VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF TELECOMMUNICATIONS

MOŽNOSTI VÝPOČTU VZÁJEMNÉ INFORMACE Z ČASOVÉ ŘADY

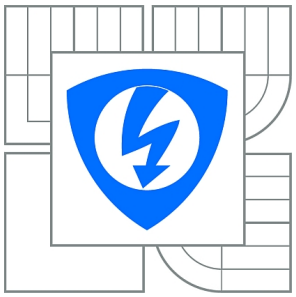
BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE IVO HUBR
AUTHOR

BRNO 2011

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

DEPARTMENT OF TELECOMMUNICATIONS

MOŽNOSTI VÝPOČTU VZÁJEMNÉ INFORMACE Z ČASOVÉ ŘADY

POTENTIAL CALCULATION OF MUTUAL INFORMATION FROM A TIME SERIES

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

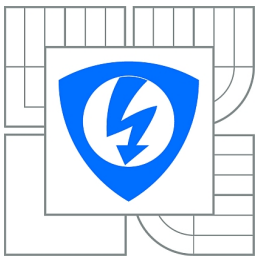
AUTOR PRÁCE — IVO HUBR

AUTHOR

VEDOUCÍ PRÁCE — Ing. JIŘÍ MEKYSKA

SUPERVISOR

BRNO 2011



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky a komunikačních technologií

Ústav telekomunikací

Bakalářská práce

bakalářský studijní obor
Teleinformatika

Student:	Ivo Hubr	***ID:***	106480
Ročník:	3	***Akademický rok:***	2010/2011

NÁZEV TÉMATU:

Možnosti výpočtu vzájemné informace z časové řady

POKYNY PRO VYPRACOVÁNÍ:

Cílem bakalářské práce je rozbor aktuální problematiky teorie informace se zaměřením na vzájemnou informaci. V rámci práce budou prostudovány různé algoritmy výpočtu vzájemné informace, přičemž tyto algoritmy budou dále implementovány v jazyce C/C++ a budou srovnány jejich vlastnosti.

DOPORUČENÁ LITERATURA:

[1] Fraser,A.M. Swinney,H.L. (1986). Independent coordinates for strange attractors from mutual information. Phys. Rev. 33A, 1134-1140.
[2] Cellucci C. J. Albano A. M. Rapp P. E. (2004). Validation of mutual information calculations: comparisons of alternative numerical algorithms

Termín zadání:	7.2.2011	***Termín odevzdání:***	2.6.2011

Vedoucí práce: Ing. Jiří Mekyska

prof. Ing. Kamil Vrba, CSc.
Předseda oborové rady

ABSTRAKT

Vzájemná informace je jedním z faktorů, využívaných při analýze síťového provozu a sestavení fázového prostoru. Úvod práce se zaměřuje na teorii informace zhlediska výpočtu vzájemné informace. K výpočtu tohoto parametru je k dispozici již řada algoritmů, které jsou v závěrečné práci podrobně rozebrány. 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 této práce tedy je srovnání efektivity, rychlost výpočtu a přesnost zmíněných algoritmů.

KLÍČOVÁ SLOVA

Vzájemná informace, analýza síťového provozu, Fraser-Swinneyho algoritmus, výpočet vzájemná informace pomocí adaptivního XY dělení, Rösslerův atraktor, Dinh-Tuan-Phamův algoritmus, efektivita, rychlost výpočtu.

ABSTRACT

Mutual information is one of the factors used in traffic analysis and preparation phase space. Begin of this work deal with information theory, focusing on the calculation of mutual information. To calculate this parameter has been available for many algorithms which are analyzing in this 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 this work is a comparison of efficiency, speed and accuracy of the calculation of these algorithms.

KEYWORDS

Mutual information, traffic analysis, Fraser-Swinney algorithm, calculation of mutual information using adaptive XY subdivision, Rössler' attractor, Dinh-Tuan-Pham algorithm, efficiency, speed of the calculation.

HUBR, Ivo *Možnosti výpočtu vzájemné informace z časové řady*: bakalářská práce. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2011. 59 s. Vedoucí práce byl Ing. Jiří Mekyska

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.

Brno ....

(podpis autora)

Na tomto místě bych rád vyjádřil poděkování vedoucímu mé bakalářské práce
Ing. Jiřímu Mekyskovi za cenné rady a veškerý čas, který mi věnoval.

OBSAH

SEZNAM OBRÁZKŮ

SEZNAM TABULEK

ÚVOD

Bakalářská práce se zaměřuje na způsoby 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 [2], ze kterého 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:

- množství skutečně přeneseného množství informace od zdroje k příjemci. [2]
- 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 většinou udáváno v bitech. V praktickém využití je vzájemná informace důležitým parametrem například pro rekonstrukci fázového prostoru, což je prostor ve kterém jsou reprezentovány všechny možné stavy systému. Dále lze s její pomocí například analyzovat obraz z hlediska četnosti detailů nebo využít při porovnávání šifrovacích klíčů v oblasti kryptografie. V tomto případě pak jde o jeden z požadavků metod 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. [8]

K výpočtu vzájemné informace byla vyvinuta již řada vhodných algoritmů a z těchto byly vybrány 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.

Tato práce je zaměřena na aplikaci prvních dvou ze zmíněných algoritmů na vstupní data, generovaná rovnicí Rösslerova atraktoru, s cílem porovnávání jejich rychlosti, 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 je kladen důraz na význam informace a charakteristiku jejích matematických parametrů.

1.1 Teorie informace

Charakteristika pojmů: [2]

- **Zpráva** je jakákoliv posloupnost rozlišitelných znaků.
- **Symboly** jsou rozlišitelné prvky ve zprávě (v grafickém znázornění jde o znaky).
- **Abeceda** je konečná množina všech symbolů, případně znaků.
- **Signál** je fyzikální nositel zprávy.
- **Kódování** je transformace zprávy vyjádřené symboly jedné abecedy na zprávu vyjádřenou symboly druhé abecedy.
- **Informace**
  - jsou vztahy mezi symboly zprávy a okolním světem, omezené ve vztazích:
    - mezi označením a významem,
    - mezi významem a jejich překladem,
  - bývá dělena do tří odvětví,
    - syntaktická neboli skladební,
    - sémantická (sémantika – nauka o významu slov),
    - pragmatická (vztah znaků k jejich mluvčím).

Obecně platí, že se zvyšující se délkou zprávy a tím i počtu symbolů, se zvyšuje i neurčitost (entropie) zprávy a snižuje hodnota vzájemné informace.

1.2 Vlastnosti informace

Signál je tvořen posloupností n kódových slov (informačních prvků) 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 (1.1) [2]

$$N_Z = N_K^n. \tag{1.1}$$

V tomto případě je tedy hovořeno o jakési formě informační kapacity daného signálu. Obdobou tohoto zápisu je také informační kapacita soustavy, kterou se v roce 1928

zabýval pan 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 (1.2) [2]

$$C = \log_2 N_S. \tag{1.2}$$

Zde N_S je počet všech možných stavů soustavy a jednotkou informační kapacity je Shannon [Sh]. 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 (1.3) [2]

$$C = \log_2(N_{S1}.N_{S2}) = \log_2 N_{S1} + \log_2 N_{S2} = C_1 + C_2. \tag{1.3}$$

Informační kapacita sama o sobě může sice udávat množství informace dané soustavy, pomíjí ovšem 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 Vyjádření množství informace pomocí náhodných jevů

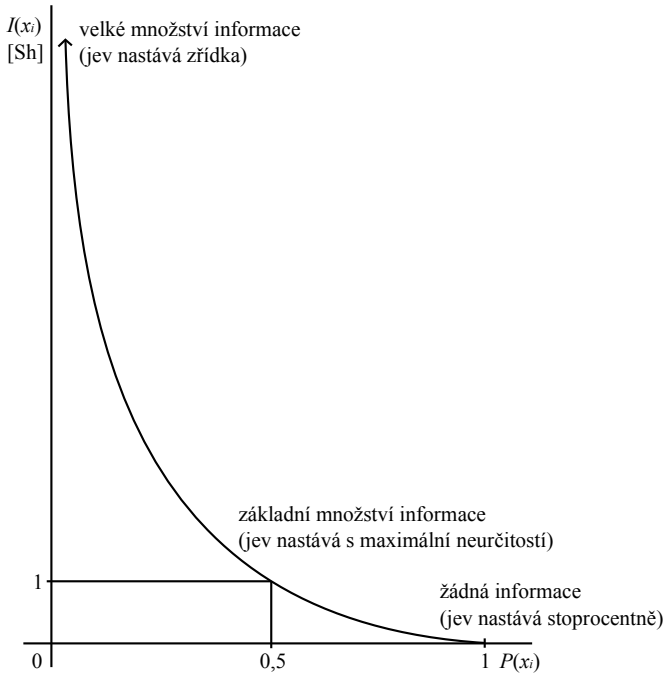
Je obecně známo, že fyzikální jevy sledujeme pomocí signálů, které jsou jimi generovány. Pokud ovšem nejsme schopni předem určit hodnoty těchto signálů v daných časových okamžicích, říkáme, že jde o jevy náhodné. Tyto jevy se vzájemně vylučují, jelikož v daný časový okamžik může být ze souboru náhodných 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 $X = \{x_1, x_2, \ldots, x_N\}$ společně s pravděpodobnostmi jejich výskytu $P = \sum_{i=1}^{N} P(x_i)$ se někdy zapisují do tzv. konečného schématu a pro tyto platí (1.4) [2]

$$\begin{pmatrix} x_1 & x_2 & x_3 & \ldots & x_i & \ldots & x_N \\ P(x_1) & P(x_2) & P(x_3) & \ldots & P(x_i) & \ldots & P(x_N) \end{pmatrix}. \tag{1.4}$$

1.2.2 Vyjádření množství informace pomocí pravděpodobnosti

Množství informace získané příjemcem po přijetí zprávy, že v daný okamžik došlo k výskytu jevu z úplného souboru vzájemně se vylučujících jevů X, je dáno následujícím vztahem (1.5): [10]

$$I(x_i) = \log_2 P(x_i) \text{ (Sh)}. \tag{1.5}$$



Obr. 1.1: Grafické znázornění vzorce pro výpočet množství informace získané příjemcem.

Z pohledu užitečnosti tedy chápeme vzorec (1.5) následovně. Pokud nastal jev, který je zákonitě nevyhnutelný, např. že jablko ze stromu vždy spadne dolů, přiřadíme nulovou informační hodnotu, podle obrázku 1.1 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 (1.6), který navíc zajišťuje výše zmiňovanou aditivitu (nalezení hodnoty proměnné sečítáním jejích dílčích hodnot).

Oproti tomuto může být také výše zmiňovaný vzorec chápán 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 log2n bitů. Každý znak, kterému přísluší vždy hodnota pravděpodobnosti, je nutno zakódovat $\log_2(1/P)$ bity. Přitom musí být množství informace spojené s výskytem tohoto znaku úměrné následujícímu výrazu (1.6): [10]

$$I = k.\log_2 \frac{1}{P} \text{ (Sh)}. \tag{1.6}$$

Konstantu k lze odvodit volbou základu logaritmu. Jednotku množství informace odvodíme pomocí tabulky 1.1.

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

Tab. 1.1: Modifikace jednotky množství informace.

Základ logaritmu	Jednotka	Hodnota v Shannonech
2	1 Sh (Shannon)	1 Sh
10	1 Hartley	3,32 Sh
e	1nat	1,44 Sh

kupříkladu jednovýstupový logický člen, na jehož výstupu 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, můžeme pravděpodobnost přijetí zprávy počítat dle vztahu (1.7) a výsledkem tedy bude (1.8):

$$P(1).P(1).P(0).P(1) = P(1)^3.P(0) = 0{,}9^3.0{,}1 = 0{,}0729, \tag{1.7}$$

$$I = -\log_2 0{,}0729 = 0{,}152 \text{ Sh}. \tag{1.8}$$

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 $I(1) = -\log_2 0,0625 = 4$ Sh.

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. Jako v předešlých případech, i nyní uvažujeme úplný soubor N jevů s konečným schématem (1.4). 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 pouze jedna z pravděpodobností bude nabývat hodnoty 1 (ostatní tedy nulové). Takový soubor již ovšem není náhodný, nýbrž deterministický, jelikož bude pravidelně docházet pouze k výskytu jevu s pravděpodobností 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. Protože současně příjemce získává informaci o tom, nastane-li daný jev. Lze tedy říci, že entropie úplného souboru jevů se číselně rovná množství informace připadající na výskyt jednoho jevu. [2]

Entropie musí jako funkce pochopitelně splňovat i několik požadavků:

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

Teorie informace uvádí několik rozlišovaných typů entropie:

a) Průměrná entropie

Požadavkům uvedeným v kapitole 1.2.3., 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 nebo také Sh/prvek. Pro průměrnou entropii platí (1.9): [10]

$$H(X) = -\sum_{i=1}^{N} P(x_i)\log_2 P(x_i) \text{ (Sh/symbol)}. \tag{1.9}$$

b) Maximální entropie

Maximální entropie při rovnoměrném rozložení pravděpodobnosti je platná pro soubory s N jevy. Pro její výpočet je obecně dáno (1.10): [10]

$$H_{max} = \log_2 N. \tag{1.10}$$

Srovnáním s (1.2) je dán závěr, 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, na základě nějakého algoritmu či grafického zobrazení.

c) Relativní entropie

Relativní entropie vyjádřuje poměr entropie a její maximání hodnoty (1.11): [10]

$$h = \frac{H}{H_{max}} \in \langle 0, 1 \rangle . \tag{1.11}$$

d) Redundance

Na základě relativní entropie je zaváděn pojem redundance nebo také nadbytečnost (1.12): [10]

$$r = 1 - h = \frac{H_{max} - H}{H_{max}} \in \langle 0, 1 \rangle . \tag{1.12}$$

1.2.4 Entropie zprávy

Pojem entropie byl v rámci teorie informace původně zaveden pro úplný soubor vzájemně se vylučujících jevů a v tomto momentě je možné hovořit i 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. Jsou rozlišovány dva případy:

a) 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 abecedy H_{abc} délkou zprávy L (počtem jejích prvků), získáme „průměrné“ množství informace nesené celou zprávou I_{zpr} dle (1.13): [2]

$$I_{zpr} = L.H_{abc} \text{ (Sh)}. \tag{1.13}$$

b) 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, jak uvádí (1.14): [2]

$$I_{zpr} < L.H_{abc} \text{ (Sh)}. \tag{1.14}$$

Mimo jiné je tento vztah interpretovatelný i tak, že informační hodnota zprávy klesla na základě vazeb mezi znaky, jelikož klesla 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 charakter. 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 (1.11) vypočíst nadbytečnost, která vyjadřuje jeho „informační rezervu“.

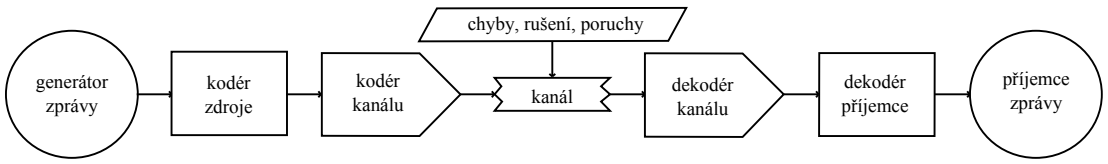
1.2.6 Kódování zpráv na rozhraní

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

Kódováním je myšlen proces změny syntaktické (skladebné) 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.

Shannonovo schéma komunikačního systému

Vědec jménem Claude Elwood Shannon v 50. letech dokázal, že takřka všechny komunikační systémy užívané od minulosti až do dnešní doby jsou pouze zvláštní případy obecného komunikačního systému. [2] Tento model definuje obrázek 1.2.



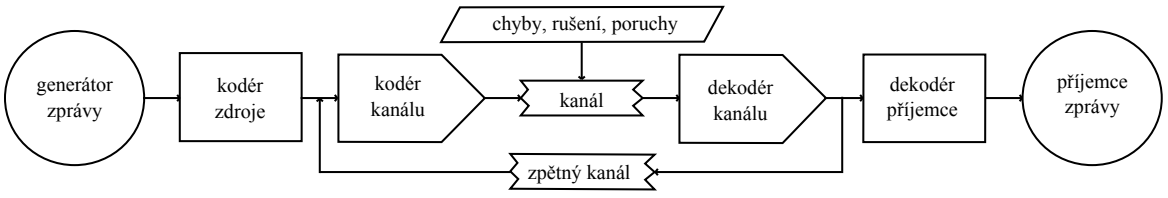
Obr. 1.2: Shannonovo schéma obecného komunikačního systému.

Účel 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.
- **Kodér kanálu** zabezpečuje spolehlivost přenosu doplněním zprávy o přídavné znaky. Pomocí těchto prvků 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 obrázku 1.2. Takové systémy nazýváme FEC (angl. Forward Error Correction) neboli dopředná korekce chyb. Takové systémy jsou úspornější z hlediska přenosové rychlosti (šířky pásma dopředného kanálu), účinnost zabezpečení je ovšem menší.

Požadavky na vysoce spolehlivý přesnost dat vedly k doplnění schémata obecného komunikačního systému o člen zpětný kanál, jak je uvedeno na obrázku 1.3. Data jsou totiž obecně zabezpečena pouze detekčním kódem a zpětný kanál je schopen na základě tohoto výsledku vyslat povel k opakování přenosu. Zpětnovazební systémy jsou zkráceně nazývány ARQ (angl. Automatic Request for Repetition) neboli automatická žádost o opakování přenosu.



Obr. 1.3: Shannonovo schéma obecného komunikačního systému se zpětnovazebním kanálem.

Zpětnovazební koncepce jsou dvojího typu:

a) Systémy s rozhodovací zpětnou vazbou DFB
DFB (Decision Feedback) – rozhodovací zpětná vazba

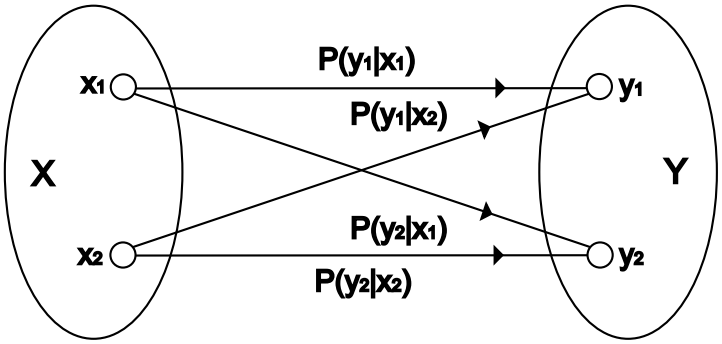
Dekodér kanálu v tomto případě vyhodnocuje věrnost jednotlivých slov ve zprávě s využitím detekčního kódu. Není-li zjištěna chyba, vyšle přijímač zpětným kanálem vysílači potvrzení ACK (angl. Acknowledgment). V opačném případě, tedy je-li zjištěna chyba, zažádá přijímač skrze zpětný kanál zasláním příkazu NACK (angl. Negative Acknowledgement) o opětovný přenos 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 volit druh kódu pečlivě s přihlédnutím k charakteru rozložení chyb.

b) 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 straně 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.

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 parametrům kanálů lze uvést několik následujících poznatků. **Bezchybový kanál** je případ, kdy informační prvek přijatého signálu vždy odpovídá témuž prvku signálu vyslaného. Opačným případem je tedy **chybový kanál**, kdy si informační prvky signálu na vstupu a výstupu ne vždy odpovídají. **Diskrétním kanálem bez paměti** je nazýván takový kanál, kde je výsledek přenosu znaku ze vstupu na výstup 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**.

Model diskrétního sdělovacího kanálu může 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 pravděpodobnostmi 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 obrázku 1.4 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 vyplývá, že vztahové závislosti (1.15) a (1.16) jsou platné pro vyslání a příjmu znaku a z těchto je také možné sestavit přímou matici kanálu (1.17): [2]



Obr. 1.4: Informační schéma binárního hlukového kanálu.

$$P(y_1|x_1) + P(y_2|x_1) = 1, \tag{1.15}$$

$$P(y_2|x_2) + P(y_1|x_2) = 1, \tag{1.16}$$

$$K_{XY} = \begin{pmatrix} P(y_1|x_1) & P(y_2|x_1) \\ P(y_1|x_2) & P(y_2|x_2) \end{pmatrix}. \tag{1.17}$$

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 (1.18) a (1.19): [2]

$$P(y_1) = P(x_1).P(y_1|x_1) + P(x_2).P(y_1|x_2), \tag{1.18}$$

$$P(y_2) = P(x_1).P(y_2|x_1) + P(x_2).P(y_2|x_2). \tag{1.19}$$

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 na výskyt symbolů pohlíželi 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 xi vyslán, pokud byl přijat znak y_j. K výpočtům těchto simultánních pravděpodobností slouží vztah (1.20) (kde $i, j \in \{1, 2\}$), ze kterého vyplývají i hledané pravděpodobnosti (1.21) (kde $i, j \in \{1, 2\}$), jejichž jmenovatel je řešen podle (1.18) a (1.19): [2]

$$P(x_i, y_j) = P(x_i).P(y_j|x_i) = P(y_j).P(x_i|y_j), \tag{1.20}$$

$$P(x_i|y_j) = \frac{P(x_i).P(y_j|x_i)}{P(y_j)}. \tag{1.21}$$

Po doplnění náležitých indexů lze sestavit matici kanálu, nyní jde ovšem o matici zpětnou (1.22) 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 (1.18) a (1.19) do (1.21) získáme konečné vzorce pro symetrický kanál (1.22) a (1.23). [2] Parametr P zde vyjadřuje hodnotu spolehlivosti a Q hodnotu nespolehlivosti.

$$K_{YX} = \begin{pmatrix} P(x_1|y_1) & P(x_2|y_1) \\ P(x_1|y_2) & P(x_2|y_2) \end{pmatrix}, \tag{1.22}$$

$$K_{XY} = \begin{pmatrix} \frac{1}{1+\frac{P(x_2).Q}{P(x_1).P}} & \frac{1}{1+\frac{P(x_2).P}{P(x_1).Q}} \\ \frac{1}{1+\frac{P(x_1).P}{P(x_2).Q}} & \frac{1}{1+\frac{P(x_1).Q}{P(x_2).P}} \end{pmatrix}. \tag{1.23}$$

1.2.7 Vzájemná informace

Teorie uváděné ve dřívějších bodech nevyjadřují reálné vlivy na přenosový kanál, který je podle obrázku 1.2 ovlivňován rušivými parametry různého charakteru. Vlivem šumu v průběhu přenosu sice dochází k poklesu množství informace, chybový

kanál ale zprávu doplňuje o takové množství dezinformace, že se pak entropie vstupní a výstupní zprávy jeví jako stejné. Tato informace je označována jako „vzájemná informace“, jejíž výpočet si lze usnadnit pomocí tzv. podmíněných a simultánních entropií.

Pro tyto úvahy i nadále poslouží model chybového kanálu uvedený na obrázku 1.4. Výskyt znaků na vstupu kanálu je zde opět popsán souborem vzájemně se vylučujících náhodných jevů X, výskyt znaků na výstupu obdobně souborem Y. V případě, že se jedná o kanál bezchybový, 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, jsme schopni vypočítat entropii těchto souborů. V případě statistické závislosti mezi těmito soubory je ale vhodné defininovat další druhy entropií.

Podmíněná entropie vstupního souboru X při známém výstupu y_j je dána obecným vztahem (1.24) [2], tedy úpravou vztahu pro průměrnou entropii:

$$H(X|y_j) = -\sum_j P(x_i|y_j)\log_2 P(x_i|y_j) \text{ (Sh/symbol)}. \tag{1.24}$$

$H(X|y_j)$ je neurčitost příjemce informace o tom, co je vysláno přes kanál ze vstupu, snížená o zjištění výstupu y_j. Pro kanál bezchybový je dokonce tato neurčitost nulová. Zprůměrováním entropie (1.24) pro všechny možné výstupy y_j dále docházíme ke vztahu (1.25) [2] pro tzv. podmíněnou entropii vstupu po čtení výstupu:

$$H(X|Y) = -\sum_j P(y_j)H(X|y_j) \text{ (Sh/symbol)}. \tag{1.25}$$

Při dalších úpravách výrazu (1.25), jako je dosazení (1.24) a použití vztahu (1.20), dostaneme výpočet entropie (1.26) [2], k čemuž je třeba znát prvky zpětné matice kanálu:

$$H(X|Y) = -\sum_i \sum_j P(x_i|y_j)\log_2 P(x_i|y_j) \text{ (Sh/symbol)}. \tag{1.26}$$

Podmíněná entropie vstupu po přečtení výstupu vyjadřuje průměrnou neurčitost o stavu vstupu po čtení výstupu, přičemž před přečtením byla neurčitost vstupu $H(X)$. Rozdílem těchto entropií získáme tzv. vzájemnou informaci ze vstupu na výstup (1.26). 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 (1.27): [2]

$$I(X,Y) = H(X) - H(X|Y) \text{ (Sh/symbol)}. \tag{1.27}$$

Podmíněná entropie výstupního souboru Y při známém vstupu x_i je dána obecným vztahem (1.28) [2], je zde tedy opět vidět obdobná úprava vztahu pro průměrnou entropii (1.9), jako v předešlém případě:

$$H(Y|x_i) = -\sum_j P(y_j|x_i)\log_2 P(y_j|x_i) \text{ (Sh/symbol)}. \tag{1.28}$$

$H(Y|x_i)$ je neurčitost příjemce informace o tom, co je přijato na výstupu, snížená o zjištění vyslání znaku x_i ze vstupu. Pro kanál bezchybový je dokonce tato neurčitost nulová. Zprůměrováním entropie (1.28) pro všechny možné vstupy x_i docházíme ke vztahu (1.29) [2] pro tzv. podmíněnou entropii výstupu po čtení vstupu.

$$H(Y|X) = -\sum_i P(x_i)H(Y|x_i) \text{ (Sh/symbol)}. \tag{1.29}$$

Při dalších úpravách výrazu (1.29) dosazením (1.28) a použitím vztahu (1.20) se dostáváme k výpočtu entropie (1.30) [2], k čemuž je třeba opět znát prvky zpětné matice kanálu:

$$H(Y|X) = -\sum_i \sum_j P(x_i|y_j)\log_2 P(y_j|x_i) \text{ (Sh/symbol)}. \tag{1.30}$$

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 (1.31) [2]. Podmíněnou entropii $H(Y|X)$ je tedy možné v tomto případě chápat jako průměrné množství informace, které se do výstupní zprávy dostalo vlivem rušivého působení kanálu, který však neměl vliv ve vstupní zprávě:

$$I(X,Y) = H(Y) - H(Y|X) \text{ (Sh/symbol)}. \tag{1.31}$$

Simultánní entropie vstupního a výstupního souboru by byla při nezávislosti vstupů a výstupů rovna součtu dílčích entropií vstupu a výstupu. Pro případ závislosti ovšem simultánní neurčitost klesá. Tento druh entropie je možné určit pomocí podmíněných entropií.

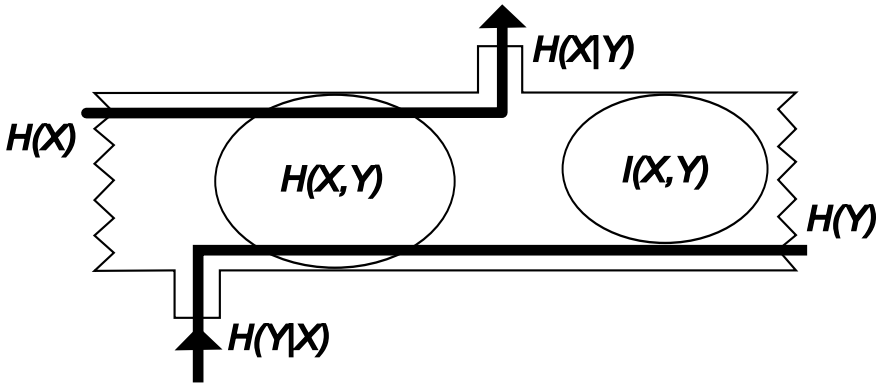
Neurčitost pozorovatele o stavech X a Y je možné snížit odečtením vstupu, tedy ziskem informace $H(X)$. Výsledkem je neurčitost o stavu Y (1.32) [2] za podmínky odečtení vstupu. Obdobně platí tato operace i pro případ stavu X, kdy byl odečten výstup (1.33) [2]. Následným odečtením těchto vztahů, tedy rovnic (1.32) a (1.33) je dán výsledek (1.34): [2]

$$H(X,Y) = H(X) - H(Y|X) \text{ (Sh/symbol)}, \tag{1.32}$$

$$H(X,Y) = H(Y) - H(X|Y) \text{ (Sh/symbol)}, \tag{1.33}$$

$$H(X) - H(X|Y) = H(Y) - H(Y|X) \rightarrow I(X,Y) = I(Y,X). \tag{1.34}$$

Číselná rovnost těchto dvou informací se promítá i do jejich názvu „vzájemná vstupně-výstupní informace“. Vzájemné vztahy mezi jednotlivými entropiemi dobře zobrazuje obrázek 1.5 a následující legenda v tabulce 1.2. Zde jsou mimo jiné zřejmé i dále uvedené nerovnosti (1.35): [2]



Obr. 1.5: Schéma informačních poměrů v chybovém kanálu.

$$MIN\{H(X), H(Y)\} \leq H(X,Y) \leq H(X) + H(Y). \tag{1.35}$$

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

Tab. 1.2: Legenda ke schematu informačních poměrů v chybovém kanálu.

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

2 PŘEHLED POUŽITÝCH ALGORITMŮ

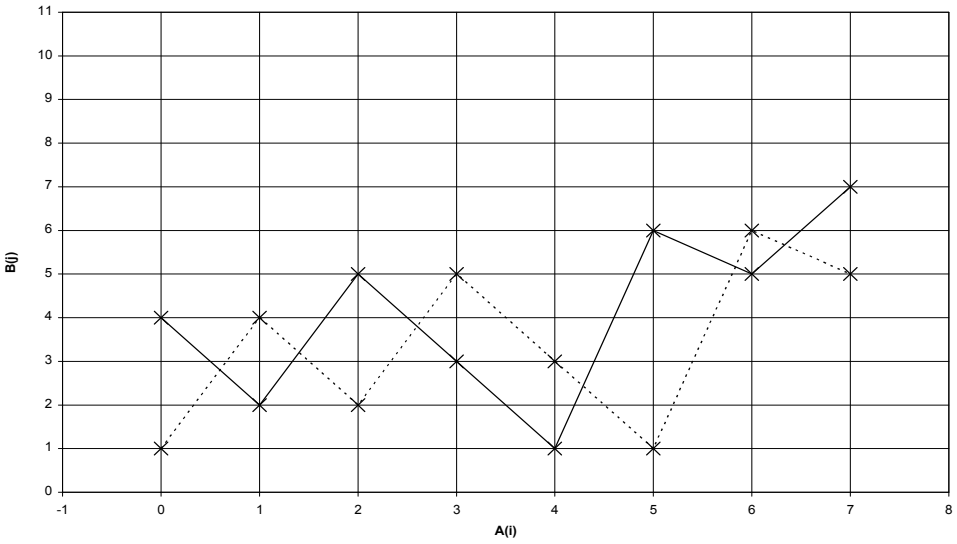
V této kapitole je podrobně rozebrán postup výpočtu třech algoritmů vyvinutých za účelem výpočtu vzájemné informace z časové řady. Konkrétně jde o Fraser-Swinneyho algoritmus, Dinh-Tuan-Phamův algoritmus a postup pro výpočet vzájemné informace pomocí adaptivního XY dělení.

2.1 Fraser-Swinneyho algoritmus

Princip Fraser-Swinneyho algoritmu (angl. The Fraser-Swinney algorithm) vychází z porovnávání dvojic časově omezených řad. Tyto řady mohou být buď naprosto rozdílné svým obsahem, při zachování stejné délky, případně se může jednat o řady obsahem stejné a jen v čase posunuté. Každá taková časově omezená řada obsahuje 2^n prvků. Příkladem mohou být např. prvky, uvedené v tabulce 1.1, které jsou následně vynášeny do dvojrozměrné souřadnicové roviny, s libovolnou volbou os, jako body, jak uvádí obrázek 2.1.

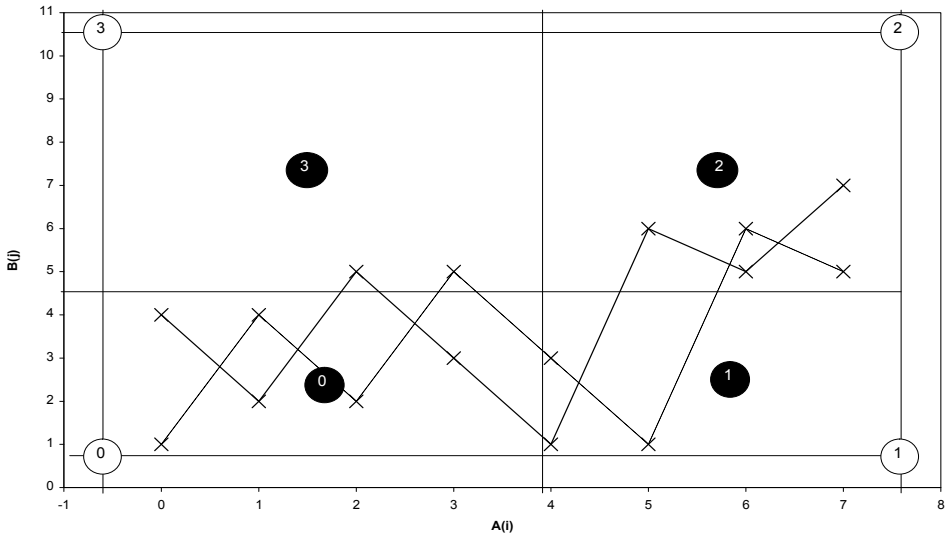
Tab. 2.1: Volba a chronologické řazení prvků dvou časově omezených řad

i	0	1	2	3	4	5	6	7
A(i)	4	2	5	3	1	6	8	7
B(i)	7	1	4	2	5	3	1	6



Obr. 2.1: Vynesení prvků časově omezených řad do souřadnicového systému.

Po vynesení prvků časově omezených řad do souřadnicového systému dochází k postupnému dělení výsledného rastru, vždy na čtyři díly (tzv. substruktury). Tento proces je znázorněn na obrázku 2.2. Je třeba, aby takto vzniklé bloky splňovaly určité podmínky a proto je dále ověřována tzv. platnost substruktury roviny. Jde o výpočty pomocí vztahů (2.3) [3] a (2.4) [3]. To se ovšem neobejde bez zjištění chybových parametrů, jejichž výpočet je řešen vztahy (2.1) [6] a (2.2) [6].



Obr. 2.2: Dělení rastru bodů na substruktury.

$$a_i \equiv N(R_{m+1}(K_m, i)), \tag{2.1}$$

$$b_{ij} \equiv N(R_{m+2}(K_m, i, j)), \tag{2.2}$$

$$\chi^2_3 = \left(\frac{16}{9}\left(\frac{1}{N}\right)\sum_{i=0}^{3}\left(a_i - \frac{N}{4}\right)^2\right) < 1{,}547, \tag{2.3}$$

$$\chi^2_{15} = \left(\frac{256}{225}\left(\frac{1}{N}\right)\sum_{i,j=0}^{3}\left(b_{ij} - \frac{N}{16}\right)^2\right) < 1{,}287. \tag{2.4}$$

Odtud $R_m(K_m)$ vyjadřuje element dělení, tedy díl (substrukturu) roviny a Km jeden z celkově možných 4^n indexů a s tím souvísí i dolní index, který určuje úroveň dělení rastru, se kterou daný vztah pracuje (tj. index 3 pro 0 až 3, tedy 4 pro každé substruktury, index 15 pro 0 až 15, resp. pro 16 substruktur). N je počet prvků (bodů) uvažované substruktury roviny. Za předpokladu, že obrazec nevyhovuje těmto podmínkám, tedy není splněna alespoň jedna z podmínek, je třeba rastr bodů dále dělit. Poměry jednotlivých obdélníků, popřípadě čtverců, mohou být libovolné. Lépe

je však, počítat s nerovnoměrným rozdělením, stále je ovšem nutné uvažovat celou plochu roviny, nikoli jen její část. Po každém dělení rastru je nutné opět provést ověření substruktury roviny. Cyklus trvá do doby, než je nalezeno takové rozdělení obrazce, pro jehož všechny díly jsou splněny obě podmínky (2.3) a (2.4).

Pro každé ověřování substruktury roviny po dělení rastru je dále počítána vzájemná informace (2.7) [6] v závislosti na splnění podmínek. Děje se tak, v případě nesplnění podmínek, podle rekurzivního vztahu (2.5) [6] a pokud podmínky splněny jsou, podle (2.6): [6]

$$F(R_m(K_m)) = N(R_m(K_m))\log(N(R_m(K_m))), \tag{2.5}$$

$$F(R_m(K_m)) = N(R_m(K_m))\log(4) + \sum_{j=0}^{3} F(R_{m+1}(K_m, j)), \tag{2.6}$$

$$I(S, Q) = (1/N_0)F(R_0(K_0)) - \log(N_0) \text{ bit}. \tag{2.7}$$

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 (2.6) lze dodat, 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

Jak již samotný název výpočetní metody napovídá, je zde podobně, jako v případě Fraser-Swinneyho algoritmu (kapitola 2.1) využit postup adaptivního dělení rastru. Tyto algoritmy jsou si svou podstatou podobné, rozdíl je ovšem v tom, že prostor je v případě této metody dělen v závislosti na stejném obsazení bodů. Hledání hranic jednotlivých substruktur se tak liší s rozdílnými vstupními daty.

Výpočet vzájemné informace by měl být ověřován testem předpokladu statistické nezávislosti (angl. null hypothesis of statistical independence). K tomu by měl navíc oddíl v rovině XY (použitý k výpočtu společného rozdělení pravděpodobnosti P_{XY}) splňovat Cochranovo kritérium očekávané E_{XY}. Konkrétně je vyžadováno splnění podmínky $E_{XY}(i, j) \geq 1$ pro všechny prvky oddílu a $E_{XY}(i, j) \geq 5$ pro nejméně 80 % prvků oddílu. V následujícím algoritmu je použito 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 podle očekávacího kritéria.

Nechť N_d označuje počet XY dvojic. N_Y 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.

2.2.1 Určení počtu prvků dělení osy

Nejprve je nutná volba rozsahu dělení na dané ose. Pro osu x jsou to tedy parametry $\mathrm{x_{min}}$ a $\mathrm{x_{min}}$. Po jejich stanovení je osa dělena 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 (2.8): [3]

$$P_X(i) = \frac{1}{N_E}. \tag{2.8}$$

Obdobně pak po stanovení $\mathrm{y_{min}}$ a $\mathrm{y_{min}}$, 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 (2.9): [3]

$$P_Y(j) = \frac{1}{N_E}. \tag{2.9}$$

Podle testu předpokladu statistické nezávislosti, je očekávané obsazení (i,j)-tého prvku při rozdělení roviny XY dáno matematickým vyjádřením (2.10): [3]

$$E_{XY}(i,j) = N_D P_X(i) P_Y(j) = \frac{N_D}{N_E^2}. \tag{2.10}$$

Odtud lze odvodit i vztah pro společné rozdělení pravděpodobnosti $P_{XY}(i,j)$ (2.11) a přihlédnout k podobnosti s výpočtem téhož parametru příbuzným Fraser-Swinneyho algoritmem. [6]

$$P_{XY}(i,j) = \frac{E_{XY}(i,j)}{N_D} \rightarrow P_{XY}(R_m(K_m)) = N R_m(K_m)/N_0. \tag{2.11}$$

Hodnota N_E je obvykle určena nalezením nejvyšší hodnoty, která přiřazuje $E_{XY}(i,j) \geq 5$ všem prvkům oddílu XY. Toto kritérium je tedy konzervativnější než Cochranovo kritérium (2.10), 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 (2.12): [3]

$$N_E \leq \left(\frac{N_D}{5}\right)^{\frac{1}{2}}. \tag{2.12}$$

2.2.2 Výpočet vzájemné informace

Po výpočtu pravděpodobností obsazení $P_{XY}(i,j)$ následuje kalkulace vzájemné výměny informace dle vzorce 2.13: [3]

$$I(X,Y) = \sum_{i=1}^{N_E}\sum_{j=1}^{N_E} P_{XY}(i,j)\log_2\left\{\frac{P_{XY}(i,j)}{P_X}(i).P_Y(j)\right\} \text{ bit.} \tag{2.13}$$

Za předpokladu, že N_D není přesně dělitelné N_E, tzn. pokud N_D není násobkem N_E a pak oddíly osy x a osy y nemají pravděpodobnosti přesně se rovnající zlomku 1 / N_E. Pak je třeba vzorec (2.13) zjednodušit na následující vztah (2.14): [3]

$$I(X,Y) = \sum_{i=1}^{N_E}\sum_{j=1}^{N_E} P_{XY}(i,j)\log_2\left\{N_E^2.P_{XY}(i,j)\right\} \text{ bit.} \tag{2.14}$$

Pokud je Cochranovo očekávací kritérium splněno (nulová hypotéza není zamítnuta), pak jsou obě sady dat statisticky nezávislé a lze tedy provést výpočty tímto algoritmem. Za těchto podmínek není vyžadován výkaz nenulových hodnot vzájemné informace. 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

Název algoritmu byl v této práci upraven podle jména autora, jelikož název oficiální je v doslovném překladu uváděn jako „Rychlý algoritmus odhadu vzájemné informace, entropie a výsledné funkce“. Autorem tohoto algoritmu je tedy vietnamský matematik Dinh Tuan Pham. Algoritmus k výpočtu vzájemné informace, obdobně jako v případě algoritmů předešlých, využívá mříže či rastru rozšířeného o jádro.

2.3.1 Výsledná funkce jako gradient entropické funkce

Nechť $\boldsymbol{Y}$ je náhodný vektor s prvky $Y_1, \ldots, Y_K$ a hustotou $p_{\boldsymbol{Y}}$. Množstevní funkce $\psi_{\boldsymbol{Y}}$ pro $\boldsymbol{Y}$ (označovaná i jako společná funkce $Y_1, \ldots, Y_K$ pro $\psi_{Y_1}, \ldots, \psi_{Y_k}$ je definována jako gradient $-\log p_{\boldsymbol{Y}}$. Může být vnímán jako gradient entropické funkce, v tom smyslu, že pro malý náhodný přírůstek $\partial \boldsymbol{Y}$ vektoru $\boldsymbol{Y}$ platí vzorec (2.15): [12]

$$H(\boldsymbol{Y} + \partial \boldsymbol{Y}) - H(\boldsymbol{Y}) \approx \mathrm{E}\left[\psi_{\boldsymbol{Y}}^T(Y)\partial \boldsymbol{Y}\right]. \tag{2.15}$$

Zde E označuje očekávání pozorovatele a $p_{\boldsymbol{Y}}$ je dále označením hustoty $Y_1, \ldots, Y_K$. Intuitivní důkaz zápisu vztahu (2.15) je uveden následujícím způsobem (2.16): [12]

$$\mathrm{E}\left[\log \frac{p_{\boldsymbol{Y}}(\boldsymbol{Y})}{p_{\boldsymbol{Y}}(\boldsymbol{Y} + \partial \boldsymbol{Y})}\right] + \mathrm{E}\left[\log \frac{p_{\boldsymbol{Y}}(\boldsymbol{Y} + \partial \boldsymbol{Y})}{p_{\boldsymbol{Y}+\partial \boldsymbol{Y}}(\boldsymbol{Y} + \partial \boldsymbol{Y})} + 1\right]. \tag{2.16}$$

Přitom vztah (2.16) lze zapsat i jako (2.17): [12]

$$\mathrm{E}\left[\log \frac{p_{\boldsymbol{Y}}(\boldsymbol{Y} + \partial \boldsymbol{Y})}{p_{\boldsymbol{Y}+\partial \boldsymbol{Y}}(\boldsymbol{Y} + \partial \boldsymbol{Y})} - \frac{p_{\boldsymbol{Y}}(\boldsymbol{Y} + \partial \boldsymbol{Y})}{p_{\boldsymbol{Y}+\partial \boldsymbol{Y}}(\boldsymbol{Y} + \partial \boldsymbol{Y})} + 1\right]. \tag{2.17}$$

Jelikož $\log x = x - 1 - (x-1)^2/2 + \ldots$, lze tedy očekávat, že hodnota tohoto výrazu bude vyšší, než $\partial \boldsymbol{Y}$ a proto jej lze zcela vypustit. Pomocí Taylorova rozvoje $\log p_{\boldsymbol{Y}}(\boldsymbol{Y} + \partial \boldsymbol{Y})$ lze dojít k požadovanému výsledku.

Z rovnic (2.16) a (2.17) připouští vzájemná informace $I(Y_1 \ldots Y_k)$ pro první řád následující rozvoj (2.18): [12]

$$\begin{gathered} I(Y_1, \ldots, Y_k) = \sum_{k=1}^{K} H(Y_k) - H(Y) I(Y_1 + \partial Y_1, \ldots, Y_k + \partial Y_k) \approx \\ \sum_{k=1}^{K} \mathrm{E}\left\{\left[\psi_{Y_k}(Y_k) - \psi_{k,Y_1,\ldots,Y_k}(Y_1, \ldots, Y_k)\right] \partial Y_k\right\}. \end{gathered} \tag{2.18}$$

Odtud $\psi_{k,Y_1,\ldots,Y_k}$ je k-tá složka této spojité funkce (angl. score function) ze spojité výsledné funkce $Y_1, \ldots, Y_K$. Funkce $\psi_k - \psi_{k,Y_1,\ldots,Yk}$ byly zavedeny dříve pod pojmem rozdílové výsledné funkce (angl. SDF-score diference functions). Mohou být chápány 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 $Y(1), \ldots, Y(p-1)$ připouští následující rozvoj prvního řádu (2.19): [12]

$$H[Y(p)|Y(1:p-1] = H[Y(1:p)] - H[Y(1:p-1)], \tag{2.19}$$

který lze rozšířit na

$$H[Y(p)+\partial Y(p)|Y(1:p-1)+\partial Y(1:p-1)]-H[Y(p)|Y(1:p-1]\approx \\ \mathrm{E}\left\{\psi^{T}_{Y(p)|Y(1:p-1)}[Y(1:p]\partial Y(1:p)\right\} \tag{2.20}$$

a odtud vyplývá, že

$$\psi_{Y(p)|Y(1:p-1)}=\psi_{Y(1:p)}-\begin{bmatrix}\psi_{Y(1:p-1)}\\0\end{bmatrix}. \tag{2.21}$$

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 $Y(1),\ldots,Y(p-1)$. Tato funkce bude pouze podmíněnou funkcí $Y(p)$ vzhledem k $Y(1),\ldots,Y(p-1)$.

2.3.2 Metody odhadu

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

a)Odhad entropie

K odhadu entropie je zapotřebí odhadu hustoty p_Y náhodného vektoru $\boldsymbol{Y}$ ze vzorku $\boldsymbol{Y}(1),\ldots,\boldsymbol{Y}(N)$, což lze realizovat dle vztahu (2.22): [12]

$$\hat{p}_Y(y)=\frac{1}{N}\sum_{n=1}^{N}\frac{\kappa[h^{-1}(x-Y(n))]}{\det h}=\hat{E}\kappa[h^{-1}(y-Y)]\det h, \tag{2.22}$$

kde κ označuje multivariační hustoty a h je parametr vyhlazení matice (mříže) [?]. Zde a v pokračování, notace $\hat{E}$ označuje operátor středního odběru vzorků. Přirozený odhad entropie $H(Y)$ je pak určen vzorcem (2.23): [12]

$$H(\boldsymbol{Y})=-\int \hat{p}_Y(\boldsymbol{y})\log\hat{p}_Y(\boldsymbol{y})d\boldsymbol{y}. \tag{2.23}$$

Tento vztah ovšem vyžaduje vícenásobnou integraci ve vícerozměrném prostoru. Proto je vhodné uvedenou integraci diskretizovat a přepsat na sumaci nějaké pravidelné mříže, jak uvádí (2.24): [12]

$$H(\boldsymbol{Y}) = -\sum_{i} \hat{p}_Y(\boldsymbol{gi}) \log \hat{p}_Y(\boldsymbol{gi}) \det \boldsymbol{g}. \tag{2.24}$$

Zde je proveden součet pro všechny vektory $\boldsymbol{i}$ s označenými celočíselnými součástmi a $\boldsymbol{g}$ je matice definující velikost a orientaci mříže. Za povšimnutí stojí možnost, vyhnout se integraci na základě odhadu entropie Y ve vztahu (2.25): [12]

$$H(\boldsymbol{Y}) = \frac{1}{N} \sum_{n=1}^{N} \hat{p}_Y[\boldsymbol{Y}(n)]. \tag{2.25}$$

Stejně tak, jako každý prvek $\hat{p}_Y[Y(n)]$ vyžaduje souhrn podmínek, znamená i tato metoda výpočet hodnoty řádu N^2. Náročnost výpočtu touto metodou je, díky volbě vhodné mříže a jádra, lineárně rostoucí s N. Dále je důležité, že umožňuje i eliminaci zkreslení.

Je nutné zvolit $\boldsymbol{g}$ úměrné $\boldsymbol{h}$, což ostatně dává smysl. Parametr $\boldsymbol{h}$ totiž ovlivňuje vyhlazení a hladší $\hat{p}_Y$ zajišťuje větší rozměr sítě, než je běžné. 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 $\boldsymbol{g} = \boldsymbol{h}$. Mříž tak totiž nevykazuje známky přílišné hrubosti. Uvažujeme-li případ, že $\boldsymbol{g} = \boldsymbol{h}/m$ pro nějaké celé číslo m, snižuje se tak rozměr mříže (rastru), ovšem dochází ke zvýšení výpočetní náročnosti faktorem m^K. Pro zjednodušení nebudeme uvažovat tuto volbu, přičemž navíc může být K průměrné velikosti, pro m^K, kde $m = 2$ příliš velké. Tímto lze dojít k odhadu dle vzorce (2.26): [12]

$$\hat{H}(\boldsymbol{Y}) = -\sum_{i} \hat{\pi}_Y(i)[\log \hat{\pi}_Y(i) - \log \det h]. \tag{2.26}$$

Odtud je $\hat{\pi}_{\boldsymbol{Y}}(\boldsymbol{i})$ dáno vztahem (2.27): [12] a může být považován za odhad pravděpodobnosti, při které náhodný vektor $\boldsymbol{h}^{-1}\boldsymbol{Y}$ náleží do buňky či oblasti vystředěné na jednotky objemu.

$$\hat{\pi}_Y(i) = \frac{1}{N} \sum_{n=1}^{N} \kappa[i - \boldsymbol{h}^{-1}\boldsymbol{Y}(n)] = \hat{E}\kappa(i - \boldsymbol{h}^{-1}\boldsymbol{Y}), \tag{2.27}$$

přičemž v praxi je multivariační jádro κ generováno z jednorozměrného jádra K podle dvou rozhodujících metod.

Tenzorový součin

$\kappa = \vartheta^{\times K}$, kde K-krát tenzorový součin ϑ je definován zápisem (2.28) [12] a odkud y_k označuje složky y

$$\vartheta^{\times K}(\boldsymbol{y}) = \sum_{k=1}^{K} \vartheta(y_k) \tag{2.28}$$

Sférická symetrie

$\kappa(\boldsymbol{y} = \mathrm{C}\vartheta(||\boldsymbol{y}||)$, kde C vyjadřuje normalizační konstantu, takže κ integruje k jedničce.

Gaussovo jádro vyhovuje jak metodě tensorového součinu, tak i metodě sférické symetrie. Nemá ovšem kompaktní podporu. Místo metody tenzorového součinu je tedy lepší použít základní drážkování (angl. cardinal spline) nebo třetí řád. Základní drážkování 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 je vhodné zvolit základní drážkování, jelikož jde o nejjednodušší postup s průběžným derivátem (potřebným k výpočtu gradientu), což je jednoznačně dáno zápisem (2.29) [12]. Krom toho, jsme tak již celkem blízko Gaussovy hustoty

$$\vartheta(u) = \begin{cases} 3/4 - u^2, & |u| \leq 1/2 \\ (3/2 - |u|)^2/2, & 1/2 \leq |u| \leq 3/2 \\ 0, & \text{mimo rozsah} \end{cases} . \tag{2.29}$$

Rychlý výpočet $\hat{\pi}_{\boldsymbol{Y}}$ vychází z faktu, že je hodnocena pravidelná mříž a uvažované jádro vychází z původního jádra, které zahrnuje násobky rastru. Například, pokud Y_k' je součástí h^{-1} a termín $\vartheta^{\times K}[\boldsymbol{i} - \boldsymbol{h}^{-1}\boldsymbol{Y}(n)]$ může být výsledek nenulový pouze v případech (2.30): [12]

$$i_k = \langle Y_k'(n) \rangle \text{ nebo } i_k = \langle Y_k'(n) \rangle \pm 1, \tag{2.30}$$

kde $k = 1, \ldots, K$. Odtud i_k je k-tý člen $\boldsymbol{i}$ a $\langle y \rangle$ označuje celé číslo nejblížší k y. Tak lze vcelku rychle spočítat $\hat{\pi}_{\boldsymbol{Y}}(\boldsymbol{i})$ následujícího algoritmu:

Prve je inicializován $\hat{\pi}_{\boldsymbol{Y}}(\boldsymbol{i})$ na hodnotu 0, pak jsou za n postupně dosazována celá čísla, tedy $n = 1, \ldots, N$ a dále se při výpočtu postupuje podle (2.31): [12]

$$\begin{gathered} \hat{\pi}_{\boldsymbol{Y}}\left[\langle Y_1'(n) \rangle + i_1, \ldots, \langle Y_K'(n) \rangle + i_P\right] = \\ \hat{\pi}_{\boldsymbol{Y}}\left[\langle Y_1'(n) \rangle + i_1, \ldots, \langle Y_K'(n) \rangle + i_P\right] + \\ \frac{1}{N} \prod_{k=1}^{K} \vartheta\left[i_k + \langle Y_k'(n) \rangle - Y_k'(n)\right], \end{gathered} \tag{2.31}$$

kde $i_k = -1, 0, 1$. Pro interval $u \in [-1/2, 1/2]$ platí rovnosti (2.32): [12]

$$\vartheta(u) = \frac{3}{4} - u^2, \ \vartheta(\pm 1 + u) = \frac{(1/2 + u)^2}{2}. \tag{2.32}$$

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

Funkce základní křivky disponují zajímavou vlastností zvanou rozdělení jednoty (angl. partition of unity), dané vztahy (2.33) a (2.34): [12]

$$\sum_{i=-\infty}^{\infty} \vartheta(u + i) \equiv u, \tag{2.33}$$

$$\sum_{i} \hat{\pi}_{\boldsymbol{Y}}(\boldsymbol{i}) = 1, \tag{2.34}$$

u kterého zanedbáváme u. Složka $\hat{\pi}_{\boldsymbol{Y}}(\boldsymbol{i})$ zde představuje diskrétní rozdělení pravděpodobnosti a odhad entropie $\hat{H}(\boldsymbol{Y})$ je entropií této distribuce plus termín $\log \det \boldsymbol{h}$. Tento odhad má ovšem malou vadu v tom, že jeho překlad není tak zcela neměnný. Přidáním konstanty k náhodnému vektoru $\boldsymbol{Y}$ se nemění jeho entropie, což je považováno v podstatě za klad. Proto je vhodné tento odhad pozměnit nejprve pomocí vystředění dat, která jsou výpočtově dána jako $\hat{\pi}_{\boldsymbol{Y}}(\boldsymbol{i})$ a k tomu uvažovat $Y'k(n)$, nikoli k-té složky od $\boldsymbol{h}^{-1}\boldsymbol{Y}(n)$, nýbrž od $\boldsymbol{h}^{-1}[\boldsymbol{Y}(n) - \bar{\boldsymbol{Y}}]$, kde $\bar{\boldsymbol{Y}} = \hat{E}\,\boldsymbol{Y}$ je označení vzorku průměrné hodnoty vektoru $\boldsymbol{Y}$.

b) Odhad vzájemné informace

Zřejmý způsob odhadu vzájemné výměny informací je rozdíl odhadu společné entropie a sumy odhadnutých marginálních entropií. Předem je ovšem nutné pro zrušení zkreslení zvolit $\boldsymbol{h}$ úhlopříčky s prvky diagonály $h_1, \ldots, h_K$, kde h_K je vyhlazovací parametr pro stanovení odhadu mezní hustoty Y_k. Pravděpodobnost $\hat{\pi}_{Y_k}(j)$ je potřebná dále k odhadu $\hat{H}(Y_k)$ a je dána vztahem (2.35): [12]

$$\hat{\pi}_{Y_k}(j) = \frac{1}{N} \sum_{n=1}^{N} \vartheta[j - Y'_k(n)] = \hat{E}\vartheta(j - Y'_k). \tag{2.35}$$

Tento výraz je možné dále přepsat pro složky vektoru $\boldsymbol{i}$ do následující podoby

$$\hat{\pi}_{Y_k}(j) = \sum_{\boldsymbol{i}:i_k=j} \hat{\pi}_{\boldsymbol{Y}}(\boldsymbol{i}). \tag{2.36}$$

Odhad vzájemné informace je pak dán zápisem (2.37): [12]

$$\hat{I}(Y_1, \ldots, Y_K] = \sum_i \hat{\pi}_{\boldsymbol{Y}}(\boldsymbol{i}) \log \frac{\hat{\pi}_{\boldsymbol{Y}}(\boldsymbol{i})}{\prod_{k=1}^{K} \hat{\pi}_{Y_k}(i_k)} \text{ bit}. \tag{2.37}$$

Od tohoto lze očekávat, že výchylka (angl. bias) v $\hat{\pi}_{\boldsymbol{Y}}(\boldsymbol{i})$ je více či méně ovlivňována prvky obsazenými v marginální pravděpodobnosti $\hat{\pi}_{Y_k}(i_k)$, neboť právě k pravděpodobnostem $\hat{\pi}_{\boldsymbol{Y}}(\boldsymbol{i})$ sahá jejich původ. Ještě důležitější je fakt, že pokud má vektor $\boldsymbol{Y}$ nezávislé nebo též samostatné složky $\hat{I}(Y_1, \ldots, Y_K)$, bude konvergovat k nule jako $n \to \infty$, bez ohledu na výběr $\boldsymbol{h}$, jelikož limit $\hat{\pi}_{\boldsymbol{Y}}(\boldsymbol{i})$ je očekávaný výsledek nezávislých náhodných proměnných, což se dále rovná součinu pravděpodobností. Tak je tedy možné zvolit poměrně velké $\boldsymbol{h}$ bez obav, že dojde k chybnému vlivu na $\hat{I}(Y_1, \ldots, Y_K)$ jako u nezávislého empirického kritéria.

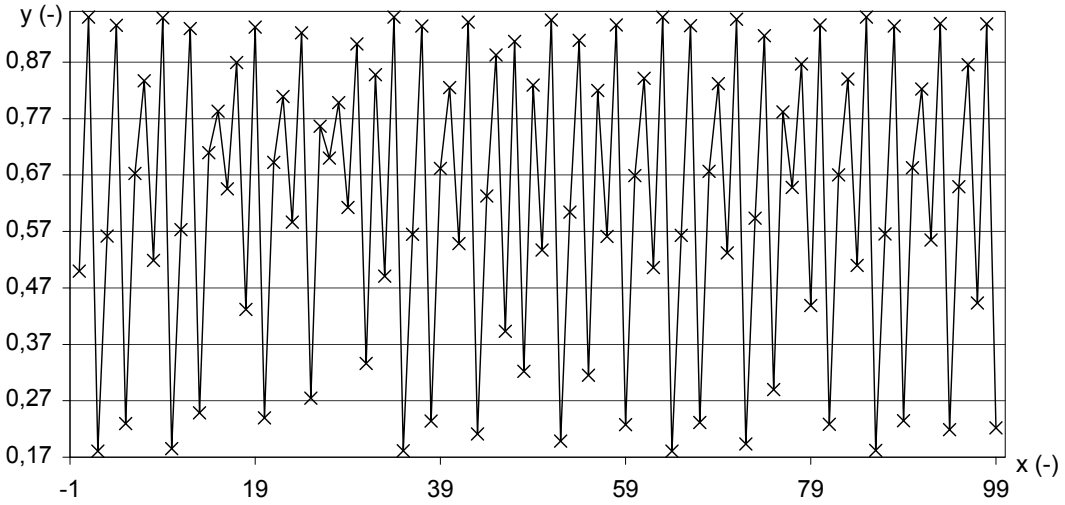
3 ANALYZOVANÁ DATA

Pro generování dat, určených k následné analýze výše popsanými algoritmy, lze v podstatě použít jakýkoli funkční předpis. Cílem je ovšem vybrat takovou funkci, která vychází z oboru teorie chaosu. Tento obor se totiž zabývá chováním jistých nelineárních dynamických systémů, které (za jistých podmínek) vykazují jev, známý jako deterministický[1] chaos. Rösslerův atraktor, který byl pro tento účel vybrán, splňuje tuto definici a je pro něj znám průběh vzájemné informace. Jeho rovnice pro dvourozměrnou soustavu souřadnic má tvar (70):

$$x(n+1) = 3{,}8 \cdot x(n) \cdot (1 - x(n)) \text{ pro } x(0) = 0{,}5 \tag{3.1}$$

Přesnost výpočtu vstupních dat lze s výhodou měnit jejich cíleným zaokrouhlením v celém rozsahu, vždy ke stejnému desetinnému místu. I přes různorodou odchylku, která se vlivem tohoto postupu ve výsledku objeví, se graf funkce Rösslerova atraktoru významně nemění. Důsledky jsou ovšem citelné ve výpočtu vzájemné informace. Názorný příklad pro srovnání odchylek maximální a minimální přesnosti je uveden v příloze A. Zde vypočtená distance, vzniklá rozdílem zaokrouhlených čísel, se pohybuje v intervalu od 0 do 0,5 a parametr p vyjadřuje stupeň přesnosti (číslo udávající posun v řádech při zaokrouhlování).

Následující obrázek 3.1 podává důkaz, že jde o chaotickou funkci a její body nemají zdánlivě nic společného.



Obr. 3.1: Zobrazení Rösslerova atraktoru pro 100 bodů.

[1] Deterministický v tom smyslu, že je dobře definovaný a neobsahuje žádné náhodné parametry.

4 PROGRAM PRO ANALÝZU ALGORITMŮ

Nadcházející řádky jsou komentářem i návodem k použití programu, který je součástí této práce. Program slouží k aplikaci, analýze a porovnání vlastností předdefinovaných algoritmů.

V programu jsou implementovány tyto algoritmy:

- Učebnicový výpočet,
- Fraser-Swinneyho algoritmus,
- Výpočet vzájemné informace pomocí adaptivního XY dělení.

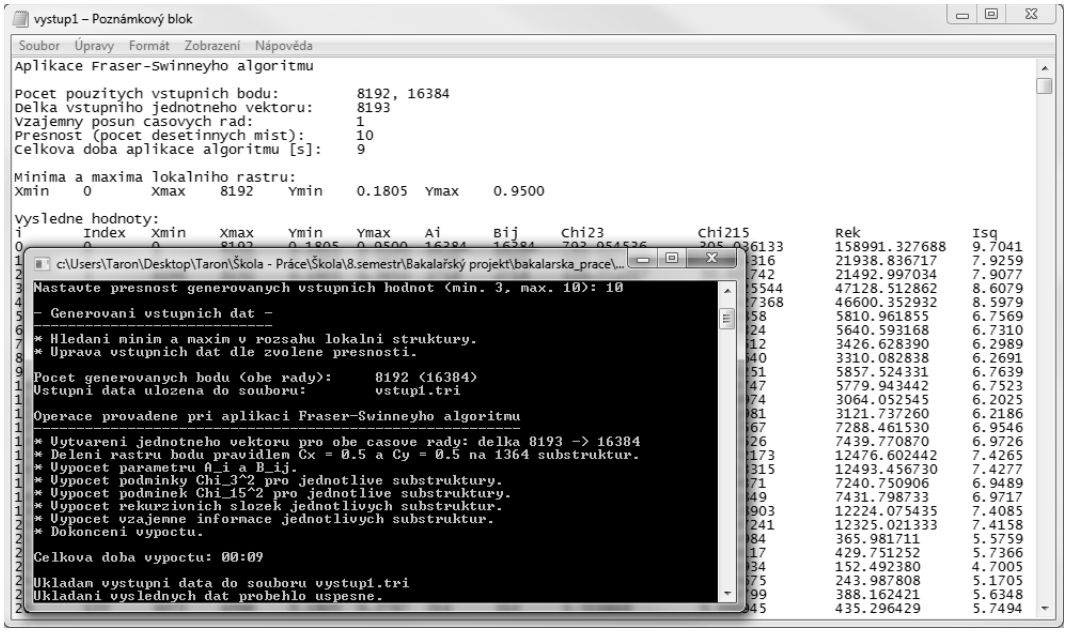
Vstupní data jsou reprezentována body, generovanými rovnicí Rösslerova atraktoru. Tato vstupní data jsou okamžitě ukládána do jednoho z předdefinovaných souborů (v závislosti na výběru algoritmu) pro možnost případného importu dat do některého z tabulkových editorů (např. Microsoft Excel, OpenOffice Calc).

4.1 Instalace a obsah adresáře

Adresář analyzačního programu „mutual_information_calc“obsahuje dva podadresáře. V prvním z nich je uložena zkompilovaná verze programu, tedy soubor s příponou EXE, ve druhém pak zdrojový kód pro možnost dalších modifikací. Popis jednotlivých souborů podadresáře se zdrojovým kódem udává tabulka 4.1. Jako vývojové prostředí, použité k naprogramování aplikace, byl zvolen Microsoft Visual C++ 2008 Express Edition. V případě, že zkompilovaný program při spuštění vykáže chybu, je třeba jej znova přeložit vhodným překladačem.

4.2 Vizuální podoba programu

Aplikace se nezabývá grafickými výstupy, prostředí je totiž výhradně konzolové a tedy i výstupy jsou textové. Okno programu reaguje na předdefinované klávesové zkratky a veškerá vstupní i výstupní data jsou ukládána do externích textových souborů pro možnost následného zpracování, ať už pro vykreslení grafů nebo jinou práci s daty. Fotografie programu i s textovým výstupem do externího souboru jsou k vidění na obrázku 4.1.



Obr. 4.1: Ukázka textových výstupů programu pro analýzu algoritmů.

4.3 Procesní posloupnost programu

Při spuštění programu je vyvoláno 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. Po volbě výpočetního mechanismu jsou obvykle vyžadovány další vstupní parametry. Generované body jsou okamžitě ukládány do jednoho z předdefinovaných textových souborů v závislosti na výběru algoritmu. Během provádění algoritmu je měřen časový úsek, po který výpočet výstupních hodnot trvá. Toto měření probíhá bez ohledu na momentální zatížení procesoru počítače, údaj tedy není pro průměrné vytížení procesoru bez zátěže výpočtem dále přepočítáván. Z toho vyplývá, že při běhu programu je doporučováno vypnout nepoužívaná okna, aby nebyl procesor zbytečně zatěžován. Tento parametr má vypovídací schopnost pro následné porovnání rychlosti výkonu jednotlivých metod. Podobně jako data vstupní, i ta výstupní jsou ukládána do textového souboru s přiděleným názvem, jak udává 4.1. Po výkonu algoritmu se program uživatele znova dotazuje na analýzu dat algoritmem jiným, dokud uživatel běh programu neukončí. Soubory dat je ovšem zapotřebí zálohovat v době mezi jednotlivými analýzami (s rozdílnými vstupními parametry), jinak může dojít ke ztrátě již uložených dat přemazáním.

Tab. 4.1: Obsah adresáře „mutual_information_calc“

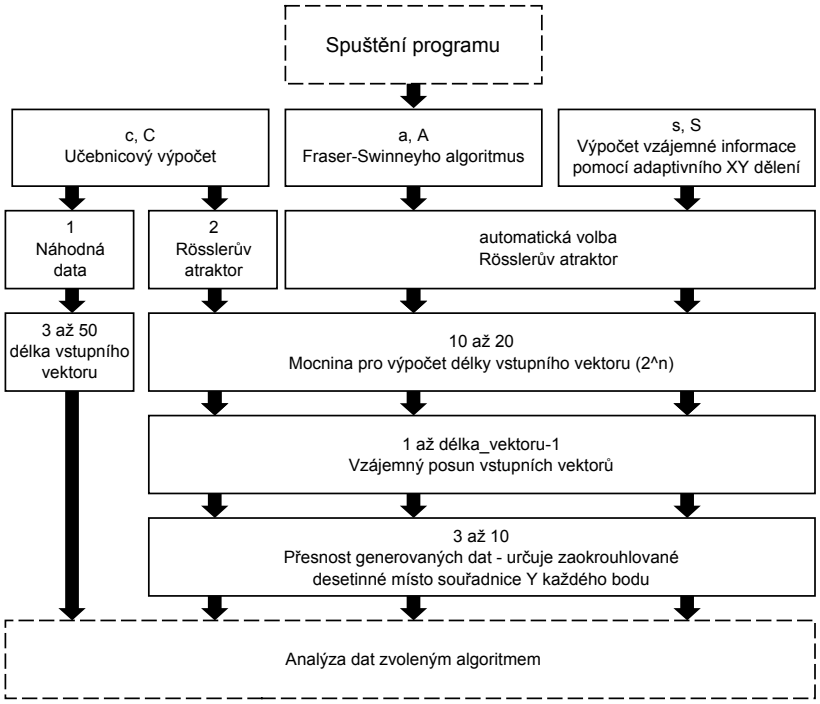
Název souboru	**Obsah souboru**
mutual_info_calc.cpp	uživatelské prostředí programu a předvolby
mutual_info_calc.sln	parametry pro otevření projektu ve vývojovém prostředí
mutual_info_calc.vcproj	parametry pro modifikace projektu ve vývojovém prostředí typu Visual Studio
stdafx.cpp	důležité globálních proměnné, struktury a pomocné funkce
stdafx.h	výpočetní funkce (zdrojové kódy výpočetních algoritmů)
targetver.h	specifikace pro operační systémy (soubor byl vytvořen vývojovým prostředím)
Textové soubory generované při chodu programu	
vstup1.tri	vstupní data pro analýzu Fraser-Swinneyho algoritmem
vstup2.tri	vstupní data pro výpočet vzájemné informace pomocí adaptivního XY dělení
vstup3.tri	vstupní data pro učebnicový výpočet
vystup1.tri	výstupní data výstupní data Fraser-Swinneyho algoritmu
vystup2.tri	výstupní data výpočtu vzájemné informace pomocí adaptivního XY dělení
vystup3.tri	výstupní data učebnicového výpočtu

4.4 Ovládání programu

K obsluze programu je používána pouze klávesnice, a to konkrétně k zadávání vstupních (zpravidla číselných) parametrů a k průchodu rozcestníky. Následující obrázek 4.2 pomáhá k urychlení obsluhy programu. Uživatel bývá aplikací dotazován na textové (v rámci rozcestníku), ale i na číselné parametry, které jsou po většinou požadovány pro výběr a nastavení parametrů vstupních dat před jejich analýzou implementovaných algoritmů.

4.5 Výstupní data

Výstupními daty programu jsou výsledky analýzy vstupních dat jedním nebo více předdefinovanými algoritmy. Tato jsou ukládána do textových souborů s koncovkou TRI bez speciálních znaků a s desetinnými tečkami (ne čárkami). Před dalším použitím těchto dat je tedy nutné pro české lokalizace programů (např. tabulkových



Obr. 4.2: Orientační strom voleb programu.

editorů) zaměnit desetinné tečky za čárky.[1] Pro případ několikanásobné analýzy jedním algoritmem je ovšem vhodné opět připomenout možnost ztráty výstupních dat, data je totiž třeba zálohovat.

[1]Klávesovou zkratkou Ctrl+H lze v programech, jako je Poznámkový blok nebo Microsoft Excel, vyvolat funkci „Nahradit“

5 PROGRAM PRO ANALÝZU ALGORITMŮ

Aplikované algoritmy se vzájemně liší svou implementační náročností, rychlostí výpočtu, i jeho přesností. To jsou také tři nejdůležitější parametry, které o charakteru těchto výpočetních metod mnoho vypovídají.

Pro generování vstupních dat je ve většině případů programem požadován parametr n, který slouží jako exponent k výpočtu množství generovaných bodů rovnicí Rösslerova atraktoru. Povolený rozsah pro proměnnou n je od 10 po 20, což odpovídá 1 024 až 1 048 576 bodům. Tento údaj je dále násoben dvěma, jelikož je počítáno se dvěma vektory (časovými řadami) nehledě na jejich vzájemný posun.

5.1 Učebnicový výpočet

Tato metoda je aplikací základního postupu pro výpočet vzájemné informace, pocházejícího z učebních materiálů. Výstupem je pouze jeden číselný údaj a proto je lepší tyto hodnoty graficky srovnávat s výstupy ostatních algoritmů, pokud je vyneseno více výsledků pro aplikaci každého z algoritmů.

Algoritmus nejprve zjistí abecedy (množiny symbolů) obou vstupních vektorů ($\boldsymbol{X}$ a $\boldsymbol{Y}$), resp. roztřídí jednotlivé symboly podle pravděpodobnosti jejich obsazení v celé délce obou vektorů. Z takto získaných údajů dále podle vztahu (1.9) počítá průměrnou entropii pro oba vektory $H(\boldsymbol{X})$ a $H(\boldsymbol{Y})$. Poté jsou v závislosti na (uživatelem zvoleném) vzájemném posunu obou vektorů hledány společné dvojice, resp. společná abeceda obou vektorů. Tyto hodnoty jsou použity pro výpočet společné entropie $H(\boldsymbol{X}, \boldsymbol{Y})$ obou vektorů opět dle předpisu (1.9). Posledním krokem je dosazení do následujícího vztahu (5.1):

$$I(\boldsymbol{X}, \boldsymbol{Y}) = H(\boldsymbol{X}) + H(\boldsymbol{Y}) - H(\boldsymbol{X}, \boldsymbol{Y}) = I(\boldsymbol{Y}, \boldsymbol{X}) \text{ bit.} \tag{5.1}$$

Jde o metodu s nízkou implementační složitostí, která sází spíše na přesnost výpočtu, než na jeho rychlost pro větší objem vstupních dat. Výpočet je značně zpomalený vlivem trojnásobného hledání abecedy a pravděpodobností výskytu symbolů (pro oba vektory samostatně i dohromady), kdy je zapotřebí porovnat každý prvek řady s každým. Urychlení je možné dosáhnout zajištěním již započtených symbolů řady tak, aby při procházení pole nedocházelo k jejich opětovnému zařazení do porovnávacího procesu. Při tomto řazení se např. pro deset prvků sníží počet operací z obvyklých 90 (v aplikaci použitý algoritmus) na pouhých 20 operací. Takto lze uvažovat průměrné zrychlení metody přibližně o polovinu. Metodu blíže popisuje obrázek 5.1

Časová řada										Symbol	Výskyt	Počet kroků
①	5	5	①	5	2	3	5	3	5	1	2/10	9
~~1~~	⑤	⑤	~~1~~	⑤	2	3	⑤	3	⑤	5	5/10	7
~~1~~	~~5~~	~~5~~	~~1~~	~~5~~	②	3	~~5~~	3	~~5~~	2	1/10	3
~~1~~	~~5~~	~~5~~	~~1~~	~~5~~	~~2~~	③	~~5~~	③	~~5~~	3	2/10	1
Celkem										4	10/10	20

Obr. 5.1: Proces zjišťování abecedy vektoru.

5.2 Fraser-Swinneyho algoritmus

Algoritmus pracuje na principu dělení rastru bodů v souřadnicovém systému o dvou osách. Po vygenerování vstupních dat tedy program průchodem tohoto pole zjistí minimální a maximální souřadnice pro obě osy a tak omezí prostor, potřebný k následujícím výpočetním operacím. V závislosti na kvantitě vstupních bodů program dále vybere nejvhodnější koeficient pro dělení tohoto rastru. Pole těchto hodnot bylo pevně stanoveno na základě externího výpočtu pro minimální předpokládaný počet bodů v entitě vzniklé dělením rastru. Tento výpočet je ovšem uvažován pro rovnoměrné rozložení bodů, a proto je nutné počítat s větší tolerancí. K ujasnění výpočtu koeficientů slouží obrázek 5.2, resp. export snímku výpočetní tabulky z programu Microsoft Excel.

		exponent										
úroveň dělení	počet substruktur	10	11	12	13	14	15	16	17	18	19	20
		1024	2048	4096	8192	16384	32768	65536	131072	262144	524288	1048576
		počet vstupních bodů										
1	4	256	512	1024	2048	4096	8192	16384	32768	65536	131072	262144
2	16	64	128	256	512	1024	2048	4096	8192	16384	32768	65536
3	64	16	32	64	128	256	512	1024	2048	4096	8192	16384
4	256	nedělit	nedělit	16	32	64	128	256	512	1024	2048	4096
5	1024	nedělit	nedělit	nedělit	nedělit	16	32	64	128	256	512	1024
6	4096	nedělit	nedělit	nedělit	nedělit	nedělit	nedělit	16	32	64	128	256
7	16384	nedělit	nedělit	nedělit	nedělit	nedělit	nedělit	nedělit	nedělit	16	32	64
8	65536	nedělit	nedělit	nedělit	nedělit	nedělit	nedělit	nedělit	nedělit	nedělit	nedělit	16
9	262144	nedělit	nedělit	nedělit	nedělit	nedělit	nedělit	nedělit	nedělit	nedělit	nedělit	nedělit
10	1048576	nedělit	nedělit	nedělit	nedělit	nedělit	nedělit	nedělit	nedělit	nedělit	nedělit	nedělit
Optimální počet substruktur		84	84	340	340	1364	1364	5460	5460	21844	21844	87380
Povolený počet úrovní dělení		3	3	4	4	5	5	6	6	7	7	8

Minimální počet bodů v substruktuře	Vzájemný posun časových řad <0,1>
10	0

Obr. 5.2: Snímek tabulky pro výpočet koeficientů dělení.

V následujícím kroku je provedeno jednorázové dělení rastru bodů na menší části (nebo také substruktury, prvky či entity dělení), jejichž počet je dán výše zmíněným výběrem koeficientu. Aplikovaný poměr dělení osy může být nestejnoměrný (1:4, 3:4

a jiné), při aplikaci algoritmu ovšem program uvažuje definovaný poměr 1:1, tedy dělení na poloviny. Dále jsou pro každou substrukturu z lokálního rastru vybrány body, které leží uvnitř prostoru vymezeného hraničními souřadnicemi substruktury. Zde se lze opět setkat se zpomalením výpočtu při srovnávání mezí konkrétní substruktury s každým prvkem v rastru a řešení na obrázku 5.1 může být tedy po jisté modifikaci platné i pro tento případ. Tímto postupem jsou získány parametry a_i (2.1) a b_{ij} (2.2) pro ověření podmínek platnosti substruktury χ^2_3 (2.3) a χ^2_{15} (2.4) a bezprostředního výpočtu hodnoty vzájemné informace na základě jejich splnění.

Algoritmus je méně časově náročný než „učebnicový výpočet“ uvedený výše. Grafická závislost tohoto parametru je nelineární a v logaritmickém měřítku má skokový charakter, což je způsobeno rozdílným dělením lokálního rastru v závislosti na počtu vstupních bodů. Pokud má být ovšem „učebnicový výpočet“ považován kvůli svým referencím za přesný, lze říci, že výpočet vzájemné informace Fraser-Swinneyho algoritmem vykazuje odchylky v řádech desetin Sh. Nejvyšší odhadovaná odchylka je přibližně rovna 0,19795 Sh pro $2 \cdot 2^{20}$ (cca 2,1 milionu) vstupních bodů. Úroveň implementační složitosti algoritmu je v porovnání s ostatními v práci uvažovanými středně složité.

Grafická vyobrazení, rekonstruovaná z výstupů programu jsou uvedena v příloze B. Obrázek B.1 znázorňuje způsob dělení lokálního rastru bodů Fraser-Swinneyho algoritmem pro vektory délky 1024 bodů se vzájemným posunem $dif = 1$. Horizontální a vertikální čáry jsou mezemi jednotlivých substruktur, čáry diagonální jsou pouze přechody mezi jednotlivými body a pro vypovídací schopnost grafu jsou bezvýznamné.. Následující obrázek B.2 zachycuje výsledné hodnoty vzájemné informace při aplikaci algoritmu na 4096 bodů (resp. 8192 pro oba vstupní vektory) pro minimální ($prec = 3$) a maximální ($prec = 10$) nastavitelnou přesnost vstupních dat. Je zde jen drobná odchylka, která pro $prec = 3$ zajišťuje spojitější průběh. Ke snižování přesnosti vstupních dat dochází zaokrouhlením souřadnice Y všech bodů k určitému desetinnému místu a to má za následek seskupování těchto bodů vlivem . Maximální odchylka těchto dvou průběhů je rovna 0,0364 Sh. Při vzájemném posunu vstupních vektorů (časových řad) dochází k posunu hodnot vzájemné informace, jak je uvedeno na obrázku B.3 pro 1024 bodů (resp. 2048 pro oba vstupní vektory). Pro parametr posunu platí rozsah hodnot $dif \in \langle 1; size - 1 \rangle$, kde $size$ vyjadřuje délku jednoho vektoru.

Přesnost výpočtu může být dále ovlivněna během zjišťování počtu bodů v konkrétní substruktuře (parametry a_i a b_{ij}). Může totiž dojít k započtení jednoho bodu vícekrát, pokud leží na sousední hranici dvou a více substruktur.

5.3 Výpočet vzájemné informace pomocí adaptivního XY dělení

Ve vztahu pro výpočet vzájemné informace touto metodou (2.13) je znatelná podobnost s předpisem pro průměrnou entropii (1.9), princip metody se ovšem ubírá opět směrem k dělení rastru bodů. Výstupem této metody je opět pouze jeden číselný údaj a proto platí při grafickém srovnávání totéž, jako v případě učebnicového výpočtu.

exp	počet bodů		Optimální	Optimální	1	2	3	4	5	6	7	8	9	10	i. d.
	jeden	oba	počet	index	2	4	8	16	32	64	128	256	512	1024	p. d.
	vektor	vektory	substruktur	dělení	4	16	64	256	1024	4096	16384	65536	262144	1048576	p. s.
10	1024	2048	8	4	512	128	32	8	nedělit	nedělit	nedělit	nedělit	nedělit	nedělit	M. p.
11	2048	4096	16	4	1024	256	64	16	nedělit	nedělit	nedělit	nedělit	nedělit	nedělit	
12	4096	8192	8	5	2048	512	128	32	8	nedělit	nedělit	nedělit	nedělit	nedělit	5
13	8192	16384	16	5	4096	1024	256	64	16	nedělit	nedělit	nedělit	nedělit	nedělit	
14	16384	32768	8	6	8192	2048	512	128	32	8	nedělit	nedělit	nedělit	nedělit	
15	32768	65536	16	6	16384	4096	1024	256	64	16	nedělit	nedělit	nedělit	nedělit	
16	65536	131072	8	7	32768	8192	2048	512	128	32	8	nedělit	nedělit	nedělit	
17	131072	262144	16	7	65536	16384	4096	1024	256	64	16	nedělit	nedělit	nedělit	
18	262144	524288	8	8	131072	32768	8192	2048	512	128	32	8	nedělit	nedělit	
19	524288	1048576	16	8	262144	65536	16384	4096	1024	256	64	16	nedělit	nedělit	
20	1E+06	2097152	8	9	524288	131072	32768	8192	2048	512	128	32	8	nedělit	

Obr. 5.3: Snímek tabulky koeficientů dělení.

Náročnost implementace metody je oproti ostatním aplikovaným algoritmům nejsložitější. K dělení rastru bodů totiž dochází s úmyslem, aby každá dělením vzniklá entita obsahovala stejný počet bodů. Pro aplikaci metody jsou ovšem uvažovány dvě časové řady stejné délky, které se překrývají. Proto je hustota bodů v intervalu $\langle dif; size + dif \rangle$ osy X dvojnásobná. Odtud dif opět značí vzájemný posun a $size$ délku jedné časové řady. Lépe je tedy vstupní body seřadit do pracovního pole a s dělením v ose X postupovat po pravidelných krocích. Dle zadaného exponentu je vypočteno množství vstupních bodů a na‘základě tohoto údaje program načte nejvhodnější koeficient dělení rastru, který je stejný pro obě osy. Pole koeficientů bylo pevně stanoveno na základě externího výpočtu pro minimální počet bodů v entitě vzniklé dělením rastru. K ujasnění výpočtu koeficientů slouží obrázek 5.3, resp. export snímku výpočetní tabulky z programu Microsoft Excel. Odtud jednotlivé zkratky vyjadřují:

- i. d. – index dělení,
- p. d. – počet prvků dělení osy,
- p. s. – počet substruktur,
- m. p. – minimální počet bodů v substruktuře.

Počet bodů v každé substruktuře je zde dán buňkami diagonály výrazů „nedělit“ (např. 8 pro $exp = 10$ apod.), což jsou buňky, kde došlo k nesplnění podmínky minimálního předpokládaného počtu bodů v substruktuře. Touto volbou koeficientů dělení je současně zaručena platnost Cochranova kritéria i podmínky (2.12). Po rozdělení osy X je pracovní pole s body opět přeskupeno. Nyní ovšem už ne v celém rozsahu, ale jen v rámci rozsahů souřadnice prvků dělení osy X. Tím se předchází zbytečným krokům při nadcházejícím hledání hranic substruktur při dělení osy Y. Ke zjištění mezí jednotlivých substruktur dochází opět postupným průchodem pracovního pole postupným krokováním, resp. inkrementací ukazatele pole o předpokládaný počet bodů, jelikož každý bod je reprezentován jedním řádkem. Výpočet pravděpodobností obsazení substruktur a kýžené vzájemné informace je již celkem jednoduchý.

Ze všech tří aplikovaných algoritmů je tento nejméně časově náročný, jelikož si největší díl trvání výpočtu bere vzestupné seřazování pracovního pole a to lze realizovat rychlým třídicím algoritmem (angl. quicksort). Z hlediska přesnosti výpočtu algoritmus pro vstupní data, daná Rösslerovým atraktorem, vykazuje zcela jiné hodnoty, než ostatní dvě metody. To je pravděpodobně způsobeno charakterem vstupních dat v tom smyslu, že žádná ze substruktur neobsahuje dva a více vstupních bodů. Řešení tohoto problému může být v dělení lokálního rastru bodů na méně substruktur nebo naopak v úvaze většího kvanta vstupních bodů. Tím se zvyšuje pravděpodobnost nalezení více stejných bodů pro jednu substrukturu. Na hodnotu vzájemné informace počítané tímto algoritmem tedy nemá při uvažovaných vstupních datech (Rösslerův atraktor) vliv změna přesnosti ani vzájemný posun vektorů.

Obrázek C.1 uvedený v příloze C znázorňuje způsob dělení lokálního rastru tímto algoritmem pro vektory délky 1024 bodů se vzájemným posunem $dif = 100$. Horizontální a vertikální čáry jsou mezemi jednotlivých substruktur, čáry diagonální jsou pouze přechody mezi jednotlivými body a pro vypovídací schopnost grafu jsou bezvýznamné.

6 ZÁVĚR

Ve své závěrečné bakalářské práci jsem provedl studii teorie informace se zaměřením na vzájemnou informaci a tří algoritmů pro výpočet tohoto parametru. V práci jsou podrobněji rozepsány tři takové algoritmy a dva z nich (Fraser-Swinneyho algoritmus a algoritmus pro výpočet vzájemné informace pomocí adaptivního XY dělení) jsou implementovány v programu, který je součástí práce. Třetím aplikovaným algoritmem je způsob výpočtu vzájemné informace, jak je znám z výukových materiálů. Tento byl pojat ve stínu svých referencí za nejpřesnější z uvedených metod a tedy za prostředek srovnávání. Výpočetní metoda, jejímž autorem je vienamský matematik Dinh Tuan Pham, je probrána pouze teoreticky, svou implementační náročností totiž přesahuje mé programátorské schopnosti. Implementované metody jsou v práci posouzeny kromě implementační náročnosti i podle rychlosti a přesnosti výpočtu.

Podle implementační náročnosti lze v práci probrané a aplikované algoritmy seřadit následovně:

- Učebnicový výpočet,
- Fraser-Swinneyho algoritmus,
- Výpočet vzájemné informace pomocí adaptivního XY dělení,
- Dinh-Tuan-Phamův algoritmus.

Aplikace byla testována na třech počítačích o rozdílném výpočetním výkonu a jejich konkrétní konfigurace jsou uvedeny v tabulce 6.1. Výsledky testů rychlosti výpočtu jsou shrnuty ve snímcích tabulek z programu Microsoft Excel na obrázcích D.1, D.2 a D.3 a v souhrnném grafu D.4. Zde jsou jednotlivé metody reprezentovány stejným typem čáry a konkrétní počítač změnou typu bodu. Tak lze zjistit, že nejrychlejším výpočetní metodou je adaptivní XY dělení a za ní následuje Fraser-Swinneyho algoritmus.

Tab. 6.1: Konfigurace použitého technického vybavení.

Číslo počítače	**1.**	**2.**	**3.**
Procesor (CPU)	Intel Celeron 420	Intel Xeon	Intel Core2Duo
Takt procesoru	1,6 GHz	2 x 2,3 GHz	2 x 1,86 GHz
Paměť RAM	1 GB	2 GB	1 GB
Vytížení CPU – bez zátěže	27 %	0 %, 0 %	0 %, 5 %
Vytížení CPU – při výpočtu	100 %	85 %, 18 %	90 %, 25 %

Tento závěr je ovšem platný jen v případě použití rychlého seřazování (angl. quicksort). Tento třídicí algoritmus se mi ovšem v rámci aplikace podařilo bezchybně zprovoznit jen pro vektory o délkách 4096 (2^{13}) bodů a menších. Program vykazoval chybu, kterou se mi vyřešit nepodařilo. Zdrojový kód třídicího algoritmu je ovšem v programu ponechán a z přiloženého zdrojového kódu je tak možné po „odkomentování“ příslušné části kódu metodu vyvolat.

Srovnání přesnosti výpočtu aplikovaných algoritmů je shrnuto tabulkou E.1 a grafem na obrázku E.1. Křivka výpočtu hodnoty vzájemné informace v závislosti na kvantitě vstupních bodů je přibližně rovnoběžná s průběhem pro učebnicový výpočet. Pokud je učebnicový výpočet považován za nejpřesnější, pak je platné tvrzení, že tyto dvě metody jsou přibližně stejně přesné, resp. Fraser-Swinneyho algoritmus vykazuje drobné odchylky (maximální odhadovaná je 0,19795 Sh pro $2{\cdot}2^{20}$ – cca 2,1 milionu vstupních bodů).

Algoritmus, používající adaptivní XY dělení, ovšem vykazuje pro vstupní data, daná Rösslerovým atraktorem, zcela jiné hodnoty, než ostatní dvě metody. To je pravděpodobně způsobeno charakterem vstupních dat v tom smyslu, že žádná ze substruktur neobsahuje dva a více vstupních bodů. Řešení tohoto problému může být v dělení lokálního rastru bodů na méně substruktur nebo naopak v úvaze většího kvanta vstupních bodů.

Programování aplikace bylo uskutečněno v prostředí Microsoft Visual C++ 2008 Express Edition a grafické výstupy generovány programem Microsoft Excel 2003 na základě výstupních dat programu.

LITERATURA

[1] ADÁMEK, J. *Kódování a teorie informace*. Skriptum ČVUT Praha, 1994. ISBN80-01-00661-1.

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

[3] 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.

[4] COCHRAN, W.G. *Some methods for strengthening the common* χ^2 *test*. Biometrics. 10, 417-451, 1954.

[5] FRASER, A.M. *Information storage and entropy in strange attractors. IEEE Trans. Inform. Theory*. 35, 245-262, 1989.

[6] 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>.

[7] HAVLÍK, J. *Teorie přenosu*. Učebnice VA Brno, U-1039, 1976.

[8] 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/>.

[9] MOON, Y.I., RAJAGOPALAN, R., LALL, U. *Estimation of mutual information using kernel density estimators*. Physical Review E., 52, 2318-2321, 1995.

[10] NĚMEC, K. *Datová komunikace*. Skriptum VUT, VUTIUM 2000.

[11] PHAM, D. T. *Mutual information approach to blind separation of stationary sourcs – IEEE Trans. Inform. Theory*, vol. 48, pp. 1935–1946, July 2002.

[12] 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>.

[13] Shannon, C.E., WEAYVER, W. *The Mathematical Theory of Communication*. Urbana : [s.n.], 1949.

[14] ŠEBESTA, V. *Přenos dat*. Skriptum FEI VUT Brno, 1990, ISBN 80-14-0229-6.

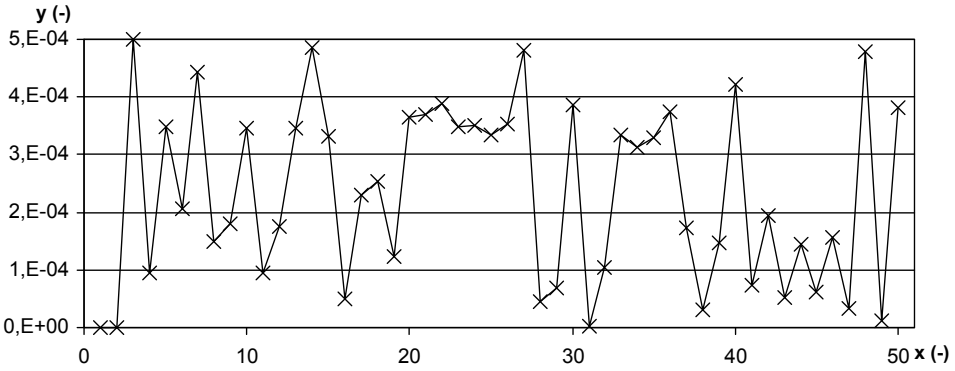
[15] WAND, M. P., JONES, M. C. *Kernel Smoothing*, Chapman & Hall, 1st edition, 1995.

SEZNAM PŘÍLOH

A PŘESNOST VSTUPNÍCH DAT RÖSSLEROVA ATRAKTORU

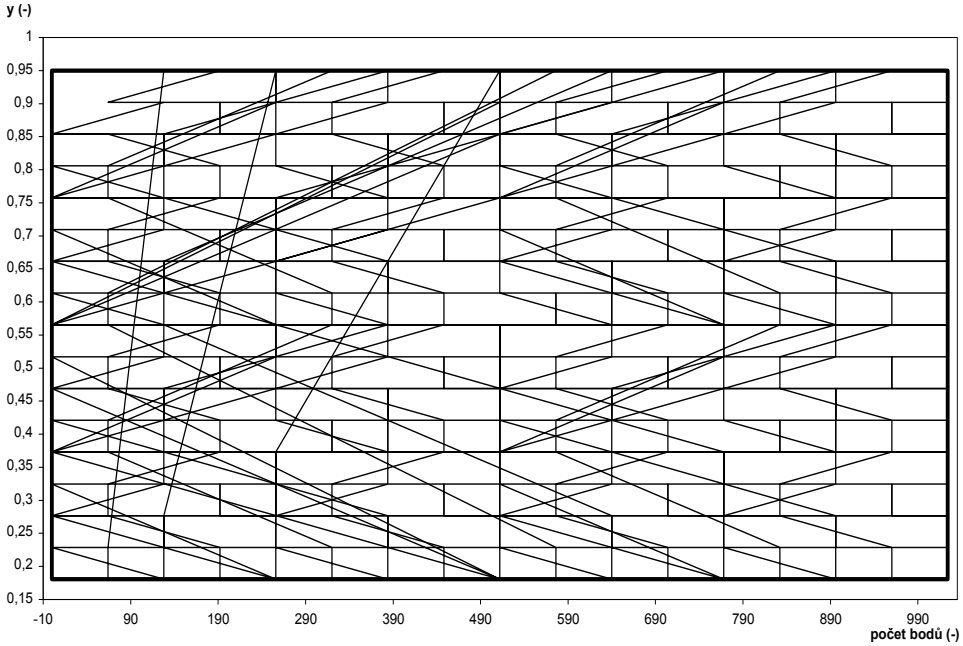
Tab. A.1: Výběr z analyzovaných prvků časové řady.

i / p	3	10	distance	i / p	3	10	distance
0	0,5	0,5	0,00E+00	25	0,274	0,274353606	3,54E-04
1	0,95	0,95	3,00E-10	26	0,757	0,756518078	4,82E-04
2	0,181	0,1805	5,00E-04	27	0,7	0,699954209	4,58E-05
3	0,562	0,56209505	9,50E-05	28	0,798	0,798069595	6,96E-05
4	0,935	0,935347978	3,48E-04	29	0,612	0,612387162	3,87E-04
5	0,23	0,229794125	2,06E-04	30	0,902	0,902002678	2,68E-06
6	0,673	0,672557382	4,43E-04	31	0,336	0,335896619	1,03E-04
7	0,837	0,83685101	1,49E-04	32	0,848	0,847666306	3,34E-04
8	0,519	0,518819309	1,81E-04	33	0,491	0,490686932	3,13E-04
9	0,949	0,948654168	3,46E-04	34	0,95	0,949670413	3,30E-04
10	0,185	0,185095864	9,59E-05	35	0,182	0,181626773	3,73E-04
11	0,573	0,573174463	1,74E-04	36	0,565	0,564826255	1,74E-04
12	0,93	0,929652892	3,47E-04	37	0,934	0,934030716	3,07E-05
13	0,249	0,24851389	4,86E-04	38	0,234	0,234145884	1,46E-04
14	0,71	0,709667999	3,32E-04	39	0,681	0,681422037	4,22E-04
15	0,783	0,782949456	5,05E-05	40	0,825	0,824926968	7,30E-05
16	0,646	0,645770501	2,29E-04	41	0,549	0,548805368	1,95E-04
17	0,869	0,869253652	2,54E-04	42	0,941	0,940948537	5,15E-05
18	0,432	0,431876613	1,23E-04	43	0,211	0,211144673	1,45E-04
19	0,932	0,932364976	3,65E-04	44	0,633	0,63293788	6,21E-05
20	0,24	0,239630005	3,70E-04	45	0,883	0,882844576	1,55E-04
21	0,692	0,692388368	3,88E-04	46	0,393	0,393034116	3,41E-05
22	0,809	0,80934952	3,50E-04	47	0,907	0,906521539	4,78E-04
23	0,586	0,586350924	3,51E-04	48	0,322	0,322012906	1,29E-05
24	0,922	0,921665369	3,35E-04	49	0,83	0,829618259	3,82E-04

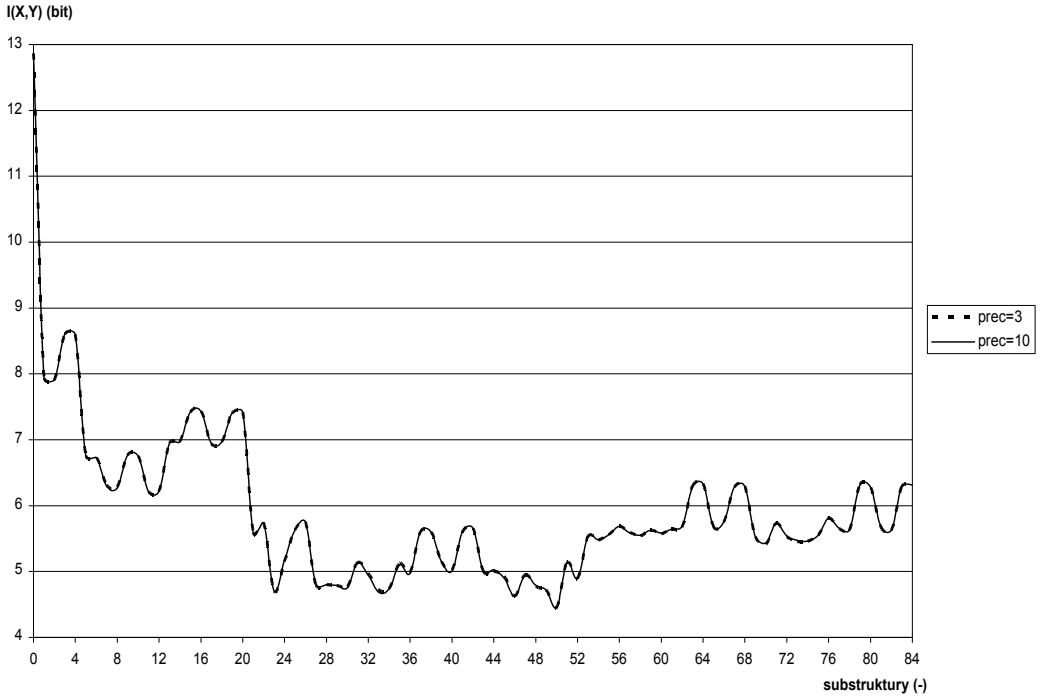


Obr. A.1: Nepřesnost vzniklá zaokrouhlením.

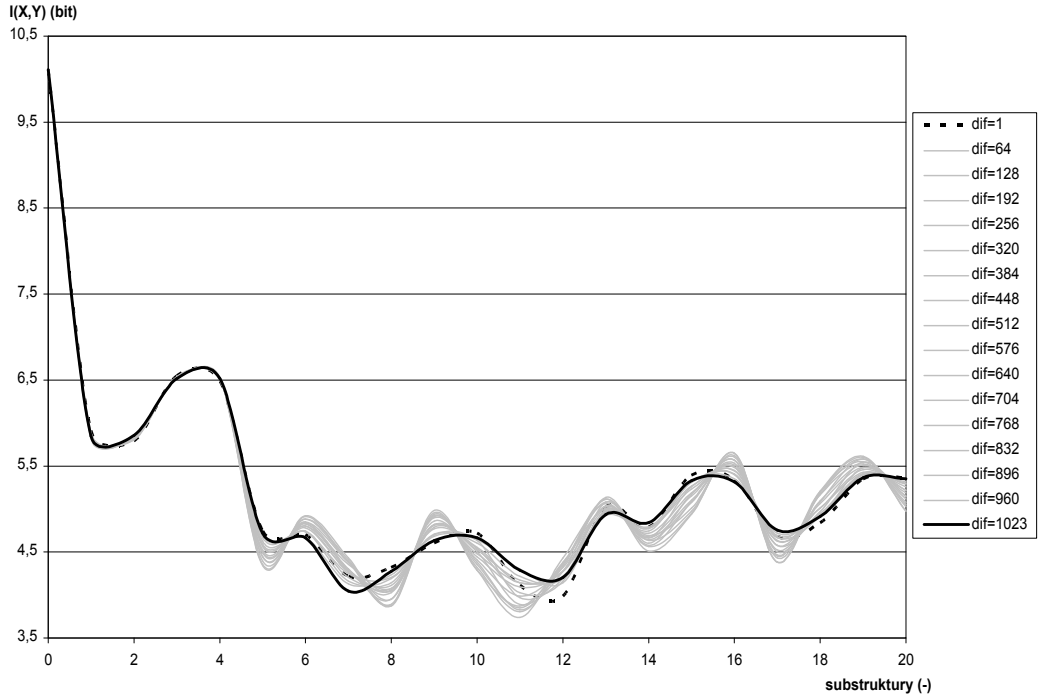
B FRASER-SWINNEYHO ALGORITMUS



Obr. B.1: Dělení lokálního rastru Fraser-Swinneyho algoritmem.

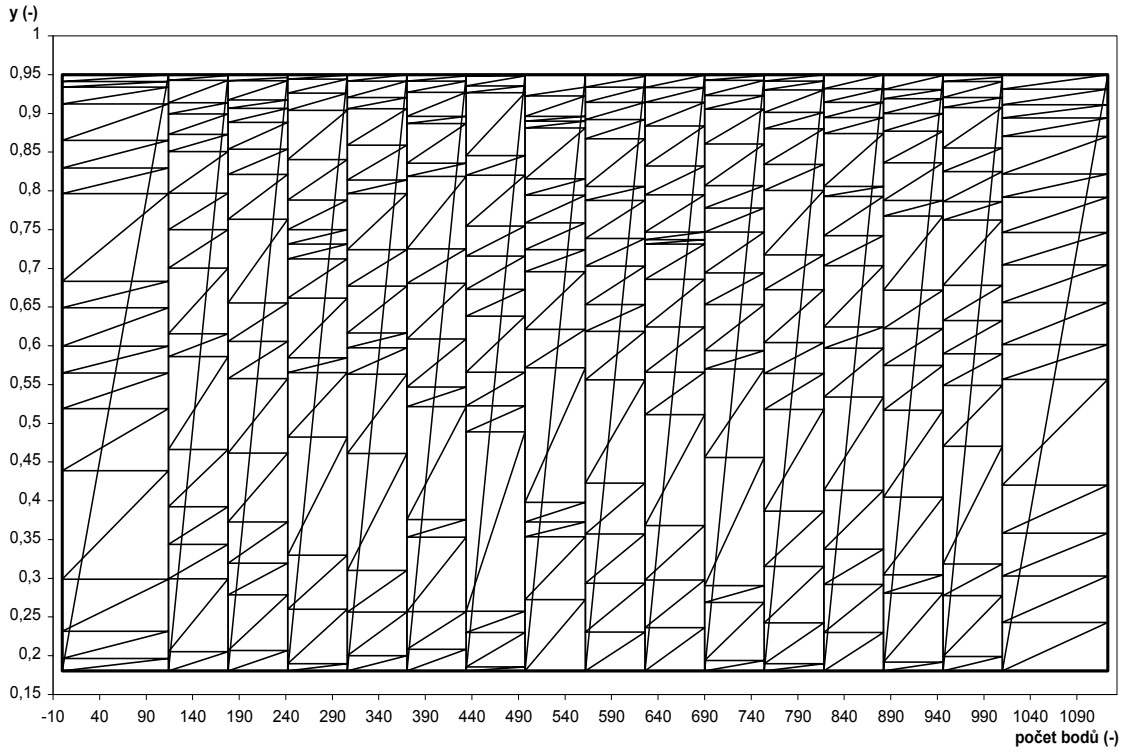


Obr. B.2: Průběh vzájemné informace (při 4096 vstupních bodů pro různou přesnost vstupních dat.



Obr. B.3: Průběh vzájemné informace pro 1024 vstupních bodů při různém vzájemném posunu časových řad.

C VÝPOČET VZÁJEMNÉ INFORMACE POMOCÍ ADAPTIVNÍHO XY DĚLENÍ



Obr. C.1: Adaptivní XY dělení lokálního rastru.

D SROVNÁNÍ ALGORITMŮ DLE RYCHLOSTI

Vstupní data		učebnicový výpočet			Fraser-Swinneyho algoritmus			Adaptivní XY dělení		
exponent	**počet bodů**	**doba trvání výpočtu**		**výpočet / odhad**	**doba trvání výpočtu**		**výpočet / odhad**	**doba trvání výpočtu**		**výpočet / odhad**
exp (-)	2*(2^exp) (-)	sekund	hh:mm:ss		sekund	hh:mm:ss		sekund	hh:mm:ss	
10	2048	2	00:00:02	výpočet	1	00:00:01	výpočet	0	00:00:00	výpočet
11	4096	4	00:00:04	výpočet	2	00:00:02	výpočet	1	00:00:01	výpočet
12	8192	18	00:00:18	výpočet	15	00:00:15	výpočet	5	00:00:05	výpočet
13	16384	47	00:00:47	výpočet	28	00:00:28	výpočet	15	00:00:15	výpočet
14	32768	168	00:02:48	výpočet	125	00:02:05	výpočet	111	00:01:51	odhad
15	65536	656	00:10:56	výpočet	355	00:05:55	výpočet	204	00:03:24	odhad
16	131072	2309	00:38:29	výpočet	2543	00:42:23	výpočet	1274	00:21:14	odhad
17	262144	5745	01:25:45	výpočet	2804	00:46:44	odhad	1912	00:31:52	odhad
18	524288	23093	06:24:53	odhad	34797	09:39:57	odhad	17571	04:52:51	odhad
19	1048576	81850	22:44:10	odhad	60032	16:40:32	odhad	35587	09:53:7	odhad
20	2097152	342553	95:09:13	odhad	417015	115:50:15	odhad	252873	70:14:33	odhad

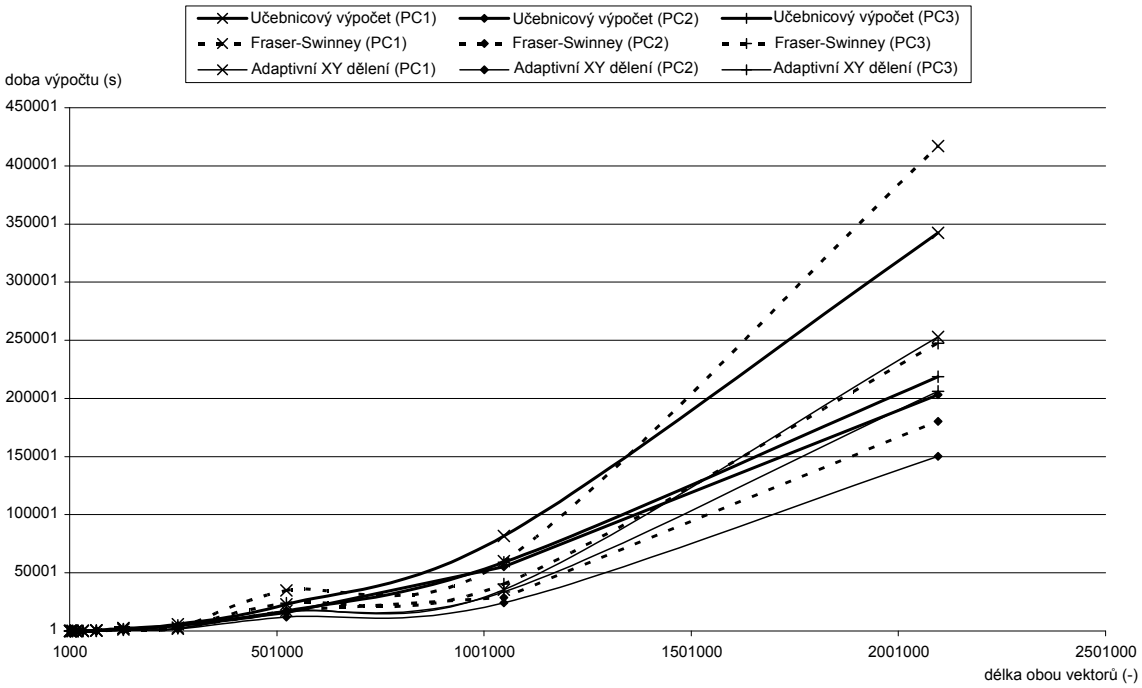
Obr. D.1: Výstupy testu aplikace na 1. počítači.

Vstupní data		učebnicový výpočet			Fraser-Swinneyho algoritmus			Adaptivní XY dělení		
exponent	**počet bodů**	**doba trvání výpočtu**		**výpočet / odhad**	**doba trvání výpočtu**		**výpočet / odhad**	**doba trvání výpočtu**		**výpočet / odhad**
exp (-)	2*(2^exp) (-)	sekund	hh:mm:ss		sekund	hh:mm:ss		sekund	hh:mm:ss	
10	2048	0	00:00:01	výpočet	0	00:00:00	výpočet	0	00:00:00	výpočet
11	4096	1	00:00:02	výpočet	1	00:00:01	výpočet	1	00:00:01	výpočet
12	8192	5	00.00:08	výpočet	4	00:00:04	výpočet	3	00:00:03	výpočet
13	16384	23	00:00:28	výpočet	8	00.00:08	výpočet	7	00:00:07	výpočet
14	32768	88	00:01:52	výpočet	67	00:01:07	výpočet	58	00:00:58	odhad
15	65536	347	00:07:10	výpočet	134	00:02:14	výpočet	108	00:01:48	odhad
16	131072	1319	00:26:31	výpočet	1058	00:17:38	výpočet	728	00:12:08	odhad
17	262144	4506	01:23:06	výpočet	2127	00:35:27	výpočet	1500	00:25:00	odhad
18	524288	15771	04:44:55	výpočet	17304	04:48:24	odhad	12089	03:21:29	odhad
19	1048576	55199	16:27:25	odhad	28857	08:00:57	odhad	24102	06:41:42	odhad
20	2097152	203197	60:44:02	odhad	180121	50:02:01	odhad	150263	41:44:23	odhad

Obr. D.2: Výstupy testu aplikace na 2. počítači.

Vstupní data		učebnicový výpočet			Fraser-Swinneyho algoritmus			Adaptivní XY dělení		
exponent	**počet bodů**	**doba trvání výpočtu**		**výpočet / odhad**	**doba trvání výpočtu**		**výpočet / odhad**	**doba trvání výpočtu**		**výpočet / odhad**
exp (-)	2*(2^exp) (-)	sekund	hh:mm:ss		sekund	hh:mm:ss		sekund	hh:mm:ss	
10	2048	1	00:00:00	výpočet	0	00:00.00	výpočet	0	00:00:00	výpočet
11	4096	2	00:00:01	výpočet	1	00:00:01	výpočet	1	00:00:01	výpočet
12	8192	8	00:00:05	výpočet	6	00:00:06	výpočet	5	00:00:05	výpočet
13	16384	28	00:00:23	výpočet	14	00:00:14	výpočet	12	00:00:12	výpočet
14	32768	112	00:01:28	výpočet	92	00:01:32	výpočet	80	00:01:20	odhad
15	65536	430	00:05:47	výpočet	188	00:03:08	výpočet	152	00:02:32	odhad
16	131072	1591	00:21:59	výpočet	1453	00:24:13	výpočet	999	00:16:39	odhad
17	262144	4986	01:15:06	výpočet	2984	00:49:44	výpočet	2104	00:35:04	odhad
18	524288	17095	04:22:51	odhad	23764	06:36:04	odhad	16479	04:34:39	odhad
19	1048576	59245	15:19:59	odhad	40485	11:14:45	odhad	33670	09:21:10	odhad
20	2097152	218642	56:26:37	odhad	247368	68:42:48	odhad	206015	57:13:35	odhad

Obr. D.3: Výstupy testu aplikace na 3. počítači.

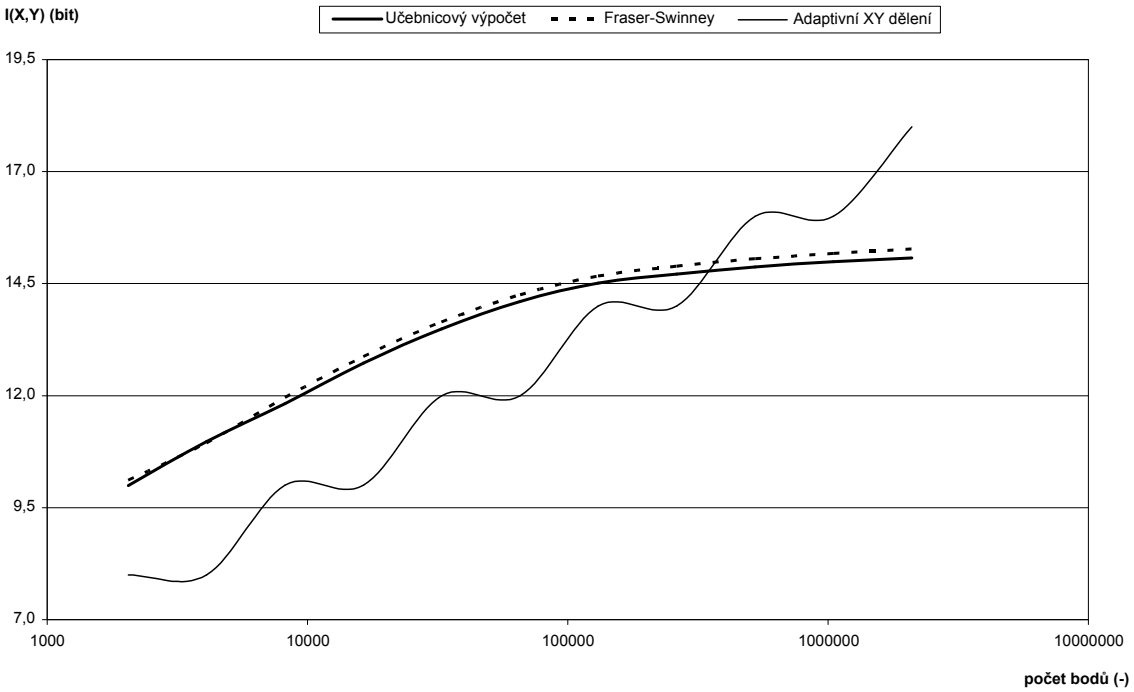


Obr. D.4: Souhrn testů na všech počítačích.

E SROVNÁNÍ ALGORITMŮ DLE PŘESNOSTI

Tab. E.1: Výběr z analyzovaných prvků časové řady.

X	Učebnicový výpočet	Fraser-Swinney	Adaptivní XY dělení
2048	9,9922	10,1098	8,0000
4096	10,9758	10,9366	8,0000
8192	11,8211	11,9547	10,0000
16384	12,7246	12,8663	10,0000
32768	13,4868	13,6365	12,0000
65536	14,0982	14,2560	12,0000
131072	14,5094	14,6752	14,0000
262144	14,7159	14,8897	14,0000
524288	14,8708	15,0526	16,0000
1048576	14,9869	15,1769	16,0000
2097152	15,0740	15,2725	18,0000



Obr. E.1: Porovnání přesnosti aplikovaných metod.