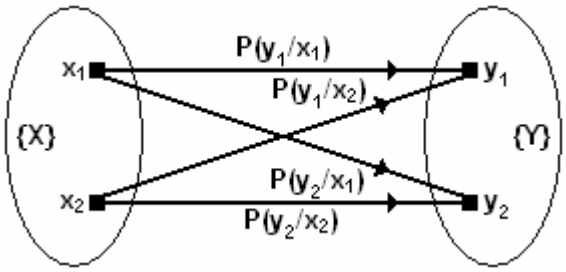
Pojem “kanál” chápeme jako souhrn prostředků pro přenos signálu od generátoru až k příjemci. Diskrétní (spojité) kanály jsou přitom určeny k přenosu diskrétních (spojitých) zpráv.

K parametům kanálů lze uvést několik následujících poznatků. **Bezhlukový (bezšumový) kanál** je případ, kdy informační prvek přijatého signálu vždy odpovídá témuž prvku signálu vyslaného.Opačným případem je tedy **hlukový (šumový) kanál**, kdy si informační prvky signálu na vstupu a výstupu ne vždy odpovídají. **Diskrétním kanálem bez paměti** nazýváme takový kanál, kde výsledek přenosu znaku ze vstupu na výstup je zcela nezávislý na předchozích znacích na vstupu. Opačným případem je zde **diskrétní kanál s pamětí**. Přenosové vlastnosti **stacionárního kanálu** jsou časově nezávislé, jinak jde o **kanál nestacionární**.

### 1.2.6.2.1. Model diskrétního sdělovacího kanálu

Nyní se budeme zabývat až na výjimky diskrétními stacionárními kanály bez paměti, které mohou v dostatečné míře posloužit co by přesný a současně jednoduchý model některých používaných sdělovacích kanálů.

Vstupem kanálu je kanálem přenášená množina znaků {X} s jejich praděpodobnostmi výskytu, zatímco na výstupu kanálu se nachází příjemcem získávaná množina znaků {Y} s jejich pravděpodobnostmi výskytu. Jsou-li množiny dvouprvkové, jde o binární kanál a jsou dány dva vstupy a dva výstupy. Podle Obr. 4 lze odvodit, že pokud byl vyslán (správně přenesen) znak $x_1$, výstupu se může objevit s určitou pravděpodobností znak $y_1$ nebo $y_2$. Stejný případ platí i pro vstupní znak $x_2$. Z toho vyplívají, že vztahové závislosti 15 a 16 pravděpodobností vyslání a příjmu znaku.Z těchto závislostí je možné sestavit přímou matici kanálu 17.




Obr. 4: Informační schéma binárního hlukového kanálu

$$P(y_1, x_1) + P(y_2, x_1) = 1 \quad (15)$$

$$P(y_1, x_2) + P(y_2, x_2) = 1 \quad (16)$$

$$K_{XY} = \begin{pmatrix} P(y_1 / x_1) & P(y_2 / x_1) \\ P(y_1 / x_2) & P(y_2 / x_2) \end{pmatrix} \quad (17)$$

Jestliže jsou známy vstupní pravděpodobnosti výskytů symbolů ‘$x_1$‘ a ‘$x_2$‘, jsme tedy schopni určit pravděpodobnosti výskytu symbolů ‘$y_1$‘ a ‘$y_2$‘ na výstupu kanálu. Děje se tak podle vztahů 18 a 19.

$$P(y_1) = P(x_1) \cdot P(y_1 / x_1) + P(x_2) \cdot P(y_1 / x_2) \quad (18)$$

$$P(y_2) = P(x_1) \cdot P(y_2 / x_1) + P(x_2) \cdot P(y_2 / x_2) \quad (19)$$

Tyto vztahy říkají, že součet dvou členů na jejich pravých stranách znamená, že se daný symbol může na výstupu objevit jako důsledek správného či chybného přenosu. Každý z těchto přenosů je ovšem podmíněn současným výskytem dvou náhodných jevů. Těmito jevy jsou výskyt znaku na vstupu kanálu s určitou pravděpodobností a především jeho transformace na výstup s danou podmíněnou pravděpodobností.

Doposud jsme pohlíželi na výskyt symbolů v závislosti na jejich pravděpodobnosti z pohledu vysílače. Ten je schopen určit pravděpodobnost výskytu znaků podle četnosti jejich vysílání. Kanál se dá ovšem obdobným způsobem popsat i z pohledu příjemce informace. Ten má již k dispozici pravděpodobnosti přijatých znaků a z těch je opět schopný výpočtem získat pravděpodobnosti znaků vyslaných. Při daných výpočtech pracujeme s podmíněnými pravděpodobnostmi $P(x_i/y_j)$, přičemž byl znak $x_i$ vyslán, pokud byl přijat znak $y_j$. K výpočtům těchto simultánních pravděpodobností slouží vztahy 20 a z toho pro nás vyplývají i hledané pravděpodobnosti 21, jejichž jmenovatel je řešen podle vzorců 18 a 19.

$$P(x_i, y_j) = P(x_i) \cdot P(y_j / x_i) = P(y_j) \cdot P(x_i / y_j) \qquad i, j \in \{1,2\} \quad (20)$$

$$P(x_i / y_j) = \frac{P(x_i) \cdot P(y_j / x_i)}{P(y_j)} \qquad i, j \in \{1,2\} \quad (21)$$

Po doplnění náležitých indexů lze sestavit matici kanálu, nyní jde ovšem o matici zpětnou 21 a součet prvků jejích sloupců je roven jedné. Prvky této matice (na

rozdíl od matice $K_{xy}$) jsou již ovšem závislé na pravděpodobnostech výskytu znaků $x_1$ a $x_2$ na vstupu. S použitím vztahů 18 a 19 do této matice a úpravou získáme konečné vzorce pro symetrický kanál 22 a 23. Parametry P a Q zde vyjadřují hodnoty spolehlivosti (P) a nespolehlivosti (Q)

$$K_{YX} = \begin{pmatrix} P(x_1 / y_1) & P(x_1 / y_2) \\ P(x_2 / y_1) & P(x_2 / y_2) \end{pmatrix} \quad (22)$$

$$K_{YX} = \begin{pmatrix} \dfrac{1}{1+\dfrac{P(x_2)\cdot Q}{P(x_1)\cdot P}} & \dfrac{1}{1+\dfrac{P(x_2)\cdot P}{P(x_1)\cdot Q}} \\ \dfrac{1}{1+\dfrac{P(x_1)\cdot P}{P(x_2)\cdot Q}} & \dfrac{1}{1+\dfrac{P(x_1)\cdot Q}{P(x_2)\cdot P}} \end{pmatrix} \quad (23)$$

### 1.2.7. Vzájemná informace

Teorie uváděné ve dřívějších bodech nevyjadřují reálné vlivy na přenosový kanál, který je podle Obr.2 ovlivňován rušivými parametry různého charakteru. Vlivem šumu v průběhu přenosu sice dochází k poklesu množství informace, hlukový kanál ale zprávu doplňuje o takové množství dezinformace, že se pak entropie vstupní a výstupní zprávy jeví jako stejné. Pro uživatele je ovšem zajímavé pouze skutečně přenesené množství informace od generátoru zprávy až k příjemci. Tato informace je označována jako „vzájemná informace“, jejíž výpočet si lze usnadnit pomocí tzv. podmíněných a simultánních entropií.

#### 1.2.7.1. Podmíněné a simultánní entropie

Pro tyto úvahy i nadále poslouží model hlukového kanálu uvedený na Obr. 4.Výskyt znaků na vstupu kanálu je zde opět popsán souborem vzájemně se vylućujících náhodných jevů {X}, výskyt znaků na výstupu obdobně souborem {Y}. V případě, že se jedná o kanál bezhlukový, tj. P=1, Q=0, jsou soubory {X} a {Y} stejného pravděpodobnostního charakteru. V případě jiném, kdy jsou pravděpodobnosti správného či chybného přenosu znaku na výstup stejné, je kanál pro přenos nepoužitelný. Pak jsou tedy soubory {X} a {Y} vzájemně statisticky nezávislé.

Pokud známe rozdělení pravděpodobností v souborech {X} a {Y}, jsem schopni vypočítat entropii těchto souborů. V případě statické závislosti mezi těmito soubory je ale vhodné defininovat další druhy entropií, co by modely tohoto propojení.

#### 1.2.7.1.1. Podmíněná entropie vstupního souboru při známém výstupu

Podmíněná entropie vstupního souboru {X} při známém výstupu $y_j$ je dána obecným vztahem 24, tedy úpravou vztahu pro průměrnou entropii 9.

$$H(X/y_j) = -\sum_j P(x_i/y_j)\log_2 P(x_i/y_j) \qquad [Sh/znak] \quad (24)$$

$H(X/y_j)$ je neurčitost příjemce informace o tom, co je vysláno přes kanál ze vstupu, snížená o zjištění výstupu $y_j$. Pro kanál bezhlukový je dokonce tato neurčitost nulová. Zprůměrováním entropie 24 pro všechny možné výstupy $y_j$ dále docházíme ke vztahu 25 pro tzv. podmíněnou entropii vstupu po čtení výstupu.

$$H(X/Y) = \sum_j P(y_j)H(X/y_j) \quad (25)$$

Při dalších úpravách výrazu 25 prve dosazením 24 a poté použitím vztahu 20 se dostáváme k výpočtu entropie 26, k čemuž je třeba znát prvky zpětné matice kanálu.

$$H(X/Y) = -\sum_i \sum_j P(x_i, y_j)\log_2 P(x_i/y_j) \quad (26)$$

Podmíněná entropie vstupu po přečtení výstupu vyjadřuje průměrnou neurčitost o stavu vstupu po čtení výstupu, přičemž před přečtením byla neurčitost vstupu H(X). Rozdílem zmíněných entropií získáme tzv. vzájemnou informaci ze vstupu na výstup 26. Podmíněnou entropii H(X/Y) je tedy možné v tomto případě chápat jako průměrné množství informace, které se “ztratilo” během přenosu ze vstupu kanálu k příjemci.

$$I(X,Y) = H(X) - H(X/Y) \quad [Sh/znak] \quad (27)$$

#### 1.2.7.1.2. Podmíněná entropie výstupního souboru při známém vstupu

Stejně tak, jako byl v předešlém bodě problém řešen z pohledu přijímače, může na něj být pohlíženo i z pozice vysílače. Podmíněná entropie výstupního souboru {Y} při známém vstupu $x_j$ je dána obecným vztahem 28, tedy úpravou vztahu pro průměrnou entropii 9.

$$H(Y/x_i) = -\sum_j P(y_j/x_i)\log_2 P(y_j/x_i) \qquad [Sh/znak] \quad (28)$$

$H(Y/y_j)$ je neurčitost příjemce informace o tom, co je přijato na výstupu, snížená o zjištění vyslání znaku $x_j$ ze vstupu. Pro kanál bezhlukový je dokonce tato neurčitost nulová. Zprůměrováním entropie 28 pro všechny možné vstupy $x_j$ dále docházíme ke vztahu 26 pro tzv. podmíněnou entropii výstupu po čtení vstupu.

$$H(Y/X) = \sum_i P(x_i)H(Y/x_i) \quad (29)$$

Při dalších úpravách výrazu 29 prve dosazením 28 a poté použitím vztahu 20 se dostáváme k výpočtu entropie 30, k čemuž je třeba znát prvky zpětné matice kanálu.

$$H(Y/X) = -\sum_i \sum_j P(x_i, y_j)\log_2 P(y_j/x_i) \quad (30)$$

Podmíněná entropie výstupu po přečtení vstupu vyjadřuje průměrnou neurčitost o stavu výstupu kanálu po čtení vstupu, přičemž před přečtením byla neurčitost výstupu H(Y). Rozdílem zmíněných entropií získáme tzv. vzájemnou informaci z výstupu na vstup 31. Podmíněnou entropii H(Y/X) je tedy možné v tomto případě chápat jako průměrné množství informace, které se do výstupní zprávy dostal ovlivem rušivého působení kanálu (nemá vliv ve vstupní zprávě).

$$I(X,Y) = H(Y) - H(Y/X) \qquad [Sh/znak] \quad (31)$$

#### 1.2.7.1.3. Simultánní entropie vstupního a výstupního souboru

Simultánní entropie je pro tento případ neurčitostí, která sleduje stavy vstupů i výstupů systému. Při statické nezávislosti vstupů a výstupů by byla tato neurčitost rovna součtu dílčích entropií vstupu a výstupu, pro případ statické závislosti ovšem simultánní neurčitost klesá. Tento druh entropie je možné přesně určit pomocí podmíněných entropií.

Neučitost pozorovatele o stavech X a Y je možné snížit odečtením vstupu, tedy ziskem informace H(X). Výsledkem je neurčitost o stavu Y 32 za podmínky odečtení vstupu. Obdobně platí tato operace i pro případ stavu X 33, což platí, byl-li odečten výstup. Odečtením rovnic 32 a 33 dále dojdeme k výsledku 34.

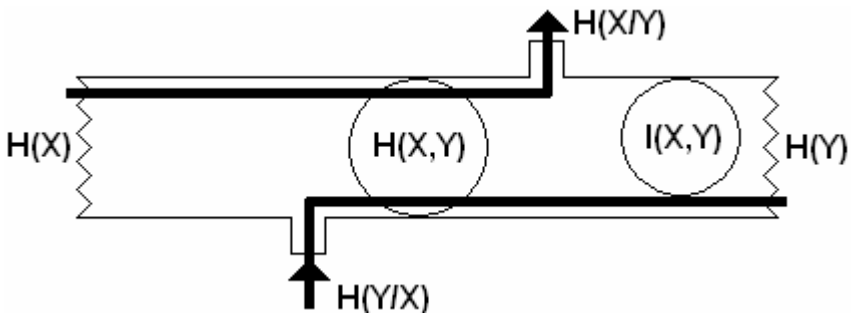
$$H(X,Y) = H(X) + H(Y/X) \quad (32)$$

$$H(X,Y) = H(Y) + H(X/Y) \quad (33)$$

$$\begin{aligned} H(X) - H(X/Y) &= H(Y) - H(Y/X) \\ I(X,Y) &= I(Y,X) \end{aligned} \quad (34)$$

Číselná rovnost těchto dvou informací se promítá i do jejich názvu – vzájemná vstupně-výstupní informace.

Vzájemné vztahy jednotlivými entropiemi dobře vyobrazuje Obr. 5, z něhož jsou mimo jiné zřejmé i dále uvedené nerovnosti 34. Legenda k tomuto obrázku je popsána v Tab. 2.




Obr. 5: Schéma informačních poměrů v hlukovém kanálu

| Entropie | | Vysvětlivka |
|---|---|---|
| H(X) | - | Entropie vstupu |
| H(Y) | - | Entropie výstupu |
| H(X/Y) | - | Ztráta informace |
| | - | Podmíněná entropie vstupu po čtení výstupu |
| H(Y/X) | - | Dezinformace dodávaná hlukovým kanálem |
| | - | Podmíněná entropie výstupu po čtení vstupu |
| H(X,Y) | - | Simultánní entropie vstupního a výstupního souboru |
| I(X,Y) | - | Vzájemná vstupně-výstupní informace |

Tab. 2: Legenda k Obr. 5

$$MIN\{H(X),H(Y)\} \le H(X,Y) \le H(X)+H(Y) \quad (34)$$

Poznámka: Je-li spolehlivost přenosu 100% nebo 0%, teoretický výsledek je v obou případech stejný. Pouze je třeba uvažovat pro druhý případ užitý inverzního kódu, tedy při přenosu jedničky se na výstupu objeví nula a naopak.

# 2. Přehled použitých algoritmů

V této kapitole podrobněji rozeberu postup výpočtu tří algoritmů pro výpočet vzájemné informace z časové řady. Konkrétně půjde o Fraser-Swinneyho algoritmus a Dinh-Tuan-Phamův algoritmus a postup pro výpočet vzájemné informace pomocí adaptivního XY dělení.

## 2.1. Fraser-Swinneyho algoritmus[2]

Princip Fraser-Swinneyho algoritmu (The Fraser-Swinney algorithm) vychází z porovnávání dvojic časově omezených řad. V každé takové řadě se vyskytuje $2^n$ prvků, přičemž je počet výpočtu následovný – názorně demonstruji na příkladě.

[2] Překlad z anglických textů: C. J. Cellucci, A. M. Albano, P. E. Rapp, *Statistic validation of mutual information calculations : Comparisons of alternative numerical algorithms*. Washington: 2004, s. 20-25.; Andrew M. Fraser, Harry L. Seinney, *Independent coordinates for strange attractors from mutual information*. Texas: 1985 [s.n.], 1985, s. 1-7.

Prvním krokem je výběr dvou časově omezených řad o $2^n$ prvcích. Vezmeme nyní v úvahu dvě časově omezené řady, z nichž každý obsahuje osum ($2^3$) prvků, jak znázorňuje Tab. 3a. Prvky dané řady vždy chronologicky od nejmenšího k největšímu seřadíme (Tab. 3b) a tyto poté vyneseme do dvojrozměrné souřadnicové roviny s libovolnou volbou os jako body, jak je vyobrazeno na Obr. 6a.

| **i** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** |
|---|---|---|---|---|---|---|---|---|
| **A(i)** | 9 | 2 | 5 | 8 | 4 | 1 | 3 | 6 |
| **B(i)** | 2 | 4 | 7 | 10 | 5 | 6 | 8 | 3 |

a)

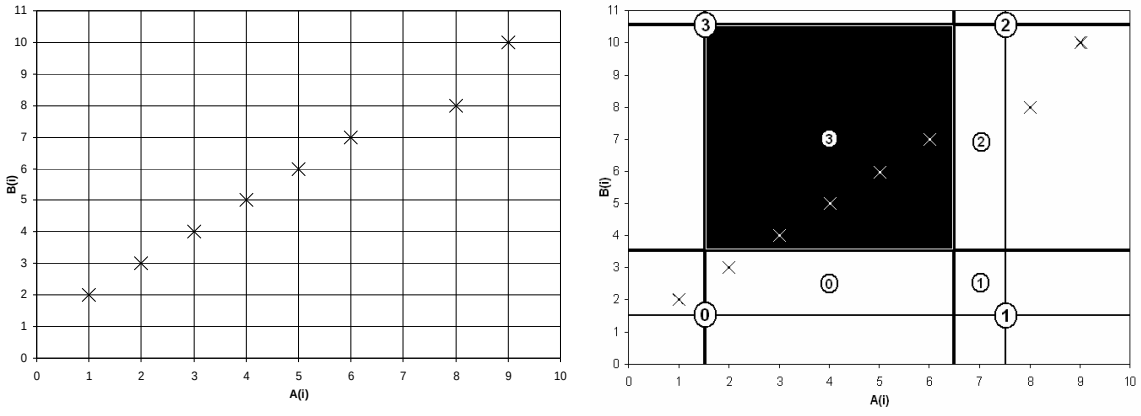
→

| **i** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** |
|---|---|---|---|---|---|---|---|---|
| **A(i)** | 1 | 2 | 3 | 4 | 5 | 6 | 8 | 9 |
| **B(i)** | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 10 |

b)

Tab. 3: Volba (a) a chronologické řazení (b) prvků dvou časově omezených řad

Nyní se dostáváme k samotnému jádru metody, kdy je zapotřebí pomocí vztahů 37 s 38 posoudit platnost substruktury roviny. To se ovšem neobejde bez zjištění chybových parametrů, jejichž výpočet je řešen vztahy 35 a 36.




a) b)

Obr. 6: Vynesení prvků (a) a následné dělení (b) souřadnicového systému

$$a_i \equiv N\left(R_{m+1}\left(K_m, i\right)\right) \quad (35)$$

$$b_{ij} \equiv N\left(R_{m+2}\left(K_m, i, j\right)\right) \quad (36)$$

$$\chi_3^2 = \left(\frac{16}{9}\left(1/N\right)\sum_{i=0}^{3}\left(a_i - N/4\right)^2\right) < 1{,}547 \quad (37)$$

$$\chi^2_{15} = \left( \frac{256}{225} (1/N) \sum_{i,i=0}^{3} \left( b_{ij} - N/16 \right)^2 \right) < 1{,}287 \quad (38)$$

Odtud $R_m(K_m)$ je elementem dělení, tedy prvkem roviny a $K_m$ jeden z celkově možných $4^n$ indexů (pro náš případ jeden ze šestnácti). Za předpokladu, že obrazec nevyhovuje těmto podmínkám (není splněna alespoň jedna z podmínek), je třeba postupně dělit souřadnicový systém, v němž jsou jednotlivé prvky vyneseny. Poměry jednotlivých obdélníků (příp. čtverců) jsou libovolné, lépe je počítat s nerovnoměrným rozdělením, stále ovšem musíme uvažovat celou plochu roviny, nikoli jen její část. Tento postup znázorňuje Obr. 6b. Po procesu dělení rastru je nutné opět provést ověření substruktury roviny. Tento cyklus trvá do doby, než je nalezeno takové rozdělení obrazce, pro jehož všechny díly jsou podmínky 37 a 38 splněny.

Pro každé ověřování substruktury roviny po dělení rastru je dále počítána vzájemná informace 41 v závislosti na splnění podmínek. Děje se tak dle rekurzivního vztahu 39 (v případě nesplnění podmínek), případně dle 40 (pokud podmínky splněny jsou). Jde o jeden z důležitých parametrů k následnému výpočtu vzájemné informace 40.

$$F\left(R_m\left(K_m\right)\right) = N\left(R_m\left(K_m\right)\right)\log\left(N\left(R_m\left(K_m\right)\right)\right) \quad (39)$$

$$F\left(R_m\left(K_m\right)\right) = N\left(R_m\left(K_m\right)\right)\log(4) + \sum_{j=0}^{3} F\left(R_{m+1}\left(K_m, j\right)\right) \quad (40)$$

$$I(S,Q) = (1/N_0)F\left(R_0\left(K_0\right)\right) - \log\left(N_0\right) \quad (41)$$

Zde $N(R_m(K_m))$ značí počet bodů obsažených v prvku roviny $R_m(K_m)$ a $N_0$ vyjadřuje počet prvků řady. Ke vzorci 40 dále dodám, že parametry obsažené v sumarizaci jsou udány následujícím dělením, proto je třeba se k výpočtu $F(R_{m+1}(K_m,j))$ zpětně vracet až po vyhodnocení parametrů substruktur.

## 2.2. Výpočet vzájemné informace pomocí adaptivního XY dělení[3]

Jak již samotný název výpočetní metody napovídá, je zde podobně, jako v případě Fraser-Swinneyho algoritmu (viz. 2.1. Fraser-Swinneyho algoritmus) využit postup adaptivního dělení rastru. Tyto algoritmy jsou si svou podstatou podobné, ovšem s tím rozdílem, že zde je bodový prostor dělen v závislosti na stejném obsazení bodů a hledání hranic jednotlivých substruktur se tak ztěžuje. Nyní se více zaměřím na samotné jádro metody.

Výpočet vzájemné informace by měl být statisticky ověřován aplikací testu předpokladu nulové statistické nezávislosti. K tomu by měl navíc oddíl v rovině XY, použitý k výpočtu společného rozdělení pravděpodobnosti $P_{XY}$, splňovat Cochranovo kriterium 45 očekávané $E_{XY}$. Konkrétně vyžadujeme pro všechny prvky oddílu splnění podmínek, kde $E_{XY}(i,j) \geq 1$ a $E_{XY}(i,j) \geq 5$ pro nejméně 80% prvků oddílu. V následujícím algoritmu užijeme očekávací kritérium ke konstrukci nehomogenního XY oddílu.

Tento postup má dvě podstatné výhody oproti rovnoměrnému rozdělení (uplatňovanému například během aplikace Fraser-Swinneho algoritmu).

- Snižuje citlivost výstupních hodnot X a Y.
- Umožňuje aproximaci oddílu s nejvyšším rozlišením v souladu s očekávacím kritériem.

Nechť $N_D$ označuje počet XY dvojic. $N_X$ vyjadřuje počet prvků, použitých při rozdělení osy X a $N_Y$ pak počet prvků, použitých k rozdělení v ose Y. Pro implementaci tohoto algoritmu, jsou si $N_X$ a $N_Y$ rovny a společně jsou pak označovány jako počet prvků dělení osy $N_E$. Nejedná se tedy o počet prvků v rovině XY (tento parametr by byl roven $N_E^2$). Specifikace $N_E = N_X = N_Y$ je vhodná pro případ, kdy je datový soubor Y zpožděnou verzí datového souboru X.

[3] Překlad anglického textu: C. J. Cellucci, A. M. Albano, P. E. Rapp, *Statistic validation of mutual information calculations : Comparisons of alternative numerical algorithms*. Washington: 2004, s. 17-20.;

### 2.2.1. Určení počtu prvků dělení osy $N_E$

Nejprve je nutná volba rozsahu dělení na dané ose, pro osu x jsou to tedy parametry $x_{min}$ a $x_{max.}$ Po stanovení $x_{min}$ a $x_{max}$, je osa x rozdělena na $N_E$ oddílů tak, že obsazenost prvků je pro každý oddíl stejná. Tato oblast je pak nehomogenní v tom smyslu, že šířka každého prvku je upravena individuálně tak, aby splňovala požadavek jednotného obsazení. Nechť $P_X(i)$ je vyjádřením pravděpodobnosti výskytu X v i-tém prvku oddílu osy x. Pro tuto pravděpodobnost platí vztah 42.

$$P_X(i) = \frac{1}{N_E} \quad (42)$$

Obdobně pak po stanovení $y_{min}$ a $y_{max}$, je osa y rozdělena na $N_E$ oddílů tak, že každý takto vytvořený element osy y je obsazen stejným počtem prvků. Opět platí analogická zákonitost dle vztahu 43.

$$P_Y(j) = \frac{1}{N_E} \quad (43)$$

Podle testu předpokladu nulové statistické nezávislosti, je očekávané obsazení (i,j)-tého prvku při rozdělení roviny XY dáno matematickým vyjádřením 44.

$$E_{XY}(i,j) = N_D P_X(i) P_Y(j) = \frac{N_D}{N_E^2} \quad (44)$$

Hodnota $N_E$ je obvykle určována nalezením nejvyšší hodnoty, která přiřazuje $E_{XY}(i,j) \geq 5$ všem prvkům oddílu XY. Toto kritérium je tedy konzervativnější než Cochranovo kritérium, které vyžaduje pro $E_{XY}$ vyšší hodnoty než pět pro nejméně 80% prvků oddílu. $N_E$ je tedy takové největší číslo, které současně splňuje podmínku 45.

$$N_E \leq \left( \frac{N_D}{5} \right)^{1/2} \quad (45)$$

### 2.2.2. Výpočet vzájemné informace

Po výpočtu pravděpodobností obsazení $P_{XY}(i,j)$ následuje kalkulace vzájemné výměny informace dle vzorce 46.

$$I(X,Y)=\sum_{i=1}^{N_E}\sum_{j=1}^{N_E}P_{XY}(i,j)\log\left\{\frac{P_{XY}(i,j)}{P_X(i)\cdot P_Y(j)}\right\} \quad (46)$$

Pokud ovšem není $N_D$ přesně dělitelné $N_E$, tzn. pokud $N_D$ není násobkem $N_E$, pak oddíly osy x a osy y nemají pravděpodobnosti, přesně se rovnající $1/N_E$. V tomto případě je pak třeba vzorec 12 zjednodušit na následující vztah 47.

$$I(X,Y)=\sum_{i=1}^{N_E}\sum_{j=1}^{N_E}P_{XY}(i,j)\log\left\{N_E^2\cdot P_{XY}(i,j)\right\} \quad (47)$$

Pokud je Cochranovo očekávací kritérium splněno a nulová hypotéza není zamítnuta, pak jsou obě sady dat statisticky nezávislé a lze tedy provést výpočty tímto algoritmem. Za těchto podmínek není vykazování nenulových hodnot vzájemné informace vyžadováno. Proto algoritmus v případě, že není zamítnuta nulová hypotéza, vrací hodnotu I(X,Y)=0, nikoli číselnou hodnotu získanou výpočtem z předchozího vztahu.

## 2.3. Dinh-Tuan-Phamův algoritmus[4]

Úvodem této kapitoly bych rád upozornil na úpravu názvu algoritmu, kdy jsem si dovolil použít jméno autora tohoto algoritmu, co by název. Algoritmus sám o sobě totiž oficiální název nemá, lze jej ovšem interpretovat, jako každý jiný - zabývající se touto problematikou, jako „Rychlý algoritmus odhadu vzájemné informace, entropie a výsledné funkce“, což je ostatně i doslovný překlad titulku dokumentu, z něhož byly čerpány tyto poznatky.

[4] Překlad anglického textu: *Dinh Tuan Pham, *Fast algorithm for estimating mutual information, Entropies and score functions*. France: 2003, s. 1-6.

Zmíněným autorem tohoto algoritmu je tedy vietnamský matematik Dinh Tuan Pham. Zmíněný algoritmus [5], obdobně jako v případě algoritmu předešlého, využívá k výpočtu vzájemné informace mříže či rastru rozšířeného o jádro.

### 2.3.1. Výsledná funkce jako gradient entropické funkce

Nechť Y je náhodný vektor s prvky $Y_1...Y_K$ a hustotou $p_Y$. Množstevní funkce $\psi_Y$ pro Y (také označováno jako společné funkce $Y_1...Y_K$ označující $\psi_{Y1}...\ _{Yk}$) je definována jako gradient – log $p_Y$. Může to být vnímáno jako gradient entropické funkce, v tom smyslu, že pro „malý“ náhodný přírůstek $\partial$Y vektoru Y platí vzorec 48 až po první řád.

$$H(Y+\partial Y)-H(Y)\approx E\left[\psi_Y^T(Y)\partial Y\right] \quad (48)$$

Zde E označuje očekávání pozorovatele a $p_Y$ je označením hustotu Y.
Stručný intuitivní důkaz tohoto zápisu vztahu 47 je uveden následujícím způsobem 48.

$$E\left[\log\frac{p_Y(Y)}{p_Y(Y+\partial Y)}\right]+E\left[\log\frac{p_Y(Y+\partial Y)}{p_{Y+\partial Y}(Y+\partial Y)}\right] \quad (49)$$

Přitom vztah 48 můžeme zapsat i jako 49.

$$E\left[\log\frac{p_Y(Y+\partial Y)}{p_{Y+\partial Y}(Y+\partial Y)}-\frac{p_Y(Y+\partial Y)}{p_{Y+\partial Y}(Y+\partial Y)}+1\right] \quad (50)$$

Jelikož $\log x=x-1-(x-2)^2/2+...$, lze tedy očekávat, že hodnota tohoto výrazu bude vyšší, než $\partial$Y a proto jej můžeme zcela vypustit. Pomocí Taylorova rozvoje o $\log p_Y(Y+\partial Y)$ se dostáváme k požadovanému výsledku.

Z rovnic 49 a 50 připouští vzájemná informace $I(Y_1...Y_K)$ pro první řád následující rozvoj 51

$$I(Y_1,...,Y_k)=\sum_{k=1}^{K}H(Y_k)-H(Y) \quad (51)$$

*rozšíříme na*

$$I(Y_1+\partial Y_1,...,Y_k+\partial Y_k)-I(Y_1,...,Y_k)\approx\sum_{k=1}^{K}E\left\{\left[\psi_{Y_k}(Y_k)-\psi_{k,Y_1...Y_K}(Y_1,...,Y_k)\right]\partial Y_k\right\}$$

Odtud $\psi_{k,Y1}...Y_k$ je k-tá složka této spojité funkce ze spojité výsledné funkce (score functions) $Y_1...Y_K$. Funkce $\psi_k$-$\psi_{k,Y1}...Y_k$ byly zavedeny dříve pod pojmem rozdílové výsledné funkce (SDF-score diference function). Mohou být vnímány jako složky gradientu vektoru funkce vzájemné informace.

Obdobně lze analogicky upravit i následující výraz, pro posloupnost náhodných veličin {Y(n)}. Podmíněná entropie Y(p) vzhledem k Y(1),...,Y(p-1) připouští následující rozvoj prvního řádu 52.

$$H[Y(p)\,|\,Y(1:p-1)]=H[Y(1:p)]-H[Y(1:p-1)]$$

*rozšíříme na*

$$H[Y(p)+\partial Y(p)\,|\,Y(1:p-1)+\partial Y(1:p-1)]-H[Y(p)\,|\,Y(1:p-1)]\approx$$
$$\approx E\left\{\psi^{T}_{Y(p)|Y(1:p-1)}[Y(1:p)]\partial Y(1:p)\right\} \quad (52)$$

*odkud*

$$\psi_{Y(p)|Y(1:p-1)}=\psi_{Y(1:p)}-\begin{bmatrix}\psi_{Y(1:p-1)}\\0\end{bmatrix}$$

Výše uvedená funkce je jiná než gradient vektoru $\log p_{Y(p)|Y(1:p-1)}$, kde $p_{Y(p)|Y(1:p-1)}$ je podmíněno hustotou Y(p) vzhledem k Y(1),..., Y(p-1). Tato funkce bude pouze podmíněnou SCORE funkcí Y(p) vzhledem k Y(1),...,Y(p-1).

### 2.2.2. Metoda odhadu

Hlavní myšlenkou je v prvé řadě odhad entropie (společné, marginální a podmíněné), pak jejich gradient jako odhad rozdílu SCORE podmíněné funkce, podle vztahů popsaných v předchozí části. Tímto způsobem lze odhadnout kritérium pro „nevidomé“ třídění a jeho gradient. Jako nezávislý odhad SCORE funkce je poskytován pouze odhad gradientu teoretického kritéria, který je často odlišný od gradientu kritéria odhadovaného.

### 2.2.2.1 Odhad entropie

K odhadu entropie je zapotřebí odhadu hustoty $p_Y$ náhodného vektoru Y ze vzorku Y(1),...,Y(N), což lze realizovat dle vztahu 53.

$$\hat{p}_Y(y) = \frac{1}{N}\sum_{n=1}^{N}\frac{\kappa\left[h^{-1}(x - Y(n))\right]}{\det h} = \hat{E}\,\frac{\kappa\left[h^{-1}(y - Y)\right]}{\det h} \quad (53)$$

kde κ označuje multivariační hustoty a h je parametr vyhlazení matice (mřížky). Zde a v pokračování, notace Ê označuje operátor středního odběru vzorků. Přirozený odhad H(Y) je pak určen vzorcem 54.

$$\int \hat{p}_Y(y)\log \hat{p}_Y(y)dy \quad (54)$$

Vztah 20 ovšem vyžaduje vícenásobnou integraci ve vícerozměrném prostoru. Proto tedy raději výše uvedené integrace discretizujeme a přepíšeme na sumaci nějaké pravidelné mříže, dle vztahu 55.

$$\sum \hat{p}_Y(gi)\log \hat{p}_Y(gi)\det g \quad (55)$$

Zde je proveden součet pro všechny vektory i s označenými celočíselnými součástmi a g je matice definující velikost a orientaci mřížky. Všimněme si možnosti vyhnout se integraci na základě odhadu entropie Y ve vztahu 56.

$$\frac{1}{N}\sum_{n=1}^{N}\hat{p}_Y[Y(n)] \quad (56)$$

Tato metoda ovšem znamená výpočet hodnoty řádu $N^2$, stejně tak, jako každý prvek $p_Y[Y(n)]$ sám o sobě vyžaduje souhrn podmínek N. Naše metoda, díky vhodné volbě mřížky a jádra, má výpočetní náročnost lineárné rostoucí s N, jak bude uvedeno níže. Dále je důležitý fakt, že umožňuje eliminaci zkreslení, jak bude dále také rozebráno.

Je nutné zvolit g úměrné h, což ostatně dává smysl. Parametr h totiž ovlivňuje vyhlazení a hladší $\hat{p}_Y$ zajišťuje větší než běžný rozměr sítě. Je ovšem také třeba brát v úvahu volbu koeficientu úměrnosti při kalkulacích požadavků a ztrát přesnosti, v důsledku diskretizace. Obecně platí, že nejvhodnější jádro připadá při rovnosti, kdy g=h, mříž tak totiž nevykazuje známky přílišné hrubosti. Uvažujeme-li g=h/m pro nějaké celé číslo m, snižuje se tak rozměr mříže (rastr), ovšem dochází ke zvýšení výpočetní náročnosti faktorem $m^K$. Pro zjednodušení nebudeme uvažovat tuto volbu, přičemž navíc pro K průměrné velikosti, může být $m^K$ pro m=2 příliš velké. Tímto přicházíme k odhadu dle vzorce 57.

$$\hat{H}(Y) = -\sum_i \hat{\pi}_Y(i)\left[\log \hat{\pi}_Y(i) - \log\det h\right] \quad (57)$$

$$\hat{\pi}_Y(i) = \frac{1}{N}\sum_{n=1}^{N} \kappa\left[i - h^{-1}Y(n)\right] = \hat{E}\kappa\left(i - h^{-1}Y\right) \quad (58)$$

Odtud $\hat{\pi}_Y(i)$ je dáno vztahem 58 a můžeme jej vnímat jako odhad pravděpodobnosti, při které náhodný vektor $h^{-1}Y$ náleží do buňky či oblasti vystředěné na jednotky objemu.

V praxi je multivariační jádro κ generováno z jednorozměrného jádra K dle dvou rozhodujících metod.

a) TENSOROVÝ SOUČIN: $\kappa=K^{xK}$, kde K-krát tenzorový součin K je definován matematickým zápisem 59, a odkud $y_k$ označuje složky y.

$$K^{xK}(y) = \sum_{k=1}^{K} K(y_k) \quad (59)$$

b) SFÉRICKÁ SYMETRIE: κ (y)=CK(||y||), kde C vyjadřuje normalizační konstantu, takže κ integruje k jedničce.

Všimněme si, že Gaussovo jádro vyhovuje jak metodě tensorového součinu, tak i metodě sférické symetrie. Nemá ovšem kompaktní podporu. Místo metody tenzorového součinu tedy budeme používat základní drážkování (cardinal spline) nebo třetí řád.

Připomeňme si, že základní drážkování (cardinal spline) řádu r je hustota součtu r nezávislých náhodných veličin na intervalu [-1/2, 1/2]. To inklinuje Gaussovu hustotou (až do škálování) stejně, jako se zvyšuje r při centrálním limitním teorému. Nyní zvolíme třetí základní drážkování, jelikož jde o nejjednodušší postup s průběžným derivátem (stav potřebný k výpočtu gradientu), což je jednoznačně dáno zápisem 60. Krom toho, jsme tak již vcelku blízko Gaussovy hustoty.

$$\mathrm{K}(u) = \begin{cases} 3/4 - u^2, & |u| \le 1/2 \\ (3/2 - |u|)^2 / 2, & 1/2 \le |u| \le 3/2 \\ 0, & \textit{mimo rozsah} \end{cases} \quad (60)$$

Rychlý výpočet $\hat{\pi}_Y$ vychází z faktu, že je hodnocena pravidelná mříž a jádro je produktem jádra s podporou násobku rastru. Skutečně, pokud $Y'_k$ je součástí $h^{-1}$ a termín $K^{xK}[i-h^{-1}Y(n)]$ může být nenulový pouze v případě 27.

$$i_k = \langle Y'_k(n) \rangle \quad \textit{nebo} \quad i_k = \langle Y'_k(n) \rangle \pm 1 \qquad k = 1,...,K \quad (61)$$

Odtud $i_k$ je k-tý člen i a <y> označuje celé číslo nejblížší k y. Tak lze rychle spočítat $\hat{\pi}_Y$ (i) následujícího algoritmu:

Prve je inicializován $\hat{\pi}_Y$ (i) na hodnotu 0, pak jsou za n postupně dosazována celá čísla, tedy n=1,...,N, aktualizace pokračuje dle vztahu 62.

$$\begin{aligned} &\hat{\pi}_Y\left[\langle Y'_1(n)\rangle + i_1,...,\langle Y'_K(n)\rangle + i_P\right] = \\ &\qquad \hat{\pi}_Y\left[\langle Y'_1(n)\rangle + i_1,...,\langle Y'_K(n)\rangle + i_P\right] + \frac{1}{N}\prod_{k=1}^{K}\mathrm{K}\left[i_k + \langle Y'_k(n)\rangle - Y'_k(n)\right], \qquad i_k = -1,0,1 \end{aligned} \quad (62)$$

Všimněme si, že pro interval uϵ[-1/2,1/2] platí rovnosti 63.

$$\mathrm{K}(u) = 3/4 - u^2, \qquad \mathrm{K}(\pm 1 + u) = (1/2 \mp u)^2 / 2 \quad (63)$$

a proto jsou k výpočtu poměrně jednoduché.

Výše uvedený algoritmus vyžaduje smyčku skrze všechny údaje (soubory informací), tedy aktualizaci ‘$3^K$‘ pravděpodobností v každém kroku. V důsledku toho počet indexů i, pro které $\hat{\pi}_Y$ (i) není nula, nemůže překročit $3^K$N a v obecních případech je mnohem méně indexů. Náročnost výpočtu $\hat{\pi}_Y$ (i), stejně jako i odhad entropie, je O($3^K$N) a lineárně rostoucí s N.

Funkce základní křivky disponují zajímavou vlastností zvanou „rozdělení jednoty“, udané vztahy 64 a 65, u kterého zanedbáváme u.

$$\sum_{i=-\infty}^{\infty} \mathrm{K}(u+i) \equiv u \quad (64)$$

$$\sum_i \hat{\pi}_Y(i) = 1 \quad (65)$$

Složky $\hat{\pi}_Y$ (i) tedy představují diskrétní rozdělení pravděpodobnosti a odhad entropie $\hat{H}$(Y) je entropií této distribuce plus termín log det h, jak udává dříve zmíněný vztah 23. Tento odhad má ovšem malou vadu, jelikož translace není tak zcela neměnná. Přidáním konstanty k náhodnému vektoru Y se nemění jeho entropie, což považujeme v podstatě za klad. Proto trochu pozměníme tento odhad, nejprve pomocí centrování dat, která jsou výpočtově $\hat{\pi}_Y$ (i), bereme jako Y’$_k$(n), nikoli k-té složky od $h^{-1}$Y(n), nýbrž od $h^{-1}$[Y(n)-$\tilde{Y}$], kde $\tilde{Y}$ =$\hat{E}$Y je označení vzorku střední hodnoty.

### 2.2.2.2. Odhad vzájemné informace

Zřejmý způsob odhadu vzájemné výměny informací je rozdíl odhadu společné entropie a sumy odhadnutých marginálních entropií. Ovšem, předem musíme pro zrušení zkreslení zvolit h úhlopříčky s diagonálními prvky $h_1$,...,$h_k$, kde $h_K$ je vyhlazovací parametr pro stanovení odhadu mezní hustoty $Y_k$. Potom pravděpodobnost $\hat{\pi}_{Y_k}$ (j) je potřebná k odhadu $\hat{H}(Y_k)$ a je daná vztahem 66.

$$\hat{\pi}_{Y_k}(j) = \frac{1}{N}\sum_{n=1}^{N} \mathrm{K}\left[j - Y_k'(n)\right] = \hat{E}\mathrm{K}\left(j - Y_k'\right) \quad (66)$$

Vztah bude zahrnovat stejné proměnné $Y'_k=[Y_k-\tilde{Y}_k]/h_k$ jako se vyskytují při výpočtu $\hat{\pi}_Y$ (i) v bodě 1.2.2.1. Odhad vzájemné výměny informací je dán zápisem 67, kde pro sumu $\hat{\pi}_{Y_k}$ (i) platí rovnost 68.

$$\hat{I}(Y_1...Y_K) = \sum_i \hat{\pi}_Y(i) \cdot \log \frac{\hat{\pi}_Y(i)}{\prod_{k=1}^{K} \hat{\pi}_{Y_k}(i_k)} \quad (67)$$

$$\hat{\pi}_{Y_k}(j) = \sum_{i:i_k=j} \hat{\pi}_Y(i) \quad (68)$$

Z tohoto zápisu lze očekávat, že sklon (bias) v $\hat{\pi}_Y$ (i) je více či méně ovlivňován prvky obsazenými v marginální pravděpodobnosti $\hat{\pi}_{Y_k}$ ($i_k$), neboť sám původ těchto pravděpodobností vychází z $\hat{\pi}_Y$ (i). Ještě důležitější je fakt, že pokud má vektor Y nezávislé (samostatné) složky $\hat{I}(Y_1,..., Y_K)$, bude konvergovat k nule jako $n\to\infty$, bez ohledu na výběr h, jelikož limit $\hat{\pi}_Y$ (i) je očekávaný výsledek nezávislých náhodných proměnných, což se dále rovná součinu pravděpodobnosti. Tak je tedy možné zvolit poměrně velké h bez obav, že by došlo k návratu $\hat{I}(Y_1,..., Y_K)$ jako u nezávislého empirického kritéria.

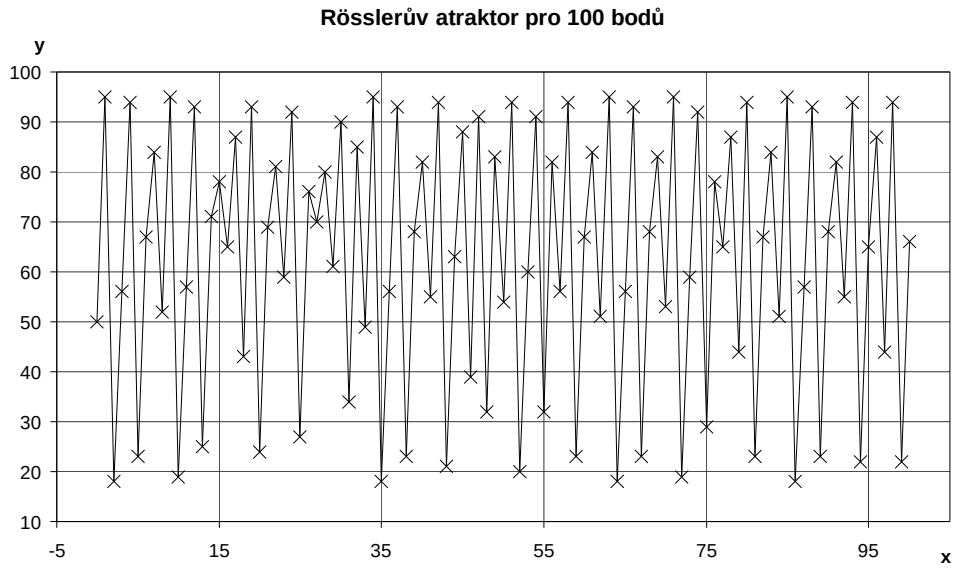
## 3. Analyzovaná data

Pro generování dat, určených k následné analýze (v předcházející kapitole popisovanými) algoritmy pro výpočet vzájemné informace, jsem si vybral tzv. Rösslerův atraktor. Jeho definice vychází z oboru teorie chaosu, zabývajícího se chováním jistých nelineárních dynamických systémů, které (za jistých podmínek) vykazují jev, známý jako deterministický chaos.[5] Pro Rösslerův atraktor je znám průběh vzájemné informace a jeho rovnice pro dvourozměrnou soustavu souřadnic má tvar vztahu 69.

$$x(n+1) = 3{,}8 \cdot x(n) \cdot (1 - x(n)) \; pro \; x(0) = 0{,}5 \quad (69)$$

[5] Citace z WWW: Wikipedia. *Teorie chaosu*. Dostupný z WWW: <http://cs.wikipedia.org/wiki/Teorie_chaosu>

Na nasledujícím znázornění Graf 1 je patrné, že jde o chaotickou funkci a její body nemají zdánlivě nic společné. Záměrně jsem vybral pouhý výřez funkce Rösslerova atraktoru pro názornou ukázku posloupnosti – jak je pospojováno čarou.




Graf 1: Zobrazení Rösslerova atraktoru pro 100 bodů

## 4. Program pro analýzu algoritmů

Tato kapitola a její podkapitoly jsou komentářem i návodem k použití programu, který je součástí mé bakalářské práce. Program slouží k aplikaci, analýze a porovnání efektivity a náročnosti libovolného ze dvou předdefinovaných algoritmů (nebo všech algoritmů současně).

K analýze použitými algoritmy jsou konkrétně tyto:

- Fraser-Swinneyho algoritmus
- Výpočet vzájemné informace pomocí adaptivního XY dělení

Vstupní data jsou reprezentována body, generovanými pro tento konkrétní případ pomocí rovnice Rösslerova atraktoru. Generovaná vstupní data jsou okamžitě ukládána do souboru jednoho z předdefinovaných souborů pro možnost následného importu souboru a použití dat v některém z tabulkových editorů (např. Microsoft Excel, OpenOffice Calc).

### 4.1. Obsah adresáře programu

Adresář analyzačního programu „mutual_information_calc“ obsahuje zdrojový kód programu, který je možné zkompilovat a provozovat v některém z vývojových prostředí jazyka C++. V mém případě se jednalo o vývojové prostředí Microsoft Visual C++ 2008 Express Edition. Záměrně dokládám nezkompilovaný program (zdrojový kód), který je možné si prohlédnout „zblízka“.

Soubory obsahují komentáře k jednotlivým krokům, které program při svém běhu vykonává. Bližší informace o struktuře adresáře „mutual_info_calc“ udává Tab.4.

| Název souboru | Obsah souboru |
|---|---|
| mutual_info_calc.cpp | definice uživatelského prostředí programu a voleb |
| mutual_info_calc.sln | definice parametrů pro otevření projektu ve vývojovém prostředí |
| mutual_info_calc.vcproj | definice parametrů pro modifikace projektu v vývojovém prostředí typu Visual Studio |
| stdafx.cpp | definice důležitých globálních proměnných, struktur a základních funkcí |
| stdafx.h | soubor výpočetních funkcí |
| targetver.h | specifikace pro operační systémy (soubor byl vytvořen vývojovým prostředím) |
| vstup1.tri | textový soubor pro ukládání generovaných vstupních dat pro analýzu Fraser-Swinneyho algoritmem |
| vstup2.tri | textový soubor pro ukládání generovaných vstupních dat pro výpočet vzájemné informace pomocí adaptivního XY dělení |
| vystup1.tri | textový soubor pro ukládání výstupních dat Fraser-Swinneyho algoritmu |
| vystup2.tri | textový soubor pro ukládání výstupních dat výpočtu vzájemné informace pomocí adaptivního XY dělení |

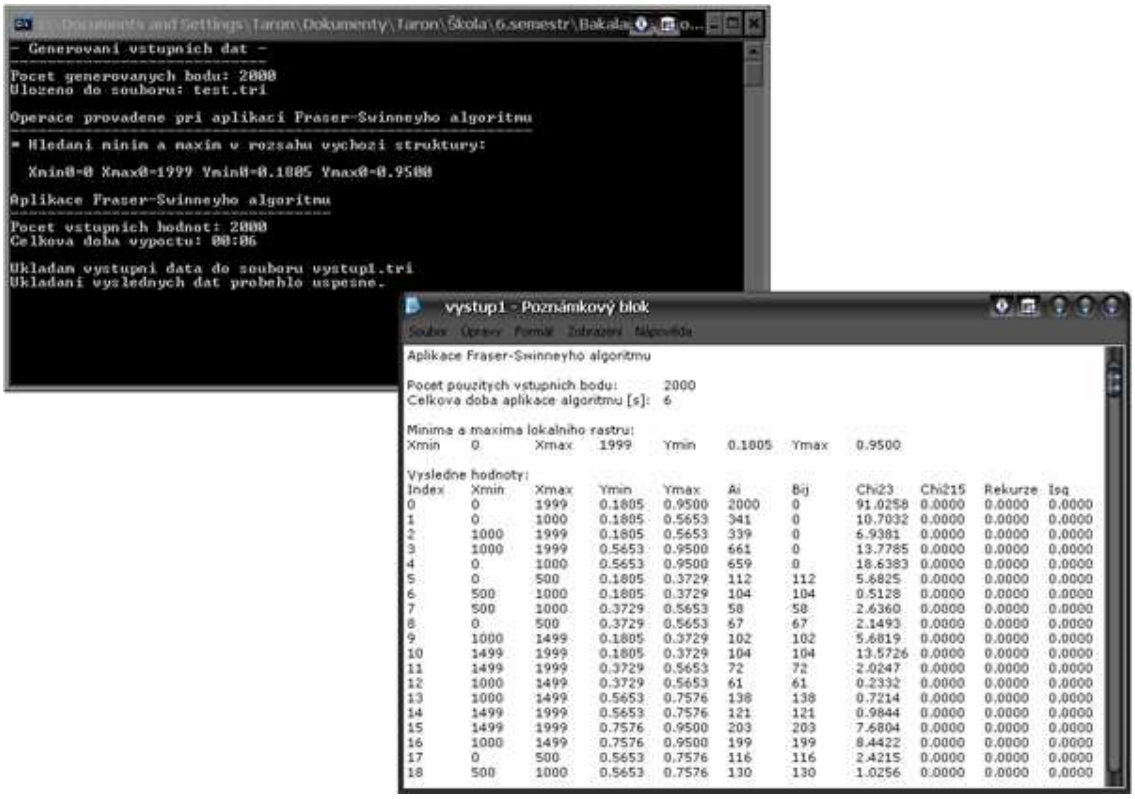
Tab. 4: Obsah adresáře „mutual_info_calc“

### 4.2. Instalace programu

Instalace programu není potřebná, jelikož adresáře se zdrojovým kódem i zkompilovaným jádrem programu jsou snadno přenosné.

### 4.3. Vizuální podoba programu

Aplikace se nezabývá grafickými výstupy a jde tedy pouze o okno pro výstupy textové. Toto reaguje na předdefinované klávesové zkratky (viz. 4.5. Ovládání programu) a lze zadávat například volbu bezprostředně prováděné analýzy algoritmu pro určitý rozsah vstupních dat. Veškerá vstupní i výstupní data jsou ukládána do externích textových souborů pro možnost následného zpracování v některém z tabulkových editorů a to ať už pro vykreslení grafů nebo jinou práci s daty. Ukázky textových výstupů programu jsou k vidění na Obr. 7.




Obr. 7: Ukázky textových výstupů programu pro analýzu algoritmů

### 4.4. Procesní posloupnost programu

Při spuštění programu je zobrazeno okno pro textové výstupy (podobné příkazovému řádku ve Windows nebo Shellu v OS Unix či Linux). Uživatel je následně programem dotazován na výběr bezprostředně analyzovaného algoritmu a jakmile tak učiní, jsou vyžadovány další případné vstupní parametry, pro upřesnění aplikace algoritmu. Generované body jsou okamžitě ukládány do jednoho z předdefinovaných

textových souborů v závislosti na výběru algoritmu. Během provádění algoritmu je bez ohledu na momentální zatížení procesoru počítače měřen časový úsek, po který trvá výpočet výstupních hodnot. Výstupní data jsou uložena do textového souboru s přiděleným názvem (jak udává Tab. 4 v rámci kapitoly 4.1. Obsah adresáře programu) a dochází k dalšímu předložení dotazu na volbu bezprostředně prováděného algoritmu, dokud uživatel běh programu neukončí.

Při běhu programu se doporučuje vypnout nepoužívaná okna, jelikož aktuální zatížení procesoru počítače může výraznou měrou ovlivnit výslednou dobu provádění algoritmu.

### 4.5. Ovládání programu

K ovládání programu slouží zařízení klávesnice, a to především k volbě analyzovaného algoritmu klávesovou zkratkou. Občas je ovšem zapotřebí zadat i číselný parametr (viz. 4.4. Procesní posloupnost programu). Činnost kláves klávesnice popisuje Tab. 4.

| **Klávesové zkratky** | **Vykonávané operace** |
|---|---|
| Enter | potvrzení volby |
| 0 až 9 | zadávání číselných parametrů |
| a, A | analýza vstupních dat Fraser-Swinneyho algoritmem |
| e | ukončení programu |
| r, R | volba pravidelného dělení rastru beze zbytku pro algoritmus výpočtu vzájemné informace pomocí adaptivního XY dělení |
| t, T | volba režimu nepravidelného dělení rastru pro algoritmus výpočtu vzájemné informace pomocí adaptivního XY dělení |
| s, S | výpočet vzájemné informace pomocí adaptivního XY dělení |

Tab. 5: Ovládání programu pomocí klávesnice

### 4.6. Vstupní data

Vstupní data pro analýzu algoritmu jsou generována přímo v jeho pracovním procesu a okamžitě ukládána do - pro aktuální algoritmus - předdefinovaného souboru (viz Tab. 4 v rámci kapitoly 4.1. Obsah adresáře programu). V případě tohoto projektu jsou data generována podle Rösslerova atraktoru. Více o tomto ovšem v kapitole 3. Analyzovaná data.

### 4.7. Výstupní data

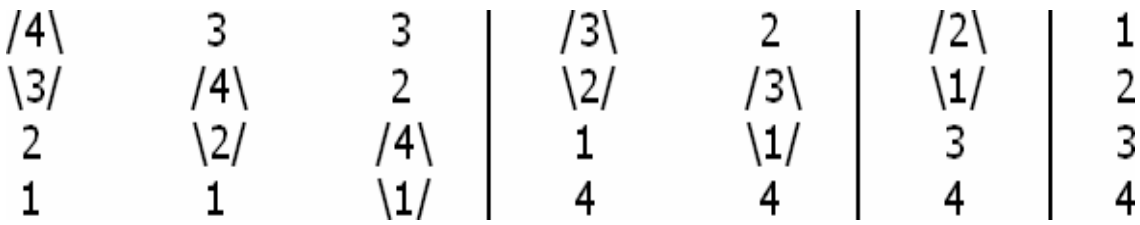
Výstupními daty programu jsou výsledky analýz vstupních dat jednotlivými předdefinovanými algoritmy. Tyto údaje jsou automaticky ukládány do jednoho ze dvou předdefinovaných souborů, jak udává Tab. 4 v rámci kapitoly 4.1. Obsah adresáře programu. Výstupem programu je tedy jeden nebo více textových souborů bez speciálních znaků, použitelný pro možná další zpracování například v tabulkových editorech. V případě ukládání do souborů ovšem upozorňuji na možnost ztráty výstupních dat při opětovném přepisu souborů, data je tedy třeba zálohovat.

# 5. Aplikace jednotlivých algoritmů

Aplikované algoritmy se vzájemně liší svou implementační náročností a jsem se v rámci realizace rogramu uchýlil k doplňkové metodě. Každým z algoritmů, které jsou v programu implementována, je vyžadována vstupní hodnota „n“ pro určení počtu generovaných bodů ($2^n$), kterých by mělo být minimálně 1024. Hodnota konstanty n by tedy neměla být menší než deset.

### 5.1. Fraser-Swinneyho algoritmus

Algoritmus ve svém počátku chronologicky seřadí generované vstupní body. Pro tento účel jsem použil jednoduchou funkci, která vždy porovná dva body v rámci následného vyhodnocení umístění bodů vymění, nebo pokračuje k další dvojici bodů. Postup přesněji demonstruje Obr. 8.

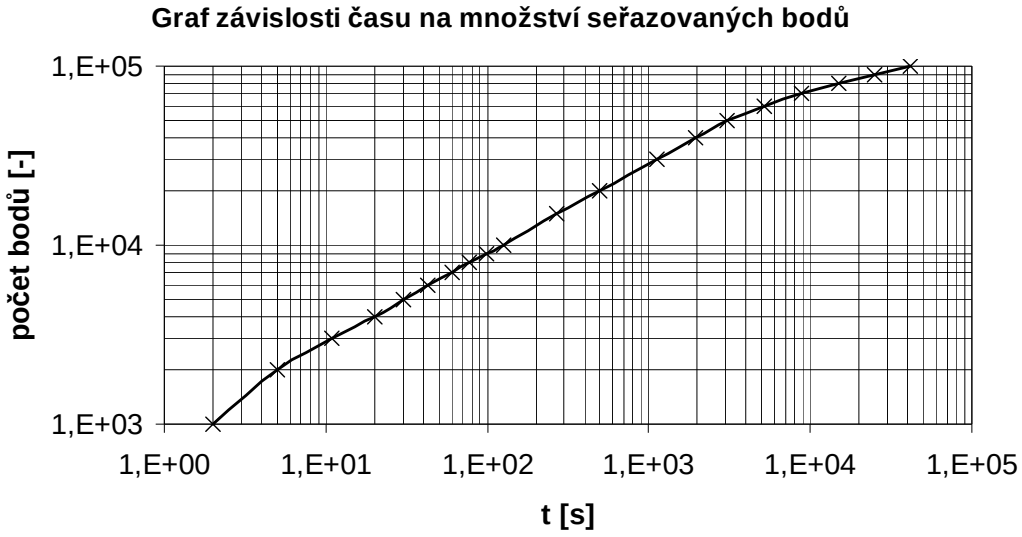



Obr. 8: Ukázka postupu chronologického řazení prvků

Tato metoda seřazování je sice jednoduchá jak logikou, tak i implementaci, pro aplikaci na rozměrnější pole bodů ale poměrně pomalá, i přes fakt, že je v každém cyklu (na Obr. 8 odděleny svislými čarami) ubrán jeden krok, který by tak byl prováděn navíc. Výsledky analýzy rychlosti seřazovací metody jsou shrnuty v Tab. 6 a téměř lineární závislost znázorňuje Graf 2.

| **Výpočet nebo Odhad** | **Počet bodů** | **Trvání** | | **Růstový poměr** | **Rozdíl poměrů** |
|---|---|---|---|---|---|
| | | **program** | **přepočet** | | |
| | | **hod : min : s** | **[s]** | | |
| Výpočet | 1000 | 0 : 0 : 2 | 2 | - | - |
| Výpočet | 2000 | 0 : 0 : 5 | 5 | 2,5000 | - |
| Výpočet | 3000 | 0 : 0 : 11 | 11 | 2,2000 | 0,3000 |
| Výpočet | 4000 | 0 : 0 : 20 | 20 | 1,8182 | 0,3818 |
| Výpočet | 5000 | 0 : 0 : 30 | 30 | 1,5000 | 0,3182 |
| Výpočet | 6000 | 0 : 0 : 43 | 43 | 1,4333 | 0,0667 |
| Výpočet | 7000 | 0 : 1 : 0 | 60 | 1,3953 | 0,0380 |
| Výpočet | 8000 | 0 : 1 : 17 | 77 | 1,2833 | 0,1120 |
| Výpočet | 9000 | 0 : 1 : 39 | 99 | 1,2857 | 0,0024 |
| Výpočet | 10000 | 0 : 2 : 6 | 126 | 1,2727 | 0,0130 |
| Výpočet | 15000 | 0 : 4 : 30 | 270 | 2,1429 | 0,8701 |
| Výpočet | 20000 | 0 : 8 : 19 | 499 | 1,8481 | 0,2947 |
| Výpočet | 30000 | 0 : 18 : 30 | 1110 | 2,2244 | 0,3763 |
| Výpočet | 40000 | 0 : 32 : 26 | 1946 | 1,7532 | 0,4713 |
| Výpočet | 50000 | 0 : 50 : 42 | 3042 | 1,5632 | 0,1899 |
| Odhad | 60000 | 1 : 27 : 7 | 5227 | 1,7183 | 0,1551 |
| Odhad | 70000 | 2 : 27 : 2 | 8822 | 1,6877 | 0,0306 |
| Odhad | 80000 | 4 : 8 : 0 | 14880 | 1,6867 | 0,0010 |
| Odhad | 90000 | 6 : 55 : 46 | 24946 | 1,6765 | 0,0102 |
| Odhad | 100000 | 11 : 33 : 0 | 41580 | 1,6668 | 0,0097 |

Tab. 6: Časová závislost na množství seřazovaných bodů




Graf 2: Graf závislosti času na množství seřazovaných bodů

Zobrazovaná charakteristika může být lineární za předpokladu, že časový úsek trvání metody bude měřen v setinách sekundy. Nikoli sekundovým standartním C++ časovačem (timer), jako tomu bylo v mém případě – ten totiž měřené hodnoty udává vpouze sekundách. Grafická závislost mimo jiné informuje o relativním zpoždění při výpočtu algoritmu, které se tak projeví značnou měrou.

Po seřazení bodů, „vybere“ algoritmus z vytvořeného pole bodů maximální a minimální hodnoty souřadnic (Xmin, Xmax, Ymin, Ymax), které tvoří počáteční hranice rastru a slouží k jeho následnému dělení. V rastru jsou tou dobou již vyneseny vstupní body dané generovanými daty. Souhrnný přehled těchto hraničních souřadnic udává následující Tab. 7. Podotýkám, že tyto hodnoty jsou stejné pro počet bodů minimálně do hodnoty $2^{14}$, tedy 16384 vstupních bodů.

| Xmin | Xmax | Ymin | Ymax |
|---|---|---|---|
| 0 | 16383 | 0,1805 | 0,95 |

Tab. 7: Hraniční souřadnice rastru vstupních bodů

Vzorová vizualizace výstupního průběhu vzájemné informace pro Fraser-Swinneyho algoritmus je uvedena v Příloze 1.

## 5.2. Výpočet vzájemné informace pomocí adaptivního XY dělení

Implementace metody je poměrně nenáročná a také průběh vypočtené vzájemné informace je přesnější, resp. spojitější. Dělení rastru je na rozdíl od Fraser-Swinneho algoritmu jednorázové a v závislosti na počtu bodů, v prvku roviny obsažených. Při správném přednostním odhadu výsledného počtu prvků je předem jistá i pravdivost Cochranova kritéria a proto není nutné, se dodatečně rekurzivně vracet k přepočtu již stanovených hodnot.

Při aplikaci algoritmu jsem postupoval prvotním rozdělením osy X na pravidelné úseky. To je vzhledem v pravidelnému nárůstu hodnot v dané ose X pro použitý Rösslerův atraktor snadno aplikovatelné. Vzhledem k faktu, že počet prvků dělení osy X se rovná počtu prvků dělení osy Y, lze tak dojít k závěru, kolik bodů by měl každý prvek obsahovat. Zjištění, zda daný bod leží či neleží na úrovni dané hodnoty pro osu Y lze zjistit již jednoduchým srovnáváním a následným „krokováním“ v pravidelných intervalech ve směru osy Y. Tak lze snadno v hranicí oboustranně ořízntém prvku roviny dohledat potřebný počet bodů. I přes chaotický charakter vstupních dat je tedy zaručen kýžený počet XY dvojic v omezeném prvku roviny.

Vzorová vizualizace výstupního průběhu vzájemné informace její výpočet pomocí adaptivního XY dělení je uvedena v Příloze 2.

## 5.3. Srovnání aplikovaných algoritmů

V rámci srovnání aplikovaných algoritmů pro výpočet vzájemné informace se v této kapitole zaměřím na dva konkrétní parametry, kterými se zmiňované metody reprezentují i v praktickém využití. Mezi ně patří v prvé řadě rychlost výpočtu a dále pak přesnost, rexp. spojitost vypočteného průběhu vzájemné informace.

### 5.3.1. Rychlost výpočtu

V rámci analýzy rychlosti výpočtu jsem dospěl k zajímavému poznatku. Výpočet vzájemné informace Fraser-Swinneyho algoritmem se jeví být rychlejší pro menší datové toky a následně pro větší množství dat. V případě výpočtu vzájemné informace

pomocí adaptivního dělení je výsledek přesně opačný, přičemž rychlejší aplikace je bez pochyby při pravidelném dělení, kdy $N_D$ je přesně dělitelné $N_E$.

Závislost časového limitu na počtu zpracovávaných bodů je zřejmá z Přílohy 3.

### 5.3.2. Spojitost průběhu vzájemné informace

Při porovnání přesnosti lze již vizuálním hodnocením průběhu vzájemné informace určit, že přesnější metodou je výpočet vzájemné informae pomocí adaptivního XY dělení (eliminující zjevné píky).

Vzorové porovnání přesnosti resp. spojitosti vizualizace výstupního průběhu vzájemné informace je uvedena v Příloze 4.

# Závěr

V závěrečné bakalářské práci jsem se zabýval teorií informace a dále vzájemné informace, přičemž jsem provedl analýzu a následnou implementaci tří rozdílných algoritmů určených k výpočtu parametru vzájemné informace. S jedním z nich bylo pracováno pouze teoreticky (Dinh-Tuan-Pham algoritmus – pracovní název) a se zbývajícími dvěma v praktické aplikaci (Fraser-Swinneyho algoritmus a algoritmus pro výpočet vzájemné informace pomocí adaptivního XY dělení). Tyto dva byly vzájemně posuzovány z hlediska rychlosti výpočtu, implementační náročnosti a přesnosti resp. spojitosti průběhu. Fraser-Swinneyho algoritmus se jeví jako implementačně náročnější, oproti algoritmu pro výpočet vzájemné informace pomocí adaptivního XY dělení, který je i přesnější. Dinh-Tuan-Phamův algoritmus nebyl prakticky testován z důvodu jeho implementační náročnosti, a proto byl popsán pouze z hlediska teoretického.

Jako nejvhodnější ze skupiny algoritmů, vyvinutých pro výpočet vzájemné informace, bych tedy upřednostnil výpočet pomocí adaptivního XY dělení. Jelikož byla vstupní data generována pomocí rovnice Rösslerova atraktoru, je tím usnadněna i implementace zmiňovaného algoritmu. Nárůst hodnot v ose X je totiž pravidelný a toho jsem se při implementaci algoritmu také držel. Tento algoritmus je i rychlejší (pro větší počet prvků) než Fraser-Swinneyho algoritmus.

## Seznam grafů:



## Seznam obrázků:



## Seznam tabulek:

# Literatura

[1] BIOLEK, D.: *Datová komunikace*. Skriptum VUT, VUTIUM 2002.

[2] CELLUCCI, C.J., ALBANO, A.M, RAPP, P.E. *Statistic validation of mutual information calculations : Comparisons of alternative numerical algorithms*. Washington : [s.n.], 2004. Dostupný z WWW:<http://www.dtic.mil/cgi-bin/GetTRDoc?Location=U2&doc=GetTRDoc.pdf&AD=ADA445843>. The Fraser-Swinney algorithm, s. 20-25.

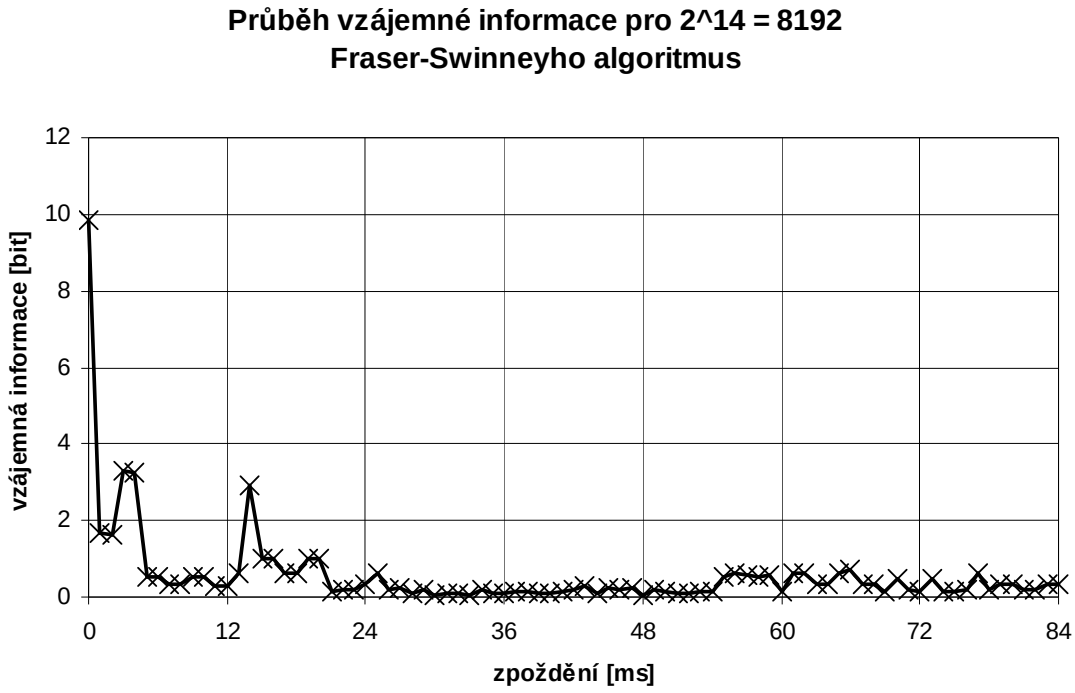
[3] FRASER, Andrew M., SWINNEY, Harry L. *Independent coordinates for strange attractors from mutual information*. Texas : [s.n.], 1985. 7 s. Dostupný z WWW: <http://chaos.utexas.edu/manuscripts/1064949034.pdf>.

[4] KACÁLEK, Jan, MÍČA, Ivan. *Nelineární analýza a predikce síťového provozu*. VUT v Brně, Elektrorevue 2009. Dostupný z WWW: <http://elektrorevue.cz/cz/clanky/komunikacni-technologie/0/nelinearni-analyza-a-predikce-si-oveho-provozu/>.

[5] PHAM, Dinh Tuan. *Fast algorithm for estimating mutual information, Entropies and score functions*. France : [s.n.], 2003. 6 s. Dostupný z WWW: <http://ljk.imag.fr/membres/Dinh-Tuan.Pham/BSS/mutinf-score.pdf>.

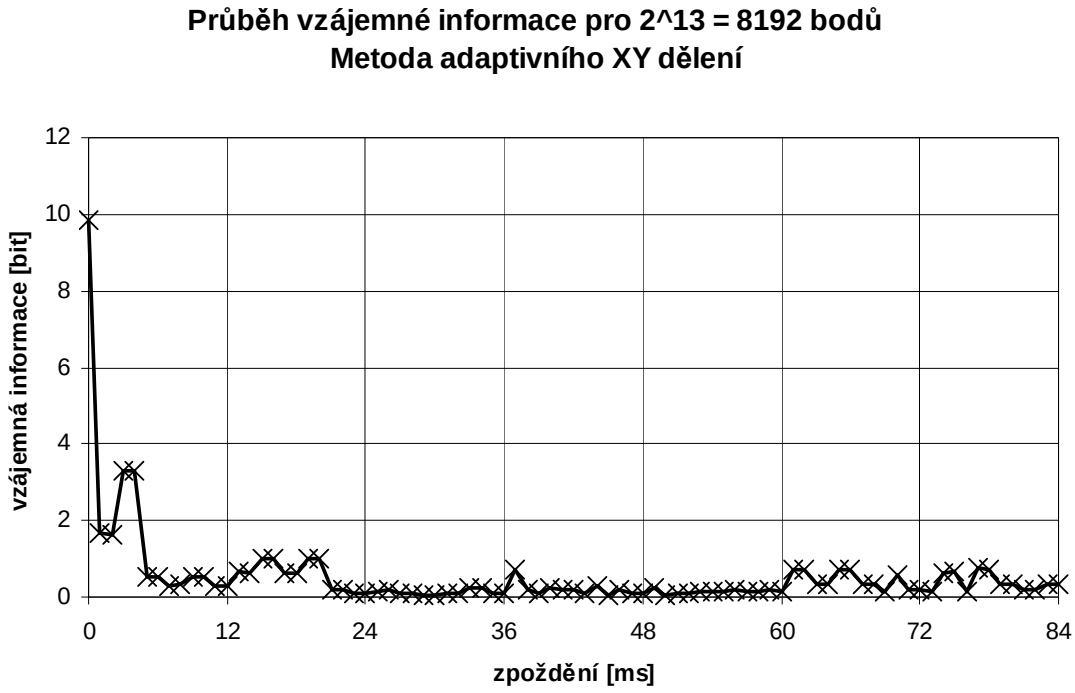
[6] WIKIPEDIA. *Teorie chaosu*. Dostupný z WWW: <http://cs.wikipedia.org/wiki/Teorie_chaosu>

## Seznam příloh




**Report programu:**

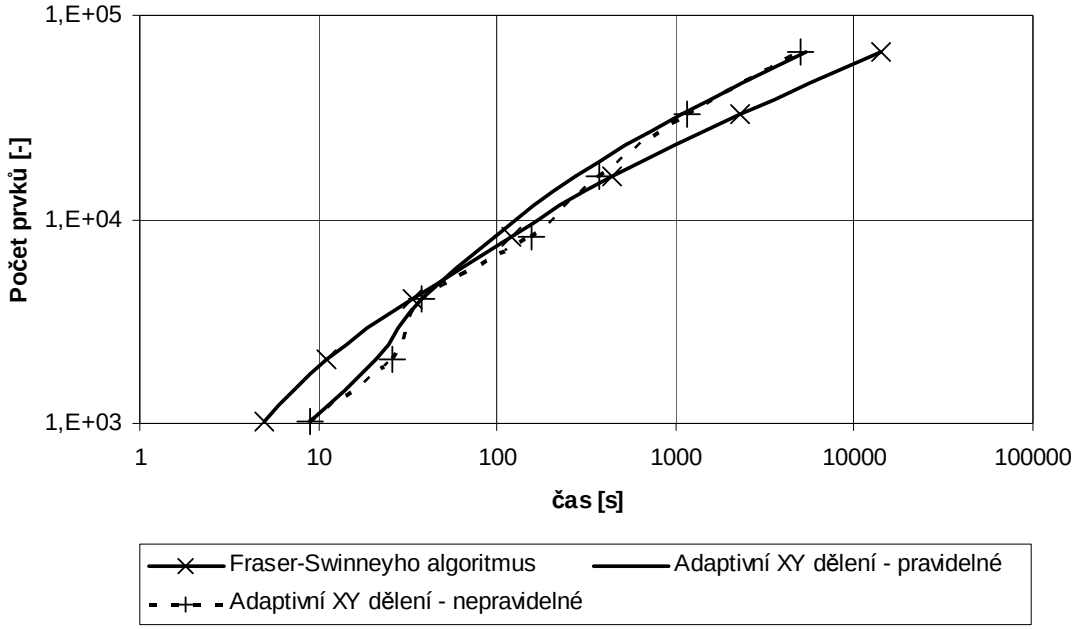
- Analyzovaný algoritmus:
  - Fraser_Swinneyho algoritmus
- Generace vstupních dat: rovnice Rösslerova atraktoru
- Vstupní parametr z klávesnice: n = 13
- Počet vstupních bodů: $2^{13} = 4096$
- Počet výstupních hodnot: 85



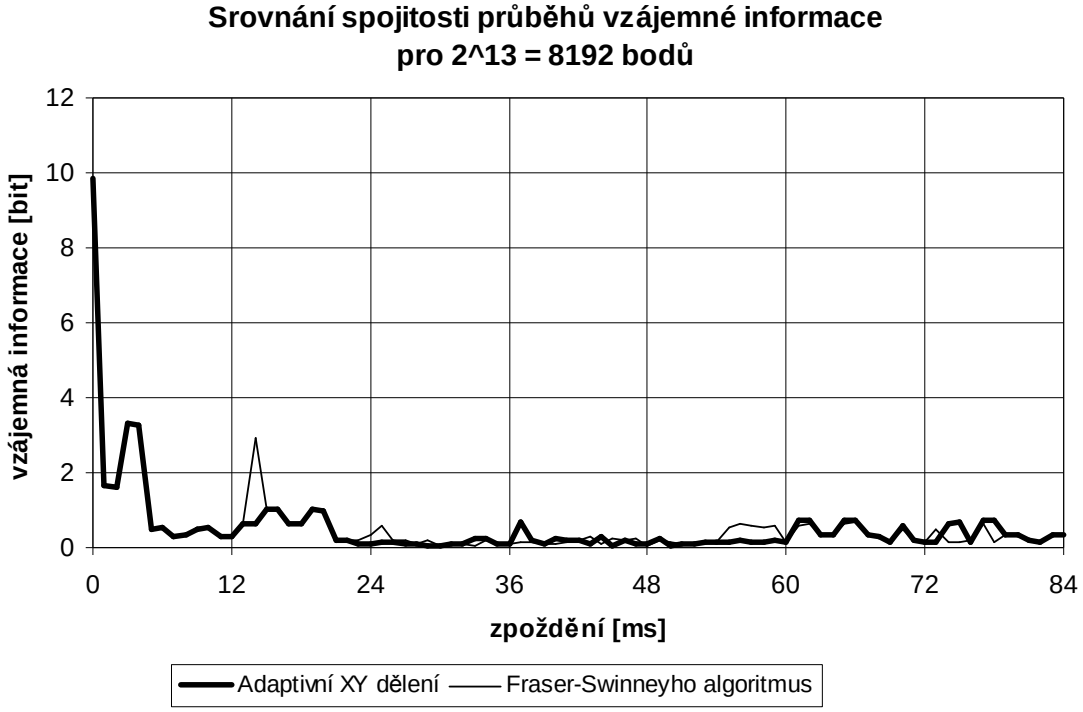

**Report programu:**

- Analyzovaný algoritmus:
  - Výpočet vzájemné informace pomocí adaptivního XY dělení
- Generace vstupních dat: rovnice Rösslerova atraktoru
- Vstupní parametr z klávesnice: n = 14
- Počet vstupních bodů: $2^{14} = 8192$
- Počet výstupních hodnot: 85

## Srovnání časové závislosti na počtu vstupních bodů




| Index n | Počet bodů 2^n | Doba výpočtu | | |
|---|---|---|---|---|
| | | Fraser-Swinneyho algoritmus | AXY dělení | |
| | | | pravidelné | nepravidelné |
| [-] | [-] | [s] | [s] | [s] |
| 10 | 1024 | 5 | 9 | 9 |
| 11 | 2048 | 11 | 21 | 26 |
| 12 | 4096 | 34 | 38 | 38 |
| 13 | 8192 | 120 | 97 | 157 |
| 14 | 16384 | 441 | 272 | 372 |
| 15 | 32768 | 2308 | 1083 | 1150 |
| 16 | 65536 | 14020 | 5400 | 5000 |




**Report programu:**

- Analyzované algoritmy:
  - Fraser-Swinneyho algoritmus
  - Výpočet vzájemné informace pomocí adaptivního XY dělení
- Generace vstupních dat: rovnice Rösslerova atraktoru
- Vstupní parametr z klávesnice: n = 14
- Počet vstupních bodů: $2^{14} = 8192$
- Počet výstupních hodnot: 85