

# Mining Attributed Interaction Networks on Industrial Event Logs

Martin Atzmueller and Benjamin Kloepper

<sup>1</sup> Tilburg University, Department of Cognitive Science and Artificial Intelligence,  
Warandelaan 2, 5037 AB Tilburg, The Netherlands

`m.atzmuller@uvt.nl`

<sup>2</sup> ABB AG, Corporate Research Center,  
Wallstadter Str. 59, 68526 Ladenburg, Germany

`benjamin.kloepper@de.abb.com`

**Abstract.** In future Industry 4.0 manufacturing systems reconfigurability and flexible material flows are key mechanisms. However, such dynamics require advanced methods for the reconstruction, interpretation and understanding of the general material flows and structure of the production system. This paper proposes a network-based computational sensemaking approach on attributed network structures modeling the interactions in the event log. We apply descriptive community mining methods for detecting patterns on the structure of the production system. The proposed approach is evaluated using two real-world datasets.

## 1 Introduction

In the context of Industry 4.0, future manufacturing systems will be more flexible in order to answer more readily to changing market demands [24] as well as disturbances in the production systems. In particular, this is one of the key aspects in the concept of Industry 4.0 [26] or Cloud Manufacturing [27]. Important capabilities of such flexible and robust manufacturing systems are reconfigurability and flexible material flows [27]. As a consequence, the relationships between elements in the production systems like industrial robots, machining centers and material handling systems become more dynamic as well and the interaction between the (resource) elements becomes thus also more difficult to comprehend. On the other hand, understanding the general material flow and the structure of the production systems is required for continuous improvements processes, for instance process mapping is a key activity in the six-sigma process [11].

This paper proposes a network-based approach to recreate the material flow and resource interactions from the log files of the individual components of a production systems. We model log-files as attributed network structures, connecting devices by links labeled with log statements. This allows to detect densely connected groups of devices with an according description of (log) statements. In our experiments, we apply two real-world datasets from serial production systems with a clear hierarchical structure providing a ground truth for evaluating the performance of the proposed algorithmic approach. Our results show the impact and efficacy of our novel network-based analysis and mining approach.

## 2 Related Work

Below, we discuss related work concerning the analysis of industrial (alarm) event logs, i. e., in alarm management and in the context of process mining.

### 2.1 Analysis of Alarm Event Logs

Analysis of event logs has been performed in the context of alarm management systems, where sequential analysis is performed on the alarm notifications. In [13], an algorithm for discovering temporal alarm dependencies is proposed which utilizes conditional probabilities in an adjustable time window. In order to reduce the number of alarms in alarm floods, [2] also performed root cause analysis with a Bayesian network approach and compared different methods for learning the network probabilities. A pattern-based algorithm for identifying causal dependencies in the alarm logs is proposed in [25], which can be used to aggregate alarm information and therefore reduce the load of information for the operator. Furthermore, [6, 10] target the analysis of sequential event logs in order to detect anomalies using a graph-based approach. Finally, [21] investigate the prediction of the risk increase factor in nuclear power plants using complex network analysis using topological structure.

In contrast to those approaches, the proposed approach is not about sequential analysis of event logs, nor on the given static network structures. Instead, we provide a network-based approach transforming event logs into (attributed) networks capturing the static interactions and dependencies captured in the event log. The goal is to identify structural dependencies and relations of the production process. Thus, similar to evidence networks in the context of social networks, e. g., [18], we aim to infer the (explicit) structural relations given observed (implicit) interactions between the industrial equipment and devices.

### 2.2 Analysis of Event Logs using Process Mining

Process Mining [1] aims at the discovery of business process related events in a sequential event log. The assumption is that event logs contain fingerprints of business processes, which can be identified by sequence analysis. One task of process mining is conformance checking [19, 22] which has been introduced to check the matching of an existing business process model with a segmentation of the log entries. Furthermore, for process mining and anomaly analysis there have been approaches based on subgroup discovery, e. g., [23], and subgraph mining, e. g., [14] based on log data; while these neglect the temporal (sequential) dimension, they only focus on the respective patterns not including a priori knowledge, while not including relational, i. e., network modeling.

Compared to these approaches, we do not use any apriori (process) knowledge for our analysis. In contrast, we use a purely data-driven approach, where we perform a feature-rich network-based approach on the event log data. For that, we transform the (event log) interaction data into an attributed interaction network which is then exploited for mining cluster/community structures together with an explicit description – enhancing interpretation and understandability.

### 3 Method

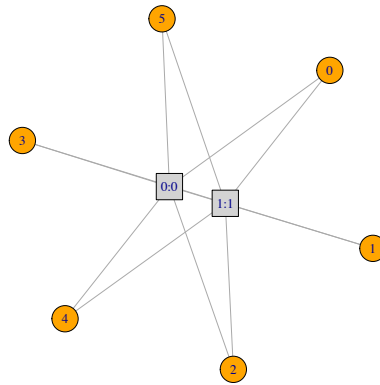
In Industry 4.0 environments like complex industrial production plants, intelligent data analysis is a key technique for providing advanced data science capabilities. In that context, computational sensemaking [5] aims to develop methods and systems to “make sense” of complex data and information – to make the implicit explicit; important goals are then to comprehensively model, describe and explain the underlying structure in the data [4]. This paper presents a computational sensemaking approach using descriptive pattern mining. The proposed approach consists of three steps: (1) We model the event log as a bimodal network represented as a bipartite graph. (2) We create an attributed graph structure using a projection operator with labels taken from the bimodal structure. (3) Finally, we apply pattern mining (i. e., descriptive community mining) on the attributed graph, in order to detect structural patterns and relations.

#### 3.1 Modeling Attributed Interaction Networks from Event Logs

In the following, we use the data shown in Table 1 as an example for demonstrating the individual steps of the proposed approach. As can be seen in the table, it considers log entries corresponding to a certain *device* and *event\_type* in addition to a timestamp. We focus on the *device* and *event\_type* information creating a bimodal network. However, first we aggregate the *event\_type* information for a device, such that equal *event\_types* for a specific *device* are merged into a single link between *device* and the corresponding *event\_type*, respectively. In our example, line #1 and line #13 would thus be merged into a single link. The resulting bipartite graph is shown in Figure 1. This can already be considered as an attributed graph, where we interpret links between the devices labeled by their common *event\_types*. In our example, every device is connected to every other device with a link labeled with the common 0 : 0 (“Safety Stop Activate”) and 1 : 1 (“System is in Safety Stop”) event types.

#	Device	Event_Type	Timestamp
1	0	0:0	12.08.12 07:23
2	1	1:1	12.08.12 07:23
3	2	1:1	12.08.12 07:23
4	2	0:0	12.08.12 07:23
5	0	1:1	12.08.12 07:23
6	1	0:0	12.08.12 07:23
7	3	1:1	12.08.12 07:24
8	4	0:0	12.08.12 07:24
9	4	1:1	12.08.12 07:24
10	5	1:1	12.08.12 07:24
11	5	0:0	12.08.12 07:24
12	3	0:0	12.08.12 07:24
13	0	0:0	12.08.12 10:59

**Table 1.** Exemplary (anonymized) log event data, visualized by the bipartite graph to the right.



**Fig. 1.** Bipartite graph (example data, left): *devices* (orange circles) and linked *event types* (gray squares).

### 3.2 Descriptive Community Mining

Community detection [20] aims at identifying densely connected groups of nodes in a graph; using attributed networks, we can additionally make use of information assigned to nodes and/or edges. For mining attributed network structures, we apply the COMODO algorithm presented in [7]: It focuses on *description-oriented community detection* using subgroup discovery [3], and aims at discovering the top- $n$  communities (described by community patterns). COMODO utilizes efficient pruning approaches for scalability, for a wide range of standard community evaluation functions. Its results are a set of patterns (given by conjunctions of literals, i. e., attribute–value pairs) that describe a specific subgraph – indicating a specific community consisting of a set of nodes. An example in the context of the analysis of event logs is given by the pattern:  $event_1 AND event_2 AND event_5$  indicating the event (types)  $event_1$ ,  $event_2$ , and  $event_5$  being jointly connected to the same set of devices. This pattern then directly corresponds to the (covered) subgraph.

**Algorithmic Overview** COMODO utilizes both the graph structure, as well as descriptive information of the attributed graph. As outlined above, we transform the graph data into a new dataset focusing on the edges of the graph  $G$ : Each data record in the new dataset represents an edge between two nodes. The attribute values of each such data record are the common attributes of the edge’s two nodes. For efficiency, COMODO utilizes an extended FP-tree (frequent pattern tree) structure inspired by the FP-growth algorithm [15], which compiles the data into a prefix pattern tree structure, cf. [9,17]. Our adapted tree structure is called the *community pattern tree* (CP-tree) that allows to efficiently traverse the solution space. The tree is built in two scans of the graph dataset and is then mined in a recursive divide-and-conquer manner. Efficient pruning is implemented using optimistic estimates [7]. For community evaluation a set of standard evaluation functions exists, including the Modularity function [20]. As a result, COMODO provides the top- $n$  patterns according to a given community evaluation function. For a more detailed description, we refer to [7].

**Community Postprocessing** As a final result, we aim at a disjoint partition of the set of nodes in our input graph – which should correspond to the different levels (and category groups). However, the set of communities (or clusters) provided by COMODO can overlap. For the industry 4.0 use case this property is very useful, because overlapping resource communities are expected due to reconfigurability and flexible material flows. In the given dataset, however the devices in the production system are organized in a two-level hierarchy with non-overlapping groups. Thus, we apply a postprocessing step, in order to obtain a disjoint partition of the graph from the given set of top- $n$  patterns. Essentially, given the communities, we construct a similarity graph for the set of nodes: For each pair of nodes, we check the number of times they are contained in a community (pattern), and create a weighted edge accordingly, normalized by the total number of patterns. Then, we uncover (disjoint) communities on the (pruned) similarity graph by a further community detection step.

## 4 Results

In this section, we first describe the characteristics and context of the applied real-world datasets. After that, we present results and discuss them in detail.

### 4.1 Datasets

Two real-world datasets from the industrial domain are used in this work. Both datasets are from serial production facilities with several production lines and cell. The first dataset (*Log-Data-A*) contains data from 59 industrial machines and devices from 8 different production lines and 7 production cells. The second dataset (*Log-Data-B*) contains data from 48 machines and devices from 2 production lines with 16 production cells. Basically, each device is assigned to a production line and production cell, where the production lines can be considered as *level 1 categories*, and the production cells as *level 2 categories*, representing the production hierarchy. In the dataset, this information can be used as ground-truth in order to evaluate the mined patterns and communities, respectively. Since the community structures should represent the material flows, this directly corresponds to the respective level 1 and level 2 categories. It is important to note that these categories are a disjoint partitioning of the set of devices, respectively. Therefore, as explained above, we also aim at a disjoint partitioning of the graph given the set of communities.

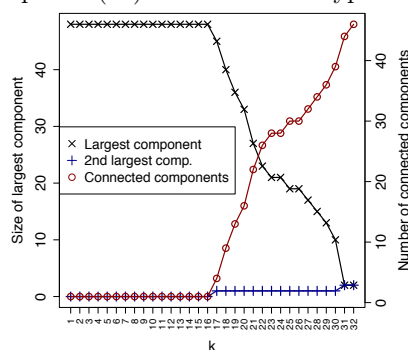
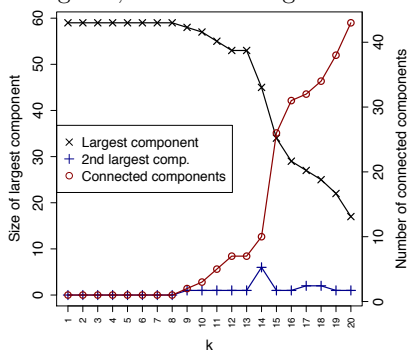
The event logs contain both normal events, warnings and error events and partially capture the standard activity of the devices (e.g. motor starts and stops, program starts), operator interactions (e.g. safety stops, switching operation modes) and information of interactions with supplementary process like cooling water supply. Due to serial production fashion, products pass through the production lines in a sequential fashion. Consequently, activities of machines and devices are triggered according to the production line and cell structure. Furthermore, the product flow closely interlinks the industrial machines and devices and failure and problems propagate usually forward through the production systems. These features make the two datasets ideal to develop a proof of concept of recovering the flow of material in a production systems from the event log data generated by the individual machines and devices. Table 2 summarizes the characteristics of both datasets.

**Table 2.** Characteristics of the real-world datasets

Dataset	#Devices	#EventTypes	#Prod. Lines	# Prod. Cells	# Events
Log-Data-A	59	356	8	7	50000
Log-Data-B	48	102	2	16	50000

## 4.2 Results and Discussion

First, we take a look at the connectivity structure of our attributed graphs. Figures 2-3 depict according (extended) KNC-Plots [8, 16] that visualize the number  $k$  of common neighbors of the nodes in the original bipartite graph, as well as the sizes of the largest and 2nd largest components. In our case,  $k$  indicates the number of common *event\_types* connecting the respective *device* nodes. Overall, the graphs exhibit a strong connectivity structure: As we can see in the figures, there is strong connectivity up to 8 (16) common event\_types.



**Fig. 2.** KNC-Plot: Log-Data-A Dataset. **Fig. 3.** KNC-Plot: Log-Data-B Dataset.

For community detection aiming at reconstructing the production system structure in our application scenario, we applied the COMODO algorithm using the modularity evaluation function, with no minimal support threshold. Regarding the only parameter, i. e., determining  $n$  for the top- $n$  patterns, we experimented with different selections, where we used  $n = 20$  for interpretability. However, with other selections the results as outlined below were quite stable. Finally, for the postprocessing step constructing the similarity graph we pruned edges with a weight below 0.1 such that edges needed to be “supported” by at least 2 community patterns in order to be included in the final similarity graph. For determining the final set of disjoint communities, we utilized the edge betweenness [20] method.

Table 3 shows our results using the Normalized Mutual Information (NMI) measure for comparing community structures using the (production line/cell) category information as ground truth for the different communities/clusters. We compared different baseline methods to our proposed approach using the COMODO algorithm using standard algorithms as included in the igraph [12] software package, i. e., edge betweenness, fast greedy, Infomap, label propagation, leading eigenvector, and louvain. In particular, Infomap and label propagation yielded NMI values of 0, detecting no structure. As we can observe in the table, COMODO outperforms all the other algorithms, while the baselines yield relatively low NMI values discovering no relations. Thus, in comparison to the baselines the proposed approach using COMODO does not only outperform standard community approaches, but also provides descriptive patterns that can be used for inspection, interpretation and explanation.

**Table 3.** Results: NMI for Different Community Detection Approaches

Algorithm / NMI	Log-Data-A		Log-Data-B	
	Level1	Level2	Level1	Level2
Edge Betweenness	0.32	0.20	0.02	0.11
Fast Greedy	0.48	0.24	0.01	0.15
Leading Eigenvector	0	0	0.01	0.15
Louvain	0.48	0.24	0.01	0.15
COMODO	0.67	0.53	0.19	0.78

## 5 Conclusions

This paper presented a network-based approach to recreate production system structures and resource interactions from industrial event log data. We modeled those as attributed networks and detected densely connected groups of devices with an according description of (log) statements. For evaluation, we applied two real-world datasets. Our results indicated the impact and efficacy of the proposed network-based approach, outperforming standard community detection baselines while also providing descriptive patterns for interpretation and explanation.

Beyond confirming the applicability of event log analysis for reconstructing resource interactions and material flows, the analysis can also help to detect hotspots in the production process, e. g., segments of the production process in which high amounts of events are generated and thus potentially require special attention in continuous improvement processes like Six Sigma. Thus, advanced hotspot analysis and anomaly detection are interesting directions for future work. Also, analyzing the evolution of the network – capturing dynamics and temporal dependencies in the event logs – is another interesting direction to consider.

## Acknowledgements

This work has been partially funded by the German Research Foundation (DFG) project “MODUS” (grant AT 88/4-1) and by the EU project Productive 4.0.

## References

1. Aalst, W.: Process Mining: Discovery, Conformance and Enhancement of Business Processes. Springer (2011)
2. Abele, L., Anic, M., Gutmann, T., Folmer, J., Kleinstauber, M., Vogel-Heuser, B.: Combining Knowledge Modeling and Machine Learning for Alarm Root Cause Analysis. In: MIM. pp. 1843–1848. IFAC (2013)
3. Atzmueller, M.: Subgroup Discovery. WIREs DMKD **5**(1), 35–49 (2015)
4. Atzmueller, M.: Onto Explicative Data Mining: Exploratory, Interpretable and Explainable Analysis. In: Proc. Dutch-Belgian Database Day. TU Eindhoven (2017)
5. Atzmueller, M.: Declarative Aspects in Explicative Data Mining for Computational Sensemaking. In: Proc. DECLARE. Springer (2018)

6. Atzmueller, M., Arnu, D., Schmidt, A.: Anomaly Detection and Structural Analysis in Industrial Production Environments. In: Proc. IDSC. Salzburg, Austria (2017)
7. Atzmueller, M., Doerfel, S., Mitzlaff, F.: Description-Oriented Community Detection using Exhaustive Subgroup Discovery. *Inf. Sci.* **329**, 965–984 (2016)
8. Atzmueller, M., Hanika, T., Stumme, G., Schaller, R., Ludwig, B.: Social Event Network Analysis: Structure, Preferences, and Reality. In: Proc. IEEE/ACM ASONAM. IEEE Press, Boston, MA, USA (2016)
9. Atzmueller, M., Puppe, F.: SD-Map - A Fast Algorithm for Exhaustive Subgroup Discovery. In: Proc. PKDD. pp. 6–17. Springer (2006)
10. Atzmueller, M., Schmidt, A., Kloeppe, B., Arnu, D.: HypGraphs: An Approach for Analysis and Assessment of Graph-Based and Sequential Hypotheses. In: *New Frontiers in Mining Complex Patterns (Proc. NFMCP)*. LNAI, Springer (2017)
11. Chen, J.C., Li, Y., Shady, B.D.: From Value Stream Mapping Toward a Lean/Sigma Continuous Improvement Process: An Industrial Case Study. *International Journal of Production Research* **48**(4), 1069–1086 (2010)
12. Csardi, G., Nepusz, T.: Package igraph: Network Analysis and Visualization (2014)
13. Folmer, J., Schuricht, F., Vogel-Heuser, B.: Detection of Temporal Dependencies in Alarm Time Series of Industrial Plants. *Proc. IFAC* pp. 24–29 (2014)
14. Genga, L., Potena, D., Martino, O., Alizadeh, M., Diamantini, C., Zannone, N.: Subgraph Mining for Anomalous Pattern Discovery in Event Logs. In: *New Frontiers in Mining Complex Patterns. Proc. NFMCP*. pp. 181–197. Springer (2016)
15. Han, J., Pei, J., Yin, Y.: Mining Frequent Patterns Without Candidate Generation. In: Proc. SIGMOD. pp. 1–12. ACM Press (2000)
16. Kumar, R., Tomkins, A., Vee, E.: Connectivity Structure of Bipartite Graphs via the KNC-Plot. In: Proc. WSDM. pp. 129–138. ACM Press (2008)
17. Lemmerich, F., Becker, M., Atzmueller, M.: Generic Pattern Trees for Exhaustive Exceptional Model Mining. In: Proc. ECML-PKDD. Springer (2012)
18. Mitzlaff, F., Atzmueller, M., Benz, D., Hotho, A., Stumme, G.: Community Assessment using Evidence Networks. In: *Postproc. MSM/MUSE 2010*. Springer (2011)
19. Munoz-Gama, J., Carmona, J., van der Aalst, W.M.P.: Single-Entry Single-Exit Decomposed Conformance Checking. *Inf. Syst.* **46**, 102–122 (2014)
20. Newman, M.E., Girvan, M.: Finding and Evaluating Community Structure in Networks. *Phys Rev E Stat Nonlin Soft Matter Phys* **69**(2), 1–15 (2004)
21. Rifi, M., Hibti, M., Kanawati, R.: A Complex Network Analysis Approach for Risk Increase Factor Prediction in Nuclear Power Plants. In: Proc. International Conference on Complexity, Future Information Systems and Risk. pp. 23–30 (2018)
22. Rozinat, A., Aalst, W.: Conformance Checking of Processes Based on Monitoring Real Behavior. *Information Systems* **33**(1), 64–95 (2008)
23. Sani, M.F., van der Aalst, W.M.P., Bolt, A., García-Algarra, J.: Subgroup Discovery in Process Mining. In: *BIS. LNBIP*, vol. 288, pp. 237–252. Springer (2017)
24. Theorin, A., Bengtsson, K., Provost, J., Lieder, M., Johnsson, C., Lundholm, T., Lennartson, B.: An Event-Driven Manufacturing Information System Architecture for Industry 4.0. *Intl. Journal of Production Research* **55**(5), 1297–1311 (2017)
25. Vogel-Heuser, B., Schütz, D., Folmer, J.: Criteria-based Alarm Flood Pattern Recognition Using Historical Data from Automated Production Systems (aPS). *Mechatronics* **31** (2015)
26. Weyer, S., Schmitt, M., Ohmer, M., Gorecky, D.: Towards Industry 4.0-Standardization as the Crucial Challenge for Highly Modular, Multi-Vendor Production Systems. *Proc. IFAC* **48**(3), 579–584 (2015)
27. Wu, D., Greer, M.J., Rosen, D.W., Schaefer, D.: Cloud Manufacturing: Strategic Vision and State-Of-The-Art. *JMSY* **32**(4), 564–579 (2013)