Assignment 1

Smitesh Patil

2023-02-01

1. Basic measures

```
library(igraph)
```

```
## Warning: package 'igraph' was built under R version 4.1.3
```

```
library(kableExtra)
```

```
## Warning: package 'kableExtra' was built under R version 4.1.3
```

```
library(gt)
```

```
## Warning: package 'gt' was built under R version 4.1.3
```

```
dib_graph<-read.graph("dib2.graphml",format="graphml")
```

1.1 Give the number of nodes and edges

```
cat("num vertices:", vcount(dib_graph), "\n")
```

```
## num vertices: 8969
```

```
cat("num edges :", ecount(dib_graph), "\n")
```

```
## num edges : 46750
```

1.2 Is the network strongly or weakly connected. If neither, what is the distribution of component sizes.

1. Strongly connected components

```
strong_component = as.data.frame(table(factor(components(dib_graph, mode="strong")$csize)))
names(strong_component)[1] = "Component Size"
kbl(strong_component)
```

Component Size	Freq
1	3024
2	180
3	25
4	5
5	1
6	1
5479	1

Answer: The table above shows strongly connected components in the directed network. There is one component with 5479, 6 and 5 nodes and a distribution of component with sizes varying from 1 to 4.

2. Weakly connected components

```
weak_component = as.data.frame(table(factor(components(dib_graph, mode="weak")$csize)))
names(weak_component)[1] = "Component Size"
kable(weak_component)
```

Component Size	Freq
2	30
3	11
4	1
8872	1

Answer: The table above shows weakly connected components in the directed network. There is one component with 8872 and 4 nodes with 30 weakly connected components of size 2 and 11 components with size 3.

1.3 What is the diameter of the network ?

```
cat("The diameter of the network is : ", diameter(dib_graph, directed = T,
    unconnected = TRUE, weights = NA), "\n")
```

```
## The diameter of the network is :  18
```

1.4 What is the average path length of the network ?

```
cat("The average path length of the network :", mean_distance(dib_graph,
    directed = T), "\n")
```

```
## The average path length of the network : 6.017593
```

1.5 What is the clustering coefficient of the network ?

```
cat("The clustering coeff of the graph is :", transitivity(dib_graph, type = "localaverage"),
    "\n")
```

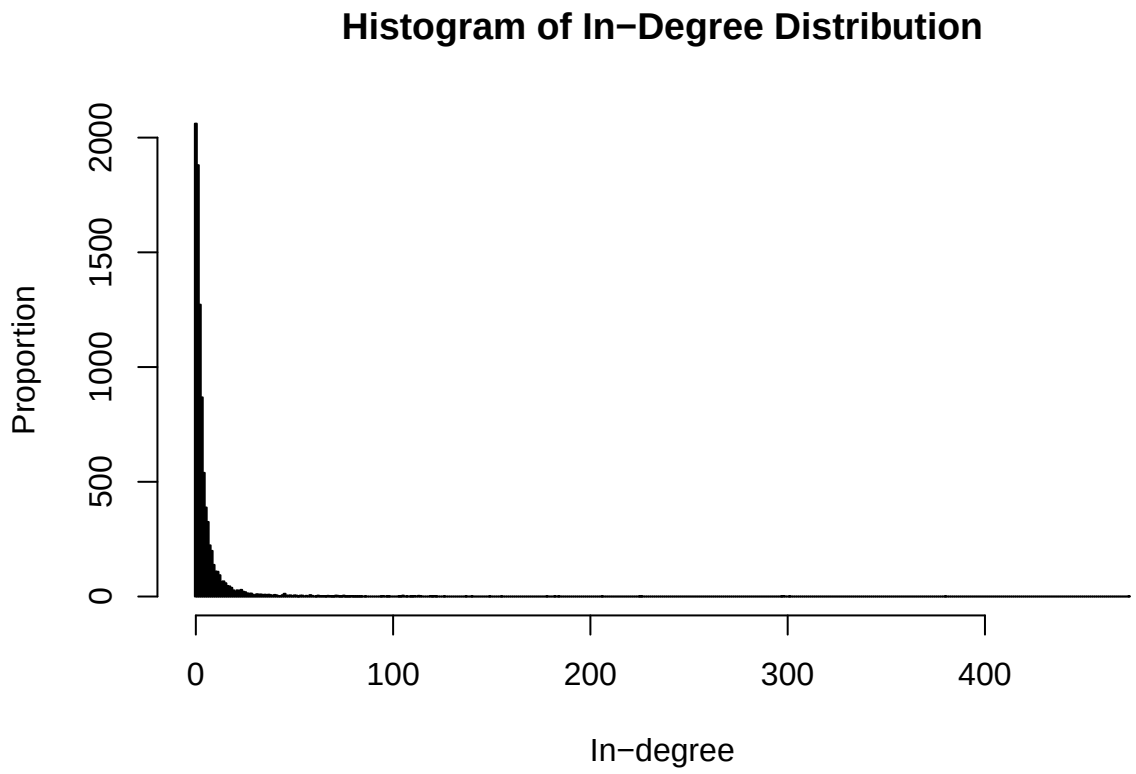
```
## The clustering coeff of the graph is : 0.2300017
```

1.6 What is the in- and out-degree distribution ?

```r
deg <- degree(dib_graph, mode = "in")
cat("The in-degree distribution of the graph varies from ", min(deg), "to ",
    max(deg))
```

```
## The in-degree distribution of the graph varies from  0 to  473
```

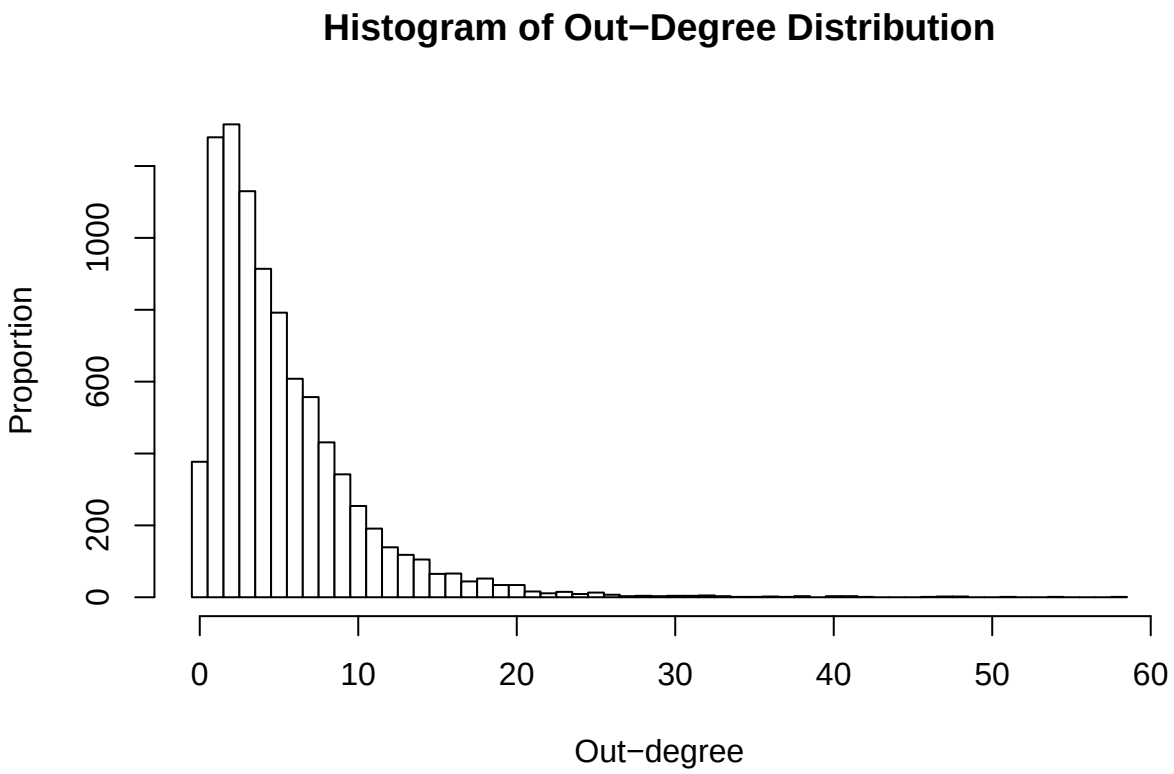
```r
hist(deg, breaks = (min(deg) - 1):(max(deg)) + 0.5, xlab = "In-degree",
    ylab = "Proportion", main = "Histogram of In-Degree Distribution",
    border = "black", col = "white", )
```




```r
deg <- degree(dib_graph, mode = "out")
cat("The in-degree distribution of the graph varies from ", min(deg), "to ",
    max(deg))
```

```
## The in-degree distribution of the graph varies from  0 to  58
```

```r
hist(deg, breaks = (min(deg) - 1):(max(deg)) + 0.5, xlab = "Out-degree",
    ylab = "Proportion", main = "Histogram of Out-Degree Distribution",
    border = "black", col = "white")
```



Section 2 - Bowtie Analysis

2.1 What percentage of the network comprises the strongly connected component, the incomponent, the out-component, and tube.

```
### Strongly connected components
clu <- components(dib_graph, mode = "strong")

scc_index <- which.max(clu$csize)
scc <- V(dib_graph)[which(clu$membership == scc_index)]$name

scc_career <- V(dib_graph)[which(clu$membership == scc_index)]$career
scc_century <- V(dib_graph)[which(clu$membership == scc_index)]$century


### IN components
IN_component = c()
IN_component_career = c()
IN_component_century = c()
vertices <- V(dib_graph)

non_SCC <- vertices[!(vertices$name %in% scc)]

for (v in non_SCC) {

    dist <- bfs(dib_graph, root = v, mode = "out", unreachable = F, dist = T)$dist
```

```
    connected_to_SCC <- !is.nan(dist[scc])

    if (TRUE %in% connected_to_SCC) {
        IN_component <- c(IN_component, V(dib_graph)[v]$name)
        IN_component_career <- c(IN_component_career, V(dib_graph)[v]$career)
        IN_component_century <- c(IN_component_century, V(dib_graph)[v]$century)
    }
}

### OUT components
OUT_component = c()
OUT_component_career = c()
OUT_component_century = c()

non_SCC2 <- vertices[!(vertices$name %in% c(scc, IN_component))]

for (v in non_SCC2) {
    dist <- bfs(dib_graph, root = v, mode = "in", unreachable = F, dist = T)$dist
    connected_to_non_SCC <- !is.nan(dist[scc])
    if (TRUE %in% connected_to_non_SCC) {
        OUT_component <- c(OUT_component, V(dib_graph)[v]$name)
        OUT_component_career <- c(OUT_component_career, V(dib_graph)[v]$career)
        OUT_component_century <- c(OUT_component_century, V(dib_graph)[v]$century)
    }
}

### Tube components
tube = c()
tube_career = c()
tube_century = c()
nodes_minus_SCC <- vertices[!(vertices$name %in% scc)]
g_minus_SCC <- induced_subgraph(dib_graph, nodes_minus_SCC)
out = c()

for (v in V(g_minus_SCC)[IN_component]) {
    paths <- all_simple_paths(g_minus_SCC, from = v, to = V(g_minus_SCC)[OUT_component],
        mode = "out", cutoff = -1)
    tube <- c(tube, names(unlist(paths)))
}

tube <- unique(tube)

tube <- tube[!(tube %in% c(IN_component, OUT_component))]


tube_data = c()
tube_data <- V(dib_graph)[V(dib_graph)$name %in% tube]
tube_data_career <- tube_data$career
tube_data_century <- tube_data$century


### tendril
```

```
in_tendril = c()
Tendril = vertices[!(vertices$name %in% c(scc, IN_component, OUT_component,
    tube))]$name


nodes_IN_component_Tendrils <- vertices[vertices$name %in% c(Tendril, IN_component)]
g_IN_component_Tendrils <- induced.subgraph(dib_graph, nodes_IN_component_Tendrils)


for (v in Tendril) {
    paths <- all_simple_paths(g_IN_component_Tendrils, from = v,
             to = V(g_IN_component_Tendrils)[IN_component],
             mode = "in", cutoff = -1)

    if (length(paths) > 0) {
        in_tendril <- c(in_tendril, v)
    }
}

out_tendril <- c()
nodes_OUT_component_Tendrils <- vertices[vertices$name %in% c(Tendril,
    OUT_component)]

g_OUT_component_Tendrils <- induced.subgraph(dib_graph, nodes_OUT_component_Tendrils)

for (v in Tendril) {
    paths <- all_simple_paths(g_OUT_component_Tendrils, from = v,
             to = V(g_OUT_component_Tendrils)[OUT_component],
             mode = "out", cutoff = -1)

    if (length(paths) > 0) {
        out_tendril <- c(out_tendril, v)
    }
}
```

```
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 4.1.3
```

```
components_prop <- c(length(scc)/length(vertices), length(IN_component)/length(vertices),
    length(OUT_component)/length(vertices), length(tube)/(length(vertices)))

names(components_prop) <- c("SCC", "IN Component", "OUT component", "Tube")

kable(components_prop, col.names = "Percentage")
```

	Percentage
SCC	0.6108819
IN Component	0.3075036
OUT component	0.0248634
Tube	0.0013379

2.2 What are the top five careers for people in the in-component, out-component, scc and tube components?

```
library(dplyr)

scc_career <- trimws(unlist(strsplit(scc_career, split = ",")))
scc_career <- as_tibble(table(scc_career)) %>%
    arrange(desc(n)) %>%
    head(5)


IN_component_career <- trimws(unlist(strsplit(IN_component_career, split = ",")))
IN_component_career <- as_tibble(table(IN_component_career)) %>%
    arrange(desc(n)) %>%
    head(5)

OUT_component_career <- trimws(unlist(strsplit(OUT_component_career, split = ",")))
OUT_component_career <- as_tibble(table(OUT_component_career)) %>%
    arrange(desc(n)) %>%
    head(5)


tube_component_career <- trimws(unlist(strsplit(tube_data_career, split = ",")))
tube_component_career <- as_tibble(table(tube_component_career)) %>%
    arrange(desc(n)) %>%
    head(5)
```

```
names(scc_career)[1] <- "SCC career"
names(IN_component_career)[1] <- "In career"
names(OUT_component_career)[1] <- "Out career"
names(tube_component_career)[1] <- "Tube career"

names(scc_career)[2] <- "SCC Count"
names(IN_component_career)[2] <- "In Count"
names(OUT_component_career)[2] <- "Out Count"
names(tube_component_career)[2] <- "Tube Count"


first_table <- cbind(scc_career, IN_component_career) %>%
    gt() %>%
    tab_header("Top Five Career in SCC and IN Component") %>%
    tab_spanner(label = "SCC", columns = c("SCC career", "SCC Count")) %>%
    tab_spanner(label = "IN Component", columns = c("In career", "In Count"))

second_table <- cbind(OUT_component_career, tube_component_career) %>%
    gt() %>%
    tab_header("Top Five Career in OUT Component and Tube") %>%
    tab_spanner(label = "OUT Component", columns = c("Out career", "Out Count")) %>%
    tab_spanner(label = "Tube Component", columns = c("Tube career", "Tube Count"))

first_table
```

Top Five Career in SCC and IN Component

SCC	IN Component

SCC career	SCC Count	In career	In Count
Politics	1909	Politics	591
Religion	1004	Religion	487
Literature	587	Literature	312
Military	502	Business and Finance	240
Gentry and Aristocracy	486	Journalism and Broadcasting	224

```
second_table
```

Top Five Career in OUT Component and Tube

OUT Component		Tube Component	
Out career	Out Count	Tube career	Tube Count
Religion	37	Military	3
Science and Technology	26	Science and Technology	2
Politics	25	The Sea	2
Sport	20	Travel and Exploration	2
Administration and Diplomacy	18	Administration and Diplomacy	1

2.3 What are the top five centuries represented in the in-component, out-component, scc and tube components?

```
library(dplyr)

scc_century <- as_tibble(table(scc_century)) %>%
    arrange(desc(n)) %>%
    head()

IN_component_century <- as_tibble(table(IN_component_century)) %>%
    arrange(desc(n)) %>%
    head()

OUT_component_century <- as_tibble(table(OUT_component_century)) %>%
    arrange(desc(n)) %>%
    head()

tube_component_century <- as_tibble(table(tube_data_century)) %>%
    arrange(desc(n)) %>%
    head()

names(scc_century)[1] <- "SCC century"
names(IN_component_century)[1] <- "In century"
names(OUT_component_century)[1] <- "Out century"
names(tube_component_century)[1] <- "Tube century"

names(scc_century)[2] <- "SCC count"
names(IN_component_century)[2] <- "In count"
names(OUT_component_century)[2] <- "Out count"
names(tube_component_century)[2] <- "Tube count"
```

```
cbind(scc_century, IN_component_century, OUT_component_century, tube_component_century) %>%
    gt() %>%
    tab_header("Top Five Century in SCC, IN Component, OUT Component, Tube") %>%
    tab_spanner(label = "SCC", columns = c("SCC century", "SCC count")) %>%
    tab_spanner(label = "IN Component", columns = c("In century", "In count")) %>%
    tab_spanner(label = "OUT Component", columns = c("Out century", "Out count")) %>%
    tab_spanner(label = "Tube Component", columns = c("Tube century", "Tube count"))
```

Top Five Century in SCC, IN Component, OUT Component, Tube

SCC		IN Component		OUT Component		Tube Component	
SCC century	SCC count	In century	In count	Out century	Out count	Tube century	Tube count
19	1871	19	1090	19	112	20	8
18	1133	20	789	20	52	19	3
20	597	18	419	18	39	18	1
17	595	17	170	17	10	20	8
16	474	16	120	16	4	19	3
15	118	13	36	13	3	18	1

Section 3 - Centrality /Authority

3.1 Produce a table that shows the most influential people in each century using 3 different measures of centrality/authority

```
library(sjmisc)
```

```
## Warning: package 'sjmisc' was built under R version 4.1.3
```

```
century_data <- list(
  induced.subgraph(dib_graph, V(dib_graph)[V(dib_graph)$century == 9]),
  induced.subgraph(dib_graph, V(dib_graph)[V(dib_graph)$century == 10]),
  induced.subgraph(dib_graph, V(dib_graph)[V(dib_graph)$century == 11]),
  induced.subgraph(dib_graph, V(dib_graph)[V(dib_graph)$century == 12]),
  induced.subgraph(dib_graph, V(dib_graph)[V(dib_graph)$century == 13]),
  induced.subgraph(dib_graph, V(dib_graph)[V(dib_graph)$century == 14]),
  induced.subgraph(dib_graph, V(dib_graph)[V(dib_graph)$century == 15]),
  induced.subgraph(dib_graph, V(dib_graph)[V(dib_graph)$century == 16]),
  induced.subgraph(dib_graph, V(dib_graph)[V(dib_graph)$century == 17]),
  induced.subgraph(dib_graph, V(dib_graph)[V(dib_graph)$century == 18]),
  induced.subgraph(dib_graph, V(dib_graph)[V(dib_graph)$century == 19]),
  induced.subgraph(dib_graph, V(dib_graph)[V(dib_graph)$century == 20]))


page_rank <- vector(mode = "list", length = 12)
authority <- vector(mode = "list", length = 12)
eigen_centrality <- vector(mode = "list", length = 12)
between <- vector(mode = "list", length = 12)
close <- vector(mode = "list", length = 12)

century_list <- c("9th century", "10th century", "11th century", "12th century",
```

```
    "13th century", "14th century", "15th century", "16th century", "17th century",
    "18th century", "19th century", "20th century")

names(century_data) <- century_list


for (i in 1:length(century_data)) {
    page_rank[[i]] <- page_rank(century_data[[i]], directed = TRUE, damping = 0.85)$vector
    page_rank[[i]] <- page_rank[[i]] %>%
        sort(decreasing = TRUE) %>%
        head(1)
    page_rank[[i]] <- paste(names(page_rank[[i]]), "\n", round(page_rank[[i]],
        2))

    eigen_centrality[[i]] <- eigen_centrality(century_data[[i]], weights = NA)$vector
    eigen_centrality[[i]] <- eigen_centrality[[i]] %>%
        sort(decreasing = TRUE) %>%
        head(1)
    eigen_centrality[[i]] <- paste(names(eigen_centrality[[i]]), "\n",
        round(eigen_centrality[[i]], 2))

    between[[i]] <- betweenness(century_data[[i]], weights = NA)
    between[[i]] <- between[[i]] %>%
        sort(decreasing = TRUE) %>%
        head(1)
    between[[i]] <- paste(names(between[[i]]), "\n", round(between[[i]],
        2))

    authority[[i]] <- authority_score(century_data[[i]], scale = TRUE)$vector
    authority[[i]] <- authority[[i]] %>%
        sort(decreasing = TRUE) %>%
        head(1)
    authority[[i]] <- paste(names(authority[[i]]), "\n", round(authority[[i]],
        2))


}
```

```
out <- tibble("Page Rank" = unlist(page_rank),
              "Eigen Centrality" = unlist(eigen_centrality),
              "Between Centrality" = unlist(authority))
out <- rotate_df(out)

colnames(out) <- century_list

out <- as.data.frame(out)

first <- out[c("9th century", "10th century", "11th century", "12th century")] %>%
    gt(rownames_to_stub = TRUE) %>%
    tab_header(title = "1. Centrality for centuries from 9th to 12th")

second <- out[c("10th century", "11th century", "12th century", "13th century")] %>%
    gt(rownames_to_stub = TRUE) %>%
```

```
    tab_header(title = "2. Centrality for centuries from 10th to 13th")

third <- out[c("14th century", "15th century", "16th century", "17th century")] %>%
    gt(rownames_to_stub = TRUE) %>%
    tab_header(title = "3. Centrality for centuries from 14th to 17th")

fourth <- out[c("18th century", "19th century", "20th century")] %>%
    gt(rownames_to_stub = TRUE) %>%
    tab_header(title = "4. Centrality for centuries from 18th to 20th")

first
```

1. Centrality for centuries from 9th to 12th

	9th century	10th century	11th century	12th century
Page Rank	Flann Sinna 0.15	Brian Boru 0.1	Muirchertach Ua Briain 0.09	John (King of England) 0.06
Eigen Centrality	Flann Sinna 1	Brian Boru 1	Muirchertach Ua Briain 1	RuaidrÃ Ua Conchobair 1
Between Centrality	Flann Sinna 1	Brian Boru 1	Muirchertach Ua Briain 1	Henry II 1

```
second
```

2. Centrality for centuries from 10th to 13th

	10th century	11th century	12th century	13th century
Page Rank	Brian Boru 0.1	Muirchertach Ua Briain 0.09	John (King of England) 0.06	Richard Burgh 0.05
Eigen Centrality	Brian Boru 1	Muirchertach Ua Briain 1	RuaidrÃ Ua Conchobair 1	Richard Burgh 1
Between Centrality	Brian Boru 1	Muirchertach Ua Briain 1	Henry II 1	Richard Burgh 1

```
third
```

3. Centrality for centuries from 14th to 17th

	14th century	15th century	16th century	17th century
Page Rank	Richard II 0.08	Gerald FitzGerald 0.07	Hugh O'Neill 0.03	James Butler 0.06
Eigen Centrality	James Butler 1	Gerald FitzGerald 1	Hugh O'Neill 1	James II and VII 1
Between Centrality	Richard II 1	Gerald FitzGerald 1	Hugh O'Neill 1	James II and VII 1

```
fourth
```

4. Centrality for centuries from 18th to 20th

	18th century	19th century	20th century
Page Rank	Daniel O'Connell 0.03	Charles Stewart Parnell 0.02	Jack Lynch 0.02
Eigen Centrality	Wolfe Tone 1	Ã‰amon De Valera 1	Garret FitzGerald 1
Between Centrality	Daniel O'Connell 1	Ã‰amon De Valera 1	Jack Lynch 1

```
dib_graph <- dib_graph + vertices("Smitesh Patil")
dib_graph <- dib_graph + edges("Margaret Alice Joyce", "Smitesh Patil")

vertices <- V(dib_graph)

in_tendril = c()
Tendril = vertices[!(vertices$name %in% c(scc, IN_component, OUT_component,
    tube))]$name


nodes_IN_component_Tendrils <- vertices[vertices$name %in% c(Tendril, IN_component)]
g_IN_component_Tendrils <- induced.subgraph(dib_graph, nodes_IN_component_Tendrils)


for (v in Tendril) {
    paths <- all_simple_paths(g_IN_component_Tendrils, from = v,
             to = V(g_IN_component_Tendrils)[IN_component],
             mode = "in", cutoff = -1)

    if (length(paths) > 0) {
        in_tendril <- c(in_tendril, v)
    }
}


for (v in in_tendril) {
    paths <- all_simple_paths(g_IN_component_Tendrils, from = v,
             to = V(g_IN_component_Tendrils)[IN_component],
             mode = "out", cutoff = -1)

    if (length(paths) > 0) {
        in_tendril <- in_tendril[!v]
    }
}

for (v in out_tendril) {
    paths <- all_simple_paths(g_OUT_component_Tendrils, from = v,
             to = V(g_OUT_component_Tendrils)[OUT_component],
             mode = "in", cutoff = -1)

    if (length(paths) > 0) {
        out_tendril <- out_tendril[!v]
    }
}
```