Analysis of weighted networks

3. Multiplication of networks

Vladimir Batagelj

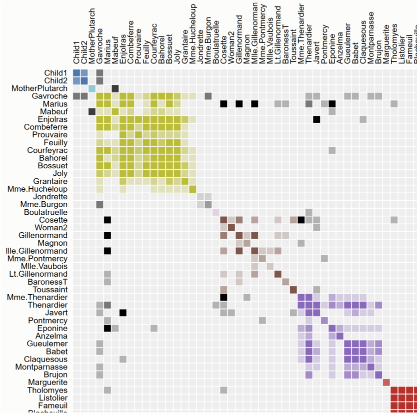
IMFM Ljubljana, UP IAM Koper, UL FMF Ljubljana

INSNA Sunbelt 2025 workshop

June 23-29, 2025, Paris, France

Outline

1. Matrices
2. Multiplication
3. Fractional approach
4. AMC
5. Conclusions
6. References



Les Misérables

Vladimir Batagelj: vladimir.batagelj@fmf.uni-lj.si

Current version of slides (June 23, 2025 at 22 : 23): slides PDF

https://github.com/bavla/Nets

Two mode networks

A multipartite or multirelational network can be split into a collection of simpler one or two-mode networks.

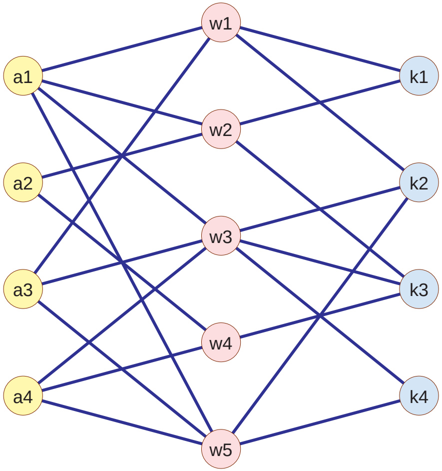
Often, such a collection is already produced in the phase of transforming source data into networks.

A typical example is bibliographic networks (WA, WK, WJ, Ci, WC, etc.) on a selected topic created from data from a bibliographic database (Web of Science, Scopus, OpenAlex, etc.).

For creating bibliographic networks based on OpenAlex, we developed an R package `OpenAlex2Pajek`.

Two mode networks

AW and WK – Authors, Works, and Keywords



Two mode networks

In igraph, a two-mode network has a logical node property type describing the partition of the set of nodes.

```
> library(jsonlite); library(httr)
> # AW <- read_graph("AW.net",format="pajek")
> AW <- read_graph(paste0(nWdir,"/data/AW.net"),format="pajek")
> AW
> V(AW)[[]]
> E(AW)[[]]
> is_bipartite(AW)
> AW <- delete_vertex_attr(AW,"type")
> AW
> is_bipartite(AW)
> V(AW)$type <- bipartite_mapping(AW)$type
> is_bipartite(AW)
> AW$name <- "Authors-Works"
> AW$date <- date()
> AW$twomode <- TRUE
> AW
> plot(AW,main=AW$name)

> # WK <- read_graph(WKfile,format="pajek")
> WK <- read_graph(paste0(nWdir,"/data/WK.net"),format="pajek")
> WK
```

NetsJSON

For archiving networks, we can use JSON format supported by the package `netsWeight`.

```
> saveRDS(AW,file="AW.rds")
> AW1 <- readRDS(file="AW.rds")
> AW1
> write_graph_netsJSON(AW,file="AW.json")
> AW2 <- netsJSON_to_graph(fromJSON("AW.json"),directed=TRUE)
> AW2
> AW3 <- netsJSON_to_graph(fromJSON("AW.json"),directed=FALSE)
> AW3
> V(AW3)[[]]
> E(AW3)[[]]
> graph_attr(AW3)
```

We can combine individual networks into a collection

```
> Bib <- list(format="Collection",
+ info=list(date=date(),by="VB"),nets=list(AW=AW,WK=WK))
> str(Bib)
> saveRDS(Bib,file="BibNets.rds")
```

Two mode networks

A classical example of a two-mode network is the Davis Southern women's network [Batagelj (2022)].

```
> # davis <- read.csv(file.choose(), header=FALSE)
> davis <- read.csv(paste0(nWdir,"/data/davis.csv"))
> D <- graph_from_data_frame(davis, directed=FALSE)
> V(D)$type <- bipartite_mapping(D)$type
> V(D)$name[19:32] <- paste("event-",1:14,sep="")
> is_bipartite(D)
> D$name <- "Deep South"
> D$by <- "Davis, A., Gardner, B.B., Gardner, M.R."
> D$date <- 1941; D$twomode <- TRUE
> saveRDS(D,file="davis.rds")
```

Matrices

igraph provides a link to sparse matrices as supported in the package Matrix. When working with a single weight, it is usually useful to do computations with the corresponding sparse matrices obtained by the netsWeight's function as_sparse_matrix and finally convert (if needed) the resulting matrix back to a network using igraph's function graph_from_matrix.

Multiplication of networks

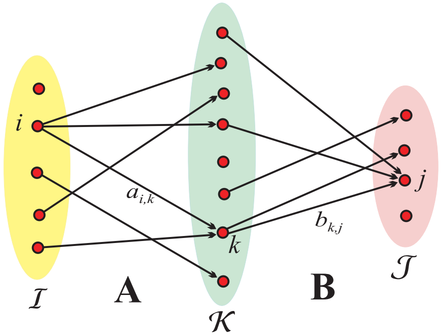
To a simple (no parallel arcs) two-mode *network* $\mathcal{N} = (\mathcal{I}, \mathcal{J}, \mathcal{A}, w)$; where $\mathcal{I}$ and $\mathcal{J}$ are sets of *nodes*, $\mathcal{A}$ is a set of *arcs* linking $\mathcal{I}$ and $\mathcal{J}$, and $w : \mathcal{A} \to \mathbb{R}$ (or some other semiring) is a *weight*; we can assign a *network matrix* $\mathbf{W} = [w_{i,j}]$ with elements: $w_{i,j} = w(i,j)$ for $(i,j) \in \mathcal{A}$ and $w_{i,j} = 0$ otherwise.

Given a pair of compatible networks $\mathcal{N}_A = (\mathcal{I}, \mathcal{K}, \mathcal{A}_A, w_A)$ and $\mathcal{N}_B = (\mathcal{K}, \mathcal{J}, \mathcal{A}_B, w_B)$ with corresponding matrices $\mathbf{A}_{\mathcal{I}\times\mathcal{K}}$ and $\mathbf{B}_{\mathcal{K}\times\mathcal{J}}$ we call a *product of networks* $\mathcal{N}_A$ and $\mathcal{N}_B$ a network $\mathcal{N}_C = (\mathcal{I}, \mathcal{J}, \mathcal{A}_C, w_C)$, where $\mathcal{A}_C = \{(i,j) : i \in \mathcal{I}, j \in \mathcal{J}, c_{i,j} \neq 0\}$ and $w_C(i,j) = c_{i,j}$ for $(i,j) \in \mathcal{A}_C$. The product matrix $\mathbf{C} = [c_{i,j}]_{\mathcal{I}\times\mathcal{J}} = \mathbf{A} * \mathbf{B}$ is defined in the standard way

$$c_{i,j} = \sum_{k\in\mathcal{K}} a_{i,k} \cdot b_{k,j}$$

In the case when $\mathcal{I} = \mathcal{K} = \mathcal{J}$ we are dealing with ordinary one-mode networks (with square matrices).

Multiplication of networks



$$c_{i,j} = \sum_{k \in N_A(i) \cap N_B^-(j)} a_{i,k} \cdot b_{k,j}$$

If all weights in networks $\mathcal{N}_A$ and $\mathcal{N}_B$ are equal to 1 the value of $c_{i,j}$ counts the number of ways we can go from $i \in \mathcal{I}$ to $j \in \mathcal{J}$ passing through $\mathcal{K}$, $c_{i,j} = |oN_A(i) \cap iN_B(j)|$.

Multiplication of networks

The standard matrix multiplication has the complexity $O(|\mathcal{I}| \cdot |\mathcal{K}| \cdot |\mathcal{J}|)$ – it is too slow to be used for large networks. For sparse large networks we can multiply much faster considering only nonzero elements.

> **for** k **in** $\mathcal{K}$ **do**
>   **for** (i,j) **in** $iN_A(k) \times oN_B(k)$ **do**
>   **if** $\exists c_{i,j}$ **then** $c_{i,j} := c_{i,j} + a_{i,k} \cdot b_{k,j}$
>   **else** new $c_{i,j} := a_{i,k} \cdot b_{k,j}$

In general, the multiplication of large, sparse networks is a 'dangerous' operation since the result can 'explode' – it is not sparse.

Multiplication of networks

From the network multiplication algorithm, we see that each intermediate node $k \in \mathcal{K}$ adds to a product network a complete two-mode subgraph $K_{N_A^-(k), N_B(k)}$ (or, in the case $\mathcal{I} = \mathcal{J}$, a complete subgraph $K_{oN(k)}$). If both degrees $\deg_A(k) = |iN_A(k)|$ and $\deg_B(k) = |N_B(k)|$ are large, then already the computation of this complete subgraph has a quadratic (time and space) complexity – the result 'explodes'.

If at least one of the sparse networks $\mathcal{N}_A$ and $\mathcal{N}_B$ has small maximal degree on $\mathcal{K}$ then also the resulting product network $\mathcal{N}_C$ is sparse.

If for the sparse networks $\mathcal{N}_A$ and $\mathcal{N}_B$ there are in $\mathcal{K}$ only few nodes with large degree and no one among them with large degree in both networks then also the resulting product network $\mathcal{N}_C$ is sparse.

Derived networks

We can use the multiplication to obtain new networks from existing *compatible* two-mode networks. For example, from basic bibliographic networks **WA** and **WK** we get

$$\mathbf{AK} = \mathbf{WA}^T \cdot \mathbf{WK}$$

a network relating authors to keywords used in their works, and

$$\mathbf{Ca} = \mathbf{WA}^T \cdot \mathbf{Ci} \cdot \mathbf{WA}$$

is a network of citations between authors.
Networks obtained from existing networks using some operations are called *derived* networks. They are very important in the analysis of collections of *linked* networks.
What is the meaning of the product network? In general, we could consider weights, addition and multiplication over a selected semiring [**?**]. In this presentation, we will limit our attention to the traditional addition and multiplication of real numbers.
The weight $\mathbf{AK}[a, k]$ is equal to the number of times the author a used the keyword k in his/her works.

. . . Derived networks

Using network multiplication we can also transform a given two-mode network, for example **WA**, into corresponding ordinary one-mode networks (*projections*)

$$p_{row}(\mathbf{WA}) = \mathbf{WW} = \mathbf{WA} \cdot \mathbf{WA}^T \quad \text{and} \quad p_{col}(\mathbf{WA}) = \mathbf{AA} = \mathbf{WA}^T \cdot \mathbf{WA}$$

The obtained projections can be analyzed using standard network analysis methods. This is a traditional recipe how to analyze two-mode networks. Often the weights are not considered in the analysis; and when they are considered we have to be very careful about their meaning.

The weight $\mathbf{WW}[p, q]$ is equal to the number of common authors of works p and q.

The weight $\mathbf{AA}[a, b]$ is equal to the number of works that author a and b coauthored. In a special case when $a = b$ it is equal to the number of works that the author a wrote. The network **AA** is describing the *coauthorship* (collaboration) between authors and is also denoted as **Co** – the "first" coauthorship network.

Derived networks

From bibliographic networks
WA – works × authors – authorship network
WK – works × keywords
Ci – works × works – citation network
we can get other interesting networks
$\mathbf{AK} = \mathbf{WA}^T * \mathbf{WK}$ – authors × keywords
$\mathbf{Co} = \mathbf{WA}^T * \mathbf{WA}$ – coauthorship
$\mathbf{ACi} = \mathbf{WA}^T * \mathbf{Ci} * \mathbf{WA}$ – citations between authors

co_{ij} = the number of works that authors i and j wrote together
It holds: $co_{ij} = co_{ji}$.

Using the weights co_{ij} we can determine the Salton's cosine similarity or Ochiai coefficient between authors i and j as

$$S(i,j) = \cos(i,j) = \frac{co_{ij}}{\sqrt{co_{ii}\,co_{jj}}}$$

Outer product decomposition

For vectors $x = [x_1, x_2, \ldots, x_n]$ and $y = [y_1, y_2, \ldots, y_m]$ their *outer product* $x \circ y$ is defined as a matrix

$$x \circ y = [x_i \cdot y_j]_{n \times m}$$

then we can express the previous observation about the structure of product network as the *outer product decomposition*

$$\mathbf{C} = \mathbf{A} \cdot \mathbf{B} = \sum_k \mathbf{H}_k \quad \text{where} \quad \mathbf{H}_k = \mathbf{A}[k, \cdot] \circ \mathbf{B}[k, \cdot]$$

For binary (weights) networks we have $\mathbf{H}_k = K_{N_A^-(k), N_B(k)}$.

Example A

Outer product decomposition

As an example let us take the binary network matrices **WA** and **WK**:

$$\mathbf{WA} = \begin{array}{c} \\ w_1 \\ w_2 \\ w_3 \\ w_4 \\ w_5 \end{array} \begin{array}{c} \begin{array}{cccc} a_1 & a_2 & a_3 & a_4 \end{array} \\ \begin{bmatrix} 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \end{bmatrix} \end{array}, \quad \mathbf{WK} = \begin{array}{c} \\ w_1 \\ w_2 \\ w_3 \\ w_4 \\ w_5 \end{array} \begin{array}{c} \begin{array}{cccc} k_1 & k_2 & k_3 & k_4 \end{array} \\ \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix} \end{array}$$

and compute the product $\mathbf{H} = \mathbf{WA}^T \cdot \mathbf{WK}$. We get a network matrix $\mathbf{H}$ which can be decomposed as

... Example A

Outer product decomposition

$$\begin{array}{c|cccc} \mathbf{H} & k_1 & k_2 & k_3 & k_4 \\ \hline a_1 & 2 & 3 & 2 & 2 \\ a_2 & 1 & 0 & 2 & 0 \\ a_3 & 1 & 3 & 1 & 2 \\ a_4 & 0 & 2 & 2 & 2 \end{array} = \begin{array}{c|cccc} \mathbf{H}_1 & k_1 & k_2 & k_3 & k_4 \\ \hline a_1 & 1 & 1 & 0 & 0 \\ a_2 & 0 & 0 & 0 & 0 \\ a_3 & 1 & 1 & 0 & 0 \\ a_4 & 0 & 0 & 0 & 0 \end{array} + \begin{array}{c|cccc} \mathbf{H}_2 & k_1 & k_2 & k_3 & k_4 \\ \hline a_1 & 1 & 0 & 1 & 0 \\ a_2 & 1 & 0 & 1 & 0 \\ a_3 & 0 & 0 & 0 & 0 \\ a_4 & 0 & 0 & 0 & 0 \end{array}$$

$$\begin{array}{c|cccc} \mathbf{H}_3 & k_1 & k_2 & k_3 & k_4 \\ \hline a_1 & 0 & 1 & 1 & 1 \\ a_2 & 0 & 0 & 0 & 0 \\ a_3 & 0 & 1 & 1 & 1 \\ a_4 & 0 & 1 & 1 & 1 \end{array} + \begin{array}{c|cccc} \mathbf{H}_4 & k_1 & k_2 & k_3 & k_4 \\ \hline a_1 & 0 & 0 & 0 & 0 \\ a_2 & 0 & 0 & 1 & 0 \\ a_3 & 0 & 0 & 0 & 0 \\ a_4 & 0 & 0 & 1 & 0 \end{array} + \begin{array}{c|cccc} \mathbf{H}_5 & k_1 & k_2 & k_3 & k_4 \\ \hline a_1 & 0 & 1 & 0 & 1 \\ a_2 & 0 & 0 & 0 & 0 \\ a_3 & 0 & 1 & 0 & 1 \\ a_4 & 0 & 1 & 0 & 1 \end{array}$$

. . . Outer product decomposition

In the paper [?] it was shown that there can be problems with the network **Co** when we try to use it for identifying the most collaborative authors. By the outer product decomposition, the coauthorship network **Co** is composed of complete subgraphs on the set of the work's coauthors. Works with many authors produce large complete subgraphs, thus blurring the collaboration structure, and are over-represented by their total weight. To see this, let $S_x = \sum_i x_i$ and $S_y = \sum_j y_j$ then the *contribution* of the outer product $x \circ y$ is equal

$$T_{xy} = \sum_{i,j} (x \circ y)_{ij} = \sum_i \sum_j x_i \cdot y_j = \sum_i x_i \cdot \sum_j y_j = S_x \cdot S_y$$

In general, each term $\mathbf{H}_w$ in the outer product decomposition of the product **C** has different total weight $T(\mathbf{H}_w) = \sum_{a,k} (\mathbf{H}_w)_{ak}$ leading to over-representation of works with large values. In the case of coautorship network **Co** we have $S(\mathbf{WA}[w, .]) = \mathrm{odeg}_{\mathbf{WA}}(w)$ and therefore $T(\mathbf{H}_w) = \mathrm{odeg}_{\mathbf{WA}}(w)^2$. To resolve the problem, we apply the fractional approach.

Fractional approach

To make the contributions of all works equal we can apply the *fractional* approach by normalizing the weights: setting $x' = x/S_x$ and $y' = y/S_y$ we get $S_{x'} = S_{y'} = 1$ and therefore $T(\mathbf{H}'_w) = 1$ for all works w.

In the case of two-mode networks **WA** and **WK** we denote

$$S_w^{\mathbf{WA}} = \begin{cases} \mathrm{wod}_{WA}(w) & \mathrm{odeg}_{\mathbf{WA}}(w) > 0 \\ 1 & \mathrm{odeg}_{\mathbf{WA}}(w) = 0 \end{cases} = \max(1, \mathrm{wod}_{WA}(w))$$

(and similarly $S_w^{\mathbf{WK}}$) and define the *normalized* matrices

$$\mathbf{WAn} = n(\mathbf{WA}) = \mathrm{diag}(\frac{1}{S_w^{\mathbf{WA}}}) \cdot \mathbf{WA}, \quad \mathbf{WKn} = n(\mathbf{WK}) = \mathrm{diag}(\frac{1}{S_w^{\mathbf{WK}}}) \cdot \mathbf{WK}$$

In real life networks **WA** (or **WK**) it can happen that some work has no author. In such a case $S_w^{\mathbf{WA}} = \sum_a \mathbf{WA}[w,a] = 0$ which makes problems in the definition of the normalized network **WAn**. We can bypass the problem by setting $S_w^{\mathbf{WA}} = 1$, as we did in the above definition.

Normalized product

Then the *normalized product* matrix is $\mathbf{AKt} = \mathbf{WAn}^T \cdot \mathbf{WKn}$.

Denoting $\mathbf{F}_w = \frac{1}{S_w^{\mathbf{WA}} S_w^{\mathbf{WK}}} \mathbf{H}_w$ the outer product decomposition gets form

$$\mathbf{AKt} = \sum_w \mathbf{F}_w$$

Since

$$T(\mathbf{F}_w) = \begin{cases} 1 & (\mathrm{odeg}_{\mathbf{WA}}(w) > 0) \wedge (\mathrm{odeg}_{\mathbf{WK}}(w) > 0) \\ 0 & \text{otherwise} \end{cases}$$

we have further

$$\sum_{a,k} \mathbf{F}[a,k] = \sum_{a,k} \sum_w \mathbf{F}_w[a,k] = \sum_w T(\mathbf{F}_w) = |W^+|$$

where $W^+ = \{w \in W : (\mathrm{odeg}_{\mathbf{WA}}(w) > 0) \wedge (\mathrm{odeg}_{\mathbf{WK}}(w) > 0)\}$.
In the network **AKt**, the contribution of each work to the bibliography is 1. These contributions are redistributed to arcs from authors to keywords.

Normalizations

in netsWeight

```
normalize_matrix_Markov(M)
normalize_matrix_Newman(M)
normalize_matrix_Balassa(M)
normalize_matrix_activity(M)
```

Example B

Outer product decomposition

For matrices from Example A, we get the corresponding diagonal normalization matrices

$$\mathrm{diag}\left(\frac{1}{S_w^{\mathbf{WA}}}\right) = \begin{array}{c} \\ w_1 \\ w_2 \\ w_3 \\ w_4 \\ w_5 \end{array} \begin{array}{c} \begin{array}{ccccc} w_1 & w_2 & w_3 & w_4 & w_5 \end{array} \\ \begin{bmatrix} 1/2 & 0 & 0 & 0 & 0 \\ 0 & 1/2 & 0 & 0 & 0 \\ 0 & 0 & 1/3 & 0 & 0 \\ 0 & 0 & 0 & 1/2 & 0 \\ 0 & 0 & 0 & 0 & 1/3 \end{bmatrix} \end{array}$$

$$\mathrm{diag}\left(\frac{1}{S_w^{\mathbf{WK}}}\right) = \begin{array}{c} \\ w_1 \\ w_2 \\ w_3 \\ w_4 \\ w_5 \end{array} \begin{array}{c} \begin{array}{ccccc} w_1 & w_2 & w_3 & w_4 & w_5 \end{array} \\ \begin{bmatrix} 1/2 & 0 & 0 & 0 & 0 \\ 0 & 1/2 & 0 & 0 & 0 \\ 0 & 0 & 1/3 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1/2 \end{bmatrix} \end{array}$$

compute the normalized matrices

a_1 a_2 a_3 a_4 k_1 k_2 k_3

. . . Example B

Outer product decomposition

outer products such as

$$\mathbf{F}_1 = \begin{array}{c c} & \begin{array}{cccc} k_1 & k_2 & k_3 & k_4 \end{array} \\ \begin{array}{c} a_1 \\ a_2 \\ a_3 \\ a_4 \end{array} & \left[\begin{array}{cccc} 1/4 & 1/4 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1/4 & 1/4 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{array}\right] \end{array} \qquad \mathbf{F}_5 = \begin{array}{c c} & \begin{array}{cccc} k_1 & k_2 & k_3 & k_4 \end{array} \\ \begin{array}{c} a_1 \\ a_2 \\ a_3 \\ a_4 \end{array} & \left[\begin{array}{cccc} 0 & 1/6 & 0 & 1/6 \\ 0 & 0 & 0 & 0 \\ 0 & 1/6 & 0 & 1/6 \\ 0 & 1/6 & 0 & 1/6 \end{array}\right] \end{array}$$

and finally the product matrix $\mathbf{AKt} = \mathbf{WAn}^T \cdot \mathbf{WKn} =$

$$\sum_{w=1}^{5} \mathbf{F}_w = \begin{array}{c c} & \begin{array}{cccc} k_1 & k_2 & k_3 & k_4 \end{array} \\ \begin{array}{c} a_1 \\ a_2 \\ a_3 \\ a_4 \end{array} & \left[\begin{array}{cccc} 0.50000 & 0.52778 & 0.36111 & 0.27778 \\ 0.25000 & 0.00000 & 0.75000 & 0.00000 \\ 0.25000 & 0.52778 & 0.11111 & 0.27778 \\ 0.00000 & 0.27778 & 0.61111 & 0.27778 \end{array}\right] \end{array}$$

Normalized co-authorship network

Let $\mathbf{N}$ be the normalized version $\forall p \in W : \sum_{i \in A} n_{pi} \in \{0, 1\}$ obtained from $\mathbf{WA}$ by $n_{pi} = wa_{pi} / \max(1, \mathrm{odeg}_{WA}(p))$, or by some other rule determining the author's contribution – the *fractional* approach. Then the *normalized co-authorship network* is

$$\mathbf{Cn} = \mathbf{N}^T \cdot \mathbf{N}$$

cn_{ij} = the total contribution of 'collaboration' of authors i and j to works.

It holds $cn_{ij} = cn_{ji}$ and $\sum_{i \in A} \sum_{j \in A} n_{pi} n_{pj} = 1$.

The total contribution of a complete subgraph corresponding to the authors of a work p is 1.

$$\sum_{i \in A} \sum_{j \in A} cn_{ij} = |W^+|$$

$W^+ = \{w \in W : \deg(w) > 0\}$.

Normalized strict co-authorship network

Newman defined a *strict normalization* $\mathbf{N}'$ obtained from $\mathbf{WA}$ by $n'_{pi} = wa_{pi} / \max(1, \mathrm{odeg}_{WA}(p) - 1)$. Then the *normalized strict co-authorship network* is

$$\mathbf{Cn}' = \mathbf{N}^T \cdot \mathbf{N}'$$

The diagonal (loops) of the so-obtained network $\mathbf{Cn}'$ is set to 0.

The network $\mathbf{Cn}'$ doesn't consider the contribution of single-author works.

OpenAlex, client-libraries, OpenAlex2Pajek

```
> setwd(wdir <- "C:/Users/vlado/docs/papers/2025/AS/AMC")
> library(httr); library(jsonlite)
> source("https://raw.githubusercontent.com/bavla/Rnet/master/R/Pajek.R"
> source("https://raw.githubusercontent.com/bavla/OpenAlex/
    main/code/OpenAlex2Pajek.R")
> sID <- "S61442588"
> R <- OpenAlexSources(sID,step=250)
OpenAlex2Pajek / Sources Mon May 26 05:59:58 2025
...
865 source S61442588 works collected Mon May 26 06:00:00 2025
..
 5231 citing works collected Mon May 26 06:05:46 2025
...
 12530 cited works collected Mon May 26 06:05:54 2025
12758 different works Mon May 26 06:05:54 2025
> csv <- file("worksAMC.csv","w",encoding="UTF-8")
> write(R,sep="\n",file=csv)
> close(csv)
```

```
> OpenAlex2PajekAll(NULL,name="AMC",listF="worksAMC.csv")
OpenAlex2Pajek / All - Start Mon May 26 06:10:05 2025
*** OpenAlex2Pajek / All - Process Mon May 26 06:10:05 2025
Mon May 26 06:13:38 2025  n = 500
Mon May 26 06:16:25 2025  n = 1000
...
Mon May 26 07:19:31 2025  n = 12500
*** OpenAlex2Pajek / All - Data Collected Mon May 26 07:20:44 2025
hits: 12758 works: 137751 authors: 10849 anon: 185 sources: 1192
>>> Citation Cite
>>> publication year
>>> type of publication
>>> language of publication
>>> cited by count
>>> countries distinct count
>>> referenced works
>>> Authorship WA
>>> Sources WJ
>>> Countries WC
>>> Keywords WK
*** OpenAlex2Pajek / All - Stop Mon May 26 07:21:54 2025
```

```
> library(httr); library(jsonlite); library(pryr)
> OAdir <- "https://raw.githubusercontent.com/bavla/OpenAlex/"
> OAdata <- paste0(OAdir,"refs/heads/main/data/")
> OAcode <- paste0(OAdir,"main/code/")
> source(paste0(OAcode,"OpenAlex2Pajek.R"))

> WA <- read_graph(paste0(OAdata,"AMC/WA.net"),format="pajek")
> (nW <- sum(!V(WA)$type))
> (nA <- sum(V(WA)$type))

> (x <- exp(1))   # space needed
> bytes(x)
[1] "40 05 BF 0A 8B 14 57 69"
> as.integer(object.size(WA))/nW/nA/8
[1] 0.001895938

> WJ <- read_graph(paste0(OAdata,"AMC/WJ.net"),format="pajek")
> (nJ <- sum(V(WJ)$type))
> Ci <- read_graph(paste0(OAdata,"AMC/Ci.net"),format="pajek")

> source <- "S61442588"  # OA id of ACM
> (is <- which(V(WJ)$name==source))
[1] 137766
> s <- rep(0,nJ); s[is-nW] <- 1
> MWJ <- as_sparse_matrix(WJ)
> Wj <- MWJ %*% s
> sum(Wj)
[1] 865
```

We first clean the networks **Ci**, **WA**, **WJ**, . . ., removing multiple links and loops.
The product $\mathbf{u} = \mathbf{A} \cdot \mathbf{v}$ of the network $\mathbf{A}$ with the vector $\mathbf{v}$ is defined as

$$u_i = \sum_{j:(i,j) \in L} A_{ij} \cdot v_j$$

We need the index j of the node $source =$ `"S61442588"` representing AMC in the set of journals J. We can get it from **WJ**: `is <- which(V(WJ)$name==source)` and $j = is - nW = 15$.
We start with the set W_j of all works published by the journal j.

$$W_j = \{w : WJ[w,j] > 0\}$$

Let $\mathbf{w}_j$ be its characteristic vector. Then $\mathbf{w}_j = \mathbf{WJ} \cdot [j]$ where $[j]$ is a vector over J having 1 at the jth place. We create the vector $[j]$

AMC analysis

works citing

Next, for the set W_j, we determine the set W_I of citing works and the set W_O of cited works.

$$W_I = \{w : \exists z \in W_j : Ci[w,z] > 0\} \quad \text{and} \quad W_O = \{w : \exists z \in W_j : Ci[z,$$

The vectors $\mathbf{d}_I = \mathbf{Ci} \cdot \mathbf{w}_j$ and $\mathbf{d}_O = \mathbf{Ci}^T \cdot \mathbf{w}_j$

$$d_I(i) = \sum_k Ci[i,k] \cdot w_j(k), \quad d_O(i) = \sum_k Ci^T[i,k] \cdot w_j(k) = \sum_k Ci[k,i] \cdot w_j(k)$$

count: $d_I(i)$ - how many works from W_j are cited by the work i; and $d_O(i)$ - how many works from W_j are citing the work i.
Inspect the vector dI. We list the largest 20 nodes. We will collect from OpenAlex the additional information about the selected works. It turns out that the authors' names are not directly accessible as a data field - they are contained inside the field "authorships". To extract them, we use the function `authors` embedded in the function `unitsInfo`.

AMC analysis

adding information

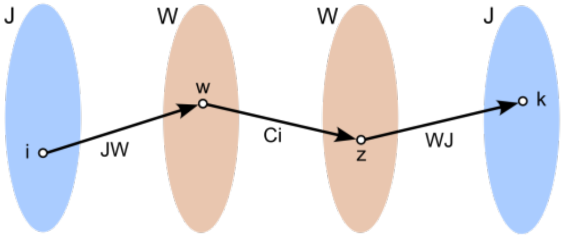
Now we are ready to get the information about the selected works. Some data ("authors" and "title") can be very long. To get a readable report we truncate them.

```
> MCi <- as_sparse_matrix(Ci); namCi <- rownames(MCi)
> dI <- MCi %*% Wj
> p <- order(dI,decreasing=TRUE)
> selI <- p[1:20]
> (topI <- cbind(selI,namCi[selI],dI[selI]))
> selW <- paste0("id,language,countries_distinct_count,cited_by_count,",
+   "relevance_score,publication_year,title,authorships")
> IWi <- unitsInfo(IDs=namCi[selI],units="works",select=selW,order="inpu
> rep <- data.frame(id=IWi$id,cdc=IWi$countries_distinct_count,
+   cby=IWi$cited_by_count,dI=dI[selI],year=IWi$publication_year,
+   authors=substr(authors(IWi),1,35),title=substr(IWi$title,1,45))
> rep
```

Some improvements: add source; in names, use only the last name, or initials + last name; ...
We can check selected works - for example W1846554597.
Using the same approach as for $\mathbf{d}_I$ we get also results for $\mathbf{d}_O$.

AMC analysis

Citations between journals



$$\mathbf{JJ} = \mathbf{WJ}^T \cdot \mathbf{Ci} \cdot \mathbf{WJ}$$

$JJ[i,k] = \#$ of citations of a work from journal i to a work from journal $k \equiv \#$ of times journal i cites journal k.
$n_{JJ} = 1192$, $m^A_{JJ} = 20011$, and 234 loops.
Inspecting the weights, we select the threshold $t = 100$. We make a link cut at level t.
There is a problem – "unitsInfo" doesn't like the source "Sunknown". We replace it with resource 1 (duplicated). Now it works.

AMC analysis

Citations between journals

```
> MJJ <- crossprod(MWJ,MCi) %*% MWJ
> nJJ <- nrow(MJJ); mJJ <- sum(MJJ>0); kJJ <- sum(diag(MJJ)>0)
> JJ <- graph_from_adjacency_matrix(MJJ,mode="directed",weighted=TRUE)
> JJ$project <- "Ars Mathematica Contemporanea"
> JJ$name <- "journal citation network"
> JJ$date <- date()
> JJ$by <- "Vladimir Batagelj"
> saveRDS(JJ,file="AMC_JJ.rds")

> w <- E(simplify(JJ))$weight
> r <- order(w,decreasing=TRUE)
> w[r[1:100]]
> LC <- link_cut(simplify(JJ),atn="weight",100)
> (S <- V(LC)$name)
> un <- 3; S[un] <- S[1]  # unknown
> selS <- "id,issn_l,country_code,type,is_oa,cited_by_count,
+   works_count,display_name"
> IS <- unitsInfo(IDs=S,units="sources",select=selS,order="input")
> IS$display_name[un] <- IS$id[un] <- "Sunknown"; IS$issn_l[un] <- NA
> rep <- data.frame(id=IS$id,issn_l=IS$issn_l,journal=IS$display_name)
> rep
> V(LC)$source <- IS$display_name
> lo <- layout_with_dh(LC)
> plot(LC,layout=lo,edge.width=log(w),vertex.color="pink",
+    vertex.size=12,vertex.label=V(LC)$source,vertex.label.cex=0.7)
```

AMC analysis

Shorten the journal titles: Mathematics, Mathematical, Mathematische, Mathematicae $\rightarrow$ M; Combinatorics, Combinatorial $\rightarrow$ Comb; Computation, Computational $\rightarrow$ Comp; Computer Science $\rightarrow$ CS; Proceedings of the $\rightarrow$ P; Transactions of the $\rightarrow$ T; Bulletin of the $\rightarrow$ B; Journal of $\rightarrow$ J; Mathematical Society $\rightarrow$ MS; Discrete $\rightarrow$ Disc, etc.

Note, the network WJ is essentially a "function".

There are other options to produce an "interesting" subnetwork

- Extract the node cut for weighted degree at selected level.
- Determine the set of "interesting" nodes as a weighted degree core (Ps-core).

$$\mathbf{ACiA} = \mathbf{WA}^T \cdot \mathbf{Ci} \cdot \mathbf{WA}$$

$ACiA[a,b] = \#$ of citations of a work of author a to a work of author $b \equiv \#$ of times author a cites author b.
$n_{ACiA} = 10849$, $m^A_{ACiA} = 248183$, and 2251 loops.
Inspecting the weights we select the threshold $t = 100$. We make a link cut at level t.
To get a more readable network visualization, we have to replace IDs with the corresponding author names. The procedure is as in the case of journals. Normalized version!?

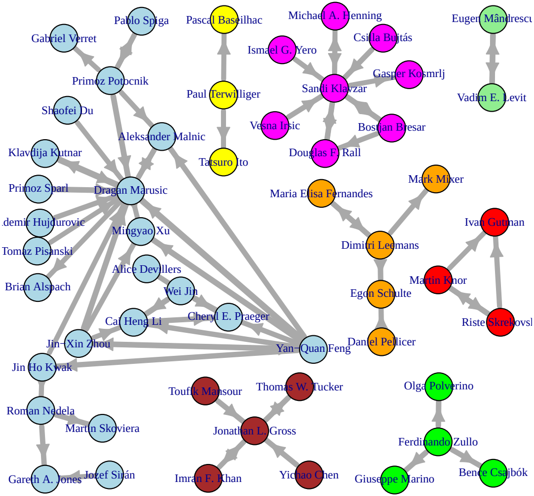
AMC authors

$$\mathbf{a}_j = \mathbf{WA}^T \cdot \mathbf{w}_j$$

$a_j(a) = \#$ of works in the journal j co-authored by the author a.
Again we need to replace IDs in the report by author names.

AMC analysis

Citations between authors



Conclusions

1. The package `netsWeight` is still in development. New functions will be added (islands, hubs and authorities, trusses, etc.) and it will probably merged with some other packages (`ClusNet`, `NetsJSON`,???)
2. The example is dealing with bibliographic network analysis. The proposed approach can be used for analysis of any collection of linked networks.

Acknowledgments

The computational work reported in this presentation was performed using R. The code and data are available at GitHub/Vlado.

This work is supported in part by the Slovenian Research Agency (research program P1-0294 and research project J5-4596), and prepared within the framework of the COST action CA21163 (HiTEc).

References I

- Batagelj, V. On fractional approach to analysis of linked networks. Scientometrics 123, 621–633 (2020). https://doi.org/10.1007/s11192-020-03383-y
- Batagelj, V, Cerinšek, M: On bibliographic networks. Scientometrics 96 (2013) 3, 845-864.
- Batagelj, Vladimir. Analysis of the Southern women network using fractional approach. Social Networks 68 (2022): 229-236.
- Batagelj, V: On fractional approach to analysis of linked networks. Scientometrics 123 (2020) 2: 621-633
- Batagelj, V, Maltseva, D: Temporal bibliographic networks. Journal of Informetrics, Volume 14, Issue 1, February 2020, 101006.
- Maltseva, D, Batagelj, V: Social network analysis as a field of invasions: bibliographic approach to study SNA development. Scientometrics, 121(2019)2, 1085-1128