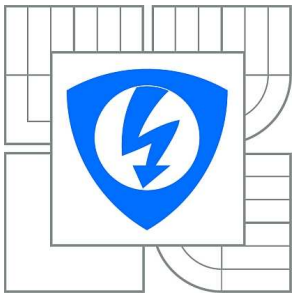

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY


FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

ÚSTAV BIOMEDICÍNSKÉHO INŽENÝRSTVÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

DEPARTMENT OF BIOMEDICAL ENGINEERING


# ZPRACOVÁNÍ GENOMICKÝCH SIGNÁLŮ FRAKTÁLY

PROCESSING OF FRACTAL GENOMIC SIGNALS

DIPLOMOVÁ PRÁCE

MASTER'S THESIS


AUTOR PRÁCE — Bc. JIŘÍ NEDVĚD

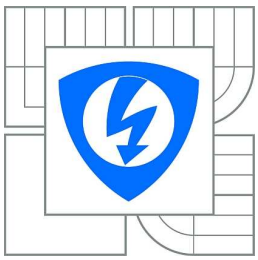
AUTHOR

VEDOUCÍ PRÁCE — Ing. MARTIN VALLA

SUPERVISOR



BRNO 2012



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**

**Fakulta elektrotechniky a komunikačních technologií**

**Ústav biomedicínského inženýrství**

# Diplomová práce

magisterský navazující studijní obor
**Biomedicínské a ekologické inženýrství**

| | | | |
|---|---|---|---|
| ***Student:*** | Bc. Jiří Nedvěd | ***ID:*** | 106665 |
| ***Ročník:*** | 2 | ***Akademický rok:*** | 2011/2012 |

**NÁZEV TÉMATU:**

## Zpracování genomických signálů fraktály

**POKYNY PRO VYPRACOVÁNÍ:**

1) Seznamte se s kódováním DNA v bioinformatice. 2) Proveďte literární rešerši v oblasti reprezentace genomických sekvencí a seznamte se s popisem DNA obrazu sestrojeným metodou CGR (Chaos Game Representation). Zjistěte možnosti analýzy takto sestrojeného obrazu metodou BCM (Box Counting Method). 4) Navrhněte algoritmus grafické reprezentace DNA a fraktálové analýzy pro klasifikaci DNA a implementujte jej do grafického uživatelského prostředí MATLAB. Aplikace bude umožňovat vykreslení dvou sekvencí pro vzájemné porovnání a možnost rozšíření o reprezentaci v zobrazení frekvenční CGR. 5) Funkčnost vytvořeného programu ověřte na reálných datech. 6) Proveďte diskusi získaných výsledků a zhodnoťte využitelnost dosažené práce. Uveďte návrhy na případná vylepšení.

**DOPORUČENÁ LITERATURA:**

[1] ZU-GUO, Y., ANH, V. Fractals in DNA sequence analysis. IOP electronic journals : Chinese Physics [online]. 2002, is. 12 [cit. 2002-07-20], s. 1313-1318.
[2] GARIAEV, Peter , et al. Fractal Structure in DNA code and human language: Towards a semiotics of biogenetic information. In International Journal of Computing Anticipatory Systems. [s.l.] : [s.n.], 2002. s. 255-273. ISBN 2-9600262-7-6. ISSN 1373-5411 .

| | | | |
|---|---|---|---|
| ***Termín zadání:*** | 6.2.2012 | ***Termín odevzdání:*** | 18.5.2012 |

***Vedoucí práce:*** Ing. Martin Valla

**prof. Ing. Ivo Provazník, Ph.D.**
*Předseda oborové rady*

**UPOZORNĚNÍ:**

# Abstrakt:

Diplomová práce má ukázat možnosti klasifikace genomických sekvencí pomocí CGR a FCGR převedení na obraz. Z těchto obrazů se vypočte klasifikátor pomocí metody BCM. Dále je zde popsán vytvořený program a jeho možnosti při klasifikaci. Na konci je srovnáno množství sekvencí pro různé nastavení programu.



# Klíčová slova:

klasifikace, fraktál, genomická sekvence, DNA, CGR (Chaos Game Reprezentece), FCGR (Frekvenční Chaos Game Reprezentace), BCM (Box Counting Metoda)



# Abstract:

This diploma project is showen possibilities in classification of genomic sequences with CGR and FCGR methods in pictures. From this picture is computed classificator with BCM. Next here is written about the programme and its opportunities for classification. In the end is compared many of sequences computed in different options of programme.



# Key words:

classification, fractal, genomic sequence, DNA, CGR (Chaos Game Representation), FCGR (Frequency Chaos Game Representation), BCM (Box Counting Method)

# Bibliografická citace:

NEDVĚD, J. *Zpracování genomických signálů fraktály*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2012. 41 s. Vedoucí diplomové práce Ing. Martin Valla.

# Prohlášení

Prohlašuji, že svou diplomovou práci na téma zpracování genomických signálů fraktály jsem vypracoval samostatně pod vedením vedoucího diplomové 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 diplomové práce dále prohlašuji, že v souvislosti s vytvořením této práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

V Brně dne 5. května 2012 ……

podpis autora

# Poděkování

Děkuji vedoucímu své diplomové práce ing. M. Vallovi za podnětné rady a odbornou pomoc.

V Brně dne 5. května 2012 ……

podpis autora

# Obsah:

# Seznam obrázků:

# 1. Úvod

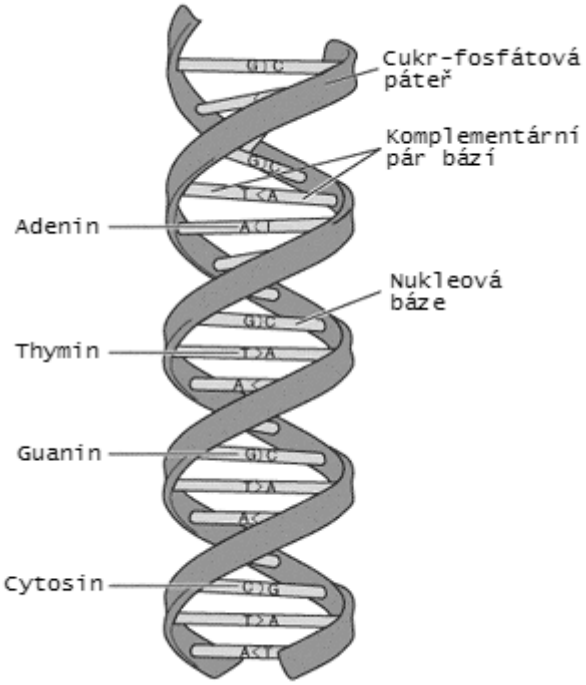
Na počátku práce je shrnuto kódování DNA v bioinformatice, vlastní zdroj a získávání genomických sekvencí a výběr zvoleného typu dat. Dále je zde vysvětlena metoda chaos game reprezentace (zkratka CGR, více v kap. 2.4.1. Chaos game reprezentace sekvence), jejím pozměněním je získána metoda frekvenční CGR (FCGR, více v kap. 2.4.4. Frekvenční CGR), která je pak zpracovávána pomocí metody box counting (BCM, více v kap. 2.4.3. Box counting method). Následně budou předvedeny další možnosti plošné reprezentace sekvence. BCM zpracováním se ze sekvence získají hodnoty dimenzí, ze kterých se sestaví graf. Z grafu je pak odečtena hodnota směrnice trendu, která je považována za multifraktální koeficient a odpovídá jen té dané sekvenci. Všechny tyto metody jsou zapracovány do programu *GenomeFCGR* vytvořeném v programu *Matlab* verze 7.10.0 (R2010a) za použití bioinformatického toolboxu. Všechny možnosti tohoto programu budou také v práci popsány. V závěru práce pak budou výsledky pro několik sekvencí (viz kap. 3.3.2. Vybrané sekvence), hodnocení programu a návrhy na další zlepšení programu (více v kap. 3.4. Návrhy na vylepšení programu). V samotném závěru pak jsou doporučené hodnoty nastavení programu, které pak lze použít k samotné klasifikaci sekvencí.

# 2. DNA a její kódování

DNA je zkratka chemického názvu deoxyribonukleové kyseliny, která v sobě uchovává dědičnou informaci o celém jedinci. Celá genetická výbava, nazývána **genom**, je uchovávána v jádře každé buňky patřičného organismu. Genom je rozdělen do chromozomů (více viz kap. 0.

Běžný popis DNA sekvencí, nebo také genomických sekvencí, se většinou děje podle délky sekvence, podle chromozomu, ze kterého byla sekvence získána, nebo podle druhu, ze kterého sekvence pochází. Další popis sekvence bude rozebrán v kap. 2.4. Popis DNA sekvencí.

Chromozom), kde je zamotána v podobě dvojšroubovice (Obrázek 1: Dvojšroubovice DNA.). Chromozom se pak dále dělí na geny, což jsou základní jednotlivé dědičné znaky. [1]



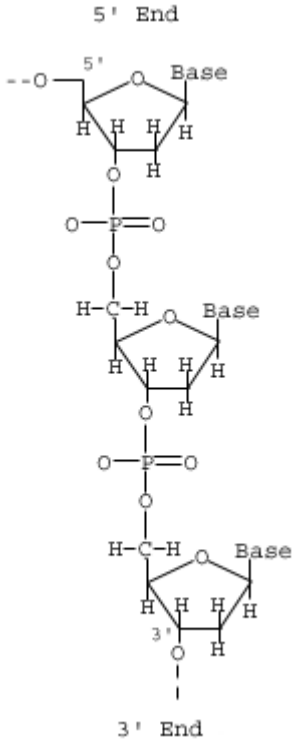

**Obrázek 1: Dvojšroubovice DNA. [1]**

## 2.1. Struktura DNA v organismu

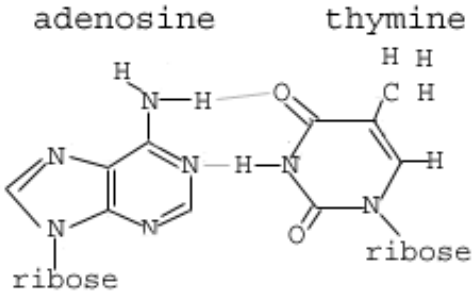
Dvojšroubovice je tvořena cukrofosfátovou páteří s posloupností cukr-fosfát-cukr (Obrázek 2). Na každý cukr, jmenovitě deoxyribóza, je navázána jedna z nukleových bází. Báze neboli nukleotidy jsou čtyři – dvě purinové, mezi které patří adenin (zkratka A) a guanin (zkratka G), a dvě pyrimidinové, cytosin (zkratka C) a thymin (zkratka T). Dvě na sebe navázané báze tvoří komplementární pár. Komplementární páry tvoří vazby adenin – thymin, nebo opačně (Obrázek 3), které jsou svázány dvěma vodíkovými vazbami, a vazby guanin – cytosin, nebo opačně (Obrázek 4), které jsou pohromadě drženy pomocí tří vodíkových vazeb.

Thymin (T) je v RNA (ribonukleová kyselina) nahrazen při transkripci DNA uracylem (U), který patří mezi báze pyrimidinové, a v používaných genomických sekvencích se neobjevuje. Transkripce je dělení DNA, kdy se dvojšroubovice rozdělí na samostatná vlákna a druhé vlákno se znova vytvoří pomocí RNA. Tato vlastnost zajišťuje, že pro uchování dědičné informace stačí pouze jeden řetězec dvojšroubovice a i pro kódování je použita posloupnost znaků bází pouze z jednoho řetězce. [1, 2, 3]

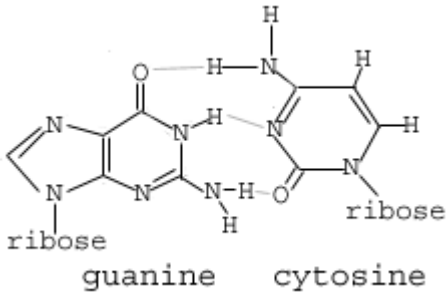
Při dále použitých metodách vykreslování je jedno, který z těchto řetězců je použit. Opačný či komplementární řetězec DNA vytvoří obraz sekvence, jehož obraz bude jen středově otočen vůči přímému vláknu DNA.




**Obrázek 2: Cukrofosfátová páteř dvojšroubovice DNA. Je označována úsekem mezi koncem od třetího uhlíku deoxyribozy na jednom konci řetězce (označeno 3 konec – 3 end) a na druhém konci od pátého uhlíku (označeno 5 konec – 5 end). V místě Base je navázána jedna z bází. [5]**




**Obrázek 3: Komplementární pár tvořený adeninem a thyminem. [4]**



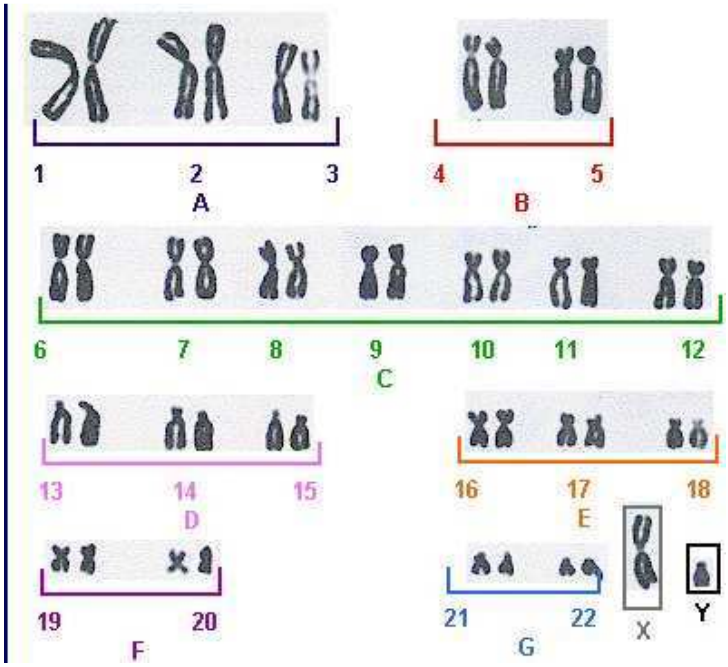

**Obrázek 4: Komplementární pár tvořený guaninem a cytosinem. [4]**

Běžný popis DNA sekvencí, nebo také genomických sekvencí, se většinou děje podle délky sekvence, podle chromozomu, ze kterého byla sekvence získána, nebo podle druhu, ze kterého sekvence pochází. Další popis sekvence bude rozebrán v kap. 2.4. Popis DNA sekvencí.

## 2.2. Chromozom

Dvojšroubovice je doslova smotána do chromatid a dvě chromatidy jsou spojeny přes jednu centromeru. Podle polohy centromery se chromozomy dělí na metacentrické, kde je centromera uprostřed, submetacentrické, kde je téměř uprostřed, akrocentrické, kde se blíží spíše kraji, a telocentrické, kde je spojení na koncích. Telocentrické chromozomy se v lidských chromozomech nevyskytují. Podle velikosti, a právě podle polohy centromery, lze chromozomy rozdělit do následujících sedmi skupin (Obrázek 5): [12]

- A – chromozomy 1, 2, 3 – velké, metacentrické nebo submetacentrické chromozomy
- B – chromozomy 4, 5 – velké, metacentrické chromozomy
- C – chromozomy 6, 7, 8, 9, 10, 11, 12, X – středně velké submetacentrické chromozomy
- D – chromozomy 13, 14, 15 – středně velké akrocentrické chromozomy; první dva mají na jednom konci satelity
- E – chromozomy 16, 17, 18 – krátké metacentrické nebo submetacentrické chromozomy
- F – chromozomy 19, 20 – krátké metacentrické chromozomy
- G – chromozomy 21, 22, Y – krátké akrocentrické chromozomy; první dva obsahují satelity




**Obrázek 5: Rozděleni chromozomů podle podobností. [12]**

## 2.3. Kódování pomocí bází

Posloupnost nukleotidových bází nese celou dědičnou informaci v neměnné podobě. Jedná se o jednoduché zapsání zkratek bází do posloupnosti, jak jdou ve šroubovici za sebou. Informace v takovéto sekvenci je rozdělená do úseků nesoucích informaci, tzv. exonů, a úseků oddělujících, tzv. intronů. Přesná funkce intronů není doposud známá, ale spekuluje se o určité ochranné funkci exonů, nebo o funkci, která má kontrolovat informace v exonech. Sekvenci pak lze rozdělit na **kodóny**, neboli triplety, což je posloupnost tří po sobě jdoucích nukleotidů, která reprezentuje jednu aminokyselinu. Triplety kódují aminokyseliny, které jsou pak syntetizovány v buňce. Počet aminokyselin je asi 20, zatímco možných tripletů je 64 (čtyři na třetí). Nepoměr napovídá, že jednu aminokyselinu lze kódovat více triplety (Tabulka 1). [3]

**Tabulka 1: Kód tripletu a aminová kyselina. [6]**

| | | Druhé písmeno tripletu | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | T | | C | | A | | G | | | |
| První písmeno tripletu | T | TTT | Fenylalanin | TCT | Serin | TAT | Tyrosin | TGT | Cystein | T | Třetí písmeno tripletu |
| | | TTC | | TCC | | TAC | | TGC | | C | |
| | | TTA | Leucin | TCA | | TAA | STOP | TGA | STOP | A | |
| | | TTG | | TCG | | TAG | | TGG | Tryptofan | G | |
| | C | CTT | Leucin | CCT | Prolin | CAT | Histidin | CGT | Arginin | T | |
| | | CTC | | CCC | | CAC | | CGC | | C | |
| | | CTA | | CCA | | CAA | Glutamin | CGA | | A | |
| | | CTG | | CCG | | CAG | | CGG | | G | |
| | A | ATT | Izoleucin | ACT | Treonin | AAT | Asparagin | AGT | Serin | T | |
| | | ATC | | ACC | | AAC | | AGC | | C | |
| | | ATA | | ACA | | AAA | Lysin | AGA | Arginin | A | |
| | | ATG | Methionin, START | ACG | | AAG | | AGG | | G | |
| | G | GTT | Valin | GCT | Alanin | GAT | Kyselina asparágová | GGT | Glycin | T | |
| | | GTC | | GCC | | GAC | | GGC | | C | |
| | | GTA | | GCA | | GAA | Kyselina glutamová | GGA | | A | |
| | | GTG | | GCG | | GAG | | GGG | | G | |

Z tabulky výše je jasné, že pokud získáme reprezentaci DNA ve formě posloupnosti aminokyselin, tak nelze zpětně jednoznačně odvodit původní DNA posloupnost. Posloupnost aminokyselin je také využívána a zpracovávána jako nositel informace v bioinformatice, zde ale využita nebude. Program *Matlab* sice umí převádět tyto posloupnosti zpět do posloupnosti bází, ale z důvodu této nejednoznačnosti bylo od ní upuštěno.

## 2.4. Popis DNA sekvencí

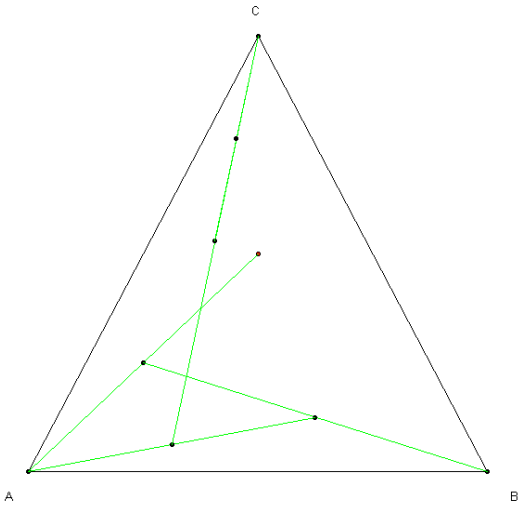
Sekvenci DNA lze reprezentovat různými způsoby. Statistický popis sekvencí znamená, že se spočítají jednotlivé báze a sestaví se tabulka s procentuálním zastoupením jednotlivých bází. [13]

Zde bude použita metoda CGR (Chaos game representation), která sekvenci převede do obrazu, a její další odnož FCGR (frekvenční CGR). Pro porovnání obrazů je potřeba z nich dostat číslo, které se bude porovnávat. K tomu bude použita metoda BCM (box counting method).

### 2.4.1. Chaos game reprezentace sekvence

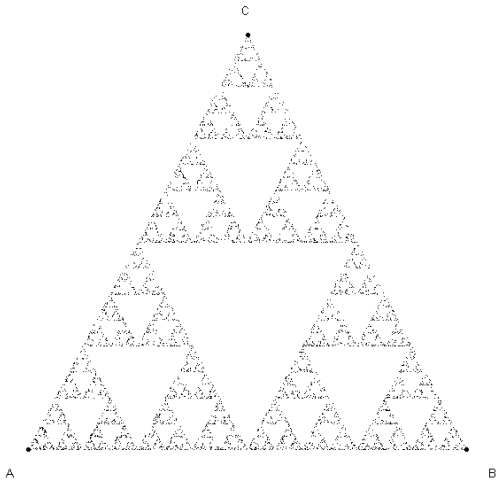
Tato metoda byla odvozena pro tvorbu fraktálu s názvem Sierpinského trojúhelník, který patří mezi iterační funkční systémy. Jde o černý rovnostranný trojúhelník (může být i obecný), ve kterém se deterministickým způsobem vytvoří fraktál. Vlastně stačí spojit středy stran a ohraničené pole vyjmout, nebo obarvit na bílo. Toto je první iterace, kdy z jednoho černého trojúhelníku vzniknou tři menší. V každé další iteraci se pak vezmou všechny černé trojúhelníky a provede se s nimi stejný proces. Fraktál vzniká při nekonečném počtu iterací, tak aby se splnilo pravidlo, že obraz je nezávislý na přiblížení. [14]

Jiný způsob vykreslení je použití metody chaos game. Pro vytvoření stačí zadat vrcholy trojúhelníku a libovolný počáteční bod uvnitř trojúhelníku, nebo lze zvolit i jeden z vrcholů. Jednoduchou iterací, kdy je náhodně zvolen jeden z vrcholů a v poloviční vzdálenosti mezi počátečním bodem a vrcholem se vykreslí nový bod, se postupně bude plocha trojúhelníka zaplňovat. Nový bod je pro následující iteraci počátečním bodem. Náčrt iterací je na obrázku níže (Obrázek 6). Do obrázku byly přikresleny spojnice vrcholů pro lepší představu trojúhelníku. [7]

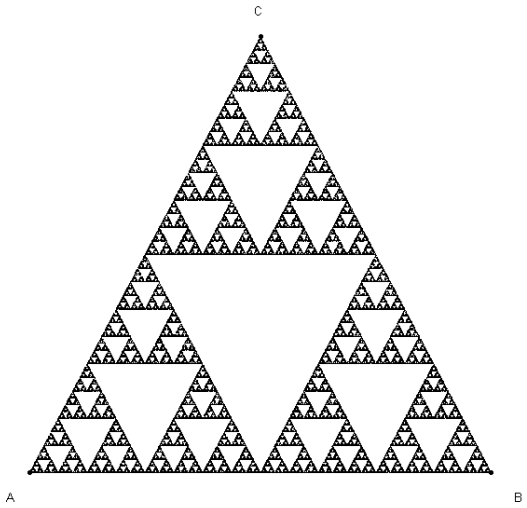



**Obrázek 6: Konstrukce Sierpinského trojúhelníku CGR metodou. [generováno vlastním programem v Matlabu] [8]**

Přesnost nebo plnost Sierpinského trojúhelníku konstruovaného pomocí této metody je dána počtem iterací, což je přímo úměrné počtu vykreslovaných bodů. Následující obrázky osvětlí tuto problematiku. Vlevo je obrázek po pěti tisících iteracích, neboli po vykreslení pěti tisíc bodů (Obrázek 7). Vpravo pak tentýž obrázek po stech tisících iteracích (Obrázek 8).

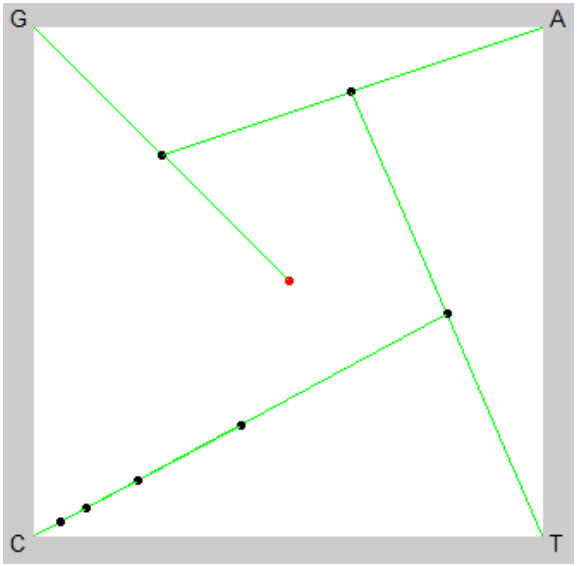



**Obrázek 7: CGR konstrukce Sierpinského trojúhelníku po 5 000 iteracích. [generováno vlastním programem v Matlabu] [8]**

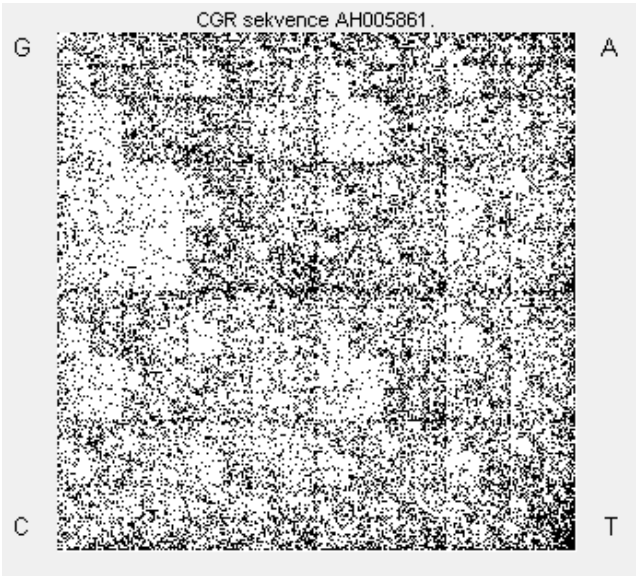



**Obrázek 8: CGR konstrukce Sierpinského trojúhelníku po 100 000 iteracích. [generováno vlastním programem v Matlabu] [8]**

Pro vykreslení DNA úseků je nutno obraz i program upravit. Místo trojúhelníku použijeme čtverec, kdy vrcholy ponesou názvy, nebo spíše zkratky bází. Prvek náhody, který vybíral vrchol, je vyřazen. Vrchol, ke kterému se bude iterace ubírat, určí posloupnost sekvence, kdy první iteraci určuje první znak v sekvenci, druhou druhý, atd. (Obrázek 9). Celý obraz sekvence je opět tedy tvořen body, které reprezentují každou jednotlivou bázi v sekvenci. Délka sekvence je opět shodná s počtem bodů vytvořených v obraze. Na obrázcích níže je naznačena počáteční posloupnost ve čtverci (počátečních 7 znaků ze sekvence - Obrázek 9) a vedle je obraz celé sekvence ‚Homo sapiens tight junction protein ZO-2 (TJP2) gene‘, s lokusem AH005861 a délkou 26 841 bp (Obrázek 10).




**Obrázek 9: Počátečních sedm bází (znaků) v sekvenci ‚Homo sapiens tight junction protein ZO-2 (TJP2) gene'. První (červený) bod není pro zobrazení sekvence zobrazen. Zde jen zdůrazňuje počátek. [Generováno vlastním skriptem v Matlabu – *ukazkaCGR.m*]**



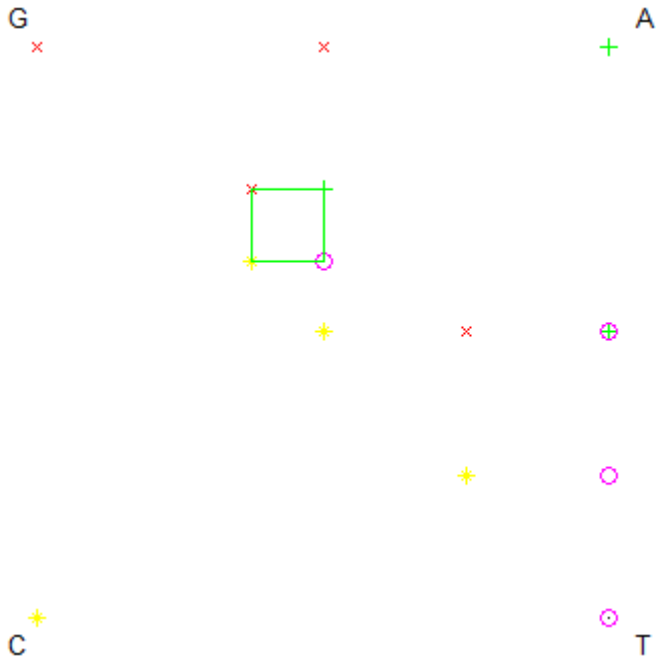

**Obrázek 10: CGR celé sekvence ‚Homo sapiens tight junction protein ZO-2 (TJP2) gene' s délkou, nebo také s počtem černých bodů 26 841. [Výřez z finálního programu GenomeFCGR vytvořeného v prostředí Matlab]**

Hlavní výhodou CGR oproti FCGR (viz kap. 2.4.4. Frekvenční CGR) je čistě černo-bílé zobrazení, kdy pozadí je bílé a body nesoucí informaci o sekvenci černé. Nevýhodou je, že v Matlabu je použita funkce *plot* a výsledný obraz sekvence není možnost uložit jako obraz pro další zpracování, třeba metodou BCM. Lze sice ručně obrazy uložit a následně znova načíst, ale není možné nijak ovlivnit velikost obrazu, ani velikost jednotlivých vykreslovaných bodů.

### 2.4.2. Dopady tripletu

Při použití čtverce GACT (z Obrázek 9), a budou-li brány kombinace tripletů (z Tabulka 1), lze pak odvodit místa dopadu ve čtverci pro daný triplet s ohledem na polohu vrcholů čtverce. Pro zjednodušení bude požito extrémů – vrcholů čtverce jako počátku metody CGR (viz kap. 2.4.1. Chaos game reprezentace sekvence). Jako sekvence bude použita posloupnost znaků v hledaném tripletu. Pro následující příklad bude použit start kodón, který kóduje aminokyselinu methionin, a má posloupnost ATG znaků v tripletu. Vybraná plocha je ohraničena zelenou čarou a tvoří ji čtverec (Obrázek 11). Pro ověření správnosti jsou jednotlivé body, které k sobě patří, zobrazeny stejným tvarem a barvou. Bod, který má počátek ve vrcholu G, je červené x. Při procházení posloupnosti se nejdříve přiblíží k vrcholu A a vznikne nový bod s červeným x uprostřed strany GA. Při druhé iteraci se přiblíží na poloviční vzdálenost k vrcholu T a nakonec opět do poloviční vzdálenosti k vrcholu G. Všechny takto vzniklé body mají tvar červeného x. Stejné je to pro vrchol A, kde body mají tvar zeleného plus, pro vrchol C, kde body mají tvar žluté hvězdy, i pro vrchol T,

kde jsou ve tvaru fialového kroužku. Je-li aminokyselina kódována více možnostmi, tak je potřeba projet tento cyklus několikrát s použitím jiné posloupnosti tripletu.




**Obrázek 11: Hledání polohy tripletu ve čtverci GACT. [Generováno vlastním programem v Matlabu]**

Oproti tomuto výpočtu dopadů daného tripletu do pozice v obraze existuje jiná definice tripletů v obraze. Je to definice vzhledem k vrcholům čtverce. Čtverec je rozdělen svislou čárou v polovině strany a další vodorovnou čárou opět v polovině strany na stejné čtyři části. Tyto části pak nesou název podle přilehlého vrcholu, takže jsou čtverce A, C, G a T. Každý z těchto čtverců je opět rozdělen na další čtyři menší a označí se zase podle vrcholu, ke kterému mají nejblíže (druhé písmeno označení). Příklad: čtverec A je rozdělen na čtverce s označením AA, AC, AG a AT. Celý obraz pak obsahuje šestnáct čtverců. A protože triplet je složen ze tří znaků, tak se celý postup opět opakuje. Vznikne šedesát čtyři čtverců a opět se připojí k označení písmeno vrcholu, ke kterému čtverec směřuje směrem od středu děleného čtverce. Příklad: čtverec AA je rozdělen na AAA, AAC, AAG a AAT. Poloha těchto tripletů je odlišná od předešlé. [8]

### 2.4.3. Box counting method

Box counting method (užívaná zkratka: BCM) je metoda, která slouží k výpočtu multifraktálního koeficientu daného obrazu. Název pochází z prvních písmen anglických slov a překlad plně popisuje metodu, jakou počítá – počítání oblastí, nebo v našem případě počítání černých bodů neboli pixelů.

Jedná se také o iterační metodu, kdy v každé iteraci je vypočtena dimenze obrazu. Výpočet dimenze se provádí podle vztahu 1 a pro její výpočet je potřeba znát velikost oblasti, pod kterou se spočítají černé pixely. Převládají-li černé body nad bílými, pak je celá maska

vyhodnocena jako černý bod a započítá se do množství pro výpočet dimenze. Pokud je více bílých bodů, bude maska vyhodnocena jako bílá a nebude hrát roli při výpočtu (viz Obrázek 12). Maska projde celý obraz, přesahy zanedbá a poté se spočítá dimenze. Při další iteraci se velikost masky zvětší o jeden pixel a celý proces se opakuje. [15]

Tyto dimenze jsou pak proloženy křivkou, jejíž směrnice je hledaným multifraktálním koeficientem a zároveň klasifikátorem dané sekvence. [8]

Fraktální dimenze *D* udává míru nepravidelnosti daného objektu. Pro výpočet fraktální dimenze je požita rovnice 1. [9]

$$D = \frac{\log N}{\log\left(\frac{1}{r}\right)}\ [-], \tag{1}$$

kde *D* je fraktální dimenze, *N* je faktor změny délky, neboli počet soběpodobných částí, $\frac{1}{r}$ je faktor změny měřítka, *r* je velikost soběpodobné části.

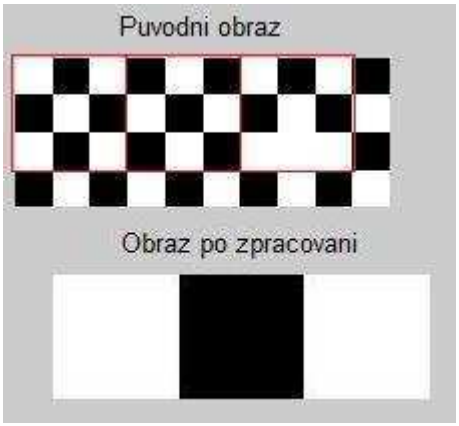
Pro potřebu výpočtu multifraktálního koeficientu je nutno rovnici přepsat do tvaru obecné rovnice přímky $y = k \cdot x + q$, kde člen $q$ lze zanedbat. Vznikne rovnice (2)

$$\log(N) = D \cdot \log\left(\frac{1}{r}\right), \tag{2}$$

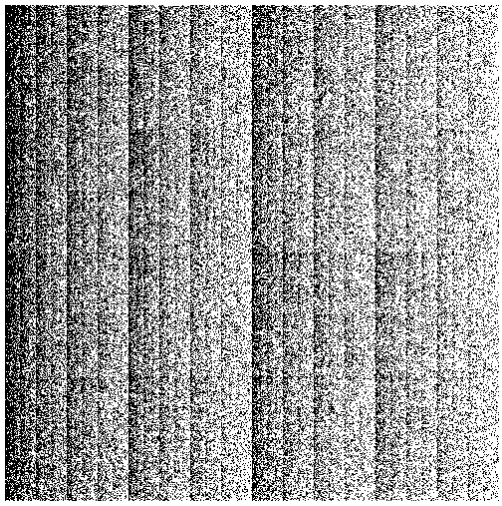
a pro multifraktální koeficient $F_k$ jako směrnice trendu vznikne výsledná závislost (3)

$$\log(N) = F_k \cdot \log\left(\frac{1}{r}\right). \tag{3}$$

Pro obrazy je pak *N* počet černých bodů po projetí obrazu maskou a *r* je velikost masky v pixelech. Postup při jednotlivých přiblíženích vysvětlují následující obrázky. Obrázek vlevo (Obrázek 12) ukazuje, jak se pixely přepočítají po projetí maskou o velikosti 3 x 3 pixely. Necelé přesahy obrazu se zanedbávají. Na obrázku vpravo (Obrázek 13) je sekvence ‚Homo sapiens ATPase, pro přenos $Cu^{2+}$, alfa polypeptid (ATP7A), na chromozomu 10‘, s lokusem NG_013224 a s délkou 146 699 bp, která bude ukázkově zpracována pomocí BCM. [7]

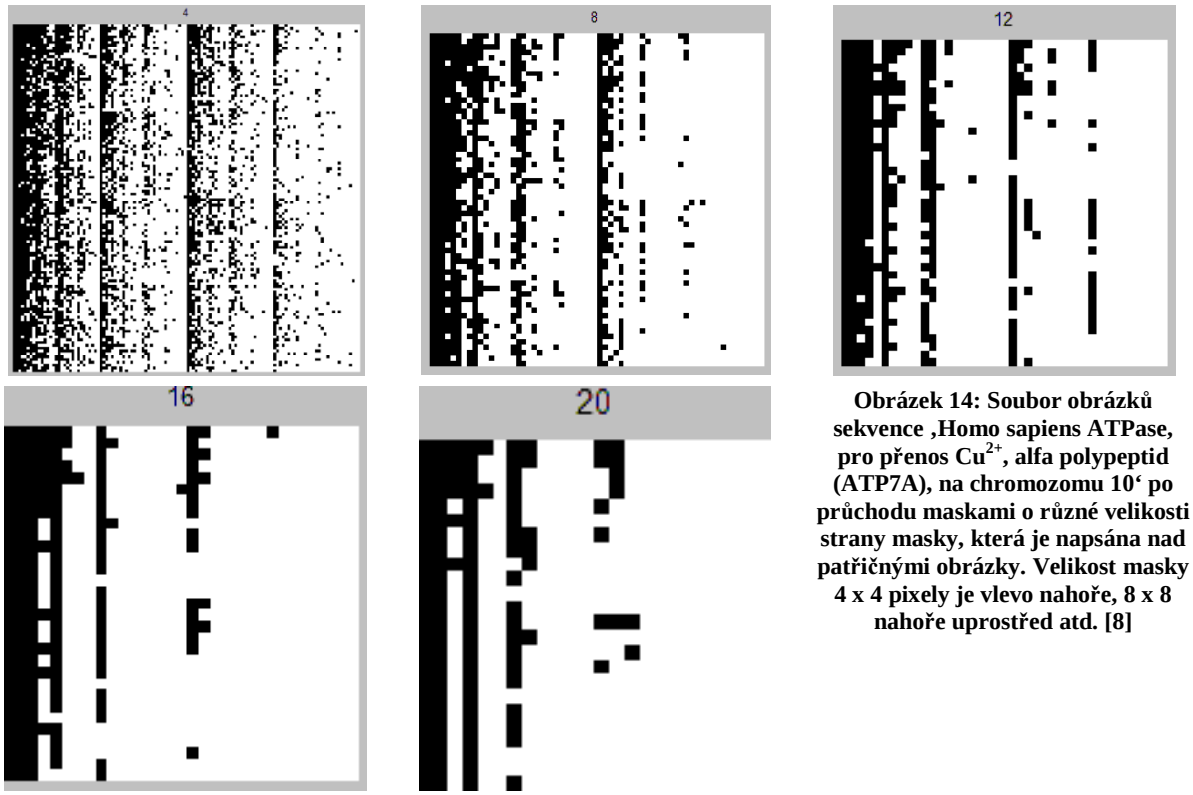



**Obrázek 12: Původní a následný obraz po aplikaci masky o velikosti 3 x 3 pixely. [8, generováno autorem v Matlabu]**



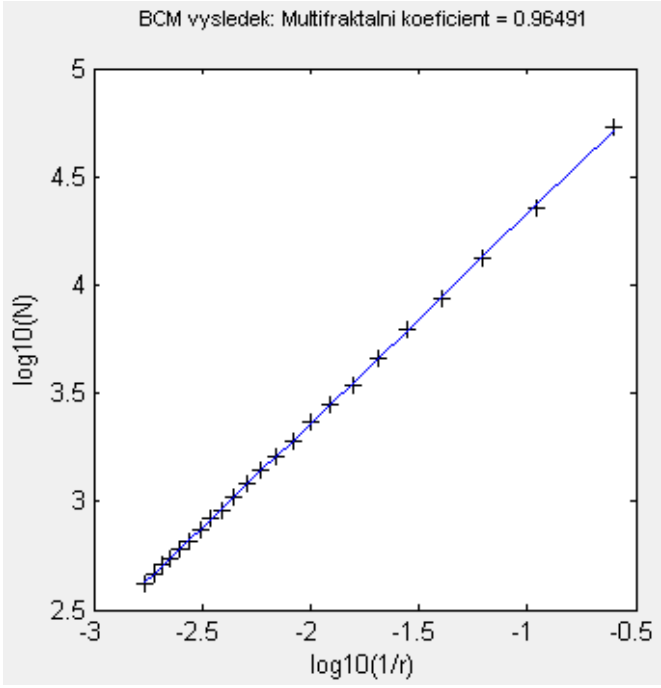
**Obrázek 13: Sekvence ‚Homo sapiens ATPase, pro přenos $Cu^{2+}$, alfa polypeptid (ATP7A), na chromozomu 10'. [8]**

U velikostí masky se sudým počtem pixelů pod ní, může nastat situace, kdy bude počet černých bodů pod maskou roven polovině plochy masky (polovině počtu pixelů). Zde pak program přisoudí následnému bodu barvu černou. Rozhodovací úroveň je nastavena na 0,49999, to znamená, že při větším poměru počtu černých bodů k velikosti masky, bude plocha pod maskou brána jako černá. Pokud bude poměr menší, pak jako bílá. Na následujících obrázcích (Obrázek 14) jsou zobrazeny výstupní obrázky zpracování metodou BCM sekvence NG_013224 na obrázku výše (Obrázek 13) po průchodu maskou o velikosti strany 4, 8, 12, 16 a 20 pixelů. [8]




**Obrázek 14: Soubor obrázků sekvence ‚Homo sapiens ATPase, pro přenos $Cu^{2+}$, alfa polypeptid (ATP7A), na chromozomu 10' po průchodu maskami o různé velikosti strany masky, která je napsána nad patřičnými obrázky. Velikost masky 4 x 4 pixely je vlevo nahoře, 8 x 8 nahoře uprostřed atd. [8]**

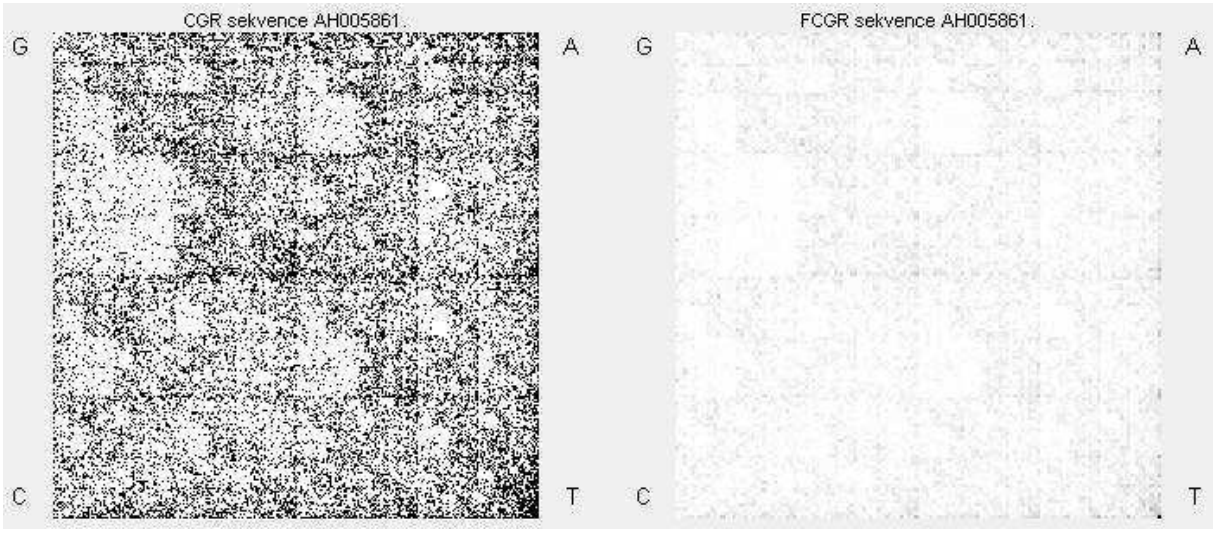
Po každém přiblížení dojde ke spočítání černých bodů a výpočtu dimenze podle rovnice (1). Spočítané dimenze vyneseme do grafu podle vztahu (3) a po jejich proložení přímkou, u které hledáme směrnici, se získá hledaný multifraktální koeficient. Pro dříve rozebranou sekvenci je výsledek pro $F_k$ = 0,96491 (Obrázek 15).




**Obrázek 15: Výsledný obraz BCM pro sekvenci ‚Homo sapiens ATPase, pro přenos $Cu^{2+}$, alfa polypeptid (ATP7A), na chromozomu 10'. [Výřez z programu GenomeFCGR]**

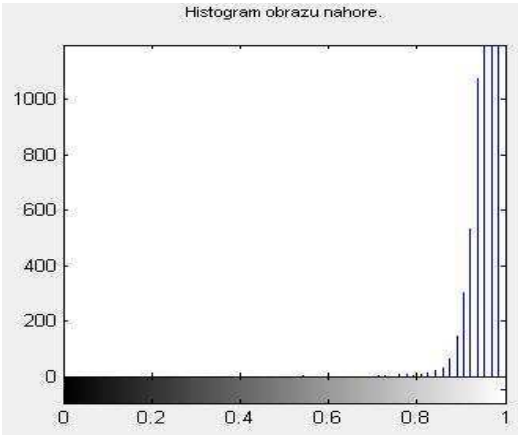
### 2.4.4. Frekvenční CGR

Frekvenční CGR, zkráceně FCGR, je jinak pojatá stejná metoda vykreslování jako u CGR (viz kap. 2.4.1. Chaos game reprezentace sekvence). Liší se v provedení, kdy si na počátku nadefinujeme nulový (prázdný) obraz a do jednotlivých pixelů postupně budeme přičítat dopady chodu programu. Po dopočítání celé sekvence se obraz upraví do rozsahu hodnot (0; 1) a musí se invertovat. Místo s největším počtem dopadů pak bude černé a místo bez dopadu bude bílé. Místa s menším počtem dopadů než je v černém bodě budou mít různý odstín šedi v závislosti na počtu dopadů. Vytvořenou matici pak lze přímo pokládat za obraz, lze ji zpracovávat a upravovat jako obraz. I celý způsob výpočtu je rychlejší, protože se jedná o matici hodnot, se kterou umí Matlab lépe pracovat. Na obrázku níže je výřez z vytvořeného programu, na kterém je stejná sekvence vypočtena metodou CGR a FCGR (Obrázek 16). Obraz FCGR je počítán do obrazu 128 x 128 pixelů.



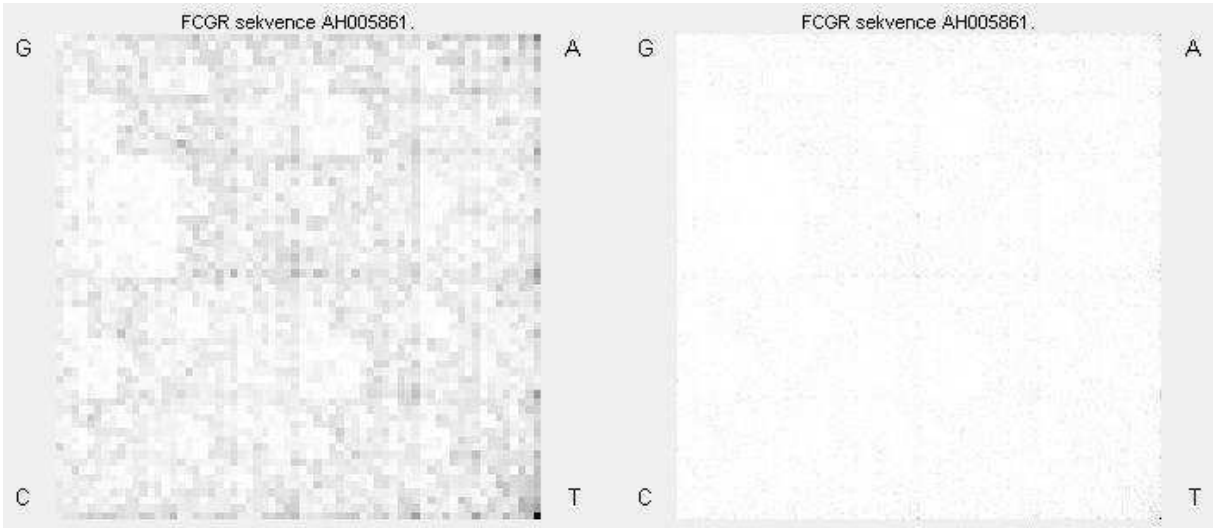

**Obrázek 16: Rozdíl ve vykreslení CGR a FCGR pro sekvenci ‚Lidský gen proteinu TJP2'. [Vlastní program GenomeFCGR]**

Toto zobrazení má také své nevýhody. Pokud bude velké množství dopadů pouze do jednoho místa, pak v obraze bude pouze jeden čistě černý pixel, a ostatní body budou natlačeny do světle šedé až do bílé barvy. Na následujícím obrázku (Obrázek 17) je ukázán tento případ pomocí histogramu obrazu výše (Obrázek 16 vpravo). Histogram je uzpůsoben tak, aby měl rozsah barev od nuly (černá) až po jedničku (bílá). Čistě černý bod může být jen jeden, proto na svislé ose nejde v této oblasti nic vidět. Svislá osa popisuje množství pixelů a jednotlivé čáry pak odpovídají množství pixelů na daném odstínu šedi.



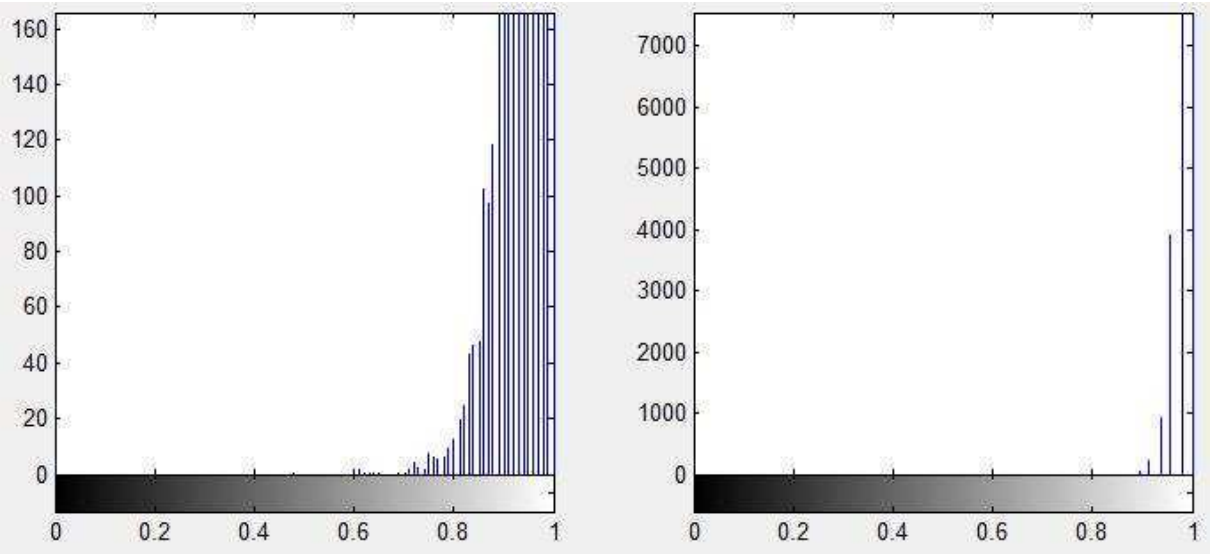

**Obrázek 17: Histogram obrazu pro Obrázek 16 vpravo.**

Dojem z obrazu sekvence je značně závislý na zvolené velikosti počítaného obrazu. Máme-li, jako na obrázku níže (Obrázek 18), dva obrazy stejné sekvence s různou velikostí strany obrazu, pak nemusí být mezi nimi ani podobnost.




**Obrázek 18: Rozdíl obrazů pro různou velikost FCGR vykreslení sekvence ‚Lidský gen proteinu TJP2'. Vlevo je obraz o velikosti 64 x 64 pixelů a vpravo je obraz o velikosti 256 x 256 pixelů. [GenomeFCGR]**

Změnou velikosti se změní nejen subjektivní vjem, ale i střední hodnota v obraze, což lze jasně vidět z histogramů těchto obrazů (Obrázek 19). Zde je jasně patrný posun čar směrem k bílé (1) u většího obrazu a na svislé ose se změnil počet pixelů s daným tónem šedi.




**Obrázek 19: Histogramy obrazů z Obrázku 16. [GenomeFCGR]**

Také musí dojít k úpravě BCM metody, kdy hodnoty se pod maskou opět sečtou, podělí se velikostí masky a rozhodování o následném bodu se provádí vzhledem ke střední hodnotě úrovně šedi. Právě plovoucí rozhodovací úroveň nedovoluje žádný zásah do obrazu, neboť by došlo ke změně počtu černých bodů po projetí metodou BCM a následnému ovlivnění celého multifraktálního koeficientu.

## 2.4.5. Další možnosti reprezentace sekvencí

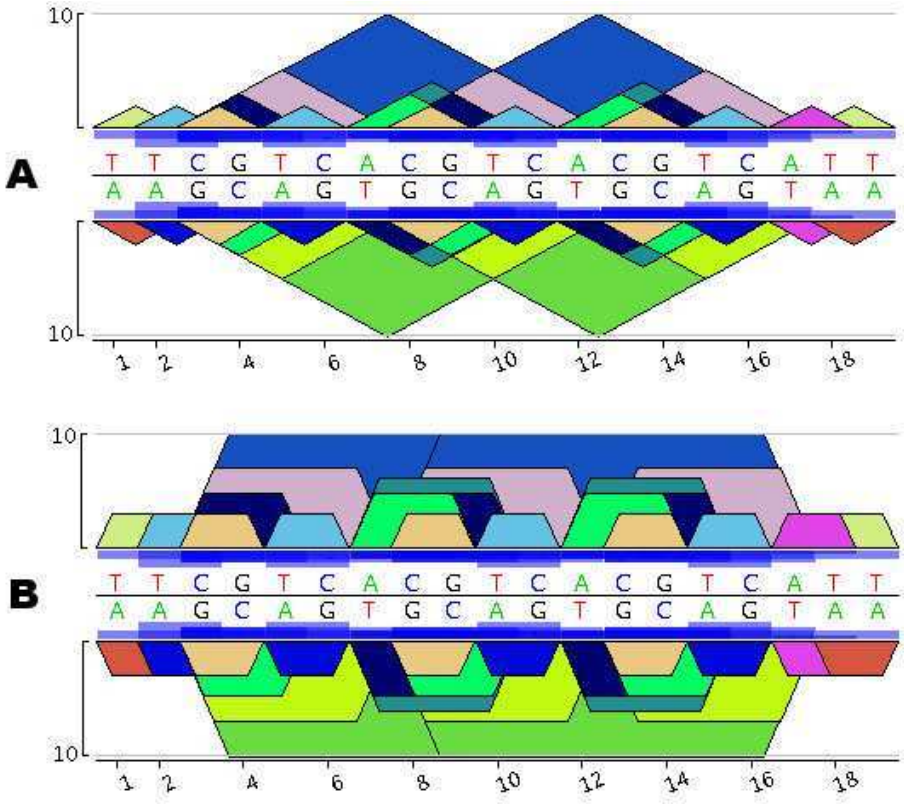
Další možnosti reprezentace genomických sekvencí v ploše jsou zde uvedeny pro doplnění. Dále v práci nebudou využity.

### 2.4.5.1. Pyramidový diagram

Pyramidové diagramy, zkráceně pygramy, se používají pro krátké sekvence. Pro sekvenci *S* o délce *N* vznikne dvojrozměrný obraz, kde *S* a všechny ‚přesné maxima opakování' (exact maximal repeat = EMR), jsou na ose *x*. Ose *x* náleží faktor *k* a ose *y* faktor *l*. Mapování se pak provádí podle definice: *i*-tý nukleotid z *S* je na souřadnici (*i/k*, 0); EMR o velikosti *m* odpovídá intervalu (*i/k*, *(i+m)/k*) na *i*-té pozici v *S*. Nad hledanou EMR o *m* se vytvoří pyramida (rovnoramenný trojúhelník) o výšce *δm/l*. *δ* nabývá hodnoty +1 pro normální vlákno a -1 pro opačné, komplementární vlákno. [16]

Pro odlišení jednotlivých EMR se požívají různé barvy. Shodné EMR mají stejnou barvu a to i v obou vláknech, to je v normálním i v komplementárním. Celý diagram je vždy symetrický podle osy *x*. Velikost každé pyramidy je dána množstvím výskytu EMR v sekvenci *S* a vrchol pyramidy je umístěn nad střed dané EMR. [16]

Příklad pygramu pro krátkou genomickou sekvenci ve tvaru: 5'-TTCGTCACGTCACGTCATT-3' je na obrázku níže (Obrázek 20). Nahoře je klasický pygram a pod ním s logaritmickou osou *y*. [16]

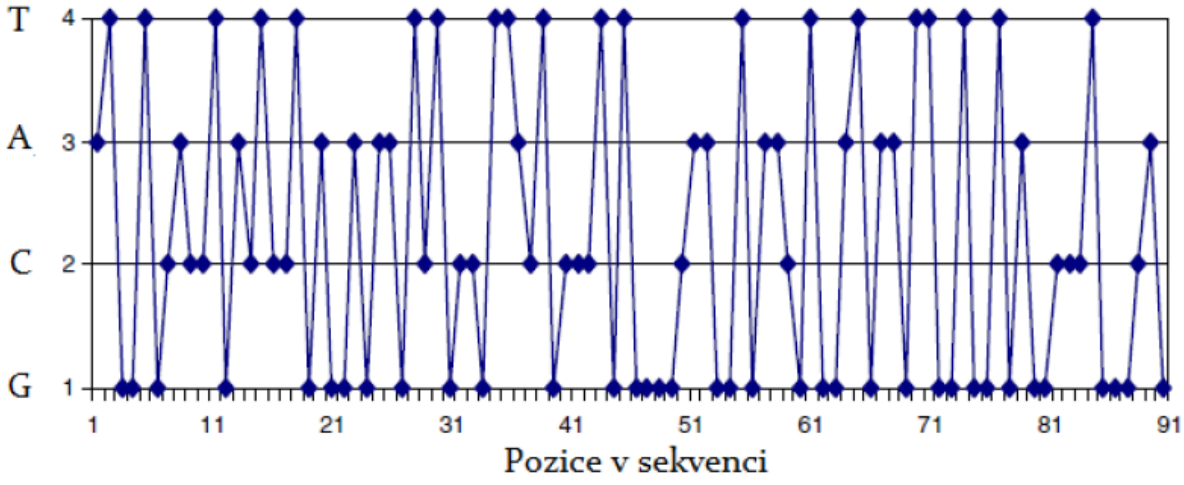



**Obrázek 20: Ukázka pyramidového diagramu pro krátkou DNA sekvenci. [16]**

#### 2.4.5.2. Spektrum like reprezentace

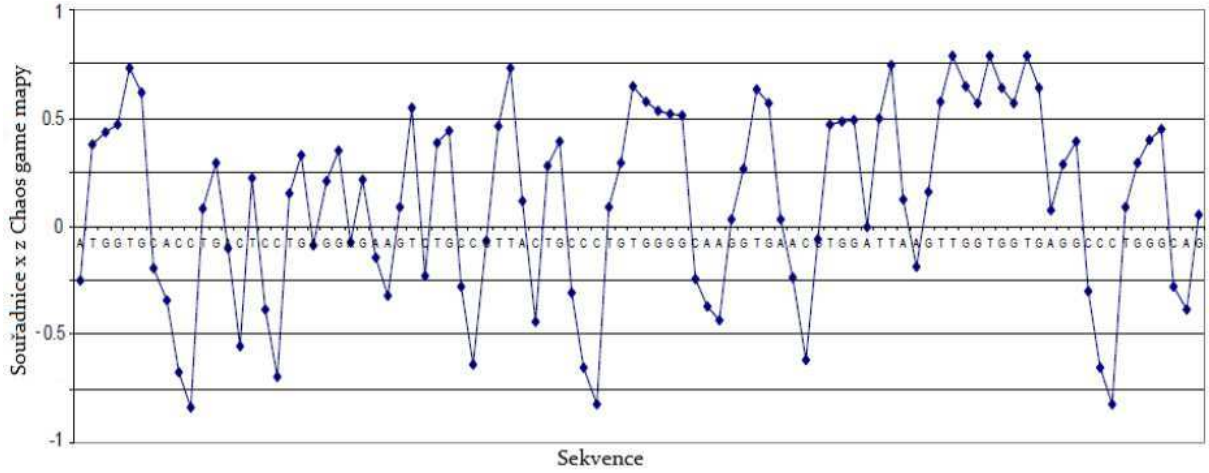
Základní spektrum like reprezentace je grafické vyjádření sekvence do plochy *xy*. Osa *x* nese informaci o pořadí dané báze v sekvenci a hodnota *y* je dána druhem báze – pro G je 1, pro C je 2, pro A je 3 a pro T je 4. Pro danou pozici s daným nukleotidem se zakreslí bod. Všechny body jsou pak spojeny křivkou. [17]

Příklad takové reprezentace je na obrázku níže (Obrázek 21). Je zde zobrazen první exon genu pro lidský β-globin. [13]




**Obrázek 21: Like reprezentace sekvence. [13]**

Další spektrum like je přepočet z 2D zobrazení sekvence pomocí chaos game. Je dán čtverec s vrcholy A(-1, -1), T(-1, 1), G(1, 1), C(1, -1) a počátkem iterací ve středu čtverce (v bodě (0, 0)). Po vykreslení sekvence do čtverce pomocí bodů, se postupně měří vzdálenosti na ose *x* k jednotlivým bodům a ty jsou pak použity jako hodnoty *y* pro vykreslení like spektra. Pod tímto textem je like spektrum sekvence pro první exon genu lidského β-globinu vykreslené pomocí popsaného postupu (Obrázek 22). [17, 13]




**Obrázek 22: Like spektrum sekvence přepočtené přes chaos game čtverec. [13]**

#### 2.4.5.3. Reprezentace podle S. Yau

Jde o další metodu, která sekvenci převede do plochy. Metoda nese jméno svého objevitele S. S. T. Yaua. Opět je použit souřadný systém *xy*, ve kterém jsou nadefinovány jednotkové vektory pro čtyři nukleotidy ve směrech ± 30° a ± 60° od kladné části osy *x*. Počátek iterací je opět v bodě (0; 0) a posloupnost bází určuje posun počátečního pro další iteraci. Vektory pro jednotlivé báze jsou rozepsány níže. [18]

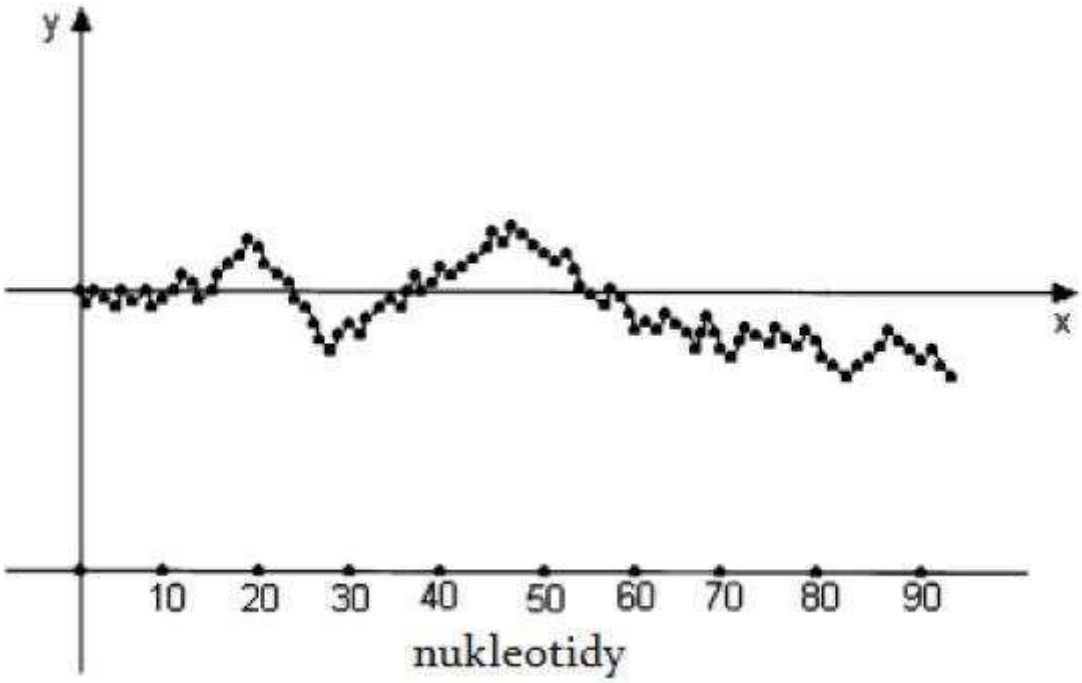
A: $\left(\frac{1}{2}; -\frac{\sqrt{3}}{2}\right)$

C: $\left(\frac{\sqrt{3}}{2}; \frac{1}{2}\right)$

G: $\left(\frac{\sqrt{3}}{2}; -\frac{1}{2}\right)$

T: $\left(\frac{1}{2}; \frac{\sqrt{3}}{2}\right)$

Křivka podle Yaua pro první exon genu lidského β-globinu je na obrázku níže (Obrázek 23). [13]




**Obrázek 23: Yauova křivka pro úsek DNA sekvence. [13]**

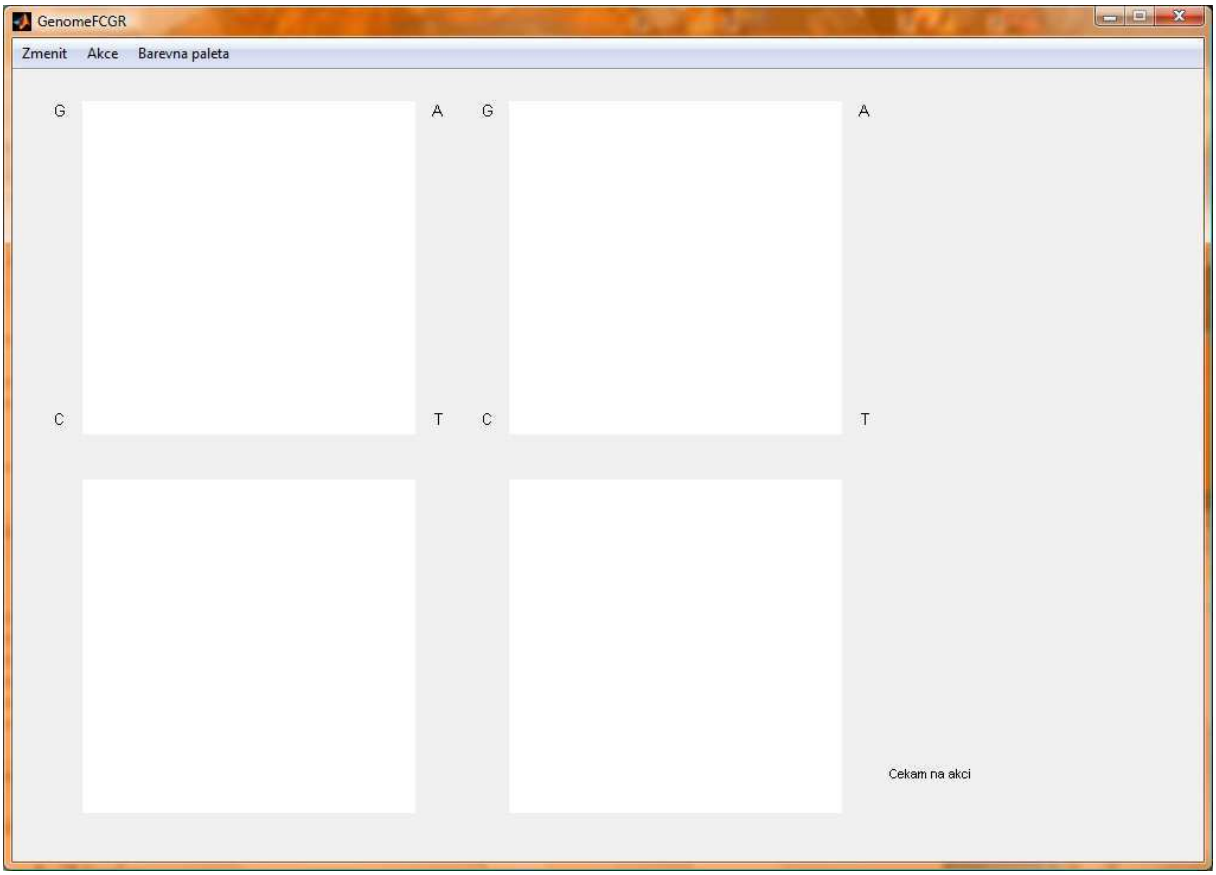
# 3. Sestavený program

Vytvořený program již zde byl několikrát zmíněn. Nese název *GenomeFCGR* a je navržen i s uživatelským rozhraním (GUI) pomocí *Matlabu* verze 7.10.0 (R2010a) s bioinformatickým toolboxem, ze kterého je použita funkce *getgenbank* pro načítání sekvencí z NCBI databáze. Na počátku této kapitoly bude program představen, dále budou rozebrány jeho funkce a nakonec bude předloženo několik výsledků, které byly pomocí něj získány. Na konci kapitoly budou doplněny zjištěné nedostatky a doporučení pro lepší činnost programu.

## 3.1. Program

Po spuštění programu *GenomeFCGR* se objeví windowsovské okno (Obrázek 24). Většinu plochy okna zabírají čtyři bílé plochy, kde se budou zobrazovat spočítané sekvence, anebo kde se bude zobrazovat jakákoli akce s obrazy. U horních dvou plošek jsou přednastaveny vrcholy, které lze změnit (viz 3.2.1. Změna velikosti obrazu). Předdefinovaná velikost obrazu spočtené metodou FCGR (více viz 2.4.4. Frekvenční CGR) je nastavena na 128 x 128 pixelů a lze ji také změnit (viz 3.2.2. Změna pořadí vrcholů čtverce). Vpravo dole se na šedém podkladě zobrazuje právě vykonávající akce, nebo rada či nápověda pro uživatele.

Na hlavním řádku lze vidět tři akční tlačítka. Pod tlačítkem *Zmenit* jsou možnosti změnit velikost a změnit pořadí vrcholů. Tlačítko Akce bude podrobně rozebráno v následující kapitole (3.2. Funkce programu). Poslední tlačítko mění barevnou škálu vykreslovaných FCGR obrazů pro lepší vnímání rozdílů. *Barevna paleta* má v sobě několik předvolených barevných palet, které jsou standardně využity v Matlabu. Mezi předvolené palety patří paleta *jet*, která má 0 reprezentovanou modrou barvou a směrem k 1 přechází přes zelenou, žlutou až po červenou. Druhá využitá je paleta *hot*, označena jako *Tepla*, která má 0 tmavě červenou, pak postupně světlá do oranžové až žluté a 1 je čistě bílá. Další v pořadí je paleta *cool*, označena jako *Studena*, kde je 0 zelenomodrá, pokračuje přes modrou až do fialové (1). Předposlední barevná paleta je *autumn*, označená jako *Podzimni*, kde je 0 tmavě červená a přechází přes oranžovou do žluté (1). Poslední položka vrátí změnu do šedotónové, to je ta, ve které standardně počítá celý program *GenomeFCGR*.




**Obrázek 24: Úvodní obrazovka programu *GenomeFCGR*. [GenomeFCGR]**

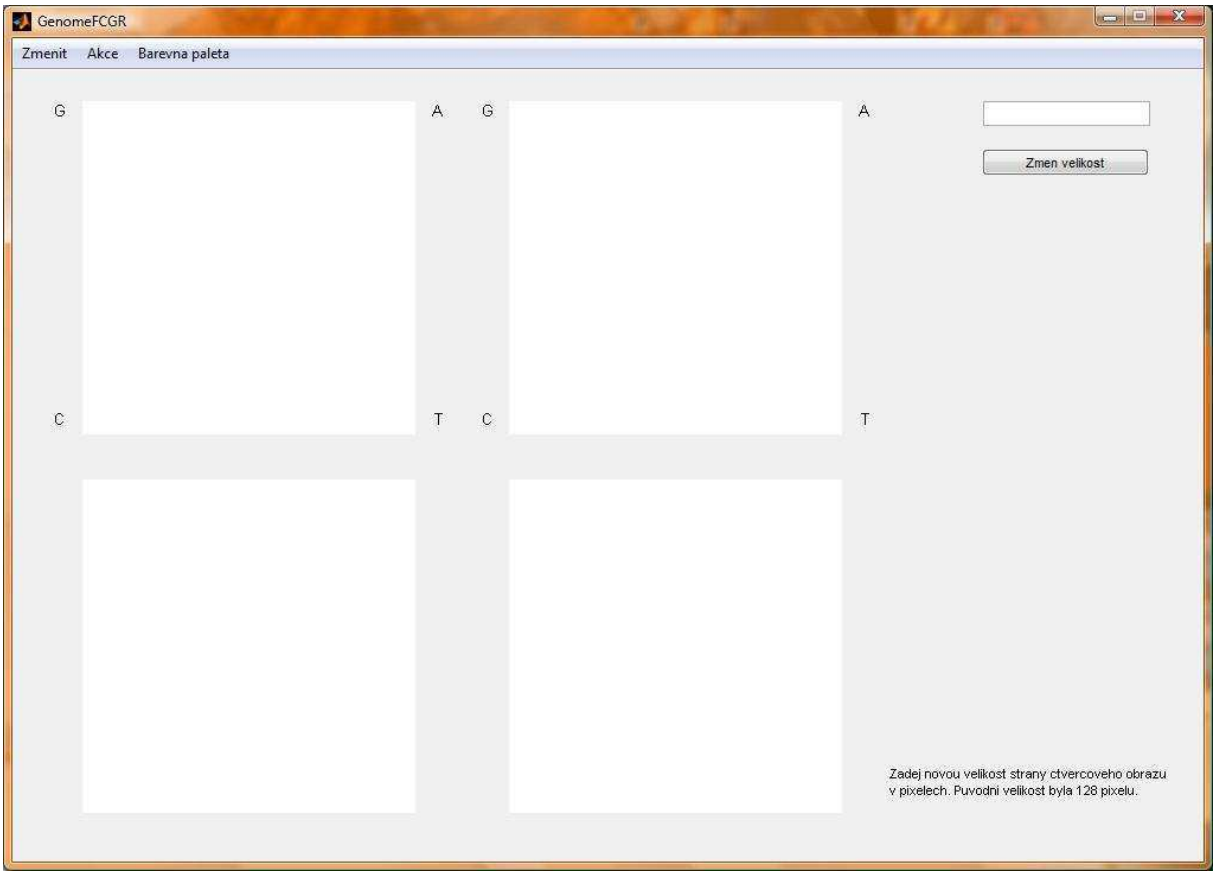
Program obsahuje i jakousi nápovědu, kdy už při najetí myší na danou položku se vpravo dole objeví popis, co tato položka umožňuje. Slouží pro lepší orientaci v programu a pochopení funkce, která je pod danou položkou skryta. Nápověda tady nebude nijak popsána. Ještě je potřeba poznamenat, že program byl programován na anglické klávesnici, a tak je zde potlačena diakritika.

## 3.2. Funkce programu

Zde budou postupně rozebrány všechny možnosti programu *GenomeFCGR*.

### 3.2.1. Změna velikosti obrazu

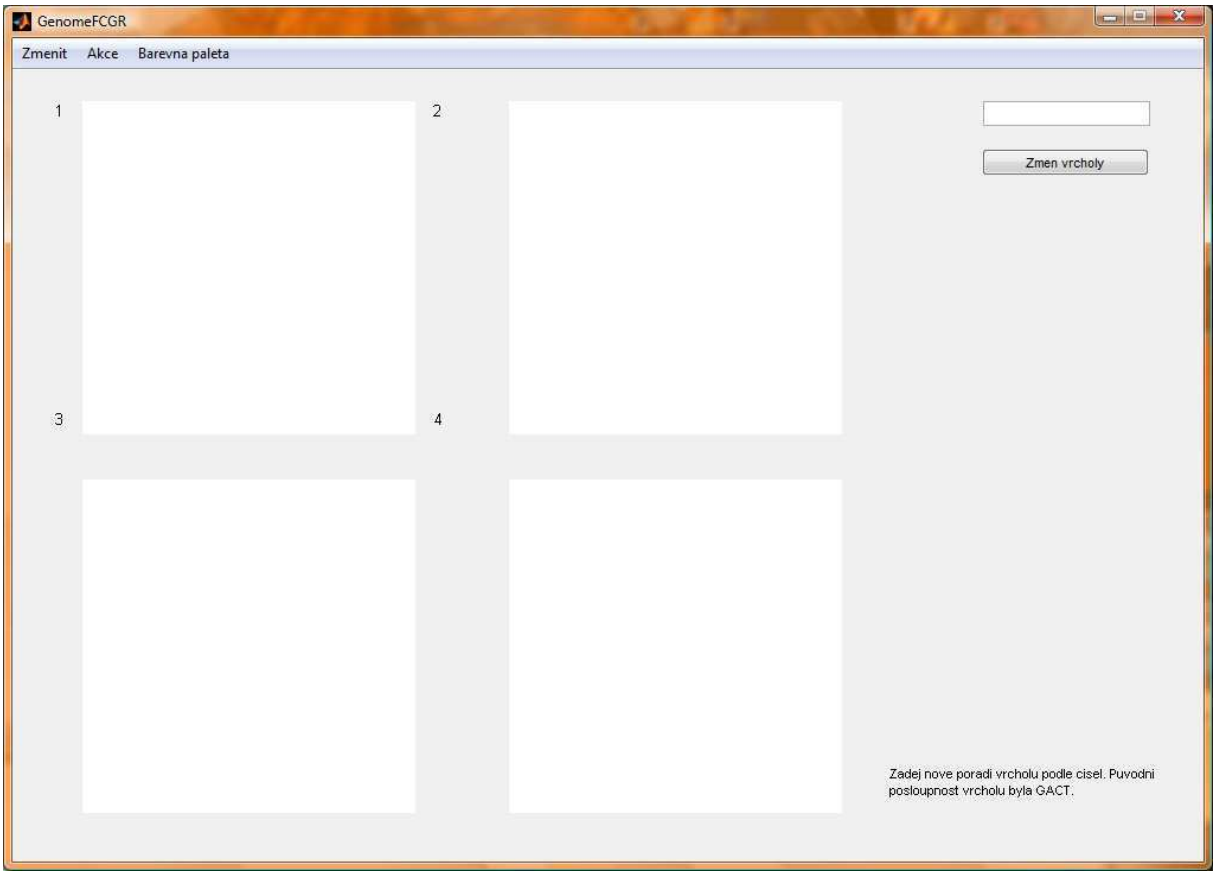
Změna velikosti obrazu je součástí položky *Zmenit*. Po kliknutí na *Zmenit* se objeví dvě volby a po vybrání položky *Velikost obrazu* se vpravo nahoře v okně objeví editovací okno s tlačítkem *Zmen velikost*. Vlevo dole se nápis změnil na následující: *Zadej novou velikost strany ctvercoveho obrazu v axelech. Puvodni velikost byla 128 pixelu*. Program čeká celočíselnou hodnotu počtu pixelů na stranu čtvercového obrazu. Program ošetřuje, zda zadané znaky jsou čísla. O úspěšnosti nebo neúspěšnosti změny velikosti program informuje textem vpravo dole. Vše, co zde bylo popsáno, je vidět na dalším obrázku (Obrázek 25).




**Obrázek 25: Okno programu *GenomeFCGR* při změně velikosti obrazu. [GenomeFCGR]**

### 3.2.2. Změna pořadí vrcholů čtverce

Opět je to součástí položky *Zmenit*. Po zvolení možnosti *Polohu vrcholu* se okno podobá oknu pro změnu velikosti obrazu, jen s rozdílným textem dole: *Zadej nove poradi vrcholu podle cisel. Puvodni posloupnost vrcholu byla GACT*. Čísla se zobrazí u levé horní bílé plochy (Obrázek 26). Program opět počká na zadání čtyř znaků vrcholů, ověří, zda jde o velké znaky A, C, G, nebo T, a správně zvolenou změnu odsouhlasí.




**Obrázek 26: Okno programu *GenomeFCGR* při změně pořadí vrcholů. [GenomeFCGR]**

### 3.2.3. Tlačítko akce

Tlačítko *Akce* umožňuje hlavní práci s programem. Obsahuje několik podseznamů, které patří k dané položce nad ní. První nabídka po jeho stisku je pouze dvoupoložková. Zde si uživatel vybere, zda chce obraz sekvence spočítat, nebo již existující obraz načíst do programu. Volba *Nacti obraz sekvence* je jasná a dává na výběr, zda načíst existující obraz do horního levého nebo pravého bílého pole. Po zvolení umístění opět vyskočí editovací pole a tlačítko s textem *Nacti*. Je potřeba zadat celý název chtěného obrazu sekvence i s příponou. V textu dole je povel, který také popisuje správný tvar názvu pro načtení obrazu.

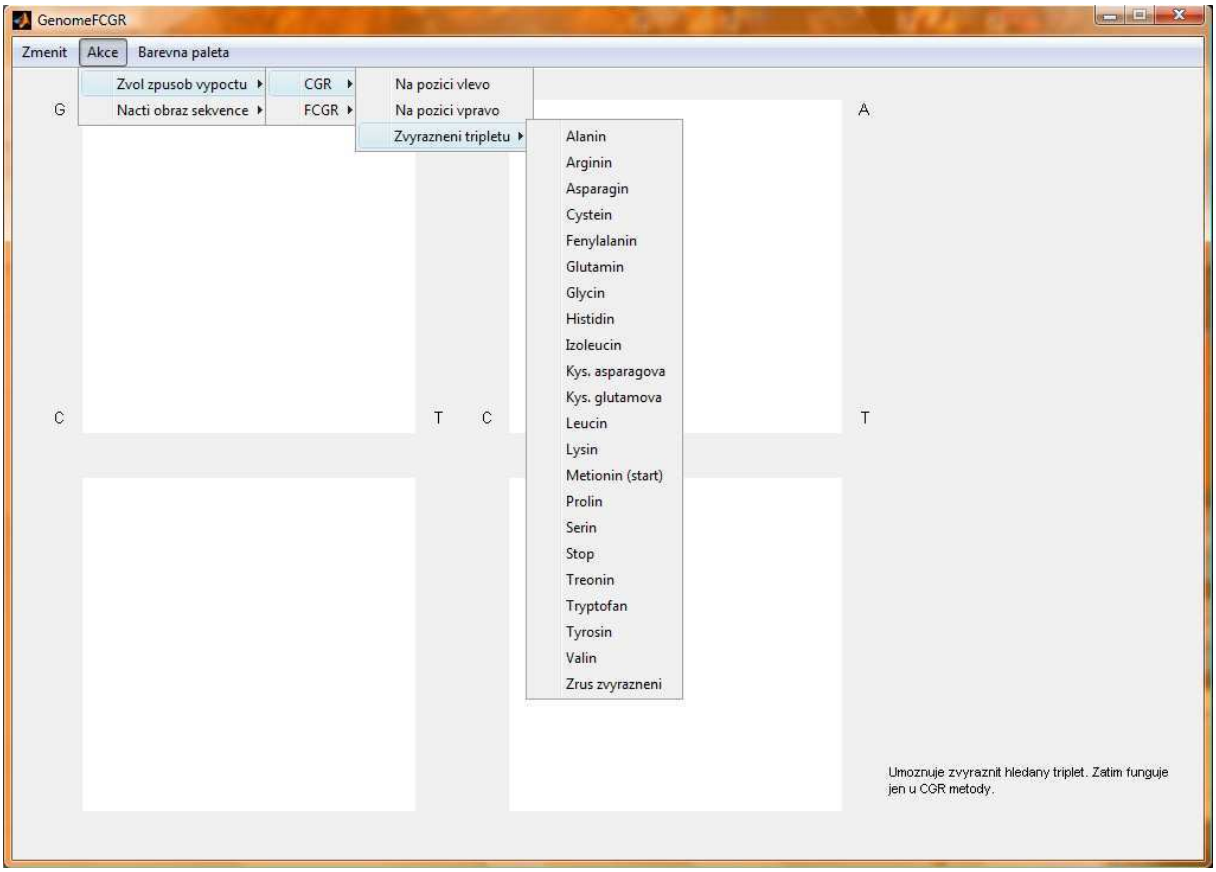
Volba *Zvol zpusob vypoctu* je nejrozsáhlejší v programu a shrnuje v sobě popsané metody z kap. 2.4. Popis DNA sekvencí, především CGR (2.4.1. Chaos game reprezentace sekvence), FCGR (2.4.4. Frekvenční CGR) a zpracování BCM (2.4.3. Box counting method). Za touto volbou (celá volba *Akce -> Zvol zpusob vypoctu*) je umístěna volba metody, která je opět dvojí – *CGR* nebo *FCGR*.

Při volbě spočtení sekvence se program nejdříve podívá do složky, zda sekvenci už někdy použil, a zda je uložena v *mat-filu*. Pokud program nenajde hledaný soubor, pokusí se připojit na internet a zkusí stáhnout hledanou sekvenci z veřejné databáze NCBI pomocí funkce z bioinformatického toolboxu *getgenbank*. Program stahuje pouze sekvenci. Po stažení

se sekvence sama uloží do *mat-filu* pro další použití, urychlení dalších výpočtů a práce se sekvencí.

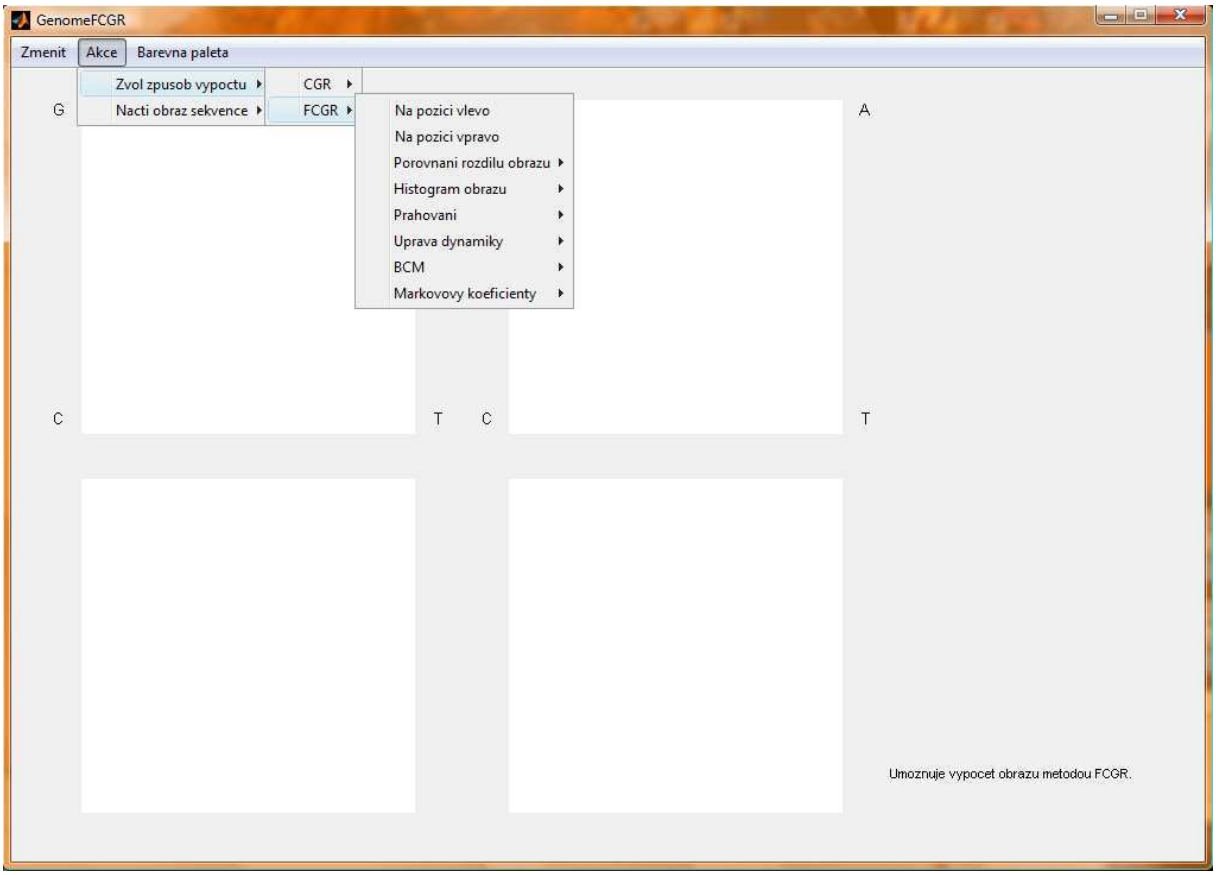
Program pro přehlednost vypisuje nad každý ze čtyř čtverců, co je v něm zobrazeno. Pro popis sekvencí užívá lokusy, což jsou jednoznačně identifikovatelné kódy, podle kterých lze na stránkách NCBI dohledat informace o dané sekvenci. Některé sekvence, se kterými zde autor pracuje, mohly být podrobeny změně nebo modifikaci od doby, kdy byly poprvé staženy.

Metoda *CGR* má v sobě opět volbu místa počítání (vpravo nebo vlevo) a navíc obsahuje volbu zvýraznění tripletu (viz kap. 2.4.2. Dopady tripletu), která zobrazí místo, kam dopadají body s potřebnou kombinací tří po sobě jdoucích znaků pro kódování dané aminokyseliny. Některé sekvence jsou značně dlouhé a výpočet může zabrat nemalý časový úsek, a proto je při výpočtu obrazu zobrazen průběh vpravo dole pomocí textu a procent výpočtu. Údaj pomáhá nejen odhadnout čas potřebný pro výpočet obrazu sekvence, ale také má zobrazovat funkčnost programu. V průběhu výpočtu jsou navíc ukládány hodnoty *x* a *y* souřadnic pro jednotlivé vypočtené body a to z toho důvodu, že je použita funkce *plot*, která umožňuje do sebe dále dokreslovat, ale mazat už ne. Dokreslování je využito při volbě libovolného tripletu. Pro smazání zvýraznění se musí celý obraz smazat a všechny body opětovně vykreslit do čisté bílé plochy. Toto znovu vykreslení bohužel zabere skoro stejný čas, jako předcházející výpočet, takže žádnou velkou časovou úsporu tato poslední položka mezi triplety neposkytuje (celá cesta: *Akce -> Zvol zpusob vypoctu -> CGR -> Zvyrazneni tripletu -> Zrus zvyrazneni*). Také při opětovném vykreslování je vpravo dole textem vypisován průběh pro lepší odhad času, který proces zabere. Nabídka za volbou *CGR* je na Obrázek 27.



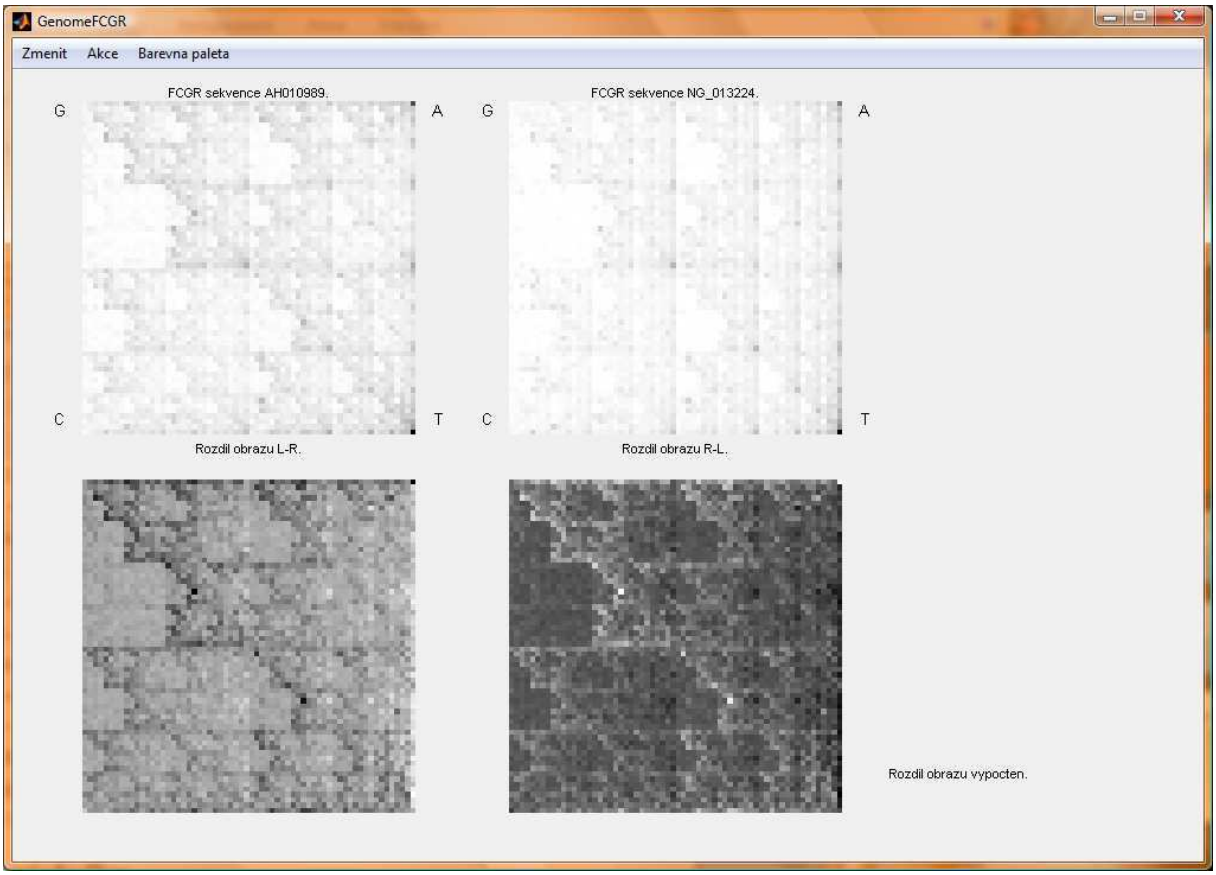

**Obrázek 27: Program *GenomeFCGR* – volba výpočtu *CGR*. [GenomeFCGR]**

Metoda *FCGR* nabídne opět možnost vykreslit obraz vlevo nebo vpravo, dále pak porovnání dvou obrazů FCGR, zobrazení histogramů, prahování, úprava dynamiky obrazu, BCM zpracování obrazu a výpočet markovových koeficientů. Nabídka za volbou *FCGR* je na Obrázek 28.



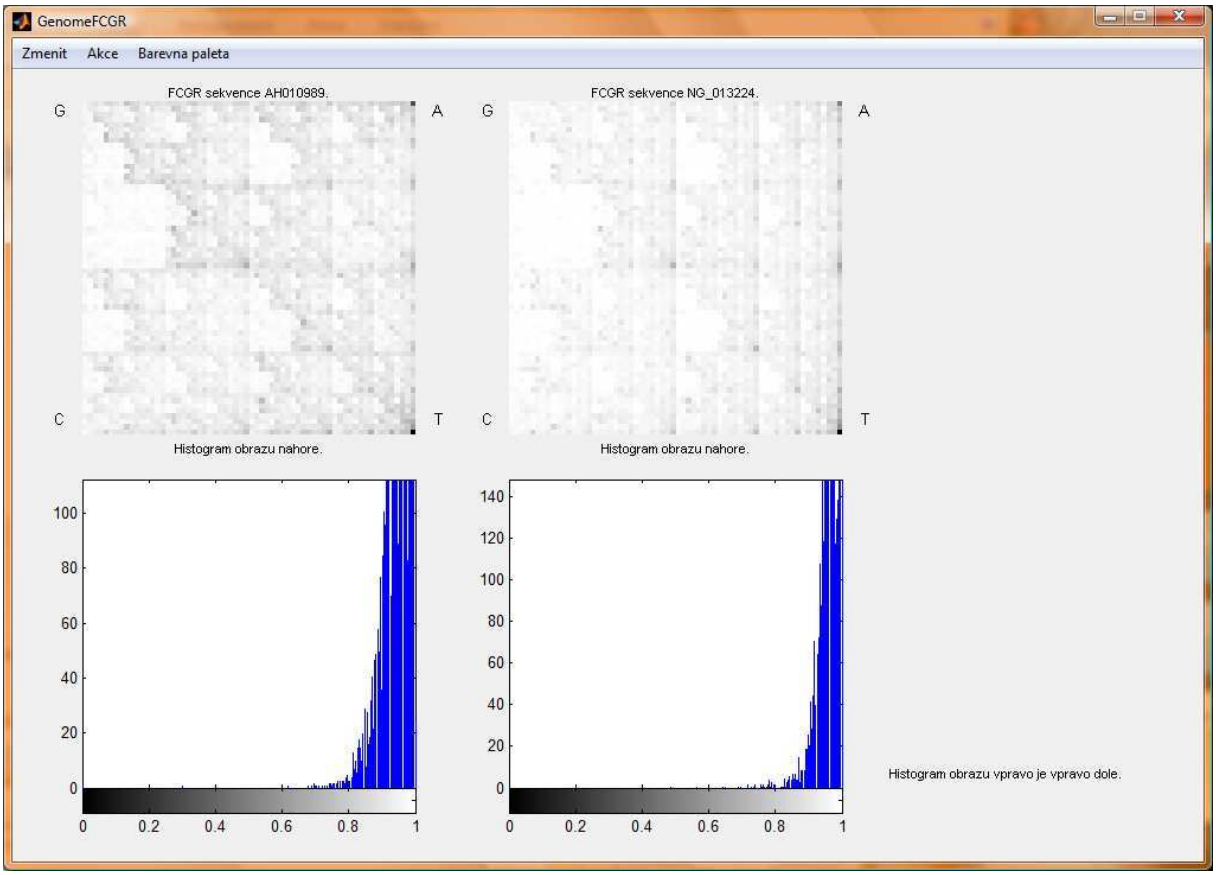

**Obrázek 28: Program *GenomeFCGR* – volba výpočtu *FCGR*. [GenomeFCGR]**

Výpočet metodou FCGR byl popsán v kap. 2.4.4. Frekvenční CGR a uživatel si vybere místo výpočtu (levý horní nebo pravý horní bílý čtverec). Porovnání dvou obrazů FCGR pod položkou *Porvnani rozdilu obrazu* (celá cesta: *Akce -> Zvol zpusob vypoctu -> FCGR -> Porovnani rozdilu obrazu*) zobrazí rozdíl dvou obrazů. V podnabídce je možnost výběru, kdy může být odečten pravý obraz od levého a výsledek je pak zobrazen vlevo dole, nebo naopak levý od pravého a výsledek pak bude vpravo dole. Pro objasnění celé situace je použito dvou sekvencí z databáze NCBI – první (v Obrázek 29 nahoře vlevo) je lidský gen FGFR2, nese označení lokusem AH010989 a má délku 25 008 bp. A druhá (v Obrázek 29 nahoře vpravo) je ‚*Homo sapiens ATPase, Cu++ transporting, alpha polypeptide (ATP7A) on chromosome X*‘, dána lokusem NG_013224 a délkou 139 699 bp. Pod nimi jsou zobrazeny patřičné rozdíly (Obrázek 29). Obrazy mají velikost 64 x 64 pixelů.



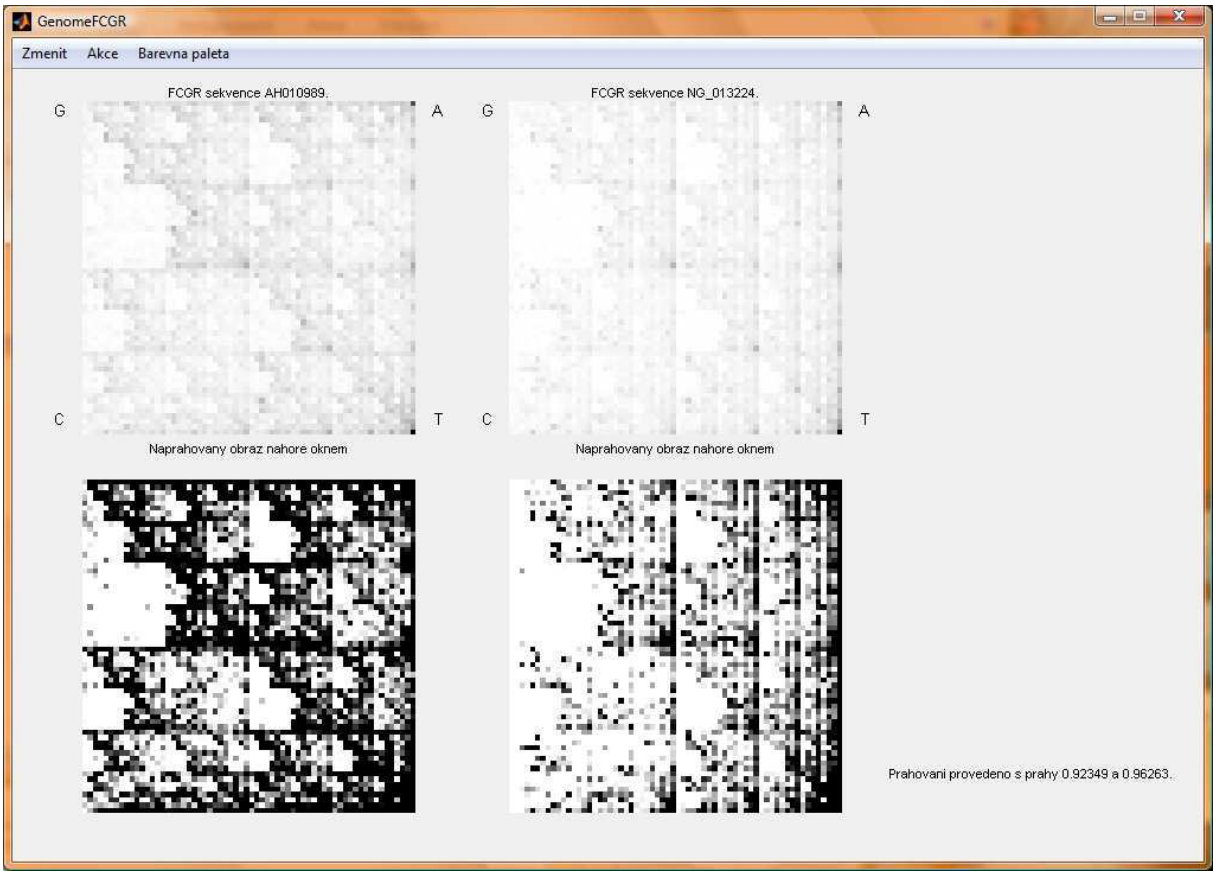

**Obrázek 29: Program *GenomeFCGR* – vykreslení rozdílu obrazů dvou sekvencí vykreslených metodou FCGR. [GenomeFCGR, Matlab]**

Další volbou při výběru metody FCGR je možnost zobrazení histogramů obrazů (celá cesta: *Akce -> Zvol zpusob vypoctu -> FCGR -> Histogram obrazu*). Je zde opět volba, která umožní vypočítat histogram pouze pro levý nebo pro pravý obraz. Pro výpočet obou histogramů musí uživatel vybrat tuto volbu opakovaně. Obrázek níže (Obrázek 30) zobrazuje program GenomeFCGR po vypočtení sekvencí AH010989 a NG_013224 metodou FCGR o velikosti strany čtvercového obrazu 64 pixelů (popis sekvencí je nad Obrázek 29), a po zobrazení jejich histogramů.



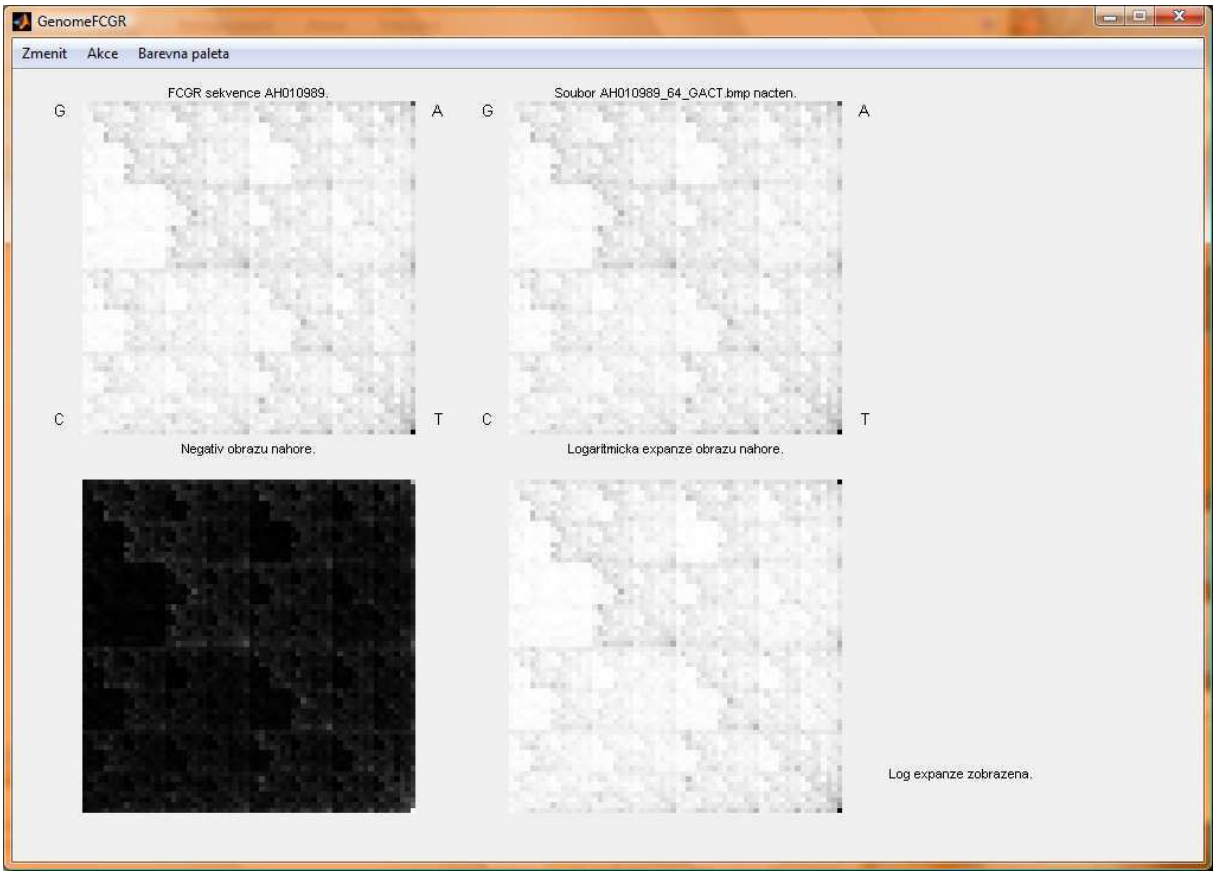

**Obrázek 30: Program *GenomeFCGR* – vykreslení histogramů obrazů dvou sekvencí vykreslených metodou FCGR. [GenomeFCGR, Matlab]**

Prahování je další volba v nabídce (celá cesta: *Akce -> Zvol zpusob vypoctu -> FCGR -> Prahovani*). Prahování je v programu GenomeFCGR dvojí – klasické s jednou hodnotou (označeno jako *Jednoduche*), nebo prahování pomocí okna (označeno jako *Okno*). Při obou volbách vykreslí program histogram obrazu vždy pod každý existující obraz sekvence, to znamená, že v případě jednoho obrazu zobrazí pod něj histogram, u dvou obrazů zobrazí oba histogramy, a v případě žádného obrazu bude uživatel upozorněn na nutnost nejdříve načíst nebo spočítat obraz sekvence. Pak uživatel při jednoduchém prahování zvolí kliknutím na levé tlačítko myši do libovolného histogramu obrazu práh, který určí, že hodnoty s tmavší barvou než má práh (menší hodnota než prahová) budou černé a body světlejší (větší hodnota) budou bílé. Hodnota prahu se vypíše textem vpravo dole a nad naprahovaným obrazem se objeví nadpis ‚*Naprahovany obraz nahore*'. Při volbě okna uživatel vybere opět kliknutím na levé tlačítko myši v histogramu obrazu dva body. Situace, kdy první kliknutí bude mít nižší hodnotu (bude blíže nule nebo černé) než druhé kliknutí, zobrazí výřez z obrazu, kde body s nižší hodnotou než je první klinutí budou černé, body s vyšší hodnotou než druhé kliknutí budou bílé a body mezi kliknutími budou roztaženy do celého rozsahu mezi černou a bílou (změní se odstín stupně šedi – viz Obrázek 31). V případě obráceného pořadí kliknutí dojde nejdříve k inverzi obrazu, což znamená, že bílé body budou černé a černé budou bílé, a zbytek postupu při zobrazení okna bude stejný. Zvolené prahy se vypíšou v textu vpravo dole.



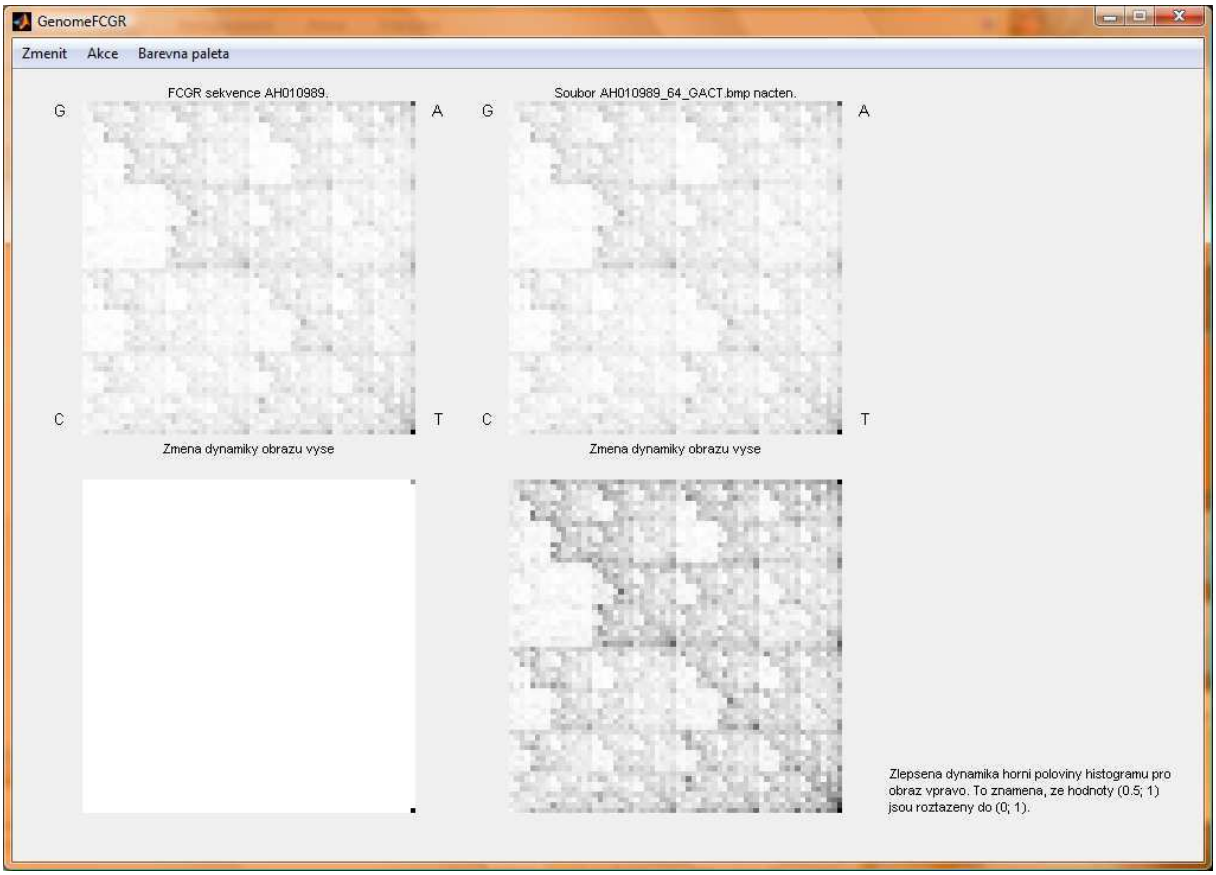

**Obrázek 31: Program *GenomeFCGR* – prahování oknem dvou obrazů sekvencí vykreslených metodou FCGR. [GenomeFCGR, Matlab]**

Následuje položka úprava dynamiky (celá cesta: *Akce -> Zvol zpusob vypoctu -> FCGR -> Uprava dynamiky*). Celá tato volba je určena pouze ke zlepšení vjemu nevýrazného obrazu sekvence zejména při delších sekvencích (nad milión bp) nebo při vykreslení většího obrazu (nad velikosti 256 x 256 pixelů). Při větší velikosti obrazu se zvyšuje počet světlých a bílých pixelů (viz Obrázek 18). Položka úprava dynamiky obsahuje čtyři operace – negativ, logaritmus, zvýraznění spodní poloviny histogramu a zvýraznění horní poloviny histogramu. Pro porovnání jednotlivých úprav bude použita jediná sekvence a to AH010989. Na obrázku dále (Obrázek 32) je vlevo dole zobrazen negativ obrazu sekvence, což je vlastně pouze aplikace vztahu pro výsledný obraz, který je roven jedné mínus odstínu šedi v původním obraze (výstup = 1 – vstup), a vpravo dole je logaritmická expanze obrazu (výstup = log10(vstup)).



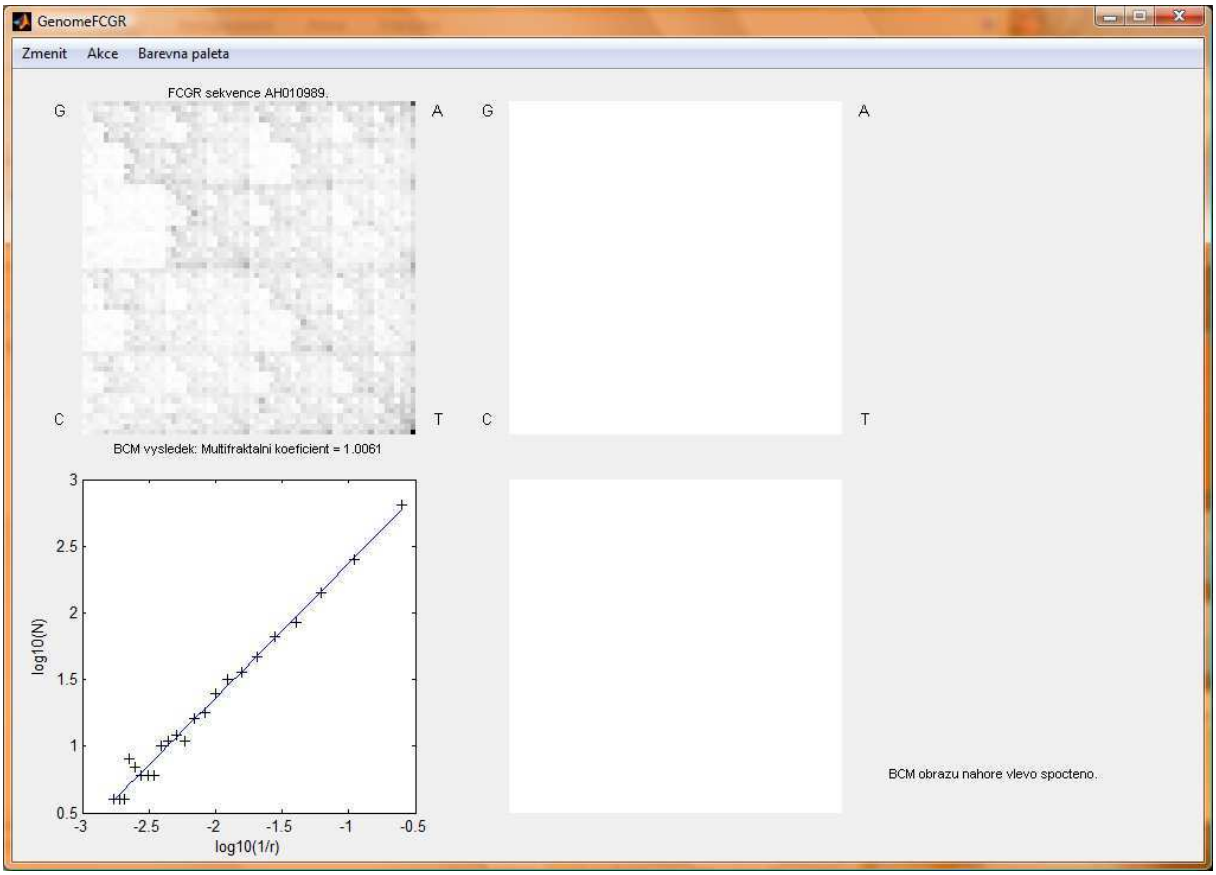

**Obrázek 32: Program *GenomeFCGR* – metoda FCGR – úprava dynamiky. Vlevo dole negativ a vpravo dole logaritmická expanze. [GenomeFCGR, Matlab]**

Další dvě volby v úpravě dynamiky jsou přednastavené prahy při prahování pomocí okna. Při zvýraznění spodní části histogramu se hodnoty nad 0,5 (50 % šedá) přebarví na bílo a rozsah 0 – 0,5 se roztáhne na 0 – 1 (na celý rozsah černá až bílá). Dojde vlastně k potlačení míst v obraze, kde je malý nebo žádný počet dopadů. Při zvýraznění horní části histogramu dojde k potlačení míst, kde je hodně dopadů (do 0,5), ale po roztažení (0,5 – 1 -> 0 – 1) jsou viditelná místa, kde ještě nějaký dopad byl a při prvním zobrazení byl zanedbán. Místa bez dopadu zůstanou stále bílá. Vše ukazuje obrázek níže (Obrázek 33). Vlevo dole je vidět, že body do hodnoty 0,5 jsou pouze dva. Toto vykreslení dostane smysl při větší velikosti obrazu (zde stále jen 64 x 64 pixelů). Každá změna dynamiky je nelineární a slouží jen k lepšímu vjemu obrazu sekvence. Pro samotný výpočet smysl nemá. Pokud by se počítalo s takto upraveným obrazem sekvence, zaneslo by to značnou chybu do výpočtu jejího multifraktálního koeficientu.




**Obrázek 33: Program *GenomeFCGR* – metoda FCGR – úprava dynamiky. Vlevo dole zlepšena dynamika dolní poloviny histogramu a vpravo dole zlepšena dynamika horní poloviny histogramu. [GenomeFCGR, Matlab]**

Předposlední položkou ve výpočtu pomocí metody FCGR je zpracování obrazu metodou BCM popsané v kap. 2.4.3. Box counting method. Tato metoda dává smysl celému programu. Výpočtem z obrazu se dostane hodnota multifraktálního koeficientu, na jehož základě lze numericky srovnat dvě sekvence. Číselné hodnoty pro různé sekvence jsou shrnuty v následující kapitole (3.3. Získané výsledky). Obrázek níže ukazuje program po výpočtu koeficientu metodou BCM (Obrázek 34) pro sekvenci AH010989. Hledaný multifraktální koeficient je vypsán nad zobrazeným výsledkem BCM.




**Obrázek 34: Program *GenomeFCGR* – metoda FCGR – výpočet multifraktálního koeficientu metodou BCM. [GenomeFCGR, Matlab]**

Poslední položka slouží pro výpočet Markovových koeficientů, které lze využít pro simulaci a následné předpovídání chování sekvence. Vypočtené koeficienty se vypíší do *Command Window* v Matlabu a v běhu programu se zobrazí obraz koeficientů jako obraz FCGR o velikosti 8 x 8 pixelů. [11]

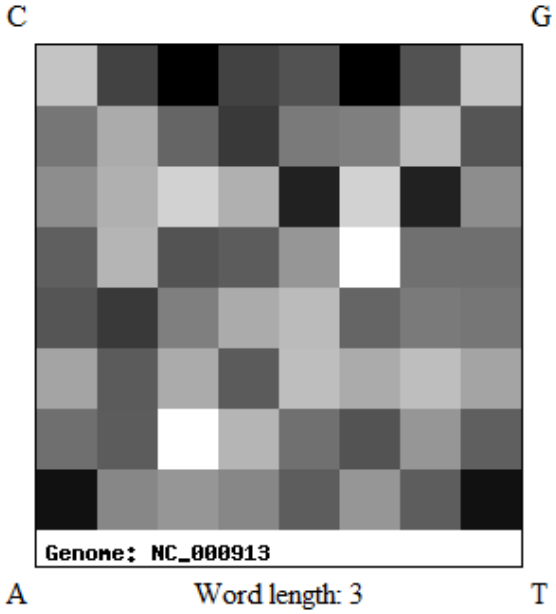
## 3.3. Získané výsledky

V této kapitole bude nejdříve ověřena platnost spočítaných obrazů sekvencí a následně pak budou spočteny vybrané sekvence. Na konci kapitoly budou dodány tipy na vylepšení činnosti programu.

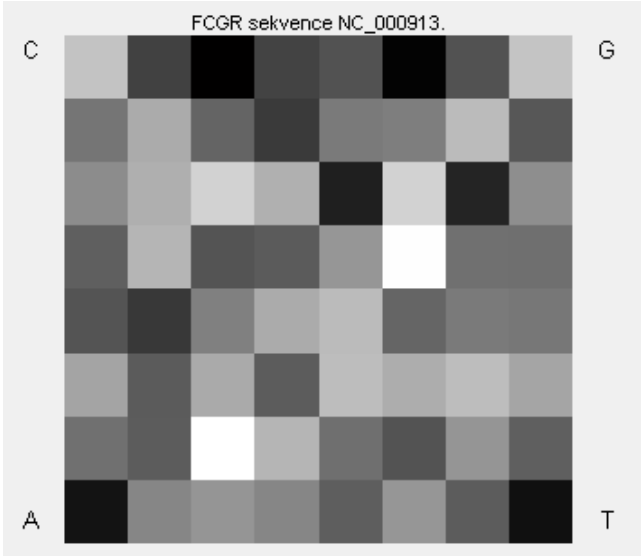
### 3.3.1. Ověření správnosti převodu sekvence

Ověření správnosti bude provedeno pohledovým porovnáním dvou obrazů téže sekvence vykreslených metodou FCGR (více viz kap. 2.4.4. Frekvenční CGR), kde jako srovnávací obraz byl nalezen obraz ve zdroji [10]. Je použita sekvence ‚Escherichie coli str. K-12 substr. MG1655 chromosomu, kompletní genomická sekvence', nesoucí lokus NC_000913 a mající délku 4 639 675 bp. Porovnání obrázků je pod textem. Vlevo je obrázek sekvence ze zdroje [10] (Obrázek 35) a vedle je obrázek spočten programem *GenomeFCGR* (velikost 8 x 8 pixelů, výřez z okna programu, Obrázek 36). Program se musí přenastavit

podle vzorového obrazu, to znamená, že je nutno změnit velikost počítaného obrazu na 8 x 8 pixelů (viz kap. 3.2.1. Změna velikosti obrazu) a je nutno upravit polohu vrcholů na posloupnost CGAT (viz kap. 3.2.2. Změna pořadí vrcholů čtverce).




**Obrázek 35: Obraz sekvence E. coli. [10]**




**Obrázek 36: Obraz sekvence E. coli spocitany programem *GenomeFCGR*. Výřez z okna programu. [GenomeFCGR]**

Z porovnání obrázků výše je patrné, že program *GenomeFCGR* dává stejný obraz pro sekvenci E. coli jako ve zdroji [10]. Lze tedy tvrdit, že program vykresluje metodu FCGR správně, nebo shodně jako program ve zdroji [10].

### 3.3.2. Vybrané sekvence

V tabulce na další straně (Tabulka 2) jsou vypočteny hodnoty multifraktálního koeficientu $F_k$ pomocí BCM (viz kap. 2.4.3. Box counting method) pro vybrané sekvence z veřejné databáze NCBI. Obrazy jsou počítány se základním nastavením, to je s velikostí obrazu 128 x 128 pixelů a posloupností vrcholů GACT. Všechny spočtené obrazy i s jejich BCM výsledky jsou na přiloženém CD. Vybrané sekvence (lokus je jednoznačný identifikátor):

Sekvence 1: Citrobakterie ATCC 29220, lokus NZ_ABWL02000007, délka 274 831 bp
Sekvence 2: Lidská ATPáza, měďnatý přenašeč a alfa polypeptid ATP7A, lokus NG_013224, délka 146 699 bp
Sekvence 3: Escherichia coli, lokus NC_000913, délka 4 639 675 bp
Sekvence 4: Lidský 13. chromozóm, lokus NT_027140, délka 1 821 999 bp
Sekvence 5: Arabidopsis thaliana chromozóm 1, lokus AE005173, délka 14 668 883 bp
Sekvence 6: Lidský gen FGFR2, lokus AH010989, délka 25 008 bp
Sekvence 7: Lidský gen proteinu TJP2, lokus AH005861, délka 26 841 bp

Sekvence 8: Lidský chromozóm Y, lokus NT_167199, délka 34 821 bp
Sekvence 9: Lidský tumor potlačující protein LRP1B, lokus AH011233, délka 81 393 bp
Sekvence 10: Gen myši domácí Egfr, lokus AH009944, délka 56 717 bp
Sekvence 11: Protein myši domácí Ap3b1, lokus AH009888, délka 35 113 bp
Sekvence 12: Gen oreochromis niloticus KLR1, lokus AH013711, délka 170 520 bp
Sekvence 13: Lidský preprokolagen alfa COL5A2, lokus AH011142, délka 103 859 bp

**Tabulka 2: Vybrané sekvence z NCBI. [Spočítáno v GenomeFCGR]**

| Číslo sekvence | Lokus sekvence | $F_k$ (výstup BCM) |
|---|---|---|
| 1 | NZ_ABWL02000007 | 0,98349 |
| 2 | NG_013224 | 0,99028 |
| 3 | NC_000913 | 0,97987 |
| 4 | NT_027140 | 0,97916 |
| 5 | AE005173 | 0,99120 |
| 6 | AH010989 | 0,96908 |
| 7 | AH005861 | 0,96481 |
| 8 | NT_167199 | 1,03060 |
| 9 | AH011233 | 0,99406 |
| 10 | AH009944 | 0,97792 |
| 11 | AH009888 | 0,97922 |
| 12 | AH013711 | 0,96946 |
| 13 | AH011142 | 0,97858 |

Při pohledu na hodnoty multifraktálních koeficientů při stejném nastavení výpočtu pro různé sekvence jsou různé. Lze jej z tohoto pohledu použít jako kvalifikátor a na jeho základě lze sekvence srovnávat.

Další zpracovávané sekvence jsou sekvence celých lidských chromozómů. Z důvodů dlouhých sekvencí byly obrazy spočteny na školním serveru s pomocí vzdáleného přístupu. Je potřeba mít dostatečnou velikost paměti RAM (na školním serveru bylo k dispozici až 12 GB RAM paměti). Sekvence nejdou přímo načíst do programu právě kvůli své velikosti, tak byly staženy ručně položkou v NCBI *send to file* a uloženy ve tvaru *genbank*. Následné zpracování bylo provedeno skriptem *upravasekvenci.m*, která má v sobě použitou funkci *prevodnaobraz*, která sekvenci převede na obraz. Ve skriptu je potřeba na čtvrtém řádku zadat lokus sekvence, dojde k jejímu otevření ze tvaru struktury s příponou *.gb* a načtení sekvence do souboru *.mat*. Následně se vypočtou dva obrazy pomocí dříve zmíněné funkce s velikostí 512 x 512 pixelů a 1024 x 1024 pixelů. Skript i funkce jsou přiloženy na CD. Menší obrazy nebyly počítány, tak v obrázcích není informace dobře viditelná. Tyto velikosti byly zvoleny vzhledem k délkám

sekvencí a z důvodů předpokladu, že při větším obrazu bude méně zaokrouhlovacích chyb. Budou zde porovnány multifraktální koeficienty v závislosti na velikosti obrazu (Tabulka 3).

**Tabulka 3: Zpracování celých lidských chromozómů. [Výsledky z GenomeFCGR]**

| Číslo chromozomu | Lokus sekvence | $F_k$ (512x512) | $F_k$ (1024x1024) |
|---|---|---|---|
| 1 | CM000462 | 1,0497 | 0,99538 |
| 2 | CM000463 | 1,0696 | 0,99066 |
| 3 | CM000464 | 1,0696 | 0,99076 |
| 4 | CM000465 | 1,0747 | 0,98727 |
| 5 | CM000466 | 1,0724 | 0,98928 |
| 6 | CM000467 | 1,0679 | 0,99067 |
| 7 | CM000468 | 1,0516 | 0,99457 |
| 8 | CM000469 | 1,0703 | 0,98942 |
| 9 | CM000470 | 1,0563 | 0,99422 |
| 10 | CM000471 | 1,0566 | 0,99375 |
| 11 | CM000472 | 1,0722 | 0,98987 |
| 12 | CM000473 | 1,0457 | 0,99581 |
| 13 | CM000474 | 1,0733 | 0,98760 |
| 14 | CM000475 | 1,0548 | 0,99440 |
| 15 | CM000476 | 1,0396 | 0,99735 |
| 16 | CM000477 | 1,0064 | 1,00100 |
| 17 | CM000478 | 0,9730 | 1,00308 |
| 18 | CM000479 | 1,0732 | 0,98762 |
| 19 | CM000480 | 0,9663 | 1,00441 |
| 20 | CM000481 | 1,0413 | 0,99616 |
| 21 | CM000482 | 1,0602 | 0,99156 |
| 22 | CM000483 | 0,9882 | 1,00141 |
| X | CM000484 | 1,0793 | 0,98647 |
| Y | CM000485 | 1,0784 | 0,98445 |

Tabulka 3 srovnává multifraktální koeficienty pro jednotlivé chromozomy a pro dvě spočítané velikosti. Výjimečná shoda u druhého a třetího chromozómu je náhodná – změna může být na dalším desetinném místě, které zde není zobrazeno. Další možnost shody nebo blízké hodnoty může nastat podobností samotných chromozómů (viz kap. 0.

Běžný popis DNA sekvencí, nebo také genomických sekvencí, se většinou děje podle délky sekvence, podle chromozomu, ze kterého byla sekvence získána, nebo podle druhu, ze

kterého sekvence pochází. Další popis sekvence bude rozebrán v kap. 2.4. Popis DNA sekvencí.

Chromozom). Opět by se dal multifraktální koeficient použít jako kvalifikátor.

Další tabulka (Tabulka 4) zkoumá celkovou závislost na velikosti pro dvě vybrané sekvence. Použité sekvence budou sekvence 2 a 12 popsané na počátku kap. 3.3.2. Vybrané sekvence. Položka 128 má za sebou v závorce napsáno standard, který označuje výchozí nastavení programu *GenomeFCGR*.

**Tabulka 4: Vliv velikosti obrazu na velikost multifraktálního koeficientu. [GenomeFCGR]**

| Velikost strany obrazu | NG_013224 | AH013711 |
|---|---|---|
| 32 | 1,06698 | 1,08914 |
| 64 | 1,02611 | 1,02700 |
| 128 (standard) | 0,99028 | 0,96946 |
| 256 | 0,97326 | 0,96830 |
| 512 | 0,96491 | 0,97028 |

Opět je jasné viditelné, že pro vytvoření systému pro klasifikaci sekvencí je potřeba přesně definovat podmínky výpočtu. U první sekvence (NG_013224) by se možná našel trend, se kterým koeficient klesá, ale u druhé sekvence (AH013711) se již trend stejný nalézt nedá. K chybě taky přispívá výpočet jednotlivých dimenzí metodou BCM (viz kap 2.4.3. Box counting method).

Poslední parametr, který ovlivní spočtený multifraktální koeficient, je pořadí vrcholů popsaných okolo čtverce. Samozřejmě to změní i vrcholy v běhu programu. Posloupnosti jsou uvedeny tak, jak se zadávají do programu *GenomeFCGR*. V tabulce níže (Tabulka 5) je ukázáno, jak pořadí vrcholů ovlivní výpočet. Opět budou použity dvě sekvence použité u předcházející tabulky a budou vypočteny do obrazu o velikosti 64 x 64 pixelů.

**Tabulka 5: Vliv pořadí vrcholů na velikosti multifraktálního koeficientu. [GenomeFCGR]**

| Posloupnost vrcholů | NG_013224 | AH013711 |
|---|---|---|
| GACT (standard) | 1,02611 | 1,02700 |
| CGAT | 1,03907 | 1,01389 |
| ACGT | 1,04071 | 1,05222 |

Zde je změna pochopitelná. Hlavní část změny nese výpočet jednotlivých dimenzí pomocí metody BCM (viz kap 2.4.3. Box counting method). Zde je hlavním problémem ono zanedbání přesahu obrazu při necelém násobku velikosti strany masky ku velikosti strany

obrazu. Jediný smysl výměny vrcholů je pro jiný vjem téhož obrazu. Některé body sice mohou při běhu programu vlivem zaokrouhlovacích chyb padnout do sousedního pixelu, ale to má na výsledný multifraktální koeficient jen zanedbatelný vliv.

## 3.4. Návrhy na vylepšení programu

Při používání programu bylo nalezeno několik nedostatků, které by mohly bát v budoucí verzi napraveny.

### 3.4.1. Načítání

Při načítání sekvencí je potřeba zadat celý název. Šikovnější by bylo ukládat sekvence do vlastní složky, třeba s názvem sekvence. A pak v chodu programu by vyskočilo okno s možností výběru chtěné sekvence pomocí kliknutí myší.

Obdobně by to šlo řešit pro načítání obrazů sekvencí. Toto bylo započato při ukládání obrazů sekvencí, kdy se nejdříve uloží lokus sekvence, následuje podtržítko (znak: ‚_‘), pak se uloží velikost obrazu, následuje opět podtržítko a nakonec se zapíše pořadí vrcholů. Načítání už nebylo dotaženo do konce a vypisovat celý název do pole je docela zdlouhavé. Celý tento postup ukládání je proto, aby se zachovala informace o velikosti obrazu, i když by šla vyčíst z velikosti obrazu, ale hlavně o pořadí vrcholů, které je důležité pro další porovnávání obrazů. Opět nedotaženo do konce a při načtení obrazu se neaktualizuje velikost ani pořadí vrcholů.

### 3.4.2. Prahování

U velkých obrazů, jako jsou obrazy sekvencí celých chromozomů, číselně obrazy větší než 512 x 512 pixelů, se zvyšuje počet bílých nebo velmi světlých bodů. Trefit se přesně prahem do požadovaného místa je dosti obtížné. Nabízí se zadávání prahů pomocí číselného zadání, nebo předvolit práh třeba na střední hodnotu šedi v obraze, nebo jednoduše přidat lupu pro zvětšení požadované oblasti histogramu.

### 3.4.3. Počet dopadů do tripletu

Počet dopadů je spočítán v položce markovovy koeficienty. Jsou zde následně převedeny na procentuální zastoupení v předem definované ploše. Počet dopadů lze jedním řádkem dopsat a třeba vypsat do *Command Window*. Tyto hodnoty by pak přímo nesly informace o počtu tripletů v sekvenci.

Druhá možnost jak zjistit dopady do daného místa je zjistit souřadnice, kde se triplet nachází, což řeší volba *Zvyrazneni tripletu* a následným vybráním průniku intervalů na ose *x* a *y* lze vybrat jen body, které spadají do obou intervalů současně. Je možné to vyřešit dvojím prahováním pro každou osu, kdy jsou vybrány body spadající do hledaného intervalu třeba na ose *x*, a následně prahováním na ose y dodělat výběr z bodů, které prošly prvním dvojím

prahováním. Jen pro připomenutí, vypočtené body metodou CGR se ukládají do proměnné *handles.cgr1* pro levý obraz, nebo *handles.cgr2* pro obraz pravý. Slouží sice primárně pro zrušení zvýraznění tripletů a opětovné vykreslení CGR obrazu, ale s malou úpravou programu lze z proměnných spočítat i počet tripletů.

### 3.4.4. Samotný výpočet

Při volbě dlouhé sekvence bude výpočet CGR řádově v desítkách minut, proto by se hodilo tlačítko pro přerušení vykreslování a celého výpočtu. U FCGR takový problém není – je podstatně rychlejší. Velikost sekvence omezuje hlavně hardware v podobě RAM paměti.

### 3.4.5. Další možnosti v úpravě dynamiky obrazu

Velké obrazy mají histogram obrazu nevyvážen a značně posunut k bílé a světlým odstínům šedi. Pro lepší vjem by to chtělo nadefinovat převod v obraze, který by vysoký počet bílých a světlých bodů snížil, a roztáhl je do větší části spektra. Pokus vykompenzovat tuto změnu byl učiněn v podobě logaritmické expanze obrazu, která však v oblasti hodnot (0; 1) nevedla k požadovanému efektu. Lepší by asi bylo nadefinovat vlastní křivku, která by zvýraznila nízké hodnoty (černé a tmavé; blízké 0) a zároveň potlačila světlé (bílé a světlé; blízké 1).

### 3.4.6. Úprava chodu BCM

Pro zpřesnění výpočtu dimenzí a následně i multifraktálního koeficientu by šla metoda BCM upravit tak, aby přesah nezanedbávala, ale aby byl přesah doplněn o střední hodnotu do velikosti masky, a tak i necelá část obrazu mohla přispět ke své dimenzi. Může to vést ke zpřesnění výsledků a hlavně nezávislosti na pořadí vrcholů.

### 3.4.7. Kontrola vrcholů

Při změně vrcholů je provedena kontrola zadání znaků, zda jde o znaky A, C, G a T. Jen je zapomenutá kontrola pro kontrolu, zda zadaný znak je zadán pouze jednou, a lze tudíž zadat vrcholy i v posloupnosti AGCA. Běh programu pak při výpočtu nedá správné výsledky.

# 4. Závěr

Celá práce ukazuje možnosti práce s genomickými sekvencemi, jejich klasifikaci a porovnávání navzájem. Má poukázat na všechny problémy s tím spojené a nutnost dohody o nastavení parametrů pro možnost klasifikace.

Popsaný program *GenomeFCGR* vytvořený v prostředí *Matlab* verze 7.10.0 (R2010a) má ukázat možnosti práce se dvěma genomickými sekvencemi, různě modifikovat jejich obrazy a vzájemně je srovnávat. Program umožňuje dva způsoby převedení sekvence na obraz a to metodou CGR (více viz kap. 2.4.1. Chaos game reprezentace sekvence a v programu pak kap. 3.2.3. Tlačítko akce) a metodou FCGR (více viz kap. 2.4.4. Frekvenční CGR a v programu pak kap. 3.2.3. Tlačítko akce). Hlavní funkce pro klasifikaci je však BCM zpracování (více viz kap. 2.4.3. Box counting method a v programu v kap. 3.2.3. Tlačítko akce), které umožní z FCGR obrazu získat hodnotu multifraktílního koeficientu. Toto číslo je pak možno použít k třídění a srovnávání obrazů sekvencí i sekvencí samotných. Nechybí zde ani barevné podkreslení pro lepší optický vjem rozdílů, které nejsou nebo jsou jen málo viditelné v šedotónovém zobrazení.

Doporučené hodnoty, nalezené užíváním programu, pro nastavení programu, nebo pro budoucí klasifikaci jsou: velikost FCGR obrazu 64 x 64 pixelů, ujednotit nastavení vrcholů – doporučení autora je posloupnost GACT, kde je subjektivní dojem z obrazu sekvence nejlepší. Pokud by se obraz před zpracováním měl prahovat, pak doporučuji práh ve střední hodnotě odstínu šedi v obraze. Bohužel tato volba byla opomenuta při tvorbě programu a je to zmíněno v kap. 3.4.2. Prahování, která spadá do kap. 3.4. Návrhy na vylepšení programu. V této kapitole jsou také rozebrány všechny nalezené nedostatky programu, které vyplynuly z jeho používání při výpočtech.

# Použité zdroje:

[1] *Molecular Station – DNA Structure*. Dostupné z WWW: http://www.molecularstation.com/images/DNA-structure.gif

[2] *Exploring the Deep Frontier – DNA History*. Dostupné z WWW: http://www.ceoe.udel.edu/extreme2004/genomics/dnahistory.html

[3] JELÍNEK, J., TICHÁČEK, V.: *BIOLOGIE pro střední školy gymnazijního typu*. Olomouc: Fin Publishing, 1996. 415 s., 1. vydání.

[4] Sparknotes: Structure of Nucleic Acid; Bases, Sugars and Phosphates. Dostupné z WWW: http://www.sparknotes.com/biology/molecular/structureofnucleicacids/section2.rhtml

[5] Sparknotes: *Structure of Nucleic Acid*; Nucleotides and Nucleic Acids. Dostupné z WWW: http://www.sparknotes.com/biology/molecular/structureofnucleicacids/section1.rhtml

[6] Volně přístupný polský e-learning: *GENETYKA*. Dostupné z WWW: http://e-learning5.webpark.pl/genetics.htm

[7] TURNER, M. J., BLACKLEDGE, J. M., ANDREWS, P. R.: *Fractal Geometry in Digital Imaging*. Leicester, Academic Press 1998, ISBN 0-12-703970-8

[8] NEDVĚD, J.: *Fraktál v sekvenci DNA*. Brno: Vysoké učení technické v Brně, Fakulta elektroniky a komunikačních technologií, 2010. 45s. 7příl. Vedoucí bakalářské práce Ing. Martin Valla.

[9] HINNER, M.: *Jemný úvod do fraktálů*. 1993. Dostupné z WWW: http://martin.hinner.info/math/Fraktaly/

[10] BIKANDI, J.; MIRA, A.: *OligoWeb. Chaos Game Representation: CGR/FCGR/ZCGR* http://insilico.ehu.es/oligoweb/info/CGR.php

[11] Almeida, J. S.; Carrico, J. A.; Maretzek, A.; Noble, P. A.; Fletcher, M.: *Analysis of genomic sequence by Chaos Game Representation*. Bioinformatics Vol. 17 no. 5 2001, Pages 429-437. Staženo z bioinformatics.oxfordjournals.org na VUT Brno 29. 9. 2010

[12] Sieber, V. K.: *Chomosomes and Karyotypes: Karyotype*. Dostupné z WWW: http://homepages.uel.ac.uk/V.K.Sieber/human.htm

[13] PITNER, V. *Reprezentace a zpracování genomických signálů*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2011. 43 s., 7 příl., Vedoucí bakalářské práce Ing. Martin Valla.

[14] VANČURA, J.: *Fraktály*. Kapitola – Fraktální geometrie. Dostupné z WWW: http://www.fractals.webz.cz/fraktalygeo.htm

[15] Box-Counting Dimension of a Gasket. Dostupné z WWW: http://classes.yale.edu/fractals/fracanddim/boxdim/GasketBoxDim/GasketBoxDim.html

[16] DURAND, P.: *Pygram and an application to non-coding region analysis. BMC Bioinformatics* [online]. 2006, 7:477, [cit. 2010-12-28]. Dostupný z WWW: http://www.biomedcentral.com/1471-2105/7/477

[17] RANDIĆ, M.: *Another look at the chaos-game representation of DNA. Chemical Physics Letters* [online]. 2008, 456, [cit. 2011-04-20]. Dostupný z WWW: http://www.sciencedirect.com/science/article/pii/S000926140800345X

[18] YAU, S. S. -T.: *DNA sequence representation without degeneracy*. *Nucleic Acids Research* [online]. 2003, 31, 12, [cit. 2011-05-15]. Dostupný z WWW: http://nar.oxfordjournals.org/content/31/12/3078.full.pdf+html?sid=146a4489-aec0-487f-9c5b-043d773a7373

# Obsah přiloženého CD:

- diplomová práce (DP) v pdf + obrázky
- obrázky programu se všemi výpočty popsanými v DP
- program *GenomeFCGR* + potřebné funkce a pomocné skripty
- obrázky sekvencí celých chromozomů
- používané sekvence ve tvaru *lokus.mat* + obrázky