Why we need a process-driven network analysis

Mareike Bockholt[1] and Katharina A. Zweig[1]

Department of Computer Science, Algorithm Accountability Lab,
TU Kaiserslautern, Gottlieb-Daimler-Straße 48, 67663 Kaiserslautern, Germany
{mareike.bockholt, zweig}@cs.uni-kl.de

Abstract. A network representation is a powerful abstraction of a complex system, on which a full range of readily available methods from network analysis can be applied. A network representation is suitable if indirect effects are of interest: if A has an impact on B and B has an impact on C, it is assumed that also A has an impact on C. This implies that some process is flowing through the network. For a meaningful network analysis, the network process, the network representation, and the applied network measure cannot be chosen independently [3, 4, 9, 30]. We propose a process-driven perspective on network analysis, which takes into account the network process additionally to the network representation. In order to show the necessity of this approach, we collected four data sets of real-world processes. As first step, we show that the assumptions of standard network measures about the properties of a network process are not fulfilled by the real-world process data. As second step, we compare the network usage pattern by real-world processes to the usage pattern of the corresponding shortest paths and random walks. Our results support the importance of a process-driven network analysis.

Keywords: network analysis, network processes, dynamics on networks, paths

1 Introduction

In the last two decades, there has been an increasing interest in the behaviour of complex systems, i.e., systems consisting of entities interacting with each other. Examples of complex systems include social systems of human interactions, biological systems of protein-protein interactions, or transportation systems as the world-wide air transportation system. A popular approach for analyzing such systems is using network analysis [1] where the observations of the system's behaviour are transformed into a graph structure: entities are represented by nodes and their interactions by edges. Having a network representation of a system allows the application of network analytic methods such as measuring the structure of the network, identifying groups of densely connected nodes [12], detecting network motifs [17], or finding the most central node [16].

In the network analysis community, the transformation of a system into a network representation seems natural by now, but it is actually a considerable simplification of the system which is rarely unique: for the same system, there is always more than one plausible and well-defined network representation [30].

Let us give a small example: Consider a data set containing a sample of passengers' tickets of domestic flights within the US (as introduced in Section 3). For each passenger's journey, the data set contains one entry for each non-stop flight connection of the journey, including start and destination airport of the sub-journey, airline, type and size of the airplane, etc. One can easily come up with at least a dozen of different plausible sounding network representations of the same data set: nodes might represent single airports or cities. An edge might be inserted if there is a single flight from one airport to the other, or if there are flights with a minimum volume, or at a regular basis, etc.

It has been shown that seemingly trivial decisions in creating the network representation have an impact on the structure of the network and on the results of the network analytic methods. Butts illustrated that different choices of node aggregations have a substantial impact on the fundamental properties of the resulting network [4]. Similarly, Choudhury experimented with different plausible edge definitions in an email communication network and demonstrated the large effect of different thresholds on the same data set [5]. Hence, the choice of the "right" network representation is crucial for the interpretability and relevance of the results of methods applied on the network representation.

A network is a convenient structure for a system representation if not only pairwise interactions between entities are of relevance. If only pairwise interactions are relevant, a list of separate dyadic relations is a sufficient representation. A network representation (and the applicable methods) is suitable if *indirect effects* are of interest: if there is an effect of entity A on entity B, and an effect of B on C, it is assumed that there is a kind of influence of A on C. The presence of indirect effects, however, implies that something is flowing through the network from node to node by following the edges. In a social system, a person can have an impact on a friend of a friend by forwarding a piece of information [13], by spreading a rumor [8] or by transferring a behavior [6]. In other systems, physical goods are transferred, diseases are spread [20], or humans use the system as infrastructure, such as passengers in transportation systems [14, 7].

We emphasize that only the presence of a *network process* makes a network representation meaningful. The presence of a network process is assumed by the majority of network measures. For example, the concept of graph distance is only meaningful if something is using those paths, hence, all metrics containing path lengths require the presence of a network process. As another example, the classic centrality indices degree, closeness and betweenness centrality were introduced by Freeman with the idea in mind that they measure a node's importance with respect to a specific process [10]. The measures, however, expect the process to be uniform in several aspects: the process is expected to be present at all nodes and edges with the same probability and same intensity, and to start and end in every node with the same probability. Thus, network analytic measures are using a static network representation and a simplified model of possible network flow processes instead of looking at the real network flow process.

We claim that any network analytic approach will benefit from a process-driven perspective where the network representation and the corresponding real process are both taken into account. We call this *process-driven network analysis*. This means that deducing a network representation and applying available network measures needs to be in accordance with the network process of interest.

For demonstrating the relevance of a process-driven network analysis, we collected four data sets containing process data and the corresponding network structure from different scenarios. We investigate whether the usage pattern of the network by the real-world processes are uniform as expected by standard network measures, i.e., whether the simplified model is a good proxy for the real-world process. Furthermore, standard network measures mostly expect the network process to move on shortest paths (e.g., closeness and betweenness centrality) or on random walks (e.g., random walk betweenness centrality [18], Google's PageRank [19], or community detection by WalkTrap [21]). We compare the network usage of the real-world process data to the usage of those extreme cases of process models. We show:

(i) Neither of the four real-world processes are uniform in node or edge usage. Similarly to real-world networks' degree distribution, there are a few hub nodes which are used many times, and a large number of nodes which are used once or never.

(ii) For neither of the real-world processes, it holds that all node pairs have the same probability of being start and target of the process.

(iii) Two of the data sets can be sufficiently approximated by a shortest-path-model, while two others cannot.

(iv) We simulate the real-world process by a set of random agents. Although, for a fair comparison, starting nodes and outreach potential of the agents are tied to the real-world process, the random agents do not reproduce the usage pattern of the real-world process, for three of the four data sets.

Structure of the paper Section 2 gives a short overview of related work. Section 3 describes the used data sets, before Section 4 investigates the usage pattern of the networks by the real process. Section 5 compares the network usage by the real process to its usage by shortest paths and random walks. The article concludes with Section 6.

Definitions and notations A graph $G = (V, E)$ consists of a node set V and an edge set $E \subseteq V \times V$. We consider directed graphs where the edges are ordered tuples. We associate a graph with a weight function $\omega : E \to \mathbb{R}$ assigning weights to the edges, an unweighted graph is associated with the trivial weight function $\omega(e) = 1 \forall e \in E$. A walk is an alternating finite sequence of nodes and edges $P = (v_1, e_1, v_2, \dots, e_{k-1} v_k)$ with $v_i \in V$ for $i \in \{1, \dots, k\}$ and $e_j = (v_j, v_{j+1}) \in E$ for $j \in \{1, \dots, k-1\}$. If the nodes and edges of P are distinct, we call P a path. If a node v is contained in a walk P, we write $v \in P$. The start and end node of a walk P are denoted by $s(P) = v_1$ and $t(P) = v_k$. The length of a walk P, is defined as $\omega(P) = \sum_{i=1}^{k-1} \omega(e_i)$. The distance from node v to node w, $d(v, w)$, is length of the shortest path from v to w.

2 Related work

The relevance of network processes for network analysis is not new. An important contribution was made by Borgatti [3] who identified two dimensions by which exemplary network process can be distinguished. He linked the identified process properties to existing centrality indices. Consider closeness centrality as an example: the closeness centrality for a node v is defined as the inverse of the sum of the shortest path lengths from every node to v. Having a high closeness value means that the node can be reached quickly from any other node (on average). Therefore, it is assumed that there is a process flowing through the network using shortest paths or by a broadcasting mechanism. Applying closeness centrality on a network with a process which has neither of those properties, e.g., the spread of a disease, yields non-interpretable results [3]. Dorn et al. [9] bring together Borgatti's insight that network process and network measure are dependent, and Butts' insight that network representation and network measure are dependent [4], and call this interdependence Trilemma of network analysis. This idea is elaborated on by Zweig [30].

Recent works [29, 25] have questioned the common practice of transforming process data (as described in our example above) into a network representation where an edge is inserted from node i to node j if the process data contains a connection from i to j, a so-called first-order network. In this representation, dependencies contained in the process data where the choice of the next node depends on the previously visited nodes, are lost. Xu et al. [29] argue that the network representation itself should reflect those dependencies and propose higher-order networks.

Approaches analyzing data sets of one specific process exist a lot, for example spreading of diseases [20, 23], spreading of rumors [8], propagation of health behaviors [6], or human navigation in information networks [28]. Approaches exploiting the information contained in the process data in order to draw insights about the network itself have been proposed by several authors: Weng et al. show the effect of an information diffusion process on the network evolution [27]. Rosvall et al. derive community structures in the network by incorporating real process data into a Markov chain simulation [24].

3 Data sets

In order to compare real-world processes with a shortest-path model and model based on random agents, we restrict our analysis to processes with a transfer mechanism, i.e., processes consisting of indivisible entities *moving* from node to node. Suitable data sets satisfies the following requirements and are described in the following paragraphs (see also Table 1): (i) it contains trajectories of process entities, i.e., a set of walks $\mathcal{P} = \{P_1, \dots, P_k\}$ (ii) the process trajectories can be mapped onto the network structure, (iii) the process entities have a target, (iv) which they try to reach as fast as possible.

Data set	Nodes	Edges	Process
Airline O&D Survey (DB1B) [22]	airports	non-stop airline connections	passengers
London Transport (LT) [26]	public transport stations	public transport connections	passengers
Wikispeedia (Wiki) [28]	Wikipedia articles	hyperlinks	players
Rush Hour (RH) [15]	configurations	valid game moves	players

Table 1: Overview of the used data sets.

Airline Origin and Destination Survey (DB1B) The Bureau of Transportation Statistics provides an Airline Origin and Destination Survey (DB1B) for every quarter year [22]. It contains a 10 % sample of airline tickets from reporting airline carriers within the US. For each itinerary, the data contains its start and destination airport and intermediate stops. The process of interest is passengers traveling by airplane in the network of airports. We use the data of the years 2010 and 2011. Passengers' journeys were split into outbound and return trip. We create a node for every airport, airports for which the journey data contains sub-itineraries done by bus or tram, are merged. A directed edge (v, w) is created if the data contains at least one itinerary with a flight connection from an airport in v to an airport in w. We consider an unweighted version of the network and a version with edges weighted with geographic distance between the airports.

London Transport (LT) Transport of London, a governmental authority responsible for public transport within the region of London, provides the Rolling Origin and Destination Survey [26], containing a 5 % sample of all passengers' journeys with an Oyster Card, an electronic ticket, during a week in November 2017. For each journey, the data contains its start and destination station as well as the stations of train changes. We used the Underground timetables to construct a multi-layer network where each layer represents one underground line: a node represents an underground station, an edge in layer i is inserted from v to w if w can be reached from v with line i without changing trains. Note that this does not yield a graph in the form of a chain as a line plan would suggest, but the transitive closure of the chain. An edge (v, w) in layer i is weighted with the minimal travel time from v to w using line i. Note also that we did not take into account the time schedules of the lines. A one-layer network is constructed by merging the layers into one and taking the minimal edge weight.

Wikispeedia (Wiki) West et al.[28] provide logs of persons playing Wikispeedia. In this game, a player is given (or chooses) two Wikipedia articles and the goal is to navigate from the start article to the target article by following the hyperlinks. West collected more than 50 000 logs by offering the game on his web page. The node set is a subset of Wikipedia articles, directed edges are hyperlinks between articles. We consider only walks reaching the target and exclude moves revoked by the player via an Undo button.

Rush Hour (RH) This data set also contains game logs of players, attempting to solve an instance of a Rush Hour game. This single-player sliding-block puzzle consists of a board with 6×6 cells and a designated exit, representing a parking lot. Cars (blocks of width of one cell and of length of two or three cells) are placed on the board horizontally or vertically. Goal is to move the cars (forwards or backwards, not sideways) such that a designated target car can exit the board. The network is the state space of the game instance, containing all board configurations reachable from the start configuration by valid moves, and an undirected edge represents a valid move. The game logs were collected by Jarušek and Pelánek by their web-based tool for education [15]. Three different game instances were used for our analysis: game A is a very easy instance with an optimal solution length 3 while games B and C are of medium difficulty with an optimal solution length 11 and 13, respectively. Only walks ending in a solution configuration are considered.

Data set	$\lvert V\rvert$	$\lvert E\rvert$	$\lvert\mathcal{P}\rvert$	Path length		Coverage
				Range	Average	
DB1B	462	12499	$86m$	$[42, 26922]$	1909	100 %
LT	268	13173	$4.8m$	$[1, 107]$	16.3	100 %
Wiki	4592	119804	51306	$[1, 82]$	5	90 %
RH Game A	364	1524	3044	$[3, 33]$	5	63 %
RH Game B	6769	33142	1965	$[11, 59]$	15	11 %
RH Game C	830	4037	1472	$[13, 95]$	26	53 %

Table 2: Properties of the used data sets. $|V|$ and $|E|$ denote the cardinality of node and edge set of the underlying graph, $|\mathcal{P}|$ the number of available walks.

4 Uniformity of network usage

Our first goal is to investigate whether a process-driven network analysis is a necessary approach at all. Most network measures which assume the presence of a network process, expect the process to be uniform. Consider betweenness centrality as an example: for a node v, the betweenness centrality is defined as $c_B(v) = \sum_{s,t\in V, s\neq v\neq t} \frac{\sigma_{st}(v)}{\sigma_{st}}$ where σ_{st} is the number of shortest paths from s to t and $\sigma_{st}(v)$ the number of shortest paths from s to t containing v, and with the convention $\frac{0}{0} = 0$. Several assumptions are included in this measure: (i) It is assumed that a process is flowing through the network using shortest paths (see also Borgatti[3]). (ii) It is summed over all node pairs (excluding a few), therefore assumed that there is process flow between each pair of nodes. (iii) Each node pair can contribute a value between 0 and 1 to the betweenness value, hence each node pair is considered as equally important. It needs to be tested if that simple model is a good proxy for the real flow behaviour.

For the four used data sets, however, it turns out–unsurprisingly–that those assumptions are not met. Table 2 shows the coverage of the networks by the corresponding process, i.e., the fraction of nodes which are visited by the process at least once. Except for London Transport and DB1B where the network was constructed from the process data, between 10 and 89 % of the nodes were not visited by the process. We compute the usage $nu(v) = |\{P \in \mathcal{P} | v \in P\}|$ of each node v, i.e., the number of process walks in which the node is contained in.

Figure 1 shows the cumulative distribution of the node usage for each process data set. We find that for all data sets, there are a few nodes which are used by many process entities, while the majority of the nodes is used at most once by the process entities. This disproves–for the used data sets–the assumption that real-world processes are present in all nodes with the same probability and intensity.

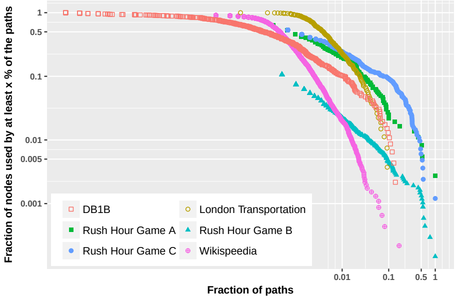


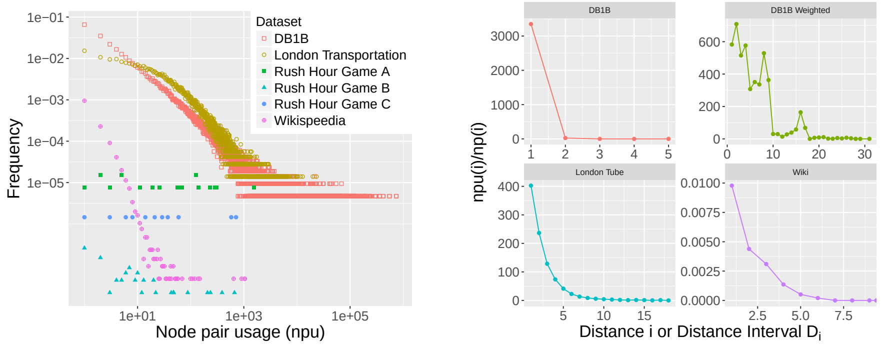
Fig. 1: Relative node usage of the real network processes.

We perform the same analysis for node pairs: for each node pair $(s,t) \in V \times V$, we count how many process entities start in s and end in t (referred to as *node pair usage*, *npu*). We make a similar observation as before: only a fraction (between 42 % and 46 % for DB1B and LT, 0.14 % for Wiki, and less than 0.01 % for the RH games) of all node pairs is used as source and target of the process. While this is not surprising for the RH games since all players start in the same node and end in one of the solution nodes, this is less expected for the two transportation systems, DB1B and LT. The distribution of the node pair usage (of pairs with $npu > 0$) is shown in Figure 2a. We find that for the RH games, the frequency of different node pair usage values (with $npu > 0$) is approximately the same. For Wiki and the transportation systems, the majority of node pairs is used rarely as source and destination, i.e, having a low node pair usage, while there are a few node pairs with a high node pair usage.

For those three data sets, we investigate which node pairs are used more often than expected (assuming each node pair was chosen as start and end with the same probability). For this purpose, we consider the node pairs of the network separately by distance. For a graph distance i, let $np(i) = |\{(s,t) \in V \times V | d(s,t) = i\}|$ be the number of node pairs in the graph with this distance, and let $npu(i) = |\{P \in \mathcal{P} | d(s(P), t(P)) = i\}|$ the number of process entities with distance i between start and end node. Note that we do not consider the length of the *process walk*, but of the *shortest path*. For the weighted network versions (DB1B with edges weighted by geographic distance, and LT with edges weighted by travel time), we introduce distance intervals of size 500 km and 5 minutes, respectively, and define $np(i)$ and $npu(i)$ for a distance interval i accordingly.

Figure 2b shows the node pair usage $npu(i)$ divided by the number of node pairs $np(i)$ for each distance (or distance interval). We observe that the node pair usage is dependent on the graph distance of the pairs: for each data set, closer nodes are much more used as source and target than nodes which are further apart–far more than expected if node pairs were picked uniformly at random.

This observation might not be surprising for transportation systems, but the same observation has also been done in other networks. Friedkin considered communication networks and found a *horizon of observability* [11]: a distance



(a) Source-target-frequency (normalized by the total number of node pairs) on a log-log-scale.

(b) Node pair usage normalized by the number of node pairs, by distance (or distance interval) of the nodes.

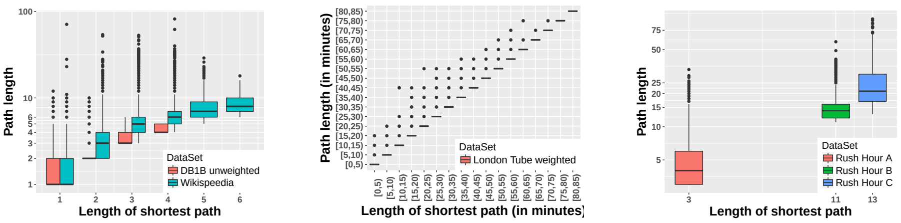
Fig. 2: Source-target-frequency of the real network processes.

in communication networks beyond which persons are unlikely to be aware of the existence of another. Also for the data sets used here, there seems to be a horizon of reachability: a distance beyond which there is only a negligible (or no) amount of process flow. All findings, (i) not all nodes are relevant for the network process, (ii) a few nodes are used heavily while most nodes are used only a few times, (iii) especially close nodes are source and target of network processes, could have been expected for the used data sets. They, however, do have consequences for network analysis in general: network analytic measures pretend a homogeneity of the network process–each node, each edge, each node pair is considered to be of the same quality. Our analysis shows that they are not if the real-world network process is taken into account. We argue that the properties of the network process need to be taken into account when deducing the network representation and applying network analytic methods on it.

5 Models of processes

The last section showed that the network processes at hand do not use the underlying network uniformly. This is in contradiction to the implicit assumptions of most network measures which expect a uniform network process. The two extreme possibilities of simulating real network processes are shortest paths and random walks. We are going to compare the real trajectories with both extremes.

Processes as shortest paths For comparing a real process walk P to its corresponding shortest path, we compare its length to the length of the shortest path from $s(P)$ to $t(P)$. Figure 3 shows that for the transportation processes DB1B and LT, on average, the length of the real walks is close to the length of the shortest path. This is not true for the game data sets. For the RH games, the lengths of the real walks strongly depend on the game instance. For game game B and C, only 14 % (game B) and 2 % (C) of the real walks have the same length



(a) Wiki and DB1B (b) LT (c) RH games

Fig. 3: Path lengths of different network processes compared to the length the shortest path.

as the optimal path. The real walks of the Wikispeedia game are also longer than their corresponding shortest paths, but on average only by one step.

Processes as random walks In contrast to shortest paths, the other extreme of a model for process trajectories are simulations by agent-based random walks: an agent starts at a node, moves to a randomly chosen neighbor node, and continues this procedure until a stopping criterion is reached. In order to compare the real trajectories to the trajectories of the random agents, the random agents are restricted by the real trajectories in the following way: for each real trajectory $P \in \mathcal{P}$, a random agent starts in $s(P)$, and performs a random walk as long it has not exceeded the length of P. By this procedure, it is made sure that the number of agents is equal to the number of real process entities, the node usage of the start nodes is equal for real and random agents, and the random agents have the same "outreach potential" as the real trajectories.

We implement two variants of neighbor choices: the agent picks uniformly at random from all neighbors of the current node (*uniform neighbor choice*), or from the neighbors except of the directly previously visited one (*backwards-restricted neighbor choice*). For data set LT, a second variant for the length restriction is implemented: the agent continues its walk until it has reached the same number of line changes as contained in the real trajectory. For each data set and its walk set $\mathcal{P}$, sets of random walks are created by repeating the above procedure $N = 500$ times. For the data sets DB1B and LT, we sample a subset of 0.1 % (DB1B) and 10 % (LT) of all real trajectories and tie (and compare) the random walks to those subsets.

Fig. 4 shows the cumulative node usage by the real-world process and by the random agents (for each node the mean value over the 500 iterations is used). We also compare the node usage by the real process and by the random walks for each single node (see Fig. 5). We observe that for DB1B and LT, the random agents and the real entities yield a different usage pattern. On the same time, there is a high correlation of node usage by the real and random trajectories (Pearson correlation coefficient between 0.81 and 0.87). For the RH games, the findings are opposite: the node usage distribution is similar for random and real trajectories while having a lower correlation (Pearson correlation coefficient 0.77

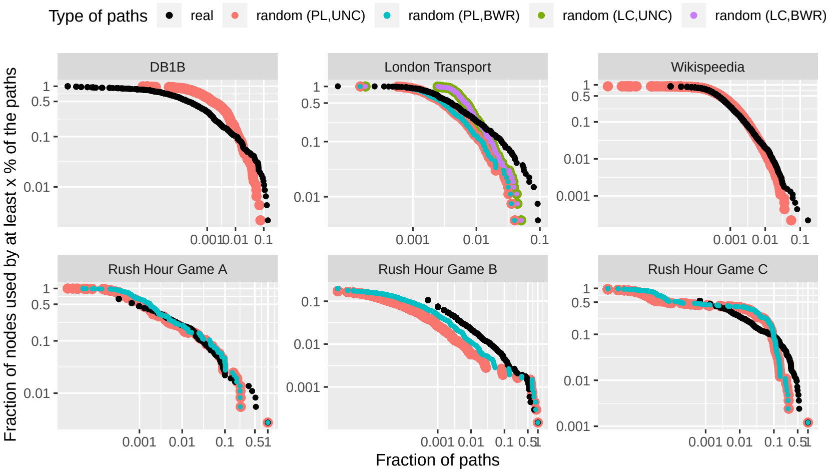


Fig. 4: Cumulative node usage distribution of real and random trajectories. The random walks implement a uniform neighbor choice (UNC) or a backwards-restricted neighbor choice (BWR), their length is restricted by the length (PL) or by the number of line changes of the corresponding real trajectory (LC).

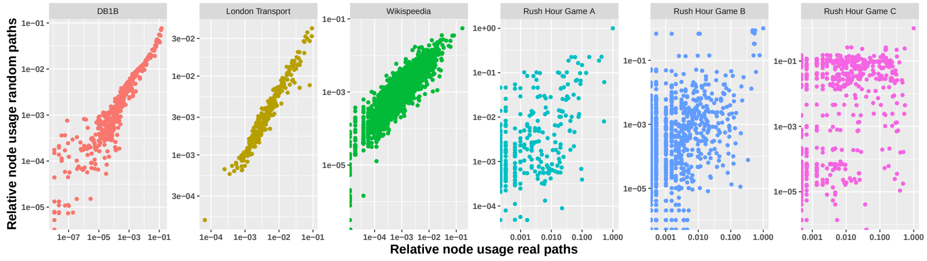


Fig. 5: Node usage of the real paths vs mean node usage of the random walks (UNC) on a log-log scale. Node usage is normalized by the number of (real or random) agents.

(A), 0.64 (B) and 0.36 (C)). On the same time, for all data sets, it is the same set of nodes which is used the most often by real or random trajectories.

6 Conclusion and future work

A network representation is often a convenient abstraction for a system if indirect effects are of interest. This implies that a process is flowing through the network. A meaningful network analysis needs to take into account the properties of these processes. We call this perspective *process-driven network analysis*. In this work, we collected four data sets containing processes flowing on walks towards a goal. We find that none of those processes shows a uniform usage pattern, i.e., a few nodes are used very often by the process, while most nodes are used at most once. The same holds for node pairs being source and target of the processes. This has consequences for network measures expecting a uniform usage of the network. We furthermore compared the real-world processes to basic simulations, i.e., simulations by shortest paths and by random walks. We find that the transportation paths are, as expected, close to shortest paths in terms of length, while the game paths are not. Random walks where the source and potential outreach are fixed by the real-world process are able to show a similar node usage ranking for three of the four data sets, their node usage distribution is different though. This has consequences for network measures expecting a process moving on shortest paths or randomly through the network.

In order to take these insights into account, the following approaches might be applicable: If data of real-world processes are available, a network of higher order can be constructed and used for analysis [25]. Another option is the incorporation of real-world process data into existing network measures [2]. For considering the horizon of observability of processes [11], it might be a valid approach to separately analyze subgraphs of the network instead of the complete network. In general, we argue for a careful and thoughtful application of network measures on network representations.

References

1. S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D.-U. Hwang. Complex networks: Structure and dynamics. *Physics Reports*, 424(4-5):175–308, 2006.
2. M. Bockholt and K. A. Zweig. Process-driven betweenness centrality measures. In *Network Intelligence Meets User Centered Social Media Networks*, Lecture Notes in Social Networks, pages 17–33. Springer, 2018.
3. S. P. Borgatti. Centrality and network flow. *Social Networks*, 27(1):55–71, 2005.
4. C. T. Butts. Revisiting the foundations of network analysis. *Science*, 325(5939):414–416, 2009.
5. M. D. Choudhury, W. A. Mason, J. M. Hofman, and D. J. Watts. Inferring relevant social networks from interpersonal communication. In *Proceedings of the 19th international conference on World wide web - WWW '10*. ACM Press, 2010.
6. N. A. Christakis and J. H. Fowler. The collective dynamics of smoking in a large social network. *New England Journal of Medicine*, 358(21):2249–2258, 2008. PMID: 18499567.
7. V. Colizza, A. Barrat, M. Barthélemy, and A. Vespignani. The role of the airline transportation network in the prediction and predictability of global epidemics. *Proceedings of the National Academy of Sciences*, 103(7):2015–2020, 2006.
8. M. De Domenico, A. Lima, P. Mougel, and M. Musolesi. The anatomy of a scientific rumor. *Scientific reports*, 3:2980, 2013.
9. I. Dorn, A. Lindenblatt, and K. A. Zweig. The trilemma of network analysis. In *Proceedings of the International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 9–14, 2012.
10. L. C. Freeman. A set of measures of centrality based on betweenness. *Sociometry*, pages 35–41, 1977.
11. N. E. Friedkin. Horizons of observability and limits of informal control in organizations. *Social Forces*, 62(1):54–77, 1983.

12. M. Girvan and M. E. J. Newman. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences*, 99(12):7821–7826, 2002.
13. A. Guille, H. Hacid, C. Favre, and D. A. Zighed. Information diffusion in online social networks: A survey. *SIGMOD Rec.*, 42(2):17–28, 2013.
14. R. Guimerá and L. A. N. Amaral. Modeling the world-wide airport network. *The European Physical Journal B*, 38(2):381–385, 2004.
15. P. Jarušek and R. Pelánek. Analysis of a Simple Model of Problem Solving Times. In *Intelligent Tutoring Systems*, volume 7315 of *Lecture Notes in Computer Science*, pages 379–388. Springer, 2012.
16. D. Koschützki, K. A. Lehmann, L. Peeters, S. Richter, D. Tenfelde-Podehl, and O. Zlotowski. Centrality indices. In U. Brandes and T. Erlebach, editors, *Network Analysis*, volume 3418 of *Lecture Notes in Computer Science*, pages 16–61. Springer, 2005.
17. R. Milo. Network motifs: Simple building blocks of complex networks. *Science*, 298(5594):824–827, 2002.
18. M. E. Newman. A measure of betweenness centrality based on random walks. *Social networks*, 27(1):39–54, 2005.
19. L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford InfoLab, 1999.
20. R. Pastor-Satorras and A. Vespignani. Epidemic spreading in scale-free networks. *Physical Review Letters*, 86(14):3200–3203, 2001.
21. P. Pons and M. Latapy. Computing communities in large networks using random walks. In *Computer and Information Sciences*, pages 284–293. Springer, 2005.
22. RITA TransStat. Origin and Destination Survey database (DB1B), 2016. https://www.transtats.bts.gov/DatabaseInfo.asp?DB_ID=125.
23. L. E. C. Rocha, F. Liljeros, and P. Holme. Simulated epidemics in an empirical spatiotemporal network of 50,185 sexual contacts. *PLoS Computational Biology*, 7(3):e1001109, 2011.
24. M. Rosvall, A. V. Esquivel, A. Lancichinetti, J. D. West, and R. Lambiotte. Memory in network flows and its effects on spreading dynamics and community detection. *Nature communications*, 5, 2014.
25. I. Scholtes. When is a network a network? In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '17. ACM Press, 2017.
26. Transport for London. Rolling Origin and Destination Survey (RODS), 2017. http://www.tfl.gov.uk/info-for/open-data-users/our-feeds.
27. L. Weng, J. Ratkiewicz, N. Perra, B. Gonçalves, C. Castillo, F. Bonchi, R. Schifanella, F. Menczer, and A. Flammini. The role of information diffusion in the evolution of social networks. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '13, pages 356–364. ACM Press, 2013.
28. R. West and J. Leskovec. Human wayfinding in information networks. In *Proceedings of the 21st international conference on World Wide Web*, pages 619–628. ACM Press, 2012.
29. J. Xu, T. L. Wickramarathne, and N. V. Chawla. Representing higher-order dependencies in networks. *Science Advances*, 2(5):e1600028, 2016.
30. K. A. Zweig. *Network Analysis Literacy.* Springer, 2016.