COMENIUS UNIVERSITY IN BRATISLAVA
FACULTY OF MATHEMATICS, PHYSICS AND INFORMATICS

GRAPHLET STRUCTURE ANALYSIS OF THE REAL NETWORKS

Master's thesis

2021 Bc. Michal Puškel

COMENIUS UNIVERSITY IN BRATISLAVA
FACULTY OF MATHEMATICS, PHYSICS AND INFORMATICS

GRAPHLET STRUCTURE ANALYSIS OF THE REAL NETWORKS

Master's thesis

Study programme:	Applied Informatics
Field of study:	2511 Applied Informatics
Training center:	Department of applied informatics
Supervisor:	doc. RNDr. Mária Markošová, PhD.

Bratislava, 2021 Bc. Michal Puškel

53156631

Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky

ZADANIE ZÁVEREČNEJ PRÁCE

Meno a priezvisko študenta:	Bc. Michal Puškel
Študijný program:	aplikovaná informatika (Jednoodborové štúdium, magisterský II. st., denná forma)
Študijný odbor:	aplikovaná informatika
Typ záverečnej práce:	diplomová
Jazyk záverečnej práce:	anglický
Sekundárny jazyk:	slovenský

Názov: Graphlet structure analysis of the real networks
Analýza grafletovej štruktúry reálnych sietí

Anotácia: Študent vytvorí nový softvér alebo vylepší existujúci softvér na analýzu grafletovej štruktúry komplexných sietí. S pomocou tohto softvéru zanalyzuje grafletovú štruktúru sietí, reálnych dát a vypočíta všetky miery s touto štruktúrou súvisiace. Zanalyzuje tiež grafletovú štruktúru umelo vytvorených sietí a porovná s dátami.

Cieľ: Analyzovať grafletovú štruktúru reálnych sietí pomocou vylepšeného alebo vlastného softvéru.

Vedúci:	doc. RNDr. Mária Markošová, PhD.
Konzultant:	Mgr. Peter Náther, PhD.
Konzultant:	Mgr. Andrej Jursa
Katedra:	FMFI.KAI - Katedra aplikovanej informatiky
Vedúci katedry:	prof. Ing. Igor Farkaš, Dr.

Spôsob sprístupnenia elektronickej verzie práce:
prípustná pre vlastnú VŠ

Dátum zadania: 25.09.2017

Dátum schválenia: 09.10.2017

prof. RNDr. Roman Ďurikovič, PhD.
garant študijného programu

....

študent vedúci práce

I hereby affirm that this Master's thesis represents my own written work and that I have used no sources and aids other than those indicated. All passages quoted from publications or paraphrased from these sources are properly cited and attributed.

. . . .

Bratislava, 2021 Bc. Michal Puškel

Acknowledgement

I would like to thank all people around me supporting me while writing this thesis. I would like to thank especially my supervisor doc. RNDr. Mária Markošová, PhD., as well as all the other members of FMFI GRG (Graph research group): doc. RNDr. Tatiana Jajcayová, PhD., Mgr. Andrej Jursa, RNDr. Martin Nehéz, PhD., Mgr. Peter Náther, PhD., doc. RNDr. Boris Rudolf, PhD. I would like to thank all my family and friends as well.

Abstract

One of the goals of this master thesis is to finally find out if an attribute of a functional brain network *„to suffer from the Alzheimer disease"* shows off in the structure of its graph. Because the comparison of the networks structure using isomorphism is a hard problem, we have decided to use probabilistic comparison proposed by Nataša Pržulj in her paper Biological Network Comparison Using Graphlet Degree Distribution.

Another goal is to improve our existing network distributed system to compute graphlet degree distribution.

Ultimatelly we implemented our own version of combinatorial approach to graphlet counting (orca) inspired by paper of Tomaž Hočevar: A combinatorial approach to graphlet counting.

Keywords: graphs, networks, degree distribution, graphlet degree distribution, orca

Abstrakt

Jedným z cieľov tejto diplomovej práce je konečne zistiť, či sa atribút funkčnej siete mozgu *„mať Alzheimerovu chorobu"* prejavuje v štruktúre siete. Keďže porovnávať štruktúru sietí priamo pomocou izomorfizmu je ťažký problém, rozhodli sme sa využiť metodiku pravdepodobnostného porovnania navrhnutú Natašou Pržulji v práci Biological Network Comparison Using Graphlet Degree Distribution.

Našim ďalším cieľom je vylepšiť naše už existujúce softvérové distribuované paralelizované riešenie na výpočet grafletovej stupňovej distribúcie.

Napokon sme implementovali našu vlastnú verziu kobminatorického prístupu k počítaniu grafletov (orca) inšpirovanú publikáciou Tomaža Hočevara: A combinatorial approach to graphlet counting.

Kľúčové slová: grafy, siete, distribúcia stupňa vrcholov, distribúcia orbít grafletov, orca

Contents

1 Introduction **1**

1.1 Graph comparison using graphlet degree distribution 1

2 Goals of the thesis **16**

3 Graph theory **17**

4 Solution **18**

4.1 The brute force, but as good as even possible 18

4.2 Orca 18

4.2.1 Custom orca implementation 19

4.3 Workflow 19

5 Results **20**

6 Conclusion **21**

Chapter 1

Introduction

This thesis is a direct sequel to our former thesis [Pu7]. Let's briefly refresh the problem we solve in next section 1.1.

1.1 Graph comparison using graphlet degree distribution

In this thesis we strive for finishing our ultimate goal to find out, whether attribute of brain functional network "to suffer by Alzheimer's disease" is projected to its graph structure.

We obtained input dataset of examined brain functional networks from work of McCarthy and col. [MBF14]. Dataset consists of space standardized brain functional networks from 40 research participants. Participants were divided into 3 groups: young people, elder people with diagnosed tendency for Alzheimer's disease and elder people without this diagnosis. All these networks are represented by simple, connected, undirected graphs.

Should we be able to distinguish between two subsets of input dataset according to diagnosed disease by usage a computer program comparing graph structure, we may assume that Alzheimer's disease is really projected to brain functional networks structure.

This input dataset comes from original research of Buckner and col. [BSS$^+$00], later this data was examined by McCarthy and col. [MBF14]. The Dataset was acquired by functional magnetic resonance (fMRI).

These networks stand for networks of active parts of brain, activated by some (cognitive) activity. At increased activity of some specific part of brain, that specific part of brain consumes much more oxygen and this effect can be scanned via fMRI scanner. Consequently 3D model of brain is created from these pictures and finally graph model is created from these pictures.

Let's ask ourselves a question: How to measure a graph structure? What actually and exactly is the graph structure?

The graph structure is something we all sense just by the pure intuition. For instance, let's compare two paths, picture 1.1 shows us:

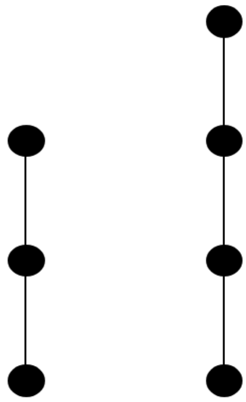

Figure 1.1: Two different graphs with similar structure (definition **??**).

We observe similar structure amongst graphs, in spite of theirs different cardinalities of node set and edge set.

Figure 1.2 portraits significant difference between path and star.

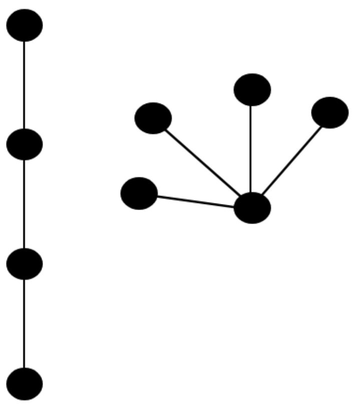

Figure 1.2: Two different graphs with different structure (definition **??**), (definition **??**).

Though, the pure intuition is great, we can't make a mathematics model for comparison out of that; it goes without saying it's impossible to code executable program for comparison, we need, this way. So, let's try to examine node degree (definition **??**), or more precisely average node degree (definition **??**) of mentioned graphs.

Nonetheless it is not hard at all to find a counterexample at this primitive approach. And so we did on image 1.3 where this graph statistic measure equals for both graphs despite different structure of path and star.

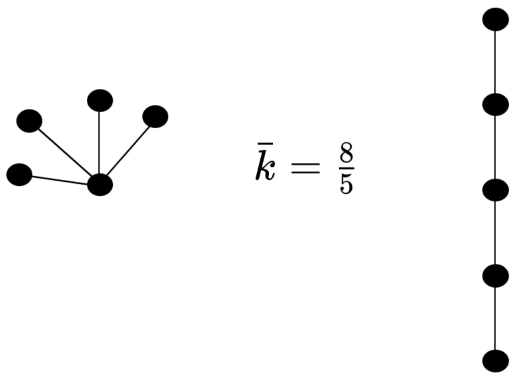

Figure 1.3: Average node degree comparison of two different graphs with different structure.

So, let's give the node degree distribution (definition **??**) a try, or more precisely node degree sequence (definition **??**).

But for graphs displayed in image 1.4 even this statistic measure will fail again to distinguish graph structure.

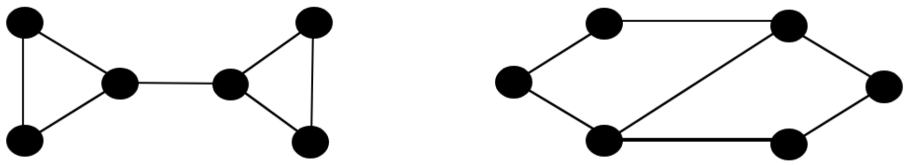

Figure 1.4: Node degree sequence comparison of two different graphs with different structure.

$$Node\ degree\ sequence = (3, 3, 2, 2, 2, 2)$$

However it is obvious, that graphs illustrated in image 1.4 do have absolutelly different graph structure; for example, one of them contains bridge (definition **??**), the other one contains Hamiltonian cycle (definition **??**).

While examining different graphs' structure the best measure to distinguish it is equivalence relation *graph isomorphism* (definition **??**).

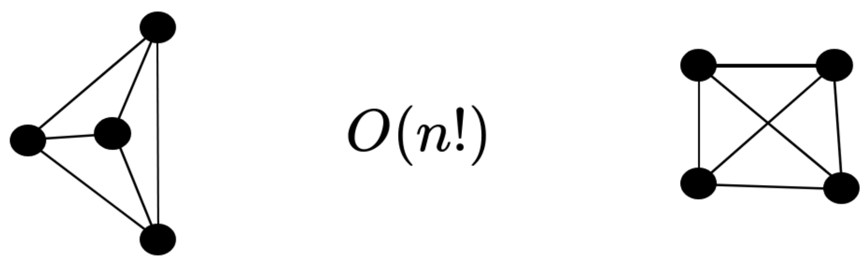

Figure 1.5: Two isomorphic graphs with absolutely same structure.

Even it might be not obvious at first glance, graphs displayed in picture 1.5 do have same structure.

However solution of mentioned problem suffers from unacceptable computational complexity. The largest graph examined in this thesis contains 8394 nodes and 450042 edges (definition **??**).

While investigating graph isomorphism very strict condition on the same count of nodes and edges is applied. We investigate all of $n!$ possible bijections, where n represents amount of nodes. As an illustration, let us consider 10 nodes, that would imply computational complexity of $10! = 3628800$ possible investigated bijections.

Nevertheless in paper Pržulj [Pr7] was introduced revolutionary heuristic for measuring structure similarity of two graphs while observing distribution of appearance of nodes in different orbits of specifically chosen graphlets illustrated in figure 1.6. This approach is the direct generalization of node degree distribution.

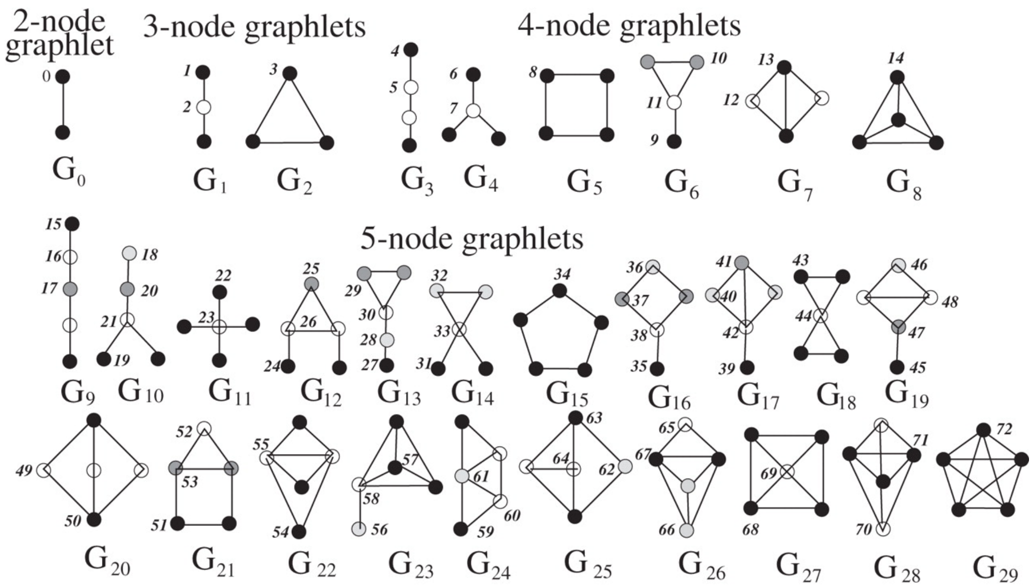

Figure 1.6: Automorphism orbits $0, 1, 2, \ldots, 72$ of 30 2, 3, 4 and 5-nodes graphlets $G_0, G_1, \ldots, G_{29}$; in graphlet $G_i,\ i \in \{0, 1, \ldots 29\}$, Colors are chosen arbitrary; nodes of the same color belong to the same orbit within that graphlet Pržulj [Pr7].

After enumeration of all 73 orbit distributions (called graphlet degree distribution) of two graphs we apply set of mathematical formulas displayed in picture 1.7 on achieved data to get real number in range $\langle 0, 1\rangle$ determining probability of structure match of two graphs, where 0 means no match and 1 means very high probability of match.

$$S_G^j(k) = \frac{d_G^j(k)}{k} \qquad T_G^j = \sum_{k=1}^{\infty} S_G^j(k) \qquad N_G^j(k) = \frac{S_G^j(k)}{T_G^j}$$

$$D^j(G,H) = \left(\sum_{k=1}^{\infty} [N_G^j(k) - N_H^j(k)]^2 \right)^{\frac{1}{2}} \qquad A^j(G,H) = 1 - D^j(G,H)$$

$$A_{arith}(G,H) = \frac{1}{73} \sum_{j=0}^{72} A^j(G,H) \qquad A_{geo}(G,H) = \left(\prod_{j=0}^{72} A^j(G,H) \right)^{\frac{1}{73}}$$

Figure 1.7: Mathematical formulas to determine graphlet degree distribution agreement between two graphs Pržulj [Pr7].

Graphlet degree distribution can be computed by raw brute force as we strove for in our previous thesis featuring *the brute force, but as good as even possible algorithms* [Pu7].

But it is worth to mention mind-blowing novel combinatorial approach to graphlet counting published in paper by Hočevar [HD14].

For each orbit, we choose some node y from the corresponding graphlet and observe the graphlets and orbits in which the node of interest x appears if we add edges between y and other nodes in graphlet.

Let x be the node of interest, let y be the node whose edges we observe and let x_1, x_2 and x_3 be the other three nodes in the graphlet.

Figure 1.8 illustrates counting of appearances of x in o_{59}, which belongs to G_{24}. We will focus on the node marked by y, which is connected to the nodes marked by x_1 and x_3. Removing y reduces G_{24} into a diamond, G_7, with x in orbit o_{12}.

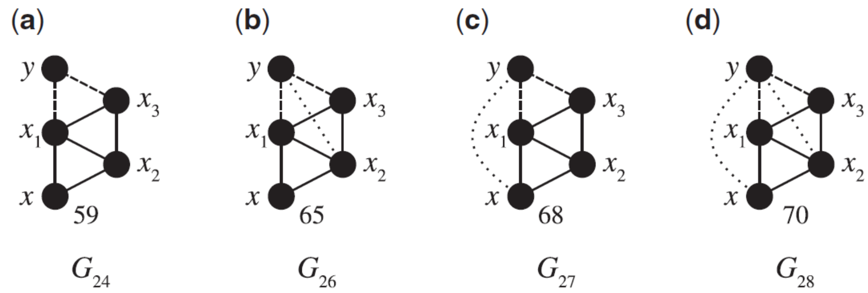

Figure 1.8: Computing orbit count o_{59}; figures show graphlets for different edges between y and other nodes and the orbits of x Hočevar [HD14].

Now assume that we are computing orbits for a certain node x and discover some induced subgraph $H \cong G_7$ with x in o_{12}. We assign labels x_1, x_2 and x_3 to the remaining nodes as shown in the figure 1.8. Altogether, the graph G contains $c(x_1, x_3)$ common neighbours of x_1 and x_3. Although all these nodes are (by definition of $c(x_1, x_3)$) connected to x_1 and x_3, some are also connected to x_2 or x, or both. Figure 1.8 shows all four possibilities, which give graphlets G_{24}, G_{26}, G_{27} and G_{28} with x in orbits o_{59}, o_{65}, o_{68} and o_{70}, respectively. Therefore, $o'_{59} + o'_{65} + o'_{68} + o'_{70} = c(x_1, x_3) - 1$, where o'_i denote orbits of x with respect to H.

For the relation between o_{59}, o_{65}, o_{68} and o_{70} for the entire graph, we sum this over all possible induced G_7 with x in o_{12}. After considering the symmetries that cause counting the same graphlet multiple times with different assignments of y, x_1, x_2 and x_3, we get formula shown in picture 1.9.

$$o_{59} + 4o_{65} + 2o_{68} + 6o_{70} = \sum_{\substack{x_1, x_2, x_3: \\ x_1 < x_2 \wedge x_3 \notin N(x), \\ G[\{x, x_1, x_2, x_3\}] \cong G_7}} c(x_1, x_3) + c(x_2, x_3) - 2$$

Figure 1.9: Formula denoting relation between o_{59}, o_{65}, o_{68} and o_{70} for the entire graph Hočevar [HD14].

Condition $x_1 < x_2$ (under some arbitrary ordering of nodes) is needed to consider each graphlet G_7 just once. The other two conditions put x in o_{12}. The second term in the sum, $c(x_2, x_3)$, accounts for the case in which the roles of x_1 and x_2 are exchanged.

Using a similar construction for other orbits, except for o_{72} and first 15 orbits i.e. $o_0, o_1, \ldots, o_{14}$, gives 57 linear equations for 58 orbits displayed in pictures 1.10, 1.11, 1.12, 1.13, 1.14, 1.15, 1.16, 1.17. Hence these formulas allows us to eliminate threat of n^5 computational complexity while enumerating 5 node graphlets, for the cost of memory and our algorithms will mostly feature only n^4 computational complexity.

$$
\begin{aligned}
P_{14}(x,u,v,t) &= u < v < t \wedge G[\{x,u,v,t\}] \cong G_8 \\
P_{13}(x,u,v,t) &= v < t \wedge (v,t) \notin E \wedge G[\{x,u,v,t\}] \cong G_7 \\
P_{12}(x,u,v,t) &= u < v \wedge (x,t) \notin E \wedge G[\{x,u,v,t\}] \cong G_7 \\
P_{11}(x,u,v,t) &= u < v \wedge u,v \notin N(t) \wedge G[\{x,u,v,t\}] \cong G_6 \\
P_{10}(x,u,v,t) &= x,u \notin N(t) \wedge G[\{x,u,v,t\}] \cong G_6 \\
P_{9}(x,u,v,t) &= v < t \wedge v,t \notin N(x) \wedge G[\{x,u,v,t\}] \cong G_6 \\
P_{8}(x,u,v,t) &= u < v \wedge u,v \in N(x) \wedge G[\{x,u,v,t\}] \cong G_5 \\
P_{7}(x,u,v,t) &= u < v < t \wedge u,v,t \in N(x) \wedge G[\{x,u,v,t\}] \cong G_4 \\
P_{6}(x,u,v,t) &= v < t \wedge x,v,t \in N(u) \wedge G[\{x,u,v,t\}] \cong G_4 \\
P_{5}(x,u,v,t) &= u,v \in N(x) \wedge t \in N(v) \wedge G[\{x,u,v,t\}] \cong G_3 \\
P_{4}(x,u,v,t) &= x,v \in N(u) \wedge t \in N(v) \wedge G[\{x,u,v,t\}] \cong G_3
\end{aligned}
$$

Figure 1.10: Conditions, P_i, define the order of nodes and put x in orbit o_i; e.g. in P_{13} node x is in orbit o_{13}. Some right-hand sides refer to the number of common neighbours of three nodes $|N(u) \cap N(v) \cap N(t)|$; with some abuse of notation, we write this as $c(u,v,t)$. For consistency, we also use $c(u)$ to denote the degree of a point, $|N(u)|$. Hočevar [HD14].

$$
\begin{aligned}
2o_{71} + 12o_{72} &= \sum_{u,v,t:\, P_{14}(x,u,v,t)} (c(x,u,v)-1) + (c(x,u,t)-1) + (c(x,v,t)-1) \\
o_{70} + 4o_{72} &= \sum_{u,v,t:\, P_{14}(x,u,v,t)} c(u,v,t) - 1 \\
4o_{69} + 2o_{71} &= \sum_{u,v,t:\, P_{13}(x,u,v,t)} c(x,v,t) - 1 \\
o_{68} + 2o_{71} &= \sum_{u,v,t:\, P_{13}(x,u,v,t)} c(u,v,t) - 1 \\
o_{67} + 12o_{72} + 4o_{71} &= \sum_{u,v,t:\, P_{14}(x,u,v,t)} (c(x,u)-2) + (c(x,v)-2) + (c(x,t)-2) \\
o_{66} + 12o_{72} + 2o_{71} + 3o_{70} &= \sum_{u,v,t:\, P_{14}(x,u,v,t)} (c(u,v)-2) + (c(u,t)-2) + (c(v,t)-2) \\
2o_{65} + 3o_{70} &= \sum_{u,v,t:\, P_{12}(x,u,v,t)} c(u,v,t) \\
o_{64} + 2o_{71} + 4o_{69} + o_{68} &= \sum_{u,v,t:\, P_{13}(x,u,v,t)} c(v,t) - 2 \\
o_{63} + 3o_{70} + 2o_{68} &= \sum_{u,v,t:\, P_{12}(x,u,v,t)} c(x,t) - 2 \\
2o_{62} + o_{68} &= \sum_{u,v,t:\, P_{8}(x,u,v,t)} c(u,v,t) \\
2o_{61} + 4o_{71} + 8o_{69} + 2o_{67} &= \sum_{u,v,t:\, P_{13}(x,u,v,t)} (c(x,v)-1) + (c(x,t)-1)
\end{aligned}
$$

Figure 1.11: Equations for solving orbits o_{71} - o_{61} Hočevar [HD14].

$$
\begin{aligned}
o_{60} + 4o_{71} + 2o_{68} + 2o_{67} &= \sum_{u,v,t:\, P_{13}(x,u,v,t)} (c(u,v) - 1) + (c(u,t) - 1) \\
o_{59} + 6o_{70} + 2o_{68} + 4o_{65} &= \sum_{u,v,t:\, P_{12}(x,u,v,t)} (c(u,t) - 1) + (c(v,t) - 1) \\
o_{58} + 4o_{72} + 2o_{71} + o_{67} &= \sum_{u,v,t:\, P_{14}(x,u,v,t)} c(x) - 3 \\
o_{57} + 12o_{72} + 4o_{71} + 3o_{70} + o_{67} + 2o_{66} &= \sum_{u,v,t:\, P_{14}(x,u,v,t)} (c(u) - 3) + (c(v) - 3) + (c(t) - 3)
\end{aligned}
$$

Figure 1.12: Equations for solving orbits o_{60} - o_{57} Hočevar [HD14].

$$
\begin{aligned}
3o_{56} + 2o_{65} &= \sum_{u,v,t:\, P_{9}(x,u,v,t)} c(u,v,t) \\
3o_{55} + 2o_{71} + 2o_{67} &= \sum_{u,v,t:\, P_{13}(x,u,v,t)} c(x,u) - 2 \\
2o_{54} + 3o_{70} + o_{66} + 2o_{65} &= \sum_{u,v,t:\, P_{12}(x,u,v,t)} c(u,v) - 2 \\
o_{53} + 2o_{68} + 2o_{64} + 2o_{63} &= \sum_{u,v,t:\, P_{8}(x,u,v,t)} c(x,u) + c(x,v) \\
2o_{52} + 2o_{66} + 2o_{64} + o_{59} &= \sum_{u,v,t:\, P_{10}(x,u,v,t)} c(u,t) - 1 \\
o_{51} + 2o_{68} + 2o_{63} + 4o_{62} &= \sum_{u,v,t:\, P_{8}(x,u,v,t)} c(u,t) + c(t,v) \\
3o_{50} + o_{68} + 2o_{63} &= \sum_{u,v,t:\, P_{8}(x,u,v,t)} c(x,t) - 2 \\
2o_{49} + o_{68} + o_{64} + 2o_{62} &= \sum_{u,v,t:\, P_{8}(x,u,v,t)} c(u,v) - 2 \\
o_{48} + 4o_{71} + 8o_{69} + 2o_{68} + 2o_{67} + 2o_{64} + 2o_{61} + o_{60} &= \sum_{u,v,t:\, P_{13}(x,u,v,t)} (c(v) - 2) + (c(t) - 2)
\end{aligned}
$$

Figure 1.13: Equations for solving orbits o_{56} - o_{48} Hočevar [HD14].

$$o_{47} + 3o_{70} + 2o_{68} + o_{66} + o_{63} + o_{60} = \sum_{u,v,t:\, P_{12}(x,u,v,t)} c(x) - 2$$
$$o_{46} + 3o_{70} + 2o_{68} + 2o_{65} + o_{63} + o_{59} = \sum_{u,v,t:\, P_{12}(x,u,v,t)} c(t) - 2$$
$$o_{45} + 2o_{65} + 2o_{62} + 3o_{56} = \sum_{u,v,t:\, P_{9}(x,u,v,t)} c(v,t) - 1$$
$$4o_{44} + o_{67} + 2o_{61} = \sum_{u,v,t:\, P_{11}(x,u,v,t)} c(x,t)$$
$$2o_{43} + 2o_{66} + o_{60} + o_{59} = \sum_{u,v,t:\, P_{10}(x,u,v,t)} c(v,t)$$
$$o_{42} + 2o_{71} + 4o_{69} + 2o_{67} + 2o_{61} + 3o_{55} = \sum_{u,v,t:\, P_{13}(x,u,v,t)} c(x) - 3$$
$$o_{41} + 2o_{71} + o_{68} + 2o_{67} + o_{60} + 3o_{55} = \sum_{u,v,t:\, P_{13}(x,u,v,t)} c(u) - 3$$
$$o_{40} + 6o_{70} + 2o_{68} + 2o_{66} + 4o_{65} + o_{60} + o_{59} + 4o_{54} = \sum_{u,v,t:\, P_{12}(x,u,v,t)} (c(u) - 3) + (c(v) - 3)$$
$$2o_{39} + 4o_{65} + o_{59} + 6o_{56} = \sum_{u,v,t:\, P_{9}(x,u,v,t)} (c(u,v) - 1) + (c(u,t) - 1)$$

Figure 1.14: Equations for solving orbits o_{47} - o_{39} Hočevar [HD14].

$$o_{38} + o_{68} + o_{64} + 2o_{63} + o_{53} + 3o_{50} = \sum_{u,v,t:\, P_{8}(x,u,v,t)} c(x) - 2$$
$$o_{37} + 2o_{68} + 2o_{64} + 2o_{63} + 4o_{62} + o_{53} + o_{51} + 4o_{49} = \sum_{u,v,t:\, P_{8}(x,u,v,t)} (c(u) - 2) + (c(v) - 2)$$
$$o_{36} + o_{68} + 2o_{63} + 2o_{62} + o_{51} + 3o_{50} = \sum_{u,v,t:\, P_{8}(x,u,v,t)} c(t) - 2$$
$$2o_{35} + o_{59} + 2o_{52} + 2o_{45} = \sum_{u,v,t:\, P_{4}(x,u,v,t)} c(u,t) - 1$$
$$2o_{34} + o_{59} + 2o_{52} + o_{51} = \sum_{u,v,t:\, P_{4}(x,u,v,t)} c(x,t)$$
$$2o_{33} + o_{67} + 2o_{61} + 3o_{58} + 4o_{44} + 2o_{42} = \sum_{u,v,t:\, P_{11}(x,u,v,t)} c(x) - 3$$

Figure 1.15: Equations for solving orbits o_{38} - o_{33} Hočevar [HD14].

$$
\begin{aligned}
2o_{32} + 2o_{66} + o_{60} + o_{59} + 2o_{57} + 2o_{43} + 2o_{41} + o_{40} &= \sum_{u,v,t:\ P_{10}(x,u,v,t)} c(v) - 3 \\
o_{31} + 2o_{65} + o_{59} + 3o_{56} + o_{43} + 2o_{39} &= \sum_{u,v,t:\ P_{9}(x,u,v,t)} c(u) - 3 \\
o_{30} + o_{67} + o_{63} + 2o_{61} + o_{53} + 4o_{44} &= \sum_{u,v,t:\ P_{11}(x,u,v,t)} c(t) - 1 \\
o_{29} + 2o_{66} + 2o_{64} + o_{60} + o_{59} + o_{53} + 2o_{52} + 2o_{43} &= \sum_{u,v,t:\ P_{10}(x,u,v,t)} c(t) - 1 \\
o_{28} + 2o_{65} + 2o_{62} + o_{59} + o_{51} + o_{43} &= \sum_{u,v,t:\ P_{9}(x,u,v,t)} c(x) - 1 \\
2o_{27} + o_{59} + o_{51} + 2o_{45} &= \sum_{u,v,t:\ P_{4}(x,u,v,t)} c(v,t) \\
o_{26} + 2o_{67} + 2o_{63} + 2o_{61} + 6o_{58} + o_{53} + 2o_{47} + 2o_{42} &= \sum_{u,v,t:\ P_{11}(x,u,v,t)} (c(u) - 2) + (c(v) - 2) \\
2o_{25} + 2o_{66} + 2o_{64} + o_{59} + 2o_{57} + 2o_{52} + o_{48} + o_{40} &= \sum_{u,v,t:\ P_{10}(x,u,v,t)} (c(u) - 2)
\end{aligned}
$$

Figure 1.16: Equations for solving orbits o_{32} - o_{25} Hočevar [HD14].

$$
\begin{aligned}
o_{24} + 4o_{65} + 4o_{62} + o_{59} + 6o_{56} + o_{51} + 2o_{45} + 2o_{39} &= \sum_{u,v,t:\ P_{9}(x,u,v,t)} (c(v) - 2) + (c(t) - 2) \\
4o_{23} + o_{55} + o_{42} + 2o_{33} &= \sum_{u,v,t:\ P_{7}(x,u,v,t)} c(x) - 3 \\
3o_{22} + 2o_{54} + o_{40} + o_{39} + o_{32} + 2o_{31} &= \sum_{u,v,t:\ P_{6}(x,u,v,t)} c(u) - 3 \\
o_{21} + 3o_{55} + 3o_{50} + 2o_{42} + 2o_{38} + 2o_{33} &= \sum_{u,v,t:\ P_{7}(x,u,v,t)} (c(u) - 1) + (c(v) - 1) + (c(t) - 1) \\
o_{20} + 2o_{54} + 2o_{49} + o_{40} + o_{37} + o_{32} &= \sum_{u,v,t:\ P_{6}(x,u,v,t)} c(x) - 1 \\
o_{19} + 4o_{54} + 4o_{49} + o_{40} + 2o_{39} + o_{37} + 2o_{35} + 2o_{31} &= \sum_{u,v,t:\ P_{6}(x,u,v,t)} (c(v) - 1) + (c(t) - 1) \\
2o_{18} + o_{59} + o_{51} + 2o_{46} + 2o_{45} + 2o_{36} + 2o_{27} + o_{24} &= \sum_{u,v,t:\ P_{4}(x,u,v,t)} c(v) - 2 \\
2o_{17} + o_{60} + o_{53} + o_{51} + o_{48} + o_{37} + 2o_{34} + 2o_{30} &= \sum_{u,v,t:\ P_{5}(x,u,v,t)} c(u) - 1 \\
o_{16} + o_{59} + 2o_{52} + o_{51} + 2o_{46} + 2o_{36} + 2o_{34} + o_{29} &= \sum_{u,v,t:\ P_{4}(x,u,v,t)} c(x) - 1 \\
o_{15} + o_{59} + 2o_{52} + o_{51} + 2o_{45} + 2o_{35} + 2o_{34} + 2o_{27} &= \sum_{u,v,t:\ P_{4}(x,u,v,t)} c(t) - 1
\end{aligned}
$$

Figure 1.17: Equations for solving orbits o_{24} - o_{15} Hočevar [HD14].

As we can see our thesis' topic features untrivial graph theory and thus we deem it worthy to explain key terms in chapter 3.

Chapter 2

Goals of the thesis

One of the goals of this thesis is to finish research if property of functional brain network to suffer from Alzheimer's disease is projected to graph structure.

We need to hotfix existing software application to correctly enumerate graphlet degree distribution and carry out the experiment again.

Next we need to extend functionality of our application to actually compare graph structure by graphlet degree distribution agreement.

But the main goal of this thesis is to create our own implementation of combinatorial approach to graphlet counting (orca). Try our best to create faster solution that already existing, provided by author Hočevar [HD14].

We shall as well verify graphlet degree distribution agreement measure as suitable for graph structure comparison by comparing some trivially distinguishable networks (for example, Erdős–Rényi, Barabási–Albert, random geometric models).

Chapter 3

Graph theory

TODO teoria grafov - iba to co potrebujem bude vychadzat z predoslej prace

- lepsie vysvetlit orbity / automorfizmy

+ vysvetlit aj BA ER GEO - resp tie umele ake zmeram / co su bezskalove...

nieco z komplexnyxh sieti - iba to co sa tyka mojej prace

Chapter 4

Solution

bla bla kombinatorcke riesenie na kt som si najskor netrufol (orca) vs. chcel som svoje nieco spravit (brute force GDD konzolova aplikacia)

4.1 The brute force, but as good as even possible

TODO "STRUCNE!!!" iba popisat co som urobil zmeral v predoslej praci

- bol to distrib system
- vyhody nevyhody

4.2 Orca

TODO vysvetlit co je Orca
-aj s obrazkami podrobnejsie 4node graflety

4.2.1 Custom orca implementation

TODO trochu popisat co som sparalalelizoval v orce - ci sa zrychlila..

4.3 Workflow

TODO popisat / zhrnut co sa spravilo experimenty

- distrib system meranie (este bak)
- preteceny integer (este bak)
- dalsie meranie cez letne prazdniny
- modul na GDD agreement
- implementacia s arbitrarnou presnostou (zavisi iba od RAM) - vlastna odmocnina
- podozrive vysledky
- odhalena chyba v orbite [55] - porovnane vsetky vysledky meranych grafov moje vs orca
- modul na spracovanie vysledkov z orcy mojim gdd agreementom - konecny experiment
- overenie GDD agreementu s nejaky umelymi

Chapter 5

Results

We patched our own distributed software solution featuring parallelism for graphlet degree distribution enumeration of arbitrary simple, connected, undirected graph.

Next we carried out series of multiple experiments in computer science labs I-H3 and I-H6 FMFI UK with functional networks of participants divided into 3 groups: young people, elder people with diagnosed tendency for Alzheimer's disease and elder people without this diagnosis.

Consequently, we improved our own software solution to be able to compare graphlet degree distribution agreement of measured graphs with it.

We compared gained graphlet degree distribution agreements.

We verified graphlet degree distribution agreement measure as suitable for graph structure comparison by distinguishing some trivially different networks.

We successfully implemented our own version of orca (combinatorial approach to graphlet counting). Thanks to parallel *go routines* featured by *GO* programming language we achieved significant speed up compared to original not parallel *c++* version.

Chapter 6

Conclusion

We assumne Alzheimer's disease is not projected to functional networks graph structure, as we were not able to distinguish graphs from different participants' groups by graphlet degree distribution agreement.

We don't suggest examining 6 node graphlets for graphlet degree distribution comparison as future work, as we deem it the pure waste of time and energy.

Even our previous solution *the brute force, but as good as even possible algorithms* [Pu7] achieved surprisingly good performance results for specific graphs, we recommend using combinatorial approach to graphlet counting (orca) for graphlet degree distribution enumeration.

Bibliography

[BSS+00] Randy L. Buckner, Abraham Z. Snyder, Amy L. Sanders, Marcus E. Raichle, and John C. Morris. Functional brain imaging of young, nondemented, and demented older adults. *Journal of Cognitive Neuroscience*, 12(supplement 2):24–34, 2000.

[HD14] Tomaž Hočevar and Janez Demšar. A combinatorial approach to graphlet counting. *Bioinformatics*, 30(4):559–565, 12 2014.

[MBF14] Paul McCarthy, Lubica Benusková, and Elizabeth A. Franz. The age-related posterior-anterior shift as revealed by voxelwise analysis of functional brain networks. In *Front. Aging Neurosci.*, 2014.

[Pr7] Nataša Pržulj. Biological network comparison using graphlet degree distribution. *Bioinformatics*, 23(2):e177–e183, 01 2007.

[Pu7] Michal Puškel. *Pravdepodobnostné porovnávanie funkčných sietí mozgu (Bakalárska práca)*. Univerzita Komenského v Bratislave, 2017.

List of Figures

1.1 Two different graphs with similar structure (definition ??). . . 3
1.2 Two different graphs with different structure (definition ??), (definition ??). 3
1.3 Average node degree comparison of two different graphs with different structure. 4
1.4 Node degree sequence comparison of two different graphs with different structure. 5
1.5 Two isomorphic graphs with absolutely same structure. 6
1.6 Automorphism orbits $0, 1, 2, \ldots, 72$ of 30 2, 3, 4 and 5-nodes graphlets $G_0, G_1, \ldots, G_{29}$; in graphlet $G_i,\ i \in \{0, 1, \ldots 29\}$, Colors are chosen arbitrary; nodes of the same color belong to the same orbit within that graphlet Pržulj [Pr7]. 7
1.7 Mathematical formulas to determine graphlet degree distribution agreement between two graphs Pržulj [Pr7]. 8
1.8 Computing orbit count o_{59}; figures show graphlets for different edges between y and other nodes and the orbits of x Hočevar [HD14]. 9
1.9 Formula denoting relation between o_{59}, o_{65}, o_{68} and o_{70} for the entire graph Hočevar [HD14]. 10

1.10 Conditions, P_i, define the order of nodes and put x in orbit o_i; e.g. in P_{13} node x is in orbit o_{13}. Some right-hand sides refer to the number of common neighbours of three nodes $|N(u) \cap N(v) \cap N(t)|$; with some abuse of notation, we write this as $c(u, v, t)$. For consistency, we also use $c(u)$ to denote the degree of a point, $|N(u)|$. Hočevar [HD14]. 11
1.11 Equations for solving orbits o_{71} - o_{61} Hočevar [HD14]. 11
1.12 Equations for solving orbits o_{60} - o_{57} Hočevar [HD14]. 12
1.13 Equations for solving orbits o_{56} - o_{48} Hočevar [HD14]. 12
1.14 Equations for solving orbits o_{47} - o_{39} Hočevar [HD14]. 13
1.15 Equations for solving orbits o_{38} - o_{33} Hočevar [HD14]. 13
1.16 Equations for solving orbits o_{32} - o_{25} Hočevar [HD14]. 14
1.17 Equations for solving orbits o_{24} - o_{15} Hočevar [HD14]. 14

Appendix

Attachment consits of USB flash drive containing all source codes as well as supplementary materials.