UNIVERSITÉ NICE SOPHIA ANTIPOLIS

UNIVERSITÀ DEGLI STUDI DELL'AQUILA

Erasmus Mundus Consortium "MathMods"

Double Master's Degree Programme in
Mathematical Modelling in Engineering: Theory, Numerics, Applications

Master Sciences, Technologies, Santé à finalité Recherche Mention Mathématiques et Interactions Spécialité Mathématiques Pures et Appliquées	Laurea Magistrale in Ingegneria Matematica
UNIVERSITÉ NICE SOPHIA ANTIPOLIS	UNIVERSITÀ DEGLI STUDI DELL'AQUILA

In the framework of the
Consortium Agreement and Award of a Joint/Multiple Degree 2013-2019

Master's thesis

Aggregation of Link Streams for Multiscale Analysis of Dynamical Interactions

CANDIDATE:
Hindol Rakshit
(*matricola: 239090*)

SUPERVISORS:
Prof. Matthieu Latapy
Dr. Robin Lamarche-Perrin
Ms. Tiphaine Viard
Complex Networks Team
Laboratoire d'Informatique de Paris 6, UPMC - CNRS

2014/2015

Contents

Extended Abstract

A SIGNIFICANT CHALLENGE in dynamical network research is the accurate mathematical description of spatiotemporal *events* and detectable *communities* from an ***inferential*** as well as a ***compressional*** perspective. The *inferential* perspective allows us to answer particular macro-scale questions about the dataset; for example, the evolution of particular research communities by analyzing dynamical co-citation networks, the detection of groups of close knit friends by analyzing online social interactions over a period of time, or even the detection of evolving protein interaction modules by analyzing protein interaction networks. Whereas, the *compressional* perspective comes from the realization that real world dynamical networks typically model millions, or even billions of interactions, and handling them computationally is a challenge in itself.

Traditional graph theoretical approaches with discrete time variation, where data is modeled as a temporal sequence of graphical interactions between a set of nodes, often fail to capture the true dynamics of such data as real world interactions tend to have continuous temporal lengths. Hence it is necessary to consider discussing this problem upon an environment which incorporates continuous time length description with interaction data. This is where the ***link stream*** formalism comes in.

A link stream L is a tuple (T, V, E) which models interactions (E) between a set of nodes (V) over a time interval (T). A link (b, e, u, v) is in E if two nodes u and v in V have interacted from time b to time e $(b, e \in T, b \leq e)$. Link streams model many real-world interactions between individuals, email exchanges, or network traffic in continuous time. We use this formalism to describe spatiotemporal *modules*, *i.e.,* groups of nodes with same/equivalent interaction behavior over a time interval. These *spatiotemporal* modules have dual functionality: describing spatiotemporal events in the dataset (*inferential perspective*) and compressing the original link stream into a much smaller, yet meaningful representation (*compressional perspective*)

This thesis proposes a mathematical framework to deal with modular decomposition and subsequent compression of link streams. We start by generalizing the definition of modules in graphs to link streams. In a graph, a module M is a subset of V such that all pairs of nodes in M have the same neighborhood. Analogously, we define a module in a link stream as a couple (M, T^*) consisting of a set of nodes $M \subseteq V$ and a time interval $T^* \subseteq T$ such that all nodes in M share the same neighborhood over T^*.

These *spatiotemporal* modules can be used to achieve a *lossless* compression of the original link stream. However, it might also be interesting to look at *relaxed* versions of module structures, *i.e.,* communities where nodes have *approximately* similar neighborhood structure over a fixed time interval, as these kinds of modules capture significant macro-scale similarity features in real-world interaction data by ignoring insignificant micro-scale variations. Hence, we reduce the stringency of

module definition to achieve more meaningful (yet lossy) module decomposition of link streams. Intuitively, a relaxed module (M, T^*) is such that the neighborhood similarity of any pair of nodes in M over T^* is higher than a given threshold ϵ (first precision parameter). We quantify this idea by extending the well known *Jaccard similarity index* for spatiotemporal neighborhoods. We also take into consideration intra-modular linkage structure, and quantify the idea of having either a *sparse* or a *dense* module, based on whether the linkage density of the module induced sub stream is either lower than a threshold η or higher than $1 - \eta$ (second precision parameter). Hence the spatiotemporal modules in our study are depicted as ϵ, η-*relaxed modules*.

Following module decomposition, the link stream is compressed (*i.e.,* modules are depicted as "supernodes" and modular interactions as "superlinks"). This compression is achieved using the minimum possible number of modules such that the loss of information between the original and the decompressed link stream is minimal. This is achieved by proposing an optimizable *objective function* which is interpreted as a trade-off between the ***divergence*** and the ***complexity*** of the system.

The *divergence* term quantifies the loss of information between the original and the decompressed link stream, where the decompressed link stream is achieved by decoupling the modules to represent all the nodes and temporal interactions between them. The divergence term in our study has been proposed as the Kullback-Leibler (KL) divergence between the empirical weight distribution of the original and the decompressed link stream. Hence, the best case scenario is when the divergence is closer to zero, *i.e.,* minimal difference between the original and the decompressed weight distribution.

The *complexity* term can be interpreted as the data miner's "resource" constraint. Our interpretation of complexity is simply the number of modules used to describe the complete link stream. Hence, we quantify complexity in our study as the cardinality of the set of modules.

It is interesting to note that the precision parameters ϵ and η control the information loss with continuous variation. E.g., high ϵ and η produce more stringent modules (leading to *overfitting*), whereas low ϵ and η produce larger over-simplified modules (leading to *underfitting*). Hence we can use ϵ and η to convert the module decomposition based compression problem into a parametrized optimization problem: *"Select an upper bound to the complexity level in such a way that the divergence is minimized"*, and vice versa (the dual problem). Ultimately, the goal of this thesis is to describe the well-posedness of this optimization problem, so that efficient algorithms can be developed for multiscale analysis of real world link streams using modular decomposition.

* * *

CHAPTER 1

Motivation

1.1 Context

WITH THE ADVENT of modern computational tools, massive sets of spatiotemporal interaction data are arising in various domains of application, ranging from bioinformatics to online social networks. Storing, processing, learning on and inferring from such abundant data requires innovative mathematical and computational methods. For almost a decade, interaction data across different domains of interest has been widely studied and analyzed using the theory of complex networks [1-3]. A plethora of measures have been proposed to deal with network complexity [4]. Most of these measures try to identify groups of nodes having similar interaction behavior, *e.g.,* groups of nodes which are *densely* connected to each other, or groups of nodes which are connected to similar nodes in the rest of the network [5, 6]. Such groups of nodes can correspond to several meaningful domain specific *community* structures, ranging from protein complexes in the human interactome to research communities in co-citation networks [4].

One major problem with most existing network complexity research in the last decade is the inadequate concentration on the *temporal* aspect of real world networks [7]. Even though a lot of researchers have started to acknowledge for some time that most real world networks are *dynamic* in nature [8], *i.e.,* interactions between nodes can have time signatures (either time instances or intervals). Capturing these dynamics is of utmost importance for meaningful representations of evolving community structures in real world dynamical networks.

Several approaches have been proposed to tackle with temporal dynamics of real networks [7] based on a strong foundation in graph theory. The most common approach relies on analyzing series of graph snapshots: given a time window Δ, one considers a graph $G_t = (V_t, E_t)$ and changes in the interaction patterns induced by a constant change in time from t to $t+\Delta$, which generates a series of time constrained graphs $G_t, G_{t+\Delta}, G_{t+2\Delta}$, and so on [7, 10, 11].

Two significant problems with this approach are, **(1)** choosing the exact value of Δ: small Δ produces redundant snapshots, whereas, large Δ oversimplifies the underlying dynamics and losses microscale variations, though being computationally

efficient to deal with; and **(2)** depending on the internal dynamics of the system, the value of Δ might change over time as well as space, hence choosing a fixed value of Δ might give rise to significant loss of inter-snapshot dynamics. As a matter of fact, a lot of work has been done in designing methods to come up with efficient choices for the value of Δ [12]. In addition, some work has also been done in other related fields (*e.g.,* signal processing) in *selective* aggregation: *i.e.,* aggregating spatiotemporal data in space to observe the temporal evolution, and vice versa. However, it is undeniable that considering a two-dimensional (jointly in space and time) approach for evolving interactions provide the exact representation of the dynamic nature of real world networks. This is where the *link stream* formalism [8, 10] comes in.

Link stream theory (presented in detail in **Chapter 2**) is one of the few frameworks which captures the dynamics of an interacting system by modeling interactions between a set of nodes in such a way that each interaction is signed with a continuous duration of time (*e.g.,* node a and node b interact from time t_1 to time t_2), as opposed to some other versions of dynamical networks which have been studied over the last few years[7]. We considered this formalism as the fundamental framework for this thesis due to its joint consideration of spatiotemporality.

Based on the link stream framework, the question we ask is: how to give accurate mathematical description of spatiotemporal *events* and detectable *communities* from an ***inferential*** as well as a ***compressional*** perspective.

The inferential perspective allows us to detect the evolution of macro-scale events with time, *e.g.,* detecting groups of friends in online social link streams, or, detecting social events by looking into intra-office email link streams. Analogous work has been done in the field of static networks in terms of *community detection* [5] and *modular decomposition* [15].

The compressional perspective comes from the realization that real world dynamical networks typically model millions, or even billions of interactions, and handling them computationally is a challenge in itself. Hence it would be much better if we can express smaller groups of nodes with similar functionality as singular *supernodes* and aggregate interactions between different node groups as *superlinks*, then we can achieve a much smaller representation for the same dataset [16].

This thesis work is an attempt to combine these two perspectives for dynamical networks upon the link stream formalism. We have designed a mathematical framework for *spatiotemporal* modular decomposition in link streams in a way that the spatiotemporal *modules* (the set of which creates an *exhaustive partition* of the original link stream) can be expressed as *supernodes* and intra-modular interactions can be expressed as *superlinks*. These aggregate structures can be used to **(1)** compress the original link stream in space and time, and, **(2)** give a macro-scale representation of events in the original dataset.

One noteworthy feature of such a compression is that, the compressed link stream gives a lossless representation of the original link stream. Hence, this framework accurately captures micro-scale variations of the dataset . However, we also looked at graph similarity based *parameter relaxation* to capture the macro-scale features

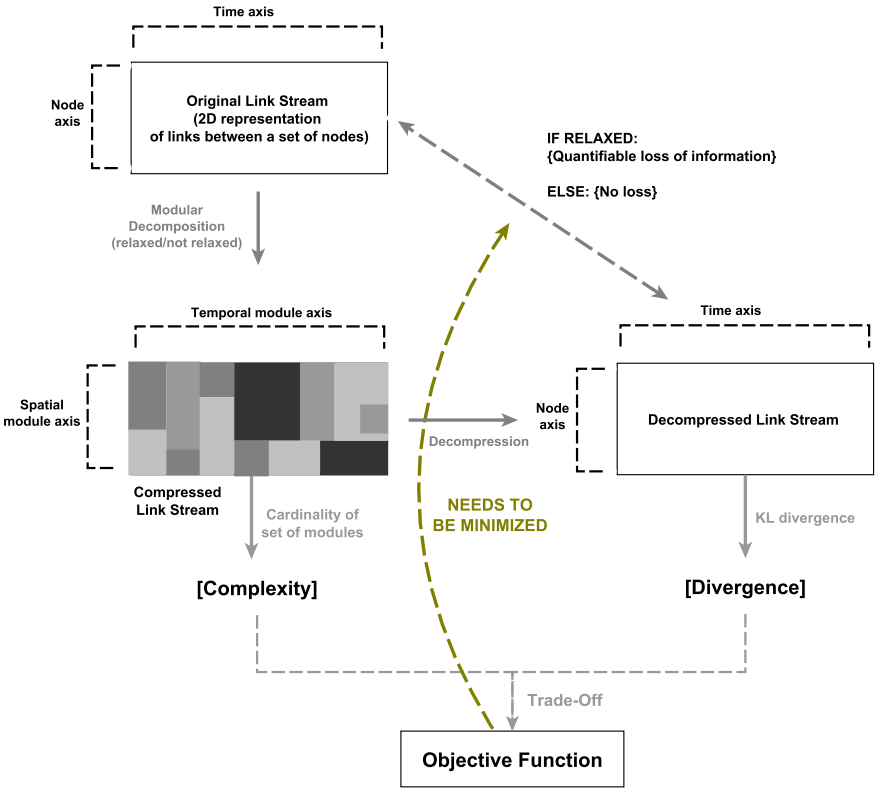


Figure 1.1: **The objective function and its relation to the compression process**

of the dataset at the cost of loosing the less significant micro-scale features (detailed mathematical discussion in **Chapter 3**).

This aggregate representation gives us a (lossy) compressed version of the original link stream. Now, it is quite useful to the data miner if he/she can minimize this loss of information, without loosing the macro-scale features.

Hence we proposed an objective function which needs to be optimized to achieve the most fruitful compression of the link stream. This objective function is proposed as a trade-off between the ***divergence*** and the ***complexity*** of the compressed representation (Figure 1.1). The divergence term quantifies the loss of information between the original and the decompressed link stream, where the decompressed link stream is achieved by removal (dis-aggregation) of the module boundaries to represent all the nodes and temporal interactions between them. The divergence term in our study has been proposed as the Kullback-Leibler (KL) divergence between the empirical weight distribution of the original and the decompressed link stream. And the complexity term, also interpreted as the data miner's "resource" constraint, is interpreted as the number of modules used to describe the compressed link stream. Hence, we quantify complexity in our study as the cardinality of the set of modules (detailed mathematical discussion in **Chapter 4**).

1.2 Relevant Previous Work

THE FIELD OF dynamic graph compression is quite new within the growing field of complex networks research. Recent and most relevant work on dynamic graph theory have focused on dynamical community detection [17, 18] and clustering [19]. There has also been some work in graph compression in the static case [16, 20, 21] and in the dynamic case [22 23] which gave us a good environment to base our work upon; albeit not sufficient as these works were based on series of graph snapshots, whereas our intended framework incorporated continuous time description of graphical interactions. That is why we based our work more on **(1)** state-of-the-art methods for modular decomposition in case of static networks, and **(2)** for information theoretic compression in static networks; and then we tried to combine these two approaches and generalize for link streams to take care of the spatiotemporality.

The idea of *strong* and *relaxed* modular decomposition in the static case was thoroughly discussed in 2007 by J. Reichardt and D. R. White [24], which in turn was based on F. Lorrain and H.C. White's seminal work on *structural equivalence* in social networks [25] and Doreian *et al.*'s work on *generalized blockmodeling* [26]. In addition, the idea of information theoretic compression in static networks has recently been dealt with in Lamarche-Perrin *et al.* [21]. Hence, these two papers ([21] and [24]), originating from static network research, formed the basis of our work.

The chief portion of this thesis work involves the spatiotemporal generalization of these two concepts (strong/relaxed modular decomposition and information theoretic compression) for a link stream framework. In addition, we introduce *parametrized relaxation* for modular decomposition which gives us additional control over the information loss. Ultimately, we discuss (in **Chapter 5**) how to control graph theoretical parameters to achieve optimal compression (min loss, max representation) for a generic link stream (no real database considered) from an inferential perspective.

1.3 Thesis Outline

CHAPTER 2 IN this thesis introduces the link stream framework in detail. First we discuss the pre-existing structure, and then we discuss the new notations and definitions we introduced to deal with our problem. The pre-existing link stream framework describes unweighted links only. Hence following that we introduce weight of interactions in a link stream. Subsequently we show that the empirical weight distribution in a link stream follows a hybrid (*discrete* in space, *continuous* in time) probability distribution. **Chapter 3** introduces spatiotemporal modules in a link stream and discusses strong and relaxed cases in detail. **Chapter 4** explores the compression and decompression problem in link streams. **Chapter 5** proposes the objective function to be optimized and explores its well-posedness for a generic link stream.

* * *

CHAPTER 2

Introduction To Link Streams

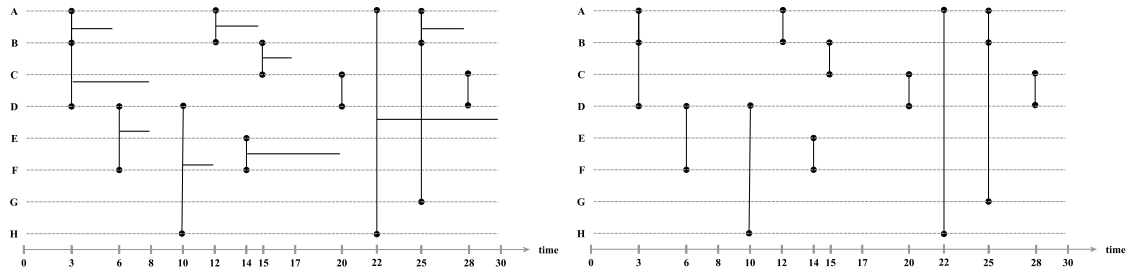
2.1 Context

INTERACTION DATA ARISING in different domains has traditionally been modeled using a graph, *i.e.,* a couple $G = (V, E)$ where V is the set of nodes and $E \subseteq V \times V$ is the set of links. A link $(u, v) \in E$ means that the nodes u and v interact with each other in the concerned model. As mentioned in **Chapter 1**, this simplistic model generates a mathematical language which is used extensively to study interaction models across all scientific, economic and sociological disciplines. However, this model does take into account temporal signatures of individual interactions. This is what the link stream formalism overcomes by generating a mathematical language to study evolving interactions over time. This formalism has been under rapid development for a few years now from a theoretical as well as an applied perspective. This chapter gives a complete introduction to the pre-existing link stream framework, coupled with a new set of notations and definitions that we devised to tackle the problem of spatiotemporal compression in a link stream. The new notations and definitions introduced are embedded as text boxes.

2.2 Basic Structure

A LINK STREAM [8, 10] is a triple $L = (T, V, E)$ where $V = \{v_1, v_2, \cdots, v_n\}$ is a set of nodes ($|V| = n$), $T = [\alpha, \omega]$ is the time span of the stream, and $E \subseteq \{(b, e, v_i, v_j) \in T \times T \times V \times V : b \leq e\}$ gives the set of temporal links in L (Figure 2.1a). Hence, a link $l = (b, e, v_i, v_j)$ is in E if the nodes v_i and v_j have continuously interacted from time b to time e. A special case of a general link stream is an *instantaneous* link stream: *i.e.,* a link stream $L = (T, V, E)$ such that for all $(b, e, v_i, v_j) \in E$, $b = e$ (Figure 2.1b).

The total number of nodes $|V| = n$ is called the *order* of L, the total number of links $|E| = m$ is denoted as the *size* of L, and the range of T, *i.e.,* $\omega - \alpha$ gives the *duration* of L, and is denoted by $\bar{L}$. The duration of any link $l = (b, e, v_i, v_j) \in E$ is given as $\bar{l} = e - b$.



(a) An example of a **general link stream.** (b) An example of an **instantaneous link stream.**

Figure 2.1: **Two different examples of link streams.** Both link streams contain 8 nodes and 12 links, and are represented as $L = (T, V, E)$ where $T = [1, 30]$ and $V = \{A, B, C, D, E, F, G, H\}$. In the general case, $E = \{(3, 6, A, B), (3, 6, B, D), (6, 8, D, F), ...\}$ whereas, in the instantaneous case, $E = \{(3, A, B), (3, B, D), (6, D, F), ...\}$. Each node is represented as a dotted horizontal line (node levels). Each link is represented by a line between two node levels corresponding to individual time stamps in the time axis (instantaneous case). The horizontal lines in the middle of each link show the duration of each link.

In addition to the duration of a link, the duration of a time interval $T^* = [\alpha^*, \omega^*] \subseteq T$ is given by $\langle T^* \rangle = max(T^*) - min(T^*) = \omega^* - \alpha^*$ We use this angular bracket notation to denote the *size* of any continuous set in this thesis.

The link streams considered in our study are *undirected*: *i.e.,* we make no distinction between the links (b, e, v_i, v_j) and (b, e, v_j, v_i). Moreover, the link streams were considered to be *simple*: *i.e.,* $u \neq v$ and $b \leq e$ for all $(b, e, v_i, v_j) \in E$, and $[b, e] \cap [b', e'] = \emptyset$ for all $(b, e, v_i, v_j), (b', e', v_i, v_j) \in E$. For any $v_i, v_j \in V$ and $t \in T$, we say that v_i and v_j interact in L at time t if there exists a link $(b, e, v_i, v_j) \in E$ such that $t \in [b, e]$.

Given two link streams $L = (T, V, E)$ and $L' = (T', V', E')$, we say that L' is a *sub-stream* of L if $V' \subseteq V$ and $T' \subseteq T$ and for all $v_i, v_j \in V'$ and $t \in T'$, if v_i and v_j interact at time t in L', then they also interact at time t in L. This is denoted by the expression $L' \subseteq L$, and $L' = L$ if and only if $L' \subseteq L$ and $L \subseteq L'$. Given two links $l = (b, e, v_i, v_j)$ and $l' = (b', e', v'_i, v'_j)$, we say that l' is a *sub-link* of l if $v'_i = v_i,\ v'_j = v_j$ and $[b', e'] \subseteq [b, e]$. It is noticeable that if $L = (T, V, E)$ and $L' = (T', V', E')$ are simple link streams, then $L' \subseteq L$ if and only if for all $l' \in E'$, there exists an $l \in E$ such that $l' \subseteq l$. E.g., in Figure 2.1a, a link stream $L' = (T', V', E')$ where $T = [0, 8]$, $V = \{A, B, D\}$ and $E = \{(3, 6, A, B), (3, 8, B, D)\}$ is a sub-stream of L as $T' \subset T$, $V' \subset V$ and $E' \subset E$. Similarly we can also find sub-streams in the instantaneous link stream (Figure 2.1b) by taking spatiotemporal subsets of the original link stream.

A sub-stream of L *induced* by a set of nodes $M \subseteq V$ and a time interval $T^* \subseteq T$ is given by $L[(M, T^*)] := (T^*, M, E^*)$ where $E^* = \{(b', e', v_i, v_j) \in T^* \times T^* \times M \times M : b' \leq e'\} \subseteq E$.

2.3 Basic Statistical Measures

THIS SUB-SECTION INTRODUCES several statistical measures defined on a link stream. As this is completely new concept and fundamental to our work, we are going to define the measures while drawing analogies to equivalent measures in a static graph [27]. In case of a static graph $G = (V, E)$, density of G is the probability that when one randomly selects two nodes u and v then there is a link (u, v) in E:

$$\delta(G) = \frac{2m}{n(n-1)} \tag{2.1}$$

Where $n = |V|$ denotes the number of nodes and $m = |E|$ denotes the number of links in G. Analogously in a link stream, the density $\delta(L)$ is defined as the probability of finding an interaction while taking two random nodes u and v and a random time instant t (*i.e.,* there exists a link $(b, e, u, v) \in E$ such that $t \in [b, e]$). This density is only defined for link streams such that $n \geq 2$ and $\omega > \alpha$.

$$\delta(L) = \frac{2\sum_{l \in E} \bar{l}}{n(n-1)(\omega - \alpha)} \tag{2.2}$$

In the example presented in Figure 2.1a, $|V| = 8$, $\omega - \alpha = 30$, and $\sum_{l \in E} \bar{l} = 3 + 5 + 2 + ... = 34$. Therefore,

$$\delta(L) = \frac{2 \cdot 34}{8(8-1) \cdot 30} = \frac{68}{1680} \approx 0.04.$$

The considered link stream is sparse, as the density is closer to 0.

We can notice from Figures 2.1a and 2.1b and that a link stream has a two-dimensional representation. Hence it is interesting to see the *area* covered by a link stream and a sub-stream, and the fractional area of a link stream covered by a sub-stream.

The area of a link stream (T, V, E) is given by $A(L) = |V|\langle T \rangle$. Therefore, in case of both the link streams depicted in figures 2.1a and 2.1b, $A(L) = |V|\langle T \rangle = 8 \cdot 30 = 240$.

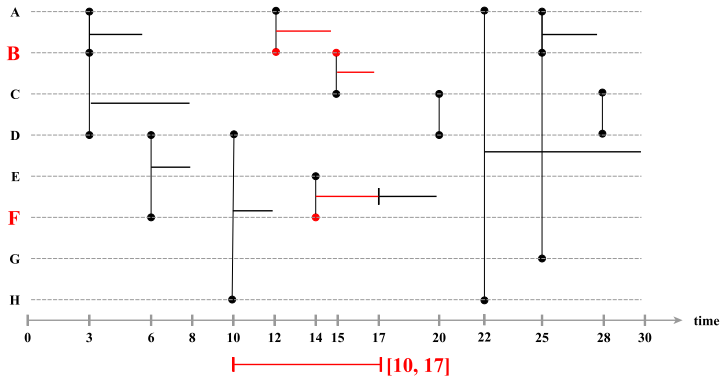
Hence $|V| = n$ and $T = [\alpha, \omega]$ implies that $A(L) = n(\omega - \alpha)$. Similarly, the area of a sub-stream $L(M, T^*)$ (as defined in the previous section) is given by $A(L(M, T^*)) = |M|\langle T^* \rangle$. Therefore, the fractional area of L covered by $L(M, T^*) = L'$ (say) is defined as:

$$\bar{A}_L(L') = \frac{A(L')}{A(L)} \in [0, 1]. \tag{2.3}$$

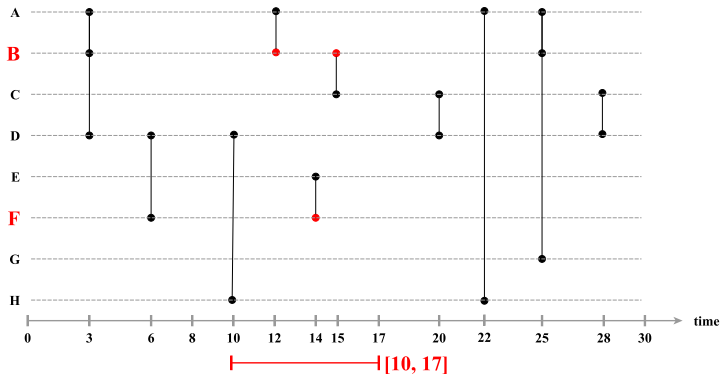
This equation is valid under the assumption that $A(L) \neq 0$, which is only possible if L is not a null link stream (*i.e.,* $L = (T, V, E) \neq (\emptyset, \emptyset, \emptyset)$). Therefore $\bar{A}_L(L') = 0$ denotes that L' is a null link stream, whereas, $\bar{A}_L(L') = 1$ denotes $L' = L$.

Going back to our previous example of sub-streams (**Section 2.2**), if we choose $L' = (T', V', E')$ to be our sub-stream where $T = [0, 8]$, $V = \{A, B, D\}$ and $E = \{(3, 6, A, B), (3, 8, B, D)\}$, then the area of the sub-stream L' is $A(L') = 3 \cdot 8 = 24$, therefore $\bar{A}_L(L') = 24/240 = 0.1$.

In graph G, the neighborhood $N(v)$ of a node $v \in V$ is the set of nodes linked to v: $N(v) = \{u \in V : \exists (u, v) \in E\}$ and the degree of v is defined as $d(v) = |N(v)|$,



(a) Spatiotemporal neighborhood of nodes B and F in a general link stream.



(b) Spatiotemporal neighborhood of nodes B and F in an instantaneous link stream.

Figure 2.2: **Spatiotemporal neighborhoods in link streams.** Here we choose $T' = [10, 17] \subset T$. Hence in **(a)**, $N_{T'}(B) = \{(A, t_1)_{t_1 \in [12,15]}, (C, t_2)_{t_2 \in [15,17]}\}$ and $N_{T'}(F) = \{(E, t_3)_{t_3 \in [14,17]}\}$. Whereas, in **(b)**, $N_{T'}(B) = \{(A, 12), (C, 15)\}$ and $N_{T'}(F) = \{(E, 14)\}$.

where $|...|$ denotes the cardinality of any discrete neighborhood set. The average degree of G is gives as $d(G) = \frac{1}{n} \sum_{v \in V} d(v)$. It can also be noticed that the following relation between density (**equation 2.1**) and average degree holds: $\delta(G) = \frac{d(G)}{n-1}$.

Generalizing this idea in case of a link stream requires a bit more rigor as individual interactions in a link stream have spatial (nodes connected to) as well as temporal (duration of each interaction) signatures.

The spatiotemporal neighborhood $N_{T^*}(v_i)$ of a node $v_i \in V$ within the interval $T^* \subseteq T$, is defined as $N_{T^*}(v_i) = \{(v_j, t) : \exists (b, e, v_i, v_j) \in E, t \in [b, e] \cap T^*\}$ (Figures 2.2a and 2.2b). Accordingly, the *size* of $N_{T^*}(v_i)$, or the *degree* of v_i in T^* is defined as:

$$d_{T^*}(v_i) = |N_{T^*}(v_i)| = \sum_{(b,e,v_i,v_j) \in E} \frac{\langle [b, e] \cap T^* \rangle}{\langle T^* \rangle} \tag{2.4}$$

As we choose a subset of T as the considered time interval while defining $N_{T^*}(v_i)$, we can also restrict this definition *spatially* to denote that we only choose to inquire about the nodes linked to $v_i \in M \subseteq V$ over T^* outside M: $N^M_{T^*}(v_i) = \{(v_j, t) \in N_{T^*}(v_i) : v_i \in V \backslash M\}$. Therefore the size of $N^M_{T^*}(v_i)$ is defined as:

$$|N^M_{T^*}(v_i)| = \sum_{\substack{(b,e,v_i,v_j)\in E \\ v_j \in V\backslash M}} \frac{\langle [b,e] \cap T^* \rangle}{\langle T^* \rangle} \tag{2.5}$$

Considering a single time instant t instead of an interval T^* reduces the spatiotemporal neighborhood framework to a spatial (discrete) framework, returning the set of nodes linked to v_i at a time instant $t \in T$: $N_t(v_i) = \{v_j \in V : \exists (b, e, v_i, v_j) \in E, t \in [b, e]\}$. Correspondingly, the degree of v_i at time t is the size of $N_t(v_i)$, *i.e.,* $d_t(v_i) = |N_t(v_i)|$. By extension, the degree of v_i in L is defined as:

$$d_T(v_i) \equiv d(v_i) := |\bigcup_{t\in T} N_t(v_i)| = \sum_{(b,e,v_i,v_j)\in E} \frac{e-b}{\omega - \alpha} = \sum_{l \in L(v_i)} \frac{\bar{l}}{\omega - \alpha}. \tag{2.6}$$

where $L(v_i)$ gives the sub-stream of L induced by v_i, i.e., part of the original link stream L only containing links from v_i to other nodes in V (i.e., in $V\backslash\{v_i\}$). Therefore the average degree in L is defined as:

$$d(L) = \frac{1}{n} \sum_{v_i \in V} d(v_i). \tag{2.7}$$

Just like simple graphs, there exists the following relation between the density (**equation 2.2**) and average degree in L:

$$\begin{aligned}
\delta(L) = \frac{2\sum_{l\in E} \bar{l}}{n(n-1)(\omega-\alpha)} &= \frac{2\sum_{l\in E} \frac{\bar{l}}{\omega-\alpha}}{n(n-1)} \\
&= \frac{2\frac{1}{2}\sum_{v\in V}\sum_{l\in L(v)} \frac{\bar{l}}{\omega-\alpha}}{n(n-1)} \\
&= \frac{\sum_{v\in V} d(v)}{n(n-1)} \\
&= \frac{d(L)}{n-1}.
\end{aligned}$$

2.4 Neighborhood Similarity

IN A SIMPLE graph G, two nodes u and v are said to have similar neighborhoods if $N(u) = N(v)$. This is a binary measure which returns 1 only if all the neighbors of u in G are also neighbors of v, and 0 otherwise. However, if we want to understand the *degree* of neighborhood similarity of u and v, *i.e.,* approximate (fractional) similarity of the neighborhoods of u and v based on the number of common neighbors, we would need to use the Jaccard similarity index [28], which is defined for any two sets A, B as:

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} \in [0, 1] \tag{2.8}$$

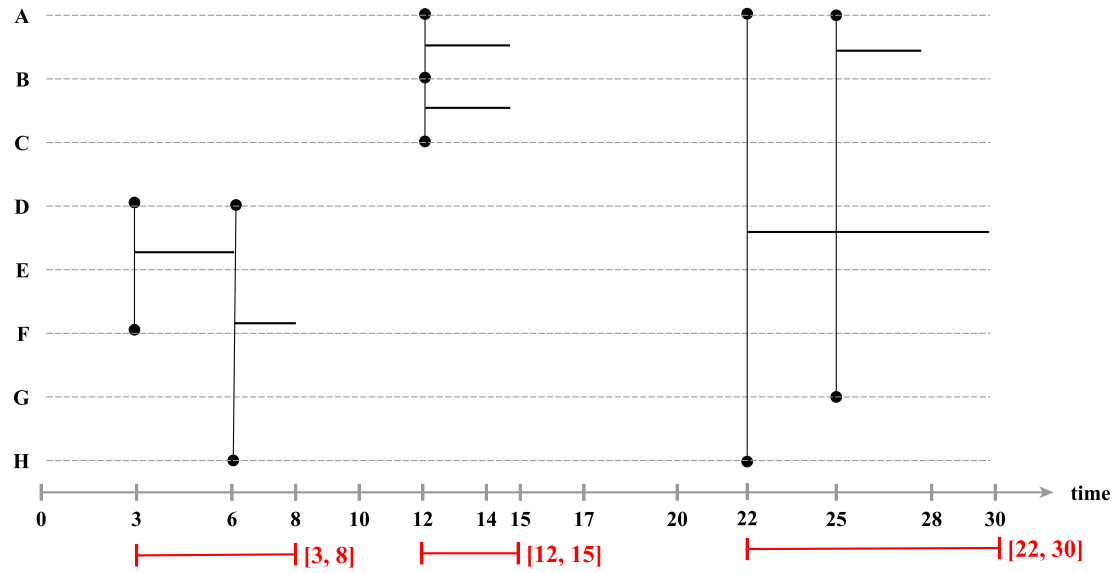


Figure 2.3: **Spatiotemporal neighborhood similarity in a link stream.** Here we show the neighborhood similarity of nodes F and H over the interval $[3, 8]$, nodes A and C over $[12, 15]$, and nodes G and H over $[22, 30]$.

Therefore, the neighborhood similarity for u and v is given as $J(N(u), N(v))$. Correspondingly, if we want to measure the similarity of neighborhoods of u and v outside a given subset of nodes M, then we evaluate $J(N(u)\backslash M, N(v)\backslash M)$. This index works well for discrete sets when the cardinality expression $|...|$ is taken as the cardinality of discrete sets (as taken conventionally), but fails to capture similarity of continuous intervals. Hence even though We use the Jaccard similarity to calculate the similarity between node neighborhoods, we consider the expression $|...|$ defined as in **equation 2.4** and **2.5** for spatiotemporal neighborhood sets and restricted neighborhood sets in L.

The spatiotemporal neighborhood similarity between two nodes v_i and v_j over a time interval $T^* \subseteq T$ in L is defined by the measure $\epsilon_{T^*} : V \times V \to [0, 1]$ as:

$$\epsilon_{T^*}(v_i, v_j) = J(N_{T^*}(v_i), N_{T^*}(v_j)) = \frac{\sum_{(b,e,u,v_i)\in E \cap (b,e,u,v_j)\in E} \langle [b, e] \cap T^* \rangle}{\sum_{(b,e,u,v_i)\in E \cup (b,e,u,v_j)\in E} \langle [b, e] \cap T^* \rangle} \tag{2.9}$$

By extension, the restricted similarity between two nodes $v_i, v_j \in M \subseteq V$ over T^* is defined by the measure $\epsilon^M_{T^*} : M \times M \to [0, 1]$ as:

$$\epsilon^M_{T^*}(v_i, v_j) = J(N^M_{T^*}(v_i), N^M_{T^*}(v_j)). \tag{2.10}$$

Let us now calculate the neighborhood similarity of nodes in the link stream depicted in Figure 2.3 corresponding to several time intervals. Let, $T^*_1 = [3, 8]$, $T^*_2 = [12, 15]$ and $T^*_3 = [22, 30]$. Nodes F and H both interact with node D in T^*_1, but the duration of their interactions do not intersect. Therefore:

$$\epsilon_{T^*_1}(F, H) = \frac{0}{\langle [3, 6] \cap T^*_1 \rangle + \langle [6, 8] \cap T^*_1 \rangle} = 0.$$

Similarly for the nodes A and C over T_2^*:

$$\epsilon_{T_2^*}(A, C) = \frac{\langle [12, 15] \cap T_2^* \rangle + \langle [12, 15] \cap T_2^* \rangle}{\langle [12, 15] \cap T_2^* \rangle + \langle [12, 15] \cap T_2^* \rangle} = 1.$$

and for nodes G and H over T_3^*:

$$\epsilon_{T_3^*}(G, H) = \frac{\langle [25, 28] \cap T_3^* \rangle}{\langle [22, 30] \cap T_3^* \rangle + \langle [25, 28] \cap T_3^* \rangle} = \frac{\langle [25, 28] \rangle}{\langle [22, 30] \rangle + \langle [25, 28] \rangle} = \frac{3}{11} \approx 0.27.$$

Calculations for $\epsilon_{T^*}^M(\cdot, \cdot)$ follows accordingly.

2.5 Conclusion

IN THIS CHAPTER we have introduced the link stream framework, coupled with some new notations and definitions that we have generated to deal with the problem of spatiotemporal compression in dynamical networks. In the next chapter, we are going to introduce the idea of spatiotemporal *aggregates* or *modules* using the presented framework, which we are going use eventually to address our research problem.

* * *

CHAPTER 3

Modules In Link Streams

3.1 Context

BEFORE DEFINING MODULES in link streams, it is good to have a look at where the idea of *spatiotemporal* modules come from. As we discussed before in **Chapter 1 Section 1.2**, the idea of modular decomposition in static graphs was discussed in detail by Reichardt & White [24]. In their work, they mentioned that two nodes are *structurally equivalent* if they have the exact same neighbors, and they are *regularly equivalent* if they are connected in the same way as to equivalent others (Figure 3.1). A structural equivalence module in a graph would be a set of nodes which have the exact same neighbors outside the module.

We initially extended this idea to link streams, by saying that a module in a link stream L (as defined in **Chapter 2**) is a couple (M, T^*), where $M \subseteq V$ is some subset of nodes and $T^* \subseteq T$ is a time interval such that the nodes in M have the exact same neighbors over T^* (Figure 3.2). Then, we refined this idea from two different perspectives to make our framework stronger:

(1) The considered modules are *strong* by construction, *i.e,* two nodes need to have the *exact* same neighbors (both spatially and temporally) outside the module to satisfy the above mentioned idea. We relaxed this stringent criterion by saying that nodes in a module should have *approximately* similar neighbors to be in a module. Therefore, based on this idea alone, the definition of a module is given as:

Definition 1. ***(ϵ-Relaxed Modules)*** *Given a subset of nodes $M \subseteq V$, a finite time interval $T^* \subseteq T$ and a precision parameter $\epsilon \in [0, 1]$, the couple (M, T^*) is called an ϵ-relaxed module, if $\epsilon^M_{T^*}(v_i, v_j) \geq \epsilon$ for all $v_i, v_j \in M$. ($\epsilon^M_{T^*}$ defined as in **eq** 2.10)*

Therefore, an ϵ-relaxed module is *strong* one when $\epsilon = 1$, *i.e.,* $\epsilon^M_{T^*}(v_i, v_j) = 1$, $\forall\ v_i, v_j \in M$. In other words, the strength of a module increases as we slide the precision parameter ϵ closer to 1.

One trivial case of an ϵ-relaxed module is the original link stream L itself. All the nodes in V have zero neighbors outside L, hence they all have similar (trivial)

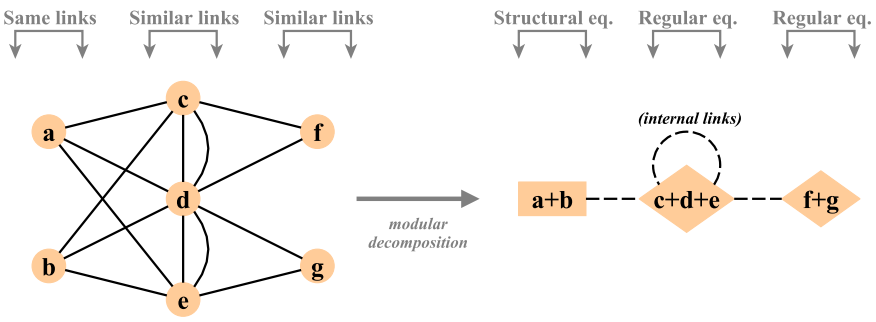


Figure 3.1: **Structural and regular equivalence in a simple graph.** Nodes **a** and **b** have the exact same neighbors (**c**, **d**, **e**), hence they form a *structural equivalence* class. Whereas, node groups (**c**, **d**, **e**) and (**f**, **g**) form two separate *regular equivalence* classes. The figure on the right gives the mirror representation of the graph in terms of the structural and regular equivalence classes as singular nodes.

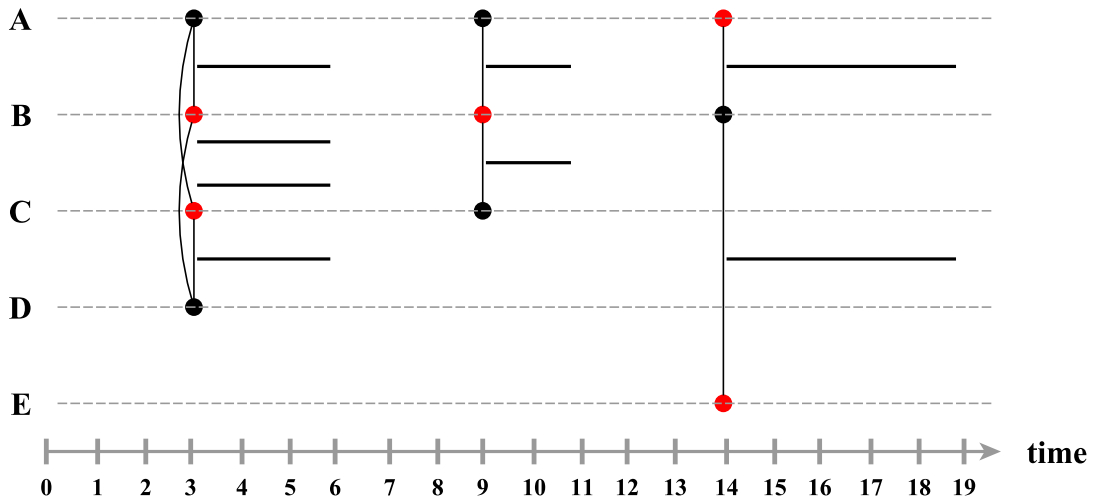


Figure 3.2: **Module formation in a link stream.** Similarity in interaction generates 6 modules of the form (M, T^*) in this link stream: $(\{A, D\}, [3, 6])$, $(\{B, C\}, [3, 6])$, $(\{A, C\}, [9, 11])$, $(\{B\}, [9, 11])$, $(\{A, E\}, [14, 19])$ and $(\{B\}, [14, 19])$. We can also notice some modules with zero interactions, *e.g.,* $(\{A, B, C, D, E\}, [0, 3])$, $(\{A, B, C, D, E\}, [7, 9])$, $(\{E\}, [0, 13])$, $(\{C\}, [11, 19])$ etc.

neighborhood structures with $\epsilon = 1$. Therefore, the original link stream L is a 1-relaxed module. We can also say that for any $M \subseteq V$ and $T^* \subseteq T$, the couple (M, T^*) in L is a (trivial) 1-relaxed module.

(2) In addition to considering approximate neighborhood similarity, it might also be interesting to see if the modules in our link stream are internally sparse (*i.e.,* nodes loosely interact within) or dense (*i.e.,* nodes highly interact within). We might also have interesting modules which are neither internally sparse nor dense, but rather form interaction structures which correspond to meaningful motifs with respect to the data set under investigation. Hence, **Definition 1** is refined accordingly to incorporate the idea of internal structure in the following manner:

Definition 2. ***(ϵ, η-Relaxed Module)*** *An ϵ-relaxed module (M, T^*) is an ϵ, η-relaxed module in L, given another precision parameter $\eta \in [0, 1/2]$ if (M, T^*) generates a sub-stream $L(M, T^*) \subseteq L$ such that either **(C1)** $\delta[L(M, T^*)] < \eta$ or **(C2)** $\delta[L(M, T^*)] > 1 - \eta$.*

*If **C1** is satisfied, then (M, T^*) is a **sparse** module, whereas if **C2** is satisfied, then it is a **dense** module.*

Notice that the idea of having either a sparse or a dense module is directly linked to the chosen value of η, *i.e.,* sparsity/density of a module increases as we slide η closer to 0, and decreases otherwise.

Going back to our previous examples of trivial modules, we can say that L forms a $1, \eta$-relaxed module where $\eta = \delta(L)$. Whereas, every couple $(\{v\}, [t, t])$ in L for all $v \in V$ and $t \in T$ forms a $1, 0$-relaxed module.

Therefore, considering the two above-mentioned perspectives, we can not only make the module framework well-defined, but in addition we can see that the relaxation parameters ϵ and η serve as *control* parameters for module strength, which in turn relates significantly to our objective of achieving an optimal link stream compression. I will discuss this inter-dependence in detail in the following chapters, especially in **Chapter 5**, but for now we need to define some additional properties of modules in our link stream to proceed with our approach to solve the problem of modular decomposition and compression in link streams.

3.2 Module Characterization

AS WE CAN see in Figure (3.3), a module in a link stream has a rectangular (two-dimensional) presentation. It is important for us to have some idea about the physical attributes of a modules, *i.e.,* neighborhood/internal interaction strength of a module, area covered by a module etc., in order to be able to perform comparative analyses. Therefore, in this section, we are going to define several properties of the above described module architecture which will be useful for comparative analyses.

Definition 3. ***(Strength of a Module)*** *The strength of an ϵ, η-relaxed module (M, T^*) is given by the couple $S(M, T^*)$ which is defined as:*

$$S(M, T^*) = (\epsilon(M, T^*), \eta(M, T^*)) \tag{3.1}$$

where $\epsilon(M, T^*)$ is the neighborhood strength of (M, T^*), that is the minimum pairwise restricted neighborhood similarity for all pairs of nodes $v_i, v_j \in M$, i.e.,

$$\epsilon(M, T^*) = \min_{v_i, v_j \in M} \epsilon^M_{T^*}(v_i, v_j) \tag{3.2}$$

And, $\eta(M, T^*)$ is the internal linkage strength of (M, T^*), that is the density of the module induced sub-stream:

$$\eta(M, T^*) = \delta[L(M, T^*)]. \tag{3.3}$$

It is noticeable that for all $v_i, v_j \in M$, $\epsilon^M_{T^*}(v_i, v_j) \leq \epsilon$ as (M, T^*) is already defined as an ϵ, η-relaxed module. Still, we take the minimum of $\epsilon^M_{T^*}(v_i, v_j)$ as $\epsilon(M, T^*)$ instead of taking ϵ directly, as the minimum restricted neighborhood similarity measure can vary for different modules corresponding to a fixed ϵ. But this sort of a specificity is not required for the internal linkage density as $\eta(M, T^*)$ considers all nodes and time instances in the module at the same time. **Definition 3** gives us a proper framework to finally define **strong** modules in a link stream, analogous to the structurally equivalent modules in a static graph.

Definition 4. ***(Strong Modules)*** *An ϵ, η-relaxed module (M, T^*) is called a* ***strong module*** *if either $S(M, T^*) = (1, 0)$ (sparse strong module) or $S(M, T^*) = (1, 1)$ (dense strong module).*

In case of real world link streams, the idea of having a strong/relaxed spatiotemporal module relates to strong/relaxed spatiotemporal *communities*. For example, two (strong or relaxed) modules (M, T^*_1) and (M, T^*_2) formed over the same set of nodes $M \subseteq V$ but different time intervals (*i.e.,* $T^*_1 \cap T^*_2 = \emptyset$) translates to the observation that the nodes in M have similar interaction behavior over T^*_1 and T^*_2. This indicates the temporal significance of the node set M in terms of community formation. Similarly, two modules (M_1, T^*) and (M_2, T^*) formed over two different sets of nodes M_1 and M_2 (*i.e.,* $M_1 \cap M_2 = \emptyset$) but on the same time interval T^* denotes that two different sets of nodes have similar interaction signatures over the same time interval. This indicates the significance of the time interval T^* in terms of multiple spatial events (Figure 3.3). Moreover, the above defined parameter $S(\cdot, \cdot)$ describes the internal as well as neighborhood strength of the modules.

It would be interesting if we can compare the *size* of these constrained modules to determine the scale of macroscopic patterns in our link stream. For example, in case of modules (M, T^*_1) and (M, T^*_2), the larger module is the one with larger temporal span, *i.e.,* $\langle T^*_1 \rangle \geq \langle T^*_2 \rangle$ would indicate that (M, T^*_1) is larger than (M, T^*_2). Similarly, in case of the modules (M_1, T^*) and (M_2, T^*), $|M_1| \leq |M_2|$ would indicate that (M_1, T^*) is smaller than (M_2, T^*). We formalize this idea by extending the no-

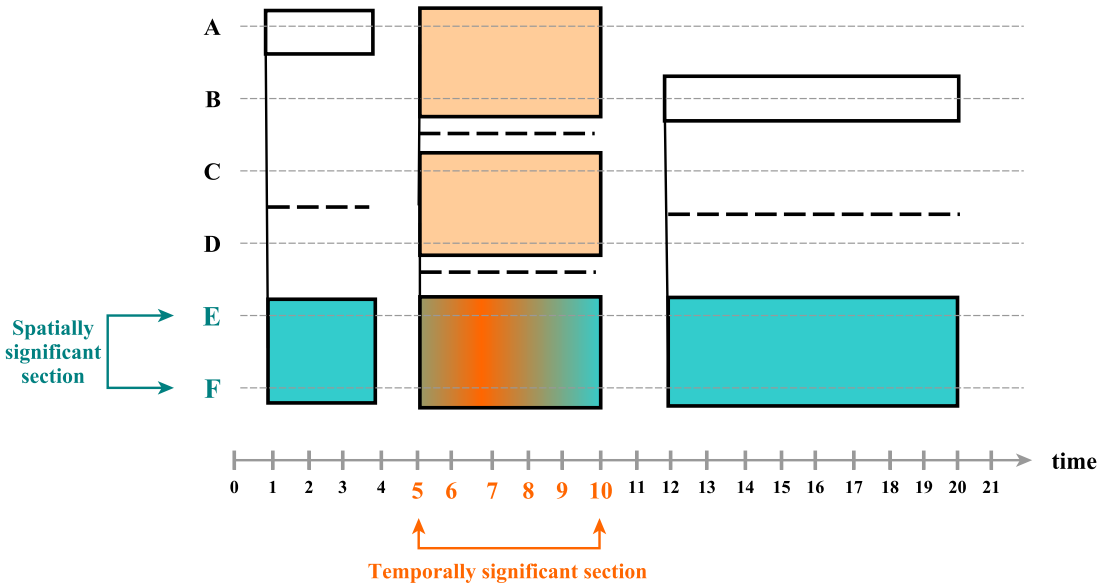


Figure 3.3: **Module representation in a link stream.** Notice that nodes E and F show spatial significance, whereas, time interval $[5, 10]$ shows temporal significance in terms of module formation.

tion of a fractional area of a link stream covered by a sub-stream (equation **2.3**) in order to define the *normalized area* of a module, which returns a ratio between 0 and 1 denoting the area of the link stream covered by the module. This equivalence is made possible due to the observation that any module (or any pair of a node set and a time interval) generates a sub-stream in L. Therefore we can essentially use the same notation to describe the normalized area covered by a module as we do in case of a sub-stream.

Definition 5. ***(Normalized Area of a Module)*** *The normalized area of an ϵ, η-relaxed module (M, T^*) is denoted by the expression $\bar{A}_L(M, T^*)$ which gives the normalized area of the sub-stream generated by (M, T^*):*

$$\bar{A}_L(M, T^*) = \frac{|M|}{|V|} \frac{\langle T^* \rangle}{\langle T \rangle} \tag{3.4}$$

Hence going back to the previous two examples, in case of (M, T_1^*) and (M, T_2^*), $\langle T_1^* \rangle \geq \langle T_2^* \rangle \Rightarrow \bar{A}_L(M, T_1^*) \geq \bar{A}_L(M, T_2^*)$. And in case of (M_1, T^*) and (M_2, T^*), $|M_1| \leq |M_2| \Rightarrow \bar{A}_L(M_1, T^*) \leq \bar{A}_L(M_2, T^*)$.

Therefore, the two functions $S(M, T^*)$ and $\bar{A}_L(M, T^*)$ characterize the strength and the area of a module.

3.3 Module Maximality

OUR OBJECTIVE BEHIND modular decomposition is to transform the modules into singular aggregate instances in the image link stream (compression process; as

will be discussed in detail in the next chapter). But intuitively, we can understand that the compression process becomes much more meaningful if the minimal number of modules are used to express the whole link stream (complexity; discussed in **Chapter 1 Section 1.1**). Therefore it *is* relevant for us to define *maximal* modules, *i.e.,* modules which are *not included* in any other module subject to spatial and temporal constraints. This expression helps us in obtaining a partition (not necessarily unique) of a given link stream in terms of maximal modules. By design, these maximal modules have the maximal normalized area among all modules while preserving the strength. First, we will distinguish between structural and temporal maximality, and then we will see that joint spatiotemporal maximality follows subsequently.

Definition 6. ***Structural Maximality*** *Given $M \subseteq V$, $(\epsilon, \eta) \in [0,1] \times [0,1/2]$ and for any $T^* \subseteq T$, the couple (M, T^*) is an ϵ, η-relaxed T^*-structurally maximal module if $S(M, T^*) \geq (\epsilon, \eta)$ and there does not exist any set of nodes $M' \supset M^*$ such that $S(M', T^*) \geq (\epsilon, \eta)$.*

Definition 7. ***(Temporal Maximality)*** *Given $T^* \subseteq T$, $(\epsilon, \eta) \in [0,1] \times [0,1/2]$ and for any $M \subseteq V$, the couple (M, T^*) is an ϵ, η-relaxed M-temporally maximal module if $S(M, T^*) \geq (\epsilon, \eta)$ and there does not exist any time interval $T'^* \supset T^*$ such that $S(M, T'^*) \geq (\epsilon, \eta)$.*

Combining the idea of spatial and temporal maximality, we can define an ϵ, η-relaxed spatiotemporally maximal module as a couple (M, T^*) which satisfies both the criterion stated in definitions **6** and **7**.

3.4 Conclusion

IN THIS CHAPTER, we have introduced spatiotemporal modules in link streams, and discussed several structural properties of the modules. The theory of spatiotemporal modules establishes the basic framework upon which we subsequently deal with the problem of link stream compression. In the next chapter, we are going to discuss how we can use these modules to compress a link stream into an image link stream, which retains the macro-scale features of the original dataset constrained to an optimizable amount of information loss.

* * *

CHAPTER 4

Spatiotemporal Compression in Link Streams

4.1 Context

AS WE DISCUSSED in the previous chapter, we wish to compress a given link stream into a smaller image link stream which retains the macro-scale features of the original dataset. This objective has recently been applied to static graphs [21]. This chapter generalizes such a compression scheme into a spatiotemporal framework, to be applied to link streams. Graph compression (static or dynamic) is said to be *lossless* if no information about the original dataset is lost in the compression process, *i.e.,* we find back the exact structure of the original graph when we decompress the compressed version (Figure 4.1).

However, as discussed in **Chapter 1**, such compression becomes more meaningful for real networks if the modules are less rigid in including nodes with approximately same neighborhood structures. This gives rise to a better understanding of community formation in the original dataset, even though we do not obtain the exact structure of the original graph when we decompress the (lossy) compressed representation (Figure 4.2).

Hence, the first order of business is to define a spatiotemporal generalization of lossy graph compression for link streams using modules (which we do in this chapter), and then to discuss how we can minimize this loss of information to maintain the macro-scale features (discussed in the next chapter).

As we can notice in Figure 4.2, lossy compression/decompression in a graph changes the weight of interaction for several links. In a graph $G = (V, E)$, the weight of interaction between two nodes v_i and v_j is defined by the function $w_G : V \times V \to \mathbb{R}^+$. Subsequently, the empirical distribution is defined as [21]:

$$p_G(v_i, v_j) = \frac{w_G(v_i, v_j)}{\sum_{(v'_i, v'_j) \in E} w_G(v'_i, v'_j)}.$$

The decompression process conserves the total weight of interaction in the graph, while changing the empirical weight distribution of the decompressed graph, which is then compared to $p_G(v_i, v_j)$ to quantify the amount of information loss.

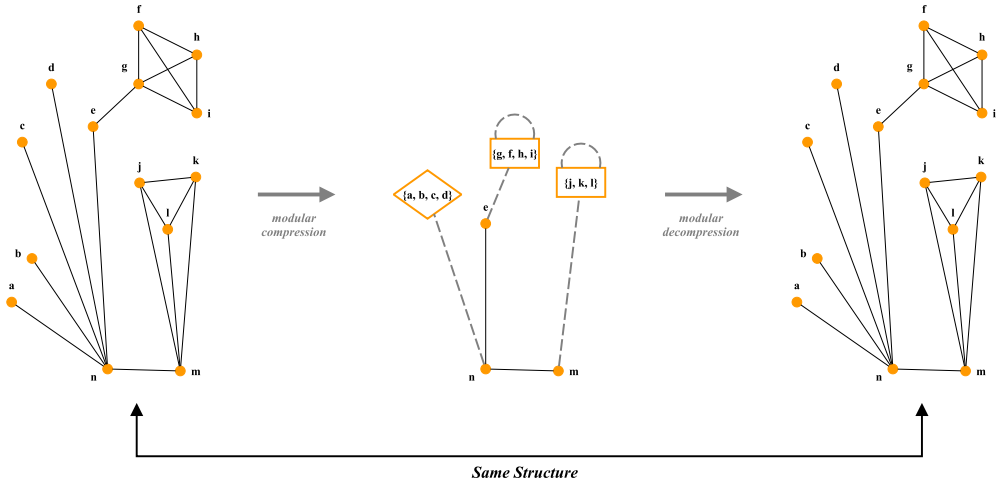


Figure 4.1: **Modular compression and decompression in a simple static graph.** First we identify *modules*, or groups of nodes with similar interaction behavior in the graph on the left. We obtain two different kinds of modules (as shown in the middle figure): the diamond shaped node ($\{a, b, c, d\}$) is a *sparse* module, *i.e.,* none of the nodes within interact with each other, and the square shaped nodes ($\{g, f, h, i\}$ and $\{j, k, l\}$) are *dense* modules (as indicated by the self-loop). Such a decomposition reduces the original graph structure of 14 nodes and 18 links to a compressed version having 6 nodes and 7 links (the figure in the middle). Subsequently we dis-aggregate the module boundaries to obtain the decompressed graph, which matches exactly with the original graph.

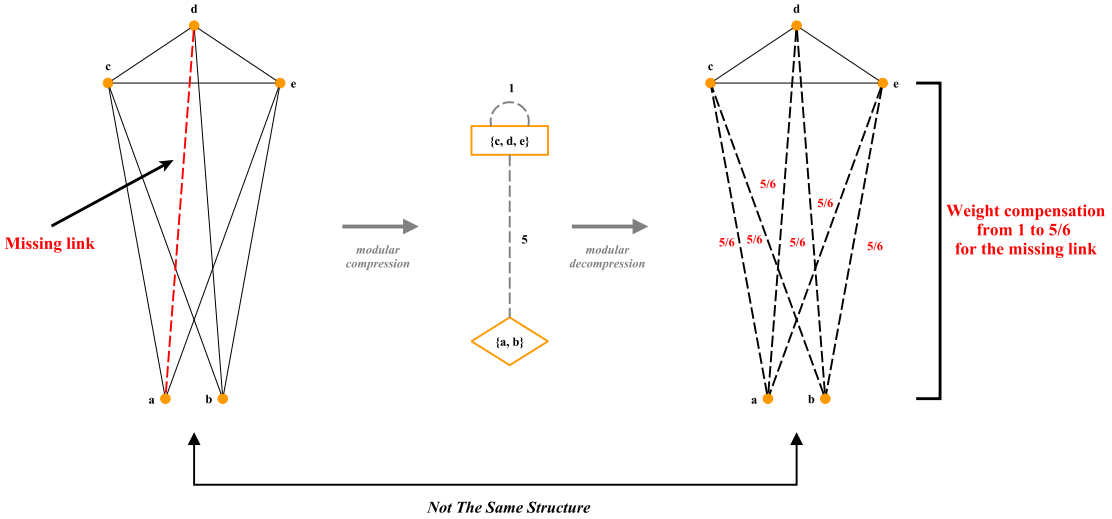


Figure 4.2: **Lossy compression and decompression in a simple static graph.** On the left hand side, we have a static graph with a missing link (as shown by the dotted red line). The figure in the middle gives the aggregated representation of the original graph. The decompression process compensates for this missing link by assigning weight to each intra-modular link equal to the ratio of the number of links present between the two modules $\{a, b\}$ and $\{c, d, e\}$ (5) and total number of possible links between these two modules ($3 \times 2 = 6$).

To be able to do that in our case, first we need to have an idea about the weight of (spatiotemporal) interactions in a link stream so that we can define the empirical weight distribution in a link stream. That is what the next section introduces, based upon the general structure of link streams defined in **Chapter 2**.

4.2 Weight of Link Stream Interactions

IN A LINK STREAM L, the weight of interaction between two nodes u and v in V, at a time instance $t \in T$ is equal to 1 if there exists a link between u and v at time t, and 0 otherwise. That means, for a fixed t, the weight function $w : T \times V \times V \to \{0, 1\}$ is defined as:

$$w(t, v_i, v_j) = \begin{cases} 1, & \text{if } \exists\ (b, e, v_i, v_j) \in E \text{ such that } t \in [b, e] \\ 0, & \text{otherwise} \end{cases} \tag{4.1}$$

In addition, $w(t, v_i, v_i) = 0\ \forall\ v_i \in V$ and $\forall\ t \in T$ as we do not allow any self-loop in the link stream.

The aggregated weight of interaction between v_i and v_j over a time interval $T^* = [b^*, e^*] \subseteq T$ is given by:

$$w(T^*, v_i, v_j) = \int_{t \in T^*} w(t, v_i, v_j)\ dt. \tag{4.2}$$

Then, the aggregated weight of interaction between two sets of nodes V_1 and V_2 over the time interval T^* is given by:

$$w(T^*, V_1, V_2) = \sum_{v_i, v_j \in V_1 \times V_2} w(T^*, v_i, v_j). \tag{4.3}$$

We notice that this equation can be applied to the whole link stream by taking $T^* = T$ and both $V_1, V_2 = V$; *i.e.,* $w(T, V, V)$ gives the total weight of interaction in the link stream. Now, as we only considered a binary function to define the weight of interaction in our link stream (equation **4.1**), $w(T, V, V)$ counts the number of links present in L and assigns a value of 1 to each of the links at each instance of time. In other words, $w(T, V, V)$ returns the sum of all link durations, *i.e.,* $w(T, V, V) = \sum_{l \in E} \bar{l}$.

Then we define the empirical weight distribution for a link stream as:

$$f(t, v_i, v_j) = \frac{w(t, v_i, v_j)}{w(T, V, V)}. \tag{4.4}$$

The following equation intuitively shows how the density $\delta(L)$ of L is related to the empirical distribution $f(t, v_i, v_j)$:

$$\delta(L) = \frac{2\sum_{l\in E}\bar{l}}{|V|(|V|-1)\langle T\rangle} = \frac{2\cdot w(T,V,V)}{|V|(|V|-1)\langle T\rangle} = \frac{2\cdot w(t,v_i,v_j)}{|V|(|V|-1)\langle T\rangle f(t,v_i,v_j)}$$

$$\text{or, } \forall t\in T, v_i, v_j \in V,\ f(t,v_i,v_j)\cdot\delta(L) = \frac{2\cdot w(t,v_i,v_j)}{|V|(|V|-1)\langle T\rangle}. \tag{4.5}$$

The structure of equation **4.4** tells us that we can also define the empirical weight distribution for aggregates, *i.e.,* $f(T^*, M, M)$ for all $T^* \subseteq T$ and $M \subseteq V$ as:

$$f(T^*, M, M) = \frac{w(T^*, M, M)}{w(T, V, V)}.$$

In addition, let us say that the couple (M, T^*) generates an induced sub-stream $L[(M,T^*)] = (T^*, M, E')$. Then the following relation holds between the density of the sub-stream $\delta(L[(M,T^*)])$ and $f(T^*,M,M)$:

$$\begin{aligned}
\delta(L[(M,T^*)]) &= \frac{2\sum_{l\in E'}\bar{l}}{|M|(|M|-1)\langle T^*\rangle} \\
&= \frac{2\cdot w(T^*,M,M)}{|M|(|M|-1)\langle T^*\rangle} \\
&= \frac{2\cdot f(T^*,M,M)\cdot w(T,V,V)}{|M|(|M|-1)\langle T^*\rangle} \\
\Rightarrow \frac{\delta(L[(M,T^*)])}{f(T^*,M,M)} &= \frac{2\cdot w(T,V,V)}{|M|(|M|-1)\langle T^*\rangle}.
\end{aligned}$$

Theorem 1. *For all* $v_i, v_j \in V$ *and* $t \in T$, $f(t,v_i,v_j) \geq 0$ *and*

$$\sum_{v_i,v_j\in V\times V}\int_{t\in T} f(t,v_i,v_j)\ dt = 1. \tag{4.6}$$

Proof. Positivity of $f(t,v_i,v_j)$ follows directly from construction, as negative link weights are not allowed in our formalism. And,

$$\begin{aligned}
\sum_{v_i,v_j\in V\times V}\int_{t\in T} f(t,v_i,v_j)\ dt &= \sum_{v_i,v_j\in V\times V}\int_{t\in T}\frac{w(t,v_i,v_j)}{w(T,V,V)}\ dt \\
&= \sum_{v_i,v_j\in V\times V}\frac{1}{w(T,V,V)}\int_{t\in T} w(t,v_i,v_j)\ dt \\
&= \sum_{v_i,v_j\in V\times V}\frac{w(T,v_i,v_j)}{w(T,V,V)} \\
&= \frac{1}{w(T,V,V)}\sum_{v_i,v_j\in V\times V} w(T,v_i,v_j) \\
&= \frac{w(T,V,V)}{w(T,V,V)} = 1.
\end{aligned}$$

□

4.3 Link Stream Compression

COMPRESSION OF A link stream $L = (T, V, E)$ is achieved by expressing L in terms of spatiotemporal aggregates and interactions within the aggregates (keeping T fixed), instead of nodes and their interactions. As we discussed in the previous chapter, a link stream can be completely decomposed into a countable number of spatiotemporal modules (strong or relaxed). Among all the possible modules that we can detect in L w.r.t. fixed values of $\epsilon \in [0, 1]$ and $\eta \in [0, 1/2]$, we choose a family of modules $\mathbb{P} = \{(M_i, T_i^*)\}_{i=1}^{m}$, $m \in \mathbb{N}$, to perform the compression, such that the modules are pairwise disjoint, $\bigcup_{i=1}^{m}(M_i, T_i^*) = (V, T)$ and there does not exist any $m' \in \mathbb{N}$ such that $m' < m$ and $\bigcup_{i=1}^{m'}(M_i, T_i^*) = (V, T)$; *i.e.,* no subsequence of modules included in $\mathbb{P}$ which forms an exhaustive partition of L. It is noticeable that w.r.t. fixed ϵ and η, $\mathbb{P}$ is not a unique collection of modules; there can be several such partitions that we can choose for compression as long as the chosen partition satisfies the above mentioned conditions.

Given a partition $\mathbb{P}$, we then define the compressed link stream L^C as follows:

$L^C = (T, V^C, E^C)$ where $V^C = \{M_i : (M_i, T_i^*) \in \mathbb{P}\}$ is the set of *supernodes* and $E^C \subseteq T \times T \times V^C \times V^C$ is the set of *superlinks*.

A superlink $l^C = (b^C, e^C, M_i, M_j) \in E^C$ *iff* $\exists$ modules $(M_i, T_i^*), (M_j, T_j^*) \in \mathbb{P}$ such that $b^C = min(T_i^* \cap T_j^*)$ and $e^C = max(T_i^* \cap T_j^*)$; *i.e.,* the supernodes M_i and M_j interact in L^C from time $min(T_i^* \cap T_j^*)$ to $max(T_i^* \cap T_j^*)$ if modules (M_i, T_i^*) and (M_j, T_j^*) exist in $\mathbb{P}$.

We see that the weight of interaction between two supernodes M_i and M_j over the interval of their interaction $[b^C, e^C] = T_i^* \cap T_i^*$ is the aggregate weight of interaction in the original link stream, as defined in the previous section (Equation **4.3**), *i.e.,* $w(T_i^* \cap T_i^*, M_i, M_j)$.

4.4 Link Stream Decompression

AS WE MENTIONED briefly in **Chapter 1**, the decompression of a link stream is achieved by dis-aggregating the spatiotemporal modules to represent all nodes and temporal interactions between them. Therefore, the decompressed link stream L^D has the same physical structure as L, *i.e.,* both L and L^D are defined upon the same time interval T, set of nodes V and (approximately) the same set of links E. However, the only thing that changes is the weight of interactions in L^D. This can be observed by noting that in L^C, $w(T_i^* \cap T_i^*, M_i, M_j)$ represents the compressed weight of interaction between two sets of nodes M_i and M_j over the time interval $T_i^* \cap T_i^*$. Therefore, decompressing this weight function gives us the weight of interaction between two nodes $v_i \in M_i$ and $v_j \in M_j$ at any time $t \in T_i^* \cap T_i^*$. Hence, the weight of interaction in L^D is the function $w^D : T \times V \times V \to \mathbb{R}^+$ is defined as follows:

For all $(M_i, T_i^*), (M_j, T_j^*) \in \mathbb{P}$,

$$w^D(t, v_i, v_j) = \begin{cases} \dfrac{w(T_i^* \cap T_i^*, M_i, M_j)}{|M_i| \cdot |M_j| \cdot \langle T_i^* \cap T_i^* \rangle} & \text{if } M_i \neq M_j. \\ \dfrac{2 \cdot w(T_i^* \cap T_i^*, M_i, M_j)}{|M_i| \cdot (|M_j| - 1) \cdot \langle T_i^* \cap T_i^* \rangle} & \text{if } M_i = M_j. \end{cases} \tag{4.7}$$

This allows us to define the empirical weight distribution of L^D as:

$$f^D(t, v_i, v_j) = \frac{w^D(t, v_i, v_j)}{w(T, V, V)} \tag{4.8}$$

4.5 Conclusion

THIS CHAPTER INTRODUCES spatiotemporal compression in a link stream using modules that we defined in the previous chapter. The compressed link stream L^C reveals the macro-scale properties of L by presenting interactions between modules or communities in it. We have defined the weight of interaction in a link stream L, which allowed us to formulate the empirical weight distribution $f(t, v_i, v_j)$ of L. Subsequently we show how we can decompress a compressed link stream to compute the new empirical weight distribution $f^D(t, v_i, v_j)$. It is noticeable that as the modules considered were ϵ, η-relaxed, $f^D(t, v_i, v_j)$ is not equal to $f(t, v_i, v_j)$ unless $\epsilon = 1$ and $\eta = 0$. In the next chapter, we are going to discuss how do we calculate the *deviation* of f^D from f in terms of the Kullback-Leibler (KL) divergence, and subsequently we will discuss how can we optimize the divergence, without loosing much of the macroscopic features presented in L^C.

* * *

CHAPTER 5

Modular Optimization Problem

5.1 Context

FOLLOWING THE RELAXED modular decomposition and subsequent compression of a link stream, we analyze if it serves the purpose of unveiling the macro-scale features (in terms of modules) in the spatiotemporal dataset, without loosing too much of micro-scale variations. However, as we discussed in the previous chapter, relaxed decomposition (and subsequent compression) of a link stream using spatiotemporal modules cannot be achieved uniquely, but rather there can be several ways to compress a link stream using several partition classes of relaxed modules. Therefore, it is important for us to quantitatively compare the original and the decompressed link stream w.r.t. a chosen module partition, so that we can decide which partition serves our purpose of a meaningful spatiotemporal compression.

5.2 The Building Blocks

AS WE HAVE briefly explained in **Chapter 1**, we formalize the modular optimization problem as a trade-off between the *divergence* and the *complexity* of the compressed link stream. This chapter explains in detail these components of the optimization problem, and then proposes an objective function which needs to be optimized to achieve the best possible compression of a link stream.

5.2.1 Divergence

IN AN ANALOGOUS work on static graph compression by Lamarche-Perrin *et al* [21], the divergence was modeled as the Kullback-Leibler (KL) divergence between the empirical weight distributions of the original and the decompressed graph, where the KL divergence between two discrete probability distributions p and q defined over the same measurable space $(\Omega, \sigma(\Omega))$ (Ω: domain, $\sigma(\Omega)$: sigma algebra over Ω) is defined as:

$$D(p \mid\mid q) = \sum_{x \in \Omega} p(x) log_2 \frac{p(x)}{q(x)}. \tag{5.1}$$

and the sum is replaced by an integral when p and q are continuous probability distributions. $D(p \mid\mid q)$ measures the difference between the distributions, where typically p stands for the *true* distribution of data, whereas q represents a model or an approximation upon the data [29].

We use this model to find the divergence between the empirical weight distribution f of the original link stream L and the approximate distribution f^D of the decompressed link stream L^D (both defined in the previous chapter), given its efficiency in measuring the difference between a given and a construed probability distribution (in our case, the empirical weight distribution). Therefore, the divergence term in our study is $D(f \mid\mid f^D)$. It is noticeable that $D(f \mid\mid f^D)$ attains its minimum possible value 0 when $f = f^D$, *i.e.,* when L and L^D have the same empirical distributions. Similarly, $D(f \mid\mid f^D)$ attains high values when f and f^D are very different from each other.

5.2.2 Complexity

THERE CAN BE several characterizations of complexity based on the field of study. But in general, complexity is defined as the "resource constraint" of a data miner, *i.e.,* how much resource the data miner needs to answer a specific question about a dataset in hand. We interpret complexity in our study as the number of spatiotemporal relaxed modules required to compress the link stream. Hence, the complexity term in our study is defined as the cardinality of the partition $\mathbb{P}$, *i.e.,* $|\mathbb{P}|$ as it returns the exact number of modules required to compress a link stream.

It is noticeable that while compressing the link stream $L = (T, V, E)$, the minimum attainable complexity is $|\mathbb{P}| = 1$, *i.e.,* the only module considered is L itself; and the maximum attainable complexity is infinity which occurs when every couple $(\{v\}, t)$ is a module in L.

5.3 Proposing The Optimization Problem

NOW THAT WE have defined the divergence and complexity of our compressed link stream, we are in a position to define the objective function that needs to be minimized to achieve optimal compression. We need to keep in mind that complexity is high when we have high number of small modules, which leads to a low divergence as smaller modules approximate micro-scale interactions more appropriately (**low** $D(f \mid\mid f^D)$, **high** $|\mathbb{P}|$). Contrariwise, complexity is low when we have a low number of modules, which leads to a high divergence as the bigger modules neglect micro-scale variations to reveal macro-scale patterns (**high** $D(f \mid\mid f^D)$, **low** $|\mathbb{P}|$). Therefore, the desired (primal) optimization problem is read as:

$$\begin{aligned} \underset{\mathbb{P}}{\text{minimize}} \quad & D(f \mid\mid f^D) \\ \text{subject to} \quad & |\mathbb{P}| \leq c,\ c \in \mathbb{N}. \end{aligned} \tag{5.2}$$

Subsequently, the dual problem is proposed as:

$$\begin{aligned} \underset{\mathbb{P}}{\text{minimize}} \quad & |\mathbb{P}| \\ \text{subject to} \quad & D(f \mid\mid f^D) \leq d,\ d \in \mathbb{R}^+. \end{aligned} \tag{5.3}$$

Therefore, solving the primal or the dual optimization problem returns the best compression scheme for a link stream which reveals the macro-scale spatiotemporal modules present in the dataset (in terms of relaxed modules) with minimal possible loss of micro-scale variations.

5.4 Conclusion

THIS CHAPTER INTRODUCES an objective function as a trade-off between the divergence and complexity of the compressed link stream, and converts the module decomposition problem into a module optimization problem. This marks the conclusion of the theoretical framework we developed to address the problem of lossy spatiotemporal compression in a link stream. In the next chapter we discuss some of the limitations of our work and suggest possible future extensions to this framework.

* * *

CHAPTER 6

Final Remarks & Perspectives

IN THIS THESIS work, we have introduced a mathematical framework for an optimal spatiotemporal modular decomposition based link stream compression. This chapter summarizes the main takeaways from this report, and discusses possible future extensions to our work. Altogether, this is a purely mathematical work and no datasets were considered to test our theory.

Chapter 2 in this thesis introduces the link stream framework: existing theory and some extensions required to approach the modular decomposition problem.

In Chapter 3, we use this framework to define spatiotemporal modules (strong & relaxed) in a link stream, and discuss several properties.

In Chapter 4, we discuss how we can use the module framework to compress a link stream L into a compressed image link stream L^C, which reveals the macroscopic modular nature of the original structure. We also discuss how we can embed a constant weight distribution to the links in L, which helps us determine the aggregate weight of interactions in L^C, and subsequently construct a decompressed version of the L^C, defined by L^D.

Chapter 5 discusses how we can convert this data mining problem into an optimization problem, where we propose our objective as a trade-off between the divergence and the complexity of the system.

One primary goal of our work is to design and develop an algorithm for spatiotemporal compression in link streams. This could happen from two different perspectives:

1. Constructing a greedy approach to mine the spatiotemporal modules in a link stream to construct the desired partition $\mathbb{P}$, with respect to the optimization parameters (divergence and complexity).

2. Constructing a Monte-Carlo approach to find random partitions which minimizes the objective function in the proposed modular optimization problem.

It is also important to work on the structural relation between spatiotemporal modules and spatiotemporal cliques [27], so that existing algorithmic approaches for cliques can be modified to mine modules in a link stream.

It is of utmost importance to see if we can give a form for optimal modular partition in a link stream with respect to fixed values of the defined precision parameters ϵ and η, and the variation of the optimal partition with respect to controlled variation of the precision parameters. This might lead us to an optimal choice for the precision parameters with respect to the dataset in consideration. It would also be interesting to see if we can incorporate functional interaction weight in our study, to make the modular decomposition problem more realistic with respect to real world interactions.

Our research work combines link stream theory and information theoretic compression to achieve spatiotemporal compression in a link stream. Theoretically, this framework is still in its infancy, and a lot needs to be done before we can proceed towards full-scale applications. However, we strongly believe that this foundational work can be used to achieve relaxed decomposition of a link stream in an optimal manner to reveal the modular behavior of the involved nodes, which would be a valuable contribution to the ever growing field of dynamical network research.

* * *

References

1. Watts, D. & Strogatz, S. Collective dynamics of small-world networks. *Nature* **393**, 440–442 (1998).

2. Barabasi, A. & Albert, R. Emergence of scaling in random networks. *Science* **286**, 509–512 (1999).

3. Albert, R. & Barabasi, A. Statistical mechanics of complex networks. *RevModPhys* **74**, 47–97 (2002).

4. Newman, M. E. J. *Networks: An introduction*, (Oxford University Press, New York, 2010).

5. Fortunato, S. Community detection in graphs. *Physics Reports* **486**, 75174 (2010).

6. Newman, M. & Girvan, M. Finding and evaluating community structure in networks. *Phys Rev E* **69**, 026113 (2004).

7. Holme, P. & Saramaki, J. (eds.), *Temporal Networks*, Understanding Complex Systems, (Springer-Verlag Berlin Heidelberg, 2013)

8. Magnien, C. & Tarissan, F. Time Evolution of the Importance of Nodes in dynamic Networks. In *International Symposium on Foundations and Applications of Big Data Analytics (FAB)*, in conjunction with International Conference on Advances in Social Networks Analysis and Mining (ASONAM), Paris, France, 2015.

9. Abry, P. Baraniuk, R. Flandrin, P. Riedi, R. & Veitch, D. Multiscale nature of network traffic. *Signal Processing Magazine, IEEE,* **19**(3):28–46, (2002).

10. Viard, T. & Latapy, M. Identifying roles in an IP network with temporal and structural density. In *Computer Communications Workshops (INFOCOM WKSHPS)*, pages 801–806, (2014).

11. Lee, J. D. & Maggioni, M. Multiscale analysis of time series of graphs (2011).

12. R. S. Caceres, R., & T. Berger-Wolf. *Temporal Networks, chapter: Temporal Scale of Dynamic Networks* (Springer Link, 2013).

13. Aynaud, T. Guillaume, J. L. Static community detection algorithms for evolving networks. *WiOpt'10: Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks*, May 2010, Avignon, France. pp.508-514, (2010).

14. Abry, P. Veitch, D. Flandrin, P. Long range dependence: Revisiting aggregation with wavelets. *Journal of Time Series Analysis* Volume 19, Issue 3, pages 253–266, (1998).

15. Gagneur, J., Krause, R., Bouwmeester, T. & Casari, G. Modular decomposition of protein-protein interaction networks. *Genome Biol* **5**, R57 (2004).

16. Ahnert, S. E. Generalised power graph compression reveals dominant relationship patterns in complex networks. *Scientific Reports* **4**: 4385 (2014).

17. Sun, J. Faloutsos, C. Papadimitriou, S. & Yu, P. S. GraphScope: Parameter-free Mining of Large Time-evolving Graphs. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 687-696 (2007).

18. Aynaud, T. & Guillaume, J. L. Multi-Step Community Detection and Hierarchical Time Segmentation in Evolving Networks. in *Proceedings of the 5th SNA-KDD Workshop* (2011).

19. Guigourès, R. Boullé, M. & Rossi, F. A Triclustering Approach for Time Evolving Graphs. *IEEE 12th International Conference on Data Mining Workshops.* Pages: 115 - 122 (2012).

20. Dhillon, I. Mallela, S. Modha, D. Information-Theoretic Co-clustering. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge*

discovery and data mining. Pages 89-98 (2003).

21. Lamarche-Perrin, R. Tabourier, L. Tarissan, F. Multilevel Analysis of Co-authorship Networks. *Second European Conference on Social Networks (EUSN'16)*, Paris (2016).

22. Liu, W. Kan, A. Chan, J. Bailey, J. Leckie, C. Jian, P. Kotagiri, R. On Compressing Weighted Time-evolving Graphs. In *Proceedings of the 21st ACM international conference on Information and knowledge management.* Pages 2319-2322 (2012).

23. Léo, Y. Crespelle, C. Fleury, E. Non-Altering Time Scales for Aggregation of Dynamic Networks into Series of Graphs. In *11th International Conference on emerging Networking EXperiments and Technologies – CoNEXT* (2015).

24. Reichardt, J. & White, D. R. Role models for complex networks. *Eur. Phys. J. B* **60**, 217–224 (2007).

25. Lorrain, F & White, H. C. Structural Equivalence Of Individuals In Social Networks. *J. Math. Sociol.* **1**, 49 (1971).

26. P. Doreian, V. Batagelj, A. Ferligoj, *Generalized Blockmodeling.* (Cambridge University Press, New York, 2005)

27. Viard T. Flots de liens pour la modélisation d'interactions temporelles et application à l'analyse de trafic IP. PhD Thesis, UPMC (2016).

28. Jaccard, P. Étude comparative de la distribution florale dans une portion des Alpes et des Jura, *Bulletin de la Société Vaudoise des Sciences Naturelles* 37: 547–579. (1901).

29. Kullback, S., Leibler, R.A. On information and sufficiency. *Annals of Mathematical Statistics.* **22** (1): 79–86 (1951).

* * *