Some ideas

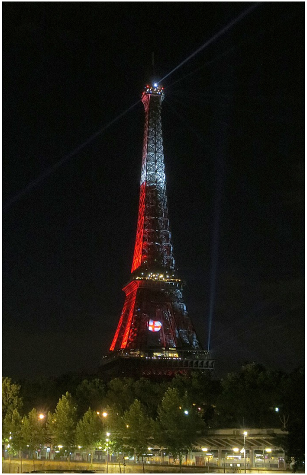
Vladimir Batagelj

IMFM Ljubljana and IAM UP Koper

Kinsources Conference
July 7-8, 2016,Collège de France, Paris

kinsources
archive, share, analyze and compare kinship data

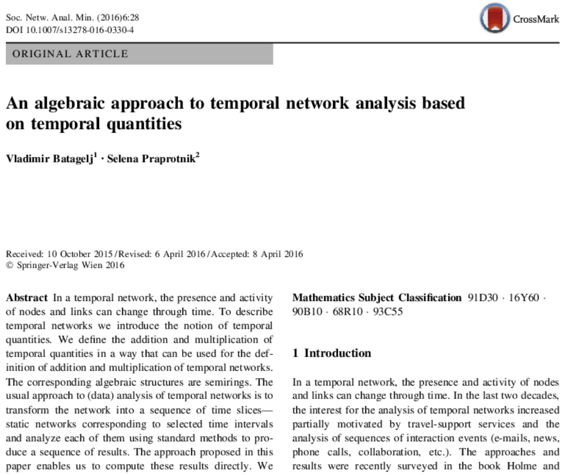
Outline



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

Current version of slides (July 25, 2016 at 14 : 50):
Kinsources'16 slides PDF

Temporal networks



Soc. Netw. Anal. Min. (2016)6:28
DOI 10.1007/s13278-016-0330-4

ORIGINAL ARTICLE

An algebraic approach to temporal network analysis based on temporal quantities

Vladimir Batagelj[1] · Selena Praprotnik[2]

Received: 10 October 2015 / Revised: 6 April 2016 / Accepted: 8 April 2016
© Springer-Verlag Wien 2016

Abstract In a temporal network, the presence and activity of nodes and links can change through time. To describe temporal networks we introduce the notion of temporal quantities. We define the addition and multiplication of temporal quantities in a way that can be used for the definition of addition and multiplication of temporal networks. The corresponding algebraic structures are semirings. The usual approach to (data) analysis of temporal networks is to transform the network into a sequence of time slices—static networks corresponding to selected time intervals and analyze each of them using standard methods to produce a sequence of results. The approach proposed in this paper enables us to compute these results directly. We

Mathematics Subject Classification 91D30 · 16Y60 · 90B10 · 68R10 · 93C55

1 Introduction

In a temporal network, the presence and activity of nodes and links can change through time. In the last two decades, the interest for the analysis of temporal networks increased partially motivated by travel-support services and the analysis of sequences of interaction events (e-mails, news, phone calls, collaboration, etc.). The approaches and results were recently surveyed in the book Holme and

on-line, arXiv

Temporal quantities

We describe temporal quantities with sequences

$$a = ((s_i, f_i, v_i))_{i=1}^{k}$$

where each triple says: a temporal quantity a has on the time interval $[s_i, f_i)$ the value v_i.
The following are two temporal quantities a and b represented in Python

```
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)]
```

The temporal quantity a has on interval $[1, 5)$ value 2, on interval $[6, 8)$ value 1, on interval $[11, 12)$ value 3, etc. Outside the specified intervals its value is undefined, ⌘. We require

$$v + ⌘ = v \qquad \text{and} \qquad v \cdot ⌘ = ⌘$$

Addition and multiplication

We define the sum $s = a + b$ and the product $p = a \cdot b$ of temporal quantities a and b (over selected semiring) as

$$s(t) = a(t) + b(t) \qquad \text{and} \qquad p(t) = a(t) \cdot b(t)$$

For example, for

```
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)]
```

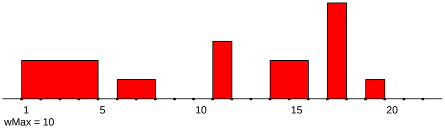
we get

```
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),
     (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)]
```

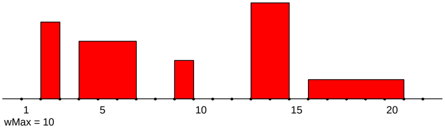
They are visually displayed at the bottom half of figures on the following slides.

Addition of temporal quantities.

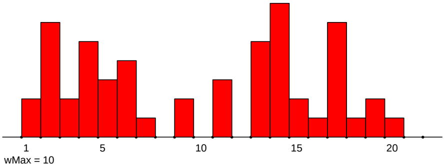
a :



b :

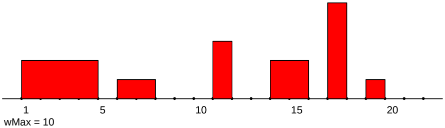


$a + b$:

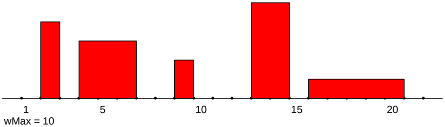


Multiplication of temporal quantities.

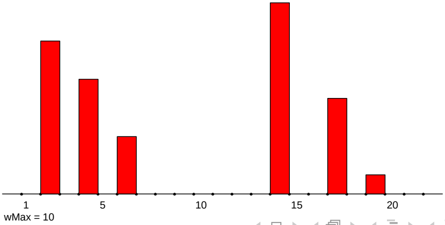
a :



b :



$a \cdot b$:



Possible application in genealogical analysis.

Assume that temporal quantities are describing the life span of people. For example $p = [(1903, 1995, 1)]$ and $q = [(1931, 2016, 1)]$. Then

$$p + q = [(1903, 1931, 1), (1931, 1995, 2), (1995, 2016, 1)]$$

Extending the sum to a group of people we get the temporal quantity describing how the size of the group is changing through time.

Other applications can be developed.

Networks in D3.js

- Force: Force-Directed Graph, Force Layout & Matrix Market Format, 3D Force Layout; An A to Z of extra features for the d3 force layout
- Directed: Directed Graph Editor, Directed Edges (Curves and Arrow Markers), Mobile Patent Suits
- Matrix: Co-occurrence Matrix
- Hive: Hive Plots
- Chord: Chord Diagram, Hierarchical Edge Bundling
- Applications: Linked JAZZ, Ontology Visualization, Visualizing Package Dependencies, Connectome explorer for the "brain" of C. elegans, Gene functional interaction networks
- More: D3 gallery, The Big List of D3.js Examples - Christophe Viau, Over 2000 D3.js Examples and Demos

Project

- **netJSON**: develop a JSON based format for description of networks. It should be "complete" – it can be used also to describe multi-relational, temporal, two-mode networks, and collections of networks. netJSON network description can be extended with a layout information. (jsongraph)
- **netD3.js**: collect and adapt for netJSON selected existing network visualization solutions based on D3.js, and develop new ones.

netJSON could serve as a data exchange format among network analysis programs (conversion program from/to netJSON).

Programers may export their results in netJSON and use net3D.js for their visualization.

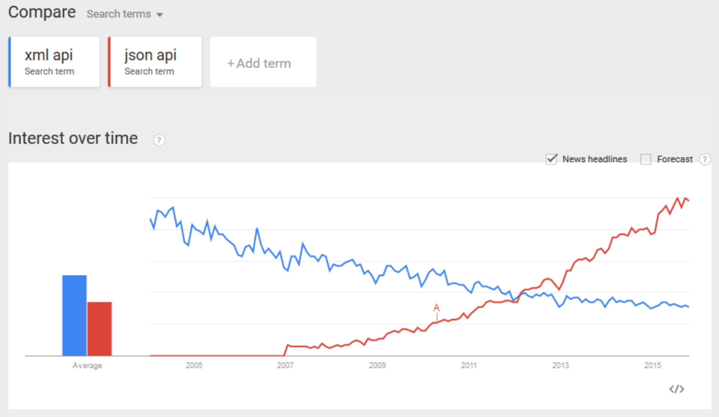
netJSON and netD3.js for network analysts

network → netJSON → SVG → {PDF, PNG, EPS}

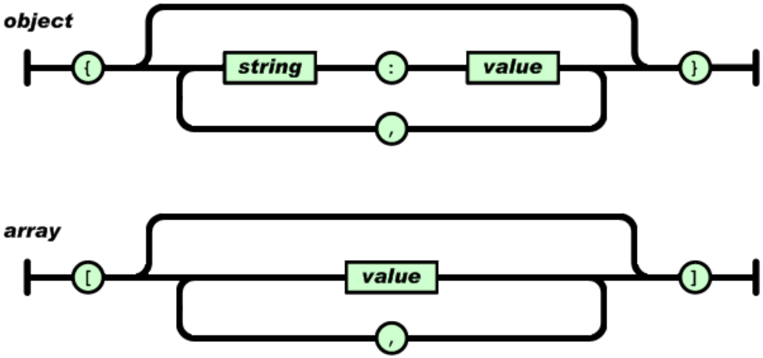
- Prepare your network data in netJSON format (in a text editor, from Excel tables using R, export from SNA programs and convert to netJSON). Add the layout information.
- Use selected netD3.js templates to visualize the network.
- Optionally, save the SVG picture, enhance it in some vector graphics editor (AI, Inkscape) and export it in selected format (PDF, ESP, PNG, . . .).

XML api – JSON api

In near past, for description of structured data the XML (Extensible Markup Language) was mostly used. In last five years a JSON format started to replace it. Google trends



JSON grammar



Networks in JSON format / basic netJSON

class.json

```
{ "netJSON": "basic",
  "info":{ "network": "class", "org": 1, "nNodes": 15,
     "nArcs" : 30, "nEdges": 13, "nWeak" : 1,
     "title" : "borrowing study materials",
     "meta"  : { "date": "October 2015", "author": "V. Batagelj" }
  },
  "nodes": [
     { "id":  1, "short": "m02", "x": 0.1857, "y": 0.2781, "size": 1 },
     { "id":  2, "short": "m03", "x": 0.5482, "y": 0.6169, "size": 1 },
     { "id":  3, "short": "w07", "x": 0.2219, "y": 0.4526, "size": 2 },
     { "id":  4, "short": "w09", "x": 0.8078, "y": 0.3223, "size": 2 },
...
     { "id": 14, "short": "m89", "x": 0.4000, "y": 0.8469, "size": 1 },
     { "id": 15, "short": "m96", "x": 0.3482, "y": 0.1778, "size": 1 }
  ],
  "links": [
     { "type": "arc",  "source": 6, "target": 15, "weight": 1 },
     { "type": "arc",  "source": 2, "target":  7, "weight": 1 },
...
     { "type": "arc",  "source":15, "target":  3, "weight": 1 },
     { "type": "edge", "source": 6, "target": 12, "weight": 1 },
...
     { "type": "edge", "source": 4, "target": 12, "weight": 1 },
     { "type": "edge", "source": 6, "target": 13, "weight": 1 }
  ]
}
```

Mode "general"

```
{ "netJSON": "general",
  "info":{ "network": "biblio",
     "title" : "bibliographic networks",
     "meta"  : { "date": "October 2015", "author": "V. Batagelj" }
  },
  "default": { "layout": {"coord": "circular"},
    "style": {"nodes": {"shape":"circle", "r":1, "color":"red"},
      "links": {"line" : "solid", "lw":2, "color": "blue"}
    },
    "links": {"linktype":"arc"}
  },
  "A": { "type":"nodes",
         "info": {"org": 1, "nNodes": 35},
         "style": { "color": "red", "shape": "circle", "size":5},
         "data": [ {"id":1, "short": "lab1", ...  },
                    ...
                 ],
  },
  "B": { "type":"nodes",
         ...
  },
```

Mode "general"

```
  "R": { "type":"links",
         "nodes": ["A","B"],
         "linktype": "edge",
         "data": [{ "type": "arc", "id":1, "n1":a, "n2":b, "w":v, ...},
           ...
         ]
       },
  "v": { "type":"vector",
         "info": { "size": 35, "type": "integer" },
         "nodes": "A",
         "data": [3,7,5,1,8,2,....]
       },
  ...
  "g2": { "type": "layout",
          "nodes": ["A","B"], "links":"R",
          "assign": { "coord": ["v1","v2"], "r":"v3", "color": "p1", ...
          "style": { "shape": "square" }
        }
}
```

Example

```
> setwd("C:/Users/batagelj/work/Python/graph/SVG/force")
> source("C:/Users/batagelj/work/Python/graph/SVG/force/ERgraphJS.R")
> ErdosRenyiNet('ERgraph.js',50,85)
> chrome <- "C:/Program Files (x86)/Google/Chrome/Application/Chrome.exe"
> args <- c(
+   "C:/Users/batagelj/work/Python/graph/SVG/force/ForceJS.html",
+   "-incognito",
+   "--app=google.com",
+   "--window-size=500,300" )
> system2(chrome,invisible=TRUE,wait=FALSE,args=args)
> ErdosRenyiNet('ERgraph.js',200,350)
> system2(chrome,invisible=TRUE,wait=FALSE,args=args)
```

ERgraphJS.R, ERgraph.js, ForceJS.html, forceNet.js.

D3.js solutions for kinship data can be developed.

Conclusions

- Explore the possibility of using the algebraic approach to temporal networks for solving problems in kinship analysis.
- Develop a JSON format and D3.js based visualization solutions for kinship analysis.
- Include in Kinsources the option that allows registered users to comment on a selected data set (analyses, publications, additional meta data, inconsistencies, etc.).