Approaches to Analysis of Temporal Networks

Vladimir Batagelj

University of Ljubljana

1220. Sredin seminar, FMF/IMFM, Ljubljana
November 20, 2013

Outline

Temporal networks in Pajek

In a *temporal network*, the presence and activity of vertices and lines can change through time. Pajek supports two types of descriptions of temporal networks based on *presence* and on *events* (Pajek 0.47, July 1999). Here, we describe only an approach to capturing the presence of vertices and lines.

A *temporal network* $\mathcal{N}_T = (\mathcal{V}, \mathcal{L}, \mathcal{P}, \mathcal{W}, \mathcal{T})$ 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, vertices $v \in \mathcal{V}$ and lines $l \in \mathcal{L}$ 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 vertex v and $T(l)$, $T \in \mathcal{W}$, the activity set of time points for line l. The following *consistency* condition is imposed: If a line $l(u, v)$ is active in time point t then its end-vertices u and v should be active in time t. Formally we express this by

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

Temporal networks in Pajek

We denote a network consisting of lines and vertices active in time, $t \in \mathcal{T}$, by $\mathcal{N}(t)$ and call it the (network) *time slice* or *footprint* of t. Let $\mathcal{T}' \subset \mathcal{T}$ (for example, a time interval). The notion of a time slice is extended to $\mathcal{T}'$ by

$$\mathcal{N}(\mathcal{T}') = \bigcup_{t \in \mathcal{T}'} \mathcal{N}(t)$$

The time $\mathcal{T}$ is usually either a subset of integers, $\mathcal{T} \subseteq \mathbb{Z}$, or a subset of reals, $\mathcal{T} \subseteq \mathbb{R}$. In Pajek $\mathcal{T} \subseteq \mathbb{N}$.
$T(v)$ and $T(l)$ are usually described as a sequence of intervals.

Examples

Citation networks (WoS)
$\mathcal{V} = \{$ works $\}$
$\mathcal{L} = \{(u, v) : u$ cites $v\}$
$\mathcal{T} = \{$dates (years)$\}$
$T(V) = \{$publication date (year) of $v\}$
$T(u, v) = [$publication date (year) of $u, *]$

Project collaboration networks (Eu site, Sicris)
$\mathcal{V} = \{$ institutions $\}$
$\mathcal{L} = \{(u, v) : u$ and v work on a joint project$\}$
$\mathcal{T} = \{$dates$\}$
$T(V) = \mathcal{T}$
$T(u, v) = \{[a, b]$: exists a project P such that u and v are partners on P; a is the start and b is the finish date of $P\}$

Example
Multi-relational temporal network – KEDS/WEIS

```
% Recoded by WEISmonths, Sun Nov 28 21:57:00 2004
% from http://www.ku.edu/~keds/data.dir/balk.html
*vertices 325
1 "AFG" [1-*]
2 "AFR" [1-*]
3 "ALB" [1-*]
4 "ALBMED" [1-*]
5 "ALG" [1-*]
 ...
318 "YUGGOV" [1-*]
319 "YUGMAC" [1-*]
320 "YUGMED" [1-*]
321 "YUGMTN" [1-*]
322 "YUGSER" [1-*]
323 "ZAI" [1-*]
324 "ZAM" [1-*]
325 "ZIM" [1-*]
*arcs :0 "*** ABANDONED"
*arcs :10 "YIELD"
*arcs :11 "SURRENDER"
*arcs :12 "RETREAT"
 ...
*arcs :223 "MIL ENGAGEMENT"
*arcs :224 "RIOT"
*arcs :225 "ASSASSINATE TORTURE"
*arcs
224: 314 153 1 [4]              890402  YUG     KSV     224  (RIOT)  RIOT-TORN
212: 314 83 1 [4]               890404  YUG     ETHALB  212  (ARREST PERSON) ALB ETHNIC JAILED
224: 3 83 1 [4]                 890407  ALB     ETHALB  224  (RIOT)  RIOTS
123: 83 153 1 [4]               890408  ETHALB  KSV     123  (INVESTIGATE)   PROBING
 ...
42: 105 63 1 [175]              030731  GER     CYP     042  (ENDORSE)       GAVE SUPPORT
212: 295 35 1 [175]             030731  UNWCT   BOSSER  212  (ARREST PERSON) SENTENCED TO PRIS
43: 306 87 1 [175]              030731  VAT     EUR     043  (RALLY) RALLIED
13: 295 35 1 [175]              030731  UNWCT   BOSSER  013  (RETRACT)       CLEARED
121: 295 22 1 [175]             030731  UNWCT   BAL     121  (CRITICIZE)     CHARGES
122: 246 295 1 [175]            030731  SER     UNWCT   122  (DENIGRATE)     TESTIFIED
121: 35 295 1 [175]             030731  BOSSER  UNWCT   121  (CRITICIZE)     ACCUSED
```

Kansas Event Data System *KEDS*

Examples

Genealogies (GEDCOMs)

$\mathcal{V} = \{$ people $\}$

$\mathcal{L} = \mathcal{L}_1 \cup \mathcal{L}_2$

$\mathcal{L}_1 = \{(u, v) : u$ is a parent $v\}$

$\mathcal{L}_2 = \{(u, v) : u$ and v are married$\}$

$\mathcal{T} = \{$days$\}$

name function $\nu \in \mathcal{P}$: $\nu(v) =$ name of person v, $v \in \mathcal{V}$

gender function $g \in \mathcal{P}$: $g : \mathcal{V} \to \{M, F\}$

$T(v) = [\text{birth}(v), \text{death}(v)]$

for $(u, v) \in \mathcal{L}_1$: $T(u, v) = [\text{birth}(v), \min(\text{death}(u), \text{death}(v))]$

for $(u, v) \in \mathcal{L}_2$: $T(u, v) =$

$[\text{marriage}(u, v), \min(\text{death}(u), \text{death}(v), \text{divorce}(u, v))]$

Note: other (basic) kinship relations $\mathcal{L}_i$ can be included.

Examples

Phone calls (Orange data)
Airplanes timetable (Amadeus)

Approaches to temporal networks

We will describe a framework for analysis of temporal networks.
The time dimension was added to networks in different disciplines.
The earliest are the transport(ation) network analysis (Bell and Iida [3], Correa & [5]), project scheduling (CPM, Pert) in Operations Research (Moder [17]) and constraints networks in Artificial Intelligence (Dechter [8]).
There are also qualitative approaches to temporal networks. See for example Allen [1] and Vilain & [22]. For statistical approaches see Kolaczyk's book [14] and Snijders Siena page [21].
In last two decades the interest for analysis of temporal networks increased partially motivated by travel-support services and analysis of sequences of events (e-mails, news, phone calls, etc.). The approaches and results are surveyed by Holme and Saramäki in the paper [11] and the book [12].
Nicosia & [19] and Kempe & [13]

Approaches to temporal networks

Another overview was produced by Casteigts & [7] (see also [6]) based on their formalization of temporal networks – time-varying graphs or TVGs.
There are two important views on temporal networks:

- a network is providing the constraints to activities on it (for example, the network determined by air-flights time table).
- a network is representing interactions of events among actors (for example, KEDS networks, citation networks, etc.)

Processes on networks:

- travel, transmission
- difussion, broadcasting, elections
- flow

Walks in temporal networks

When dealing with walks in temporal networks we can usually consider two additional information – weights on lines:

- the *transition time* $\tau \in \mathcal{W}$; $\tau\colon \mathcal{L} \to \mathbb{R}_0^+$. $\tau(l)$ is equal to the time needed to traverse the line l. If the function τ is not given we can assume $\tau(l) = 0$ for all lines l.
- the *value* (length, cost, etc.) $w \in \mathcal{W}$; $w\colon \mathcal{L} \to \mathbb{R}$. If the function w is not given we can assume $w(l) = 1$ for all lines l.

In applications related to flows in network we need an additional weight

- the *capacity* $c \in \mathcal{W}$; $c\colon \mathcal{L} \to \mathbb{R}_0^+$. $c(l)$ is equal to the maximum of quantity of items transfered in a time unit over line l. If the function c is not given we can assume $c(l) = \infty$ (no limits) for all lines l.

In real-life networks the values of functions τ, w and c can also vary through time. For example, $\tau(a)$ can depend on the overall traffic in the network. In the following we shall assume that they are constant on each line.

Journeys

A *temporal walk* or *journey* $\sigma(v_0, v_k; t_0)$ from (source) vertex v_0 to (destination) vertex v_k starting at time $t_0 \in T(v_0)$ is a finite sequence

$$(t_0, v_0, (t_1, l_1), v_1, (t_2, l_2), v_2, \ldots, v_{k-2}, (t_{k-1}, l_{k-1}), v_{k-1}, (t_k, l_k), v_k)$$

where $l_i \in \mathcal{L}$, $t_i \in T$, $i = 1, 2, \ldots k$. The triples $v_{i-1}, (t_i, l_i)$ tell that in the vertex v_{i-1} at time t_i the line l_i was selected for the next transition. The sequence σ has to satisfy the conditions: the line l_i links vertex v_{i-1} to vertex v_i and is active during the transition:

- $l_i(v_{i-1}, v_i)$
- $t'_{i-1} \leq t_i$
- $[t_i, t'_i] \subseteq T(l_i)$

for $i = 1, 2, \ldots k$; where $t'_i = t_i + \tau(l_i)$ and $t'_0 = t_0$.
The number k is called a *length* of the walk σ. The *time used* by the walk σ is equal to

$$t(\sigma) = t'_k - t_0$$

and its *value*

$$w(\sigma) = \sum_{i=1}^{k} w(l_i)$$

Some quantities, regularity

Note: by the consistency from the third condition it follows that $t_i \in T(v_i)$ and $t'_i \in T(v_{i+1})$.

Departure time: $dep(\sigma) = t_1$
Arrival time: $arr(\sigma) = t'_k$
Duration: $dur(\sigma) = arr(\sigma) - dep(\sigma)$

May be the null walk has also to be introduced.

A temporal walk is *regular* if also

- $[t_0, t_1] \subseteq T(v_0)$
- $[t'_{i-1}, t_i] \subseteq T(v_i)$, for $i = 1, 2, \ldots k - 1$.

While waiting for the next step (transition) in vertex v_i this vertex should be all the time active.

Types of temporal networks

In the following we shall observe the temporal network inside a *time window* $[a, b] \subseteq \mathcal{T}$.

It seems that some special classes of temporal networks can be important in solving the problems on temporal networks:

a) *fixed vertices*: $T(v) = \mathcal{T}$ for each $v \in \mathcal{V}$

b) *discrete* (time): $\mathcal{T}$ is a finite set

c) *integer* (time): $\mathcal{T} \subseteq \mathbb{Z}$ and $\tau : \mathcal{L} \to \mathbb{N}$

d) *single interval*: $T(v)$s and $T(l)$s consist of single intervals

Fixed vertices or single interval temporal networks are regular.

Measures

Using quantities such as shortest duration, earliest arrival, etc. the standard network measures (degrees, betweenness, closeness, clustering index, etc.) can be extended (in different ways) to temporal networks. See for example Nicosia & [19].

Note that some of these measures are essentially time functions. In developing measures a kind of averaging over the time (window) is needed.

Measures based on time slices

The approach, supported by Pajek, based on slices is valid only in the case when $\tau(l) = 0$ for each line l. The slices are considered as static networks on which different structural properties (degrees, closeness, betweenness, clustering coefficient, etc.) can be computed thus producing different time series describing the evolution of a network.

Activity:

$$T(v,t) = \begin{cases} 1 & t \in T(v) \\ 0 & t \notin T(v) \end{cases}$$

$$act(v) = \int_{\mathcal{T}} T(v,t)dt$$

The corresponding optimization problems: fastest walk for the time used and minimal cost (or shortest) walk for the value, were partially (for special cases of temporal networks) studied in different fields such as artifical intelligence and operations research.

Adaptations of [19], [11] and [6] , [23].

Problems and algorithms

$$Q \in \{\text{short}, \text{fast}, \text{early}, \ldots\}$$

The Q-est journey from vertex u to vertex v for a given starting time t.
The Q-est journey from vertex u to vertex v for any starting time $t \in T(u)$.

One approach to algorithms is to reduce the problem to traditional problems on static networks.
The other option is to develop new algorithms (Casteigts &[6], George & [9], Xuan & [23]), and others.

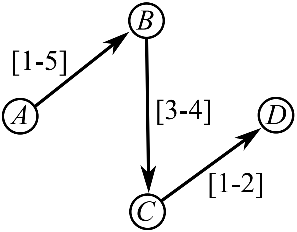
Properties of journeys

Reachabilty relation: vertex v is reachable from vertex u (in time window W), $u \rightsquigarrow_W v$, iff a journey exists from vertex u to vertex v in time window W.

$$\mathrm{InHor}(v; W) = \{u \in \mathcal{V} : u \rightsquigarrow_W v\}$$

$$\mathrm{OutHor}(v; W) = \{u \in \mathcal{V} : v \rightsquigarrow_W u\}$$

In static networks the reachabilty relation is transitive. This is not true for temporal networks.



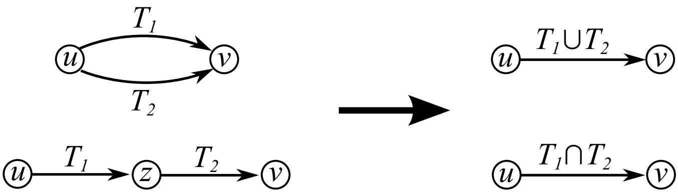
$W = [1, 5]$. $A \rightsquigarrow_W C$ and $C \rightsquigarrow_W D$, but not $A \rightsquigarrow_W D$

Semiring

For networks with $\tau = 0$ we can determine the reachability time sets

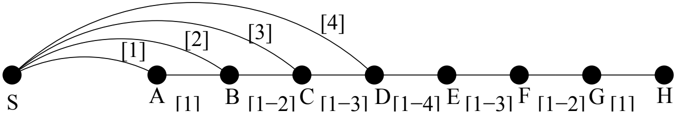
$$T(u, v) = \{t \in \mathcal{T} : u \rightsquigarrow_{[t]} v\}$$

using the semiring [2]
parallel: $T(u, v) = T_1(u, v) \cup T_2(u, v)$
sequential: $T(u, v) = T_1(u, z) \cap T_2(z, v)$



Properties of journeys

Xuan & [23]: the foundamental property for Dijkstra's shortest path algorithm is the property that the prefix paths of the shortest paths are the shortest paths themselves.



This is not true for temporal networks.
$W = [1, 4]$. The shortest journey from S to H is

$$\sigma(S, H) = \{S, A, B, C, D, E, F, G, H\}$$

The shortest journey from S to E is

$$\sigma(S, E) = \{S, D, E\}$$

Transformation to static networks

In OR they often transform (Ford ?) the temporal network $\mathcal{N} = (\mathcal{V}, \mathcal{L}, T, \{\tau, w\})$ with integer time $\mathcal{T} = \{0, 1, 2, \ldots, T\}$ into a traditional network $\mathcal{N}' = (\mathcal{V}', \mathcal{L}', w)$ determined as follows:

a) for each active time point $t \in T(v)$ of vertex v we produce its copy (v, t)

$$\mathcal{V}' = \bigcup_{v \in \mathcal{V}} \{v\} \times T(v)$$

b) the line $l' \in \mathcal{L}'$ is linking vertices $(v_1, t_1), (v_2, t_2) \in \mathcal{V}'$, $l'((v_1, t_1), (v_2, t_2))$, iff

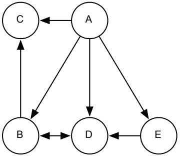
$$\exists l \in \mathcal{L} : (l(v_1, v_2) \wedge [t_1, t_2] \subseteq T(l) \wedge t_2 - t_1 = \tau(l))$$

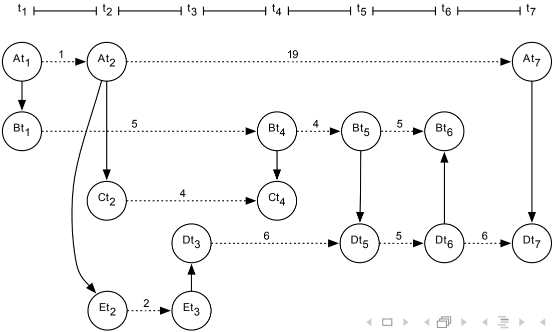
c) $w(l') = w(l)$

Using this transformation some traditional network problems for this type of temporal networks can be transformed to larger traditional problems for static networks.

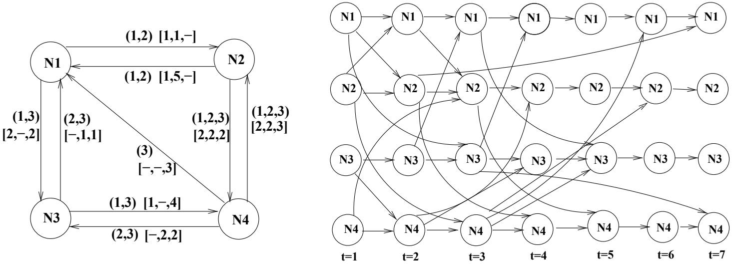
Transformation $\tau = 0$ [15]

Sender	Recipient	Time
A	B	t_1=0
A	C,E	t_2=1
E	D	t_3=3
B	C	t_4=5
B	D	t_5=9
D	B	t_6=14
A	D	t_7=20





Transformation to static networks [9]



(a) Time−aggregated Graph

(b) Time Expanded Graph

SP-TAG algorithm [9]

```
Input:
  1) G(N,E): a graph G with a set of nodes N and a set of edges E;
     Each node n ∈ N has a property:
        Node Presence Time Series : series of positive integers;
     Each edge e ∈ E has two properties:
       Edge Presence Time Series,
       Travel_time series : series of positive integers;
     σ_{u,v}(t) - travel time of edge uv at time t.
  2) s: Source node, s ⊆ N; 3) d: Destination node, d ⊆ N;
  4) t_start: Start Time;
Output: Shortest Route from s to d for t_start
Method:
        c[s] = t_start; ∀v ≠ s, c[v] = ∞;
        // c[u] is the cost at the node u.
        Insert s in priority queue Q.
        while Q is not empty do {
            u = extract_min(Q);
            for each node v adjacent to u do {
                t = min_t((u,v), c[u]);
                if t + σ_{u,v}(t) < c[v] {
                    c[v] = t + σ_{u,v}(t); parent[v] = u;
                    if v is not in Q, insert v in Q;
                }
            update Q;
            }
        }
    }
Output the route from s to d.
```

BEST algorithm [9]

```
Input:
  G(N,E): a graph G with a set of nodes N and a set of edges E;
      Each node n ∈ N has a property:
          Node Presence Time Series : series of positive integers;
      Each edge e ∈ E has two properties:
        Edge Presence Time Series,
        Travel_time series : series of positive integers;
      σ_{u,v}(t) - travel time of edge uv at time t.
Output:
  Best Start Time shortest route from s to d;
  Intialize;
  While Queue not Empty
      v = Dequeue();
      For every node u such that uv ∈ E
        For every entry in the cost series C_u of u
          if C_u(t) > σ_uv(t) + C_v(t + σ_uv(t))
            Update C_u(t);
            Enqueue(u);
            Update the descendant array of u.
  Find the minimum entry in the node time series.
  Return the BestStartTime and the ShortestRoute;
```

TVG classes

Casteigts & in [6], pages 31–38 describe several special classes of TVGs.

Evolution

The fundamental recent paper on network evolution is Palla & [20]. Some additional ideas we can find in Greene & [10].
One of the approaches is to identify groups or communities inside the temporal network and analyze their changes: birth, death, expansion (growth), contraction (decline), merging, splitting and reorganization. The groups can be determined using different procedures (leaders, islands, etc.).
The evolution of (selected, interesting) disjoint groups can be described by a *temporal partition* $C(v, t)$ with values

$$C(v,t) = \begin{cases} 0 & v \text{ is active in time } t\text{, but not a member of any group} \\ i & v \text{ is active in time } t \text{ and a member of group } i > 0 \\ \text{NA} & v \text{ is not active in time } t \end{cases}$$

Then the i-th temporal cluster $C(i, t)$ is

$$C(i,t) = \{v \in \mathcal{V} : C(v,t) = i\}$$

In applications we have to impose some additional conditions on groups expressing a kind of stability.

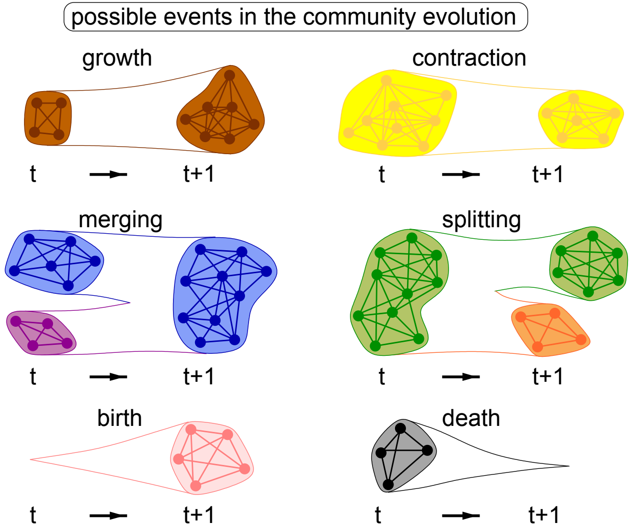
Evolution

In integer single interval temporal network we can define [20]

$$G(i,t) = \frac{C(i,t-1) \cap C(i,t)}{C(i,t-1) \cup C(i,t)}, \quad t \in \mathcal{T} \setminus \{t_0\}$$

if $C(i,t-1) \cup C(i,t) \neq \emptyset$, otherwise $G(i,t) = 0$.
Using the temporal clusters we can try to characterize the transitions (expansion, contraction, ...).

Evolution of groups [20]



Questions

- extend Pajek / create a special program for the support of analysis of temporal networks ?
- what are the complexities (NP, polynomial, subquadratic) of temporal network problems (for different types of networks) ?
- what can be used for large networks ?
- for which problems the corresponding semiring can be constructed ?
- development of measures (indices) for temporal networks.

References I

Allen, J.F. Maintaining Knowledge about Temporal Intervals. Communications of the ACM 26, 11, 832-843, November 1983.

Batagelj V.: Semirings for social networks analysis. Journal of Mathematical Sociology, 19(1994)1, 53-68.

M. G. H. Bell, Yasunori Iida: Transportation Network Analysis. Chichester: Wiley, 1997

Kenneth A. Berman: Vulnerability of scheduled networks and a generalization of Menger's Theorem. Networks, Volume 28, Issue 3, pages 125–134, 1996.

J.R. Correa, N.E. Stier-Moses. Wardrop Equilibria. Wiley Encyclopedia of Operations Research and Management Science, 2011.

A. Casteigts, P. Flocchini, Deterministic Algorithms in Dynamic Networks: Formal Models and Metrics Commissioned by Defense Research and Development Canada (DRDC), 82p, 2013.

References II

- Arnaud Casteigts, Paola Flocchini, Walter Quattrociocchi, Nicola Santoro: Time-varying graphs and dynamic networks. International Journal of Parallel, Emergent and Distributed Systems, Volume 27, Issue 5, pages 387-408, 2012
- Rina Dechter (ed.): Constraint Processing. Morgan Kaufmann, San Francisco, 2003.
- Betsy George, Sangho Kim, Shashi Shekhar: Spatio-temporal Network Databases and Routing Algorithms: A Summary of Results. D. Papadias, D. Zhang, and G. Kollios (Eds.): SSTD 2007, LNCS 4605, Springer-Verlag, Berlin, Heidelberg, pp. 460-477, 2007.
- D. Greene, D. Doyle, and P. Cunningham: Tracking the evolution of communities in dynamic social networks. In Proc. International Conference on Advances in Social Networks Analysis and Mining (ASONAM'10), 9-11 August, 2010, Odense, Denmark, pp. 176-183

References III

- Petter Holme, Jari Saramäki: Temporal networks. Physics Reports. Vol 519, Issue 3, 2012, p 97–125.
- Holme, Petter; Saramäki, Jari (Eds.) Temporal Networks. Understanding Complex Systems. Springer, 2013.
- D. Kempe, J. Kleinberg, A. Kumar. Connectivity and inference problems for temporal networks. Proc. 32nd ACM Symposium on Theory of Computing, 2000.
- Kolaczyk, E. D. (2009). Statistical Analysis of Network Data: Methods and Models. New York: Springer.
- Vassilis Kostakos: Temporal graphs
- Chia-Chen Lee, Yi-Hung Wu, Arbee L.P. Chen: Continuous Evaluation of Fastest Path Queries on Road Networks. D. Papadias, D. Zhang, and G. Kollios (Eds.): SSTD 2007, LNCS 4605, Springer-Verlag, Berlin, Heidelberg, pp. 20-37, 2007.

References IV

- Moder, Joseph J.; Phillips, Cecil R: Project Management with CPM and Pert. Second Edition, Van Nostrand Reinhold, 1970.
- Mrvar A., Batagelj V.: Analysis and visualization of Temporal Networks using Pajek. SFI/SAS JointWorkshop, The Dynamics of Groups and Institutions, Brdo pri Kranju, June 7 – 11, 2004. http://vlado.fmf.uni-lj.si/pub/networks/doc/seminar/SFI-SAS04tn.pdf
- Vincenzo Nicosia, John Tang, Cecilia Mascolo, Mirco Musolesi, Giovanni Russo, Vito Latora: Graph Metrics for Temporal Networks. Chapter in Petter Home and Jari Saramaki (Editors). Temporal Networks. Springer. 2013, p 15-40.
- Gergely Palla, Albert-László Barabási, Tamás Vicsek: Quantifying social group evolution Nature 446, 664-667 (2007).
- Snijders T.: Siena. http://www.stats.ox.ac.uk/ snijders/siena/

References V

M. Vilain, H. Kautz, P. Van Beek: Constraint Propagation Algorithms for Temporal Reasoning; A revised Report. In D.S. Weld, J. de Kleer (eds.) Readings in Qualitative Reasoning about Physical Systems, p 373-381, Morgan Kaufmann, 1990.

B. Bui Xuan, A. Ferreira, And A. Jarry. Computing Shortest, Fastest, and Foremost Journeys in Dynamic Networks. International Journal of Foundations of Computer Science, 14(2), p. 267-285, 2003.