



**HAL**  
open science

# A logical approach for temporal and multiplex networks analysis

Esteban Bautista, Matthieu Latapy

► **To cite this version:**

Esteban Bautista, Matthieu Latapy. A logical approach for temporal and multiplex networks analysis. The 10th International Conference on Complex Networks and their Applications, 2021, Nov 2021, Madrid, Spain. hal-03367149

**HAL Id: hal-03367149**

**<https://hal.science/hal-03367149v1>**

Submitted on 6 Oct 2021

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# A logical approach for temporal and multiplex networks analysis

Esteban Bautista and Matthieu Latapy

Sorbonne Université, CNRS, LIP6, F-75005 Paris, France

## 1 Introduction

Many systems generate data as a set of triplets  $(a, b, c)$ : they may represent that user  $a$  called  $b$  at time  $c$  or that customer  $a$  purchased product  $b$  in store  $c$ . These datasets are traditionally studied as networks with an extra dimension (time or layer), for which the fields of temporal and multiplex networks have extended graph theory to account for the new dimension [1]. However, such frameworks detach one variable from the others and allow to extend one same concept in many ways, making it hard to capture patterns across all dimensions and to identify the best definitions for a given dataset. This work overrides this vision and proposes a direct processing of the set of triplets. While [2] also approaches triplets directly, it focuses on specific patterns and applications. Our work shows that a more general analysis is possible by partitioning the data and building categorical propositions (CPs) that encode informative patterns. We show that several concepts from graph theory can be framed under this formalism and leverage such insights to extend the concepts to data triplets. Lastly, we propose an algorithm to list CPs satisfying specific constraints and apply it to a real world dataset.

## 2 Results

**Analysis via propositions.** We consider the most general case where all the triplet entries come from arbitrary sets  $A, B, C$ . We thus define a triplet space as  $\mathcal{S} = \{(a, b, c) | a \in A, b \in B, c \in C\}$  and a dataset as  $\mathcal{D} \subseteq \mathcal{S}$ . We also define the sub-dataset induced by  $\alpha \subseteq A, \beta \subseteq B, \gamma \subseteq C$  as  $\mathcal{D}_{(\alpha, \beta, \gamma)} = \{(a, b, c) \in \mathcal{D} | a \in \alpha, b \in \beta, c \in \gamma\}$ . Our main observation is that given  $\alpha \subseteq A, \beta \subseteq B, \gamma \subseteq C$ , we can partition  $\mathcal{D}$  into eight disjoint regions (or bins) according to whether a triplet has its entries in  $\alpha, \beta$  and  $\gamma$ . Then, we can capture how triplets in  $\mathcal{D}$  distribute across these bins via CPs. This process is illustrated in the Fig 1-Left: the large square depicts the eight possible partition bins, while the smaller squares illustrate how the triplets (crosses) may distribute and CPs be constructed to capture the distribution pattern. In a nutshell, a CP asserts or denies that all or some of the members of one group (the subject) possess the attributes of another group (the predicate), using an expression of the form: ‘Q S are P’, where S refers to the subject, P to the predicate, and Q to a quantification word which can be ‘All’, ‘Some’, or ‘No’ [3]. The expression ‘All S are P’ is a typical example. In our case, we form CPs using  $\alpha, \beta$ , or  $\gamma$  as S and the other two as P, such that the following expression holds: ‘Q (triplets with elements in) S are (in relation with at least one element from) P’. For simplicity, we omit the words in parenthesis. In Fig. 1-Left we notice that all the triplets with elements in  $\alpha$  also have elements in  $\beta$  and  $\gamma$ , thus forming: ‘All  $\alpha$  are  $\beta$  and  $\gamma$ ’. These are informative patterns: if  $\alpha$  represents customers,  $\beta$  products and  $\gamma$  stores, then ‘All  $\alpha$  are  $\beta$  and  $\gamma$ ’ indicates that customers in  $\alpha$  buy only products from  $\beta$  and only in stores from  $\gamma$ . It is thus of interest to list informative CPs. We notice that (i) universal quantifiers (All, No) are more

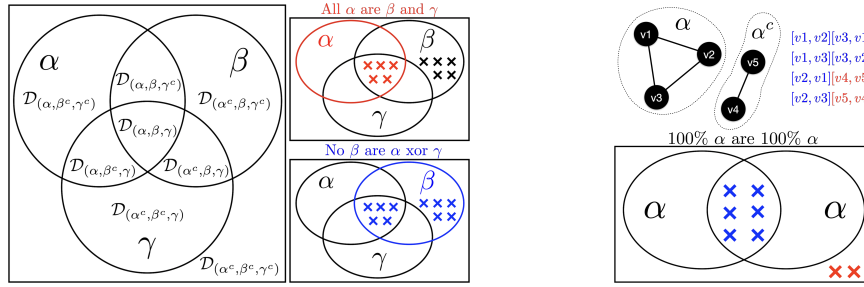


Fig. 1: **Left:** Partition bins (big square) and distribution of  $\mathcal{D}$  (crosses) into the bins with associated categorical propositions (small squares). **Right:** Framework applied to a graph composed of two clique components.

informative than particular ones (Some), yet particular propositions may be close to a universal one; and (ii) propositions above do not express how dense are the relations between S and P. We therefore extend propositions to:  $x\%$  S are  $y\%$  P, where  $x$  is the fraction of triplets in S in relation to P and  $y$  is the density of relationships between S and P. This allows us to state the algorithmic challenge of listing all propositions satisfying constraints on  $x$  and  $y$  without needing to explore the full space.

**Relation to graph theory and extensions.** Our formalism can also be used to study tuples  $(a, b)$ . By setting  $A = B = \mathcal{V}$  we address the particular case of graphs, where  $\mathcal{V}$  is to the vertex set of graph  $\mathcal{G}$ . Several concepts from graph theory may be re-formulated in terms of propositions satisfying specific constraints. An illustration is given in Fig. 1-Right. Going further, we use this re-formulation to generalize the concepts to data triplets. Our results are listed in Table 1. It can be seen that some patterns, like XOR predicates, may not be easily derived from pure graph extensions.

**Listing propositions.** We propose Algorithm 1 to list propositions of type  $x \geq x_{min}\%$   $\alpha$  are  $y \geq y_{min}\%$   $\beta$  and  $\gamma$ , where  $x_{min}$  and  $y_{min}$  are user-defined parameters. It uses the fact that disjoint subjects satisfying a predicate also satisfy it when merged. Thus, the algorithm searches valid predicates for singleton subjects and merges all those sharing predicates. We find predicates via a constructive approach where each triplet forms a region iteratively grown until the constraints are no longer satisfied. While this approach does not in general retrieve all propositions, it identifies a significant number of non-trivial patterns, and it may be improved in further work.

**Application to real-world data.** We apply Algorithm 1 to a contact network in a hospital [4]. Sets A = B consist of 29 patients and 46 healthcare workers (27 nurses, 11 doctors, 8 admin), set C represents time (1890 minutes of data). We use time as subject set and  $x_{min} = 0.7, y_{min} = 0.5$ . The algorithm finds 1456 predicates from which it forms patterns like: (i) group of 7 minutes where 85% activity corresponds to 3 nurses and 1 admin interacting with 64% density; (ii) group of 3 minutes where 84% activity corresponds to 4 doctors interacting with 66% density; (iii) group of 7 minutes where 80% activity corresponds to 3 nurses interacting with 66% density. Clearly, the patterns found are representative of the typical activity in a hospital.

**Acknowledgements.** This work is funded in part by the ANR (French National Agency of Research) under the Limass (ANR-19-CE23-0010) and FiT LabCom grants.

## References

1. M. Kivela et al., Multilayer networks, Journal of Complex Networks, Vol. 2 (2014).

2. L. Cerf et al. “Closed patterns meet n-ary relations” ACM Transactions on Knowledge Discovery from Data, (2009).
3. I. Copi and C. Cohen. Introduction to Logic. N.J: Pearson/Prentice Hall, 2005.
4. P. Vanhems et al., Estimating Potential Infection Transmission Routes in Hospital Wards Using Wearable Proximity Sensors, PLoS ONE 8(9): e73970 (2013).

Concept	Proposition in graphs	Extension
Disconnected network	$\mathcal{G}$ is disconnected if there exists $\alpha \in \mathcal{V}$ satisfying: No $\alpha$ are $\alpha^c$	$\mathcal{D}$ is disconnected if there exist $\alpha \in A, \beta \in B, \gamma \in C$ satisfying: No $\alpha$ are $\beta^c$ or $\gamma^c$ No $\beta$ are $\alpha^c$ or $\beta^c$ No $\gamma$ are $\alpha^c$ or $\beta^c$ No $\alpha^c$ are $\beta$ or $\gamma$ No $\beta^c$ are $\alpha$ or $\gamma$ No $\gamma^c$ are $\alpha$ or $\beta$
Vertex cover	$\alpha \subseteq \mathcal{V}$ is a cover of $\mathcal{G}$ if it is satisfied: No $\alpha^c$ are $\alpha^c$	$\alpha \subseteq A, \beta \subseteq B,$ and $\gamma \subseteq C$ are a cover of $\mathcal{D}$ if it is satisfied: No $\alpha^c$ are $\beta^c$ and $\gamma^c$ No $\gamma^c$ are $\alpha^c$ and $\beta^c$ No $\beta^c$ are $\alpha^c$ and $\beta^c$
Dominating set	$\alpha \subseteq \mathcal{V}$ is a dominating set of $\mathcal{G}$ if it is satisfied: Some $a_i$ are $\alpha, \forall a_i \in \alpha^c$	$\alpha \subseteq A, \beta \subseteq B,$ and $\gamma \subseteq C$ are dominating sets of $\mathcal{D}$ if it is satisfied: Some $a_i$ are $\beta$ or $\gamma, \forall a_i \in \alpha^c$ Some $c_i$ are $\alpha$ or $\beta, \forall c_i \in \gamma^c$ Some $b_i$ are $\alpha$ or $\gamma, \forall b_i \in \beta^c$
Separating set	$\alpha \subseteq \mathcal{V}$ is a separating set of $\mathcal{G}$ if it is satisfied that: $\mathcal{G}_{(\alpha^c)}$ is disconnected	$\alpha \subseteq A, \beta \subseteq B,$ and $\gamma \subseteq C$ are separating sets of $\mathcal{D}$ if it is satisfied that: $\mathcal{D}_{(\alpha^c, \beta^c, \gamma^c)}$ is disconnected
Vertex k-coloring	A disjoint partitioning $\mathcal{V} = \bigcup_{i=1}^k \alpha_i$ is a k-coloring of $\mathcal{G}$ if it is satisfied: No $\alpha_i$ are $\alpha_j, \forall i$	The disjoint partitionings $A = \bigcup_{i=1}^k \alpha_i, B = \bigcup_{i=1}^k \beta_i, C = \bigcup_{i=1}^k \gamma_i$ are a k-coloring of $\mathcal{D}$ if it is satisfied: No $\alpha_i$ are $\beta_j$ and $\gamma_l, \forall i$ No $\beta_i$ are $\alpha_j$ and $\gamma_l, \forall i$ No $\gamma_i$ are $\alpha_j$ and $\beta_l, \forall i$
Clique	$\alpha \subseteq \mathcal{V}$ is a clique of $\mathcal{G}$ if it satisfies: Some $\alpha$ are 100% $\alpha$	$\alpha \subseteq A, \beta \subseteq B,$ and $\gamma \subseteq C$ are a clique of $\mathcal{D}$ if they satisfy: Some $\alpha$ are 100% $\beta$ and $\gamma$
Cluster	$\alpha \subseteq \mathcal{V}$ is a cluster of $\mathcal{G}$ if it satisfies: Large x % $\alpha$ are $\alpha$	$\alpha \subseteq A, \beta \subseteq B,$ and $\gamma \subseteq C$ are a cluster of $\mathcal{D}$ if they satisfy: Large x% $\alpha$ are $\beta$ and $\gamma$ Large x% $\gamma$ are $\alpha$ and $\beta$

Table 1: Re-formulation of graph concepts in terms of categorical propositions and extension to data triplets. It can be seen that graph concepts relying on partitioning  $\mathcal{V}$  easily carry over to arbitrary data triplets.

#### Algorithm 1

```

1: procedure GET_PROPOSITIONS( $\mathcal{D}, x_{min}, y_{min}$ )
2:   predicateList = [ ]
3:   propositionList = [ ]
4:   for  $a_i \in A$  do
5:     predicateList.append ( get_predicates(  $\mathcal{D}_{(a_i, B, C)}, x_{min}, y_{min}$  ) )
6:   end for
7:   for  $P \in$  predicateList do
8:      $S \leftarrow$  merge_subjects(  $P,$  predicateList ) ▷ Combine all subjects ( $a_i$ ) with predicate  $P$ 
9:     propositionList.append(  $(S, P)$  )
10:  end for
11: end procedure ( return propositionList )
12: procedure GET_PREDICATES( $\mathcal{D}_{(a_i, B, C)}, x_{min}, y_{min}$ )
13:   regionList =  $\mathcal{D}_{(a_i, B, C)}$  ▷ Each triplet ( $a_i, b_j, c_k$ ) becomes a region ( $b_j = \beta, c_k = \gamma$ )
14:   predicates = [ ]
15:   while regionList do
16:     grownRegionList  $\leftarrow$  grow_regions( regionList ) ▷ For each region, grow  $\beta$  or  $\gamma$  in dir. of max. density
17:     regionList  $\leftarrow$  filter_y( grownRegionList ) ▷ Remove regions with density less than  $y_{min}$ 
18:     predicates.append( filter_x( regionList ) ) ▷ Store regions if at least  $x_{min}$  fraction of triplets fall within
19:   end while
20: end procedure ( return predicates ) ▷ These predicates satisfy  $x \geq x_{min}$   $a_i$  are  $y \geq y_{min}$   $P$ 

```