Scalable Temporal Motif Densest Subnetwork Discovery

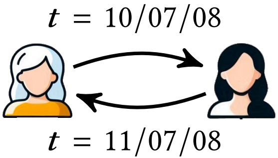
Ilie Sarpe[1,†], *Fabio Vandin*[2], *Aristides Gionis*[1]

[†]ilsarpe@kth.se

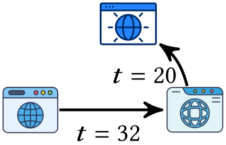
[1]KTH Royal Institute of Technology, [2]University of Padova

Temporal networks

Networks (or graphs) are powerful models used to capture pairwise relationships



(a) $E \rightarrow$ time of friendship



(b) $E \rightarrow$ time of link

Set of nodes $V = \{v_1, \ldots, v_n\} \rightarrow$ entities of the system

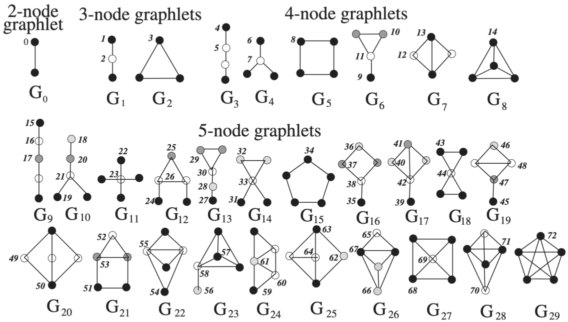
Set of (directed) edges $E = \{e_1, \ldots, e_m\} \rightarrow$ relationships among entities

Obs. Edges are of the form $e = (u, v, t) \in E$ with $u \neq v \in V$ and t is the *time* of (u, v)

We say that $G = (V, E)$ is a *temporal network*

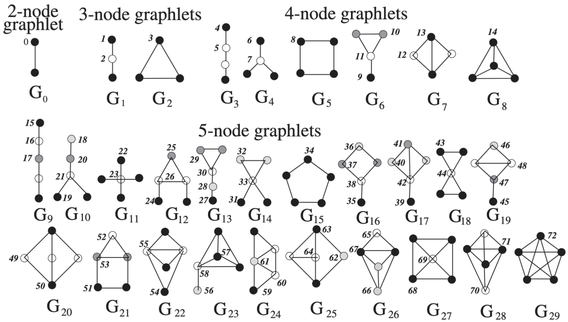
Static subgraphs and motifs

The analysis of *small patterns*, *subgraphs* or *motifs* is **fundamental** for our understanding of real graphs



Static subgraphs and motifs

The analysis of *small patterns*, *subgraphs* or *motifs* is **fundamental** for our understanding of real graphs



- Graphlet analysis for biological networks (Pržulj, 2007)
- Social network analysis (Yin et al., 2017)
- . . .

What about **temporal** networks?

Temporal network motifs

Patterns on temporal networks. (Paranjape et al., 2017; Kovanen et al., 2011)

Temporal motifs = static subgraphs + temporal ordering (+ additional information)

Where

(A) static subgraph captures topological properties in the data 

Temporal network motifs

Patterns on temporal networks. (Paranjape et al., 2017; Kovanen et al., 2011)

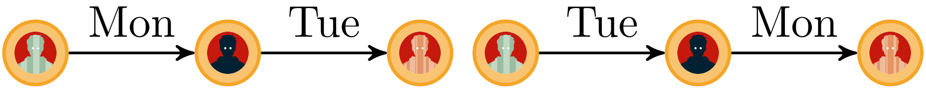
Temporal motifs = static subgraphs + temporal ordering (+ additional information)

Where

(A) static subgraph captures topological properties in the data 

(B) temporal ordering how time "flows"

Example. Information spreading along a path.



Temporal motifs by Paranjape et al. (2017)

A temporal motif is a pair $M = (K, \sigma)$ (Liu et al., 2019) where

- K is a directed and (weakly)connected multigraph with k-nodes and ℓ-edges

(a) K_1 (b) K_2

Temporal motifs by Paranjape et al. (2017)

A temporal motif is a pair $M = (K, \sigma)$ (Liu et al., 2019) where

- K is a directed and (weakly)connected multigraph with k-nodes and ℓ-edges

(a) K_1 (b) K_2

- σ is an ordering of the edges of K (modelling temporal dynamics of K)

  Example. Fixing $K = K_1$ then

(a) $\sigma_L = \langle (v_1, v_2), (v_2, v_3) \rangle$ (b) $\sigma_R = \langle (v_2, v_3), (v_1, v_2) \rangle$

Note. σ_L is time respecting while σ_R not!

Temporal motif occurrence

Given G and a value $\delta \in \mathbb{R}^+$, a *time-ordered* sequence $S = \langle (x'_1, y'_1, t'_1), \ldots, (x'_\ell, y'_\ell, t'_\ell) \rangle$ of ℓ unique edges from G is a δ-*instance* of $M = \langle (x_1, y_1), \ldots, (x_\ell, y_\ell) \rangle$ if

1. there exists a bijection h from the vertices appearing in S to the vertices of M, with $h(x'_i) = x_i$ and $h(y'_i) = y_i$, and $i \in [\ell]$;
2. the edges of S occur within δ time; i.e., $t'_\ell - t'_1 \leq \delta$.

Temporal motif occurrence

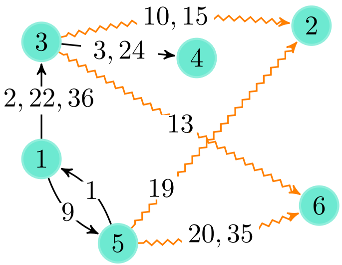
Given G and a value $\delta \in \mathbb{R}^+$, a *time-ordered* sequence $S = \langle (x'_1, y'_1, t'_1), \ldots, (x'_\ell, y'_\ell, t'_\ell) \rangle$ of ℓ unique edges from G is a δ*-instance* of $M = \langle (x_1, y_1), \ldots, (x_\ell, y_\ell) \rangle$ if

1. there exists a bijection h from the vertices appearing in S to the vertices of M, with $h(x'_i) = x_i$ and $h(y'_i) = y_i$, and $i \in [\ell]$;
2. the edges of S occur within δ time; i.e., $t'_\ell - t'_1 \leq \delta$.

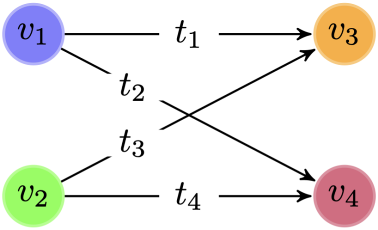
Def. Count of a temporal motif M is: $\#$ of δ-instances of M in G

Def. $\mathcal{S}_W$ set of temporal motif δ-instances among vertices in $W \subseteq V$

Temporal motif occurrence (example)

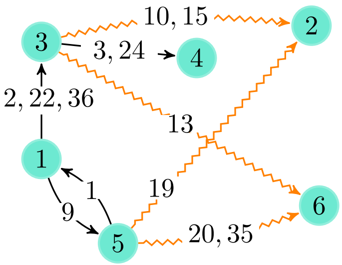


(a) Temporal graph G

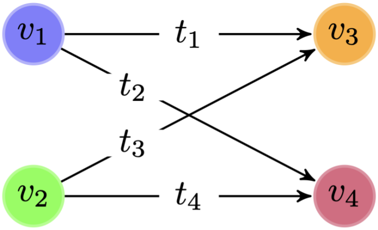


(b) Temporal motif, t_i is σ_i

Temporal motif occurrence (example)

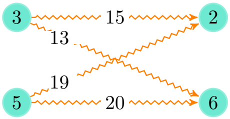


(a) Temporal graph G

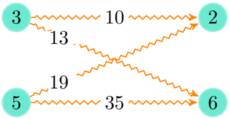


(b) Temporal motif, t_i is σ_i

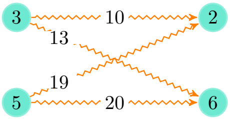
Fix $\delta = 10$



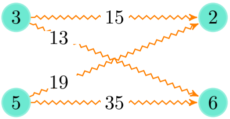
(a) ? ✗



(b) ? ✗



(c) ? ✓



(d) ? ✗

Counting temporal motifs

Typical problem addressed in literature.

Problem

Given a temporal network G, a temporal motif M and a parameter $\delta \in \mathbb{R}^+$ obtain *the count* of the temporal motif M

The problem is **NP**-hard

Counting temporal motifs

Typical problem addressed in literature.

Problem

Given a temporal network G, a temporal motif M and a parameter $\delta \in \mathbb{R}^+$ obtain *the count* of the temporal motif M

The problem is **NP**-hard

The (decision) problem is **NP**-hard even for motifs in **P** for static networks (Liu et al., 2019), star-shaped subgraphs!

⚠ **Scalability** and **efficiency** are key challenges for algorithms for temporal motifs.

Exact algorithms

(Mackey et al., 2018) *enumerates* all δ-instances of a fixed temporal motif *M*

(Pashanasangi and Seshadhri, 2021) Fast algorithms for temporal triangle counting based on degeneracy ordering

(Gao et al., 2022) improved algorithms for counting $\{2, 3\}$-node 3-edge temporal motifs

(Sarpe, 2023) improved (Mackey et al., 2018) by different matching criteria and timeline partition

(Yuan et al., 2023) dedicated hardware for counting temporal motifs

(Cai et al., 2023) exact algorithms for counting butterflies in temporal bipartite networks

(Li et al., 2024) exact algorithms based on temporal partitioning and matrix power to count 2-node $\{2, 3\}$-edge temporal motifs

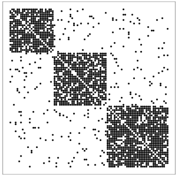
(Xia et al., 2025) efficient temporal triangle counting algorithms based on degeneracy ordering and tree-based data structures

...

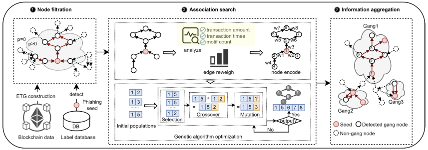
Temporal motifs (some) applications

Just a few applications, **extremely used** in practice! (Porter et al., 2022; Lei et al., 2020; Liu et al., 2024; Belth et al., 2020)

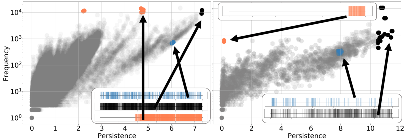
- SBMs and synthetic network generation



- Capture phishing communities

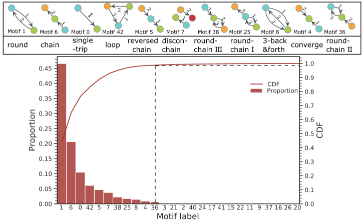


- Mining interesting events



(a) Boston Bike **(b)** Columbus Bike

- Analyze travel patterns



Enrich ML models and more...

Beyond temporal motif counts

Obs 1. *Counts* may not yield a complete view of temporal data (e.g., bursty events occurring between a small subset of nodes)

Obs 2. Counts do not identify important subnetworks $V' \subseteq V$ where instances of the motif occur

Beyond temporal motif counts

Obs 1. *Counts* may not yield a complete view of temporal data (e.g., bursty events occurring between a small subset of nodes)

Obs 2. Counts do not identify important subnetworks $V' \subseteq V$ where instances of the motif occur

We propose the new problem (Sarpe et al., 2024)

Problem – Temporal Motif Densest Subnetwork

Given a temporal network G, a temporal motif M, a parameter $\delta \in \mathbb{R}^+$, and a weighting function $\tau : \mathcal{S}_W \mapsto \mathbb{R}^+$ accounting for motif occurrences over $W \subseteq V$ obtain

$$W^* = \arg\max_{W \subseteq V} \frac{\tau(W)}{|W|} \ .$$

Temporal Motif Densest Subnetwork Discovery

Our objective

$$W^* = \arg\max_{W \subseteq V} \frac{\tau(W)}{|W|} \quad .$$

Some nice properties 😎

- Our new captures important subnetworks $W^* \subseteq V$ and realizing many occurrences of a fixed temporal motif M
- The user can flexibly select M and the weight assigned to its occurrences (τ)
- The user can pick the *time duration* δ of the motif occurrences

Applications. Temporal community detection, personalized advertisement, travels, and more...

Temporal Motif Densest Subnetwork Discovery

Our objective

$$W^* = \arg\max_{W \subseteq V} \frac{\tau(W)}{|W|} \quad .$$

Some nice properties 😎

- Our new captures important subnetworks $W^* \subseteq V$ and realizing many occurrences of a fixed temporal motif M
- The user can flexibly select M and the weight assigned to its occurrences (τ)
- The user can pick the *time duration* δ of the motif occurrences

Applications. Temporal community detection, personalized advertisement, travels, and more...

Cool but how do we solve the problem? 🤔

Challenges and scalable approach

Challenge 1.🤯 Even evaluating the objective function $\tau(W)$ for a subset $W \subseteq V$ is *challenging*, there are up to $\mathcal{O}(m^{\ell})$ occurrences M with ℓ edges!

Challenge 2.🤯 *Avoid expensive methods* such flow or ILP formulations $\rightarrow$ do not scale on large data.

Challenges and scalable approach

Challenge 1.🤯 Even evaluating the objective function $\tau(W)$ for a subset $W \subseteq V$ is *challenging*, there are up to $\mathcal{O}(m^{\ell})$ occurrences M with ℓ edges!

Challenge 2.🤯 *Avoid expensive methods* such flow or ILP formulations $\rightarrow$ do not scale on large data.

Def. *Temporal motif degree* of $v \in V$ is the $\#$ of temporal motif occurrences of a motif M in which v occurs. *Notation:* $d_v(M, G)$

Challenges and scalable approach

Challenge 1.🤯 Even evaluating the objective function $\tau(W)$ for a subset $W \subseteq V$ is *challenging*, there are up to $\mathcal{O}(m^{\ell})$ occurrences M with ℓ edges!

Challenge 2.🤯 *Avoid expensive methods* such flow or ILP formulations $\rightarrow$ do not scale on large data.

Def. *Temporal motif degree* of $v \in V$ is the $\#$ of temporal motif occurrences of a motif M in which v occurs. *Notation:* $d_v(M, G)$

Our solution. Rely on peeling approaches (developed for similar problems) + improve the scalability through *randomized* approximations 🎲

ALDENTE – randomized peeling

Our ALDENTE algorithms (`ProbPeel` and `HybridPeel`) leverage the following procedure.

1. Compute for each vertex $v \in V$, $\widehat{d}_v(M, G) \in (1 \pm \varepsilon) d_v(M, G)$
2. Set $V_1 \leftarrow V$; $G_1 \leftarrow G$
3. For $i \geq 1$ let V' be a *batch* of nodes from V_i with small degrees $\widehat{d}_v(M, G_i)$ (controlled by a parameter $\xi > 0$)
4. $V_{i+1} \leftarrow V_i \setminus V'$; update G_{i+1}
5. If $|V_{i+1}| < k$ **output** $\arg\max_{j \geq 1} \frac{\widehat{\tau}(V_j)}{|V_j|}$
6. Else estimate $\widehat{d}_v(M, G_{i+1}) \in (1 \pm \varepsilon) d_v(M, G_{i+1})$ for each $v \in V_{i+1}$

ALDENTE – randomized peeling

Our ALDENTE algorithms (`ProbPeel` and `HybridPeel`) leverage the following procedure.

1. Compute for each vertex $v \in V$, $\widehat{d}_v(M, G) \in (1 \pm \varepsilon) d_v(M, G)$
2. Set $V_1 \leftarrow V$; $G_1 \leftarrow G$
3. For $i \geq 1$ let V' be a *batch* of nodes from V_i with small degrees $\widehat{d}_v(M, G_i)$ (controlled by a parameter $\xi > 0$)
4. $V_{i+1} \leftarrow V_i \setminus V'$; update G_{i+1}
5. If $|V_{i+1}| < k$ **output** $\arg\max_{j \geq 1} \frac{\widehat{\tau}(V_j)}{|V_j|}$
6. Else estimate $\widehat{d}_v(M, G_{i+1}) \in (1 \pm \varepsilon) d_v(M, G_{i+1})$ for each $v \in V_{i+1}$

Obs. To compute $\widehat{d}_v(M, G) \in (1 \pm \varepsilon) d_v(M, G)$ we can use *any* state-of-the-art randomized sampling approach, for *local* motif approximation.

ALDENTE – guarantees

Guarantees (informal)

Our algorithms yield a solution $W = V_i$ for some $i \geq 1$ s.t. w.p. at least $1 - \eta$,

$$\frac{\tau(W)}{|W|} \geq \frac{1}{k} \frac{1}{(1+\xi)} \frac{(1-\varepsilon)^2}{(1+\varepsilon)^2} \frac{\tau(W^*)}{|W^*|} .$$

$\widehat{d}_v \in (1 \pm \varepsilon) d_v$ **and** $\widehat{\tau}(W) \in (1 \pm \varepsilon)\tau(W)$

Batch peeling

Greedy peeling + k-node motif-based objective (tight)

ALDENTE – guarantees

Guarantees (informal)

Our algorithms yield a solution $W = V_i$ for some $i \geq 1$ s.t. w.p. at least $1 - \eta$,

$$\frac{\tau(W)}{|W|} \geq \frac{1}{k} \, \frac{1}{(1+\xi)} \, \frac{(1-\varepsilon)^2}{(1+\varepsilon)^2} \, \frac{\tau(W^*)}{|W^*|} \; .$$

Table: ALDENTE algorithms

Name	Approximation	Parameters	Time Complexity
ProbPeel	$\frac{(1-\varepsilon)^2}{k(1+\xi)(1+\varepsilon)^2}$	$\xi > 0, \varepsilon, \eta \in (0,1)$	$\mathcal{O}\Big(r(\varepsilon,\eta)\Big(\hat{m}^\ell \log_{1+\xi}(n) + \frac{(1+\xi)n}{\xi}\Big)\Big)$
HybridPeel	$\frac{(1-\varepsilon)^2}{k(1+\xi)(1+\varepsilon)^2}$	$\xi > 0, \varepsilon, \eta \in (0,1), J > 0$	$\mathcal{O}\big(Jr(\varepsilon,\eta)\hat{m}^\ell + \gamma^{-1}k^3\, n^{k/\gamma} \log(n)\big)$

Experiments – Setup

Table: Networks used, $|E_G|$: static undirected edges, time-interval length δ.

	Network	n	m	$\lvert E_G \rvert$	Precision	Timespan	δ
Medium	Sms	44 K	545 K	52 K	sec	338 (days)	172.8 K
	Facebook	45.8 K	856 K	183 K	sec	1561 (days)	86.4 K
	Askubuntu	157 K	727 K	455 K	sec	2614 (days)	172.8 K
	Wikitalk	1 100 K	6 100 K	2 800 K	sec	2277 (days)	43.2 K
Large	Stackoverflow	2.6 M	47.9 M	28.1 M	sec	2774 (days)	172.8 K
	Bitcoin	48.1 M	113 M	84.3 M	sec	2585 (days)	7.2 K
	Reddit	8.4 M	636 M	435.3 M	sec	3687 (days)	14.4 K
	EquinixChicago	11.2 M	2 300 M	66.8 M	μ-sec	62.0 (mins)	50 K
	Venmo	19.1 K	131 K	18.5 K	sec	2091 (days)	-

Implementation. Code in C++17 with optimization, on a 72-core machine and 1TB RAM. Each experiment (5$\times$) executed for at most three hours and max RAM 200GB.

Goals.

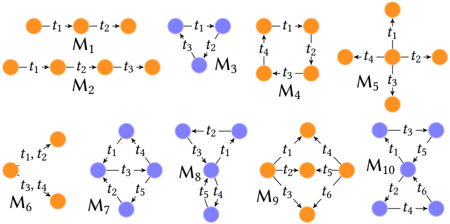
1. Show that ALDENTE is efficient, scalable and reports high quality solutions

2. Show that the TMDS captures important subnetworks

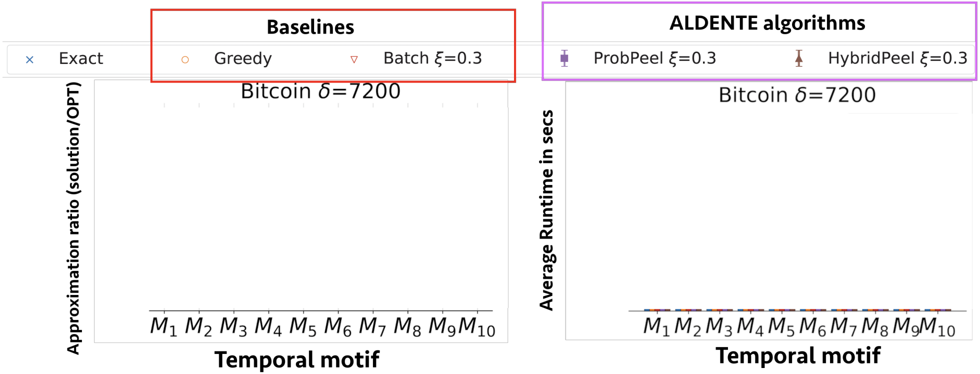
$+$ (a) Discuss memory efficiency; (b) test parameter space, and converge

Experiments – ALDENTE efficiency and scalability

Motifs.

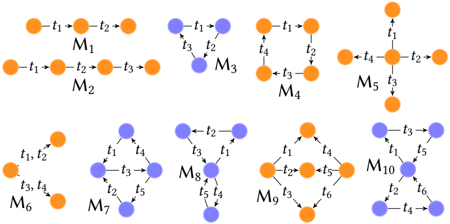


Algorithms. `Exact`, `Batch`, `Greedy`—`ProbPeel` and `HybridPeel`

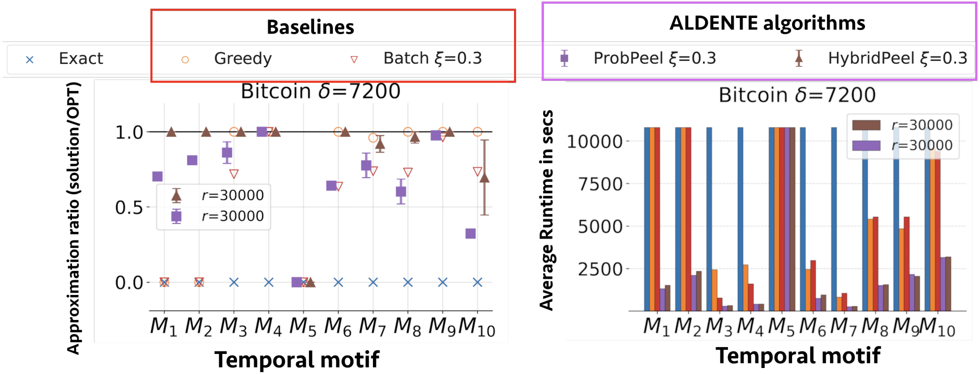


Experiments – ALDENTE efficiency and scalability

Motifs.



Algorithms. `Exact`, `Batch`, `Greedy`—`ProbPeel` and `HybridPeel`



Experiments – TMDS for discovering communities

Data. From Venmo platform $e = (u, v, t)$, user u transfer money to user v at time t + for each e the corresponding message to the transaction is *available*.

TMDS params. By using a star-shaped M and small $\delta = 7\,200$

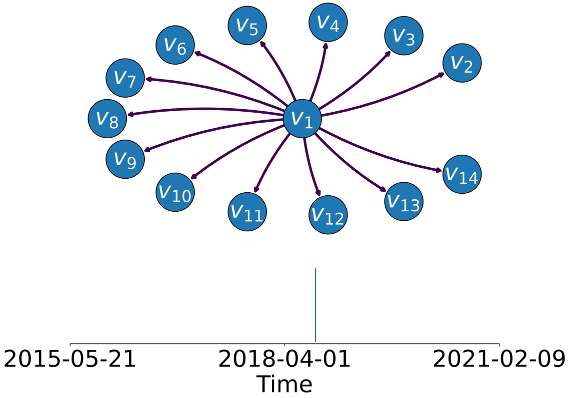


Figure: *The TMDS captures a bursty event.*

Wrap up

In our work

- We introduce a novel problem, to find the *Temporal Motif Densest Subnetwork*, capturing important properties in temporal data;
- We develop ALDENTE: two *efficient and scalable* algorithms to obtain high-quality solutions for the TMDS based on randomized sampling;
- We show empirically the *scalability of our ALDENTE* algorithms and the *usefulness of the proposed formulation*.

Many open challenges

- Theoretically control for *temporal evolution* of the optimal solution
- Control for the number of temporal motif occurrences in the optimal solution

Thank you for your attention!

- Our paper: https://arxiv.org/abs/2406.10608
- Our code: https://github.com/iliesarpe/ALDENTE

Feel free to reach out to discuss more

references I

Belth, C., Zheng, X., and Koutra, D. (2020). Mining persistent activity in continually evolving networks. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '20. ACM.

Cai, X., Ke, X., Wang, K., Chen, L., Zhang, T., Liu, Q., and Gao, Y. (2023). Efficient temporal butterfly counting and enumeration on temporal bipartite graphs. *Proceedings of the VLDB Endowment*, 17(4):657–670.

Gao, Z., Cheng, C., Yu, Y., Cao, L., Huang, C., and Dong, J. (2022). Scalable motif counting for large-scale temporal graphs. In *2022 IEEE 38th International Conference on Data Engineering (ICDE)*. IEEE.

Kovanen, L., Karsai, M., Kaski, K., Kertész, J., and Saramäki, J. (2011). Temporal motifs in time-dependent networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2011(11):P11005.

Lei, D., Chen, X., Cheng, L., Zhang, L., Ukkusuri, S. V., and Witlox, F. (2020). Inferring temporal motifs for travel pattern analysis using large scale smart card data. *Transportation Research Part C: Emerging Technologies*, 120:102810.

references II

Li, J., Qi, J., Huang, Y., Cao, L., Yu, Y., and Dong, J. (2024). Motto: Scalable motif counting with time-aware topology constraint for large-scale temporal graphs. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, CIKM '24, pages 1195–1204. ACM.

Liu, J., Chen, J., Wu, J., Wu, Z., Fang, J., and Zheng, Z. (2024). Fishing for fraudsters: Uncovering ethereum phishing gangs with blockchain data. *IEEE Transactions on Information Forensics and Security*, 19:3038–3050.

Liu, P., Benson, A. R., and Charikar, M. (2019). Sampling methods for counting temporal motifs. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, WSDM '19. ACM.

Mackey, P., Porterfield, K., Fitzhenry, E., Choudhury, S., and Chin, G. (2018). A chronological edge-driven approach to temporal subgraph isomorphism. In *2018 IEEE International Conference on Big Data (Big Data)*. IEEE.

Pan, Y., Bhalerao, O., Seshadhri, C., and Talati, N. (2024). Accurate and fast estimation of temporal motifs using path sampling. In *2024 IEEE International Conference on Data Mining (ICDM)*, pages 809–814. IEEE.

Paranjape, A., Benson, A. R., and Leskovec, J. (2017). Motifs in temporal networks. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*, pages 601–610. ACM.

references III

Pashanasangi, N. and Seshadhri, C. (2021). Faster and generalized temporal triangle counting, via degeneracy ordering. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, KDD '21. ACM.

Porter, A., Mirzasoleiman, B., and Leskovec, J. (2022). Analytical models for motifs in temporal networks. In *Companion Proceedings of the Web Conference 2022*, WWW '22. ACM.

Pržulj, N. (2007). Biological network comparison using graphlet degree distribution. *Bioinformatics*, 23(2):e177–e183.

Pu, J., Wang, Y., Li, Y., and Zhou, X. (2023). Sampling algorithms for butterfly counting on temporal bipartite graphs.

Sarpe, I. (2023). *Efficient and Rigorous Algorithms for the Analysis of Large Temporal Networks*. PhD thesis, University of Padova.

Sarpe, I. and Vandin, F. (2021a). oden: Simultaneous approximation of multiple motif counts in large temporal networks. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, CIKM '21. ACM.

Sarpe, I. and Vandin, F. (2021b). *PRESTO: Simple and Scalable Sampling Techniques for the Rigorous Approximation of Temporal Motif Counts*, pages 145–153. Society for Industrial and Applied Mathematics.

references IV

Sarpe, I., Vandin, F., and Gionis, A. (2024). Scalable temporal motif densest subnetwork discovery. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '24, pages 2536–2547. ACM.

Wang, J., Wang, Y., Jiang, W., Li, Y., and Tan, K.-L. (2020). Efficient sampling algorithms for approximate temporal motif counting. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, CIKM '20. ACM.

Xia, Y., Fang, Y., and Luo, W. (2025). Efficiently counting triangles in large temporal graphs. *Proceedings of the ACM on Management of Data*, 3(1):1–27.

Yin, H., Benson, A. R., Leskovec, J., and Gleich, D. F. (2017). Local higher-order graph clustering. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '17. ACM.

Yuan, Y., Ye, H., Vedula, S., Kaza, W., and Talati, N. (2023). Everest: Gpu-accelerated system for mining temporal motifs. *Proceedings of the VLDB Endowment*, 17(2):162–174.