Applied Network Science

RESEARCH Open Access

Towards a process-driven network analysis

Mareike Bockholt* and Katharina Anna Zweig

*Correspondence:
mareike.bockholt@cs.uni-kl.de
Algorithm Accountability Lab,
Department of Computer Science,
University of Kaiserslautern,
Kaiserslautern, Germany

Abstract

A popular approach for understanding complex systems is a network analytic one: the system's entities and their interactions are represented by a graph structure such that readily available methods suitable for graph structures can be applied. A network representation of a system enables the analysis of *indirect effects*: if A has an impact on B, and B has an impact on C, then, A also has an impact on C. This is often due to some kind of process flowing through the network, for example, pieces of informations or viral infections in social systems, passenger flows in transportation systems, or traded goods in economic systems. We argue that taking into account the actual usage of the system additionally to the static network representation of the system can yield interesting insights: first, the network representation and applicable network methods cannot be chosen independently from the network process of interest (Borgatti 2005; Dorn et al. 2012; Zweig 2016; Butts 2009). Therefore, focussing on the relevant network process in an early stage of the research project helps to determine suitable network representations and methods in order to obtain meaningful results (we call this approach *process-driven network analysis*). Second, many network methods assume that the spreading of some entity follows shortest or random paths. However, we show that not all flows are well approximated by this. In these cases, incorporating the network usage creates a real addition of knowledge to the static aggregated network representation.

Note: This is an extended and revised version of a conference article (Bockholt and Zweig 2019), published and presented at COMPLEX NETWORKS 2019.

Keywords: Network process, Process models, Network representation, Human navigation

Introduction

In the past two decades, the interest in complex systems has risen tremendously. Examples of complex systems include social systems of humans, biological systems of protein-protein-interactions, or transportation systems as the world-wide air transportation system. A popular and often natural approach for analyzing such systems is using network analysis (Boccaletti et al. 2006; Brandes et al. 2005) (see also Fig. 1): The interactions in the system are observed, these observations are then used to create a network representation of the system, i.e., a graph structure. For the network representation of

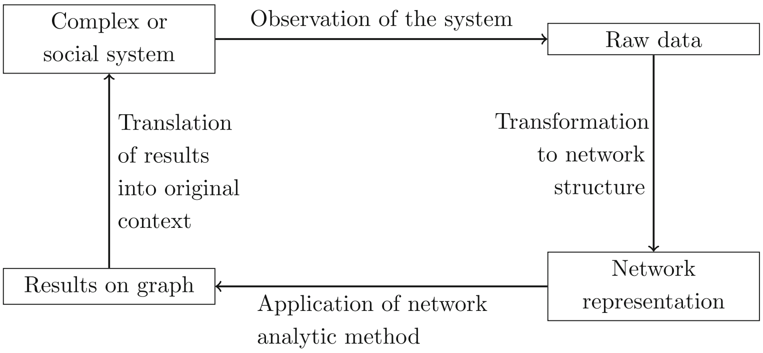

Complex or social system
Observation of the system
Raw data
Transformation to network structure
Network representation
Application of network analytic method
Results on graph
Translation of results into original context

Fig. 1 Phases of a network analytic project (Figure adopted from (Zweig 2016))

the system, where the nodes represent the actors or entities of the system and the edges represent the interactions between them, many methods are available, for example for analyzing the structure of the network, for predicting its future evolution (Liben-Nowell and Kleinberg 2007), for grouping the nodes into clusters (Girvan and Newman 2002), for detecting network motifs (Milo 2002), or for finding the most central node (Koschützki et al. 2005). For the interpretation of the network analytic results, it is assumed that they can be transferred to the real-world system and conclusions can be drawn for the real-world system as well (cf. Fig. 1).

Transforming a complex systems into a network is a strong simplification of the system: a system consisting of actors with possibly different properties interacting in different ways is transformed into a very restrictive mathematical structure, thereby creating a model of the real-world system. Simplifying the system in such a way has two sides: when appropriately chosen, such a model of a system is able to catch the essential properties of a system while being simpler and easier to handle than the real system. On the other hand, a simplification of a system naturally comes along with the loss of information. Hence, the choice of the "right" simplification is crucial for the interpretability and relevance of any results from methods used on the graph representation.

There is rarely a unique way for transforming a system into a graph representation and by choosing the transformation, implicit (or explicit) assumptions about the system are included into the representation (Zweig 2016). Already the choice of which entities of the system will be represented as nodes in the graph and which interactions of the entities in the system will be represented as edges in the graph, is not obvious: it has been shown that already different choices of levels of aggregation for the nodes will yield different results in the subsequent analysis of the network (Butts 2009). The same holds for the decision when an edge is inserted in the network (Butts 2009; De Choudhury et al. 2010).

For example, "Data sets" section will introduce a data set containing passengers' tickets of domestic flights within the US. For each passenger's journey, the data set contains an entry for each single non-stop flight of the journey, including the information of start and destination of that flight, airline, type and size of airplane, etc. Extracting a network representation from this data set can be done in various ways: nodes might represent single airports or cities (including several close airports which can also be derived in various ways). An edge from node v to node w might be included if the data contains a single flight from v to w, or if there are flights on a regular basis from v to w, or if there are flights of a certain volume, or other definitions might be reasonable as well. The edges might be unweighted, weighted by distance of the airports or flight time, they might have a capacity of available seats, or might be temporal if the flight schedule is taken into consideration. The network might consist of several layers where each layer contains the flights of a single airline, or the connections of all airlines might be merged into a single-layer network. This illustrates that the same data set can lead to a large number of different possible network representations. It is clear that a different network representation will yield different results when analyzing the network (Butts 2009; De Choudhury et al. 2010).

How to determine the most reasonable network representation? This raises the question whether there is a most reasonable network representation and how to determine it. If that is not the case, the question is whether there are at least some choices that can be ruled out. In order to answer this question, it is useful to consider when a network is a useful representation for a system at all: Network representations enable the analysis of indirect effects. When representing a system as a network, the main underlying assumption (also for all network analytic methods) is *some commonality mediated by indirect connections*: by connecting A to B and B to C in the network representation, we are not only assuming some commonality between A and B, and between B and C, but also some commonality between A and C via the indirect connection. For some systems, we can phrase the term of *commonality* as effect or influence: In social systems, a person can have an impact on the friend of a friend, in neural networks, the activity of a neuron does not only affect the directly connected ones, but also the ones connected via an intermediate neuron. If there are only direct effects, the representation as a network does not offer any additional insights and other data structures are more efficient. Moreover, a network representation of a system in which only direct effects are expected might seduce to apply analysis methods to it that yield basically uninterpretable results.

The presence of indirect effects or the commonality mediated by indirect connections can be caused by a (partly) transitive relationship or by network processes on top of the network structure: in the latter case, "something" is flowing through the network from node to node by following the edges: A person in a social network has an impact on the friend of a friend by–for example–a piece of information flowing through the network, by a rumor being spread through it, or by a trend which is transferred from one person to another. In other networks, examples for processes are physical goods being transfered from node to node, such as parcels or other items, diseases, such as HIV or influenza, computer viruses being spread from computer to computer, but also humans or other agents using the network as infrastructure, such as passengers in transportation networks. A *network process* is thus a model of how something uses a specific relationship represented by a network to spread in it. The spreading happens by a propagation from node to node by using the existing relationship between them. We differentiate between the terms network process and network flow: while the network process is the *model* of something using the network, the *network flow* is the actually observable dissemination of some entities through the network, yielding a trajectory for each entity. Figure 2 illustrates the used terminology.

Towards a process-driven network analysis Several well-known network analytic methods assume the presence of such network processes: metrics containing path length,

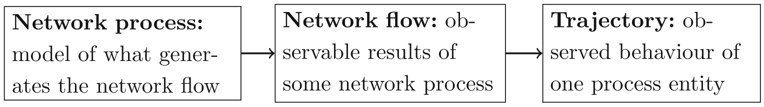

Network process: model of what generates the network flow
Network flow: observable results of some network process
Trajectory: observed behaviour of one process entity

Fig. 2 An overview of the used terminology

such as average shortest distance or diameter of the graph, assume something actually using those paths. This is particularly the case for centrality indices. The classic centrality indices–degree, closeness, and betweenness centrality–were introduced by Freeman in 1977 with the idea in mind that they measure a node's importance with respect to a specific process: "Thus, the use of these three measures is appropriate only in networks where betweenness may be viewed as important in its potential for impact on the process being examined." Freeman (1977) on the other hand, for other network methods, the presence of a network process is not necessary: Clustering methods are supposed to find groups of nodes assuming that nodes within a group do have a larger impact on each other than on nodes outside of their group. This can be caused by a network process, but also by a (partly) transitive relationship. In the following, we, however, focus on network methods which do assume the presence of a network process, such as distance-based measures and centrality indices.

Borgatti (2005) linked the properties of network processes to the well-known centrality measures. He argues that each centrality measure contains implicit assumptions about the properties of a network process, and the application of a centrality measure only yields interpretable results if the measure's assumptions match with the actual properties of the process of interest: a centrality measure quantifies the importance of a node with respect to a network process with the suitable properties. Consider betweenness centrality as an example: for calculating the betweenness centrality of a node v, for each node pair $(s, t) \in V \times V$, it is counted on which proportion of shortest paths from s to t, the node v is contained in, and these proportions are summed up for all node pairs (s, t). By counting shortest paths, the betweenness centrality can only rate the nodes' importance for a process using shortest paths. Furthermore, by considering the *proportion* of shortest paths a node v is contained in, it is assumed that the corresponding network process can only take *one* shortest path at a time, and not several simultaneously. This implies that it is assumed that the network process consists of indivisible entities. Applying betweenness centrality on a network for finding the most important node for a process which has neither of those properties, e.g., the spread of gossip, yields uninterpretable results (Borgatti 2005).

Note, however, that there are multiple application scenarios for centrality measures. The named assumptions are particularly valid if a centrality index is used for quantifying a node's importance. Centrality measures might also be seen as a description of the network topology. For calculating the closeness centrality of a node v in a network, the graph theoretic distances to v from all other nodes are summed up. This allows both, the rating of the nodes by their importance with respect to a process using shortest paths, and a quantification of a node's position in the network. For the former, the existence of a process is assumed, for the latter, this is not necessary.

Not only the applicable network measures are tied to the network process of interest, but also the network representation needs to match to the network process: the network representation needs to be chosen in a way that it represents the relationship *relevant for the process*. This concerns the type of the relationship as well as its time horizon. When the process of spreading a disease is in the focus, the corresponding network needs to contain edges which represent relations that are essential for passing on this disease. Analyzing the spread of influenza in a social network where edges represent a friendship relationship will give results which are difficult to interpret. Furthermore, the time horizon of the relationship and the process need to be in accordance: considering a relationship which exists in a scale of days, and a process which dissemination is a matter of years, will also give results hard to interpret. This is especially true (and naturally given) if a network representation is built from flow data, but is also the case when the network representation is constructed from other type of data. Consider the example of road traffic: here, the network representation can be obtained from satellite data or existing maps which is independent of the process *data*. Nevertheless, also in this case, the network *representation* is dependent on the process of interest: if we are interested in traffic by cars, a different network representation (possibly from the same satellite data) is obtained than if we are interested in the traffic of pedestrians. Therefore, depending on which type of process is in focus, the suitable network representation is to be chosen. On the same time, when considering the network flow, it is clear that the observable trajectories are constrained by the structure of underlying *system*, in the example, by the physical street infrastructure.

We can hence conclude that there is a tight relationship between network representation, network process, and network measure (see Fig. 3): the network representation affects the results of the network measures (Butts 2009), and the process of interest dictates which network measures are appropriate to be applied (Borgatti 2005). Furthermore, the process of interest dictates the network representation since the network needs to be chosen such that network and process match. The interdependence of network representation, network measure, and network process is sometimes phrased as Trilemma of Network Analysis (Dorn et al. 2012).

Coming back to the initial question of the "right" simplification for a system, we can see that it is often the process of the network which is essential for the further steps of network analysis, i.e., creation of a network and application of a method. If the research question requires to use a network measure which is implicitly tied to a network process (like centrality indices), the network process of interest constrains the appropriate network representations and applicable network measures. We call this process-centered perspective on network analysis a *process-driven network analysis* (see Fig. 3): the research question dictates the network process of interest which then constraints appropriate network representations and appropriate network measures.

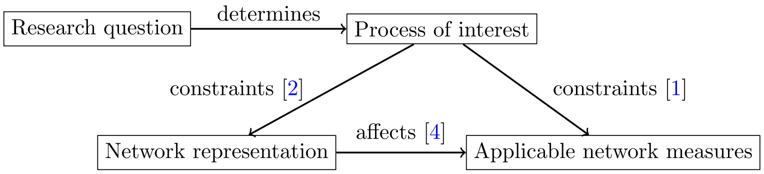
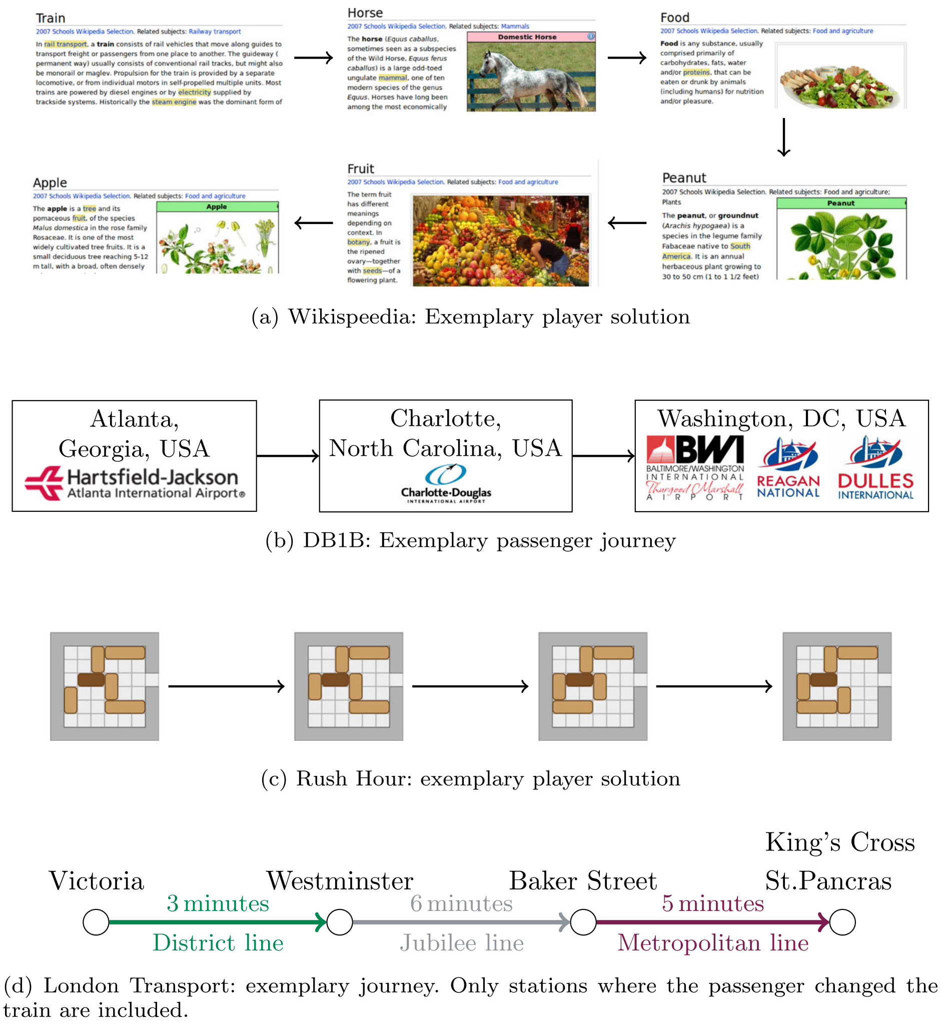


Fig. 3 A process-driven network analysis. The process of interest, determined by the research question, is at the beginning of any network analytic project



(a) Wikispeedia: Exemplary player solution

(b) DB1B: Exemplary passenger journey

(c) Rush Hour: exemplary player solution

(d) London Transport: exemplary journey. Only stations where the passenger changed the train are included.

Fig. 4 Illustration of the contained trajectories contained in the used data sets

We argue for the importance of considering network processes by two aspects:

(i) If a network process is relevant for the research question, it should be considered in an early stage of a network analysis approach. The relevant process helps to restrict the large number of possible network representation and applicable measure to smaller set of possibilities, since the process, the network representation and the measures are not independent from each other in a meaningful network analysis.

(ii) In the example given in the introduction, it is the process of interest, flow of passengers, which is used for constructing the network representation. Many different plausible static network representations can be deduced from the same flow data. It is therefore important to not only consider the static aggregated network representation derived from process flow in the system, but also the process itself. The static aggregated network representation hides the actual dynamics in the system contained in the flow data. It is hence even more important to take into account the properties of the process in order to understand the system.

Contributions of this paper We argue that in those cases where the research question is tied to a process, it is beneficial to put network process into the focus. Since this kind of network analysis starts with the network process as a model for the observed network flow on a given relationship, we call this a *process-driven network analysis* for which this paper is a starting point by making the following contributions:

(i) We propose a framework for a *process-driven network analysis*. This approach can help to reduce the set of possible network representations and possible applicable network methods for a given system.

(ii) We collected data sets of real-world processes and evaluated to which extent they agree with the simple model for them. We show that the uniform weighting of communication (where the amount of flow between each pair of nodes is weighted equally), as for example done by closeness or betweenness centrality, is not a realistic setting for real-world processes.

(iii) The simple process model contained in network measures makes assumptions of *how* the process moves through the network. It is known that shortest paths are not a realistic model in many cases. Therefore, there are many adaptions of network measures which use other trajectory modes instead of shortest paths. The adaptions used most often are random walks (as in random walk betweenness centrality (Newman 2005), Google's PageRank (Page et al. 1999), or community detection by WalkTrap (Pons and Latapy 2005)). We evaluate to which extent the real-world processes can be approximated by those two simple models–shortest paths and random walks. In order to show the impact on the results of network measures, we focus on centrality measures and compare the nodes's importance rated by classic centrality measures to the empiric importance by the real-world network flow.

(iv) Borgatti created the link between process properties and network measures (Borgatti 2005) and provided a categorization of exemplary network processes. We extend this typology in "Typology of network processes" section.

The structure of this paper is hence the following: "Definitions" will briefly introduce the necessary definitions and notations before "Related work" will present existing studies relevant to our work. "Typology of network processes" will derive a typology of process properties which extends the existing typology of (Borgatti 2005). While the process properties derived in "Typology of network processes" are qualitative properties of the *process*, the remainder of this paper contains the quantitative analysis of four data sets of network *flows*. The data sets are described in "Data sets". "Why network flows need to be considered" shows that real-worldnetwork flows contained in our data sets do not show the property of an equal amount of flow between each node pair. This demonstrates that considering the processes taking place in the system creates added value to the analysis of the static aggregated network representation. "Trajectory models: shortest paths or random walks" checks to which extent the second assumption about processes is satisfied by real-world network flows, i.e., *how* the process flows between the nodes. Since most popular network measures assume the flow on shortest paths or on random walks, the real-world network flows are compared to shortest paths and random walks with respect to several metrics. Furthermore, the nodes' empiric importance from the real-world network flow is compared to the nodes' importance measured by standard network centrality measures in order to demonstrate that different flow behaviour does affect network measures.

"Discussion and limitations" provides a discussion of the results, including their limitations, before "Conclusion" sections concludes the article[1].

Definitions

Let $G = (V, E, \omega)$ be a directed simple weighted graph with vertex set V and edge set $E \subseteq V \times V$ and a weight function $\omega : E \to \mathbb{R}$. A walk is an alternating (finite) sequence of nodes and edges $P = \left(v_1, e_1, v_2, \ldots, e_{k-1} v_k\right)$ with $v_i \in V$ for all $i \in \{1, \ldots, k\}$ and $e_j = \left(v_j, v_{j+1}\right) \in E$ for all $j \in \{1, \ldots, k-1\}$. If only simple graphs are considered, a walk is uniquely determined by its node sequence and the simpler notation of $P = (v_1, v_2, \ldots, v_k)$ can be used. 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 P are denoted by $s(P) = v_1$ and $t(P) = v_k$. The length of walk P, denoted by $|P|$, is defined as the weights of its (not necessarily distinct) edges $|P| = \sum_i \omega(e_i)$. Let $d(v, w)$ denote the length of the shortest path from node v to node w. A set of walks in graph G is denoted by $\mathcal{P}$.

As described in "Introduction" section and in Fig. 2, we will use the term *trajectory* for an observed behaviour of an entity flowing through the network. In terms of graph theory, this trajectory is a walk (or even a path). In order to distinguish between theoretically possible walks in the graph and observed trajectories of the network flow, we will use the different terms, trajectory and walk.

Related work

We see an association of our work to several areas of research.

Using network flows for analyzing the network Borgatti made an important contribution by considering possible network processes in a more detailed way and linking the properties of the network process to the application of network analytic methods, in his case centrality indices (Borgatti 2005). He distinguishes processes by two dimensions: by the mechanism of node-to-node transmission and by the type of trajectories the process takes on the network. For the latter, a process might flow on shortest paths through the network, it might flow on paths (no node and no edge is used more than once), on trails (nodes might be visited more than once, but edges are only used once), or on walks (no restriction on multiple usage of nodes or edges). For the mechanism of node-to-node transmission, he distinguishes between a transfer, a serial duplication, and a parallel duplication. Processes with a transfer mechanism include all processes in which indivisible items (such as goods, humans, etc.) flow through the network by actually *moving* from node to node. This particularly implies that a process item is at one point of time at exactly one node. This does not hold for processes with a duplication mechanism: here, the process item is transmitted to the next node while still being present at the current node. This property holds for example for diseases or information. Duplication processes can again be differentiated by serial duplication (the process is transfered to one other node a time) versus parallel duplication (the process is transfered to all neighbor nodes at a time). An example for parallel duplication are email messages which might distributed

[1] This work is the extension of a previously published conference paper (Bockholt and Zweig 2019). The present work goes beyond the scope of the previously published work by providing new results and materials: "Related work" is substantially extended, "Typology of network processes" is added, "Why network flows need to be considered" and "Trajectory models: shortest paths or random walks" is added new results, "Discussion and limitations" sections is added.

to several of a person's contacts at once, while a gossip story is transfered by a duplication mechanism, but (presumably) only from one person to one single other person at once, therefore an example for serial duplication. Borgatti also pointed out that all centrality measures implicitly contain a model for a network process (Borgatti 2005). Based on this work, (Ghosh and Lerman 2014) consider the centrality measures PageRank and Alpha Centrality and argue that PageRank assumes a transfer process (which they call conservative process) and Alpha Centrality assumes a parallel duplication process (which they call a non-conservative process). They demonstrate on two real-world data sets containing a non-conservative process that Alpha Centrality assuming a non-conservative process, yields better results than PageRank which assumes a conservative process. Several authors have questioned whether real-world processes actually fulfill the simple assumptions contained in the centrality indices, for example Chierichetti for web users (Chierichetti et al. 2012) or (Stephenson and Zelen 1989) and (Freeman et al. 1991) for information. Since those real-world processes do not follow the simple models, process trajectories can be used to draw insights about the system itself, for example, for inferring semantic similarity between articles on the base of human trajectories (West et al. 2009), for identifying missing links (West et al. 2015), for predicting the future evolution of a social network based on information flow (Weng et al. 2013), or for finding communities in networks by incorporating real process data into a Markov chain simulation (Rosvall et al. 2014). In a previous work (Bockholt and Zweig 2018), we considered the betweenness centrality and introduced process-based betweenness measure variants: these partly use the process model contained in the original betweenness centrality and partly use information about the behaviour of real-world network flows contained in empiric data sets.

In this work, we argue for a process-driven network analysis in which the process of interest needs to be considered additionally to the static network representation of a system. We hence understand the network structure as an infrastructure which is used by the process. There are alternative approaches, for example higher-order networks in which the process data is used for the *construction* of the network representation (Scholtes 2017; Xu et al. 2016): in this representation, a node does not represent a single entity of the system, but tuples of entities. In this representation, dependencies contained in the process data where the choice of the next node is dependent on previously visited nodes, is translated into the network structure (up to a certain extent, depending on the order of the network). Xu et al. (2016) argue that the network representation itself should reflect those dependencies. A further alternative approach is proposed by Lambiotte et al. (2011) who introduce so-called flow graphs, i.e., network representations which incorporate the network dynamics in the edge weights. They provide a mathematical framework how to construct flow graphs from biased and unbiased walks as well as from consensus processes.

Empiric analysis of network flows This work is not the first analyzing the properties of network flows. There have been presented many qualitative analyses of spreading processes, i.e., processes with a copy mechanism: in social networks, Bakshy et al. analyze information diffusion on Facebook by an experimental setting (Bakshy et al. 2012), Cheng et al. consider the dynamics of information cascades on Facebook (Cheng et al. 2014), Domenico et al. present a temporal and spatial analysis of rumors spread on Twitter (De Domenico et al. 2013). Other processes have been also analysed in an empirical way, for example the transfer of behaviours or attitudes in social systems, such as obesity or smoking habits (Christakis and Fowler 2007; 2008), but also the spreading dynamics of mobile phone viruses (Wang et al. 2009). A whole research community is dedicated to the dynamics of infection spreading, epidemics in scale-free networks (Pastor-Satorras and Vespignani 2001), epidemic spreading in air-transportation systems (Colizza et al. 2006), but also the spreadings of animal epidemics, for example the spread of BSE by cattle movements (Gilbert et al. 2005).

Empiric analyses of processes with a transfer mechanism are for example the analysis of humans navigating in social networks (Adamic and Adar 2005), humans solving a word puzzle (Sudarshan Iyengar et al. 2012), humans navigating in information networks (Helic et al. 2013; West and Leskovec 2012a), humans browsing the Internet (Sen and Hansen 2003), or routing of taxis in London (Manley et al. 2015).

Agent-based simulations of transfer processes Similarly to our approach of simulating a given transfer process by a walk-based model, there are other approaches proposing agent-based simulations of transfer processes. Holme suggests several variants of random walk simulations in complex networks and studies the traffic speed and density with respect to the network structure (Holme 2003). Manley presents a Markov chain Monte Carlo simulation for estimating traffic in London (Manley 2015). In the context of human navigation in information networks, West and Leskovec implement several variants of an agent-based simulation with different strategies and compare their performance with human performance for the same task (West and Leskovec 2012b). In a similar context, (Helic et al. 2013) study how human navigation in information networks can be modeled by variants of decentralized search, i.e, navigation with only local knowledge of the network. An interesting contribution in this context is the work of (Kleinberg 2000) who proved that a good performance of any decentralized search algorithm is only possible in networks with certain properties.

Sensitivity of data preprocessing A further link of our work is to prior works concerned with the impact of decisions in the data preprocessing phase. Butts analyzed the effects of choices in the network construction phase and showed that different levels of aggregation for the nodes have an effect on the basic properties of the resulting networks (Butts 2009). A similar effect can be observed for the exact definition of edges. Many interactions are naturally dyadic and can be transformed into edges in a natural way. In other cases, interactions have an intensity which leads to the question of whether all observed interactions should be transformed into an edge or whether only those with a sufficiently high intensity. By increasing the threshold of connection strength necessary for including the connection as an edge, the resulting network changes and large structural differences of the resulting networks are observed (Butts 2009). De Choudhury et al. make a similar observation for a communication network deduced from interpersonal email communication (De Choudhury et al. 2010). The choice of an appropriate threshold is not trivial: the network should only contain the “relevant” edges, but excluding edges with a low intensity will alter the network properties dramatically. According to the famous work of Granovetter, the weak ties,i.e., the edges with low intensity, are the ones serving as bridges in the network and have an important role in the network structure (Granovetter 1973).

A further aspect that needs to be considered in the transformation of a system (more specifically, observations of system interactions) into a network representation, is the temporality of the interactions. Butts gives the example of the spread of HIV virus by sexual contacts (Butts 2009). If, in a sexual contact network, an edge is created if there has been a sexual encounter, the analysis of the aggregated network might yield misleading results: the timing and the order of the sexual encounters have an impact on the spread of the HIV virus (Moody 2002). This cannot be regarded when the edges are aggregated over time.

Furthermore, (Tavassoli and Zweig 2016) demonstrate the large impact of seemingly minor modeling decisions by considering multi-layer networks and the most simple centrality measure, degree centrality. In order to apply degree centrality on a multi-layer network, it is necessary to aggregate a node's degree over the different layers. They consider different aggregation and normalization strategies which seem to be equivalent at first sight. They however find that the resulting node rankings differ considerably dependent on the aggregation and normalization strategy.

Typology of network processes

We understand a network process as a model of how something is flowing through the network whereas the network flow consists of observable trajectories of something using the network. Borgatti suggested to classify network processes along two dimensions: node-to-node transmission mechanism (transfer, serial duplication, or parallel duplication) and type of trajectory (shortest path, path, trail, or walk) (Borgatti 2005). When collecting exemplary network processes, it seems that these two dimensions are not the only qualities by which processes can differ which is why we present an extended typology of network processes in the next section (see Table 1 for an overview).

Target to reach Network processes that are goal-oriented act differently than those without a specific goal because they stop when the target is reached. Based on other constraints, such as knowledge of the network's structure, and intelligence of the flowing good to optimize the way through the network, the chosen paths might come close to shortest paths. Network processes without a goal or with short-term goals like money transfer will look more like random walks on the infrastructure. For processes with a target, we can further distinguish between the case that all process entities have exactly the same target (a player trying to solve a game), and the case that each entity has its own target.

Routing mechanism How is the route of a process item determined? For some processes, the route which is taken by the item, is already predefined before the item starts moving: for a parcel being delivered, the optimal route for it is computed before the delivery process starts. Hence, while moving through the network, there are actually no choices for routing. This also implies that the route is determined with global knowledge of the network structure (*global routing*). This is different for other processes. For a person browsing through the Internet searching for a specific information, the global knowledge of the network structure is not available, but the routing decision of which node to go next to is taken by the browsing person on the basis of the local view of the network (*local routing by process*). Yet another situation is the propagation of a piece of information through a social network. The decision of which node the piece of information is forwarded to is also on the basis of local knowledge of the network, but not taken by the process item, but the network node (*local routing by node*).

In the context of information diffusion, Milli et al. differentiate between active and passive diffusion (Milli et al. 2018). In this wording, the nodes of network can be active in the sense that they can "decide" whether they want to adopt a behavior or another process. In other types of diffusion processes such as the spreading of a disease, the nodes are passive in the sense that the diffusion process will spread to them without them having any possibility to avoid it.

Forwarding or Adopting In the case of processes routed locally by the nodes, we can distinguish between processes where, in each step, the sending nodes or the receiving nodes are of relevance: Consider the propagation of a gossip story where the current node decides to whom the story is told to. On the other hand, the propagation of a behavior or a trend is different: although the sending node decides about showing the behavior or promoting a trend, it is actually the receiving node who decides to adopt it or not. For the latter, it can be distinguished between *simple* and *complex contagion* (Centola 2010): for simple contagion, the exposure to the process by a single node is sufficient to adopt it, while for complex contagion processes, a node needs to be exposed to the specific behavior from several neighbor nodes before it adopts it (in social networks known as social reinforcement).

Interaction of network and process For the analysis of the corresponding network, it is important to know whether the process and the system interact in any way such that the one can change the other: in most cases, the network is providing

Table 1 Catalog of network process properties that determine the way an entity uses the network structure. The first two properties, transmission mechanism and trajectory type, were proposed by Borgatti (2005)

Dimension	Possible values
Transmission mechanism (Borgatti 2005)	transfer serial duplication parallel duplication
Trajectory type (Borgatti 2005)	shortest paths paths trails walks
Target to reach	same target for all entities individual targets no target
Routing mechanism	local routing by node local routing by process global routing
Adopting	forwarding adoption: simple contagion adoption: complex contagion
Interaction network and process	no interaction process shapes network
Place of process	on nodes on edges

the infrastructure for the process to move through the network, hence the process is restricted by the network structure. In other cases, there is a feedback loop such that, in the long run, the process can change the network structure. This is for example true for air transportation networks: when a connection between two airports is highly demanded, but the airline only offers an indirect connection via an intermediate stop, the airline might install a new direct connection. Hence, a new edge will be added to the network representation. Conversely, when a connection between two airports is not demanded by a sufficient number of passengers, the connection won't be offered anymore by the airline, hence, the edge is to be removed from the network. This effect has been analyzed in the context of information flow in social networks (Weng et al. 2013). In social networks, it is also quite unlikely that persons want to communicate indirectly over a long time. If both are interested in a communication, they will try to get into a direct contact. Evidence of this is by Friedkin's study on the "horizon of observability" of colleagues in a university communication network (Friedkin 1983; Zweig 2018).

On nodes or on edges Although all processes of course need the nodes *and* the edges of the network, it strikes that some processes are present on the edges of the network while others are present on the network nodes. Consider a piece of information which is present at the nodes, i.e., in the mind of the persons, and is only "present" at the network edge while being transferred to another person. A process as cars in the road network is different. This process mainly takes place on the network edges. This differentiation can be useful when centralities are applied on the network. In one case, it might be better to apply node centralities, in the other case, edge centralities might be preferred.

With this, we can improve our first definition of what a network process is: A network process is a model of an observable flow that details at least a subset of properties as described in Table 1.

Data sets

This section provides information about the used data sets, their source and their preprocessing. In order to compare the real process with shortest and random paths, we restrict the analysis to processes with a transfer mechanism, otherwise the concept of shortest paths or random agents is not meaningful. Furthermore, we require the following aspects being satisfied:

(i) The process usually traverses more than one edge in a row, i.e., the process is more than a set of dyadic relations. This excludes processes as emotional support or seeking advice because the aspect of transitivity is not given: if A supports B and B supports C, there is still no flow of support from A to C.

(ii) The process consists of one or several indivisible entities (e.g., persons) who are moving through a system. This implies in particular that each process entity is at one point of time at *one* node (processes such as infections are excluded by this requirement). Otherwise, the concept of shortest paths is not meaningful.

(iii) The route taken by an entity is only dependent on the network structure and the process itself, not driven by external events. This excludes for example the spreading of rumors in a social network: a rumor is not distributed to all social acquaintances of a person, but rather to a subset of acquaintances, eventually those whom the person met in person.

(iv) The network representation and process need to fit: the edges need to represent this type of relationship which is essential for the process. Analyzing the spread of a virus in a social network is meaningless since it is not relationship of knowing each other which is essential for the contagion of a virus, but the physical contact within a certain time interval.

There are many processes that fulfill the given requirements, they are, however, often not available as data sets. For the analysis of interest, we also require the following technical aspects to be satisfied by the data sets such that they can be used:

(i) The data must contain a network structure (V, E)

(ii) The data must contain a network flow which uses the network structure as infrastructure by moving from node to node. This excludes (GPS) movement trajectories if there is no network structure which restricts the movement. The set of trajectories is denoted by $\mathcal{P}$.

(iii) There is a reasonable mapping of the locations of the process onto the nodes of the network.

(iv) The data contains the information which item at which point of time used which edge of the network.

The following data sets satisfy those requirements and were used (Table 2 and Fig. 4 show an overview of the used data sets).

DB1B The Airline Origin and Destination Survey (DB1B) is collected and provided by the US Office of Airline Information of the Bureau of Transportation statistics (RITA TransStat 2016). It contains a 10 % sample of all airline tickets from reporting airline carriers within the US. For each ticket, the data base includes the information about origin and destination of the itinerary as well as intermediate stops. The process of interest hence consists of passengers traveling by airplane connections through a network of airports. We used the data of the years 2010 and 2011. Itineraries were split into outbound and return trip according to the corresponding data entry in the database. A node in the network represents a city. Airports of the same city area (as indicated by an entry in the data base) are contained in one node. An edge (v, w) is created if and only if the data contains at least one itinerary with the flight connection from an airport in v to an airport in w.

Table 2 Overview of the used data sets

Data set	Graph	Nodes	Edges	Process
DB1B (RITA TransStat 2016)	directed	airports	non-stop airline connections	passengers
Wikispeedia (West et al. 2009; West and Leskovec 2012a)	directed	Wikipedia articles	hyperlinks	Wikispeedia players
Rush Hour (Jarušek and Pelánek 2012)	undirected	configurations	valid game moves	players
London Transport (Transport for London 2017)	directed	Underground stations	underground connections	passengers

Wikispeedia This data set is provided by West et al. (West and Leskovec 2012a) and contains game logs of persons playing the Wikispeedia game. In this game, a player is given a pair of Wikipedia articles and the goal of the game is to reach from the one given article to the other by following the hyperlinks within Wikipedia. West collected this data by providing this game on his web page. Players could either choose a pair of articles themselves or were given a pair of articles. A subset of all Wikipedia articles was used. Hence, there is a directed network where a node represents an article, and there is a directed edge from v to w (with weight 1) if article v contains a hyperlink to article w. The network flow is persons playing Wikispeedia. We consider only solutions reaching its target. Moves which were revoked by the player via an Undo button, were not included in the paths.

London transport Transport of London, a governmental authority responsible for most of public transport within the greater region of London, provides a data set called Rolling Origin and Destination Survey, collected by Oyster cards (Transport for London 2017). An Oyster card is an electronic ticket, valid for most means of public transportation within Greater London. The data set contains a 5 % sample of all Oyster card journeys performed in a week during November 2017. For each pair of start and destination station, the data contains the frequency of usage of this pair as well as the stations where the passengers changed the means of transportation. It however does not contain the intermediate stops which is why it is not known which means of transportation was taken. We used the timetables of London underground and constructed a (multilayer) network where each layer corresponds to one underground line i and contains a set of stations as nodes and an edge between those two stations which can be reached via line i without changing trains. Note that this does not yield a chain-like graph as a Underground line plan would suggest, but the *transitive closure* of the chain. The edges are weighted by travel time, more precisely, by the shortest-possible travel time with this particular line. For a passenger's journey $A \to B \to C \to D$, it is then checked in which network layer, i.e., which underground line, the connection from A to B (and between the other intermediate stops, B to C, and C to D) is the shortest (in travel time). The complete trajectory is then the concatenation of those connections. Note that in this modelling, a trajectory only contains nodes in which the passenger changed the means of transportation. Another variant in which a trajectory also contains the station which the passenger passes through while sitting in the tube without getting off, is not used in this work. Note also that we did not consider the timing of the connections since this information is not available for the journeys: for the journey $A \to B \to C \to D$, it might be that we find that the line Bakerloo is the fastest connection from A to B, and underground line Central is the fastest connection from B to C which is then inserted in the constructed trajectory. In reality, the passenger might have taken a different line because a different line is leaving earlier. Additionally to the multilayer network, we create a one-layer network by merging all layers into one.

Rush hour This data set contains a state space of a one-player board game called Rush Hour and as process persons attempting to solve the game,i.e., persons navigating through the state space. The game consists of a board with 6×6 cells with a designated exit. Cars (blocks of width of 1 cell and of length of 2 or 3 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 for a given board configuration contains a node for each valid (and reachable) board configuration, an edge represents a valid game move. The network is undirected because all moves can be reversed and all edge weights are equal to 1. The game data was collected by Jarušek and Pelánek (2012) by their web-based tool for education. Three different games were used for our analysis. We include only solving paths, i.e., paths which end in a solution of the game.

Table 3 contains basic properties of the used data sets. For the data sets from transportation scenarios (DB1B and London Transport), the largest number of trajectories is available. Note that, for all data sets, the trajectories are not necessarily distinct, there are trajectories that are taken by many passengers or players and are therefore contained multiple times in $\mathcal{P}$. We see that the three different game instances of Rush Hour induce state spaces of different order. Rush Hour Game A is a very easy game with an optimal solution of length 3 while Game B and C are of medium difficulty with an optimal solution of length 11 and 13, respectively.

Why network flows need to be considered

Standard network analytic methods which are applied to the representation of the network do not consider any real flow taking place on the network. Instead, many network analytic methods, particularly all distance-based network measures and centrality measures, contain a simple model for the process. This model for the real network flow contains the main assumption: How does the flow travel through the network? Here, most measures assume the process flowing on shortest paths or on some kind of random walks. Furthermore, a network measure counts some property of the assumed process, for example, the number of process entities traversing through a node v. In most network measures containing such a simple process model, the amount of flow between each node pair is weighted equally, which we call a *uniform weighting of communication*. For illustration, consider the classic centrality measures closeness and betweenness centrality (Freeman 1977). The closeness centrality of a node v is defined as the sum of the lengths of the shortest paths from all other nodes to v:

$$C(v) = \sum_{w \in V} d(w, v) \tag{1}$$

The idea behind this measure is that a node v is considered as central if it can be reached from all other nodes quite fast, i.e. the average path length of any node to v is small. This assumes that for a node v, it is equally important that v can be reached quickly from a node w or from another node x. Thus, the assumed process model is based on shortest paths,

Table 3 Basic properties of the used data sets. $|V|$ and $|E|$ denote the cardinality of the node and edge set of the underlying graph, $|\mathcal{P}|$ denotes the number of observed flow trajectories in the data set. All networks consist of a single connected component

Data set	$\|V\|$	$\|E\|$	$\|\mathcal{P}\|$	Trajectory length	
				Range	Average
DB1B	462	12499	86m	[1, 12] hops	1.4 hops
Wikispeedia	4592	119804	51306	[1, 404] hops	5 hops
Rush Hour Game A	364	1524	3044	[3, 33] hops	5 hops
Rush Hour Game B	6769	33142	1965	[11, 59] hops	15 hops
Rush Hour Game C	830	4037	1472	[13, 95] hops	26 hops
London Transport	268	13173	4.8m	[1, 107] min	16.3 min

and the length of the shortest path from each node w to node v is weighted equally for the computation of the measure. A similar assumption is included in betweenness centrality: for calculating the betweenness centrality for a node v, for each node pair $s, t \in V$, the ratio of all shortest paths between s and t that contain node v (and are not equal s and t) is computed and summed up. The betweenness centrality of node v is then defined as

$$B(v) = \sum_{s,t \in V, s \neq v \neq t} \frac{\sigma_{st}(v)}{\sigma_{st}} \tag{2}$$

where σ_{st} denotes the number of shortest paths from s to t and $\sigma_{st}(v)$ denotes the number of shortest paths from s to t which contain v. Thus, also here, the assumed process model is based on shortest paths, and for computing the measure value, the amount of flow between each node pair contributes equally to the measure value: it is summed over all node pairs and each node pair can contribute at most a value of 1 to the betweenness value of any given node v. For example, applying betweenness centrality to the European rail network, being contained in all shortest paths from Paris to Berlin contributes the same value to the centrality as being contained in the single shortest path between two small villages. Hence, both measures assume the process traveling on shortest paths and the amount of flow between each node pair is weighted equally for the measure computation. We will refer to the latter as *uniform weighting of communication*. The random walk betweenness centrality (Newman 2005) is an example for a measure with uniform weighting communication and assuming random walks instead of shortest paths. In the following, we want to show to which extent those two properties are fulfilled by real-world transfer processes. We organize the following results along the two aspects, uniform weighting of communication, and flow on shortest paths or random walks:

(i) Do real processes need to be considered at all? Why is the network structure not sufficient? We will show that for real-world processes, it is not true that the amount of flow is equal between each pair of nodes, as suggested by the uniform weighting of communication. On the contrary, there are a few node pairs which are used heavily by the process while most of them are only used rarely or not at all.

(ii) Can real processes be modeled by shortest paths or random walks? Is this a sufficient approximation for a real process? We will show for which properties the real-world process shows similar features as the corresponding trajectory model with shortest paths or random walks (see "Trajectory models: shortest paths or random walks" sections).

We start with the first property, uniform weighting of communication and investigate how the network is used by the network flows contained in the data sets.

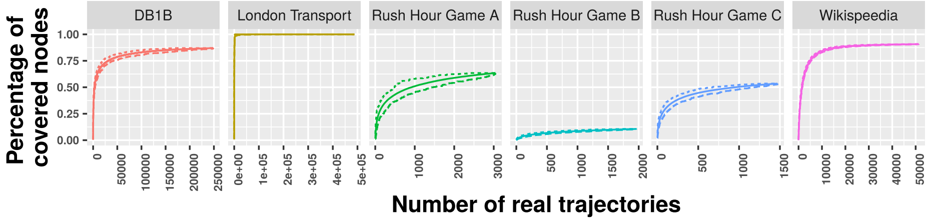
Network coverage Figure 5a shows the **coverage** $C(\mathcal{P})$ of the network by the network flow contained in $\mathcal{P}$,

$$C(\mathcal{P}) = \frac{1}{|V|} | \{v \in V | \exists P \in \mathcal{P} : v \in P\} |, \tag{3}$$

i.e., the fraction of nodes which are visited by the network process at least once. Due to the network construction, it is clear that all nodes of the DB1B network are contained in at least one passenger's journey. For the other data sets, 10 to 89 % of the nodes are not

Coverage					
DB1B	LT	RH Game A	RH Game B	RH Game C	Wiki
100 %	100 %	63 %	11 %	53 %	90 %

(a) Percentage of nodes covered by the network flow.



(b) Relative node visitation of the observed network flow. Note that for DB1B and London Transport, not all observed trajectories, but a randomly sampled subset was used.

Fig. 5 Graph coverage of the observed network flow

used at all by the flow. Hence, for those real-world network flows, it is not true that all nodes are equally important for it.

It is possible that including more network flow trajectories would cover more nodes of the network. If more trajectories of the processes were available, would the fraction of used nodes increase? Or do the observed trajectories use more or less a similar set of nodes such that adding further trajectories will not increase the number of covered nodes? In order to answer those questions, we use the set of observed trajectories $\mathcal{P}$, successively pick (and remove) trajectories from it uniformly at random, add them to an initially empty set $\mathcal{P}'$ and compute the coverage of the graph by $\mathcal{P}'$ in every iteration. This yields a sequence of coverage values $\left(C\left(\mathcal{P}'_1\right), C\left(\mathcal{P}'_2\right), \ldots, C\left(\mathcal{P}\right)\right)$ where $\mathcal{P}'_i \subseteq \mathcal{P}$ with $\left|\mathcal{P}'_i\right| = i$. In order to reduce the effect of the order in which the trajectories from $\mathcal{P}$ are drawn, the above procedure is repeated $N = 500$ times, and the minimum, maximum and average of each $C\left(\mathcal{P}'_i\right)$ over all N iterations is computed (an extended variant of this procedure is sketched in Algorithm 1).

For the data sets DB1B and London Transportation containing more than 63 million observed trajectories and almost 5 million observed trajectories, not all observed trajectories are used for the procedure, but a randomly sampled subset (0.1 % and 10 % of the real paths, respectively). For the other data sets, all available observed trajectories are used as $\mathcal{P}$.

Figure 5 shows the network coverage with increasing number of observed trajectories. It can be observed that for the data sets DB1B and Wikispeedia, the coverage reaches a saturation where adding further observed trajectories to the set does not change the set of covered nodes anymore. This is also true for the Rush Hour games A and C, but does not seem to be the case for game B. Here, at least with the set of available set of real trajectories, a saturation cannot be observed.

For the considered data sets, it seems to be due to the nature of the process that only a part of the network is covered by the observed flow trajectories, and not due to the number of available trajectories. We conclude that the process is in essence restricted to only a part of the node of the nodes for five out of the six data sets. Note that the coverage for the data set DB1B is approximately 90 % for the sampled subset of trajectories while the coverage is 100 % when all real trajectories are included. This indicates that there might be an imbalance in the frequency of node visitations, i.e., how often a node is used by the process entities. This imbalance cannot be taken into account by a simple binary attribute whether a node is used by the process or not. Furthermore, the introduced network coverage is not an informative measure if the network is constructed from trajectories–then, the network only contains nodes which were contained in at least one trajectory–or if the network is truncated to used nodes which is both sometimes done in practice. For those two reasons, the frequency of node visitations is considered in the following section.

Uniform weighting of communication We first investigate whether the network flow is present at all covered nodes with the same intensity. For this purpose, we compute for each node v its **node usage** $nu(v)$,

$$nu(v) = \frac{|\{P \in \mathcal{P} | v \in P\}|}{|\mathcal{P}|}, \tag{4}$$

i.e., the percentage of observed trajectories the node is contained in

Figure 6a shows the cumulative node usage distribution for each of the data sets. It can be seen that for all data sets, the majority of nodes are, if visited at all, only contained in a small fraction of all trajectories. On the other hand, there are very few nodes which are contained in many or even all observed trajectories. Thus, for any of the data sets (except for DB1B), the real-world network flow is not observed at all nodes, and, for any of the data sets, the real-world network flow is not observed at all nodes with the same frequency.

Most network measures weight the amount of flow between every node pair equally. In order to test whether this is justified for real-world processes, we count for every pair of nodes $(s, t) \in V \times V$ how many observed trajectories start in s and end in t, and divide by the total number of observed trajectories, what we call **node pair usage** $npu(s, t)$:

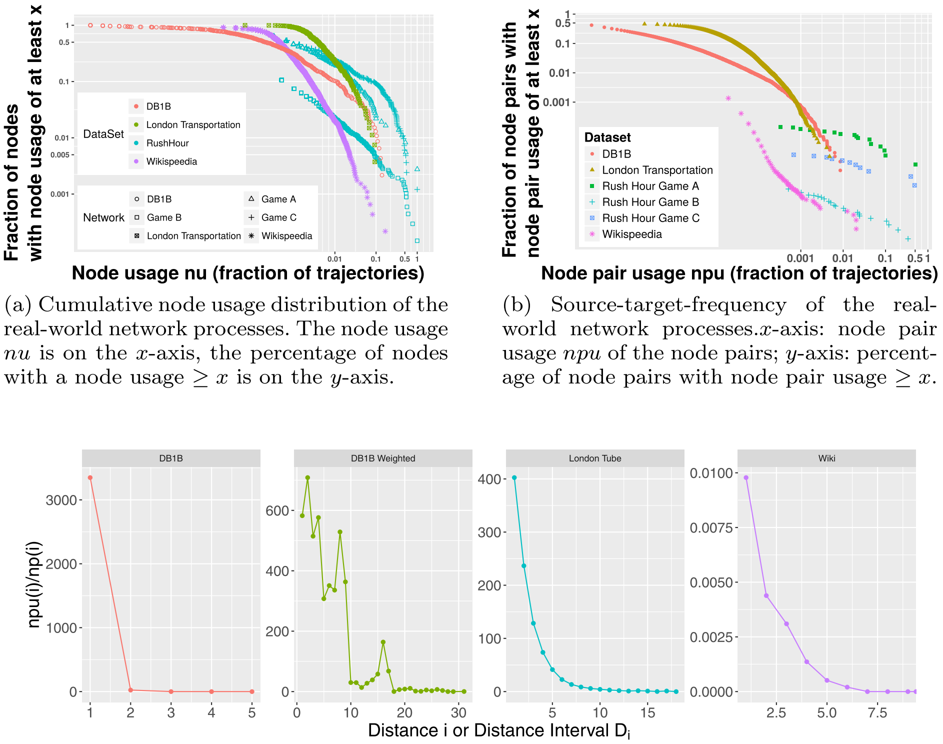
$$npu(s, t) = \frac{|\{P \in \mathcal{P} | s(P) = s, t(P) = t\}|}{|\mathcal{P}|} \tag{5}$$

The cumulative distribution is shown in Fig. 6b. We make a similar observation as before: only a fraction of all node pairs is used as source and target of the real-world process, between 42 % and 46 % for the transportation data sets, 0.14 % for Wikispeedia, and less than 0.01 % for the Rush Hour games. For Rush Hour, all players start in the same start node and aim at reaching one of a few final nodes, which is why such low numbers were expected. When restricting the computation to those node pairs which are start and solution node, the percentage of used node pairs increases to 36 %, 3 %, and 11 %, respectively, which is still low. In all considered network, even in those systems whose purpose it is to bring passengers from one place to another, there is no flow between more than half of all node pairs.

We furthermore see that for all data sets, the majority of the pairs is the source and target for very few flow trajectories while there exist a few node pairs which are the source and target for many trajectories.

In order to understand which node pairs are used most often as source and target, we count for every possible graph distance k how many trajectories go from a node s to a node t with $d(s, t) = k$ which we define as **node pair usage for a given distance** $npu(k)$:

$$npu(k) = |\{P \in \mathcal{P} | d(s(P), t(P)) = k\}| \tag{6}$$



(a) Cumulative node usage distribution of the real-world network processes. The node usage nu is on the x-axis, the percentage of nodes with a node usage $\geq x$ is on the y-axis.

(b) Source-target-frequency of the real-world network processes.x-axis: node pair usage npu of the node pairs; y-axis: percentage of node pairs with node pair usage $\geq x$.

(c) Node pair usage by the distance of the nodes. For each graph distance d, the total number of trajectories between nodes being d steps away from each other, is normalized by the total number of node pairs in the graph with distance d. For data sets with weighted edges, the distance of node pairs is divided into intervals of equal size: for London Tube intervals of 5 minutes travel time, for DB1B weighted where edges are weighted by geographic distance between the airports, intervals of 500 km distance between the airports.

Fig. 6 Node usage distribution and source-target-frequency of the real-world network processes

Note that we do not consider the length of the trajectory, but the length of the *shortest path* between source and target. Furthermore, let

$$np(k) = |\{(v, w) \in V \times V | d(v, w) = k\}| \tag{7}$$

be the **total number of node pairs** in the graph with a graph distance of k. For the data set London Transportation where the graph edges are weighted with the travel time ranging from 1 minute to 98 minutes, we introduce distance intervals of size 5, such that an edge with weight w with $0 < w \leq 5$ minutes is assigned a distance interval with the number 1, etc. For the data set DB1BW, a weighted variant of DB1B where the edges are weighted by the distance (in km) between the airports, we introduce weight intervals of size 500 km.

Figure 6c shows for each data set and each distance (interval) d the number of process entities traveling between a node pair of distance d, normalized by the total number of node pairs with distance (interval) d. For the three Rush Hour games, this analysis is not meaningful because all players start in the same node and end in one of a few final nodes (having almost the same distance). It can be seen that for the remaining data sets, node pairs with smaller distances are demanded more than node pairs with larger distances. This result is not surprising for the processes at hand: If the connection between two airports is highly demanded, a direct non-stop flight will be installed; the most crowded places in London are in the center of London and not far apart from each other, those places will be demanded more often as source and destination than other places in the peripheral areas of London. This observation has also been made in communication networks by Friedkin (1983). He found that there is a distance in communication networks, which he called a *horizon of observability*, beyond which persons are very unlikely to be aware of the existence of each other. Hence, it is very unlikely to observe any kind of flow between persons which are further apart from each other than that horizon of observability.

Summary

The previous findings that (i) not all nodes are relevant for the network flow, (ii) a few nodes are used heavily by the network flow while most are used only a few times, (iii) close nodes are overproportionally often source and target of the network flow, might have been expected for the used data sets. They do, however, have consequences for network analysis in general: the simplistic aggregation of the assumed network flow by a uniform weighting of communication, as done by many network analytic methods, particularly in centrality indices, is far from the behaviour of real-world network flows. The usage pattern of network flows is indeed a real addition of information to the static network representation.

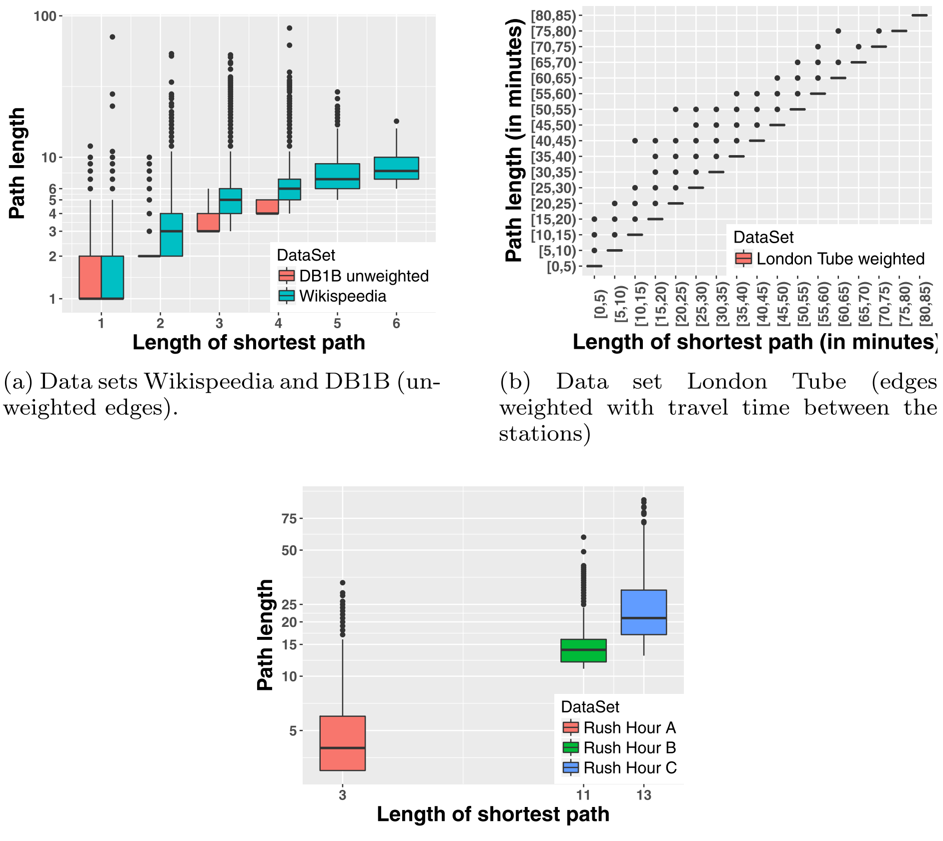
The question arises whether the data of real network flows need to be considered or whether simple simulations of the trajectories are able to capture the essential properties of real network flows. There are basically two extreme cases of trajectory models: shortest paths and random walks (of different types). The following sections will investigate whether shortest paths or random walks are at least approximate models for the dynamics of real network processes.

Trajectory models: shortest paths or random walks

In the following, we investigate whether the trajectories of the chosen real-world network flows can be approximated to a satisfactory extent by the two most simple trajectory models, shortest paths and random walks. In a sense, shortest paths and random walks can be seen as the extreme cases of the same scale as Newman puts it for the case of shortest path betweenness and random walk betweenness: they "are at opposite ends of a spectrum of possibilities, one end representing information that has no idea of where it is going and the other information that knows precisely where it is going. Some real-world situations may mimic these extremes while others, such as perhaps the small-world experiment, fall somewhere in between" (Newman 2005). In the following, we investigate by which properties the observed flow trajectories are rather like shortest paths and by which they are rather like random walks. We begin by considering the length of the real flow trajectories and how much their length deviates from the length of the shortest path.

Modeling trajectories by shortest paths

Path length For the Rush Hour games, all players start at the same node and aim at reaching the same solution node which is why the length of the shortest path is the same for all players. This is different for the other data sets where the length of the shortest path is different for each passenger or player, dependent on its target node. Figure 7 shows the lengths of the trajectories contained in the data sets. For each $P \in \mathcal{P}$, we compare



(a) Data sets Wikispeedia and DB1B (unweighted edges).

(b) Data set London Tube (edges weighted with travel time between the stations)

(c) Data sets Rush Hour

Fig. 7 Lengths of trajectories for the used data sets. For Wikispeedia, three outliers of length 404, 108, and 101 were removed for a better readability

its actual length $|P|$ with $d(s(P), t(P))$, the length of the shortest path from the start node of P to its target node. It can be seen that the trajectories of DB1B, representing passenger journeys, are well approximated by shortest paths. This is different for the trajectories of the Wikispeedia game: the observed trajectories are longer than the shortest possible path, however, on average only by 1 step (see Fig. 7a). The same comparison is done for the London Transport data set, where the length of the trajectory is the traveltime (see Fig. 7b). We observe that the approximation of shortest paths for the observed trajectories is satisfactory, but there is a considerable amount of trajectories which are up to 30 minutes longer than the shortest path.

For the three game instances of Rush hour, we observe that the lengths of the observed trajectories strongly depend on the instance (cf. Fig. 7c). For the easy game A with an optimal solution of 3 moves, 37 % of the observed trajectories are of length 3, being on average of length 5 (median 4). The observed trajectories for the two games of medium difficulty, games B and C with an optimal solution of length 11 and 13, respectively, are poorly approximated by shortest paths. Only 14 % (game B) and 2 % (game C) of the observed trajectories have the same length as the shortest path. Although their optimal solutions have similar lengths (11 and 13), the observed trajectories of game B and C have very different lengths: for game B, they are on average of length 15 (median 14) while the trajectories of game C have an average length of 26 (median 21).

We can conclude that for transportation processes, the observed trajectories are close to shortest paths. However, even for those processes, there is still a considerable amount of observed trajectories which are longer than the shortest path. For the two game processes, the observed trajectories are only poorly approximated by shortest paths. The observed trajectories are on average 1 (for Wikispeedia) to 13 (Rush Hour Game C) longer than the shortest path.

Modeling trajectories by random walks

Besides shortest paths, the other obvious simple trajectory model for *how* something flows through the network, are simulations by random walks. We use a simulation approach and implement agents performing random walks on the network and when comparing their behaviour with the real-world network flow. Most random-walk based measures do not rely on simulations, but can be computed analytically by considering the stationary distribution of the corresponding Markov chain. For example, for an infinitely long random walk with uniform probability of choosing a neighbor and with a probability of $1/|V|$ for each node v that the random walk starts in v, the probability distribution of where the random walker is, converges to a stationary distribution. The stationary distribution can be expressed by a function dependent on the degree of a node, therefore, simulating this kind of random walk will simply yield node usage values proportional to the nodes' degrees (Masuda et al. 2017). However, the previous sections showed that the real-world network flow does not show a uniform weighting of communication, this observation needs to be taken into account when modeling trajectories as random walks. Therefore, it is not clear whether the stationary distribution is the same when the source-target frequencies of the real-world network flow is taken into account. It is a possible explanation that a high observed node usage is simply due to the high number of trajectories starting or ending in this node. This is why we use a simulation approach and tie each random walk to a real trajectory contained in $\mathcal{P}$ by respecting its start node and its length. Tieing a random walk to a real trajectory allows to compare the node usage and graph coverage of real and random walks because the start nodes and the "outreach potential" is equal for real and random walks. The generated random trajectories can be seen as null-model for the real trajectories: the source node distribution and their outreach potential is fixed to the real trajectories, but the kind of *how* the entities move through the network is replaced by random choices. This allows to separate two different aspects: (i) from where does flow occur, and (ii) how does it move through the network? The generated trajectories can be seen as null-model for the empiric trajectories: their source-distribution is the same, but while the empiric trajectories are assumed to be formed according to some knowledge, the random trajectories are created by a random exploration of the network.

Given a real trajectory, represented by a walk Q in graph G, the basic procedure for a random walk is the following: An agent starts at $s(Q)$, chooses a neighbor of the current node, moves to the chosen node, and repeats this procedure until a stopping criterion is reached. The following variants are implemented

Neighbor choice Two different variants of neighbor choices were implemented:

Uniform neighbor choice (UNC) The simplest method of choosing the next node where the agent moves to is choosing uniformly at random from the set of all neighbors of the current node (*uniform neighbor choice, UNC*).

Backwards-restricted neighbor choice (BWR) The agent chooses uniformly at random from the set of neighbor nodes where the directly preceding node is excluded (*backwards-restricted neighbor choice, BWR*).

Stopping criteria Two different stopping criteria were implemented while all are intended to model that the random walks and observed trajectories have the same "outreach potential".

Path length restriction (PL) The agent moves to a neighbor node $|Q|$ times. This yields a walk Q_{RAND} with $|Q| = |Q_{RAND}|$, hence, the random agent is allowed to make as many steps as the real one Q. This criterion is referred to as *path length restriction (PL)*. If the graph is weighted, the random agent is bounded by the weighted length of Q: the random agent moves from node to node as long the agent has walked over edges with a total sum of edge weights less than $|Q|$. This yields a random walk Q_{RAND} with $|Q_{RAND}| \leq |Q|$. (As long as the random agent has walked over edges with a total weight of less than $|Q|$, it continues. It might then choose an edge with an edge weight such that the sum of edge weights is exceeded. This edge is then not added and the procedure stops.)

Line change restriction (LC) For the London transport data set where the system is modeled as a multilayer network, we introduce the *line change restriction (LC)* as a stopping criterion: the random agent is allowed to perform as many line changes as the real trajectory $|Q|$ contains layer changes. This stopping criterion is motivated by the flows of passengers in Greater London using underground connections. A passenger enters a means of transportation at some point and leaves that vehicle after a certain time. The stations at which the vehicle stops in between are of no real relevance for the passenger: the possible travel alternatives at the intermediate stops are only of relevance if the passenger drops of the current vehicle. A random agent with the first stopping criteria would yield a random walk with many layer changes which would correspond to a high number of changes of means of transportation. For this restriction, as long the allowed number of layer changes is not exceeded yet, the agent picks uniformly at random a node from the set of all possible neighbor nodes in all layers. If the picked node can be reached within the current layer, this connection is chosen and the number of layer changes does not increase. Otherwise the layer is changed. If the node cannot be reached within the current layer, but in more than one layer, the new layer is chosen randomly. When the agent has performed the maximum number of allowed layer changes, it stops.

Graph coverage Is the set of covered nodes by the real network flow due to the process properties or is the real network flow constrained by the network structure such that a random network flow will cover the same set of nodes?

We therefore implement several variants of random agents exploring the graph as described in the previous paragraph and compare the graph coverage of the observed trajectories with the coverage by the random agents. Note that the number of agents, their start nodes and their path lengths are fixed by the observed trajectories.

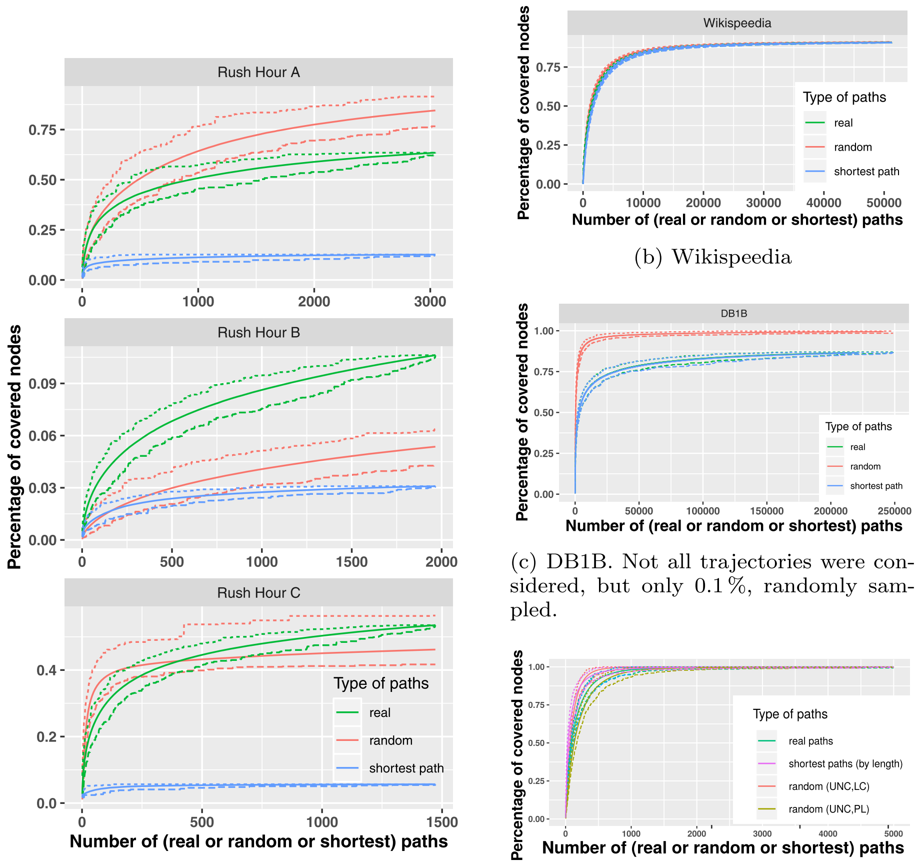
We repeat the coverage experiment with increasing number of paths, as described in "Why network flows need to be considered" section with the following addition: when a

Algorithm 1: Procedure of computing the coverage values by the observed trajectories, shortest paths, and random walks.

```
Data: graph $G = (V, E)$, set of walks $\mathcal{P}$ in $G$
for 500 times do
    Initialization:
    Sequence of coverage values by observed trajectories: $C = ()$
    Sequence of coverage values by shortest paths: $C_{sp} = ()$
    Sequence of coverage values by random walks: $C_{rand} = ()$
    $\mathcal{P}' = \emptyset$
    $\mathcal{P}_{RAND} = \emptyset$
    $\mathcal{P}_{SP} = \emptyset$
    while $\mathcal{P} \neq \emptyset$ do
        draw (and remove) $Q$ from $\mathcal{P}$
        add $Q$ to $\mathcal{P}'$
        compute coverage by $\mathcal{P}'$ and append it to $C$
        compute shortest path from $s(Q)$ to $t(Q)$, add this shortest path to $\mathcal{P}_{SP}$
        compute coverage by $\mathcal{P}_{SP}$ and append it to $C_{sp}$
        perform random walk: start a random agent in $s(Q)$, agent moves randomly to
        one of the neighbors of current nodes (restricted by neighbor choice
        mechanism), until stopping criterion is reached.
        add random walk to $\mathcal{P}_{RAND}$
        compute coverage by $\mathcal{P}_{RAND}$ and append it to $C_{rand}$
    end
    save $C$, $C_{sp}$ and $C_{rand}$
    reset $\mathcal{P}$ to initially given set of walks
end
```

(real) trajectory Q from $\mathcal{P}$ is drawn, the shortest path $SP(Q)$ from $s(Q)$ to $t(Q)$ is computed and added to an initially empty set $\mathcal{P}_{SP}$, and a random walk tied to Q is performed and added to an initially empty set $\mathcal{P}_{RAND}$. In every iteration, the coverage by the shortest paths $\mathcal{P}_{SP}$ and the coverage by the random walks $\mathcal{P}_{RAND}$ is computed. This procedure is repeated $N = 500$ times and the average, minimum and maximum coverage values over all N iterations is computed. The procedure is sketched in Algorithm 1.

Figure 8 shows the graph coverage with increasing number of observed trajectories, shortest paths and random walks. We find that the coverage by observed trajectories and shortest paths coincide perfectly for the DB1B data set, approaching a coverage value of almost 90 %. At the same time, the coverage of the random paths reaches a coverage close to 100 %. Note that the complete set of observed trajectories of DB1B covers the whole graph, while the sampled subset (although containing more than 80000 trajectories) does not. Since the subset of trajectories was sampled proportionally to their frequency (a trajectory which occurs more often in $\mathcal{P}$ has a higher probability to be sampled than a trajectory which is contained less often), this implies that there is a subset of airports which are served occasionally by flights, but much less often than the majority of airports. For Wikispeedia, observed trajectories, random walks and shortest paths yield approximately the same coverage. This is surprising because we saw that the observed trajectories of this



(b) Wikispeedia

(c) DB1B. Not all trajectories were considered, but only 0.1 %, randomly sampled.

(a) Data set with Rush Hour games

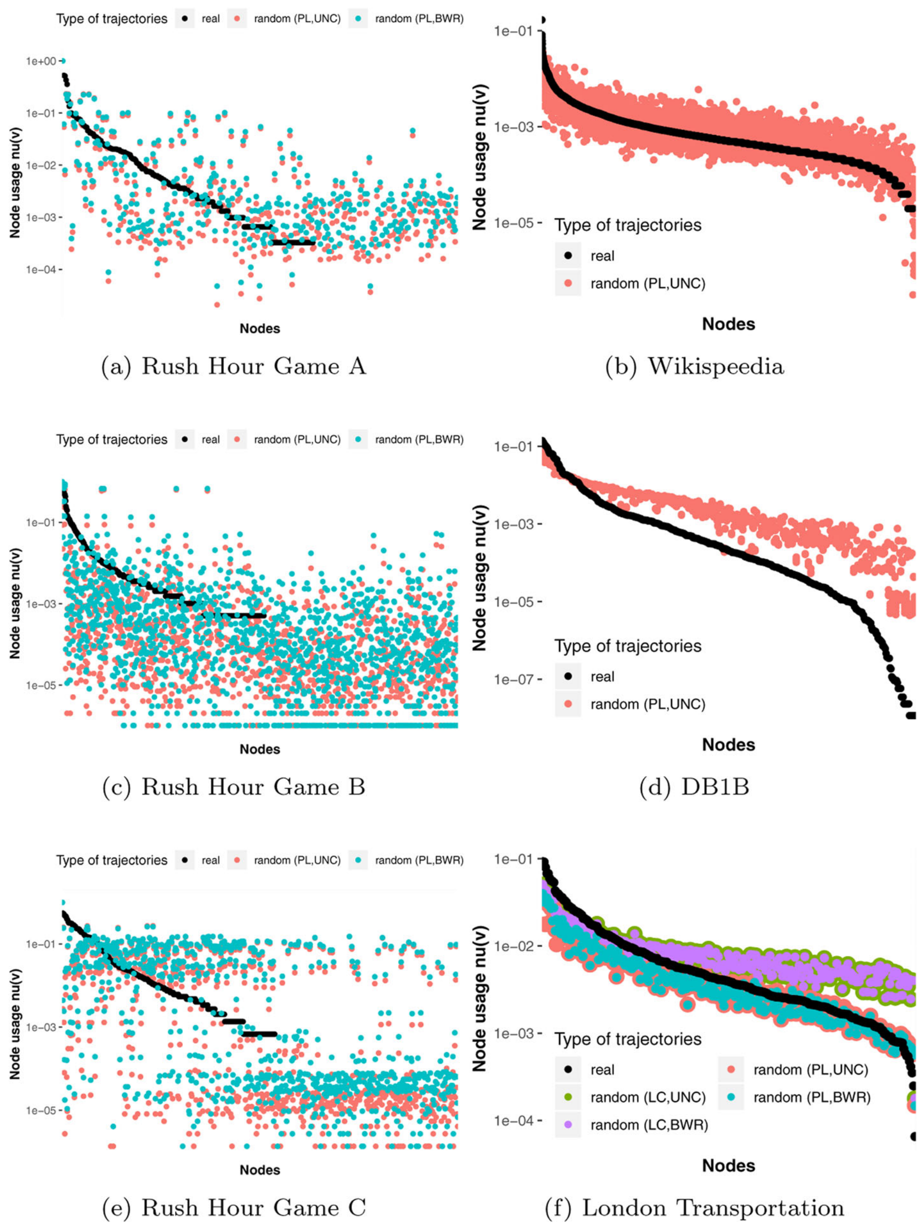
(d) London Transport. Not all trajectories were considered, but 10 %, randomly sampled.

Fig. 8 Coverage of the networks by observed trajectories, shortest paths, and random walks. The solid line represents the mean coverage value over the $N = 500$ iterations, the dashed lines the minimal and maximal coverage values over the 500 iterations

data set are not shortest paths, but longer than the shortest path by on average only one step. They are hence neither shortest paths nor random paths, but still show the same pattern of coverage.

For London Transport, the coverage of the different types of walks shows a similar behaviour and reaches a saturation close to 100 % very fast. It is, although close to each other, interesting that the set of shortest paths and the set of random walks restricted by line changes, exhibit a faster coverage than the sets of observed trajectories and random walks restricted by path length.

For the Rush Hour data sets, the shortest paths only cover a small subset of the graph. This, however, is not surprising because in this data set, there is only one start and a few target nodes for all observed trajectories. We find that the random walks and observed trajectories cover nearly the same number of nodes for game C, but not for games A and B. For game A, the random walks cover more nodes (and almost the complete graph), for game B, the observed trajectories cover more nodes than the random walks. This is due to the graph size of several thousands of nodes a smaller percentage than for game A.



(a) Rush Hour Game A (b) Wikispeedia

(c) Rush Hour Game B (d) DB1B

(e) Rush Hour Game C (f) London Transportation

Fig. 9 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, if applicable, by the weight (PW) of the corresponding real trajectory

Node usage We compare the node usage distribution of the real-world process to the one by the random walks. Figure 9 shows the cumulative distribution of the node usage of the real-world process trajectories and the (average) node usage of the random walks. Figure 10 shows for each node its real usage and its usage by the random walks: for this purpose, the nodes are plotted on the x-axis, ordered by their usage by the observed trajectories, and each node's usage by the observed trajectories and the random walks is plotted by one point.

We can make several observations:

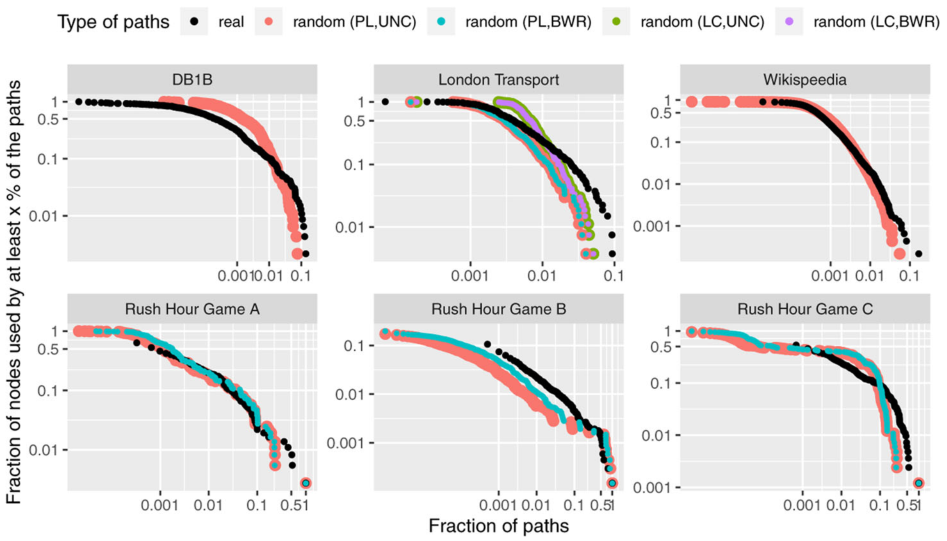


Fig. 10 Node usage. Each point represents one node and its node usage (by random walks or observed trajectories), ordered by the node usage by observed trajectories

(i) For the transportation data sets and for Wikispeedia, the node usage by the real and random walks show a high correlation (Pearson correlation coefficient between 0.81 and 0.87). On the same time, the node usage distributions are different: the cumulative node usage is somehow flattened for the random walks, i.e., the nodes used most often by the observed trajectories are also used most often by the random walks (which are actually the same set of nodes for both types of walks, see Fig. 10), but the absolute node usage value is smaller for the random walks than for the real walks. On the other side of the scale, the effect is opposite: the smallest values for the node usage by the random walks are not as small as the smallest values for the node usage by the observed trajectories. In other words, the node usage by the observed trajectories is more extreme: highly visited nodes are used very often, less visited nodes are used very rarely.

(i) For Rush Hour, the correlation of the real and random node usage is lower (Pearson correlation coefficient of 0.77 (game A), 0.64 (game B), and 0.36 (game C)). We still observe the effect that the highly used nodes have a higher node usage for the observed trajectories than for the random walks, at least for game A and C.

Node pair usage The same observation can be made when considering the node pair usage of the random walks. Since the generation of the random walks are tied to the set of observed trajectories, the start node distribution will be the same as for the observed trajectories, the source-target-distribution however is different. Figure 11 shows the cumulative distribution of the node pair usage of the random walks with uniform neighbor choice. Also here, we observe that although the random walks were tied to the observed trajectories, the most demanded node pair by the observed trajectories is more demanded than the most demanded node pair by the random walks. On the same time, the least demanded node pairs by the observed trajectories are less demanded by the least demanded node pairs by the random walks.

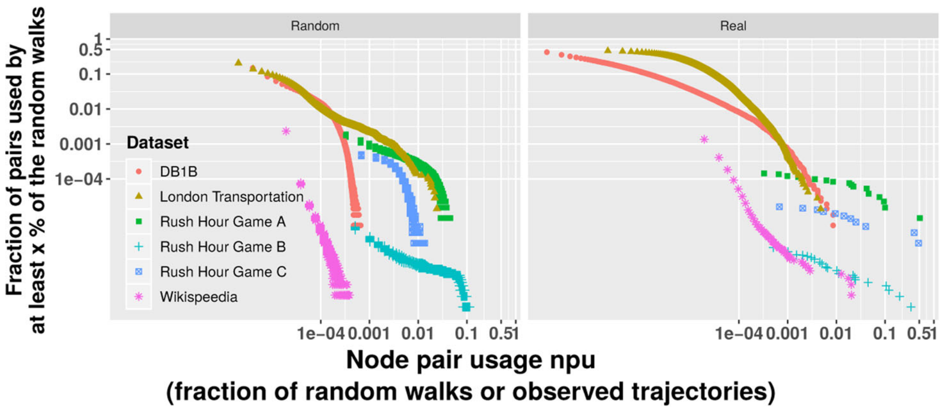


Fig. 11 Cumulative distribution of source-target-frequency of the random walks (left, with UNC and PL restriction) and observed trajectories (right)

Effect on network measures

The previous sections showed that the network flow contained in real-world data sets is not uniform in the sense that all nodes and all node pairs are used equally often. In order to distinguish whether the found usage pattern of the real-world network flow is due to its source-target-distribution or is due to the type of taken trajectories, the previous section introduced a simulation based on random walks. Also when fixing the source distribution and the outreach potential of the random walks to the observed ones from the data sets, different usage patterns are found.

Those usage pattern of the real-world network flows and their corresponding random walks are not as expected by some network measures, particularly by several centrality indices. In this section, we consider whether the violation of their assumptions has an impact on the measures' results. Therefore, for the network of each data set, we compute several standard network measures, the nodes' degree, their closeness centrality, their betweenness centrality and their PageRank centrality (Page et al. 1999) and compare the standard centrality measures to its empiric equivalent, i.e., to the node usage by real trajectories and by the random walks as introduced in "Modeling trajectories by random walks" section. The node usage by the random and real trajectories can be seen as a kind of empiric betweenness since it represents the number of trajectories a node is contained in.

We compute the Spearman correlation coefficient of each standard network measure to the node usage by random and real trajectories (see Table 4).

We observe that the nodes' empiric importance, measured by node usage, shows a high correlation to all chosen network measures only for the DB1B data set. Here, all correlation coefficients are above 0.85. For the remaining data sets, the correlations of the nodes' empiric importance are smaller, for the Rush Hour games, the correlation coefficients are even negative for some cases. Interestingly, the correlations of each network measure are approximately equal whether compared to the node usage by real trajectories or whether compared to the node usage by random walks. Furthermore, it can be observed that the correlation coefficients do not show a large variation among the different standard measures–for one data set, the correlation coefficients are similar for each measure.

Table 4 Spearman correlation coefficients of classic network measures (node degree, closeness, betweenness, and PageRank centrality) to the actual node usage by real-world trajectories (left part of the table) and to the node usage by random trajectories (as introduced in "Modeling trajectories by random walks" section. Starred values indicate a *p*-value of < 0.05

	Node usage by real trajectories				Node usage by random walks			
Data set	Degree	Closeness	Betw	Pagerank	Degree	Closeness	Betw	Pagerank
DB1B	0.96*	0.94*	0.89*	0.93*	0.97*	0.97*	0.87*	0.91*
LT	0.50*	0.51*	0.54*	0.54*	0.45*	0.47*	0.50*	0.50*
Wiki	0.71*	0.77*	0.72*	0.83*	0.79*	0.89*	0.77*	0.96*
RH A	0.007	0.15*	0.36*	0.16*	0.10	0.16*	0.39*	0.19*
RH B	-0.15*	-0.29*	0.03*	0.01	-0.22*	-0.45*	-0.05*	0.02
RH C	-0.21*	-0.12*	0.29*	0.01	-0.10*	-0.33*	0.038	-0.02

Although the data set DB1B shows the highest correlation between empiric importance and importance measured by standard centrality measures, this data set contains an illustrative example why standard centrality measures sometimes fail in quantifying a node's importance: The node representing the airport Anchorage in Alaska is rated as one the most central nodes by the standard betweenness centrality. When measuring its importance by the actual number of passengers, its importance drops considerably (its node usage by real trajectories is about 5 % of the node usage of the most used airport, being the 52nd most used airport in the data set). In this case, it has an impact that the assumption of equal amount of communication between each pair of nodes is violated: when only considering the static network structure, the node Anchorage is a gateway between the airports in the contiguous United States and the airports in Alaska. Hence, Anchorage is on all shortest paths between the airports in the contiguous United States and the airports in Alaska, yielding a high betweenness value for Anchorage. However, most of these source-target pairs are only used rarely or not at all by the real-world network flow, which is why the high importance of the node Anchorage is not supported by the empiric betweenness.

Discussion and limitations

The results presented in the previous sections showed that real-world processes use the network in a different way than the implicit process model used for the computation of standard network measures, such as centrality measures: there are a few nodes and node pairs between which a large proportion of the real flow takes place, while there is almost no flow between most node pairs. This holds for all considered process data from different contexts. When fixing the number of process entities and start node distribution, an agent-based simulation based on several random walk strategies leads to different node usage and node pair usage distributions. On the same time, the network coverage behaviour shows similar characteristics for real and random walks. Additionally, the usage of the nodes by real and random walks show a high correlation for three of the four data sets. We furthermore demonstrated that this actually does have an effect on network measures, demonstrated for standard centrality measures in "Effect on network measures" section.

Those results have relevance for network analysis in general because it shows that real-world processes show different characteristics than predicted by the simple process models implicitly contained in network measures. There are, however, several caveats in the interpretation of the results. The validity of the results depends on the quality of the used data sets. The data sets used in this work were collected under different circumstances and with different objectives. The passenger journeys of DB1B and London Transportation were collected and provided by the official governmental authorities for transportation. While the flight data seems to be a representative sample of all flight tickets, the travel data of the London transportation system does only contain the journeys of passengers using a special electronic ticket system. According to information by the authority Transport of London, in 2012, 80 % of all journeys was done by using this ticket system (Transport for London 2012). It is not clear whether journeys contained in the available data set are representative for the process of people traveling in London. The influence of the data collection method can be seen on the example of the Wikispeedia data set. The game logs were collected via a publicly available web site where a player could either choose a pair of nodes as source and target for the game or a pair of nodes was randomly chosen and assigned to the player. However, for a certain experiment (West and Leskovec 2012a), there are four node pairs which were suggested with an increased frequency. Hence, any analysis considering the frequency of source-target-pairs of the observed trajectories have very limited relevance since it is not clear by which methode the given source and target node were chosen.

Furthermore, we expect an impact of decisions in data preprocessing, such as construction of the trajectories, filtering out of trajectories, etc. on the results of the analysis. Since the results of a network analytic project are sensitive to decisions taken when constructing the network, similar effects can be expected when analyzing process data. Therefore, a robustness analysis investigating the potential impact of modeling decision on the results would be needed.

Despite the named limitations, we are still convinced that the results are relevant for the network science community and argue for the importance of a process-driven network analysis. We showed that the characteristics of real-world network flows strongly deviate from the characteristics of their simplistic model contained in network measures. This is the case for all considered real-world network flows. Any network analytic project should include the question of which network process is of interest for the research question. Decisions in constructing the network representation and the choice of network analytic methods need to be evaluated with respect to the process of interest.

Conclusion

A network representation is a useful representation of a system if *indirect* effects are of interest (Brandes et al. 2013). The presence of indirect effects can often be explained by something flowing through the network–what is also called a network process. Not only the representation as network itself, but also many network analytic methods assume the presence of a network process, for example all distance-based measures or most centrality measures (Borgatti 2005). We argue that a meaningful network analysis needs to account for the process of interest: the network representation cannot be chosen independently from the process of interest, and also the set of applicable network methods is restricted by the process of interest (Borgatti 2005; Dorn et al. 2012; Zweig 2016; Butts 2009). We call this approach *process-driven network analysis*. In order to demonstrate the relevance of this approach, we collected data sets of real-world transfer processes and first show that they do not suggest a uniform weighting of communication. In a second step, we compare the usage of the network by the real-world process to the usage of the most simple process models, i.e., shortest paths and random walks. We observe that–although the simple models show similar behaviour to the real process with respect to some properties–they are different with respect to others.

Abbreviations

UNC: Uniform neighbor choice; BWR: Backwards-restricted neighbor choice; PL: Path length restriction; PW: Path weight restriction; LC: Line change restriction; RH: Rush Hour; LT: London Transport

Acknowledgements

We thank the anonymous reviewers for their comments which helped to improve the manuscript.

Authors' contributions

MB and KZ conceived of the idea of the work; MB acquired the data, performed the analysis and wrote the initial draft. All author(s) contributed to the writing of the final manuscript and approved the last version.

Funding

Open access funding provided by Projekt DEAL.

Availability of data and materials

The datasets used and/or analysed during the current study are available from the corresponding author on reasonable request.

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Competing interests

The authors declare that they have no competing interests.

Received: 3 March 2020 Accepted: 5 August 2020
Published online: 27 August 2020

References

Adamic L, Adar E (2005) How to search a social network. Soc Networks 27(3):187–203. https://doi.org/10.1016/j.socnet.2005.01.007

Bakshy E, Rosenn I, Marlow C, Adamic L (2012) The role of social networks in information diffusion. In: Proceedings of the 21st International Conference on World Wide Web, WWW '12. ACM, New York, NY, USA. pp 519–528. https://doi.org/10.1145/2187836.2187907

Boccaletti S, Latora V, Moreno Y, Chavez M, Hwang D-U (2006) Complex networks: Structure and dynamics. Phys Rep 424(4-5):175–308. https://doi.org/10.1016/j.physrep.2005.10.009

Bockholt M, Zweig KA (2018) Process-driven betweenness centrality measures. In: Lecture Notes in Social Networks. Springer, Cham. pp 17–33. https://doi.org/10.1007/978-3-319-90312-5_2

Bockholt M, Zweig KA (2019) Why we need a process-driven network analysis. In: Complex Networks and Their Applications VIII. Springer, Cham. pp 81–93. https://doi.org/10.1007/978-3-030-36683-4_7

Borgatti SP (2005) Centrality and network flow. Soc Networks 27(1):55–71. https://doi.org/10.1016/j.socnet.2004.11.008

Brandes U, Erlebach T, (eds.) (2005) Network Analysis. Springer, Berlin Heidelberg. https://doi.org/10.1007/b106453

Brandes U, Robins G, McCranie A, Wasserman S (2013) What is network science? Netw Sci 1(1):1–15. https://doi.org/10.1017/nws.2013.2

Butts CT (2009) Revisiting the foundations of network analysis. Science 325(5939):414–416. https://doi.org/10.1126/science.1171022

Centola D (2010) The spread of behavior in an online social network experiment. Science 329(5996):1194–1197. https://doi.org/10.1126/science.1185231

Cheng J, Adamic L, Dow PA, Kleinberg JM, Leskovec J (2014) Can cascades be predicted? In: Proceedings of the 23rd International Conference on World Wide Web, WWW '14. Association for Computing Machinery, New York, NY, USA. pp 925–936. https://doi.org/10.1145/2566486.2567997

Chierichetti F, Kumar R, Raghavan P, Sarlos T (2012) Are web users really markovian? In: Proceedings of the 21st International Conference on World Wide Web, WWW '12. Association for Computing Machinery, New York, NY, USA. pp 609–618. https://doi.org/10.1145/2187836.2187919

Christakis NA, Fowler JH (2007) The spread of obesity in a large social network over 32 years. N Engl J Med 357(4):370–379. https://doi.org/10.1056/nejmsa066082

Christakis NA, Fowler JH (2008) The collective dynamics of smoking in a large social network. N Engl J Med 358(21):2249–2258. https://doi.org/10.1056/NEJMsa0706154, PMID: 18499567

Colizza V, Barrat A, Barthélemy M, Vespignani A (2006) The role of the airline transportation network in the prediction and predictability of global epidemics. Proc Natl Acad Sci 103(7):2015–2020. https://doi.org/10.1073/pnas.0510525103

De Choudhury M, Mason WA, Hofman JM, Watts DJ (2010) Inferring relevant social networks from interpersonal communication. In: Proceedings of the 19th International Conference on World Wide Web, WWW '10. Association for Computing Machinery, New York, NY, USA. pp 301–310. https://doi.org/10.1145/1772690.1772722

De Domenico M, Lima A, Mougel P, Musolesi M (2013) The anatomy of a scientific rumor. Sci Rep 3:2980. https://doi.org/10.1038/srep02980

Dorn I, Lindenblatt A, Zweig KA (2012) The trilemma of network analysis. In: Proceedings of the International Conference on Advances in Social Networks Analysis and Mining (ASONAM). IEEE, Washington, DC, USA. pp 9–14. https://doi.org/10.1109/ASONAM.2012.12

Freeman LC (1977) A set of measures of centrality based on betweenness. Sociometry 40(1):35–41. https://doi.org/10.2307/3033543

Freeman LC, Borgatti SP, White DR (1991) Centrality in valued graphs: A measure of betweenness based on network flow. Soc Networks 13(2):141–154

Friedkin NE (1983) Horizons of observability and limits of informal control in organizations. Soc Forces 62(1):54–77. https://doi.org/10.1093/sf/62.1.54

Ghosh R, Lerman K (2014) Rethinking centrality: The role of dynamical processes in social network analysis. Discrete & Continuous Dynamical Systems - B 19(5):1355–1372. http://dx.doi.org/10.3934/DCDSB.2014.19.1355,

Gilbert M, Mitchell A, Bourn D, Mawdsley J, Clifton-Hadley R, Wint W (2005) Cattle movements and bovine tuberculosis in great britain. Nature 435(7041):491–496. https://doi.org/10.1038/nature03548

Girvan M, Newman MEJ (2002) Community structure in social and biological networks. Proc Natl Acad Sci 99(12):7821–7826. https://doi.org/10.1073/pnas.122653799

Granovetter MS (1973) The strength of weak ties. Am J Sociol 78(6):1360–1380. https://doi.org/10.1086/225469

Helic D, Strohmaier M, Granitzer M, Scherer R (2013) Models of human navigation in information networks based on decentralized search. In: Proceedings of the 24th ACM Conference on Hypertext and Social Media, HT '13. ACM, New York, NY, USA. pp 89–98. https://doi.org/10.1145/2481492.2481502

Holme P (2003) Congestion and centrality in traffic flow on complex networks. Adv Compl Syst 06(02):163–176. https://doi.org/10.1142/s0219525903000803

Jarušek P, Pelánek R (2012) Analysis of a Simple Model of Problem Solving Times. In: Cerri, Stefano A. and Clancey, William J. and Papadourakis, Giorgos and Panourgia Kitty (eds). Intelligent Tutoring Systems. Lecture Notes in Computer Science, vol. 7315. Springer, Berlin Heidelberg. pp 379–388. https://doi.org/10.1007/978-3-642-30950-2_49

Kleinberg JM (2000) Navigation in a small world. Nature 406:845–845. https://doi.org/10.1038/35022643

Koschützki D, Lehmann KA, Peeters L, Richter S, Tenfelde-Podehl D, Zlotowski O (2005) Centrality indices. In: Brandes U, Erlebach T (eds). Network Analysis: Methodological Foundations. Lecture Notes in Computer Science, vol. 3418. Springer, Berlin, Heidelberg. pp 16–61. https://doi.org/10.1007/978-3-540-31955-9_3

Lambiotte R, Sinatra R, Delvenne J-C, Evans TS, Barahona M, Latora V (2011) Flow graphs: Interweaving dynamics and structure. Phys Rev E 84(1):017102. https://doi.org/10.1103/physreve.84.017102

Liben-Nowell D, Kleinberg J (2007) The link-prediction problem for social networks. J Am Soc Inf Sci Technol 58(7):1019–1031. https://doi.org/10.1002/asi.20591

Manley E (2015) Estimating urban traffic patterns through probabilistic interconnectivity of road network junctions. PLOS ONE 10(5):0127095. https://doi.org/10.1371/journal.pone.0127095

Manley EJ, Addison JD, Cheng T (2015) Shortest path or anchor-based route choice: a large-scale empirical analysis of minicab routing in london. J Transp Geogr 43:123–139. https://doi.org/10.1016/j.jtrangeo.2015.01.006

Masuda N, Porter MA, Lambiotte R (2017) Random walks and diffusion on networks. Phys Rep 716-717:1–58. https://doi.org/10.1016/j.physrep.2017.07.007

Milli L, Rossetti G, Pedreschi D, Giannotti F (2018) Complex Networks & Their Applications VI. In: Cherifi C, Cherifi H, Karsai M, Musolesi M (eds). Springer, Cham. pp 305–313. https://doi.org/10.1007/978-3-319-72150-7_25

Milo R (2002) Network motifs: Simple building blocks of complex networks. Science 298(5594):824–827. https://doi.org/10.1126/science.298.5594.824

Moody J (2002) The importance of relationship timing for diffusion. Soc Forces 81(1):25–56. https://doi.org/10.1353/sof.2002.0056

Newman ME (2005) A measure of betweenness centrality based on random walks. Soc Networks 27(1):39–54. https://doi.org/10.1016/j.socnet.2004.11.009

Page L, Brin S, Motwani R, Winograd T (1999) The PageRank citation ranking: Bringing order to the web Technical report, Stanford InfoLab

Pastor-Satorras R, Vespignani A (2001) Epidemic spreading in scale-free networks. Phys Rev Lett 86(14):3200–3203. https://doi.org/10.1103/physrevlett.86.3200

Pons P, Latapy M (2005) Computing communities in large networks using random walks. In: Computer and Information Sciences - ISCIS 2005. Springer, Berlin, Heidelberg. pp 284–293. https://doi.org/10.1007/11569596_31

RITA TransStat (2016) Origin and Destination Survey database (DB1B)

Rosvall M, Esquivel AV, Lancichinetti A, West JD, Lambiotte R (2014) Memory in network flows and its effects on spreading dynamics and community detection. Nat Commun 5:1–13. https://doi.org/10.1038/ncomms5630

Scholtes I (2017) When is a network a network? multi-order graphical model selection in pathways and temporal networks. In: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '17. ACM, New York, NY, USA. pp 1037–1046. https://doi.org/10.1145/3097983.3098145

Sen R, Hansen MH (2003) Predicting web users' next access based on log data. J Comput Graph Stat 12(1):143–155. https://doi.org/10.1198/1061860031275

Stephenson K, Zelen M (1989) Rethinking centrality: Methods and examples. Soc Networks 11(1):1–37. https://doi.org/10.1016/0378-8733(89)90016-6

Sudarshan Iyengar S, Veni Madhavan C, Zweig KA, Natarajan A (2012) Understanding human navigation using network analysis. Top Cogn Sci 4(1):121–134. https://doi.org/10.1111/j.1756-8765.2011.01178.x

Tavassoli S, Zweig KA (2016) Most central or least central? how much modeling decisions influence a node's centrality ranking in multiplex networks. In: 2016 Third European Network Intelligence Conference (ENIC), Wroclaw. pp 25–32. https://doi.org/10.1109/ENIC.2016.012

Transport for London (2012) Join in the celebrations across the capital this summer with a limited edition summer oyster card. https://tfl.gov.uk/info-for/media/press-releases/2012/june/join-in-the-celebrations-across-the-capital-this-summer-with-a-limited-edition-summer-oyster-card. Accessed 29 Nov 2019

Transport for London (2017) Rolling Origin and Destination Survey (RODS). http://www.tfl.gov.uk/info-for/open-data-users/our-feeds. Accessed 29 Nov 2019

Wang P, González MC, Hidalgo CA, Barabási A-L (2009) Understanding the spreading patterns of mobile phone viruses. Science 324(5930):1071–1076. https://doi.org/10.1126/science.1167053

Weng L, Ratkiewicz J, Perra N, Gonçalves B, Castillo C, Bonchi F, Schifanella R, Menczer F, Flammini A (2013) 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. ACM, New York, NY, USA. pp 356–364. https://doi.org/10.1145/2487575.2487607

West R, Leskovec JBreslin JG, Ellison NB, Shanahan JG, Tufekci Z (eds) (2012) Human wayfinding in information networks. ICWSM. The AAAI Press, New York. https://doi.org/10.1145/2187836.2187920

West R, Leskovec J (2012) Automatic versus human navigation in information networks. In: Breslin JG, Ellison NB, Shanahan JG, Tufekci Z (eds). ICWSM. The AAAI Press

West R, Paranjape A, Leskovec J (2015) Mining missing hyperlinks from human navigation traces: A case study of wikipedia. In: Proceedings of the 24th International Conference on World Wide Web, WWW '15. International World Wide Web Conferences Steering Committee, Geneva. pp 1242–1252. isbn = 9781450334693, https://doi.org/10.1145/2736277.2741666

West R, Pineau J, Precup D (2009) Wikispeedia: An online game for inferring semantic distances between concepts. In: Proceedings of the 21st International Jont Conference on Artifical Intelligence, IJCAI'09. Morgan Kaufmann Publishers Inc, San Francisco, CA, USA. pp 1598–1603

Xu J, Wickramarathne TL, Chawla NV (2016) Representing higher-order dependencies in networks. Sci Adv 2(5):1600028. https://doi.org/10.1126/sciadv.1600028

Zweig KA (2016) Network Analysis Literacy. Springer, Vienna. https://doi.org/10.1007/978-3-7091-0741-6

Zweig K (2018) Friedkin 1983: Horizons of observability and limits of informal control in organizations. In: Schlüsselwerke der Netzwerkforschung. Springer, Wiesbaden. pp 213–215. https://doi.org/10.1007/978-3-658-21742-6_48

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.