

Temporal Bibliographic Networks

Vladimir Batagelj^{1,2,3} Daria Maltseva¹

ORCID: 0000-0002-0240-9446, 0000-0003-1789-1711

¹ National Research University Higher School of Economics,
Myasnitskaya, 20, 101000 Moscow, Russia.

²Institute of Mathematics, Physics and Mechanics,
Jadranska 19, 1000 Ljubljana, Slovenia

³University of Primorska, Andrej Marušič Institute, 6000 Koper, Slovenia
vladimir.batagelj@fmf.uni-lj.si
d.maltseva@mail.ru

March 1, 2019

Abstract

We present two ways (instantaneous and cumulative) to transform bibliographic networks, using the works' publication year, into corresponding temporal networks based on temporal quantities. We also show how to use the addition of temporal quantities to define interesting temporal properties of nodes, links and their groups thus providing an insight into evolution of bibliographic networks. Using the multiplication of temporal networks we obtain different derived temporal networks providing us with new views on studied networks. The proposed approach is illustrated with examples from the collection of bibliographic networks on peer review.

Keywords: social network analysis, temporal networks, linked networks, bibliographic networks, temporal quantities, semiring, network multiplication, fractional approach.

MSC: 01A90, 91D30, 90B10, 65F30

JEL: C55, D85

1 Introduction

From data collected from bibliographic databases (WoS, Scopus, Google scholar, Bibtex, etc.) we can construct different bibliographic networks. For example using the program WoS2Pajek we obtain from data collected from WoS the following two-mode networks: the authorship network **WA** on works \times authors, the journalship network **WJ** on works \times journals, the keywordship network **WK** on works \times keywords, and the (one-mode) citation network **Cite** on works. We obtain also the following node properties: the partition *year* of works by publication year, the *DC* partition distinguishing between works with complete description ($DC[w] = 1$) and the cited only works ($DC[w] = 0$), and the vector of number of pages *NP*. Analyzing these networks we can get distributions of frequencies of different units (authors, journals, keywords) describing overall properties of networks. We can also identify the most important units (Cerinšek and Batagelj, 2015). An important tool in the analysis of linked (collections of) networks is the network multiplication that produces derived networks linking not directly linked sets of units – for example, the network $\mathbf{AK} = \mathbf{WA}^T \cdot \mathbf{WK}$ links authors to keywords (Batagelj and Cerinšek, 2013).

A more detailed insight in the evolution of bibliographic networks is enabled by considering also the temporal information. In the paper Batagelj and Praprotnik (2016) a longitudinal approach to analysis of temporal networks based on temporal quantities was presented. It is an alternative to the traditional cross-sectional approach. In this paper we show how to apply the proposed approach to temporal bibliographic networks. It can be used also in other similar contexts.

First we describe two ways how the year of publication can be combined with traditional bibliographic networks to get their temporal versions – the instantaneous and the cumulative. Afterward we present different ways to analyze these networks and networks derived from them using network multiplication.

The proposed approach is illustrated with examples on networks from the collection of bibliographic networks on peer review (Batagelj et al., 2017) on works with complete descriptions. The sizes of different sets of units are as follows: $|W| = 22104$, $|A| = 62106$, $|J| = 6716$, and $|K| = 36275$.

2 Temporal networks

A *temporal network* $\mathcal{N}_T = (\mathcal{V}, \mathcal{L}, \mathcal{T}, \mathcal{P}, \mathcal{W})$ is obtained by attaching the *time*, \mathcal{T} , to an ordinary network where \mathcal{T} is a set of *time points*, $t \in \mathcal{T}$.

In a temporal network, nodes $v \in \mathcal{V}$ and links $l \in \mathcal{L}$ are not necessarily present or active in all time points. Let $T(v)$, $T \in \mathcal{P}$, be the *activity set* of time points for node v and $T(l)$, $T \in \mathcal{W}$, the activity set of time points for link l .

Besides the presence/absence of nodes and links also their properties can change through time.

2.1 Temporal quantities

We introduce a notion of a *temporal quantity*

$$a(t) = \begin{cases} a'(t) & t \in T_a \\ \mathfrak{K} & t \in \mathcal{T} \setminus T_a \end{cases}$$

where T_a is the *activity time set* of a , $a'(t)$ is the value of a in an instant $t \in T_a$, and \mathfrak{K} denotes the value *undefined*.

We assume that the values of temporal quantities belong to a set A which is a *semiring* $(A, +, \cdot, 0, 1)$ for binary operations $+: A \times A \rightarrow A$ and $\cdot: A \times A \rightarrow A$. The semiring $(\mathbb{R}_0^+, +, \cdot, 0, 1)$ where $+$ is addition and \cdot is multiplication of numbers is called a *combinatorial* semiring. For solving the shortest path problems on networks the semiring $(\mathbb{R}_0^+ \cup \{\infty\}, \min, +, \infty, 0)$ is used (Baras and Theodorakopoulos, 2010).

We can extend both operations to the set $A_{\mathfrak{K}} = A \cup \{\mathfrak{K}\}$ by requiring that for all $a \in A_{\mathfrak{K}}$ it holds

$$a + \mathfrak{K} = \mathfrak{K} + a = a \quad \text{and} \quad a \cdot \mathfrak{K} = \mathfrak{K} \cdot a = \mathfrak{K}.$$

The structure $(A_{\mathfrak{K}}, +, \cdot, \mathfrak{K}, 1)$ is also a semiring.

Let $A_{\mathfrak{K}}(\mathcal{T})$ denote the set of all temporal quantities over $A_{\mathfrak{K}}$ in time \mathcal{T} . To extend the operations to networks and their matrices we first define the *sum* (parallel links) $a + b$ as

$$(a + b)(t) = a(t) + b(t) \quad \text{and} \quad T_{a+b} = T_a \cup T_b.$$

The *product* (sequential links) $a \cdot b$ is defined as

$$(a \cdot b)(t) = a(t) \cdot b(t) \quad \text{and} \quad T_{a \cdot b} = T_a \cap T_b.$$

Let us define temporal quantities $\mathbf{0}$ and $\mathbf{1}$ with requirements $\mathbf{0}(t) = \mathbb{K}$ and $\mathbf{1}(t) = 1$ for all $t \in \mathcal{T}$. Again, the structure $(A_{\mathbb{K}}(\mathcal{T}), +, \cdot, \mathbf{0}, \mathbf{1})$ is a semiring.

To produce a software support for computation with temporal quantities we limit it to temporal quantities that can be described as a sequence of disjoint time intervals with a constant value

$$a = [(s_i, f_i, v_i)]_{i \in 1..k}$$

where s_i is the starting time and f_i the finishing time of the i -th time interval $[s_i, f_i)$, $s_i < f_i$ and $f_i \leq s_{i+1}$, and v_i is the value of a on this interval. Outside the intervals the value of temporal quantity a is undefined, \mathbb{K} . Therefore

$$T_a = \bigcup_{i \in 1..k} [s_i, f_i).$$

To illustrate both operations let us consider temporal quantities a and b (Batagelj and Praprotnik, 2016):

$$\begin{aligned} a &= [(1, 5, 2), (6, 8, 1), (11, 12, 3), (14, 16, 2), (17, 18, 5), (19, 20, 1)] \\ b &= [(2, 3, 4), (4, 7, 3), (9, 10, 2), (13, 15, 5), (16, 21, 1)] \end{aligned}$$

The following are the sum $s = a + b$ and the product $p = a \cdot b$ of temporal quantities a and b over combinatorial semiring.

$$\begin{aligned} s &= [(1, 2, 2), (2, 3, 6), (3, 4, 2), (4, 5, 5), (5, 6, 3), (6, 7, 4), (7, 8, 1), (9, 10, 2), (11, 12, 3), \\ &\quad (13, 14, 5), (14, 15, 7), (15, 16, 2), (16, 17, 1), (17, 18, 6), (18, 19, 1), (19, 20, 2), (20, 21, 1)] \\ p &= [(2, 3, 8), (4, 5, 6), (6, 7, 3), (14, 15, 10), (17, 18, 5), (19, 20, 1)] \end{aligned}$$

They are visually displayed in Figure 1.

To support computations with temporal quantities and analysis of temporal networks based on them the Python libraries TQ and Nets were developed (Batagelj, 2017). They were used in analyses presented in this paper. In the examples we used a collection of bibliographic networks on peer review from Batagelj et al. (2017).

2.2 Temporal affiliation networks

Let the binary *affiliation* matrix $\mathbf{A} = [a_{ep}]$ describe a two-mode network on the set of events E and the set of participants P :

$$a_{ep} = \begin{cases} 1 & p \text{ participated at the event } e \\ 0 & \text{otherwise} \end{cases}$$

The function $d : E \rightarrow \mathcal{T}$ assigns to each event e the date $d(e)$ when it happened. Assume $\mathcal{T} = [first, last] \subset \mathbb{N}$. Using these data we can construct two temporal affiliation matrices:

- **instantaneous** $\mathbf{Ai} = [ai_{ep}]$, where

$$ai_{ep} = \begin{cases} [(d(e), d(e) + 1, 1)] & a_{ep} = 1 \\ [] & \text{otherwise} \end{cases}$$

- **cumulative** $\mathbf{Ac} = [ac_{ep}]$, where

$$ac_{ep} = \begin{cases} [(d(e), last + 1, 1)] & a_{ep} = 1 \\ [] & \text{otherwise} \end{cases}$$

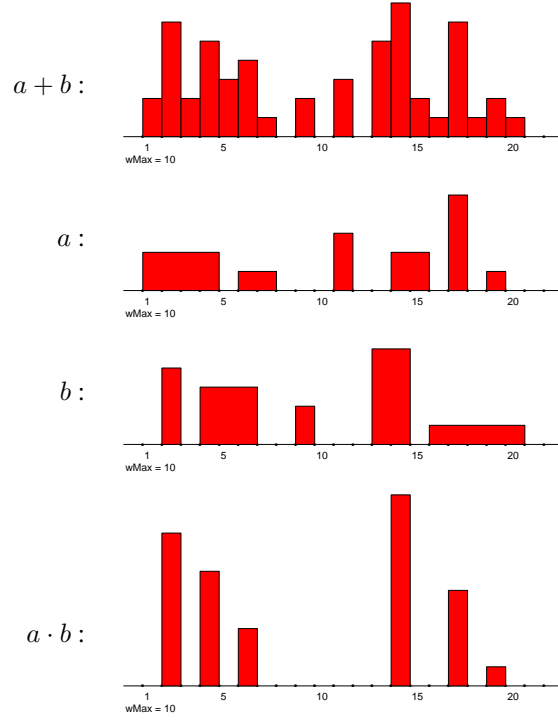


Figure 1: Sum and product of temporal quantities

In general a temporal quantity a is called *cumulative* iff it has for $t, t' \in \mathcal{T}$ the property

$$t \in T_a \wedge t' > t \Rightarrow t' \in T_a \wedge a(t') \geq a(t)$$

A sum and product (over combinatorial semiring) of cumulative temporal quantities are cumulative temporal quantities.

For a temporal quantity $a = [(s_i, f_i, v_i)]_{i \in 1..k}$ its *cumulative* $cum(a)$ is defined as

$$cum(a) = [(s_i, s_{i+1}, V_i)]_{i \in 1..k}$$

where $s_{k+1} = last$ and $V_i = \sum_{j=1}^i v_j$.

A temporal network is cumulative for a weight w iff all its values are cumulative.

The Python code for creating temporal networks from Pajek files for the peer review data is given in Appendix A.1.

2.3 Temporal properties

Let \mathbf{N} be a temporal network on $E \times P$. On it we can define some interesting temporal quantities such as *in-sum*:

$$iS(\mathbf{N}, p) = \sum_{e \in E} n_{ep}$$

and *out-sum*:

$$oS(\mathbf{N}, e) = \sum_{p \in P} n_{ep}$$

In a special case where $\mathbf{N} \equiv \mathbf{W}\mathbf{A}\mathbf{i}$ we get the *productivity of an author a*

$$pr(a) = iS(\mathbf{W}\mathbf{A}\mathbf{i}, a) = \text{number of publications of the author } a \text{ by year}$$

and for $\mathbf{N} \equiv \mathbf{W}\mathbf{A}\mathbf{c}$ we get the *cumulative productivity of an author a*

$$cpr(a) = iS(\mathbf{W}\mathbf{A}\mathbf{c}, a) = \text{cumulative number of publications of the author } a \text{ by year.}$$

It holds $cpr(a) = cum(pr(a))$.

The productivity of an author can be extended to the *productivity of a group of authors C*

$$pr(C) = \sum_{a \in C} pr(a) = \sum_{a \in C} iS(\mathbf{W}\mathbf{A}\mathbf{i}, a)$$

There is a problem with the productivity of a group. In the case when two authors from a group co-authored the same paper it is counted twice. To account for a “real” contribution of each author the fractional approach is used. It is based on normalized networks (matrices) – in the case of co-authorship on $n(\mathbf{W}\mathbf{A}) = \mathbf{W}\mathbf{A}\mathbf{n} = [wan_{wa}]$

$$wan_{wa} = \frac{wa_{wa}}{\max(1, \text{outdeg}_{\mathbf{W}\mathbf{A}}(w))}.$$

This leads to the *fractional productivity of an author a*

$$fpr(a) = iS(\mathbf{W}\mathbf{A}\mathbf{n}\mathbf{i}, a) = \text{fractional contribution of publications of the author } a \text{ by year}$$

2.3.1 Example: Temporal properties in networks on peer review

In the analysis of the ordinary authorship network $\mathbf{W}\mathbf{A}$ we get that Lutz Bornmann is the author who wrote the largest number, 61, of works on peer review (Batagelj et al., 2017). To see the dynamics of his publishing we compute his productivity

$$pr = [(2005, 2006, 4), (2006, 2007, 3), (2007, 2008, 4), (2008, 2009, 9), \\ (2009, 2010, 4), (2010, 2011, 14), (2011, 2012, 5), (2012, 2013, 7), \\ (2013, 2014, 2), (2014, 2015, 3), (2015, 2016, 6)]$$

see the top of Figure 2. The corresponding cumulative productivity is

$$cpr = [(2005, 2006, 4), (2006, 2007, 7), (2007, 2008, 11), (2008, 2009, 20), \\ (2009, 2010, 24), (2010, 2011, 38), (2011, 2012, 43), (2012, 2013, 50), \\ (2013, 2014, 52), (2014, 2015, 55), (2015, 2016, 61)]$$

see the mid of Figure 2. Note that $cpr = cum(pr)$: $7 = 4 + 3$, $11 = 4 + 3 + 4$, ...

The fractional productivity of Lutz Bornmann is

$$fpr = [(2005, 2006, 2.0), (2006, 2007, 1.333), (2007, 2008, 1.667), (2008, 2009, 3.667), \\ (2009, 2010, 1.533), (2010, 2011, 6.3), (2011, 2012, 2.033), (2012, 2013, 3.25), \\ (2013, 2014, 1.0), (2014, 2015, 3.0), (2015, 2016, 3.333)]$$

see the bottom of Figure 2. For the Python code see Appendix A.2.

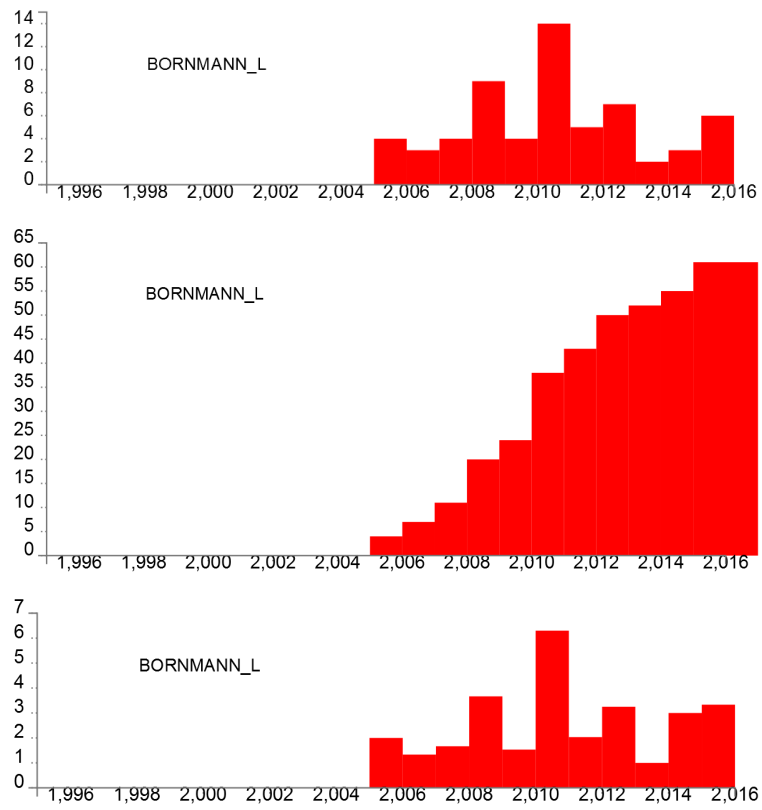


Figure 2: Productivity, cumulative productivity and fractional productivity of Lutz Bornmann

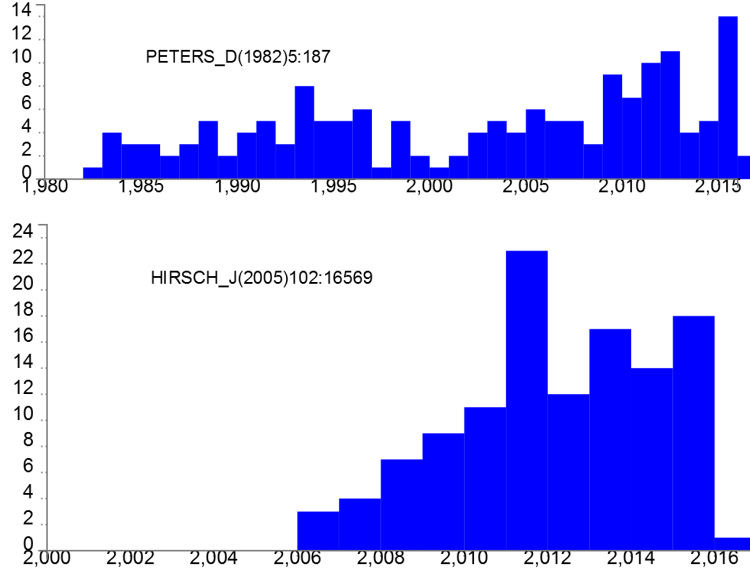


Figure 3: Citations to Peters and Hirsch

In the citation network **Cite** for the peer review bibliography the most cited, 164, paper is Peters, D. P., Ceci, S. J. (1982). Peer-review practices of psychological journals: The fate of published articles, submitted again. Behavioral and Brain Sciences, 5(2), 187-255. The temporal quantity $citP = iS(\text{Citei}, \text{PETERS_D}(1982)5:187)$ describes the number of citations to this paper through years.

$$\begin{aligned}
 citP = & [(1982, 1983, 1), (1983, 1984, 4), (1984, 1986, 3), (1986, 1987, 2), \\
 & (1987, 1988, 3), (1988, 1989, 5), (1989, 1990, 2), (1990, 1991, 4), \\
 & (1991, 1992, 5), (1992, 1993, 3), (1993, 1994, 8), (1994, 1996, 5), \\
 & (1996, 1997, 6), (1997, 1998, 1), (1998, 1999, 5), (1999, 2000, 2), \\
 & (2000, 2001, 1), (2001, 2002, 2), (2002, 2003, 4), (2003, 2004, 5), \\
 & (2004, 2005, 4), (2005, 2006, 6), (2006, 2008, 5), (2008, 2009, 3), \\
 & (2009, 2010, 9), (2010, 2011, 7), (2011, 2012, 10), (2012, 2013, 11), \\
 & (2013, 2014, 4), (2014, 2015, 5), (2015, 2016, 14), (2016, 2017, 2)]
 \end{aligned}$$

See the top of Figure 3.

Another well known paper is Hirsch, J.E. (2005). An index to quantify an individual's scientific research output. Proc Natl Acad Sci U S A. 2005 Nov 15;102(46):16569-72 with 119 citations and $citH = iS(\text{Citei}, \text{HIRSCH_J}(2005)102:16569)$

$$\begin{aligned}
 citH = & [(2006, 2007, 3), (2007, 2008, 4), (2008, 2009, 7), (2009, 2010, 9), \\
 & (2010, 2011, 11), (2011, 2012, 23), (2012, 2013, 12), (2013, 2014, 17), \\
 & (2014, 2015, 14), (2015, 2016, 18), (2016, 2017, 1)]
 \end{aligned}$$

See the bottom of Figure 3. For the Python code see Appendix A.3.

Similarly we could look at the number of works by year $wy = \sum_{j \in J} iS(\mathbf{WJi}, j)$, the popularity of a keyword k : $pop(k) = iS(\mathbf{WKi}, k)$, etc.

3 Network multiplication and derived networks

Let \mathbf{A} on $A \times P$ and \mathbf{B} on $P \times B$ be (matrices of linked two-mode) networks. Their *product* network is determined by a matrix $\mathbf{C} = [c_{i,j}]$ on $A \times B$ of the product of corresponding matrices

$$\mathbf{C} = \mathbf{A} \cdot \mathbf{B}$$

where

$$c_{i,j} = \sum_{p \in P} a_{i,p} \cdot b_{p,j}$$

For details see Batagelj and Cerinšek (2013).

Network multiplication is very important in network analysis of collections of linked networks because it enables us to construct different *derived* networks. For example, in analysis of bibliographic networks the network

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

links authors to keywords: the weight of the arc from the node a to the node k is equal to the number of works in which the author a used the keyword k .

The coauthorship network \mathbf{Co} is obtained as

$$\mathbf{Co} = \mathbf{WA}^T \cdot \mathbf{WA}$$

The weight co_{ab} is equal to total number of works authors a and b wrote together.

The network of normalized citations between authors

$$\mathbf{CiteAn} = n(\mathbf{WA})^T \cdot n(\mathbf{Cite}) \cdot n(\mathbf{WA})$$

The weight $citean_{uv}$ is equal to the fractional contribution of citations from works coauthored by u to works coauthored by v . Etc.

The network (matrix) multiplication can be straightforwardly extended to temporal networks.

3.1 Multiplication of temporal networks

Let \mathbf{A} on $A \times P$ and \mathbf{B} on $P \times B$ be (matrices of) co-occurrence networks. Then $\mathbf{C} = \mathbf{A} \cdot \mathbf{B}$ is a temporal network on $A \times B$. What is its meaning? Consider the value of its item in an instant t

$$c_{ij}(t) = \sum_{p \in P} a_{ip}(t) \cdot b_{pj}(t) = \sum_{p \in P} a_{pi}(t) \cdot b_{pj}(t)$$

For $c_{ij}(t)$ to be defined (different from \mathfrak{K}) there should be at least one $p \in P$ such that $a_{pi}(t)$ and $b_{pj}(t)$ are both defined, i.e. $t \in T_{a_{pi}} \cap T_{b_{pj}}$. Then there exists g_{pi} such that $(s_{g_{pi}}, f_{g_{pi}}, v_{g_{pi}}) \in a_{pi}$, $t \in [s_{g_{pi}}, f_{g_{pi}})$, and $a_{pi}(t) = v_{g_{pi}}$. Similarly $b_{pj}(t) = v_{h_{pj}}$. Therefore

$$c_{ij}(t) = \sum_{p: t \in T_{a_{pi}} \cap T_{b_{pj}}} v_{g_{pi}} \cdot v_{h_{pj}}$$

For binary instantaneous two-mode networks \mathbf{A} and \mathbf{B} the value $c_{ij}(t)$ of the product $\mathbf{C} = \mathbf{A} \cdot \mathbf{B}$ is equal to the number of different members of P with which both i and j have contact in the instant t .

The product of cumulative networks is cumulative itself. For binary cumulative two-mode networks \mathbf{A} and \mathbf{B} the value $c_{ij}(t)$ of the product $\mathbf{C} = \mathbf{A} \cdot \mathbf{B}$ is equal to the number of different members of P with which both i and j had contact in instants up to including the instant t .

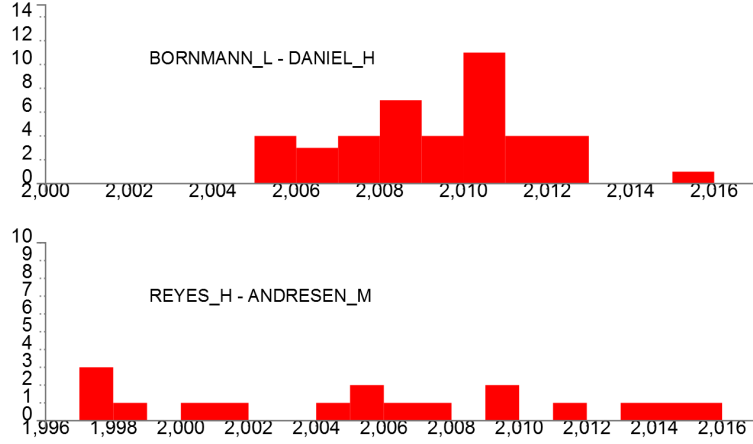


Figure 4: Coauthorship

3.1.1 Temporal co-occurrence networks

Using the multiplication of temporal affiliation networks over the combinatorial semiring we get the corresponding instantaneous and cumulative co-occurrence networks

$$\mathbf{Ci} = \mathbf{Ai}^T \cdot \mathbf{Ai} \quad \text{and} \quad \mathbf{Cc} = \mathbf{Ac}^T \cdot \mathbf{Ac}$$

The triple (s, f, v) in a temporal quantity ci_{pq} tells that in the time interval $[s, f)$ there were v events in which both p and q took part.

The triple (s, f, v) in a temporal quantity cc_{pq} tells that in the time interval $[s, f)$ there were in total v accumulated events in which both p and q took part.

The diagonal (loop) weights ci_{pp} and cc_{pp} contain the temporal quantities counting the number of events in the time intervals in which the participant p took part.

A typical example of such a network is the works authorship network \mathbf{WA} where E is the set of papers W , P is the set of authors A , and d is the publication year.

3.1.2 Example: Temporal coauthorship network

The instantaneous coauthorship network \mathbf{Coi} is obtained as

$$\mathbf{Coi} = \mathbf{Wai}^T \cdot \mathbf{Wai}$$

Bibliographic networks are usually sparse. Often also the product of sparse networks is sparse itself. Considering in computation only non zero elements it can be computed fast (Batagelj and Cerinšek, 2013). In our example, the network \mathbf{WA} has 22104 works, 62106 authors and 80021 arcs. The derived network \mathbf{Coi} has 633977 edges and was computed on a laptop in 12.7 seconds.

For the peer review data we get the largest values

$$co[\text{BORNMANN_L}, \text{DANIEL_H}] = 42,$$

$$co[\text{MOHER_D}, \text{ALTMAN_D}] = 24,$$

$$co[\text{REYES_H}, \text{ANDRESEN_M}] = 17.$$

The corresponding temporal quantities $bd = tq(\text{BORNMANN_L}, \text{DANIEL_H})$ and $ra = tq(\text{REYES_H}, \text{ANDRESEN_M})$ are

$$bd = [(2005, 2006, 4), (2006, 2007, 3), (2007, 2008, 4), (2008, 2009, 7), (2009, 2010, 4),$$

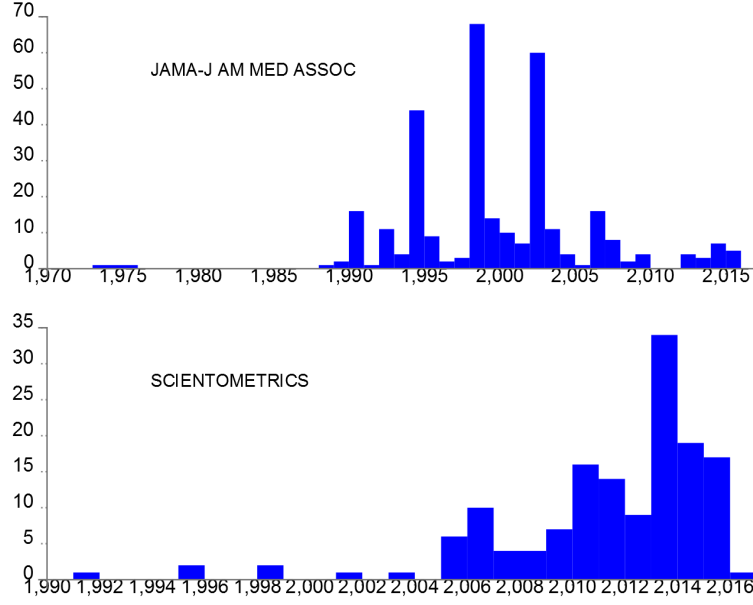


Figure 5: Selfcitations

$$\begin{aligned}
 ra = & [(2010, 2011, 11), (2011, 2013, 4), (2015, 2016, 1)] \\
 & [(1997, 1998, 3), (1998, 1999, 1), (2000, 2002, 1), (2004, 2005, 1), (2005, 2006, 2), \\
 & (2006, 2008, 1), (2009, 2010, 2), (2011, 2012, 1), (2013, 2016, 1)]
 \end{aligned}$$

Both temporal quantities are presented in Figure 4. The Python code is given in Appendix A.4.

3.1.3 Example: Temporal citations between journals

The derived network describing citations between journals is obtained as

$$JCJ = WJi^T \cdot CiteI \cdot WJc$$

Note that the third network in the product is cumulative.

The weight of the element jcj_{ij} is equal to the number of citations per year from works published in journal i to works published in journal j . In a special case when $i = j$ we get a temporal quantity describing selfcitations of journal i . In the peer review data the largest number of selfcitations are 320 in JAMA and 148 in Scientometrics. The corresponding temporal quantities $jm = jcj[JAMA, JAMA]$ and $sm = jcj[SCIENTOMETRICS, SCIENTOMETRICS]$ are:

$$\begin{aligned}
 jm = & [(1973, 1976, 1), (1988, 1989, 1), (1989, 1990, 2), (1990, 1991, 16), (1991, 1992, 1), \\
 & (1992, 1993, 11), (1993, 1994, 4), (1994, 1995, 44), (1995, 1996, 9), (1996, 1997, 2), \\
 & (1997, 1998, 3), (1998, 1999, 68), (1999, 2000, 14), (2000, 2001, 10), (2001, 2002, 7), \\
 & (2002, 2003, 60), (2003, 2004, 11), (2004, 2005, 4), (2005, 2006, 1), (2006, 2007, 16), \\
 & (2007, 2008, 8), (2008, 2009, 2), (2009, 2010, 4), (2012, 2013, 4), (2013, 2014, 3), \\
 & (2014, 2015, 7), (2015, 2016, 5)] \\
 sm = & [(1991, 1992, 1), (1995, 1996, 2), (1998, 1999, 2), (2001, 2002, 1), (2003, 2004, 1), \\
 & (2005, 2006, 6), (2006, 2007, 10), (2007, 2009, 4), (2009, 2010, 7), (2010, 2011, 16),
 \end{aligned}$$

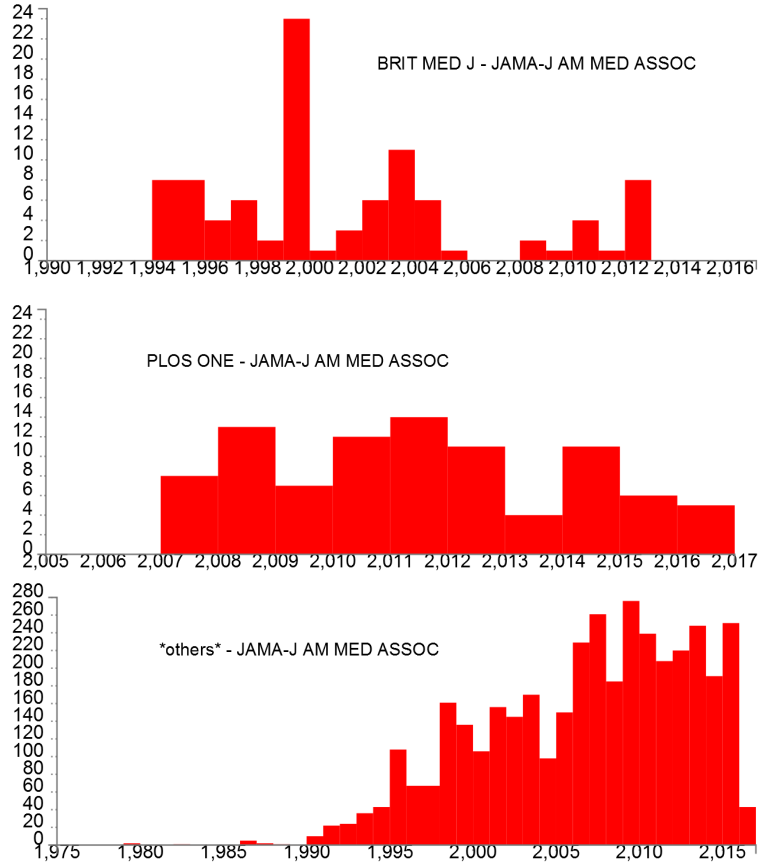


Figure 6: Citations between journals

(2011, 2012, 14), (2012, 2013, 9), (2013, 2014, 34), (2014, 2015, 19), (2015, 2016, 17),
(2016, 2017, 1)]

and are presented in Figure 5.

The largest number of citations are from journals BMJ Open (142) and Scientometrics (108) to the unknown journal *****, followed by $bj = jci[\text{BRIT MED J, JAMA-J AM MED ASSOC}]$ and $pj = jci[\text{PLOS ONE, JAMA-J AM MED ASSOC}]$ with totals 96 and 91.

$$bj = [(1994, 1996, 8), (1996, 1997, 4), (1997, 1998, 6), (1998, 1999, 2), (1999, 2000, 24), \\ (2000, 2001, 1), (2001, 2002, 3), (2002, 2003, 6), (2003, 2004, 11), (2004, 2005, 6), \\ (2005, 2006, 1), (2008, 2009, 2), (2009, 2010, 1), (2010, 2011, 4), (2011, 2012, 1), \\ (2012, 2013, 8)]$$

$$pj = [(2007, 2008, 8), (2008, 2009, 13), (2009, 2010, 7), (2010, 2011, 12), (2011, 2012, 14), \\ (2012, 2013, 11), (2013, 2014, 4), (2014, 2015, 11), (2015, 2016, 6), (2016, 2017, 5)]$$

See the top and mid part of Figure 6.

In the peer review data the journal JAMA is the most prominent. To get the temporal quantity describing citations of others to JAMA we compute $jci = iS(\mathbf{JCJ}, \text{JAMA-J AM MED ASSOC})$:

$$jci = [(1979, 1980, 2), (1982, 1983, 1), (1986, 1987, 5), (1987, 1988, 2),$$

(1988, 1989, 1), (1990, 1991, 10), (1991, 1992, 22), (1992, 1993, 24),
 (1993, 1994, 36), (1994, 1995, 43), (1995, 1996, 108), (1996, 1998, 67),
 (1998, 1999, 161), (1999, 2000, 136), (2000, 2001, 106), (2001, 2002, 156),
 (2002, 2003, 145), (2003, 2004, 170), (2004, 2005, 98), (2005, 2006, 150),
 (2006, 2007, 229), (2007, 2008, 261), (2008, 2009, 185), (2009, 2010, 276),
 (2010, 2011, 239), (2011, 2012, 208), (2012, 2013, 220), (2013, 2014, 248),
 (2014, 2015, 191), (2015, 2016, 251), (2016, 2017, 43)]

It is presented at the bottom of Figure 6. The Python code is given in Appendix A.5.

Similarly we get the temporal network describing citations between authors

$$ACA = W\mathbf{A}i^T \cdot \text{CiteI} \cdot W\mathbf{A}c$$

The weight of the element aca_{ab} is equal to the number of citations per year from works coauthored by author a to works coauthored by author b .

4 Conclusions

We presented two ways (instantaneous and cumulative) to transform bibliographic networks, using the works' publication year, into corresponding temporal networks based on temporal quantities. They are a basis for a longitudinal approach to the analysis of temporal network which is an alternative to the traditional cross-sectional approach. Introducing a time dimension can give additional insights into bibliographic networks. We also presented different methods for analyzing the obtained temporal networks and illustrated them with examples from analysis of the peer review bibliography.

We presented only some examples to show that it works. The proposed approach can be extended in some directions:

- other node and link properties;
- other derived networks combined with fractional approach;
- normalization (proportions) of temporal properties considering the changes of the “size” of network through time;
- clustering of temporal quantities to determine their types;
- temporal networks methods produce large results. Special methods for identifying and presenting (visualizing) interesting parts need to be developed.

Acknowledgments

The paper is based on presentations on 1244. Sredin seminar, IMFM, Ljubljana, April 8, 2015; PEERE meeting, Vilnius, March 7-9, 2017; and XXXVII Sunbelt workshop Beijing, China, May 30 – June 4, 2017.

This work is supported in part by the Slovenian Research Agency (research program P1-0294 and research projects J1-9187, J7-8279 and BI-US/17-18-045), project PEERE (COST Action TD1306) and by Russian Academic Excellence Project '5-100'.

References

- Baras, J.S., Theodorakopoulos, G. (2010) Path problems in networks. Morgan & Claypool, Berkeley.
- Batagelj, V. (2007) WoS2Pajek. Networks from Web of Science. Version 1.5 (2017).
<http://vladowiki.fmf.uni-lj.si/doku.php?id=pajek:wos2pajek>
- Batagelj V., Cerinšek M.(2013). On bibliographic networks. *Scientometrics*. 96 (3), 845-864
- Batagelj, V., Doreian P., V., Ferligoj, A., Kejžar N. (2014). Understanding Large Temporal Networks and Spatial Networks: Exploration, Pattern Searching, Visualization and Network Evolution.
- Batagelj, V., Ferligoj, A. Squazzoni, F. (2017) The emergence of a field: a network analysis of research on peer review. *Scientometrics*, 113: 503. <https://doi.org/10.1007/s11192-017-2522-8>
- Batagelj, V., Praprotnik, S. (2016) An algebraic approach to temporal network analysis based on temporal quantities. *Social Network Analysis and Mining*, 6(1), 1-22
- Batagelj, V. (2017) **Nets** – a Python package for network analysis. <https://github.com/bavla/Nets>
- Cerinšek, M., Batagelj, V.: Network analysis of Zentralblatt MATH data. *Scientometrics*, 102(2015)1, 977-1001.
- Clarivate Analytics (2018) **WoS**. <https://clarivate.com/products/web-of-science/databases/>
- Holme, P. (2015) Modern temporal network theory: a colloquium. *European Physical Journal B* 88, 234.

A Code in Nets

A.1 Converting Pajek net and clu files into temporal network in netsJSON

To set up an environment for computing our examples we have to put in the directory `gdir` Python files (`Nets.py`, `TQ.py`, `search.py`, `coloring.py`, `IndexMinPQ.py`) from the library `Nets`, and in the subdirectory `cdir` the files `TQchart.html`, `d3.v3.min.js` and `barData.js`. The directory `ndir` contains the network data and the directory `wdir` contains the results.

```
gdir = 'c:/path/Nets'
wdir = 'c:/path/Test/peere'
ndir = 'c:/path/WoS/peere2'
cdir = 'c:/path/Nets/chart'
import sys, os, datetime, json
sys.path = [gdir]+sys.path; os.chdir(wdir)
from TQ import *
from Nets import Network as N
net = ndir+"/WAd.net"
clu = ndir+"/Yeard.clu"
t1 = datetime.datetime.now(); print("started: ",t1.ctime(),"\n")
WAc = N.twoMode2netsJSON(clu,net,'WAcum.json',instant=False)
t2 = datetime.datetime.now()
print("\nconverted to cumulative TN: ",t2.ctime(),"\ntime used: ", t2-t1)
WAI = N.twoMode2netsJSON(clu,net,'WAIins.json',instant=True)
t3 = datetime.datetime.now()
print("\nconverted to instantaneous TN: ",t3.ctime(),"\ntime used: ", t3-t2)
cit = ndir+"/CiteD.net"
Citei = N.oneMode2netsJSON(clu,cit,'CiteIns.json',instant=True)
t4 = datetime.datetime.now()
print("\nconverted to instantaneous TN: ",t4.ctime(),"\ntime used: ", t4-t3)
ia = WAI.Index()
ic = Citei.Index()
```

A.2 Productivities of authors

```
>>> tit = 'BORNMANN_L'; b = ia[tit]
>>> pr = WAI.TQnetInSum(b)
>>> pr
[(2005, 2006, 4), (2006, 2007, 3), (2007, 2008, 4), (2008, 2009, 9),...
>>> TQ.TqSummary(pr)
(1900, 2017, 0, 14)
>>> TQmax = 15; Tmin = 1995; Tmax = 2016; w = 600; h = 150
>>> N.TQshow(pr,cdir,TQmax,Tmin,Tmax,w,h,tit,fill='red')
>>> cpr = WAc.TQnetInSum(b)
>>> cpr
[(2005, 2006, 4), (2006, 2007, 7), (2007, 2008, 11), (2008, 2009, 20),...
>>> TQmax = 65; Tmin = 1995; Tmax = 2016; w = 600; h = 250
>>> N.TQshow(cpr,cdir,TQmax,Tmin,Tmax,w,h,tit,fill='red')
>>> WAni = WAI.TQnormal()
>>> fpr = WAni.TQnetInSum(b)
>>> fpr
[(2006, 2007, 1.3333333333333333), (2007, 2008, 1.6666666666666665),...
>>> TQmax = 7; Tmin = 1995; Tmax = 2016; w = 600; h = 150
>>> N.TQshow(fpr,cdir,TQmax,Tmin,Tmax,w,h,tit,fill='red')
```

A.3 Citations between works

```
>>> tit = 'PETERS_D(1982)5:187'; c = ic[tit]
>>> ci = Citei.TQnetInSum(c)
>>> ci
[(1982, 1983, 1), (1983, 1984, 4), (1984, 1986, 3), (1986, 1987, 2), ...
>>> TQmax = 15; Tmin = 1980; Tmax = 2016; w = 600; h = 150
>>> N.TQshow(ci,cdir,TQmax,Tmin,Tmax,w,h,tit,fill='blue')
>>> tit = 'HIRSCH_J(2005)102:16569'; c = ic[tit]
>>> ci = Citei.TQnetInSum(c)
>>> ci
[(2005, 2006, 0), (2006, 2007, 3), (2007, 2008, 4), (2008, 2009, 7), ...
>>> TQmax = 25; Tmin = 2000; Tmax = 2017; w = 600; h = 250
>>> N.TQshow(ci,cdir,TQmax,Tmin,Tmax,w,h,tit,fill='blue')
```

A.4 Temporal coauthorship network

```
>>> Co = WAI.TQtwo2oneCols()
>>> Co.saveNetsJSON('CoIns.json', indent=2)
>>> Co.delLoops()
>>> C = Co.TQtopLinks(thresh=15)
>>> tit = C[0][2]+' - '+C[0][3]; bd = C[0][5]
>>> TQmax = 15; Tmin = 2000; Tmax = 2017; w = 600; h = 150
>>> N.TQshow(bd, cdir, TQmax, Tmin, Tmax, w, h, tit, fill='red')
>>> tit = C[2][2]+' - '+C[2][3]; ra = C[2][5]
>>> TQmax = 10; Tmin = 1996; Tmax = 2017; w = 600; h = 150
>>> N.TQshow(ra, cdir, TQmax, Tmin, Tmax, w, h, tit, fill='red')
>>> TQ.total(bd), TQ.total(ra)
(42, 17)
```

A.5 Citations between journals

```
>>> jrn = ndir+"/WJd.net"
>>> WJc = N.twoMode2netJSON(clu, jrn, 'WJcum.json', instant=False)
>>> WJi = N.twoMode2netJSON(clu, jrn, 'WJins.json', instant=True)
>>> JCJ = N.TQmultiply(N.TQmultiply(WJi.transpose(), Citei.one2twoMode()), WJc, True)
>>> L = JCJ.TQtopLoops(thresh=100)
>>> tit = L[0][1]; jm = L[0][3]
>>> TQmax = 70; Tmin = 1970; Tmax = 2017; w = 600; h = 200
>>> N.TQshow(jm, cdir, TQmax, Tmin, Tmax, w, h, tit, fill='blue')
>>> tit = L[1][1]; sm = L[1][3]
>>> TQmax = 35; Tmin = 1990; Tmax = 2017; w = 600; h = 200
>>> N.TQshow(sm, cdir, TQmax, Tmin, Tmax, w, h, tit, fill='blue')
>>> JCJ.delLoops()
>>> T = JCJ.TQtopLinks(thresh=70)
>>> tit = T[2][2]+' - '+T[2][3]; bj = T[2][5]
>>> TQmax = 25; Tmin = 1990; Tmax = 2017; w = 600; h = 200
>>> N.TQshow(bj, cdir, TQmax, Tmin, Tmax, w, h, tit, fill='red')
>>> tit = T[3][2]+' - '+T[3][3]; pj = T[3][5]
>>> TQmax = 25; Tmin = 2005; Tmax = 2017; w = 600; h = 200
>>> N.TQshow(pj, cdir, TQmax, Tmin, Tmax, w, h, tit, fill='red')
>>> jci = TQ.cutGE(JCJ.TQnetInSum(T[2][1]), 1e-10)
>>> TQ.TqSummary(jci)
(1979, 2017, 1, 276)
>>> TQ.total(jci)
3861
>>> tit = '*others* - '+T[2][3]
>>> TQmax = 280; Tmin = 1975; Tmax = 2017; w = 600; h = 200
>>> N.TQshow(jci, cdir, TQmax, Tmin, Tmax, w, h, tit, fill='red')
```