

A Triangular NodeTrix Visualization Interface for Overlapping Social Community Structures of Cyber-Physical-Social Systems in Smart Factories

Chun-Cheng Lin, *Member, IEEE*, Der-Jiunn Deng*, *Member, IEEE*, and Shun-Yu Jhong

Abstract—This work proposes a cyber-physical-social system (CPSS) framework for smart factories which associates the social-network-based physical factory circumference with the cyber control through a visualization interface based on NodeTrix. The NodeTrix representation consisting of adjacency matrices and node-link diagrams has been popularly used in social network analysis. Different from previous works on NodeTrix based on square-shaped matrices, the proposed NodeTrix applies triangle-shaped matrices for visualizing overlapping communities to save the displaying space, and a simulated annealing algorithm is used to reduce the crossings among links and communities. In the CPSS for smart factories, working-in-process (WIP) products are categorized according to their attributes, and are placed at different storage locations for later different production processes. By the NodeTrix-based interface for CPSS, when users with different domains and experiences work together to make instant decisions on production and logistics, the physical production environment will be adjusted accordingly and immediately through technological support of Industry 4.0. Through the interface, the information on locations, parameters, and interoperability of WIP products can be visualized and adjusted in time. By comparison with other works, the proposed representation looks promising in displaying social networks for the CPSS with fewer crossings within a smaller screen space.

Index Terms—*Social network analysis, visual analysis, information visualization, smart factory, interface.*

I. INTRODUCTION

Cyber physical system (CPS) [1], [2] tightly involves cyber and physical factors in distributed computing or grids environments to provide real-time services. It has been applied widely in different industries, e.g., transportation, avionics,

energy, healthcare, manufacturing processes, etc. And, it is usually equipped with novel technologies of wireless networks, e.g., determining the aggregation locations of APs with minimal delay of disseminating the data sensed by mobile CPSs [3]; providing real-time health monitoring and ubiquitous healthcare services with interference mitigations by the CPS based on wireless body area networks using social interaction information [4]; and investigating the quality of service, network parameter configuration, and optimization of wireless sensor networks in CPS applications [5].

Aside from the cyber space (such as CPU, networks, storage systems, etc.) and the physical space (such as location, migration, etc.), the CPS also involves the socio space and mental space for precise analysis and useful services. Recent research has tended to focus on integrating social factors into the CPS, called cyber physical social system (CPSS) [6]. This work considers the CPSS in smart factories as shown in Fig. 1, in which the physical space involves warehouses and logistics; the cyber space detects the attributes and quantity of working-in-process (WIP) products through the Internet of things (IoT) or IEEE 802.15.4 [5]; and the additional social space considers social-networking information and interoperability of the system to meet practical requirements. Through integration of these facilities and systems, the CPSS has become one of the key technologies in the Industry 4.0. However, the conventional manufacturing lacks an effective and real-time communication bridge among facilities and human. Hence, the CPSS including wireless networks and the concept of social networks can promote communications among facilities and human in physical systems, so that facilities are smarter and meet the requirements in practice.

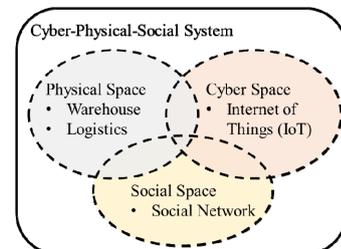


Fig. 1. Illustration of resources interaction in the CPSS for smart factories.

The previous works on CPSS can be classified into two categories: systems and applications. From the system aspect, the self-synchronization of CPSS [7], [8] was proposed to direct and control the linkages of physical, cyber, and social

This work was supported in part by MOST 105-2628-E-009-002-MY3, Taiwan. (Corresponding author: Der-Jiunn Deng.)

C.-C. Lin and S.-Y. Jhong are with Department of Industrial Engineering and Management, National Chiao Tung University, Hsinchu 300, Taiwan. E-mails: cclin321@nctu.edu.tw, jsyu0527@gmail.com.

D.-J. Deng is with Department of Computer Science and Information Engineering, National Changhua University of Education, Changhua 500, Taiwan. E-mail: djdeng@cc.ncue.edu.tw.

*D.-J. Deng is the corresponding author of this paper.

spaces. A CPSS consisting of Lego Mindstorms robots [9] was proposed to finish the task of picking-and-placing an object from production to warehouse. In addition, the system safety of CPSS [10] is crucial, for future security and privacy protection for cyber, physical, and human social worlds. From the application aspect, there have existed lots of CPSS applications, e.g., multi-level self-organization of CPSS for smart home cleaning [11], integration of the CPSS big data on cloud [12], geo-friends recommendation by GPS-based CPSS [13], a data-centric framework of CPSS for an urban big data system in a city [14], and some future CPSS applications [15].

This work proposes a CPSS for smart factories with a visualization interface. Considering a smart factory, when a WIP product is delivered to the CPSS, it will be categorized according to its attributes, and its attributes will be stored to the database. The WIP products in the same category are collected and placed at the same storage location, say the same cluster. However, it may be hard to categorize each WIP product into an individual cluster, i.e., some WIP products may be categorized into at least one cluster. Hence, the relations of these clusters of WIP products can be represented as a clustered graph with overlapping clusters. Furthermore, these relations are associated with multiple attributes of products and the staff of the factory (e.g., production managers, sales managers, manufacturing engineers, and R&D engineers). Therefore, these relations constitute a social network with overlapping community structures.

The cyber space in the CPSS for smart factories provides an interface for visualizing the network structures and interacting with their relations. Through the interface, users can adjust parameters to control the similarity of attributes among WIP products to determine the complexity degree of the structure visualization. Because some domain knowledge and experiences on the whole production can only be learned at the work field over years, junior and senior staff can work and learn together with social interactions through the interfaces on their respective terminals or mobile devices. After the decision of each actor in this social network for the production is made, classification or manufacturing processes of WIP products may be manipulated, and the corresponding physical logistics operations will be adjusted accordingly immediately.

The graph structure behind the social network for the CPSS represents the relationship among actors, e.g., interpersonal relations, business partnerships, and international relations. With recent popularity of social media services such as Facebook, Google+, and Twitter, social network analysis (SNA) have received much attention. Conventionally, sociologists have applied node-link diagrams to visualize social networks with social community structures (e.g., [16], [17]), in which each actor is drawn as a node; each relation between actors is drawn as a link; the nodes in each community consisting of the actors with strong relations are drawn closer, and the nodes from different communities are drawn apart. Recently, the related research on visualizing social networks has turned to focus on overlapping community structures of social networks, in which some actors may

belong to at least one community so that their communities are overlapped.

One of the most notable works for visualizing overlapping community structures of social networks is the NodeTrix representation [18] (Fig. 2(a)), in which the subgraph of each community is represented as a square-shaped adjacency matrix (which shows the adjacency relations of the nodes within the community); if a node appears in η communities, it has η duplicate labels respectively appearing in η matrices, and the η labels are connected by a gray curve band (a.k.a., an overlapping-node link), whose thickness reflects the number of nodes overlapped between two communities; the links between two communities (a.k.a., inter-community links) are connected by blue curves.

The visualization interface for the social network for the CPSS is based on NodeTrix. However, the conventional NodeTrix has the following flaws. Firstly, the information on adjacency relations of communities is so redundant to waste the displaying space, because the adjacency matrix is symmetric along its diagonal and has two triangle-shaped duplicates in visualization. Secondly, when representing the networks with a large number of overlapping nodes, lots of crossings among gray curve bands may be caused, so that they are hard to be visualized through the conventional NodeTrix. Lastly, the inter-community links and the overlapping-node links may not be easily identified and distinguished because they could be connected to the same connecting ports.

In light of the above, this work proposes the following improvements for the NodeTrix. Firstly, the square-shaped adjacency matrix is improved to be replaced by a triangle-shaped adjacency matrix to save the displaying space and increase the readability of the visualization. Secondly, to decrease the complexity of visualizing the information, the number of crossings among links and communities is minimized by a novel simulated annealing (SA) algorithm. Lastly, overlapping-node links are connected with diagonal sides of matrix triangles, and inter-community links are connected with vertical and horizontal sides of matrix triangles, so that they can be distinguished easily. To evaluate performance of the proposed improved NodeTrix, the simulations on three important tasks in SNA [19] using the proposed and previous methods are compared.

The rest of this work is organized as follows. Section II gives the literature review on related works of visualizing social networks with overlapping community structures. Section III proposes the methodology of the visualization interface. Section IV gives the simulation results and comparison. Section V concludes this work with future work.

II. RELATED WORK

This section first reviews the related works on CPSS, then the works on visualizing overlapping community structures, and finally the works on NodeTrix representations, upon which this work is based.

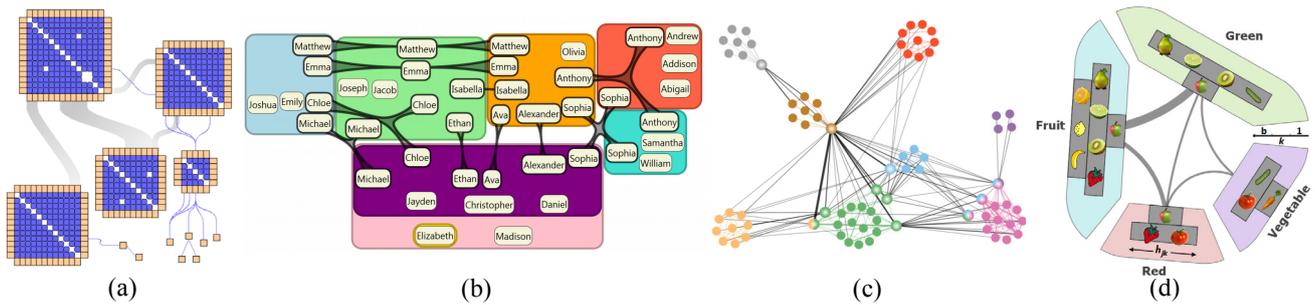


Fig. 2. Four styles of visualizing overlapping community structures: (a) NodeTrix with square-shaped adjacency matrices [18], (b) Euler diagram [20], (b) node-link diagrams [21], and (d) radial sets [22].

A. Related works on CPSS

The previous works on CPSS are mainly discussed from the aspects of systems, applications, and security architectures.

Firstly, consider the system aspect of the CPSS. To achieve self-synchronization of a command and control organization, the work in [5] established a chaotic control mechanism for the CPSS. Taking events or tasks of an organization as the input, the CPSS can automatically integrate the essential components of an organization in the four domains: 1) assigning physical resources, 2) setting up sensor networks and enabler networks, 3) constructing the command and controlling relationships in the social network, and 4) organizing and sharing relevant information as needed. Finally, the output is the entire command and control organization as a CPSS.

To enhance the ability of self-organization in CPSS, the work in [8] integrated various resources from physical, cyber, and social spaces. To achieve efficient interaction of these resources (including physical devices and humans), the authors proposed an upper ontology for CPSSs and its application for self-organization of CPSS resources, because the upper ontology has a promising share ability in the application area. Note that humans sometimes play the role in providing information, knowledge, services, etc.; and they are also the users of the CPSS in consuming information, knowledge, services, etc.

The work [9] developed a multi-level approach for self-organization, and adopted the Lego Mindstorms robots development kit to finish the task of picking-and-placing an object from production to warehouse with the following two phases. Firstly, the pipeline robot moves and scans the object, and shares the scanned information of this object with the smart space. Secondly, there are two manipulating robots: the first one has an object protection against precipitation and moves slowly; and the other one has no object protection but moves quickly. They will be self-organized according to the available information, and then one of them is decided to move the object to the warehouse.

The work in [23] investigated the reliability of the information collected by humans in evolving CPS. The authors developed a quantitative approach for assessing the reliability of correctness of collected data.

Next, consider the application aspect of the CPSS. The work

in [11] investigated optimizing the CPSS operated in the smart home cleaning scenario. The authors developed a three-level self-organization approach for the system resource interaction in CPSS: physical, planning, and strategic levels. In the physical level, a vacuum cleaner, manipulating, and light control services are involved. Then, the ordering of cleaning tasks and the cleaning with minimizing power consumption are decided in the planning and strategic levels, respectively.

The work [13] developed the CPSS for the geo-friends recommendation in GPS, as the popularization of GPS-enabled mobile devices constitutes a social network. In their proposed method, a pattern-based heterogeneous information network is established based on the location and trajectory data available, in which each link represents the association with two people's geographical information and social relationship. Then, traditional link prediction methods are used to recommend friends.

The work in [15] discussed some future trends on CPSS, in which they thought that market feasibility of CPSS can attract government agencies and private companies to exterminate corruption and to optimize interaction of time, human and financial resources following laws and regulations.

The work in [14] investigated a CPSS for urban big data, which derives and integrates information from cyber space (e.g., governments and institutions), physical space (e.g., surveillance cameras and smartphones), and social space (e.g., mobile crowd sensing and mobile social networks). This CPSS is data-centric, and its performance is improved by multispace collaborative sensing and cross-space data fusion.

The work in [12] proposed an integration framework on cloud, with an application in smart home. The authors further proposed a multi-linear data rank approach and an incremental rank update method in this framework.

Lastly, consider the security architecture aspect of the CPSS. Based on the U2IoT model, the work in [10] proposed a CPSS based security architecture to handle information, physical, and management security perspectives, and showed how the U2IoT model is supported by this architecture. Their proposed architecture combines the CPSS with future security and privacy protection.

B. Visualizing overlapping community structures

This subsection reviews four notable styles of visualizing

overlapping community structures in social networks (see Fig. 2). The first style is based on NodeTriX [18] (Fig. 2(a)). In the NodeTriX representation, the subgraph of each community of nodes is represented as a square-shaped adjacency matrix, in which node labels are the indices of rows and columns; and the entry grid at the i -th row and j -th column of the matrix is shaded (in blue in Fig. 2(a)) if nodes i and j are linked; otherwise, it is blank (in white in Fig. 2(a)). If some node is an overlapping node in η communities, it has η duplicate labels in the matrices of the corresponding communities, and are connected by gray curve bands. In addition to the gray curve bands, the links between communities are represented by solid line segments.

The second style is based on Euler diagrams [20] (Fig. 2(b)). Euler [24] in 1772 proposed the basic concept on this diagram, in which each node is represented as a simple circle attached with its label; each community is represented as a circle enclosing its member nodes, and hence each overlapping node can be observed by whether it is covered by at least one community circle. Henry et al. [20] proposed a visualization method based on Euler diagrams for overlapping community structures (Fig. 2(b)), in which each overlapping node is represented by the duplicate labels connected with each other in different community circles. By Euler diagrams, the relations inside each community can be visualized, but the relations between communities cannot.

The third style is based on node-link diagrams [25]. Different from conventional node-link diagrams, the work in [26] proposed anchored maps, in which an anchor is added to identify overlapping communities. Hence, if some node is linked with two anchors, it belongs to two communities. Fig. 2(c) shows another application of node diagrams [21], in which the nodes in the same community are represented by similar disk shapes, and the size of disk can be adjusted to identify its degree of tightness with all communities. However, when the size of a community is larger, it would be difficult to identify the relationship between nodes in this community because too many crossings among links could be caused.

The fourth style is based on radial sets [22] (Fig. 2(d)), in which each community is a set placed along the borderline of a thick ring shape; the node with more overlapping communities is placed closer to the center of the ring; and gray curves with thickness are connected among these overlapping nodes inside the ring. Although the information of overlapping communities can be visualized by this technique, it would be difficult to identify the relationship of nodes within communities.

C. Related works on NodeTriX

Henry et al. proposed a series of methods for NodeTriX for visualizing social networks [18], [19], [27].

They first proposed the MatrixExplorer [27], which alternatively applies node-link diagrams and adjacency matrices to display social networks. The idea behind the MatrixExplorer is based on the survey results from some sociologists, who made 13 recommendations. Aside from the

tools of basic interactions, they proposed an algorithm to adjust the layouts inside matrices, many layout results to be analyzed, and some auxiliary tools to visualize the network from macro-view and micro-view. In addition, users can explore the relationship of communities and outliers through these tools.

Later, the authors proposed the NodeTriX [19] for visualizing large-scale social networks, which combines two conventional visualization techniques: the node-link diagram is used for displaying the network structure, and the adjacency matrix is used for displaying the connectivity of nodes within a community. By the NodeTriX, the whole network structure and the details of each community can be visualized concurrently. The NodeTriX provides an interface that allows users to manipulate this visualization. The authors used the NodeTriX to visualize the data of coauthorship of InfoVis 2004, and analyzed the performance of NodeTriX when executing three important tasks in SNA: finding communities, finding core actors, and analyzing actor positions and outlier actors.

The authors further proposed an improved NodeTriX representation, called Duplication [18], for overlapping communities of social networks, in which a duplicate label of each overlapping node is placed to each of the communities with this overlapping node; and colors and thickness of lines are used to identify some attributes of this network. The visualization interface proposed in this work is to improve the Duplication technique [22].

III. OUR NODETRIX REPRESENTATION

This section gives details of the proposed NodeTriX representation for overlapping communities of social networks. The proposed method for drawing overlapping communities is based on the NodeTriX representation for overlapping communities [18], which transforms the subgraph of each community (Fig. 3(a1)) into a square-shaped adjacency matrix (Fig. 3(a2)), in which the entry at the i -th row and j -th column of the matrix is shaded if nodes i and j are linked (e.g., colored in red in Fig. 3(a2)); otherwise, it is a blank (e.g., colored in gray in Fig. 3(a2)).

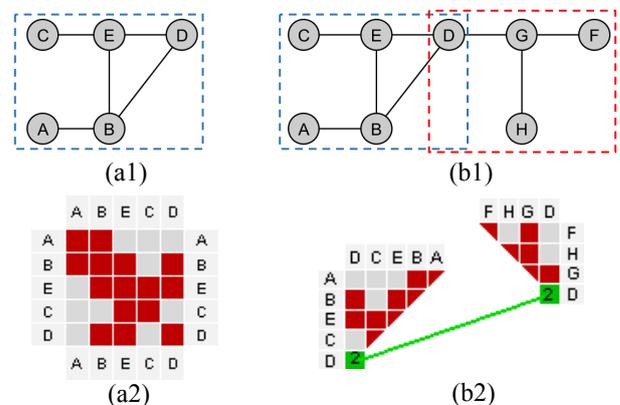


Fig. 3. Transforming node-link diagrams into adjacency matrices. (a) Conventional square-shaped adjacency matrix for a community. (b) The

proposed triangular-shaped adjacency matrix for two communities with an overlapping node labeled by 2.

However, we observe that a square-shaped matrix consists of two symmetrical triangular shapes with the same information along the matrix diagonal. To save the drawing area, the proposed method transforms each community (Fig. 3(b1)) into a triangular-shape adjacency matrix (Fig. 3(b2)), in which the adjacency information of nodes can still be observed.

For both overlapping-node links and inter-community links, the original NodeTrix representation applies thin curves and thick curve bands, respectively, to connect the labels on the four sides of the square matrix. That is, both the two link relations in the original NodeTrix are associated with the node labels. To make a difference on the two link relations, our proposed method connects each overlapping-node link with all of its entry grids on diagonal sides of matrix triangles of the corresponding communities, and the number of overlapping communities for this overlapping node is displayed on the entry grid (e.g., ‘2’ in Fig. 3(b2)); whereas the inter-community links are represented as the line segments between the labels on the horizontal and vertical sides of the matrix triangle. By doing so, the two link relations can be distinguished more easily.

Given a social network with overlapping community structures, consider that each node belongs to at least one community. The proposed algorithm of transforming this network into a triangle-shape NodeTrix representation is detailed in Algorithm 1. Details on each step in this algorithm are given in the rest of this section. In addition, two visualization modes are provided for user interface.

Algorithm 1 Triangular_NodeTrix()

Input: A social network with overlapping community structures.

Output: A triangle-shpaed NodeTrix representation.

- Step 1: Generate a nice node-link diagram layout of the community structure of the network topology using a force-directed graph drawing method.
- Step 2: Transform the node-link diagram into a triangle-shaped NodeTrix representation.
- Step 3: Use an SA algorithm to minimize the number of crossings among links.
- Step 4: The representation is adjusted according to user feedbacks based on their requirements.
- Step 5: Determine the allocation of overlapping-node links.
-

A. Generating a node-link diagram layout

Given an overlapping community structure of the network, this work adopts the force-directed graph drawing method (called organic layout) provided by the yEd graph editor [28] to produce a node-link diagram layout of this network, i.e., the geometrical positions of all nodes are determined, and each link is drawn as a straight line segment. Given an overlapping community structure of the network (i.e., a clustered graph without geometries), the basic idea of force-directed graph

drawing methods is to place each node by a charged ring and each link is replaced by a magnetic spring. Since the charged rings have repulsive forces between each other, and each magnetic spring has a spring force to keep a natural length of the link corresponding to this spring, the equilibrium state between repulsive and spring forces could result in a nice-looking node-diagram diagram layout. In addition, because the springs for inter-community links have stronger string forces than those for intra-community links, the nodes within a community are positioned closer, whereas the nodes from different communities are positioned farther.

B. Generating a triangle-shaped NodeTrix representation

After the geometric positions of nodes and communities of the node-link diagram are determined in the last subsection, this subsection transforms the node-link diagram into a triangle-shaped NodeTrix representation. First, each community is expressed as a triangle-shaped matrix (Fig. 3(b2)) in which each entry grid in this matrix is shaped if the corresponding node is linked. By triangle-shaped matrices, the drawing area can be saved. Then, if a node is overlapped in η communities, it has η duplicate node labels respectively in the η communities and are connected from the entry grids on diagonal sides of matrix triangles by a sequence of green line segments. Then, if two nodes from different communities are connected by a link, their respective node labels (maybe from the vertical or horizontal sides of the matrix triangle) are connected by a black line segment.

C. Minimizing the number of link crossings

After a triangle-shaped NodeTrix representation is obtained, some links and community triangles may be crossed or overlapped with each other. Therefore, this subsection proposes a method of rotating the orientations of community triangles to avoid these crossings. Fig. 4(a) shows four possible orientations of a community triangle. Note that although it is possible to apply other rotational angles, the four orientations in Fig. 4(a) allow users to observe the content within each community triangle, because the triangle is always right and aligned to a grid on the screen.

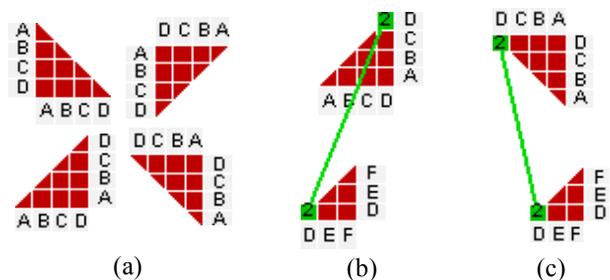


Fig. 4. Importance of orientations. (a) Four orientations of a triangle-shaped matrix. (b) An example with a crossing between a link and a community due to improper orientations of community triangles. (c) No crossing after rotating the upper community triangle of the representation in (b).

An example with a crossing between a link and a community due to improper orientations of community

triangles is shown in Fig. 4(b). If the upper community triangle of the representation in Fig. 4(b) is rotated with 90° counterclockwise as shown in Fig. 4(c), the crossing disappears. In addition to these green overlapping-node links, the black inter-community links may also cause crossings.

This work considers the following strategy to remove the crossings mentioned above. First, consider that each node in a community has two node labels in the community triangle (i.e., one from the vertical side of the triangle and the other from the horizontal side). If some link crossing occurs when the link is connected with the node label from one side of the triangle, we change the link to be connected with the node label from the other side. If this adjustment cannot remove this link crossing, we rotate the orientations of community triangles to which the two nodes incident to this link is attached until this link crossing is removed. However, if the network is of a large size, it is hard to find the linked sides and the triangle orientations with minimal number of crossings. Therefore, we further propose an SA algorithm to efficiently address this problem.

The proposed NodeTrix representation may cause the following five types of link crossings (see also Fig. 5 for illustration):

- (a) crossing between two inter-community links,
- (b) crossing between an inter-community link and a community triangle,
- (c) crossing between two overlapping-node links,
- (d) crossing between an overlapping-node link and a community triangle, and
- (e) crossing between an inter-community link and an overlapping-node link.

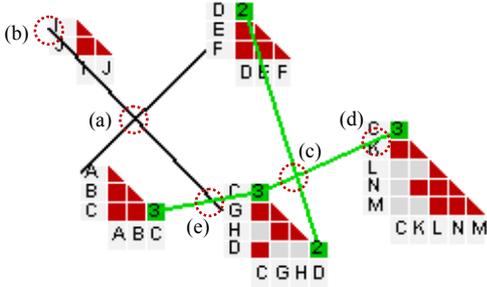


Fig. 5. An example with five types of link crossings.

SA considers an initial candidate solution for the concerned problem and improves this candidate solution iteratively. Different from hill climbing that always find a better solution to replace the current candidate solution, the SA allows a probability to accept a worse solution so as to escape from local optimal solutions. Note that the SA defines a cost function to evaluate performance of solutions. The cost function for the problem considered here is the sum of numbers of the five types of crossings mentioned above.

The key design of the proposed SA is the solution representation (i.e., how to encode the decision variables of the concerned problem into a candidate solution of the SA). Note that the geometry of the NodeTrix representation is decided by three parameters: the position (i.e., $(x,$

$y)$ -coordinate) and the orientation (i.e., one of the four orientations in Fig. 4(a)) of each community triangle and the sides (i.e., vertical and horizontal sides of a triangle) to which each inter-community link is attached. Since the position of each community triangle has been decided by the force-directed graph drawing method provided by the yEd graph editor as described in the last subsections, the candidate solution of the SA is encoded by the other two parameters. Let n denote the number of nodes, m denote the number of inter-community links, and μ denote the number of communities. Then, the candidate solution of the SA is encoded as follows: $(a_1, a_2, \dots, a_\eta \mid b_1, b_2, \dots, b_m)$ where for each $i \in \{1, 2, \dots, \eta\}$, $a_i \in \{1, 2, 3, 4\}$ represents one of four orientations of the i -th community triangle; for each $j \in \{1, 2, \dots, m\}$, $b_j \in \{1, 2\}$ represents either the vertical or the horizontal side, of the community triangle to which the j -th inter-community link is attached.

In the iterative improvement process of the SA, the current candidate solution could be improved by searching for a neighboring solution. Given a current solution, the neighboring solution in the proposed SA is to randomly change one element of the current candidate solution vector to a feasible value.

D. User feedback

Like other metaheuristic algorithms, although the proposed SA can efficiently generate a good-quality triangle-shaped NodeTrix representation, it cannot guarantee the optimality of the representation, i.e., the solution with the minimal number of crossings may not be found. Additionally, positions of community triangles are decided by a force-directed graph drawing method, which cannot guarantee the optimality either. Therefore, the proposed system provides an interface to allow users to freely rotate and flip each community triangle to reduce the number of crossings in the representation.

E. Allocation of overlapping-node links attached to matrix diagonals

This subsection introduces how to allocate the green overlapping-node links attached to matrix diagonals. The entry grid to which a green link is attached depends on the layout of the matrix, determined by the ordering of nodes on the boundary of this matrix. In general, the node ordering in the matrix is decided by the user. Hence, the system only determines the connectivity of the green links, instead of their linking ports. The connectivity of green links has the following two visualization styles:

- *Radial style*: One endpoint of each of the green links for the same overlapping node is connected to the entry grid of the same community triangle. Hence, the visualization of these green links are radial from the same entry grid.
- *Path style*: Consider the connectivity of the green links for the same overlapping node. Let k denote the number of community triangles with this overlapping node. Then, in the *path style*, the entry grids on the diagonals of these community triangles for the overlapping node are

connected one by one as a path. For instance, node C in Fig. 5 is an overlapping node in three communities, and a path consisting of two green links connect the three communities from the triangle diagonals.

In general, the path style causes fewer crossings than the radial style. Hence, this work applies the path style. In addition, green overlapping-node links are connected with green boxes on triangle diagonals (Fig. 6(a)). Each green box is attached with the number of communities for the overlapping node. By reading these numbers, the user can realize the information of overlapping nodes immediately from the visualization.

In addition, the user can give each community a name according to its attributes. The proposed system shows the community name at the corner of vertical and horizontal sides of the community triangle (Fig. 6(a)).

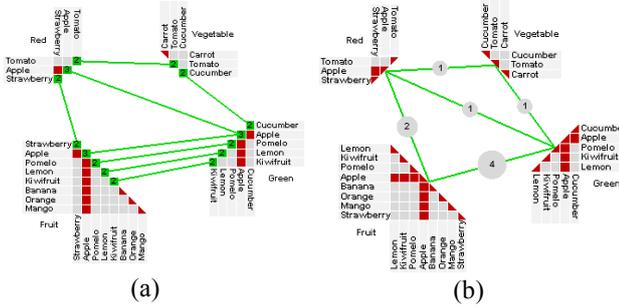


Fig. 6. Visualization mode. (a) Micro-view mode; (b) macro-view mode.

F. Visualization mode

This work provides users two visualization modes for green overlapping-node links. Firstly, the micro-view mode (Fig. 6(a)) is the same as the original setting as described in the last subsection, i.e., displaying all green overlapping-node links. Secondly, the macro-view model (Fig. 6(b)) uses a green inter-community link attached with the number of green links between two communities to connect the two communities. The advantage of the macro-view model is that when the size of the social network is large, the user can have an overview of the network from the macro-view mode. Additionally, for the user that is not familiar with the NodeTrix representation, it would be hard to identify the information on overlapping structures. Therefore, the user can alternatively use these two visualization modes to realize the overview and the details of the network.

IV. ANALYSIS AND DISCUSSION

This section analyzes and discusses the results of some instances generated by the proposed method. Two cases are studied at first, and then the results of the proposed method are compared with the results of previous methods when three core tasks in the SNA are conducted.

A. Interface for analyzing CPSS in the manufacturing industry

The smart factories in the manufacturing industry are constructed with lots of industrial sensors and IoTs, used for monitoring and tracking the production process. Once a WIP

product enters each station, the related attributes are detected and stored in the database. Based on these attributes, the WIP product is classified into some community, because it is usually involved with many social interaction of the staff from different departments. Especially, products are usually of small quantity and much diversity in the modern manufacturing industry, and hence needs to be customized. Each user in this CPSS can work together to make instant decisions, and the corresponding physical logistics operations are adjusted for this WIP product immediately. Hence, it is crucial to provide a good cyber interface to visualize the social network underlying a physical space, so as to establish a sound CPSS.

The interface is based on the proposed triangle-shaped NodeTrix representation. Consider a factory with multiple storage locations. Each storage location stores a batch of WIP products with similar attributes, which will be handled by similar processes and the same community of people. Hence, each WIP product is viewed as a node in the social network; and each storage location is viewed as a community of the social network. If two WIP products have a relationship (e.g., similar prices, similar sales, and similar manufacturing dependencies), then their corresponding nodes are connected by a link. Furthermore, some WIP products may be of the same product type but can be classified into multiple communities. Hence, these WIP products are viewed as overlapping nodes in the network. By doing so, the relationship of WIP products can be represented as a social network with overlapping community structures, and can be visualized by the proposed NodeTrix representation (Fig. 6).

By the interface based on the proposed NodeTrix representation, the user can base on the previous experiences to adjust the allocation of WIP products in each storage location. After the user makes a decision via the cyber interface, the allocation of the related WIP products will be adjusted immediately by automated material handling systems (AMHS) in the smart factory. In addition, by observing the size of each community triangle in the visualization, the user can realize the inventory conditions of WIP products, and then make appropriate decisions, further influencing the physical space.

Recently, more and more products have been requested to be customized, so that products become much diversified and lead to a shorter product life cycle. Hence, flexible replacements of components and flexible adjustments of the production line may be required at any time. Therefore, it is crucial to visualize the information of these components to further manage them.

Consider a case of the CPSS visualization interface for a bicycle manufacturing factory. Typical bicycle factories are classified into two types. The first type is to purchase raw materials and then to transform them into components of bicycles. The second type is to purchase the components produced by the first type and then to assemble them into bicycles. Therefore, the second type requires high-efficiency assembly lines to fulfill the customer orders. With mature

development of automation, bicycle factories have tended to apply robots to achieve high-efficiency production, rather than manual production. Hence, the CPSS interface for the second type of bicycle factories displays the online conditions of components on one or multiple production lines. In the CPSS, according to the information on customers' demand and the real manufacturing conditions, the staff can use the visualization interface and the cyber system to classify and reallocate the physical positions of these components in the warehouse to smooth the production process.

In this factory, a bicycle consists of 32 components that are divided into five parts: frame, transmission, wheel, steering, and brake systems. Consider a bicycle factory with five stations that are in charge of finishing the five systems, respectively. Our proposed interface for the bicycle factory is given in Fig. 7. In Fig. 7, each triangle-shape adjacency matrix represents the relationship of components in a station (system), in which each text label along the matrix borderline is associated with a component of bicycles; each red entry in a matrix represents that a process between the two corresponding components can be operated. Moreover, each label is associated with a colored box to show three inventory conditions of the corresponding component: light green (i.e., sufficient quantity), orange (i.e., not sufficient quantity), and red (i.e., shortage). Through visualizing the inventory conditions, the staff in the factory may be aware of the time when some component should be replenished, and they can monitor all process in real time.

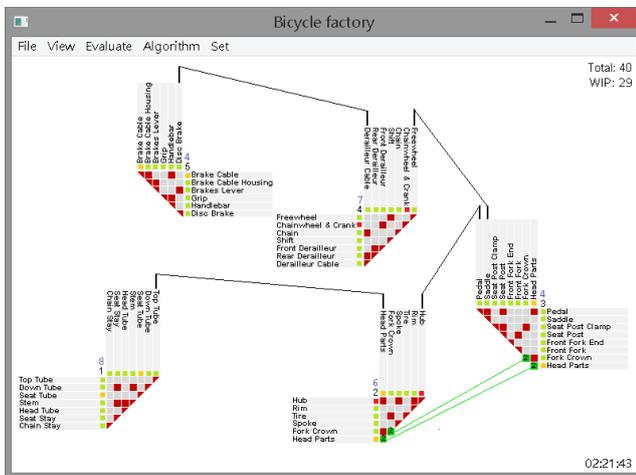


Fig. 7. The proposed interface used in a bicycle factory.

There are two numbers at the right-angle corner of each matrix: the black number indicates the ID of the station, and the blue number indicates the number of WIPs to be assembled in this station. In addition, the upper right corner of the window in Fig. 7 displays the total number of finished products and the number of WIPs in this factory. Each inter-community link (colored in black) represents the process relationship of two stations. In Fig. 7, the process is sequential from Station 1 to Stations 2, 3, 4, and 5. Overlapping-node

links (colored in green) represents that components 'Fork Crown' and 'Head Parts' are overlapping nodes in Stations 2 and 3, meaning that these two components can be assembled in either one of the two stations.

Note that positions of matrices respect the real geographical locations of stations. That is, we do not optimize these positions using the approach introduced in last sections. By doing so, the factory staff can clearly and easily observe the correspondence between the physical space and the cyber interface. For instance, in Fig. 7, the process starts from the frame system, and is finished at the brake system. The corresponding information can also be realized through this CPSS interface.

B. Interface for analyzing foods in food production industries

This subsection investigates a case in food production industries. Consider a food production factory. To satisfy the demand and preferences of a large number of customers, the factory usually provides diversified products such as frozen food to supermarkets; and heat-and-eat frozen food and fresh fruits to convenience stores. To meet a large number of demand, the factory usually stores a lot of raw materials to the warehouse, and hence, it is crucial to manage and store these raw materials. For example, fruits and vegetables are perishable, some of them cannot be stored nearby, and the temperature of storing each of them may not be the same. Therefore, before storing these raw materials to the warehouse, their food features will be recorded, e.g., color, nutrient content, variety, and quantity. If an interface for visualizing these information can be designed, the warehouse staff can easily manage and preserve these raw materials.

It is crucial to have an interface to analyze and classify food features, before producing food in factories. This subsection visualizes the graph for food features proposed by Wyatt [26] drawn as an Euler diagram in Fig. 8(a), in which each element is a food; if two foods share similarity of some features, then they are in the same community. This graph is visualized by the proposed method in Fig. 8(b).

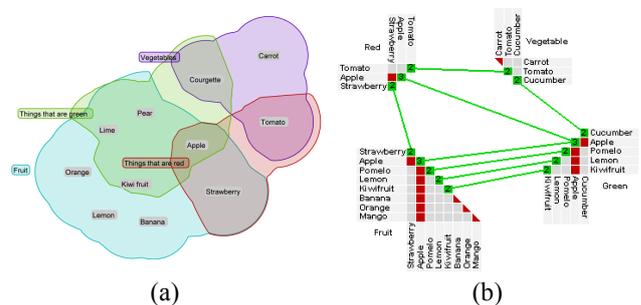


Fig. 8. Visualizing the overlapping community structure of the graph for food features by (a) the Euler diagram based method [29] and (b) the proposed method.

In the factories for producing these foods, the proposed CPSS system provides users a cyber interface with some operations. The first operation proposed in this interface is the exploration, which makes users to link old and new objects, or

enhance the linkage of some objects. Take the visualization in Fig. 8(b) for an example. In Fig. 8(b), when the user intends to analyze the relationship of apples with other foods, the user can make a judgement through visualizing the NodeTrix representation. First, we observe that ‘Apple’ belongs to the community of ‘Fruit’ (i.e., it is edible), and we can find other nodes from this ‘Fruit’ community. By the diagonal of the community triangle of ‘Fruit’, we may observe that ‘Apple’ is an overlapping node of three communities. Following the green overlapping-node link connected with ‘Apple’ in the ‘Fruit’ community, we observe that ‘Apple’ also appears in the ‘Green’ community, i.e., ‘Apple’ is green. From the ‘Green’ community, ‘Apple’ has the same color with ‘Kiwifruit’ because they are adjacent in this matrix. Following the green path of the overlapping-node links for ‘Apple’, we may observe that ‘Apple’ also appears in the ‘Red’ community. Hence, it is concluded that ‘Apple’ can be red or green. By similar visual analysis, the linkage between ‘Apple’ and other foods can be found. When there are more communities, it is concrete to get more insights into and more linkages with the information of some unknown food through the visualization, before producing this food.

The second operation proposed in this interface is to observe the content of each triangle-shaped adjacency matrix to realize the relationship among foods. For instance, the relationship of other features of the foods can also be visualized by the proposed NodeTrix representation, such as the nutrition components of foods. Consider a scenario when the user intends to handle the ‘Apple’ products but the row materials are not enough. We may explore this visualization to find some food product with similar nutrition components to be the alternative of ‘Apple’. Conversely, if the user would like to find the food whose nutrition components are much different from the ‘Apple’ product, we may also explore the details of adjacency in the matrix to find complementary products.

C. Analyzing three core tasks in the SNA

The previous work for Nodetrix representations [19] visualized the social networks of coauthorship, and conducted three core tasks in the SNA: (T1) identifying communities, (T2) identifying the core actor, and (T3) analyzing the role and position of some actor. Hence, we analyze these three tasks conducted in the proposed NodeTrix representation.

For task (T1), each triangle in the propose representation is a community, and hence the user can finish task (T1) easily (e.g., Fig. 8(b)). For task (T2), overlapping nodes are core actors in the network, because they appear in at least one community. If one would like to approach more communities, the overlapping nodes should be the first actor to be approached. Additionally, if some overlapping node disappears, the related communities may lose the bridge between them. Through the proposed NodeTrix representation, overlapping nodes can be observed by green links. Hence, task (T2) can be done easily.

For task (T3), the conventional NodeTrix representation [19]

considered three patterns for matrix layouts: (a) cross pattern, (b) block pattern, and (c) intermediate pattern. Hence, we extend this concept to the triangle-shaped matrix layouts, as shown in Fig. 9. The ‘cross pattern’ for square-shaped matrix is shown as an L-shape pattern (Fig. 9(a)). This pattern means that almost all actors in this community have no strong relations among each other, but everyone has a relation with some core actor (e.g., ‘Shneiderman’ in Fig. 9(a)). The ‘block pattern’ in Fig. 9(b) shows two triangular blocks in a community, meaning that this community can further be classified into two sub-communities with tight relations within their respective sub-community. The ‘intermediate pattern’ in Fig. 9(c) is a mix of ‘cross pattern’ and ‘block pattern’. For instance, Fig. 9(c) can be divided into two sub-communities. The members in the sub-community with ‘block pattern’ has relations among each other; whereas the members in the sub-community with ‘cross pattern’ have no relations with each other but have a relation with ‘Roth’.

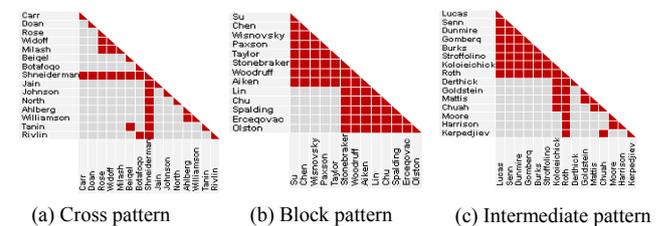


Fig. 9. Three patterns of community triangles in the proposed NodeTrix representation.

In what follows, the NodeTrix representations generated by the conventional method in [18] and the proposed method are compared in Fig. 10. Note that a part of Fig. 10(a) is truncated, but does not affect the later comparison with our result. From Fig. 10(a), it would be hard to visually recognize inter-community links (i.e., blue lines) and overlapping-node links (i.e., gray curve bands) because the two types of links may be connected from the same node labels. The proposed representation in Fig. 10(b) shows the difference between the two types of links through features of triangle shapes: inter-community links (i.e., black lines) are connected from vertical or horizontal sides of triangles, whereas overlapping-node links (i.e., green lines) are connected from diagonals of triangles. In addition, numbers of overlapping communities are printed on diagonals. Hence, it is easy to recognize them. The other advantage of the proposed NodeTrix representation is to save the drawing area. From Fig. 10, more blank space can be viewed in Fig. 10(b), and hence, it is clearer to visualize the information of this representation.

D. Analyzing the number of link crossings

The proposed visualization technique applies the SA to reduce five types of link crossings. The experimental analysis on the results in Figs. 8(b) and 9(b) are given in Table I.

The relationship in Fig. 8(a) can be represented as a node-link diagram with 11 nodes and 12 links. The proposed

approach divides this diagram into four communities with 0 inter-community link and 8 overlapping-node links, and finally generates a NodeTrix representation with 0 link crossing (Fig. 8(b)).

The relationship in Fig. 8(a) can be represented as a node-link diagram with 79 nodes and 467 links. The proposed approach divides the diagram into 12 communities with 148 inter-community links and 6 overlapping-node links, and finally generates a NodeTrix representation with 63 link crossings (Fig. 10(b)). Note that in the weights of penalizing the five types of link crossings in the SA can be adjusted according to practical applications. Since the instance of Fig. 10 has fewer overlapping-node links, the weights of penalizing types (c), (d), and (e) are reduced and the weights of penalizing types (a) and (b) are raised in the experimental setting.

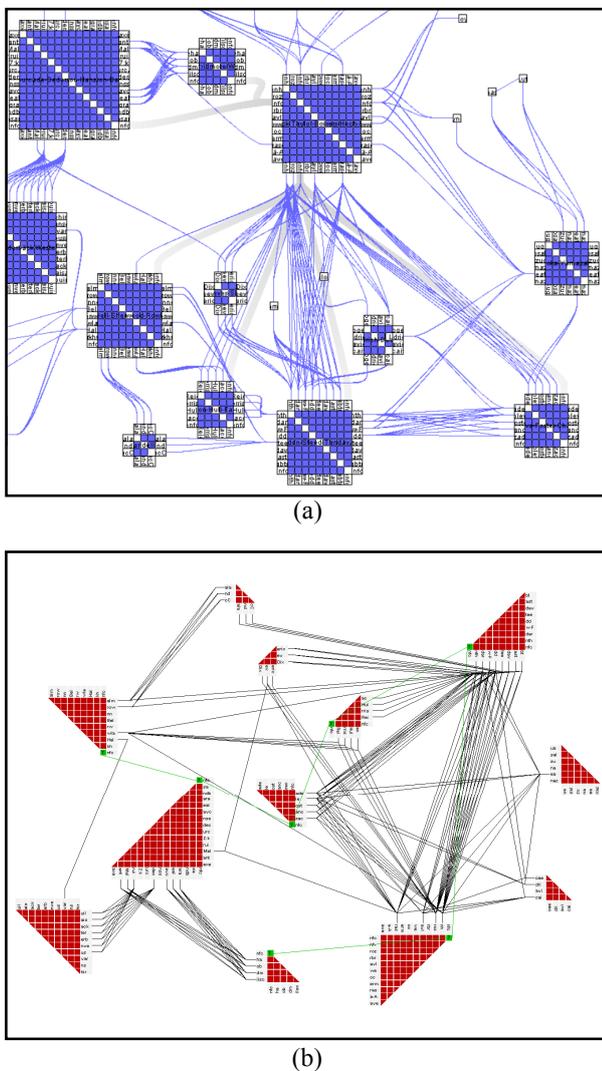


Fig. 10. Visualizing a social network with overlapping communities by (a) conventional NodeTrix method [18] and (b) the proposed method.

TABLE I
Statistics of the experimental results

Instance	Diagram		Our method		
	Number of nodes	Number of links	Inter-community links	Overlapping-node links	Link crossings
Fig. 8(b)	11	12	0	8	0
Fig. 10(b)	79	467	148	6	63

V. CONCLUSION

This work has proposed a visualization interface based on an improved NodeTrix representation for the social network with overlapping community structures for the CPSS. This NodeTrix representation is improved by applying triangle-shaped adjacency matrices to address the problem of visualizing overlapping community structures. By this visualization, the user can observe the relationships of individuals, communities, and overlapping communities, to further explore the hidden information and insights. This work further applies this interface to the CPSS for smart factories and the food production industry, i.e., the physical space of the social network is manipulated by the proposed software interface based on NodeTrix. In addition, the experimental comparison of the proposed and the pervious methods is made.

In the future, more delicate operations in this interface can be designed, so that the user can execute more complex cyber-physical operations. In addition, it would be of interest to analyze the performance of the interface on mobile devices. Furthermore, considering complex social behaviors would also enrich the functionality of this interface.

REFERENCES

- [1] L. Wang, M. Torngren, and M. Onori, "Current status and advancement of cyber-physical systems in manufacturing," *Journal of Manufacturing Systems*, vol. 37, no. 2, pp. 517–527, 2015.
- [2] M. Dong, K. Ota, L. T. Yang, A. Liu, and M. Guo, "LSCD: A low-storage clone detection protocol for cyber-physical systems," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 35, no. 5, pp. 712–723, 2016.
- [3] J. Sahoo, S. Cherkaoui, and A. Hafid, "Optimal selection of aggregation locations for participatory sensing by mobile cyber-physical systems," *Computer Communications*, vol. 74, no. 15, pp. 26–37, 2016.
- [4] Z. Zhang, H. Wang, C. Wang, and H. Fang, "Interference mitigation for cyber-physical wireless body area network system using social networks," *IEEE Transactions on Emerging Topics in Computing*, vol. 1, pp. 121–132, 2013.
- [5] F. Xia, A. Vinel, R. Gao, L. Wang, and T. Qiu, "Evaluating IEEE 802.15.4 for cyber-physical systems," *EURASIP Journal on Wireless Communications and Networking*, vol. 2011, 596397, 2011.
- [6] C. Huang, J. Marshall, D. Wang, and M. Dong, "Towards reliable social sensing in cyber-physical-social systems," in *Proc. of 2016 IEEE International Parallel and Distributed Processing Symposium*, pp. 1796–1802, 2016.
- [7] Z. Liu, D.-S. Yang, D. Wen, W.-M. Zhang, and W. Mao, "Cyber-physical-social systems for command and control," *IEEE Intelligent Systems*, vol. 26, no. 4, pp. 92–96, 2011.
- [8] A. Smirnov, T. Levashova, N. Shilov, and K. Sandkuhl, "Ontology for cyber-physical-social systems self-organisation," in *Proc. of IEEE Conference of Open Innovations Association (FRUCT16)*, pp. 101–107, 2014.
- [9] A. Smirnov, A. Kashevnik and N. Shilov, "Cyber-physical-social system self-organization: Ontology-based multi-level approach and case study,"

- in *Proc. of IEEE 9th International Conference on Self-Adaptive and Self-Organizing Systems*, pp.168–169, 2015.
- [10] H. Ning and H. Liu, “Cyber-physical-social based security architecture for future internet of things,” *Advances in Internet of Things*, vol. 2, pp. 1–7, 2012.
- [11] A. Smirnov, A. Kashevnik, and A. Ponomarev, “Multi-level self-organization in cyber-physical-social systems: smart home cleaning scenario,” in *Proc. of 7th Industrial Product-Service Systems Conference*, *Procedia CIRP*, vol. 30, pp. 329–334, 2015.
- [12] L. Kuang, L. Yang, and Y. Liao, “An integration framework on cloud for cyber physical social systems big data,” *IEEE Transactions on Cloud Computing*, in press.
- [13] X. Yu, A. Pan, L.A. Tang, Z. Li and J. Han, “Geo-friends recommendation in GPS-based cyber-physical social network,” in *Proc. of 2011 International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, IEEE Press, pp. 361–368, 2011.
- [14] B. Guo, Z. Yu, and X. Zhou, “A data-centric framework for cyber-physical-social systems,” *IT Professional*, vol. 17, no. 6, pp. 4–7, 2015.
- [15] V. Hahanov, W. Gharibi, A. P. Kudin, I. Hahanov, N. Christopher, T. Yeve, D. Krulevska, A. Yerchenko, A. Mishchenko, D. Shcherbin, and A. Priymak, “Cyber physical social systems - future of Ukraine,” in *Proc. of IEEE East-West Design & Test Symposium (EWDTS)*, pp. 1–15, 2014.
- [16] M. Goldberg, S. Kelley, M. Magdon-Ismail, K. Mertsalov, and A. Wallace, “Finding Overlapping communities in social networks,” in *Proc. of 2010 IEEE 2nd International Conference on Social Computing (SocialCom)*, pp. 104–113, 2010.
- [17] X. Deng, G. Li, M. Dong, and K. Ota, “Finding overlapping communities based on Markov chain and link clustering,” *Peer-to-Peer Networking and Applications*, vol. 10, no. 2, pp. 411–420, 2017.
- [18] N. Henry, A. Bezerianos, and J.-D. Fekete, “Improving the readability of clustered social networks using node duplication,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 14, no. 6, pp. 1317–1324, 2008.
- [19] N. Henry, J.-D. Fekete, and M. J. McGuffin, “NodeTriX: a hybrid visualization of social networks,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 13, no. 6, pp. 1302–1309, 2007.
- [20] N. H. Riche and T. Dwyer, “Untangling Euler diagrams,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 16, no. 6, pp. 1090–1099, 2010.
- [21] C. Vehlou and D. Weiskopf, “Visualizing fuzzy overlapping communities in networks,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 19, no. 12, pp. 2486–2495, 2013.
- [22] B. Alsallakh, W. Aigner, S. Miksch, and H. Hauser, “Radial sets: Interactive visual analysis of large overlapping sets,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 19, no. 12, pp. 2496–2505, 2013.
- [23] S. Wang, D. Wang, L. Su, L. Kaplan, T. F. Abdelzaher, “Towards cyber-physical systems in social spaces: The data reliability challenge,” in *Proc. of 2014 IEEE Real-Time Systems Symposium (RTSS)*, pp. 74–85, 2014.
- [24] L. Euler, *Lettres a une princesse d’Allemagne sur divers sujets de physique et de philosophie*, volume 1 letters no. 102–108. Courcier, 1772.
- [25] C. Vehlou, T. Reinhardt, and D. Weiskopf, “Visualizing fuzzy overlapping communities in networks,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 19, no. 12, pp. 2486–2495, 2013.
- [26] K. Misue, “Drawing bipartite graphs as anchored maps,” in *Proc. of the Asia-Pacific Symposium on Information Visualisation (APVIS)*, pp. 169–177, 2006.
- [27] N. Henry and J.-D. Fekete, “MatrixExplorer: a dual-representation system to explore social networks,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 12, no. 5, pp. 677–684, 2006.
- [28] yEd Graph Editor, available at <https://www.yworks.com/products/yed>. 2016.
- [29] D. F. Wyatt, Set Visualiser, available at <http://cdn-media-ie.pearltrees.com/0d/dc/0ddc07ba3984de1340170742b58ef2c8-1.jpg>, 2010.