

# Weighted networks in socio-technical systems: Concepts and challenges

Z Stojanov<sup>1</sup>, J Stojanov<sup>1</sup>, G Jotanovic<sup>2</sup> and D Dobrilovic<sup>1</sup>

<sup>1</sup> Technical faculty "Mihajlo Pupin", University of Novi Sad, Serbia

<sup>2</sup> Faculty of transport and traffic engineering, University of East Sarajevo, Republic of Srpska, Bosnia and Herzegovina

E-mail: zeljko.stojanov@uns.ac.rs

**Abstract.** Socio-technical systems join together humans and technique. Basic concepts and principles of socio-technical systems are outlined, as well as weighted networks as the appropriate mathematical models. Particular examples of socio-technical systems with various usages of weighted networks in domains such as airline connection networks, scientific collaboration, social networks, software engineering and urban traffic are presented. Comparative analysis of the selected examples is outlined with the focus on the nature and functions of nodes, links and weights. The identified challenges, such as the creation of networks, evolving nature of systems, and the need for multidisciplinary teams in the system design are discussed.

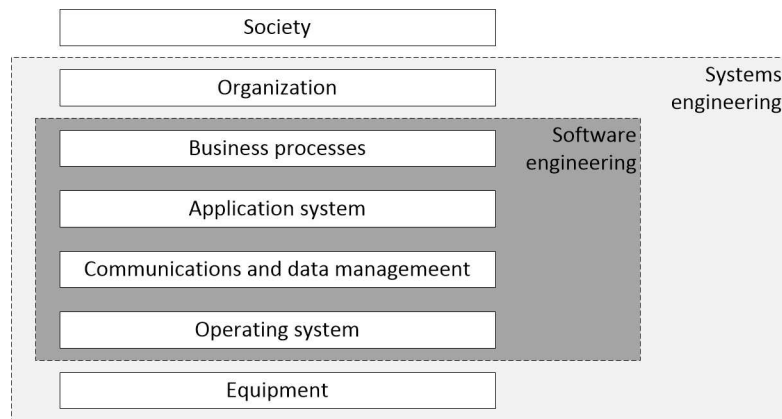
## 1. Introduction

The concept of socio-technical systems was established in the fifties of the last century in order to envelope complex relation between humans and technical systems in industry. The socio-technical concept has developed in terms of systems because it is concerned with interdependencies within an organization, as well as in terms of open system theory since it is also concerned with the environment in which the organization exists [1].

Socio-technical system is a collection of social and technical elements engaged in purposeful goal-directed manner for achieving specific behavior [2]. Due to the intrinsic complexity of contemporary socio-technical systems, their understanding requires a broad knowledge of both technical and social disciplines. Therefore, perhaps nobody has enough knowledge and understanding of socio-technical systems. Engineers and technicians usually ignore social aspects of their work, while social scientists know very little about technology and technical systems. Ropohl [3] suggested that it is important to shape both the technical and the social aspects of work in a way that preserves both humanity and efficiency. The complexity of relationships between the technical and social aspects of an organization reveals that it is not sufficient to inquire them separately, requiring new approaches for researching socio-technical systems. It has become clear that social and technical aspects should be considered interdependently, which requires dual focus and joint optimization of these aspects within an organization [4]. However, this optimization requires a wide range of knowledge and techniques because socio and technical parts do not behave the same way, making non-linear relationships and behavior common in

these systems. In this course of thinking, Fuenfschilling and Truffer [5] introduced the concept of the socio-technical regime for addressing long-term changes in various sectors in industry, which actually denotes co-evolution of institutional and technological aspects over time.

Business and overall organization of contemporary organizations depend on IT and software systems that provide meaningful communication of humans and equipment, as well as data processing. This situation emphasizes the importance of software systems in complex socio-technical systems that exist in organizations, increasing the significance of software engineering methods, tools and techniques for overall performance of socio-technical systems. Software systems are used in all segments of human life, such as education, health, transport, industry, communication, entertainment, etc. Therefore, activities related to the software life cycle cannot be treated as isolated activities regarding more complicated systems that use software. Actually, software engineering activities should be treated as an intrinsic part of systems engineering processes. Sommerville [6] proposed the socio-technical systems stack that presents socio-technical systems with layered structure, as it is shown in Figure 1. Software engineering part in socio-technical systems includes operating system, communication and data management, application system and business processes. System engineering envelopes equipment, software engineering part and organization.



**Figure 1.** Layered structure of socio-technical systems

This model stresses the importance of having a holistic view of systems, which helps in observing complex relationships between different parts, whether technical or social. This view on systems assumes that, when designing one part, it is important to observe the influence on other parts. For example, when designing a specific equipment in the industry, it is important to carefully consider software elements that will assist in using and tracking equipment use, as well as people that will operate both software and equipment. Since socio-technical systems are situated within organizations, the following organizational factors affect their design and operation [6]: *organizational changes* - changes in overall organization or changes in decision-making structures, *process changes* - changes in internal process organization or interprocess communication, and *job changes* - changes in description of work that may render some skills obsolete and require some new skills.

Complex design of socio-technical systems imposes an abstract and low level framework. Graph theory, as a mathematical discipline, produces models for socio-technical systems that are simple enough but analyzable [7]. Mathematical properties of a model are good representatives of a socio-technical systems' characteristics. Moreover, the model enables introduction of various metrics useful for detailed determination of socio-technical systems.

This paper outlines the development of socio-technical concepts and principles in the next section. The third section presents the basics of weighted networks and their use in socio-technical systems. The fourth section presents a comparative analysis of the selected examples and discussion of identified challenges. The last section contains conclusions and further research directions.

## 2. Development of socio-technical concepts and principles

Review of literature dealing with socio-technical systems, from the early work of Trist and Bamforth [8] to recent studies [2, 9], helps in refining the basic and enduring socio-technical principles. The first identified basic principles are: responsible autonomy, adaptability, and meaningfulness of tasks. These principles should ensure the best possible performance at both individual and organizational levels.

Further research on socio-technical systems has been directed toward their implementation in various fields of society and economy, which resulted in refinement of the basic principles. Albert Cherns [10] distilled experiences on socio-technical system design principles from the early period of research, and provided more detailed list of principles:

- *Compatibility.* The design process should be compatible with objectives (self-modified, adaptive to changes), which assumes active participation of people in the design of their jobs.
- *Minimal critical specification.* This principle assumes precise specification of what is essential for the system, including minimal specification of tasks, allocation of tasks to jobs and jobs to roles. Only essential objectives should be specified.
- *The sociotechnical criterion.* Identification and prevention of variance, or unpredictable events, that affect system functioning should be planned and included in system design.
- *The multifunctional principle.* Design should consider option that system functions can be performed in different ways, which requires people with different skills that can perform highly specialized tasks.
- *Boundary location.* Boundaries are used for grouping people and activities based on technology, territory and time, which requires careful management of boundaries between groups, departments, and the whole organization.
- *Information flow.* All necessary information should be provided at the place where tasks execution requires. The right information should be provided to all people and groups, which support learning and better control of processes.
- *Support congruence.* The system should support all behaviors that an organization requires.
- *Design and human values.* Organization should enable high quality of work, which is usually subjective judgment based on psychological assumptions.
- *Incompletion.* Design is iterative process, which means that all decisions may lead to new design decisions and changes in design.

From its beginning devoted to productivity enhancement in various machine-driven industries, socio-technical thinking has evolved towards contemporary systems based on knowledge work and capabilities of workers. This evolution of socio-technical thinking affected design of contemporary systems, with focus on agility in redesigning systems due to dynamic changes in business and society [9]. Based on practical experience, Hirschhorn et al. [11] suggested that in the time of mass customization in the economy, the concept of meaning should replace the concept of autonomy (worker and task), which leads to more productive workers and learning organizations that are more responsive to the challenges. This evolution of contemporary socio-technical systems is primarily shaped with development and use of information technology,

**Table 1.** Principles of socio-technical systems design

<b>Principle</b>	<b>Description</b>
Wholeness	A set of activities in a work system should make a functioning whole, rather than a collection of individual jobs.
Teams	The work group should be considered a central unit in an organization.
Process control	All variances (problems or deviations from expectations) should be identified and handled as soon as they occur.
Self-direction	System should rely on internal regulation rather than on external ones.
Multi-skilling	System should be based on redundancy of functions rather than on a redundancy of parts.
Discretion	The discretionary component of the work is essential to the success of the system.
Joint optimization	System optimization should include both individuals and technical parts.
Adaptation	System should easily adapt to changes, which is based on individual and organizational learning.
Meaning	Individual jobs should be designed to support learning, some level of decision making, and should be socially recognizable.
Incompletion	Due to constant change of environment and organization, job design cannot be treated as finished.

digitization and advanced technologies. The principles of socio-technical system design summarized by Pasmore et al. [9] are presented in table 1

The next technology movement that significantly has affected the development and implementation of socio-technical systems is the spread of IoT and smart technologies. Concept of smart cities is widely researched from a technical perspective with focus on urbanization problems, while the social component is usually neglected [12]. Shin and Jin Park [13] offered a contextualized socio-technical analysis of IoT, providing insight into its challenges and opportunities, which helps in understanding how IoT can be designed and situated within human-centered contexts. The study reveals implications on conducting design according to a combined technical and social views and how to structure design processes in multi-disciplinary teams. A user-centered approach is required to map out the IoT design space, to understand how to avoid negative individual and social consequences and to support acceptable and useful information sharing. Since smart cities are based on technology which affect people, organizations and society, they can be observed as socio-technical systems. Smart cities include variety of networks and infrastructures, such as city services, business, transport, communication, education, health-care, water and energy supply. One of the most challenging aspects of smart cities is smart transportation aimed at solving traffic congestion problem or pollution problems by providing the most relevant information to participants in traffic [14]. Graph theory is commonly used for modeling and analyzing various types of networks that models socio-technical issues in local and urban traffic systems [15].

The vast majority of modern systems can be classified as socio-technical systems. Research

and implementation issues go beyond general principles of socio-technical systems and they have to rely on the realistic context where the systems are being introduced. Since these systems are based on people and technological infrastructure, various networks may be identified that reflect relations between people, people and technical objects, and between technical objects. Networks containing people and technical objects have been researched in domains such as education [16, 17], behavioral change for energy saving [18], sustainable urban water management [19, 20], health care systems focused on e-Health interoperability [21] or personal health technology [22], social networks [23], resource sharing in social communication [24], large-scale socio-technical systems on the Web [25], air traffic management systems [26], transit service networks [27], roadside air quality control [28], traffic forecasting [29], work of military staff in steady-state and crisis threat scenarios [30], or software development projects [31, 32].

### 3. Weighted networks in socio-technical systems

#### 3.1. Terminology

Weighted graph is a mathematical notion, abstract enough to model wide variety of real phenomena. Structures of modeled phenomena imposed more popular name, weighted network. It consists of objects of two kinds: vertices (nodes, actors) and edges (links, connections), and a function that quantifies a feature of edges.

More precisely, weighted graph/network is a triple  $(V, E, f)$ , where:

- $V = \{v_i \mid i \in I\}$  is a set of vertices. Their amount  $|V|$  is graph's order. The vertices represent real objects (cities, persons, devices,...) and are usually labeled for practical reasons by  $v_1, v_2, \dots$ .  $I$  is a set of indices that enable distinction of vertices.
- $E = \{e_a \mid a \in A\}$  is a set of edges.  $A$  is a set of indices that enable distinction of edges. Amount of edges  $|E|$  is graph's size. The edges represent a relation among nodes (roads, collaborations, cable links,...). Each edge  $e_a$  is a two-element set of vertices  $e_a = \{v_i, v_j\}$ <sup>1</sup>. The edge  $e_a$  is said to be incident with the vertices  $v_i$  and  $v_j$ . And more, the vertices  $v_i$  and  $v_j$  are adjacent.
- $f : E \rightarrow [0, \infty]$  is the weight function, which assigns to each edge  $e_a$  a number, its weight that  $w(e_a) =: w_{ij}$ . The latter label contains information of the vertices connected by the edge, so it is commonly used. In the case that all edges have the same weight, the weight function is trivial, hence can be omitted and the weighted graph becomes the graph  $(V, E)$ .

Vertices and edges are mathematical terms, but actors and connections, nodes and links, and agents and links are commonly used in social and technical systems.

It could be of interest to consider only a part of the graph, some of the vertices  $V_1 \subset V$  and some of the edges, with overtaken their weights, incident with elements of  $V_1$  contained in  $E_1 \subset E$ . Obtained graph is named as a subgraph of the graph  $(V, E, f)$ .

Topological properties of weighted graph which are of the interest in modeling of socio-technical systems will be presented.

The number of edges which are incident with a vertex  $v_i$  is its degree,  $d(v_i)$ . A set of vertices that are adjacent to a vertex  $v_i$  is the set of neighbors,  $N(v_i)$ , and it contains exactly  $d(v_i)$  elements.

Path is a sequence  $v_0, e_1, v_1, e_2, \dots, v_k$ , where each edge  $e_a, a = 1, \dots, k$  is incident with vertices  $v_{a-1}$  and  $v_a$ . As the path is fully determined with the vertices, the edges in the sequence are commonly omitted. The length of the path is the number of edges it contains. Exclusively seen, the path is a subgraph.

<sup>1</sup> Edge can also be related to couple of vertices  $e_a = (v_i, v_j)$ . In that case, orientation of edges is emphasized and the graph is named as the weighted directed graph.

Two vertices  $v_i$  and  $v_j$  are connected if there is a path with the first element  $v_i$  and the last element  $v_j$ . The length of the shortest path between vertices  $v_i$  and  $v_j$  is their distance,  $d(v_i, v_j)$ . In weighted graph, a path between any two vertices  $v_i$  and  $v_j$  has, besides the length, its weight, obtained as a sum of the weights of all edges in the path. The shortest path in weighted graph implies the minimum weight.

The graph is connected if each two vertices are connected.

Importance of a vertex in connected graph reflects its significance in the connectivity, and is presented by the centrality of vertices. As the shortest path has different meaning relating to the weights, there are two kinds of centralities:

- *centrality* is determined by the notion of eccentricity of a vertex  $v_i$ , denoted by  $ecc(v_i)$ , it is the maximum distance between  $v_i$  and any other vertex in the graph. The maximum eccentricity is the graph diameter, and the vertices possessing it are central vertices. Their collection is called the graph center. So, a vertex  $v$  is the central one if

$$v = \underset{u \in V}{\operatorname{argmin}} ecc(u).$$

Analogously, the minimum eccentricity is the graph radius, vertices possessing it are peripheral vertices and their collection is the graph periphery.

- *betweenness centrality* measures appearance of each vertex in all shortest paths in the graph,

$$g(v) = \sum_{s \neq v \neq t} \frac{\sigma_{st}(v)}{\sigma_{st}},$$

where  $\sigma_{st}$  is the number of shortest paths from vertex  $s$  to vertex  $t$  and  $\sigma_{st}(v)$  is the number of those paths that pass through the vertex  $v$ . Betweenness orders vertices, and the central ones are those with the maximum betweenness.

Weighted networks can be directed, which means that each edge has a starting and target vertex, i.e. the edge is oriented. In directed weighted networks vertex has a degree, but also has input degree and output degree. These basic differences produce variety of new possibilities in their use.

More details on graph theory can be found in [33].

### 3.2. Examples

Barthélemy et al. [34] presented two case studies, the airline connection network and the scientific collaboration network, which are representatives of critical infrastructure and social system respectively. Characterizations of corresponding weighted networks are determined by use of statistical tools. The empirical results confirm the importance of the distributions of the various network's quantities and the existence of weight-topology correlations. The weights relevance is emphasized and the network modeling of complex systems has to go beyond the topology. Produced weighted networks support large heterogeneity in the capacity and intensity of the connections that exists in real networks. Considered model relies on dynamical coupling of topology and weight through the particular local properties: strength of node, weighted clustering, weighted assortativity and disparity. The model supports the weight dynamical evolution occurring when new vertices and edges are introduced in the system. Hence, the evolution of the network can be inspected analytically.

The hidden metric space weighted model introduced in [35] relies with network's topology, its weights and underlying metric space. Based on the metric space, the model produces the weighted architectures of real complex networks. Therefore, the weights of the links contain information of the interest in finding more accurate embeddings of real networks. The underlying

metric space governing the network topology determines the weighted structure of the network. Framework of the model can be seen as a novel generalization of the gravity concept, and it could reproduce the existence and the intensity of interactions within the network. The results are widely applicable to very different domains: biology, information and communication technologies and social systems. They can also be useful for navigation and searching protocols which take into account both the existence and the intensity of a connection. The evidence examples are taken from empirical data sets.

Software development process is considered as a socio-technical system with four types of objects [36]: *agents* are team members and developers, *knowledge* represent skills possessed by the team members or required for task solving, *tasks* are problems to be solved, and *resources* relates equipment used by agents and required to complete tasks. A network of a socio-technical system represents actual communications and dependencies in software development process (task dependency on other tasks, resources and skills; developer connections with other agents, their skills and equipment they can use). Socio-technical congruence measures interactions of agents, but indirectly, which could be used for proposing weights of agents' interactions. For example, by considering the following relations: agent-task (AT), knowledge-task (KT), and agent-knowledge (AK). Particular relations within the network are presented as adjacency matrices of subgraphs containing specific partitions of objects denoted by  $AT$ ,  $KT$  and  $AK$ , respectively. Further, knowledge-dependent congruence requirement matrix  $CR$  is determined,  $CR = AT \times (KT)^T \times (AK)^T$ . Socio technical congruence indicates how the obtained matrix differs from adjacency matrix  $AA$ ,

$$Congruence(CR, AA) = 1 - \frac{diff(CR, AA)}{|CR|}$$

where  $diff(CR, AA)$  and  $|CR|$  are numbers of nonzero elements in the matrices  $CR - AA$  and  $CR$ , respectively. Data set for the experiment contains data from version control system for tracking changes in project realization, survey data of team members knowledge and skills, and data from students' discussion forum. Observation of socio-technical congruences on the time base (during some weeks) has strong positive impact on the software development process.

Mavromoustakis and Karatza [24] used weighted graphs for modeling resource sharing in the opportunistic wireless environment by considering social parameters. Communication was presented by two socio-technical layers: *Social Connectivity Layer* (SCL) represents social interactions of users that use a certain platform, and *Physical Connectivity Reflection Layer* (PCRL) represents physical connectivity in wired or wireless infrastructure. Resource location is determined with social associations and interactions among individuals. Weighted undirected connected graph is used for modeling random movement of devices. The weights are determined by the intensity of social interaction between two users (nodes in network). Probability of movement from one node to the next node (one of the neighboring nodes) is calculated by using the weights of all edges associated to the current node. Since mobility is unrestricted in a wireless opportunistic network, the network topology dynamically change over time. Proposed resource sharing schema was evaluated through experimental simulation, which revealed that the reliability of successful packet delivery ratio is not aggravated by the mobility factor.

Goggins et al. [37] presented an analytic framework, based on Group Informatics Model [38], which improves awareness of virtual group dynamics by analyzing electronic trace of individuals' interactions and enables identification of emergent groups. The framework is based on the following items, which can be customized for specific cases: *domain of study* is a description of an application domain; *contextualized interactions* represents the nature of the collected interaction traces with their contextual attributes; *weighting procedure* is determination of weighting that is applied to the contextual attributes; *collaboration opportunities network* represents the nature of the network resulting from weighting for the selected domain; *aggregation procedure* represents

the aggregation algorithms and parameters used for making groups visible; *emergent groups* are the group structures emerging from the aggregation; and *group context* is description of contextual information associated to each identified group. Nodes in the network represent people and artifacts, while weights represent intensity of interactions between the nodes. The framework is applied in two domains: (1) asynchronous online learning for discourse around ideas in discussion forums, and (2) open source software development for software developer interactions with knowledge-intensive, technical artifacts of the software product.

Ferrara et al. [39] used a directed weighted graph, named *Relational Instagram Network* (RIN) for modeling and analyzing social interactions at Instagram, focusing on follower-followee relations and user communication by means of posts/comments. The analysis focuses on the structural characteristics of the Instagram network, the dynamics of production and consumption of content, and the social tagging for presenting users' interests in the content. An Instagram data set was collected by querying the Instagram API (e.g. features such as the users API, the relationships API, the media API, the comments and the likes APIs, and the tags API), which is publicly available developer library. Nodes in RIN present users, edges present relationships of the form follower-followee, while edge weights were calculated proportionally to the number of likes and comments.

Islam et al. [40] proposed a new heuristic algorithm based on underlying dependency structure in procedural programs for migrating software design using hierarchical clustering. The proposed approach produces candidate classes for an object-oriented design based on given procedural code. A procedural program is presented by a new type of graph, *Weighted Data Call Graph* (WDCG), in which a similarity measure called *Weighted Distance Matrix* (WDM) is used for computing similarity between two nodes in WDCG. Weights are presented with data structure *Entity Map*, in which all data and function nodes have calculated weights for each relationship in the graph. WDCG contains two types of nodes: *Function Nodes* for representing functions, and *Data Nodes* for representing data. WDCG contains four types of edges: *Self-Edge* connects a node to itself, *Call Edge* connects a function that calls another function, *Read Edge* connects a function that reads data, and *Write Edge* connects a function that writes data. The weight of an edge in WDCG depends on the relationship represented by the edge. The weight of a self-edge is always 1, while weights of call edge, read edge and write edge are from a predefined set of weights. Migration approach contains the following steps: generation of a WDCG from procedural code, hierarchical cluster analysis of the WDCG, identification of desired clusters based on migration objectives, and transformation of clusters to classes. The proposed algorithm was tested on a 5 procedural programs written in C language.

Traffic regulation system based on measured air quality data is presented by El Fazziki et al. [28]. The system uses big data and intelligent systems concepts for traffic regulation based on real-time and predicted air pollution indexes. The system helps in reducing vehicle emissions in the most polluted sections of the road. Data collected from sensors, contextual data, and road network available data are used for constructing a weighted graph that presents a road network in which intersections characteristics are associated with the nodes and roads characteristics are associated with the edges. Edges weights evolve based on the pollution indexes. The system is tested in Marrakech city with real-time data.

Kalloniatis et al. [30] proposed empirically derived *Situation Awareness Weighted Network* (SAWN) model, which is based on *Situation Awareness model* proposed by Endsley [41] involving perception, comprehension and projection, and Distributed Situation Awareness model proposed by Stanton et al. [42] indicating that situation awareness exists in a social and semantic network of people and information objects. The SAWN model represents a target socio-technical system as a weighted semi-bipartite network of interactions between people (human nodes) and information artefacts (product nodes). Links in the network are relations between people and between people and artefacts. The weights of the links in the network represent the levels



of social awareness that individuals acquire, which is rendered in the network by three levels proposed by Endsley through the color scheme: Green for Perception, Blue for Comprehension, and Red for Projection. The SAWN method is illustrated with empirical data from a case study related to work of Australian military staff in steady-state and crisis threat scenarios.

Binzagr and Medjahed [43] proposed a crowdsourcing approach for recommending mashup teams, which is based on analysis of online developer community and API directories at StackOverflow. The approach uses three data structures based on data from StackOverflow: interest table (developers interests in specific APIs), reputation table (developers reputation based on comments and replies) and sociometric graph (connections among developers). Interactions between developers are comments and replies, which are modeled as a weighted graph (sociometric graph). Developers are nodes in the graph, interactions represent the edges between the developers, while the number of interactions between the developers represents the weights of the edges. The authors implemented a CrowdMashup prototype software in Java, in which mashup administrators define queries for forming a team by defining the number of teams, the number of members in each team and a list of required APIs. The inputs for software are the sociometric graph, the interests table and the reputation table. The implementation software is evaluated in an experiment with real data from StackOverflow and programmableWeb.

Pattanaik et al. [44] presented a smart technique for congestion avoidance by estimating real-time traffic congestion on urban road networks, and based on that the technique predicts an alternate shortest route to the destination. The congestion on different roads is estimated by K-Means Clustering Algorithm, while Dijkstra's Algorithm is used for predicting the shortest route. The system identifies a road network from Google Maps and convert it to a weighted graph. Nodes represent intersections, while edges represent roads. Initial weights on all roads are set to 1, but based on real-time data on the number of vehicles on a particular road, the weights are updated periodically. Updated weights are used for estimating the travel time and for calculating the shortest path to the destination based on real-time congestion of roads in the network. The approach was tested over several road maps of New Delhi and simulation results indicate reduction in travel time if road congestion is used for updating the weights of the edges in the network (the roads in the city).

Cepulis and Niu [32] conducted a software requirements traceability case study aimed at creating socio-technical patches for information foraging. Software developers usually conduct a search task based on a predefined patch of information, which in requirements traceability relates to traversing a combination of technical artifacts and social structures. The authors proposed requirements socio-technical graph for analyzing human-human and human-artifact relationships that connect a traceability question to an answer. The graphs contains three node types: people, artefacts and questions/answers. The weights of the relations in the graph depend on the knowledge that possesses a person providing an answer to the posed question. The method is tested on four open source projects from the Apache software foundation and the JBoss.

Ikram et al. [45] presented an approach supported with DaDiDroid, an Android malware app detection tool for examining graph features of mobile apps, by building weighted directed graphs of the API calls. The approach detects the presence of malware code in Android apps by analyzing graph features. DaDiDroid performs static analysis of apps and extracts call graph among APIs. Nodes in the graph present API libraries, a directed link present call of a method from another API, and weights represent the numbers of times a methods from other APIs are called. Effectiveness of DaDiDroid is examined in an experiment with six publicly available data sets.

## 4. Discussions

This research on the use of weighted network in socio-technical systems revealed several questions and challenges related to network creation, types and nature of nodes, edges and weights, network evolution and multidisciplinary nature of research of socio-technical systems that need discussion.

### 4.1. Comparative analysis of presented examples

Due to the complexity and diversity of real socio-technical systems, they can be presented with homogeneous and heterogeneous networks [46]. In homogeneous network, all elements in the set of vertices are uniform, while in heterogeneous network a clustering or hierarchy exist in the set of vertices or in the set of edges.

In homogeneous networks, nodes are commonly of the same type and have the same function, such as intersections in road traffic networks [28, 44], airports in airline connection network and scientists in scientific collaboration networks [34], API libraries in software architecture network [45], or people in Instagram social network [39]. In heterogeneous networks, nodes and links can be of variety types depending on the domain modeled with the network. In many cases, vertices can present entities of different types in the same network, such as persons, resources and knowledge [32, 36], data and functions in software migration project [40], or humans and information product artefacts [30].

Relations that are represented by the edges (links, relations) can be given explicitly or implicitly, and they present actual or virtual links among vertices (nodes). Links can connect the nodes of the same type in the network, or can connect nodes of different types, depending on the network nature (the nature of domain problem analyzed by the network). The common case are links representing social interactions between people in network [24, 39, 43]. In considering social aspects of socio-technical systems, it could be necessary to additionally examine people or their communications and collaborations for more accurate modeling of links in created networks [34, 37]. In complex socio-technical systems with different types of nodes, links can also represent interactions among people and artefacts [32, 37]. However, in some cases links connect only technical artefacts, such as links representing roads or their characteristics [28, 44], links between functions and data in software systems [40], or calls between different API libraries in mobile software systems [45].

A weight function of a network models differentiation among the edges. They can be obtained by using statistical observations and analysis [34, 35, 39, 43, 45], by measuring real-time data [28, 44], by analyzing and quantifying triangulated data from the interaction context [37], by examining knowledge level of people in the network [32], or by using models such Situation Awareness model proposed by Endsley [41] that is used in [30]. In addition, weights can be predefined [40].

### 4.2. Challenges

The first challenge relates to creation of a network for a particular system. The creation of a network as mathematical model for real phenomena imposes exact and distinct determination of objects which would be represented as vertices. Statistical tools produce distribution of vertex degrees and distribution of edge weights [34, 43], which are particularly important in large networks. Further, creation of weighted networks for analyzing soci-technical networking systems can be done by using publicly available developer libraries such as Instagram API [39], or by analyzing existing video materials such as Google Maps for extracting urban road networks [44]. In software development, systems networks can be created by extracting API libraries and calls from source codes of applications [45], by analyzing traceability of questions and answers in specifying software requirements [32], or by analyzing connections among developers at software development forums such as StackOverflow [43].

As socio-technical systems are evolving systems it is of interest to embed dynamics into the weighting network structure or into the network topology. Parameters and characteristics of artifacts in systems can change over time, and this should be modeled in weighted graphs with weights that can change [28], or with topology that can change [24]. For example, dynamical evolution occurs when new vertexes and edges are introduced in the existing system [34], when network topology change is natural like in wireless opportunistic network [24], or when weights are periodically updated based on real-time data measured in the real system [44]. According to Rouse and Serban [47] it is essential to understand how causality and complexity influence the nature of change in complex socio-technical systems. Management of network evolution can be used for long-term analysis and control of socio-technical systems, as well as for prediction of their further behaviour and states.

This review of examples and socio-technical systems development reveals that weighted networks can be used in many domains of human life and work. Due to the complex nature of real socio-technical systems, systems' analysis and creation of models require forming multidisciplinary teams with experts from technical, natural and social disciplines, who can observe all relevant aspects of the system [5, 48], and that will have deep knowledge on the context to be modeled as a socio-technical system [38]. Further, it will require the use and mixing of multiple quantitative and qualitative research methods [49] for acquiring a more comprehensive understanding of socio-technical systems.

## 5. Conclusions

Presented concepts, examples and challenges related to socio-technical systems reveal their importance for understanding, modeling and analyzing complex behavior in real systems with humans and technology. The contribution of this paper is a comprehensive analysis of the global aspects of weighted networks in highly diverse socio-technical systems, as well as the discussion of the challenges in tackling with systems' design and dynamics.

Systematic literature review is planned as one of the next directions for further work. This systematic review will provide more comprehensive review of challenges in the selected areas, such as urban traffic or software development, and identify issues for further research.

## Acknowledgment

Ministry for scientific and technological development, higher education and information society Republic of Srpska supports this research under the project "Smart system based on IoT technology designed for monitoring of traffic air pollution", contract number 19.030/3-2-25-2/19.

## References

- [1] Trist E 1981 *Occasional paper No. 2, Ontario Ministry of Labour* ISSN: 0227-4426
- [2] Walker G H, Stanton N A, Salmon P M and Jenkins D P 2008 *Theoretical Issues in Ergonomics Science* **9** 479–499
- [3] Ropohl G 1999 *Techné: Research in Philosophy and Technology* **4** 186–194
- [4] Fox W M 1995 *The Journal of Applied Behavioral Science* **31** 91–105
- [5] Fuenfschilling L and Truffer B 2014 *Research Policy* **43** 772–791
- [6] Sommerville I 2011 *Software Engineering* 9th ed (Boston, MA, USA: Addison Wesley)
- [7] Hu F, Mostashari A and Xie J (eds) 2011 *Socio-Technical Networks: Science and Engineering Design* (Boca Raton, FL, USA: CRC Press, Taylor & Francis Group)
- [8] Trist E L and Bamforth K W 1951 *Human Relations* **4** 3–38
- [9] Pasmore W, Winby S, Mohrman S A and Vanasse R 2019 *Journal of Change Management* **19** 67–85
- [10] Cherns A 1976 *Human Relations* **29** 783–792
- [11] Hirschhorn L, Noble P and Rankin T 2001 *Journal of Engineering and Technology Management* **18** 241–252 ISSN 0923-4748
- [12] Kopackova H and Libalova P 2017 *2017 International Conference on Information and Digital Technologies (IDT)* (Zilina, Slovakia) pp 198–205

- [13] Shin D H and Jin Park Y 2017 *Digital Policy, Regulation and Governance* **19** 77–100
- [14] Jotanovic G, Brtka V, Curguz Z, Stojcic M and Eremija M 2018 (Bitola, North Macedonia) pp 94–98
- [15] Stojanov J, Jausevac G, Stojanov Z and Hristoski I 2019 *VII International Symposium NEW HORIZONS of Transport and Communications* (Doboj, Bosnia and Hercegovina) pp 535–541
- [16] Chu K H, Suthers D D and Rosen D 2012 *2012 45th Hawaii International Conference on System Sciences* (Maui, HI, USA) pp 3560–3569
- [17] Land S M and Zimmerman H T 2015 *Educational Technology Research and Development* **63** 229–255
- [18] Fraternali P, Cellina F, Herrera S, Krinidis S, Pasini C, Rizzoli A E, Rottondi C and Tzovaras D 2018 *2018 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)* (Athens, Greece) pp 59–64
- [19] Baki S, Rozos E and Makropoulos C 2018 *Science of The Total Environment* **622-623** 1590–1602
- [20] Britto A L, Maiello A and Quintslr S 2019 *Journal of Hydrology* **573** 1007–1020
- [21] Balka E, Whitehouse S, Coates S T and Andrusiek D 2012 *Information Systems Frontiers* **14** 1942
- [22] Bardram J E and Frost M M 2018 *Designing Healthcare That Works* ed Ackerman M S, Goggins S P, Herrmann T, Prilla M and Stary C (Academic Press) chap 10, pp 167–186 ISBN 978-0-12-812583-0
- [23] Suthers D 2017 *50th Hawaii International Conference on System Sciences* (Hawaii, USA) pp 2128–2137
- [24] Mavromoustakis C X and Karatza H D 2013 *The Computer Journal* **56** 184–197
- [25] Walton R and De Roure D 2019 *Proceedings of the 10th ACM Conference on Web Science WebSci '19* (Boston, MA, USA) pp 301–310 ISBN 9781450362023
- [26] Zafar N A 2016 *Complex Adaptive Systems Modeling* **4** 1–26
- [27] Vodopivec N and Miller-Hooks E 2019 *Reliability Engineering & System Safety* **191** 106561
- [28] El Fazziki A, Benslimane D, Sadiq A, Ouarzazi J and Sadgal M 2017 *IEEE Access* **5** 13192–13201
- [29] Zhang Q, Jin Q, Chang J, Xiang S and Pan C 2018 *2018 24th International Conference on Pattern Recognition (ICPR)* (Beijing, China) pp 1018–1023 ISSN 1051-4651
- [30] Kalloniatis A, Ali I, Neville T, La P, Macleod I, Zuparic M and Kohn E 2017 *Applied Ergonomics* **61** 178–196
- [31] Masci P, Curzon P, Furniss D and Blandford A 2015 *Innovations in Systems and Software Engineering* **11** 113130
- [32] Cepulis D and Niu N 2018 *2018 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)* (Lisbon, Portugal) pp 17–21
- [33] Gross J L and Yellen J (eds) 2003 *Handbook of graph theory* Discrete mathematics and its applications (Boca Raton, FL, USA: CRC Press, Taylor & Francis Group)
- [34] Barthélemy M, Barrat A, Pastor-Satorras R and Vespignani A 2005 *Physica A: Statistical Mechanics and its Applications* **346** 34–43
- [35] Allard A, Ángeles Serrano M, García-Pérez G and Boguná M 2017 *Nature Communications* **8**
- [36] Jiang L, Carley K M and Eberlein A 2012 *2012 International Conference on Software and System Process (ICSSP 2012)* (Zurich, Switzerland) pp 160–169
- [37] Goggins S P, Valetto G, Mascaro C and Blincoe K 2013 *User Modeling and User-Adapted Interaction* **23** 345379
- [38] Goggins S P, Mascaro C and Valetto G 2013 *Journal of the American Society for Information Science and Technology* **64** 516–539
- [39] Ferrara E, Interdonato R and Tagarelli A 2014 *Proceedings of the 25th ACM Conference on Hypertext and Social Media HT 14* (Santiago, Chile) pp 24–34
- [40] Islam M, Toma T R, Selim M, Gias A U and Khaled S M 2016 *International Journal of Information Engineering and Electronic Business(IJIEEB)* **8** 1–13
- [41] Endsley M R 1995 *Human Factors* **37** 32–64
- [42] Stanton N A, Stewart R, Harris D, Houghton R J, Baber C, McMaster R, Salmon P, Hoyle G, Walker G, Young M S, Linsell M, Dymott R and Green D 2006 *Ergonomics* **49** 1288–1311
- [43] Binzagr F and Medjahed B 2018 *International Conference on Service-Oriented Computing (ICSOC 2018)* (*Lecture Notes in Computer Science* vol 11236) pp 679–693
- [44] Pattanaik V, Singh M, Gupta P K and Singh S K 2016 *2016 IEEE Region 10 Conference (TENCON)* (Singapore, Singapore) pp 3420–3423
- [45] Ikram M, Beaume P and Kaafar M 2019 *Proceedings of the 16th International Joint Conference on e-Business and Telecommunications (ICETE 2019)* vol 2 (SECURITY) (Prague, Czech Republic) pp 211–219
- [46] Liu Z, Lai Y C and Ye N 2003 *Physical Review E* **67**(3) 031911
- [47] Rouse W B and Serban N 2011 *Information Knowledge Systems Management* **10** 25–49
- [48] Baxter G and Sommerville I 2011 *Interacting with Computers* **23** 4–17
- [49] Creswell J W 2009 *Research design: Qualitative, quantitative, and mixed methods approaches* 3rd ed (Thousand Oaks, CA, USA: SAGE Publications)