Cores in temporal networks / Longitudinal approach to core maintenance

Vladimir Batagelj

Institute of Mathematics, Physics and Mechanics, Jadranska 19,
1000 Ljubljana,
University of Primorska, 6000 Koper

Monika Cerinšek

Abelium d.o.o. R&D, Kajuhova 90, 1000 Ljubljana

February 18, 2019

Abstract

TODO: zaenkrat enako kot na konferenci

Among many formalizations of the notion of dense subnetworks of a given network (clique, s-plexes, LS sets, lambda sets, cores, etc.) only for cores exists an efficient algorithm that can compute the result in an acceptable time also for large networks.

In a graph $G = (V, E)$ a subgraph $H = (W, E(W))$ induced by the subset of nodes W is a k-core or a core of order k iff for each node v in W its internal degree $\deg_H(v)$ is greater or equal to k, and H is a maximal subgraph with this property.

Various generalizations of ordinary cores according to types of networks and measures of node importance were proposed. In our contribution we extend the notion of cores to temporal networks and present the corresponding algorithms. For illustration we present results of determining cores in selected artificial and real-life networks.

Keywords:
Mathematics Subject Classification:

1 Introduction

With data emerging from all aspects of our life, data analysis became our constant companion. Usage of applications is used in marketing analysis to better personalize advertisements, smart watches analyse our sleep cycle and activities to offer us recommendations for healthier life, brokers analyse stock markets to inform us about stock and cryptocurrency evaluation and value movement, etc. All this data is not constant but changes through time and time is important part of analysis. On other hand, network analysis is getting increasingly recognized not only in academic but also business sphere. Temporal network analysis enables us to better understand how relations among entities develop through time.

In this article we present new approach to the core maintenance problem in temporal network and explains it through artificial and real-life examples. Core maintenance problem is generalization of core decomposition problem in static network. With our method we search for strongly connected subgroups in a network. These groups can change through time, so we consider time to be an important part of our method.

In the next section we review the works that were the groundwork of our method and the works that contain ideas about different approach to the core maintenance problem. In Sections 3 we present definitions of basis concepts that lead to our method and concepts that we use in our method (temporal network, temporal core). In Section 4 we present the algorithm for our method. In Section 5 some results obtained in usage of our method of artificial and real-life data are presented.

2 Related work

The notion of k-core is being used since 1983 when it was introduced by Seidman[1]. Let $\mathcal{G} = (\mathcal{V}, \mathcal{L})$ be a graph with set of nodes $\mathcal{V}$ and set of links $\mathcal{L}$. Let k be fixed integer and let $deg(v)$ be a degree of node $v \in \mathcal{V}$. A subgraph $\mathcal{H}_k = (\mathcal{C}_k, \mathcal{L}|\mathcal{C}_k)$ induced by the subset of nodes $\mathcal{C}_k \subseteq \mathcal{V}$ is called a k-core iff $deg_{\mathcal{H}_k}(v) \geq k$, for all $v \in \mathcal{C}_k$, and $\mathcal{H}_k$ is maximal such subgraph. A degree of node is node property and if we replace it with another node property, we get the notion of generalized core which was introduced in [2]. Some other node properties are sum of incident link weights, fraction of neghbours within subset $\mathcal{C}$, maximal weight of incident links, etc. The node properties and restrictions on them are described in more detail in Section 3.

The other possible generalization of k-cores is onto two-mode networks. The notion of (p, q)-core was introduced in [3] and the notion of generalized two-mode cores was introduced in[4].

In this paper we introduce new approach to generalization of core onto temporal networks. Few approaches to core maintenance problem are already available:

- Determine nodes that need recalculation of core number when network changes only with inserting or removing links [5]
- Maintaining core number just for larger values of k [6]
- Parallel computing with added notion of superior edge set [7] [8]

Our method differs because we use longitudinal approach to be more efficient and calculate core numbers for all nodes.

[1] seidman
[2] batcore
[3] pqcore
[4] gen2modecore
[5] liyu, core1, bladyg
[6] aksu
[7] parallel
[8] core1

Core maintenance problem was also used for community identification and maintenance in [9].

3 Definitions

TODO do Generalized core

Core of order k

Network: $\mathcal{N} = (\mathcal{V}, \mathcal{L}, \mathcal{P}, \mathcal{W})$; $n = |\mathcal{V}|, m = |\mathcal{L}|$

k-core (Seidman 1983): A subgraph $\mathcal{H} = (\mathcal{C}, \mathcal{L}(\mathcal{C}))$ induced by the set of nodes $\mathcal{C}$ is a k-core or a core of order k iff $\forall v \in \mathcal{C} : deg_{\mathcal{H}}(v) \geq k$ and $\mathcal{H}$ is the maximum subgraph with this property.

The core of maximum order – main core.

The core number of node v is the highest order of a core that contains this node.

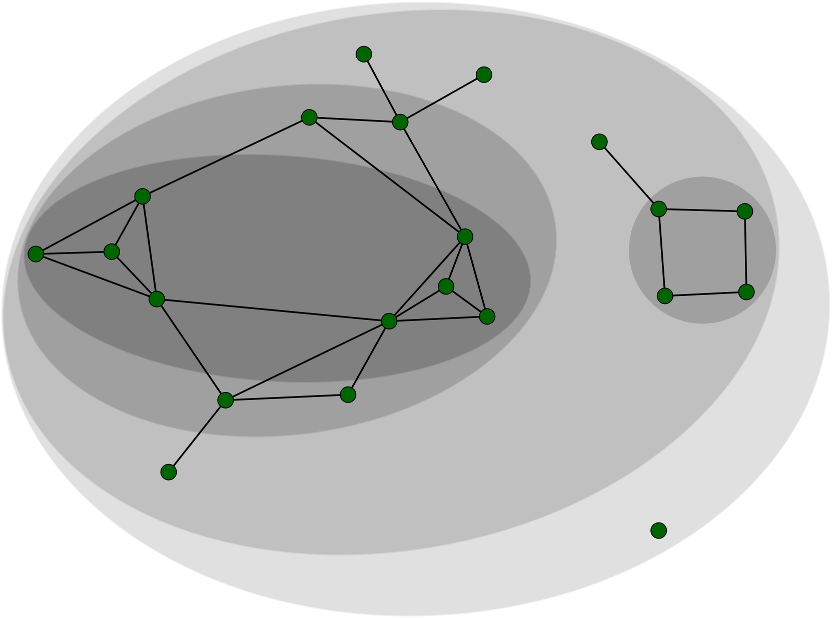

Figure 1: All k-cores of a network.

3.1 Generalized core

Node property function $p(v, \mathcal{C})$ is defined for all $v \in \mathcal{V}$ in a network $\mathcal{N} = (\mathcal{V}, \mathcal{L}, \mathcal{P}, \mathcal{W})$ and for $\mathcal{C}$ being a subset in $\mathcal{V}$. Function p is defined as $p : L \longrightarrow \mathbb{R}^+$.

Let us mark $N(v)$ as a neighbourhood of a node v in network $\mathcal{N}$ and $N(v, \mathcal{C})$ as a neighbourhood of a node v within the subset $\mathcal{C}$. We define node property function p such that two properties hold for it:

[9] community

- $p(v, \mathcal{C})$ is local: $p(v, \mathcal{C}) = p(v, N(v, \mathcal{C}))\ \forall v \in \mathcal{V}$.
- $p(v, \mathcal{C})$ is monotonic: $\mathcal{C}_1 \subset \mathcal{C}_2 \Rightarrow \forall v \in \mathcal{V} : p(v, \mathcal{C}_1) \leq p(v, \mathcal{C}_2)$..

Some node properties were proposed in [10]. Examples of node property functions are:

1. $p_1(v, \mathcal{C}) = \deg_{\mathcal{C}}(v)$: node degree within $\mathcal{C}$
2. $p_2(v, \mathcal{C}) = \text{indeg}_{\mathcal{C}}(v) + \text{outdeg}_{\mathcal{C}}(v)$: if lines are directed it holds $p_2 = p_1$
3. **$\mathbf{p_3}(\mathbf{v}, \mathcal{C}) = \sum_{\mathbf{u} \in \mathbf{N}(\mathbf{v}, \mathcal{C})} \mathbf{w}(\mathbf{v}, \mathbf{u})$ for $\mathbf{w} : \mathbf{L} \to \mathbb{R}_\mathbf{0}^+$: sum of weights of incident lines within $\mathcal{C}$**
4. $p_4(v, \mathcal{C}) = \max_{u \in N(v, \mathcal{C})} w(v, u)$ for $w : L \to \mathbb{R}$: maximal weight of incident lines within $\mathcal{C}$
5. $p_5(v, \mathcal{C}) = \frac{\deg_{\mathcal{C}}(v)}{\deg(v)}$ if $\deg(v) > 0$ else $f_5(v, \mathcal{C}) = 0$: fraction of neighbors within $\mathcal{C}$.
6. $p_6(v, \mathcal{C}) = \frac{\sum_{u \in N(v, \mathcal{C})} w(v,u)}{\sum_{u \in N(v)} w(v,u)}$ for $w : L \to \mathbb{R}_0^+$: fraction of sum of weights of incident lines within $\mathcal{C}$.

With node property functions as described above authors in [11] defined a notion of generalized cores. The subgraph $\mathcal{H} = (\mathcal{C}, \mathcal{L}(\mathcal{C}))$ induced by the set $C \subseteq \mathcal{V}$ is a p-core at level $t \in \mathbb{R}$ iff $\forall v \in \mathcal{C} : t \leq f(v, \mathcal{C})$ and $\mathcal{C}$ is maximal such set.

3.2 Temporal network and temporal core

Temporal network $\mathcal{N}_\mathcal{T} = (\mathcal{V}, \mathcal{L}, \mathcal{T}, \mathcal{P}, \mathcal{W})$ is obtained by attaching the time $\mathcal{T}$ to an ordinary network, where $\mathcal{T}$ is a set of time points: $t \in \mathcal{T}$ which are usually integers or reals. Nodes and links in temporal networks have temporal quantities assigned. Temporal quantity or TQ is a list of triples (s, f, v) and each triple determine the time interval in which node or link is active or present and its value in this time interval. Value s is inclusive start time point, value f is exclusive finish time point, and v is a node or link value.

An example of temporal network is shown in Fig. 2. All nodes in this network are present through whole time interval. Some links are directed – direction is presented with an arrow. All links have value 1 when they are present. Few links are not present through the whole time interval. For example, link from node 2 to node 4 is present from time point 3 to time point 9.

In temporal network we assign temporal quantities to nodes and links. We use the notion of $T(v)$ being the activity set of time points for the node v, and $T(l)$ being the activity set of time points for the link l. Consistency condition must be fulfilled for activity sets: If a link $l(u, v)$ is active at the time point t then its end-nodes u and v should be active at the time t :

$$T(l(u, v)) \subseteq T(u) \cap T(v).$$

[10] citiraj batcores

[11] citiraj batcores

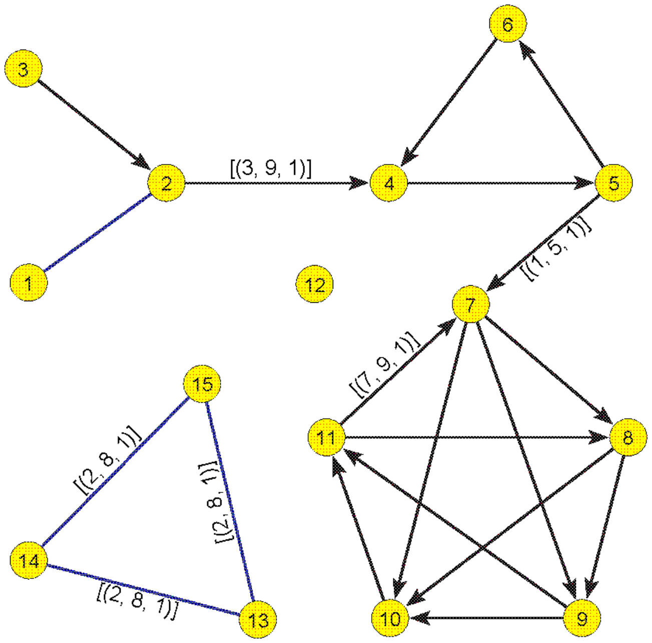

Figure 2: An example of a temporal network.

[12]

Temporal quantity a with the activity set $T_a \subseteq \mathcal{T}$ describes the changes of properties of nodes and links:

$$a = \begin{array}{ll} a'(t) & t \in T_a; \\ undefined & t \in \mathcal{T} \setminus T_a. \end{array}$$

Temporal quantities allow longitudinal approach instead of time slices. An example of this approach is shown in Fig. 3. Temporal quantites a and b are shown in upper diagrams. TQ a has contant value from time point 1 to time point 4, at time point 5 is equal to 0, etc. In lower diagrams are shown sum $a + b$ and multiplication $a * b$ of these temporal quantities. These operations are executed by time points – values at a time point are summed or multiplied to get the result.

Both k-core and generalized core can be expanded onto temporal network. There are two main approaches to determining temporal cores – longitudinal and with time slices. In both approaches the researcher must consider the core maintenance problem: the problem of maintaining core numbers for a temporal network. We selected longitudinal approach and based our approach on algorithm for k-core decomposition.

4 The algorithm

The algorithm for temporal core decomposition is based on algorithm for k-core decomposition that is presented in Alg. 1. The input for this algorithm is a

[12] cititraj clanek Praprotnik, Batagelj

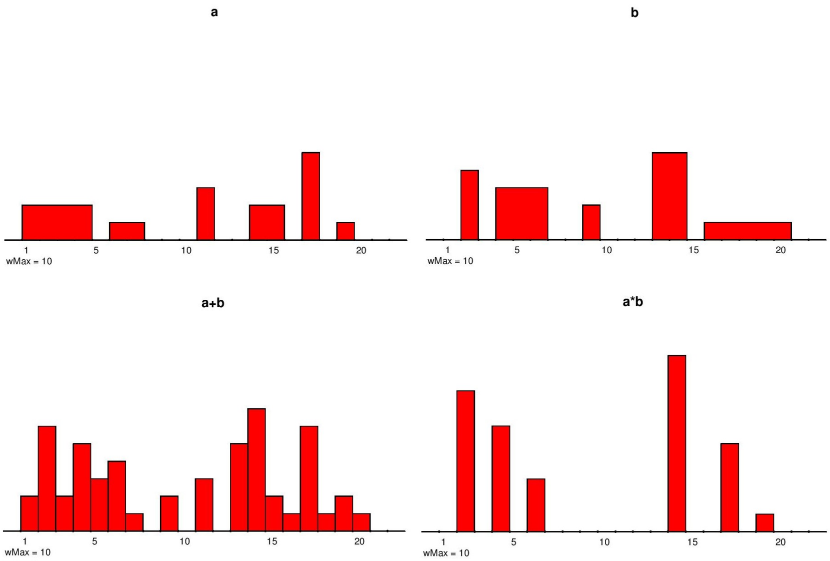

Figure 3: Summing and multiplying functions.

network. Before main loop we copy the set of nodes into set C and set a counter for core number k to 1 (lines 2 and 3 in Alg. 1). The main loop is repeated until the set C contains any node. In each step we check if there exists a node in C such that its degree is less than k. If so, we reduce degrees of its neighbours for 1, remove the node from C and set the core number of this node to $k - 1$. When no such node exists, we increase the counter k for 1. The result is list of core numbers for all nodes of the network.

Algorithm 1: Core decomposition algorithm.

```
1   CoreDecomposition($\mathcal{N}$):
2   C = V
3   k = 1
4   while C $\neq \emptyset$:
5           while $\exists$ u $\in$ C $\ni$: deg(u) < k:
6                   for v $\in$ N(u, C):
7                           deg(v) = deg(v) - 1
8                   C = C\v
9                   core(u) = k - 1
10          k = k + 1
11  return core
```

We expanded Alg. 1 onto temporal quantities and the new algorithm is shown in Alg. 2. Input for this algorithm is temporal network. Before main loop we copy node temporal quantities into new set D, define set of core numbers for node in shape of temporal quantities, removed temporal quantities from D that have a degree value equal to 0, and defined the dictionary $Dmin$ which consists of keys equal to nodes and values equal to minimum degree of a node. This is shown in lines 2-5 in Alg. 2.

The main loop (lines 6-29 in Alg. 2) is repeated until the set D is not empty.

At first we find a pair $(dmin, u)$ from *Dmin* where *dmin* is minimum of all values in *Dmin*. As *core* we define set of temporal quantities from $D[u]$ with degree equal to *dmin*. We add these temporal quantities to the set of core numbers (line 11 in Alg. 2) as their degree value represent core numbers for node u in given time intervals. We rewrite *core* into *change* with degree values equal to -1. This we need to save only those temporal quantities in $D[u]$ that have a degree larger than *dmin* where we reduce their degrees for 1 in those that are also in *change*.

Now we can check all neighbours of node u – we loop through all links of u – lines 14-25 in Alg. 2. If the other end-node of a link is not in D, we skip it and check another link. Otherwise we define *changeLink* as intersection of temporal quantities of a link and *change* : time intervals are intersections of time intervals, value is equal to -1. Link *changeLink* is actually active the same time as u with degree value equal to *dmin*. If this link is never active, we skip it and check another link neighbouring u. Now we define $diff$ as temporal quantities from $D[v]$ with degree value more than 0, where we reduce their degrees for 1 in those that are also in *changeLink*. Now we rewrite $D[v]$ as $diff$ with degree values equal to $\max(deg, dmin)$. If $D[v]$ is empty, we delete node v from D and *Dmin*. Otherwise we set a degree value for $Dmin[v]$ to be equal to minimum degree value in $D[v]$.

After checking all neighbouring links of node u we check if there are any non-empty temporal quantities in $D[u]$ – lines 26-29 in Alg. 2. If there are none, we delete node u from D and *Dmin*. Otherwise we set a degree value for u in *Dmin* to be minimum degree value in temporal quantities of $D[u]$.

Algorithm 2: Simple algorithm for cores in temporal networks.

```
1  TemporalCores($\mathcal{N}$):
2  D = {u: [tq (start, finish, deg)]}
3  CoreHierarchy = {u: [tq with deg = 0]}
4  D = (D.filter(deg > 0)).remove(empty tq)
5  Dmin = {u: min deg}
6  while D not empty:
7          (dmin, u) = (deg, u)
8                              ∋: (u, deg) ∈ Dmin ∧ deg is min deg
9          core = [tq from D[u]
10                             ∋: deg[u] from tq is equal to dmin]
11         CoreHierarchy[u].add(core)
12         change = core.set(deg = −1)
13         D[u] = D[u].add(change).cutAt(dmin)
14         for l in $\mathcal{N}$.star(u):
15                 v = other end−node of l
16                 if not v in D: continue
17                 changeLink = l.intersection(change).
18                                 set(deg = −1)
19                 if changeLink empty: continue
20                 diff = D[v].add(changeLink).cutAt(0)
21                 D[v] = diff.set(max(currentValue, dmin))
22                 if D[v] is empty:
23                         delete D[v], Dmin[v]
24                 else:
```

```
                                Dmin[v] = tq ∈ D[v] with min deg
          if D[u] empty:
                    delete D[u], Dmin[u]
          else:
                    Dmin[u] = tq ∈ D[u] with min deg
return CoreHierarchy
```

5 Results

We tested our method on artificial examples and examples from real-life with different context. Firstly we will look at artificial example to better understand what temporal network is and how do we determine temporal cores.

5.1 Artificial example

The network is shown in Fig. 4. It consists of 15 nodes that are always present. Most of the links are directed, some are undirected. Most of the links are present all the time – for them there are no temporal quantities written in figure. Links $(2, 4), (5, 7), (11, 7), (13, 14), (13, 15), (14, 15)$ are present in defines time intervals. All links have value equal to 1 when they are present. The degrees of nodes are listed in the left side of Tab. 1. For example, node 13 has a degree equal to 0 in time interval $[1, 2)$ because both neighbouring links are not present in this time interval. In time interval $[2, 8)$ are both links present, so the node has a degree equal to 2. After that its degree is again equal to 0, because no neighbouring link is present.

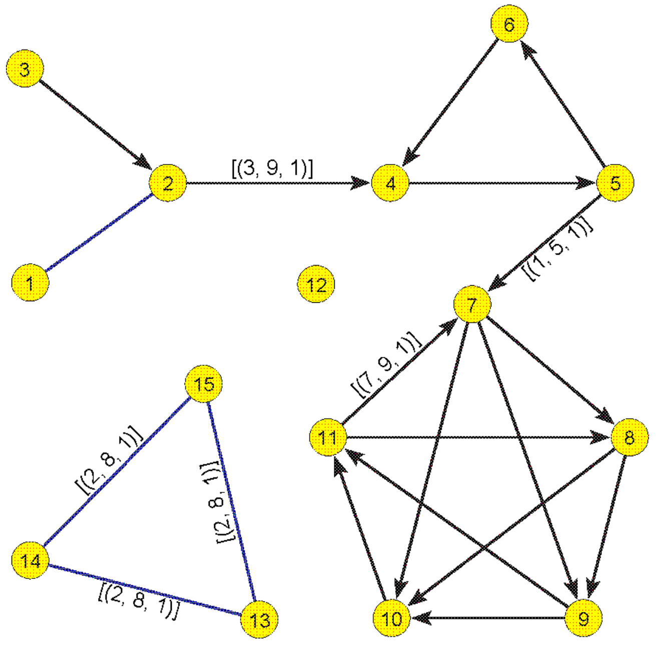

Figure 4: Visualization of first artificial example of a temporal network.

Table 1: Degrees and core numbers for first artificial network's nodes.

Node	**Degree**	**Core number**
1	(1, 9, 1)	(1, 9, 1)
2	(1, 3, 2), (3, 9, 3)	(1, 9, 1)
3	(1, 9, 1)	(3, 9, 1)
4	(1, 3, 2), (3, 9, 3)	(1, 9, 2)
5	(1, 5, 3), (5, 9, 2)	(1, 9, 2)
6	(1, 9, 2)	(1, 9, 2)
7	(1, 5, 4), (5, 7, 3), (7, 9, 4)	(1, 7, 3), (7, 9, 4)
8	(1, 9, 4)	(1, 7, 3), (7, 9, 4)
9	(1, 9, 4)	(1, 7, 3), (7, 9, 4)
10	(1, 9, 4)	(1, 7, 3), (7, 9, 4)
11	(1, 7, 3), (7, 9, 4)	(1, 7, 3), (7, 9, 4)
12	(1, 9, 0)	(1, 9, 0)
13	(1, 2, 0), (2, 8, 2), (8, 9, 0)	(1, 2, 0), (2, 8, 2), (8, 9, 0)
14	(1, 2, 0), (2, 8, 2), (8, 9, 0)	(1, 2, 0), (2, 8, 2), (8, 9, 0)
15	(1, 2, 0), (2, 8, 2), (8, 9, 0)	(1, 2, 0), (2, 8, 2), (8, 9, 0)

In the algorithm for temporal core decomposition we need to define sets D, $CoreHierarchy$, and $Dmin$. For this examples these are:

- $D = \{1 : [(1, 9, 1)], 2 : [(1, 3, 2), (3, 9, 3)], ..., 15 : [(1, 2, 0), (2, 8, 2), (8, 9, 0)]\}$
- $CoreHierarchy = \{12 : [(1, 9, 0)], 13 : [(1, 2, 0), (8, 9, 0)], 14 : [(1, 2, 0), (8, 9, 0)], 15 : [(1, 2, 0), (8, 9, 0)]\}$
- Remove temporal quantities with value 0 from D: $D = \{1 : [(1, 9, 1)], 2 : [(1, 3, 2), (3, 9, 3)], ..., 15 : [(2, 8, 2)]\}$
- $Dmin = \{1 : 1, 2 : 2, 3 : 1, 4 : 2, 5 : 2, 6 : 2, 7 : 3, 8 : 4, 9 : 4, 10 : 4, 11 : 3, 12 : 0, 13 : 0, 14 : 0, 15 : 0\}$

We determine $dmin$ from $Dmin$ to be 0, which is first core number we calculate cores with. The result of the algorithm is shown on the right in Tab. 1. Core number for node 1 is equal 1 all the time. Core number for node 15 is equal to 0 in time intervals $[1, 2)$ and $[8, 9)$, and is equal to 2 in time interval $[2, 8)$.

We defined another artificial temporal network (Fig. 5) with the same underlying graph structure as in previous network (Fig. 4). The presence and values of links are different which is shown in Fig. 5 and in values of nodes in Tab. 2. We selected sum of neighbouring link values as node values for this example so we can calculate p_S-core numbers for nodes. The result of the algorithm is shown on the right side of Tab. 2.

5.2 Reuters terror news network

Data for this example was obtained from the CRA (Centering Resonance Analysis) networks produced by Steve Corman and Kevin Dooley at Arizona State University. Data is available at

http://vlado.fmf.uni-lj.si/pub/networks/data/CRA/terror.htm.

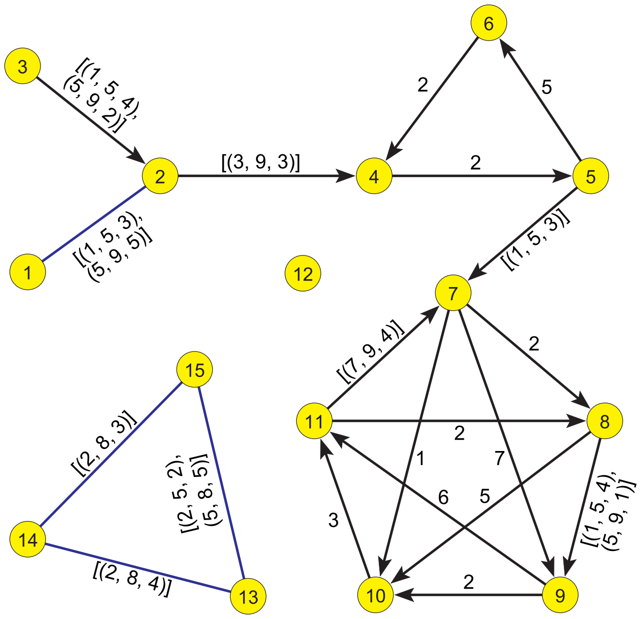

Figure 5: Visualization of second artificial example of a temporal network.

The base for this data are all stories released during 66 consecutive days by the news agency Reuters concerning the September 11 attack on the U.S. This time interval starts at 9:00 AM EST 11th of September 2001.

Nodes of a network are important words (terms). Two terms are connected if they appear in the same text unit (sentence). The link value is equal to the frequency of such appearances. There are $n = 13332$ nodes and $m = 243447$ (undirected) links in a network. 50859 of links have value more than 1 and there are no loops.

We took 50 most active nodes to induce a subnetwork to work with. A table representation of a such network can become quite fuzzy as can be seen in Tab. 3. The same goes for core numbers (Tab. 4). We are interested in largest core numbers limited our result to only those nodes with core number at least equal to 3. The bounded result is shown in Tab. 5 which is more understandable than the table of all core numbers.

Table 3: Degrees for nodes of induced subnetwork.

Node	**Degree**
1	(1, 2, 5), (2, 3, 6), (3, 4, 3), (4, 5, 5), (5, 6, 4), (6, 8, 3), (8, 10, 5), (10, 11, 3), (11, 13, 2), (13, 16, 3), (16, 17, 4), (17, 18, 5), (18, 19, 3), (19, 21, 1), (21, 22, 2), (22, 23, 1), (23, 24, 4), (24, 25, 1), (25, 29, 3), (29, 31, 2), (31, 33, 3), (33, 34, 1), (34, 36, 3), (36, 37, 2), (37, 39, 3), (39, 40, 4), (40, 41, 2), (41, 42, 0), (42, 43, 3), (43, 44, 2), (44, 45, 3), (45, 46, 1), (46, 47, 2), (47, 48, 3), (48, 49, 0), (49, 50, 4), (50, 51, 1), (51, 52, 2), (52, 53, 1), (53, 54, 0), (54, 58, 2), (58, 59, 3), (59, 60, 2), (60, 61, 4), (61, 62, 0), (62, 64, 2), (64, 65, 1), (65, 67, 2)
2	(1, 2, 27), (2, 3, 29), ..., (63, 64, 2), (64, 65, 0), (66, 67, 0)

Table 3: Degrees for nodes of induced subnetwork.

Node	**Degree**
...	
50	(1, 2, 3), (2, 3, 2), (3, 5, 1), (5, 8, 0), (8, 10, 1), (10, 11, 2), (11, 12, 1), (12, 15, 0), (15, 16, 3), (16, 17, 1), (17, 19, 0), (19, 20, 1), (20, 21, 2), (21, 22, 0), (22, 24, 1), (24, 26, 0), (26, 27, 2), (27, 28, 0), (28, 29, 1), (29, 31, 0), (31, 32, 1), (32, 33, 0), (33, 35, 1), (35, 37, 0), (37, 38, 1), (38, 42, 0), (43, 44, 2), (44, 49, 0), (49, 50, 2), (51, 57, 0), (58, 61, 0), (61, 62, 1), (62, 67, 0)

Table 2: Sums of neighbouring link values and p$_S$-core numbers for second artificial network's nodes.

Node	**Sum of values**	**p$_S$-core number**
1	(1, 5, 3), (5, 9, 5)	(1, 5, 3), (5, 9, 5)
2	(1, 3, 7), (3, 9, 10)	(1, 5, 4), (5, 9, 5)
3	(1, 5, 4), (5, 9, 2)	(1, 5, 4), (5, 9, 2)
4	(1, 3, 4), (3, 9, 7)	(1, 5, 4), (5, 9, 5)
5	(1, 5, 10), (5, 9, 7)	(1, 9, 5)
6	(1, 9, 7)	(1, 9, 5)
7	(1, 5, 13), (5, 7, 10), (7, 9, 14)	(1, 9, 10)
8	(1, 5, 13), (5, 9, 10)	(1, 9, 10)
9	(1, 5, 19), (5, 9, 16)	(1, 9, 10)
10	(1, 9, 11)	(1, 9, 10)
11	(1, 7, 11), (7, 9, 15)	(1, 9, 10)
12	(1, 9, 0)	(1, 9, 0)
13	(1, 2, 0), (2, 5, 6), (5, 8, 9), (8, 9, 0)	(1, 2, 0), (2, 5, 5), (5, 8, 7), (8, 9, 0)
14	(1, 2, 0), (2, 8, 7), (8, 9, 0)	(1, 2, 0), (2, 5, 5), (5, 8, 7), (8, 9, 0)
15	(1, 2, 0), (2, 5, 5), (5, 8, 8), (8, 9, 0)	(1, 2, 0), (2, 5, 5), (5, 8, 7), (8, 9, 0)

Table 4: Core numbers for nodes of induced subnetwork.

Node	**Core number**
1	(1, 2, 4), (2, 3, 5), (3, 5, 3), (5, 6, 4), (6, 8, 3), (8, 10, 4), (10, 11, 3), (11, 14, 2), (14, 18, 3), (18, 19, 2), (19, 21, 1), (21, 22, 2), (22, 23, 1), (23, 24, 3), (24, 25, 1), (25, 28, 2), (28, 29, 3), (29, 33, 2), (33, 34, 1), (34, 38, 2), (38, 39, 3), (39, 41, 2), (41, 42, 0), (42, 45, 2), (45, 46, 1), (46, 47, 2), (47, 48, 3), (48, 49, 0), (49, 50, 3), (50, 51, 1), (51, 52, 2), (52, 53, 1), (53, 54, 0), (54, 57, 2), (57, 58, 1), (58, 59, 2), (59, 60, 1), (60, 61, 2), (61, 62, 0), (62, 64, 2), (64, 65, 1), (65, 67, 2)
2	(1, 3, 5), (3, 6, 4), (6, 7, 5), ..., (63, 64, 1), (64, 65, 0), (66, 67, 0)
...	
50	(1, 3, 2), (3, 5, 1), (5, 8, 0), (8, 10, 1), (10, 11, 2), (11, 12, 1), (12, 15, 0), (15, 16, 3), (16, 17, 1), (17, 19, 0), (19, 20, 1), (20, 21, 2), (21, 22, 0), (22, 24, 1), (24, 26, 0), (26, 27, 1), (27, 28, 0), (28, 29, 1), (29, 31, 0), (31, 32, 1), (32, 33, 0), (33, 35, 1), (35, 37, 0), (37, 38, 1), (38, 42, 0), (43, 44, 1), (44, 49, 0), (49, 50, 2), (51, 57, 0), (58, 61, 0), (61, 62, 1), (62, 67, 0)

When we check the temporal component of the result, we found out that temporal cores of order at least 3 appeared in first 11 days and on day 30. This is consistent with the idea that the news dies out with passing time.

Table 5: Nodes with core numbers at least equal to 3.

Node		**Core number** (≥ 3)
1	united_states	(1, 2, 4), (2, 3, 5), (5, 6, 4), (8, 10, 4)
2	attack	(1, 3, 5), (3, 6, 4), (6, 7, 5), (7, 10, 4), (11, 12, 4), (30, 31, 4)
4	people	(1, 3, 5), (3, 6, 4), (6, 7, 5), (7, 8, 4)
5	afghanistan	(1, 3, 4), (5, 6, 4), (6, 7, 5), (8, 10, 4), (30, 31, 4)
6	bin_laden	(1, 4, 4), (5, 6, 4), (6, 7, 5), (7, 10, 4), (11, 12, 4)
7	new_york	(1, 3, 5), (3, 6, 4), (6, 7, 5), (30, 31, 4)
8	pres_bush	(1, 3, 5), (3, 6, 4), (6, 7, 5), (7, 10, 4), (11, 12, 4)
9	washington	(1, 3, 5), (3, 6, 4), (6, 7, 5), (7, 10, 4), (11, 12, 4)
10	official	(1, 3, 5), (3, 4, 4), (5, 6, 4), (6, 7, 5)
12	military	(1, 2, 4), (5, 6, 4), (30, 31, 4)
13	plane	(1, 3, 5), (3, 7, 4)
14	world_trade_ctr	(1, 3, 5), (3, 6, 4), (6, 7, 5), (30, 31, 4)
15	security	(1, 2, 4), (2, 3, 5), (5, 6, 4)

Table 5: Nodes with core numbers at least equal to 3.

Node		**Core number** (≥ 3)
26	terrorism	(2, 3, 4)
27	day	(2, 3, 4), (5, 6, 4)
28	week	(5, 6, 4), (6, 7, 5), (8, 10, 4), (11, 12, 4)
29	worker	(1, 2, 4), (2, 3, 5)
30	office	(1, 3, 4)
31	group	(2, 3, 4), (6, 7, 4)
32	air	(2, 3, 4), (5, 6, 4)
34	time	(1, 3, 5), (3, 4, 4), (5, 6, 4), (7, 8, 4)
35	hijack	(2, 3, 4)
36	strike	(2, 3, 4), (5, 6, 4), (6, 7, 5), (30, 31, 4)
38	flight	(2, 3, 4)
39	tell	(2, 3, 4)
40	terrorist	(1, 3, 4), (6, 7, 4)
41	airport	(2, 3, 4)
42	pakistan	(2, 3, 4), (5, 7, 4)
43	tower	(1, 3, 5), (3, 4, 4), (6, 7, 5)
45	new	(2, 3, 4)
47	wednesday	(2, 3, 5), (3, 4, 4), (8, 10, 4)
48	nation	(1, 3, 4), (5, 6, 4)
49	police	(2, 4, 4), (5, 6, 4)

We calculated also p$_S$-core numbers and the the result bounded to values at least equal to 20 sre listed in Tab. 6. The list of nodes is almost the same as in Tab. 5, but time span of these cores is wider, not limited mostly to first 11 days. The max p$_S$-core numbers through time interval is shown in Fig. 6. It drops rapidly after six days which is also consistent with the idea of news getting old with passing time.

Table 6: Nodes with p_S-core numbers at least equal to 20.

Node		**p$_S$-core number** (≥ 20)
1	united_states	(1, 2, 86), (2, 3, 71), (3, 4, 34), (4, 5, 29), (5, 6, 50), (6, 7, 47), (7, 8, 22), (15, 16, 23), (18, 19, 23), (27, 28, 23)
2	attack	(1, 3, 86), (3, 4, 44), (4, 5, 36), (5, 6, 66), (6, 7, 47), (7, 8, 22), (8, 9, 21), (9, 10, 24), (10, 11, 22), (11, 12, 24), (15, 16, 23), (18, 19, 20), (27, 28, 23)
3	taliban	(2, 3, 28), (6, 7, 20), (15, 16, 23), (27, 28, 23)
4	people	(1, 2, 48), (2, 3, 52), (3, 4, 28), (4, 5, 32), (5, 6, 29), (6, 7, 34), (18, 19, 20)
5	afghanistan	(1, 2, 22), (2, 3, 28), (5, 6, 29), (6, 7, 21), (15, 16, 23), (27, 28, 23)
6	bin_laden	(1, 2, 22), (2, 3, 28), (3, 4, 20), (5, 6, 29), (6, 7, 21), (18, 19, 20)
7	new_york	(1, 2, 101), (2, 3, 86), (3, 4, 42), (4, 5, 35), (5, 6, 66), (6, 7, 47), (8, 9, 21), (9, 10, 24), (10, 11, 22), (11, 12, 24), (15, 16, 23), (18, 19, 20)

Table 6: Nodes with p_S-core numbers at least equal to 20.

Node		**p$_S$-core number** (≥ 20)
8	pres_bush	(1, 2, 48), (2, 3, 44), (5, 6, 29), (6, 7, 21)
9	washington	(1, 2, 80), (2, 3, 61), (3, 4, 27), (4, 5, 28), (5, 6, 66), (6, 7, 47), (8, 9, 21), (9, 10, 24), (10, 11, 22), (11, 12, 24), (15, 16, 23), (18, 19, 20)
10	official	(1, 2, 40), (2, 3, 54), (3, 4, 34), (5, 6, 29), (6, 7, 36), (18, 19, 23)
12	military	(1, 2, 25), (2, 3, 42), (5, 6, 26), (15, 16, 23), (18, 19, 23), (27, 28, 23)
13	plane	(1, 3, 86), (3, 4, 44), (4, 5, 32), (5, 6, 50), (6, 7, 34)
14	world_trade_c	(1, 2, 101), (2, 3, 86), (3, 4, 44), (4, 5, 35), (5, 6, 66), (6, 7, 47), (8, 9, 20), (9, 10, 21), (10, 11, 22), (11, 12, 24), (15, 16, 23), (18, 19, 20)
15	security	(1, 2, 25), (2, 3, 30), (5, 6, 24)
16	american	(1, 2, 48), (2, 3, 30), (5, 7, 20)
17	country	(1, 2, 24), (2, 3, 31), (5, 6, 26), (18, 19, 20)
18	city	(1, 2, 60), (2, 3, 52), (3, 4, 22)
19	war	(2, 3, 34), (5, 6, 29)
20	tuesday	(1, 3, 86), (3, 4, 44), (4, 5, 36), (5, 6, 66), (6, 7, 47)
21	pentagon	(1, 3, 86), (3, 4, 44), (4, 5, 32), (5, 6, 66), (6, 7, 47), (8, 9, 20), (9, 10, 21), (10, 11, 22), (11, 12, 24), (15, 16, 23), (18, 19, 20)
22	force	(5, 6, 26)
23	government	(1, 2, 28), (2, 3, 36)
24	leader	(1, 2, 22)
25	world	(1, 2, 34), (2, 3, 44), (18, 19, 20)
26	terrorism	(5, 6, 20)
27	day	(1, 2, 21), (2, 3, 36), (5, 6, 20)
28	week	(5, 6, 35), (6, 7, 27), (7, 8, 22), (8, 9, 21), (9, 10, 24), (10, 11, 22), (11, 12, 24)
29	worker	(1, 2, 24)
30	office	(1, 2, 34), (2, 3, 20)
31	group	(2, 3, 26)
32	air	(2, 3, 34), (5, 6, 29), (27, 28, 23)
34	time	(2, 3, 36)
35	hijack	(1, 2, 67), (2, 3, 86), (3, 4, 44), (4, 5, 28), (5, 6, 50), (6, 7, 34)
36	strike	(2, 3, 29), (5, 6, 29), (18, 19, 22), (27, 28, 23)
38	flight	(1, 2, 25), (2, 3, 52), (4, 5, 20)
40	terrorist	(1, 2, 40), (2, 3, 29)
41	airport	(1, 2, 25), (2, 3, 44), (4, 5, 25), (5, 6, 24)
42	pakistan	(5, 6, 29)
43	tower	(1, 2, 101), (2, 3, 72), (3, 4, 41), (4, 5, 32), (5, 6, 38), (6, 7, 32)
44	bomb	(1, 2, 23)
45	new	(2, 3, 30)
46	buildng	(1, 2, 34), (2, 3, 44)
47	wednesday	(2, 3, 52)
48	nation	(1, 2, 31), (2, 3, 38), (5, 6, 23)

Table 6: Nodes with p_S-core numbers at least equal to 20.

Node		$\mathbf{p}_S$**-core number** (≥ 20)
49	police	(2, 3, 20)

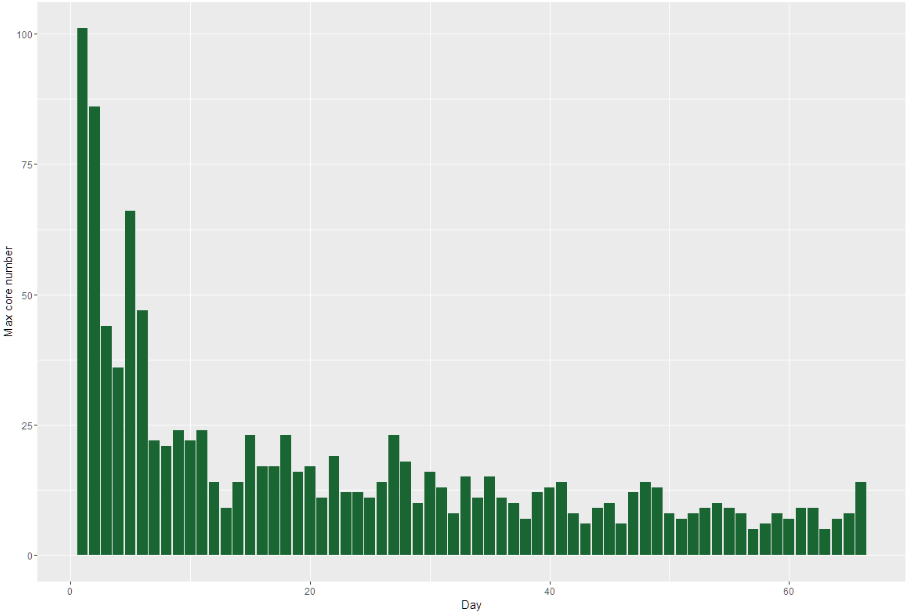

Figure 6: Maximum p_S-core numbers by days from the event.

5.3 Stem cell research

[13]

A data set on the stem cell research during 1997-2012 in Spain was collected by Gisela Cantos-Mateos. A data set consists of data on papers about stem cell research in the SCI (Science Citation Index).

A network consists of $n = 577$ nodes, representing Spanish institutions, and $m = 8578$ links, representing collaborations between institutions.

Core number represents size of a group of more collaborative institutions in which a node is included. Nodes with temporal core numbers at least equal to 20 are listed in Tab. 7.

Table 7: Nodes with core numbers at least equal to 20.

Node		**Core number** (≥ 20)
2	HCSC/M	(2010, 2011, 20), (2011, 2012, 21)
3	IN/A	(2008, 2009, 25)

[13] citiraj stembib; dodaj Å¡e vec podatkov o podatkih - recimo casovno obdobje

Table 7: Nodes with core numbers at least equal to 20.

Node		**Core number** (≥ 20)
5	CIC-IBMCC/SA	(2010, 2011, 20), (2011, 2013, 22)
6	HUS/SA	(2008, 2009, 25), (2010, 2011, 20), (2011, 2012, 21), (2012, 2013, 22)
8	IDIBELL/B	(2011, 2012, 20)
9	UB/B	(2008, 2009, 25), (2010, 2011, 20), (2011, 2013, 22)
10	UNIZAR/Z	(2008, 2009, 21), (2012, 2013, 21)
11	USAL/SA	(2008, 2009, 25), (2010, 2011, 20), (2011, 2013, 22)
12	HVH/B	(2010, 2011, 20), (2011, 2013, 22)
13	HNJ/M	(2010, 2011, 20), (2012, 2013, 22)
16	ICO/CT	(2008, 2009, 25), (2010, 2011, 20), (2012, 2013, 22)
17	HMM/MU	(2011, 2012, 22)
259	HMS/Z	(2011, 2012, 21)
20	UPC/B	(2011, 2012, 21), (2012, 2013, 22)
21	ICREA/B	(2010, 2011, 20)
22	HDM/B	(2008, 2009, 21), (2012, 2013, 22)
23	UNAV	(2008, 2009, 25), (2011, 2013, 22)
24	UPV-EHU	(2008, 2009, 21), (2010, 2011, 20)
27	HISC3/M	(2008, 2009, 25), (2010, 2011, 20), (2011, 2013, 22)
543	PFIZER/M	(2011, 2012, 21)
32	HRYC/M	(2008, 2009, 21), (2010, 2011, 20), (2011, 2013, 22)
289	HJXXIII/T	(2008, 2009, 25)
34	HCL/V	(2010, 2011, 20), (2011, 2013, 22)
35	HUGTIP/B	(2010, 2011, 20), (2012, 2013, 20)
36	UAB/B	(2008, 2009, 21), (2010, 2011, 20), (2011, 2013, 22)
37	US/SE	(2010, 2011, 20)
38	UV/V	(2008, 2009, 25), (2010, 2011, 20), (2011, 2013, 22)
40	HCL/B	(2010, 2011, 20), (2011, 2013, 22)
46	IDIBAPS/B	(2008, 2009, 21), (2010, 2011, 20), (2011, 2013, 22)
48	HSCSP/B	(2008, 2009, 21), (2010, 2011, 20), (2011, 2013, 22)
51	HBST/B	(2008, 2009, 25), (2011, 2012, 21)
53	H12O/M	(2008, 2009, 25), (2011, 2013, 21)
54	CNB	(2012, 2013, 22)
55	HUPH/M	(2011, 2012, 21), (2012, 2013, 22)
57	HCLB/Z	(2011, 2012, 21)
58	HCUN/NA	(2011, 2013, 22)
266	URL/B	(2012, 2013, 22)
62	UAM/M	(2008, 2009, 25), (2010, 2011, 20), (2011, 2013, 22)
63	UCM/M	(2008, 2009, 25), (2010, 2011, 20), (2011, 2013, 22)
65	HRS/CO	(2012, 2013, 21)
66	HCRUCES/BI	(2011, 2012, 21)
67	CIPF/V	(2008, 2009, 21)
69	UMA/MA	(2008, 2009, 21), (2010, 2011, 20), (2011, 2012, 21), (2012, 2013, 22)
72	HUMV/S	(2008, 2009, 25), (2011, 2013, 22)
73	UGR/GR	(2011, 2012, 22), (2012, 2013, 20)
74	CIBERDEM	(2008, 2009, 25)
75	SEHH	(2011, 2012, 21), (2012, 2013, 20)
76	HULP/M	(2008, 2009, 25), (2010, 2011, 20), (2011, 2013, 22)
77	UPV/V	(2008, 2009, 21)

Table 7: Nodes with core numbers at least equal to 20.

Node		**Core number** (≥ 20)
336	TERCEL	(2008, 2009, 25)
81	HVA/MU	(2011, 2012, 20), (2012, 2013, 21)
82	UM/MU	(2008, 2009, 25)
85	UA/A	(2008, 2009, 25), (2011, 2012, 20)
87	HUP/M	(2011, 2013, 22)
344	HSO/M	(2011, 2012, 21)
89	UPF/B	(2008, 2009, 21), (2012, 2013, 22)
91	CIBERNED	(2012, 2013, 22)
92	GENYO/GR	(2011, 2012, 21)
93	CBMSO/M	(2010, 2011, 20), (2011, 2012, 22), (2012, 2013, 21)
96	BACM/GR	(2011, 2013, 22)
272	ULEON/LE	(2011, 2013, 22)
310	SESCAM/TO	(2011, 2012, 21)
102	USC	(2011, 2013, 22)
103	CIBEROBN	(2011, 2012, 21)
108	HGJF/CA	(2011, 2012, 21)
109	HVN/GR	(2008, 2009, 21), (2011, 2012, 22), (2012, 2013, 21)
111	HANDERSON/M	(2011, 2012, 21)
112	INCYL	(2008, 2009, 21), (2010, 2011, 20), (2012, 2013, 21)
258	INIA/M	(2012, 2013, 22)
123	H-JAEN	(2012, 2013, 22)
124	HJC/C	(2011, 2012, 20)
403	SERGAS/C	(2008, 2009, 25)
133	HCSOL/MA	(2012, 2013, 22)
134	IBV/V	(2008, 2009, 25)
135	CRG/B	(2008, 2009, 25), (2011, 2012, 21)
535	SERIDA/O	(2011, 2012, 21)
146	HSC/GR	(2010, 2011, 20)
147	HGM/M	(2010, 2011, 20), (2011, 2013, 22)
149	IIBM/M	(2011, 2012, 22)
150	UNIOVI/O	(2010, 2011, 20)
153	UAH/M	(2008, 2009, 25)
176	HUVR/SE	(2008, 2009, 25), (2011, 2013, 22)
186	UVA	(2012, 2013, 22)
192	IRB/B	(2011, 2012, 22)
452	HVS/TO	(2011, 2012, 21)
80	HUPLFV/V	(2008, 2009, 25), (2010, 2011, 20), (2011, 2013, 22)
307	HVB/LE	(2010, 2011, 20)
232	HUB/B	(2008, 2009, 25)
492	UPNA/NA	(2012, 2013, 22)
253	UCLM	(2011, 2012, 21), (2012, 2013, 22)

In Fig. 7 are shown maxumum core numbers through years. There is a major difference in year 2008 and after this year larger groups of collaborative institutions are formed each year.

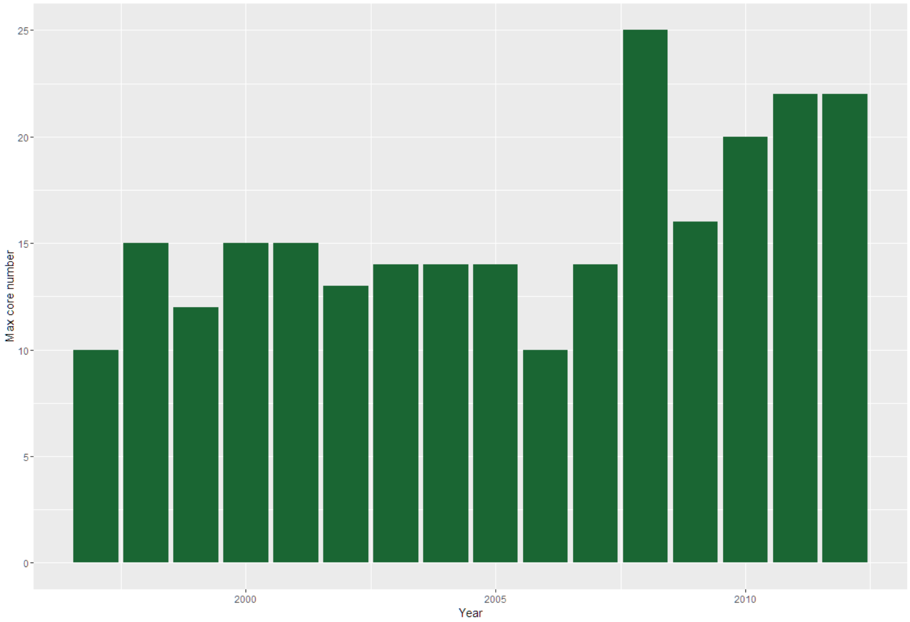

Figure 7: Maximum core numbers by years.

p_S-core numbers include in the collaboration also a number of articles produced. In Tab. 8 are listed nodes with p_S-core number at least equal to 25.

Table 8: Nodes with p_S-core numbers at least equal to 25.

Node		**p_S-core number** (≥ 25)
6	HUS/SA	(2008, 2009, 36), (2009, 2010, 26), (2010, 2011, 31), (2011, 2012, 30), (2012, 2013, 38)
9	UB/B	(2008, 2009, 36), (2009, 2010, 26), (2010, 2011, 31), (2011, 2012, 55), (2012, 2013, 39)
12	HVH/B	(2008, 2009, 26), (2009, 2010, 30), (2010, 2011, 31), (2011, 2012, 46), (2012, 2013, 38)
27	HISC3/M	(2008, 2009, 36), (2009, 2010, 30), (2010, 2011, 31), (2011, 2012, 55), (2012, 2013, 41)
32	HRYC/M	(2008, 2009, 36), (2009, 2010, 25), (2010, 2011, 30), (2011, 2012, 54), (2012, 2013, 41)
62	UAM/M	(2008, 2009, 36), (2009, 2010, 30), (2010, 2011, 31), (2011, 2012, 55), (2012, 2013, 39)
63	UCM/M	(2008, 2009, 36), (2009, 2010, 30), (2010, 2011, 31), (2011, 2012, 55), (2012, 2013, 41)
69	UMA/MA	(2008, 2009, 36), (2009, 2010, 26), (2010, 2011, 30), (2011, 2012, 32), (2012, 2013, 37)
40	HCL/B	(2008, 2009, 36), (2009, 2010, 30), (2010, 2011, 31), (2011, 2012, 55), (2012, 2013, 41)
46	IDIBAPS/B	(2008, 2009, 36), (2009, 2010, 30), (2010, 2011, 31), (2011, 2012, 55), (2012, 2013, 41)

Table 8: Nodes with p_S-core numbers at least equal to 25.

Node		$\mathbf{p}_S$**-core number** (≥ 25)
48	HSCSP/B	(2008, 2009, 36), (2009, 2010, 30), (2010, 2011, 31), (2011, 2012, 55), (2012, 2013, 41)
38	UV/V	(2008, 2009, 36), (2009, 2011, 30), (2011, 2012, 55), (2012, 2013, 41)
11	USAL/SA	(2008, 2009, 36), (2010, 2011, 31), (2011, 2012, 42), (2012, 2013, 38)
89	UPF/B	(2008, 2009, 36), (2010, 2011, 25), (2011, 2012, 34), (2012, 2013, 40)
147	HGM/M	(2009, 2010, 26), (2010, 2011, 31), (2011, 2012, 42), (2012, 2013, 39)
176	HUVR/SE	(2008, 2009, 36), (2010, 2011, 31), (2011, 2012, 55), (2012, 2013, 39)
23	UNAV	(2008, 2009, 36), (2009, 2010, 30), (2011, 2012, 54), (2012, 2013, 41)
34	HCL/V	(2008, 2009, 25), (2010, 2011, 31), (2011, 2012, 34), (2012, 2013, 38)
36	UAB/B	(2008, 2009, 36), (2009, 2011, 30), (2011, 2012, 38), (2012, 2013, 37)
58	HCUN/NA	(2008, 2009, 25), (2010, 2011, 28), (2011, 2012, 52), (2012, 2013, 41)
72	HUMV/S	(2008, 2009, 36), (2010, 2011, 25), (2011, 2012, 55), (2012, 2013, 41)
76	HULP/M	(2008, 2009, 36), (2010, 2011, 31), (2011, 2012, 46), (2012, 2013, 39)
80	HUPLFV/V	(2008, 2009, 36), (2009, 2010, 30), (2010, 2011, 31), (2011, 2013, 38)
24	UPV-EHU	(2008, 2009, 28), (2009, 2010, 30), (2010, 2011, 31)
13	HNJ/M	(2008, 2009, 27), (2010, 2011, 30), (2012, 2013, 38)
16	ICO/CT	(2008, 2009, 36), (2010, 2011, 31), (2012, 2013, 37)
51	HBST/B	(2008, 2009, 25), (2011, 2012, 33), (2012, 2013, 29)
5	CIC-IBMCC/SA	(2010, 2011, 30), (2011, 2012, 55), (2012, 2013, 41)
93	CBMSO/M	(2010, 2011, 30), (2011, 2012, 44), (2012, 2013, 37)
109	HVN/GR	(2008, 2009, 36), (2011, 2012, 46), (2012, 2013, 35)
112	INCYL	(2008, 2009, 36), (2010, 2011, 29), (2012, 2013, 32)
53	H12O/M	(2008, 2009, 36), (2011, 2012, 33), (2012, 2013, 37)
55	HUPH/M	(2010, 2011, 26), (2011, 2012, 34), (2012, 2013, 36)
2	HCSC/M	(2010, 2011, 31), (2011, 2012, 32)
35	HUGTIP/B	(2010, 2011, 31), (2012, 2013, 32)
37	US/SE	(2009, 2010, 30), (2010, 2011, 31)
67	CIPF/V	(2008, 2009, 32), (2012, 2013, 32)
73	UGR/GR	(2011, 2012, 39), (2012, 2013, 37)
3	IN/A	(2008, 2009, 36)
49	CMRB/B	(2008, 2009, 28)
307	HVB/LE	(2010, 2011, 28)
54	CNB	(2012, 2013, 41)
57	HCLB/Z	(2011, 2012, 30)
81	HVA/MU	(2009, 2010, 30), (2012, 2013, 37)

Table 8: Nodes with p_S-core numbers at least equal to 25.

Node		**$\mathbf{p}_S$-core number** (≥ 25)
82	UM/MU	(2008, 2009, 36)
85	UA/A	(2008, 2009, 25)
87	HUP/M	(2011, 2012, 47), (2012, 2013, 41)
344	HSO/M	(2011, 2012, 25)
91	CIBERNED	(2012, 2013, 38)
289	HJXXIII/T	(2008, 2009, 25)
28	CT/M	(2010, 2011, 25)
101	HGC/CS	(2012, 2013, 27)
96	BACM/GR	(2011, 2012, 38), (2012, 2013, 37)
272	ULEON/LE	(2011, 2012, 55), (2012, 2013, 41)
357	HALC/M	(2010, 2011, 26)
102	USC	(2011, 2012, 55), (2012, 2013, 41)
108	HGJF/CA	(2011, 2012, 26)
258	INIA/M	(2012, 2013, 38)
124	HJC/C	(2010, 2011, 30), (2011, 2012, 26)
10	UNIZAR/Z	(2008, 2009, 36), (2012, 2013, 25)
15	HSD/PM	(2010, 2011, 26), (2012, 2013, 27)
17	HMM/MU	(2011, 2012, 38)
259	HMS/Z	(2011, 2012, 34)
20	UPC/B	(2012, 2013, 29)
21	ICREA/B	(2010, 2011, 31)
22	HDM/B	(2008, 2009, 36), (2012, 2013, 38)
403	SERGAS/C	(2008, 2009, 36)
405	HVC/ZA	(2009, 2010, 30)
133	HCSOL/MA	(2012, 2013, 38)
134	IBV/V	(2008, 2009, 25)
135	CRG/B	(2008, 2009, 33)
535	SERIDA/O	(2011, 2012, 34)
146	HSC/GR	(2010, 2011, 28)
149	IIBM/M	(2011, 2012, 50)
150	UNIOVI/O	(2009, 2010, 25), (2010, 2011, 31)
153	UAH/M	(2008, 2009, 25)
171	UJAEN/J	(2010, 2011, 25), (2011, 2012, 27)
186	UVA	(2012, 2013, 27)
192	IRB/B	(2011, 2012, 38), (2012, 2013, 27)
266	URL/B	(2012, 2013, 27)
65	HRS/CO	(2012, 2013, 32)
66	HCRUCES/BI	(2011, 2012, 27)
74	CIBERDEM	(2008, 2009, 25)
77	UPV/V	(2008, 2009, 28)
336	TERCEL	(2008, 2009, 25)
232	HUB/B	(2008, 2009, 36)
492	UPNA/NA	(2012, 2013, 37)
245	HCAUL/LE	(2011, 2012, 26)
253	UCLM	(2011, 2012, 38), (2012, 2013, 32)

A diagram of maximum p$_S$-core numbers by year in Fig. 8 is great example of collaboration growing with each year. We already noticed that collaborations grew from year 2008 on in Fig. 7, but here we see that number of articles in those collaborations also grows.

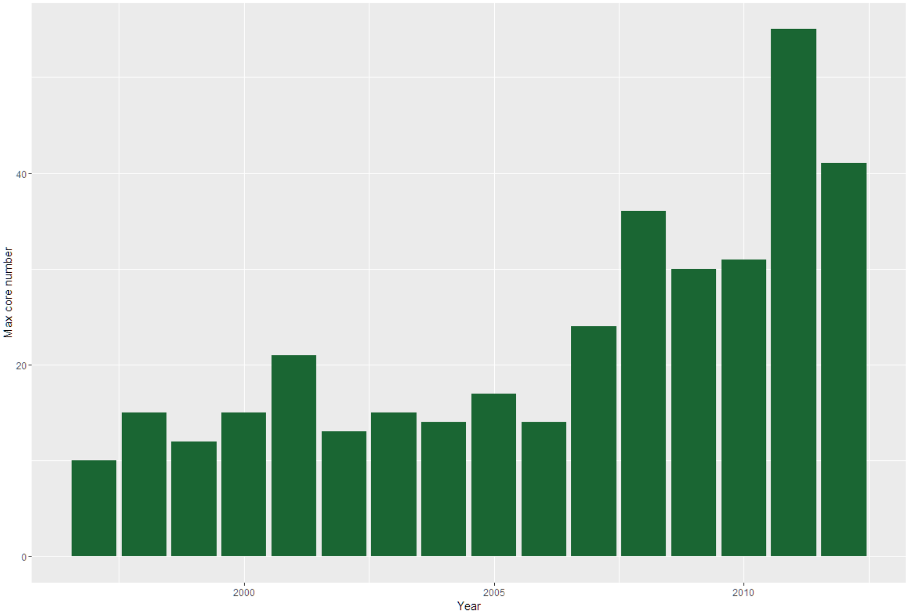

Figure 8: Maximum p_S-core numbers by years.

5.4 Violence network

14

Roberto Franzosi collected the data from the journal news in the period from January 1919 to December 1922. He was collecting information about the different types of interactions between political parties and other groups of people in Italy. The violence network contains only the data about violent actions and counts the number of interactions per month.

A network consists of $n = 29$ nodes representing groups of people and $m = 105$ links representing violent interactions.

To see which groups of people were having violent interactions with largest number of other groups we calculated temporal core numbers (see Fig. 9). Maximum core number is 3 and is present in only 9 groups, two of which are unknown ("undefined" and "?"). These core numbers appear in months 29, 31, 33, 39, and 40. To see which groups had violent interactions with at least two different groups of people in given month, see Fig. 10.

[14] citiraj violencebib, opisi casovno komponento

Table 9: Nodes with core numbers at least equal to 3.

Node		**Core number** (≥ 3)
16	workers	(29, 30, 3), (33, 34, 3), (39, 41, 3)
1	undefined	(29, 30, 3), (39, 40, 3)
2	?	(31, 32, 3), (33, 34, 3), (40, 41, 3)
3	people	(31, 32, 3), (33, 34, 3), (39, 40, 3)
4	police	(31, 32, 3), (33, 34, 3), (40, 41, 3)
21	catholics	(33, 34, 3)
7	fascists	(29, 30, 3), (31, 32, 3), (33, 34, 3), (39, 41, 3)
8	communists	(29, 30, 3)
10	socialists	(31, 32, 3), (40, 41, 3)

Table 10: Nodes with core numbers at least equal to 2.

Node		**Core number** (≥ 2)
1	undefined	(15, 16, 2), (17, 18, 2), (25, 29, 2), (29, 30, 3), (31, 32, 2), (38, 39, 2), (39, 40, 3), (41, 44, 2), (45, 46, 2), (48, 49, 2)
2	?	(14, 16, 2), (17, 18, 2), (28, 29, 2), (31, 32, 3), (32, 33, 2), (33, 34, 3), (34, 35, 2), (40, 41, 3)
3	people	(16, 18, 2), (23, 24, 2), (25, 26, 2), (28, 30, 2), (31, 32, 3), (33, 34, 3), (35, 37, 2), (39, 40, 3), (41, 43, 2), (48, 49, 2)
4	police	(11, 12, 2), (14, 20, 2), (21, 23, 2), (29, 31, 2), (31, 32, 3), (32, 33, 2), (33, 34, 3), (34, 37, 2), (38, 40, 2), (40, 41, 3)
5	land owners	(15, 16, 2), (17, 20, 2), (29, 30, 2), (36, 37, 2), (38, 40, 2), (42, 43, 2)
7	fascists	(11, 12, 2), (16, 17, 2), (19, 20, 2), (21, 24, 2), (25, 29, 2), (29, 30, 3), (30, 31, 2), (31, 32, 3), (32, 33, 2), (33, 34, 3), (34, 37, 2), (38, 39, 2), (39, 41, 3), (41, 44, 2), (45, 46, 2), (48, 49, 2)
8	communists	(28, 29, 2), (29, 30, 3), (31, 33, 2), (35, 37, 2), (43, 44, 2)
9	workers (agr)	(15, 16, 2), (17, 20, 2), (28, 30, 2), (31, 32, 2), (33, 35, 2), (38, 43, 2), (45, 46, 2)
10	socialists	(11, 12, 2), (16, 18, 2), (19, 20, 2), (22, 23, 2), (25, 26, 2), (27, 30, 2), (31, 32, 3), (33, 37, 2), (38, 40, 2), (40, 41, 3), (41, 42, 2)
12	war affected	(35, 36, 2), (39, 40, 2)
13	protesters	(15, 16, 2), (21, 22, 2), (29, 30, 2), (31, 32, 2), (38, 40, 2)
16	workers	(11, 12, 2), (14, 18, 2), (19, 20, 2), (21, 24, 2), (25, 26, 2), (27, 29, 2), (29, 30, 3), (30, 33, 2), (33, 34, 3), (34, 37, 2), (38, 39, 2), (39, 41, 3), (41, 44, 2), (45, 46, 2)
17	the right	(17, 18, 2), (41, 42, 2)
19	populars	(41, 42, 2)

Table 10: Nodes with core numbers at least equal to 2.

Node		**Core number** (≥ 2)
20	students	(17, 18, 2)
21	catholics	(33, 34, 3)
25	republicans	(26, 27, 2)
26	thugs	(29, 30, 2)
27	prisoners/arrested	(40, 41, 2)

A notion of p$_S$-core number tells us overall number of violent interactions in given month - including number of such interactions with each group in the same p$_S$-core. Most violent groups are listed in Tab. 11.

Table 11: Nodes with p$_S$-core numbers at least equal to 10.

Node		p_S**-core number** (≥ 10)
16	workers	(1, 2, 27), (10, 11, 11), (14, 15, 27), (16, 17, 11), (17, 18, 17), (18, 19, 12), (22, 23, 17), (25, 26, 11), (27, 28, 18), (28, 29, 16), (29, 30, 53), (30, 31, 56), (31, 32, 51), (32, 33, 30), (33, 34, 17), (34, 35, 71), (35, 36, 76), (36, 37, 53), (37, 38, 11), (38, 39, 23), (39, 40, 54), (40, 41, 13), (41, 42, 174), (42, 43, 25), (43, 44, 20), (45, 46, 15), (46, 47, 25)
1	undefined	(25, 26, 11), (27, 28, 12), (28, 29, 16), (41, 42, 133), (45, 46, 11)
3	people	(28, 29, 12)
4	police	(1, 2, 36), (6, 7, 15), (10, 11, 24), (12, 13, 29), (14, 15, 27), (15, 16, 13), (16, 17, 24), (17, 18, 17), (18, 19, 12), (22, 23, 17), (31, 32, 17)
7	fascists	(25, 26, 11), (27, 28, 30), (28, 29, 31), (29, 30, 64), (30, 31, 56), (31, 32, 51), (32, 33, 30), (33, 34, 24), (34, 35, 71), (35, 36, 76), (36, 37, 53), (37, 38, 13), (38, 39, 23), (39, 40, 54), (40, 41, 13), (41, 42, 174), (42, 43, 25), (43, 44, 20), (45, 46, 15), (46, 47, 25)
8	communists	(29, 30, 13), (30, 31, 10), (31, 32, 12)
9	workers (agr)	(10, 11, 24), (16, 17, 24), (28, 29, 16), (30, 31, 13), (36, 37, 11), (39, 40, 15), (43, 44, 10)
10	socialists	(10, 11, 10), (12, 13, 29), (27, 28, 30), (28, 29, 31), (29, 30, 64), (30, 31, 29), (31, 32, 17), (32, 33, 14), (33, 34, 24), (34, 35, 38), (35, 36, 23), (36, 37, 26), (37, 38, 13), (38, 39, 19), (39, 40, 54), (45, 46, 13)
12	war affected	(1, 2, 36)
13	protesters	(6, 7, 15), (15, 16, 13), (16, 17, 20)

6 Conclusion

TODO

Improve the complexity of the algorithm

Extend the algorithm to generalized temporal cores

Find user friendly presentations of results

Compare with the streaming core algorithms

Temporal Quantities - a Python 3 library for temporal network analysis: http://vladowiki.fmf.uni-lj.si/doku.php?id=tq

Acknowledgements. The second author was partially sponsored by Slovenian Research Agency (ARRS) - projects Z7-7614 (B).

References

[1] A. Ahmed, V. Batagelj, X. Fu, S.-H. Hong, D. Merrick, and A. Mrvar. Visualisation and analysis of the internet movie database. In *6th International Asia-Pacific Symposium on Visualization*, page 17–24. IEEE, 2007.

[2] H. Aksu, M. Canim, Y.-C. Chang, I. Korpeoglu, and Ö. Ulusoy. Multiresolution social network community identification and maintenance on big data platform. In *2013 IEEE International Congress on Big Data*. IEEE, 2013.

[3] H. Aksu, M. Canim, Y.-C. Chang, I. Korpeoglu, and Ö. Ulusoy. Distributed k-core view materialization and maintenance for large dynamic graphs. *IEEE Transactions on Knowledge and Data Engineering*, 26(10):2439 – 2452, 2014.

[4] S. Aridhi, A. Montresor, and Y. Velegrakis. Bladyg: A graph processing framework for large dynamic graphs. *Big Data Research*, 9, 2017.

[5] V. Batagelj and M. Zaveršnik. Fast algorithms for determining (generalized) core groups in social networks. *Advances in Data Analysis and Classification*, 5:129–145, 2010.

[6] G. Cantos-Mateos, M.A. Zulueta, B. Vargas-Quesada, and Z. Chinchilla-Rodriguez. Estudio evolutivo de la investigacion espanola con celulas madre. visualizacion e identificacion de las principales l'ineas de investigacion. *El Profesional de la Informacion*, 23(3):259–271, 2014.

[7] M. Cerinšek and V. Batagelj. Generalized two-mode cores. *Social Networks*, 42:80–87, 2015.

[8] R. Franzosi. Mobilization and countermobilization processes: From the red years (1919-20) to the black years (1921-22) in italy. a new methodological approach to the study of narrative data. *Theory and Society*, 26(2-3):275–304, 1997.

[9] R.-H. Li and J. X. Yu. Efficient core maintenance in large dynamic graphs. *IEEE Transactions on Knowledge and Data Engineering*, 26:2453–2465, 2014.

[10] S. B. Seidman. Network structure and minimum degree. *Social Networks*, 5:269–287, 1983.

[11] N. Wang, D. Yu, H. Jin, C. Qian, X. Xie, and Q.-S. Hua. Parallel algorithm for core maintenance in dynamic graphs. In *2017 IEEE 37th International Conference on Distributed Computing Systems*. IEEE, 2017.

[12] Y. Zhang, J. Xu Yu, Y. Zhang, and L. Qin. A fast order-based approach for core maintenance. In *2017 IEEE 33rd International Conference on Data Engineering*. IEEE, 2017.

.