

Analysis of the Temporal and Structural Features of Threads in a Mailing-List

Noe Gaumont, Tiphaine Viard, Raphaël Fournier-S'Niehotta, Qinna Wang, Matthieu Latapy

▶ **To cite this version:**

Noe Gaumont, Tiphaine Viard, Raphaël Fournier-S'Niehotta, Qinna Wang, Matthieu Latapy. Analysis of the Temporal and Structural Features of Threads in a Mailing-List. 7th Workshop on Complex Networks (CompleNet 2016), Mar 2016, Dijon, France. pp.107-118, 10.1007/978-3-319-30569-1_8 . hal-01345821

HAL Id: hal-01345821
https://hal.science/hal-01345821

Submitted on 15 Jul 2016

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.

Analysis of the temporal and structural features of threads in a mailing-list

Noé Gaumont[2], Tiphaine Viard[2], Raphaël Fournier-S'niehotta[1], Qinna Wang, and Matthieu Latapy[2]

[1] CNAM, CEDRIC fournier@cnam.fr,
[2] Sorbonne Universités, UPMC Univ Paris 06, CNRS, LIP6 UMR 7606, 4 place Jussieu 75005 Paris first.last@lip6.fr

Abstract. A link stream is a collection of triplets (t,u,v) indicating that an interaction occurred between u and v at time t. Link streams model many real-world situations like email exchanges between individuals, connections between devices, and others. Much work is currently devoted to the generalization of classical graph and network concepts to link streams. In this paper, we generalize the existing notions of intra-community density and inter-community density. We focus on emails exchanges in the Debian mailing-list and show that threads of emails, like communities in graphs, are dense subsets loosely connected from a link stream perspective.

1 Introduction

Exchanges in a mailing-list are often studied as complex networks: there is a link between two individuals if they exchange emails. In particular, communities in such complex networks capture groups of friends or close colleagues (individuals that exchange many more messages within the group than outside the group, typically) [2]. However, removing all time information has important consequences if one wants to study the dynamics of email exchanges.

In order to study those dynamics, one may label each link with the frequency of exchanges or the times at which they occur [5], but capturing both the structure and the dynamics of exchanges remains challenging. In particular, studying threads calls for methods that capture the temporal nature of interactions more accurately, without loosing the power of network analysis.

We propose here to model email exchanges directly as link streams, *i.e.* series of triplets (t,a,b) meaning that individuals a and b exchanged an email at time t. We then introduce notions that capture both the temporal and structural nature of these exchanges. We use a typical dataset obtained from a public mailing-list archive to illustrate our approach. We analyze this dataset using our model, with a special focus on the properties of threads within the whole archive. Our goal is to understand how the now classical concept of communities in complex networks may translate to threads in link streams representing email exchanges. Indeed, we expect the exchanges of a given thread to involve a specific set of individuals for a specific period of time, thus being dense from both structural and temporal points of view. This is illustrated in Figure 1.

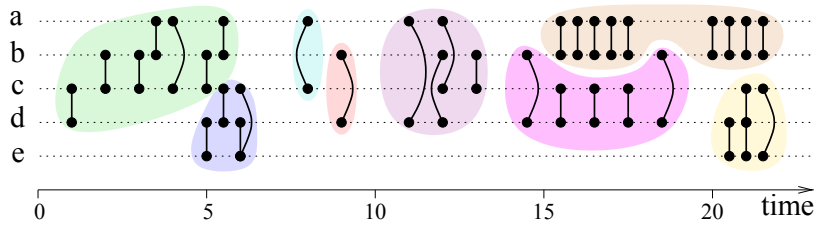


Fig. 1: An example of link stream representing email exchanges between individuals a, b, c, d and e, with threads represented by colored areas. For instance, at time 5, b and c exchange an email, as well as d and e. Threads are *a priori* dense series of exchanges involving a limited group of nodes during a limited period of time.

2 Dataset

Archives of exchanges in various mailing-lists are readily available on the web, and studying them provides very rich insights on various issues. They have the advantage of being publicly available in many cases, and some involve large amounts of users over long time periods.

A typical example is provided by Debian mailing-list [4]: it contains emails sent from over 51753 email addresses, over almost 20 years. In addition, exchanges in this mailing-list have been studied in the past [1,3,7]. Finally, this dataset provides the thread information for each message, that we can use as a ground truth. For all these reasons, we use in this paper the Debian mailing-list to illustrate and validate our approach.

More precisely, we crawled the Debian mailing-list web archive [4]. For each message m, we extract its author $a(m)$, the date $t(m)$ at which it was posted (converted into UTC time), and the message it is replying to $p(m)$ (through the IN-REPLY-TO entry), which has a corresponding author $a(p(m))$. This corresponds to an interaction between $a(m)$ and $a(p(m))$ at time $t(m)$ in the link stream. Some messages are not answers to any other message (they are directly sent to the mailing-list), and in this case we state that $p(m) = m$. Such messages are called *root* messages.

We capture the mailing-list from January 1st, 1996 to December 31st, 2014. We obtain a dataset $\mathscr{D}$ of $n = 722716$ emails sent from 51753 distinct email addresses.

Each root message m naturally induces a thread: it is the set $\mathscr{T}(m)$ of messages such that m belongs to $\mathscr{T}(m)$ and if a message m' is in $\mathscr{T}(m)$ then all messages m'' such that $p(m'') = m'$ also belong to $\mathscr{T}(m)$. In other words, $\mathscr{T}(m)$ contains exactly m, the answers to m, the answers to these answers, and so on. The focus of this paper is the study of structural and temporal features of these threads.

Our data contains incomplete threads: the ones that have an email in our dataset but began before and/or continued after the data collection period. Some threads also exhibit inconsistencies, for instance a reply has a smaller timestamp than the message it replies to. We remove those threads, as well as all threads that last for more than 2 years, or that start 2 years before the end of our data collection.

After this bias correction procedure, we obtain $n = 554233$ emails, involving 34648 distinct authors over a duration of 598532269 seconds (18 years, 11 months and 19 days) and 116999 threads.

3 Framework and notations

Our goal is to study the structural and temporal properties of threads within a mailing-list archive. In order to do so, we propose a model of the data that captures both its temporal and structural nature, and allows for easy manipulation of threads.

We model our mailing-list archive as the link stream $D = (T_D, V_D, E_D)$ with $T_D = [\alpha, \omega]$, $V_D = \{a(m) : m \in \mathscr{D}'\}$ and $E_D = \{(t(m), a(m), a(p(m))) : m \in \mathscr{D}'\}$ where $\mathscr{D}'$ is the set of emails in our dataset after cleaning. In other words, a triplet (t,u,v) in E_D indicates that individual u answered to an email of individual v at time t.

Such a link stream naturally contains sub-streams: $L' = (T', V', E')$ is a substream of $L = (T, V, E)$ if and only if $T' \subseteq T$, $V' \subseteq V$ and $E' \subseteq E$. In other words, all the interactions of L' also appear in L. Given a set of nodes S, we define the sub-stream $L(S)$ of L induced by S as the largest sub-stream of L such that all the links in $L(S)$ are between nodes in S.

Any link stream $L = (T, V, E)$ also induces a graph $G = (V_G, E_G)$ where $V_G = \{u : \exists t \in T, v \in V \text{ s.t. } (t,u,v) \in E\}$ and $E_G = \{(u,v) : \exists t \in T \text{ s.t. } (t,u,v) \in E\}$. In our case, the whole mailing-list archive induces the graph $G(D)$ among authors of emails, and each thread induces a sub-graph of $G(D)$.

In a graph $G = (V, E)$, a community structure is defined by a partition $C = \{C_i\}_{i=1..k}$ of V into k communities. In other words, $\bigcup_i C_i = V$ and $C_i \cap C_j = \emptyset$ whenever $i \neq j$. In a similar way, one may consider a link stream $L = (T, V, E)$ and a partition of its links into k sub-streams $P = \{P_i = (T_i, V_i, E_i)\}_{i=1..k}$. In other words, for any $(t,u,v) \in E$, there exists a unique j between 1 and k such that (t,u,v) is a link of E_j.

The threads in our email dataset are exactly a partition of the whole stream, which we denote by $\mathscr{T} = \{P_i\}_{i=1..k}$ where k is the number of threads and each P_i is a sub-stream representing a thread (with our notations above, there exists a message m such that $P_i = \mathscr{T}(m)$). See Figure 1.

Notice that, although the threads are a partition of the whole stream, their induced graphs may overlap: some nodes and links of $G(D)$ belong to several sub-graphs $G(P_i)$. As a consequence, threads do not induce a partition of $G(D)$ into communities. Instead, one may see the partition of D into threads as a community structure, and this is the focus of our work.

Notice finally that we consider that links are undirected (i.e. $(t,u,v) = (t,v,u)$) and happen at an instant in time (regardless, for instance, of when the message is read). Taking into account the direction and duration of links is out of the scope of this work.

4 Basic statistics

In this section, we present the basic statistics describing the threads in our dataset and the whole archive.

The most basic description of our data certainly is the number of links (*i.e.* emails) they contain, the number of distinct nodes (*i.e.* authors) involved, the number of distinct links they contain (distinct pairs of authors in direct interaction), and their duration (time from the first email to the last one). Figure 2 display the distribution of these values for each thread.

Although the largest thread lasts more than a year, most threads are contained within a few days (100000 seconds is a bit more than 24 hours). Similarly, the largest thread involves 100 messages, though all intermediate sizes are represented in the dataset. Most threads are very short and involve less than 3 messages.

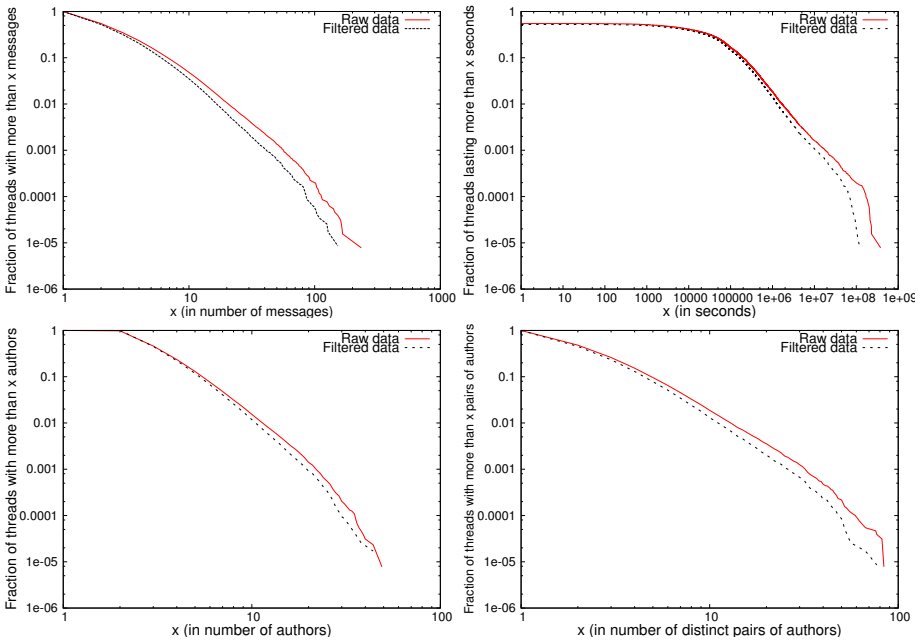


Fig. 2: Complementary cumulative distributions for basic statistics of our raw (solid line) and filtered (dotted line) datasets. Top left: thread sizes (number of messages per thread); top right: thread durations (time elapsed between the first and the last message of the thread); bottom left: number of distinct authors; bottom right: number of distinct pairs of authors.

In order to gain more insight, we observe correlations between some of these basic statistics. Figure 3 (left) shows that thread duration and size are correlated (the larger a thread is, the longer it is likely to be); notice however that for small-sized threads, all types of durations are represented. Looking at the correlations between the size of threads and the number of distinct authors involved shows that threads nearly always involve more messages than authors. This is a typical feature of mailing-lists [1] and as such is dataset-dependent.

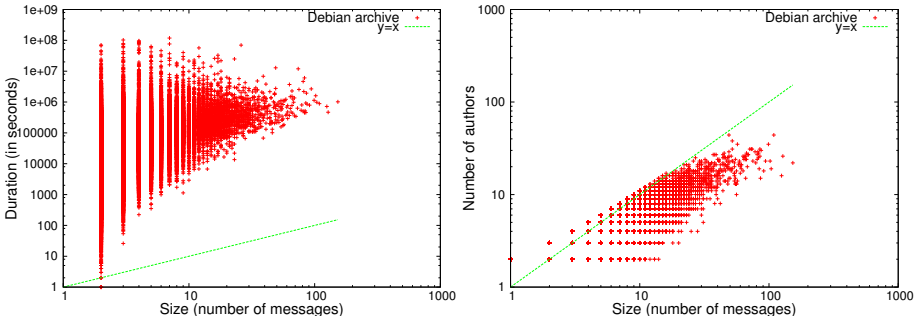


Fig. 3: Left: Correlations between size and duration of threads. Right: Correlations between size of threads and the number of authors involved.

In a link stream $L = (T,V,E)$ with $T = [\alpha,\omega]$, we define, for all $(u,v) \in V \times V$, the maximal sequence $t_{uv} = (\alpha, t_0, \ldots, t_k, \omega)$ such that for all i between 0 and k, there exists $(t_i,u,v) \in E$, and for all i between 0 and $k-1$, $t_i \leq t_{i+1}$. In other words, t_{uv} is the ordered sequence of apparitions of the link (u,v) to which we add α and ω.

We further define $\tau(u,v) = (t_{i+1} - t_i)_{i=0..k+1}$ the sequence of intercontact times of a pair of u and v in V. In other words it is the series of times elapsed between two consecutive occurrences of a link between them. Figure 4 (left) shows the inter-contact times distribution in the Debian mailing-list for all pairs of nodes (u,v).

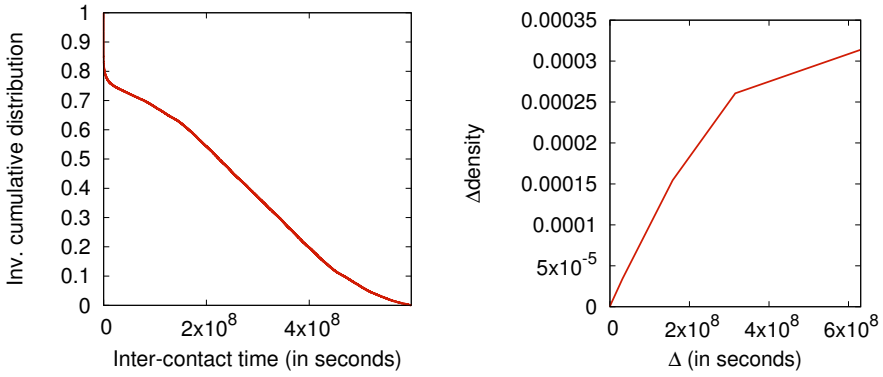


Fig. 4: Left: Inter-contact times distribution in the Debian mailing-list dataset. Right: Evolution of the Δ-density of the link stream for Δ from 1 second to 20 years.

5 Interactions within threads

The key feature of communities is the fact that they form dense subgroups. This section is therefore devoted to the study of density of interactions within threads, from both structural and temporal point of views.

5.1 Density of threads

In a graph, the density is the probability that two randomly chosen nodes are linked together. In other words, it captures the extent at which *all* nodes are directly connected to each other. The density of the graph $G(D)$ induced by our dataset is 3.139×10^{-4}.

In [6], we introduced the notion of Δ-density to capture a similar intuition in link streams, involving both structure and time. Indeed, given a duration Δ, the Δ-density of link stream L is the probability that a link appears between two randomly chosen nodes during a randomly chosen time interval of duration Δ. It captures the extent at which all nodes are directly connected to each other at least every Δ time units. Formally, it is defined as:

$$\delta_\Delta(L) = 1 - \frac{2 \cdot \sum_{u,v \in V, u \neq v} \sum_{t \in \tau(u,v)} \max(0, t-\Delta)}{|V| \cdot (|V|-1) \cdot \max(0, \omega - \alpha - \Delta)}$$

where $\tau(u,v)$ denotes the inter-contact times between u and v, and α and ω are the start and end time of the link stream.

In order to study the Δ-density in our data, we first have to choose an appropriate Δ. We use here several values which capture email dynamics at different scales: $\Delta =$ 1 minute, 1 hour, 1 day, 1 week, 1 month, 1 year and 20 years (the whole duration of the dataset). Figure 4 (right) displays the evolution of the Δ-density of the stream for all theses values of Δ. It shows that the Δ-density is small for small Δs, and converges to the density of the graph induced by the email exchanges (in our case, $3.139 \cdot 10^{-4}$).

In Figure 4 (right), the inflexion points give information on the values of Δ where the dynamics change. Still, looking at the density of the whole stream is very coarse and yields little information. A finer approach consists in looking at the Δ-density of relevant sub-streams. In our case, the threads between authors are a natural object to study.

5.2 Intra-thread density

More globally, given a graph $G = (V,E)$ and a partition $C = \{C_i\}_{i_1..k}$ of V into k communities, the density within communities of C is captured by the *intra-community density*:

$$\frac{2 \cdot \sum_i |\{(u,v) \in E,\, u \in C_i \text{ and } v \in C_i\}|}{\sum_i |C_i| \cdot (|C_i|-1)}$$

In other words, intra-community density is the probability that two nodes chosen at random in the same community are linked together.

In our case, this notion does not directly make sense: as already noticed, we do not have communities defined on $G(D)$ since the graphs induced by threads overlap. However, we extend the notion of intra-community density to link streams as follows. The

intra-thread Δ-density is the probability that two randomly chosen authors contributing to the same thread are linked together within a randomly time interval of duration Δ, for a given Δ:

$$1 - \frac{2 \cdot \sum_i \sum_{u,v \in V_i, u \neq v} \sum_{t \in \tau_i(u,v)} \max(0, t - \Delta)}{\sum_i |V_i| \cdot (|V_i| - 1) \cdot \max(0, \omega_i - \alpha_i - \Delta)}$$

where V_i is the set of authors involved in thread P_i, α_i is the time of the first message in the thread (i.e the minimal t such that there exists a $(t,u,v) \in E_i$), ω_i is the time of the last message in the thread (i.e the maximal t such that there exists a $(t,u,v) \in E_i$) and $\tau_i(u,v)$ denotes the inter-contact times in P_i.

In our data, the inverse cumulative distribution of intra-thread Δ-densities are in Figure 5 (left) for several values of Δ ranging from 1 minutes to 1 year. For each point on the x-axis, the plot gives the proportion of threads in the mailing-list that have an intra-thread Δ-density higher than x. As expected, the higher the Δ used, the higher the density is. However, there is no significant change between a Δ of 7 days and a Δ of 1 year.

Moreover, these distributions confirm that the interactions within threads are much denser (both structurally and temporally) than in the global mailing-list. Indeed, the median intra-thread Δ-density ranges from 2.69×10^{-4} to 0.28 while the link stream Δ-density ranges from 1.05×10^{-10} to 3.42×10^{-5}. The intra-thread Δ-density typically is 10^5 times larger than the global Δ-density.

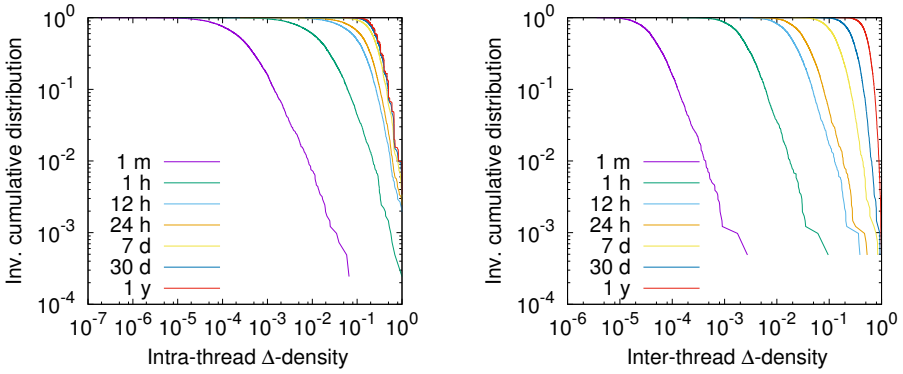


Fig. 5: Left: Inverse cumulative distributions of values of intra-thread Δ-density for different Δs. Right: Inverse cumulative distributions of values of inter-thread Δ-density for different Δs.

This shows that threads are indeed dense substreams in our link streams.

6 Relations between threads

In the previous section, we focused on structural and temporal properties *inside* threads, compared to the whole link stream. We now turn to the study of relations *between* threads.

6.1 Inter-thread density

Let us first study the density of relations between threads in a way similar to above. Given a graph $G = (V,E)$ and a partition $C = \{C_i\}_{i_1..k}$ of V into k communities, the inter-community density is the probability that two nodes chosen at random in two different communities are linked together:

$$\delta^{inter}(C_i) = \frac{1}{|C|} \sum_{j,i \neq j} \frac{|\{(u,v) \in E \text{ s.t. } u \in C_i \text{ and } v \in C_j\}|}{|C_i| \cdot |C_j|}$$

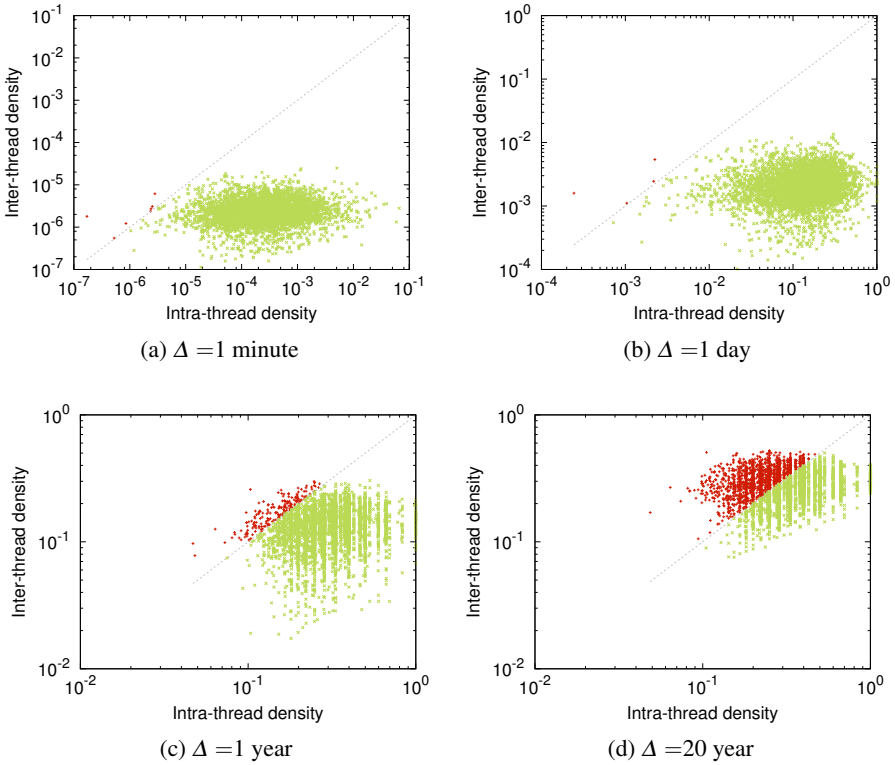
Again, this notion does not directly make sense in link streams, as threads do not induce a partition of nodes. As a consequence, we introduce the inter-thread Δ-density as the probability that two randomly chosen nodes in different communities are linked together during a time interval of duration Δ chosen at random during the time duration of both threads.

Let us define the inter-thread substream between a thread P_i and a thread P_j: $L_{ij} = (T_{ij}, V_{ij}, E_{ij})$, with $T_{ij} = [\min(\alpha_i, \alpha_j), \max(\omega_i, \omega_j)]$, $V_{ij} = V_i \cup V_j$ and $E_{ij} = \{(t,u,v) : t \in T_{ij}, u,v \in V_{ij}, (t,u,v) \in E \setminus E_i \cup E_j\}$. In other words, this is the substream containing the links between nodes of P_i or P_j that are not involved in threads P_i and P_j. The inter-thread density between P_i and P_j is the Δ-density of L_{ij}. In order to obtain the inter-thread Δ-density of P_i to all other threads, we simply average the inter-threads Δ-densities of P_i and all other threads. More precisely:

$$\delta^{inter}_{\Delta}(C_i) = \frac{1}{|C|} \sum_{j,i \neq j} \delta_{\Delta}(L_{ij})$$

In our data, the inverse cumulative distribution of inter-thread Δ-densities are displayed in Figure 5 (right) for different values of Δ. For each point on the x-axis, the plot gives the proportion of threads in the mailing-list that have an intra-thread Δ-density higher than x. Again, larger Δ correlates with larger Δ-densities. However, the inter-thread Δ-density does not plateau, even for large values of Δ. This is natural, since the number of links considered in the computation of the inter-thread Δ-density naturally grows with Δ.

In Figure 6, the correlations between the inter- and intra-thread Δ-density are plotted for some values of Δ. As expected, intra-threads are denser than inter-threads. This relation holds as Δ is bigger, even though the difference between inter and intra thread Δ shrinks. Further experimentation shows that for $\Delta = 20$ years, the difference is non-existent. The figure is omitted for brevity. This is due to the fact that the bigger the Δ, the less the temporal characteristics of threads are important.



(a) Δ =1 minute (b) Δ =1 day

(c) Δ =1 year (d) Δ =20 year

Fig. 6: Correlations between inter- and intra-thread densities for different values of Δ.

6.2 Graphs between threads

Relations between sub-streams L_i, $i = 1..k$, may have different forms, and in particular they have a temporal and a structural nature. In order to capture the temporal relations between sub-streams, one may define the temporal overlap graph as follows: $X = (V,E)$ with $V = \{i, i = 1..k\}$ and there is a link (i, j) in E whenever P_i and P_j have a temporal intersection (*i.e.* $[\alpha_i, \omega_i] \cap [\alpha_j, \omega_j] \neq \emptyset$). Likewise, one may define the node overlap graph as follows: $Y = (V,E)$ with $V = \{i, i = 1..k\}$ again and there is a link (i, j) in E whenever there is a node v involved in both P_i and P_j (*i.e.* there exists a t, a t'n a u and a u' such that there is a link (t,u,v) in P_i and a link (t',u',v) in P_j.

The graphs contain 116999 nodes (the number of threads) and about 2 million edges for the temporal overlap graph and 63 millions for the node overlap graph. These graphs encode much information about relations between threads. For instance, the degree of node i in X is the number of threads active at the same time as P_i.

We display in Figure 7 (left) the correlations between the degree in X and the thread size. There is a clear correlation between the thread duration and the degree in temporal

overlap graph when threads have a duration of at least 10^5s. Also, it appears that some time up to 10^4 threads are present simultaneously as reflected by the maximal degree.

Figure 7 (right) shows the correlations between the degree in Y and the thread duration. The correlation is less clear between the thread node size and the degree in the node overlap graph. However, the trends appears: threads with a lot of participants have a high degree in the graph.

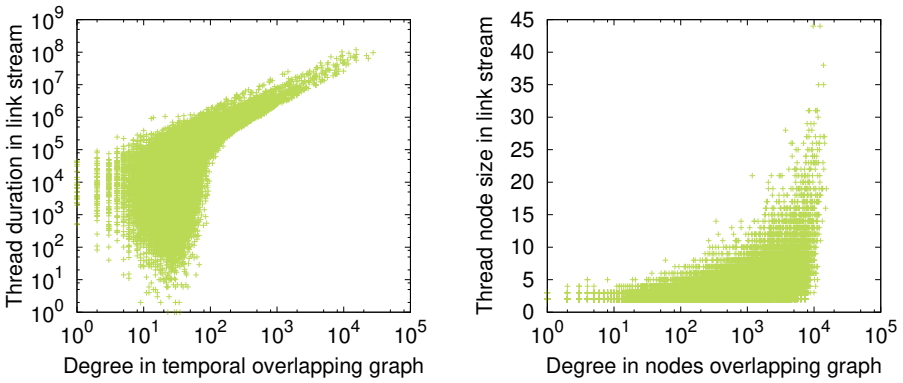


Fig. 7: Left: Correlation between the degree in the time overlap graph X and the thread size. Right: Correlation between the degree in the node overlap graph Y and the thread duration.

6.3 Quotient stream

The *quotient* graph is another key notion for studying the relations between communities in a graph $G = (V,E)$. Given a partition $C = \{C_i\}_{i=1..k}$ of V into communities, in the quotient graph $\overline{G}$ each node i, $i = 1..k$, represents community C_i and there is a link between two nodes i and j, $i \neq j$, if there is a link between a node in C_i and a node in C_j in G. See Figure 8 for an illustration. One may add on each link a weight indicating the number of links between communities. Clearly, the quotient graph captures relations between the communities under concern; for instance, its density indicates up to what point all communities have links between them.

To deepen our understanding of our data, we capture here both temporal and structural nature of relations between sub-streams. We define the quotient stream induced by a partition $P = \{P_i = (T_i, V_i, E_i)\}_{i=1..k}$ of link stream L as the stream $Q = (T_Q, V_Q, E_Q)$ such that $(P_i, P_j, t) \in E_Q$ if and only if there exists (u, v, t_1) in E_i, (u, v', t_2) in E_i and (u, v'', t) in E_j with $t_1 \leq t \leq t_2$. In other words, there is a node u that has a link within P_j occurring between two of its links in P_i. This means that u is involved in the two streams during the same time period.

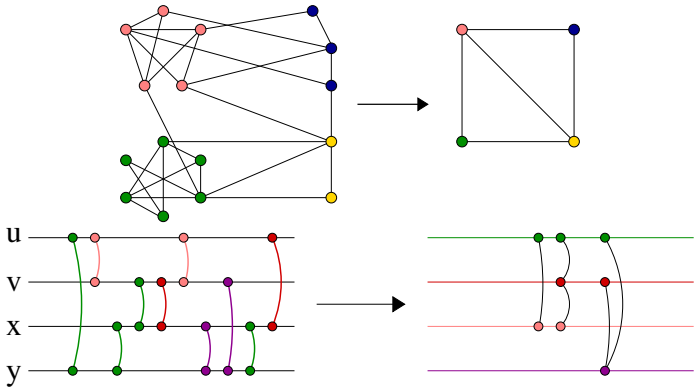


Fig. 8: Top: An example of graph exhibiting communities and its corresponding graph quotient. Bottom: An example of link stream with communities and its corresponding quotient stream.

The quotient stream induced by the threads in our dataset has 12281269 links and involves 68524 distinct nodes (i.e. threads). Since our dataset contains 116999 threads, this implies that 48475 threads are not in relation with any others.

Figure 9 shows the Δ-density of the quotient stream and the Δ-density of the original stream for different values of Δ. The quotient is not very Δ-dense, i.e. threads are not densely connected together, though it is slightly denser than the stream for large values of Δ. This is comparable to graphs.

7 Conclusion

Through the prism of link streams, we have studied the email exchanges in the Debian mailing-list over almost 20 years. From Δ-density, we define notions of *inter* thread density, *intra* thread density and quotient stream, that are generalizations of the equivalent notions in graphs. We show the relevance of these notions on a real-world dataset of email exchanges.

We have shown that threads in the mailing-list are Δ-dense substreams from a link stream perspective, just like communities are dense subgraphs from a graph perspective. Moreover, the threads appear to be denser internally than externally which is another feature of communities in graphs. We also study the relations between threads with the node and temporal overlap graphs. These graphs reveal the highly temporal and structural overlapping nature of threads. However these graphs focus either on time or structure. The quotient stream successfully accounts for both aspects, and exhibits similar properties as its graph counterpart.

Though threads are readily identified in the Debian mailing-list archive, this is usually not the case. Detecting dense substreams loosely interconnected without *a priori* knowledge remains a challenge.

Notice that our approach is dependant of a parameter Δ, that has to be chosen externally. Considering links with durations (i.e. (b,e,u,v), meaning that u and v are in-

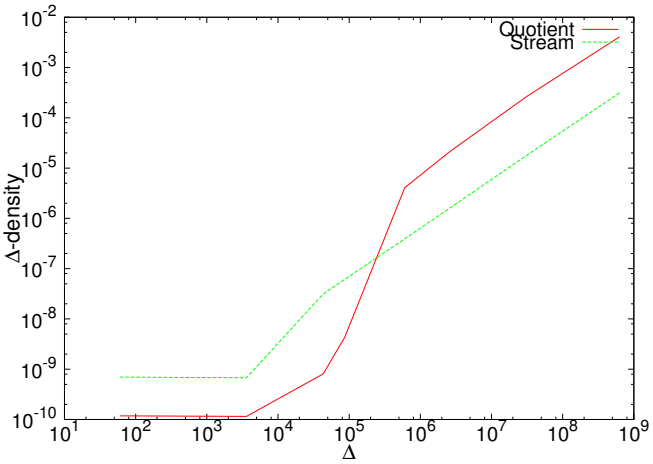


Fig. 9: Δ-density of the link stream and the quotient stream as a function of Δ, for $\Delta = 1mn, 1h, 12h, 1d, 37d, 30d, 1y$ and $20y$.

teracting continuously from b to e) instead of punctual links is a promising direction of work.

Acknowledgments. This work is supported in part by the French *Direction Générale de l'Armement* (DGA), by the Thales company, by the CODDDE ANR-13-CORD-0017-01 grant from the *Agence Nationale de la Recherche*, and by grant O18062-44430 of the French program *PIA – Usages, services et contenus innovants*.

References

1. Rémi Dorat, Matthieu Latapy, Bernard Conein, and Nicolas Auray. Multi-level analysis of an interaction network between individuals in a mailing-list. In *Annales des télécommunications*, volume 62, pages 325–349. Springer, 2007.
2. Santo Fortunato. Community detection in graphs. *Physics Reports*, 486(3):75–174, 2010.
3. Sulayman Sowe, Ioannis Stamelos, and Lefteris Angelis. Identifying knowledge brokers that yield software engineering knowledge in {OSS} projects. *Information and Software Technology*, 48(11):1025 – 1033, 2006.
4. SPI. Debian mailing-list archive: https://lists.debian.org/debian-user/.
5. Jimeng Sun, Christos Faloutsos, Spiros Papadimitriou, and Philip S. Yu. Graphscope: Parameter-free mining of large time-evolving graphs. In *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '07, pages 687–696, New York, NY, USA, 2007. ACM.
6. Jordan Viard and Matthieu Latapy. Identifying roles in an ip network with temporal and structural density. In *Computer Communications Workshops (INFOCOM WKSHPS), 2014 IEEE Conference on*, pages 801–806. IEEE, 2014.
7. Qinna Wang. Link prediction and threads in email networks. In *Data Science and Advanced Analytics (DSAA), 2014 International Conference on*, pages 470–476. IEEE, 2014.