Федеральное государственное автономное
образовательное учреждение высшего образования
«Национальный исследовательский университет
«Высшая школа экономики»

Факультет компьютерных наук

**Aleksejev Pavel**

# Implementing Islands and Main Path Approach in Python

**Project proposal**

Москва 2023

# Contents

**Abstract**

In this project, we study two different methods of graph analysis. The first method, called island approach is used to identify groups of vertices that, in some sense, form the strongest connections. The second method, called Main path approach, can be used to identify on the basis of arc weights an important small subnetwork. We are going to implement these methods on Python, apply them to real data and compare with traditional methods.

# 1 Introduction

Firstly, we need to describe what we mean by a network, A network $N = (V, L, P, W)$ consists of a graph $G = (V, L)$, where $V$ is the set of vertices and L is the set of links. Undirected links $E$ are called edges, and directed links A are called arcs. $n = card(V), m = card(L)$
$P$- vertex value functions of properties: $p : V \to A$
$W$- line value functions of properties: $w : L \to B$.
The vertex-cut of a network

$$N = (V, L, p), p : V \to \mathbf{R},$$

at selected level $t$ is a subnetwork

$$N(t) = (V', L(V'), p)$$

determined by $V' = \{x \in V : p(v) \geq t\}$ and $L(V')$ is the set of lines from L that have both endpoints $V'$. The line-cut of a network

$$N = (V, L, w), w : L \to \mathbf{R},$$

at selected level $t$ is a subnetwork

$$N(t) = (V(L'), L', w),$$

determined by $L' = \{e \in L : w(e) \geq t\}$ and $V(L')$ is the set of all endpoints of the lines from $L'$. The main idea is to make different cuts for various t, look at the components of the $N(t)$ and identify large subnetworks.

Second part of the work is dedicated to the method, which came from the citation network analysis. Citation network analysis started with the paper of Garfield et al. (1964) [1] in which the introduction of the notion of citation network is attributed to Gordon Allen. In this paper, on the example of Asimov's history of DNA [1], it was shown that the analysis "demonstrated a high degree of coincidence between an historian's account of events and the citational relationship between these events". An early overview of possible applications of graph theory in citation network analysis was made in 1965 by Garner [13]. The next important step was made by Hummon and Doreian (1989). They proposed three indices (NPPC, SPLC, SPNP) – weights of arcs that provide us with automatic way to identify the (most) important part of the citation network – the main path analysis.

V. Batagelj at his work [2] introduce an effective method for calculating the weights NPPC, SPLC, SPNP. We are going to implement this method in python and test them on real data.

# 2 Main part

let us introduce some definitions from [1]

**Definition.** *Nonempty subset of vertices $C \subseteq V$ is vertex island, if the corresponding induced subgraph $G|C = (C, L(C))$ is connected, and the values of the vertices in the neighborhood of $C$ are less than or equal to values of vertices from $C$.*

$$\max_{u \in N(C)} p(u) \leq \min_{v \in C p(v)}$$

Vertex island $C \subseteq V$ is regular vertex island, if stronger condition holds:

$$\max_{u \in N(C)} p(u) < \min_{v \in C p(v)}$$

**Definition.** *The set of vertices $C \subseteq V$ is local vertex summit, if it is regular vertex island and all of its vertices have the same value.*

Vertex island with only one local vertex summit is called simple vertex island.

**Definition.** *The set of vertices $C \subseteq V$ is edge island, if it is a singleton (degenerated island) or the corresponding induced subgraph is connected and there exists a spanning tree $T$ , such that the values of edges with exactly one endpoint in $C$ are less than or equal to the values of edges of the tree $T$*

$$\max_{(u;v)\in L: u\in C v\notin C} w((u;v)) \leq \min_{e\in L(T)} w(e)$$

Edge island $C \subseteq V$ is regular edge island, if stronger condition holds:

$$\max_{(u;v)\in L: u\in C v\notin C} w((u;v)) < \min_{e\in L(T)} w(e)$$

M. Zaversnik and V. Batagelj at their work [1] present algorithms which can be used to describe island structure os the graph.
1. First algorithm determines maximal regular vertex islands of limited size
2. Second algorithm determines the type of vertex island. Type refers to one of
– FLAT – all vertices have the same value
– SINGLE – island has only one local vertex summit
– MULTI – island has more than one local vertex summits
3. Third determines maximal regular edge islands of limited size

In our work we want to implement these alghorithms on Python and test them on data from [1] (The Edinburgh Associative Thesaurus). And also we want to test it on Air Traffic Passenger Data to determine the most connected cities.

An approach to the analysis of citation network is to determine for each unit / arc its importance or weight. These values are used afterward to determine the essential substructures in the network. Variety of methods for the case of positive weights were proposed by Hummon and Doreian [3], [4]:
1. Node pair projection count (NPPC) method: $w_d(u, v) = |R^{inv^*}(u)| \cdot |R^*(v)|$, where $R^{inv^*}$ is inverse relation to $R$
2. Search path link count (SPLC) method: $w_l(u, v)$ equals the number of "all possible search paths through the network emanating from an origin node" through the arc $(u, v) \in R$
3. Search path node pair (SPNP) method: $w_p(u, v)$ "accounts for all connected vertex pairs along the paths through the arc $(u, v) \in R$

V. Batagelj, at his work [2] provides efficient ways to compute these type of weights. We want to implement the calculation of these weights and compute them on the data from the paper [2] (US patents)

## References

[1] M. Zaversnik, V. Batagelj, Iselands. сб., 2004

[2] V. Batagelj, Efficient Algorithms for Citation Network Analysis | Preprint Series (2003)

[3] Hummon N.P., Doreian P.: Connectivity in a Citation Network: The Development of DNA Theory. Social Networks, 11(1989) 39-63.

[4] Hummon N.P., Doreian P.: Computational Methods for Social Network Analysis. Social Networks, 12(1990) 273-288.