



This is an electronic reprint of the original article. This reprint may differ from the original in pagination and typographic detail.

Sierla, Seppo; Azangoo, Mohammad; Vyatkin, Valeriy

Generating an industrial process graph from 3D pipe routing information

Published in: Proceedings of the 25th IEEE International Conference on Emerging Technologies and Factory Automation, ETFA 2020

DOI: 10.1109/ETFA46521.2020.9212175

Published: 01/09/2020

Document Version Peer-reviewed accepted author manuscript, also known as Final accepted manuscript or Post-print

Please cite the original version:

Sierla, S., Azangoo, M., & Vyatkin, V. (2020). Generating an industrial process graph from 3D pipe routing information. In *Proceedings of the 25th IEEE International Conference on Emerging Technologies and Factory Automation, ETFA 2020* (pp. 85-92). Article 9212175 (Proceedings IEEE International Conference on Emerging Technologies and Factory Automation). IEEE. https://doi.org/10.1109/ETFA46521.2020.9212175

This material is protected by copyright and other intellectual property rights, and duplication or sale of all or part of any of the repository collections is not permitted, except that material may be duplicated by you for your research use or educational purposes in electronic or print form. You must obtain permission for any other use. Electronic or print copies may not be offered, whether for sale or otherwise to anyone who is not an authorised user.

© 2020 IEEE. This is the author's version of an article that has been published by IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

Generating an industrial process graph from 3D pipe routing information

Seppo Sierla¹, Mohammad Azangoo¹, and Valeriy Vyatkin^{1, 2}

¹Department of Electrical Engineering and Automation, Aalto University, Espoo, Finland ²Department of Computer Science, Electrical and Space Engineering, Luleå University of Technology, Luleå, Sweden seppo.sierla@aalto.fi, mohammad.azangoo@aalto.fi, valeriy.vyatkin@aalto.fi

Abstract-The automatic generation of digital twins of industrial processes requires the integration of several sources of information. If the twin is expected to accurately capture thermo-hydraulic phenomena, dimensions of tanks and other process components as well as detailed pipe routing information is relevant. Such information is not comprehensively captured in **P&IDs** (Piping & Instrumentation Diagrams), but it is available from 3D CAD models. However, information about control loops is not available from 3D CAD models, but is available from P&IDs. Previous research has demonstrated the extraction of such information from machine-readable P&IDs and 3D CAD models and converting this information to graphs. Further research is expected on applying graph matching methods for integrating these separate graphs to a common graph-based data structure that captures all of the desired information. This common model could support further work to develop digital twins. A major obstacle to this is that the graphs that have currently been generated from P&IDs and 3D CAD models are at very different abstraction levels, so graph matching methods are not feasible. This article address this obstacle by building on previous work, in which graphs were generated from P&IDs and 3D CAD models. The contribution of this paper is several novel algorithms for preprocessing a 3D CAD generated graph, until it is at the same level of abstraction as a P&ID generated graph of the same industrial process. The algorithms are demonstrated in the context of a laboratory process.

Index Terms—industry 4.0, process industry, digitisation, automation, modelling and simulation, digital twins, 3D modeling, 3D CAD, digital plant, plant design.

I. INTRODUCTION

A major obstacle towards building an integrated digital plant model is that the relevant information is contained in different engineering tools. Two of the most important tools are P&ID (Process & Instrumentation Diagram) tools and 3D CADs. P&ID tools effectively capture instrumentation and control loops, while 3D CADs provide detailed information on pipe routing, component dimensions and elevations. Both kinds of information are essential for constructing process simulation models and digital twins. However, integrating information from these two sources is currently laborious and error prone. In industrial practice, it is not possible to assume that the same naming conventions have been used across tools [1], so there are no straightforward solutions for automatically identifying the same process components from the P&ID and the 3D model [2]. Some standardization efforts have been supported by leading tool vendors. The DEXPI working group

developed a specification (DEXPI) based on the ISO 15926 to define an open P&ID storage format, the Proteus XML schema [3], [4]. The PCF (Piping Component File) format for 3D isometrics is supported my major 3D process CAD tool vendors. Unfortunately, there has been no coordination between the development of DEXPI and PCF or any other such formats, so the underlying information models are very different. In order to build an integrated digital plant model, there is a need to:

- (A) Ignore irrelevant information such as graphical layout of the P&ID diagram.
- (B) Extract the relevant information to a neutral format.
- (C) Further process the information so that it is at a common level of abstraction, to enable further research on combining these information sources to an integrated digital plant model.

A solution for steps A. and B. was proposed in [2]. Proteus XML and PCF were used as source formats in step A, and graphs were used as the neutral format mentioned in step B. However, the graphs that were generated were at very different levels of abstraction and contained elements that could not be readily matched. The objective of this paper is to address step C by proposing algorithms to further process the graphs generated from PCF source information, which were more complex than the graphs generated from Proteus source information.

II. RELATED RESEARCH

A digital twin is a comprehensive physical and functional description of a component, product or system. The level of detail that is captured by the twin determines its potential for applications along the life cycle of its physical counterpart [5]. Due to high demand for digital twins, their automatic generation is interesting, since manual generation involves significant time and cost. The most two popular sources of information for automatic generation of digital twins are design phase documents, such as P&IDs, and 3D models. 3D point clouds obtained by scanning technologies have been used to provide visualizations for existing manufacturing simulations [6] and for object detection in factories [7] and other built environments [8]. However, these 3D models do not capture process components, their dimensions and connections, all

of which would be necessary source information for an executable digital twin that can be used for applications such as process optimization, forecasting and preventive maintenance; [9] presents a method for using 3D CAD source information to generate a twin for these purposes. Simulation models have been generated automatically from P&IDs for the purpose of testing of control software [10] and for Hardware-in-loop testing [11]. [12] generated low fidelity simulation models from high level specifications available at the preliminary design phase of the automation projects. Later these models were synchronized with the control and manufacturing operations management system to obtain a digital twin with improved fidelity. To generate a higher fidelity digital twin, several sources of information can be used. [13] identifies the need for integrating information from P&IDs and 3D scans, but does not address the problem of information extraction from the P&ID. Using different sources of information for automatic generation of digital twins introduces complexity. One of the main challenges in integrating multiple source of information is the need to match them. One approach to matching complex data sets is to convert them to a common level of abstraction. Graphs consisting of nodes and edges between the nodes have emerged as a simple yet powerful abstraction for this purpose in a diverse range of application areas. Several authors have studied social network analysis through graph matching. [14] propose a general-purpose approach, whereas specific applications exist for the purpose of expert matching [15], prevention of radicalization [16] and social network analysis [17]. Examples of other applications include identification of persons in surveillance systems [18], protein structure matching [19], crime matching [20], ontology matching [21] and remote sensing [22]. Applications of graph matching to industrial process design documents are limited. [1] created manually graph representations of several pulp and paper process plants and used graph matching to assess the reuse potential of the plant designs. [23] proposes an algorithm for matching the 2D and 3D graph representations of a process plant, but does not propose algorithms for creating the graph representations from the available source information. No work has applied graph matching to industrial process plant design documents that include pipe routing information, which is necessary for the generation of high fidelity thermohydraulic simulation models [9]. PCFs capture this information in detail, and for that reason any straightforward approach for generating a graph from a PCF, as in our previous work [2], will result in a graph that has much more detail than a graph that is generated from a P&ID. This is the motivation for the research in this paper, which aims to process a PCF generated graph to the same level of abstraction as a P&ID generated graph. This will make it possible for further research to integrate P&ID and 3D CAD information towards automatic generation of a digital twin. Such a generation method will surpass existing methods for automatic generation of digital twins.

III. CASE STUDY

A. Case process

The case study is a thermo-hydraulic water process [9], [24]–[26]. The process has been modelled in the Intergraph Smart 3D tool (Fig. 1). The model includes 9 pipelines, each of which has been exported as a PCF file.



Fig. 1. 3D CAD model of the case process [2]

B. A motivating example

Fig. 2 shows a graph that was generated by applying the graph generation methodology in [2] to one of the 9 PCF files exported from the model in Fig. 1 The implemented graph generator automatically outputs the graph as lists of nodes and edges in .csv files, from which the visual representation in Fig. 1 was created manually. The same approach applies to all graphs presented in this paper. Some of the nodes and edges in Fig. 2 are color coded, and the corresponding physical process components are indicated within ellipses of the same color on photographs of the physical process in Fig. 3 and Fig. 4.

IV. METHODOLOGY AND RESULTS

The structure of this section is as follows. Section IV-A elaborates the objective stated in section I as section VI specific issues to be solved. The remaining subsections present graph processing algorithms for solving the issues. The algorithms form a pipeline, with the output graph of one algorithm being the input graph of the next algorithm. It will be much easier for the reader to follow the presentation if the output and input graphs are presented in conjunction with the algorithms. This is the rationale for structuring this section as a combined "methodology and results" section.

A. General issues in graphs generated from PCF files

The goal of this paper stated in section I is to further process graphs such as the one in Fig. 2, so that they are at the same level of abstraction as graphs generated from P&ID



Fig. 2. Graph generated from one of the 9 PCF files exported from the 3D CAD model in Fig. 1



Fig. 3. Color-coded mapping of a part of the graph in Fig. 2 to the photograph of the physical process

diagrams with methods such as the one presented in [2]. The following observations in Fig. 2 are examples general issues to be expected when generating graphs from PCF files:

ISSUE1: Since every elbow and weld is captured, sections of a pipeline that do not have any tees, nozzles or instruments are represented with several nodes connected in series.

ISSUE2: Valves are represented with edges instead of nodes.

ISSUE3: Tees are not explicitly marked as a special type of node, but need to be identified as nodes with 3 edges.



Fig. 4. Color-coded mapping of a part of the graph in Fig. 2 to the photograph of the physical process

ISSUE4: The node labelled "Pipeline 2" is a pipeline connector in the PCF file referring to a corresponding connector in another PCF file. PCF files may have some redundancies, especially related to pipeline connectors being listed twice with slightly different 3D coordinates. Figure 2 indicates the redundant parts as black nodes.

ISSUE5: The pipeline connectors (nodes labelled "Pipeline 2") are used when a physical pipeline is split somewhere along the way between two PCF files. (The location of this split in our motivating example is indicated with the green ellipse in Fig. 2.)

ISSUE6: Process equipment such as tanks and pipes are not explicitly captured in the PCF, which describes pipelines ending at nozzles of such equipment All of these issues result in the graph having additional details and features, which have no counterparts in the P&ID generated graphs. Thus, these additional details and features are undesirable from the perspective of the research goal of this paper. Thus a methodology for overcoming them is presented in the following subsections.

B. Graph simplification algorithm

A graph simplification algorithm for addressing ISSUE1, ISSUE2 and ISSUE3 is presented in Fig. 5. The result of applying this algorithm to the graph in Fig. 2 is presented in Fig. 6. The color coding indicates the steps of the algorithm that generate the different types of nodes. The node label PLX - Y is shorthand for an END - CONNECTION - PIPELINE element in a PCF file with PIPELINE - REFERENCE PLX, referencing another pipeline with PIPELINE - REFERENCE PLX.



Fig. 5. Graph simplification algorithm



Fig. 6. The result of applying the algorithm in Fig. 5 to the graph in Fig. 2 to the photograph of the physical process

Fig. 7 shows the result of applying the algorithm in Fig. 5 to each of the 9 PCF files separately. Thus, Fig. 7 is 9 separate graphs. Violet circles have been added manually to indicate which nozzles correspond to the same component. Brown dashed lines have been manually added to indicate matching end connections between PCF files.

C. Redundant end connection removal algorithm

The solution to ISSUE4 is the removal of the redundant end connections and the tees that they are connected to. These are

the black nodes in Fig. 7. The remaining edges of the removed tee also need to be fixed. For example, after removing tee 294 in Fig. 7, there should be one edge from valve 343 to end connection PL2-5. An algorithm for this purpose in presented in Fig. 8. The result of applying this algorithm to the graph in Fig. 7 is presented in Fig. 9.

D. Graph fragments unification algorithm

The solution to ISSUE5 is to establish the connections that are indicated with manually drawn dashed brown lines in Fig. 9. It is not enough to just add edges corresponding to these lines. The end connection nodes should be deleted, since they will have no counterpart in the graph generated from a P&ID. Thus, there should be an edge between the nodes that are connected to the end connection nodes. For example in Fig. 9, the nodes PL5-2 and PL2-5 should be deleted along with their edges, and an edge from tee 602 to valve 343 should be added. Fig. 10 presents the algorithm for this purpose. It is notable that although the case study follows the naming convention of pipeline references starting with 'PL', the algorithm does not make any assumptions on naming conventions and just uses the variable REF for this purpose. Fig. 11 shows the result of applying the algorithm in Fig. 10 to the graph in Fig. 9.

E. Equipment node creation algorithm

The solution to ISSUE6 is to create the equipment nodes corresponding to the violet circles that have been manually added to Fig. 11. The nozzles have been created from the tag



Fig. 7. The result of executing the algorithm in Fig. 5 separately for all of the PCF files.



Fig. 8. Algorithm for removing redundant end connections



Fig. 9. The result of applying the algorithm in Fig. 8 to the graph in Fig. 7



Fig. 10. Graph fragments unification algorithm



Fig. 11. The result of applying the algorithm in Fig. 10 to the graph in Fig. 9

on the CONNECTION - REFERENCE line under an END - CONNECTION - EQUIPMENT element of a PCF file. The tags are of the following form:

< equipmentname > / < nozzlename >

The algorithm in Fig. 12 creates an equipment node from each unique $\langle equipmentname \rangle$ found from these tags. For each tag, it creates an edge from the nozzle to this equipment node. The result of applying this algorithm to the graph in Fig. 11 is shown in Fig. 13.



Fig. 12. Equipment node creation algorithm



Fig. 13. The result of applying the algorithm in Fig. 12 to the graph in Fig. 11

V. DISCUSSION

Fig. 14 shows a graph generated by applying the algorithm presented in [2] to a P&ID of the case process exported in the Proteus XML format. It is notable that this P&ID does not have all of the pipelines, so that graph is simpler, and the significance of this for further research is elaborated in the concluding section. By comparing Fig. 13 and Fig. 14, it can be observed that they are at the same level of abstraction, and that the smaller number of tees and valves in the latter is due to the absence of some of the pipelines in the original P&ID. Otherwise, the graphs are comparable. One notable difference is that the Proteus XML specifies the type of process component, such as the type of tank, vessel or pump. As was explained in section IV-E, this information cannot be found from PCF files, so the algorithm in Fig. 12 only creates nodes of type 'equipment'. Thus for graph matching purposes in further research, it may be beneficial to simplify the P&ID generated graph so that the type of all pump, tank and vessel nodes is changed to 'equipment'.

VI. CONCLUSION AND FURTHER RESEARCH

The objective of the paper was stated in section I and elaborated into 6 issues in section IV. A solution based on the PCF standard was presented in sections IV-A to IV-E and an application to a case was presented. Based on the case study, all of the issues were satisfactorily addressed. Further research for larger industrial cases can further test the scalability of the method and its robustness to glitches in the source information. The motivation for this paper has been to enable integration of P&ID and 3D CAD information to obtain an integrated graph model, so the most immediate need for further research is to perform matching between graphs such as the ones in Fig. 13 and Fig. 14 to identify the corresponding parts of the P&ID and 3D CAD models. The algorithms presented in this paper generate the node and edge listings that are the source information to a tool that implements a graph matching algorithm. Potential matching algorithms have been discussed in [1]. Finally, it is notable that our P&ID was an early design phase artifact while our



Fig. 14. Graph generated from a P&ID of the case process

3D CAD model is a detailed design phase artifact. This explains the major differences between Fig. 13 and Fig. 14. For broad industrial applicability, it is desirable to be able to match artifacts developed in different stages of the lifecycle of the plant, since it is not possible to generally assume that digital artifacts are available from all lifecycle phases. This is especially relevant for industrial process plants which have already been in operation for several decades. Additional detail and changes are introduced throughout plant design and commissioning, but also later in modernization, refurbishment and retrofit operations. Thus, one specific question for further research is to assess the possibility to match artifacts from different lifecycle phases. An example of how this question would apply to our case study would be trying to correctly match the violet nodes in Fig. 13 and Fig. 14.

VII. ACKNOWLEDGEMENTS

This work was partially supported by Business Finland project SEED (grant 4153/31/2019.)

REFERENCES

- M. Rantala, H. Niemistö, T, Karhela, S. Sierla and V. Vyatkin, "Applying graph matching techniques to enhance reuse of plant design information," Computers in Industry, Volume 107, Pages 81-98, 2019.
- [2] S. Sierla, M. Azangoo, A. Fay, V. Vyatkin and N. Papakonstantinou, "Integrating 2D and 3D Digital Plant Information Towards Automatic Generation of Digital Twins," accepted for 29th IEEE International Symposium on Industrial Electronics (ISIE), Delft, The Netherlands, June 2020.
- [3] DEXPI. "Data Exchange in the Process Industry (DEXPI)." 2019, from http://www.dexpi.org/.
- [4] Fiatech. "ISO 15926 Information Models and Proteus Mappings (IIMM)
 Proteus schema." Retrieved 2017, from http://fiatech.org/informationmanagement/projects/1161-iso-15926-information-models-and-proteusmappings-iimm.

- [5] S. Boschert and R. Rosen, "Digital Twin—The Simulation Aspect. In: Hehenberger P., Bradley D. (eds) Mechatronic Futures." Springer, Cham, pp 59-74, 2016.
- [6] M. Minos-Stensrud, O. H. Haakstad, O. Sakseid, B. Westby and A. Alcocer, "Towards Automated 3D reconstruction in SME factories and Digital Twin Model generation," 2018 18th International Conference on Control, Automation and Systems (ICCAS), Daegwallyeong, 2018, pp. 1777-1781.
- [7] M. Sommer, J. Stjepandic, S. Stobrawa and M. V. Soden, Moritz, "Automatic Generation of Digital Twin based on Scanning and Object Recognition," ASIM Fachtagung Simulation in Poduktion und Logistik 2019, Chemnitz, 2019.
- [8] V. Stojanovic, M. Trapp, R. Richter, B. Hagedorn and J. Döllner, "Towards The Generation of Digital Twins for Facility Management Based on 3D Point Clouds, ARCOM 2018: 34th Annual Conference, Belfast, UK, 2018.
- [9] G. S. Martínez, S. Sierla, T. Karhela and V. Vyatkin, "Automatic Generation of a Simulation-Based Digital Twin of an Industrial Process Plant," IECON 2018 - 44th Annual Conference of the IEEE Industrial Electronics Society, Washington, DC, pp. 3084-3089, 2018.
- [10] M. Barth, A. Fay, "Automated generation of simulation models for control code tests," Control Engineering Practice, PP. 218-230 Volume 21, Issue 2, 2013.
- [11] M. Barth, M. Strube, A. Fay, P. Weber and J. Greifeneder, "Objectoriented engineering data exchange as a base for automatic generation of simulation models," 2009 35th Annual Conference of IEEE Industrial Electronics, Porto, 2009, pp. 2465-2470.
- [12] J. G. Campos, J. S. López, J. I. A. Quiroga and A. M. E. Seoane, "Automatic generation of digital twin industrial system from a high level specification," Procedia Manufacturing, Volume 38, PP. 1095-1102, 2019.
- [13] H. Son, C. Kim and C. Kim, "3D reconstruction of as-built industrial instrumentation models from laser-scan data and a 3D CAD database based on prior knowledge," Automation in Construction, vol. 49, part B, pp. 193-200, 2015.
- [14] C. Davalas, D. Michail and I. Varlamis, "Graph matching on social networks without any side information," 2019 IEEE International Conference on Big Data (Big Data), Los Angeles, CA, USA, pp. 1060-1065, 2019.
- [15] W. fei Fan, X. Wang and Y. Wu, "ExpFinder: Finding experts by graph pattern matching," 2013 IEEE 29th International Conference on Data Engineering (ICDE), Brisbane, QLD, pp. 1316-1319, 2013.
- [16] B. W. K. Hung, A. P. Jayasumana and V. W. Bandara, "Finding Emergent Patterns of Behaviors in Dynamic Heterogeneous Social Networks," in IEEE Transactions on Computational Social Systems, vol. 6, no. 5, pp. 1007-1019, Oct. 2019.
- [17] G. Liu et al., "MCS-GPM: Multi-Constrained Simulation Based Graph Pattern Matching in Contextual Social Graphs," in IEEE Transactions on Knowledge and Data Engineering, vol. 30, no. 6, pp. 1050-1064, 1 June 2018.
- [18] Y. Huang, H. Sheng and Z. Xiong, "Person re-identification based on hierarchical bipartite graph matching," 2016 IEEE International Conference on Image Processing (ICIP), Phoenix, AZ, pp. 4255-4259, 2016.
- [19] F. Othman, I. Umar, R. Abdullah and A. Rathi, "Parallel Bipartite Graph Algorithm for Protein Structure Matching Using OpenMP," 2010 Second International Conference on Computer Research and Development, Kuala Lumpur, pp. 25-29, 2010.
- [20] N. Qazi and B. L. W. Wong, "Behavioural & Tempo-Spatial Knowledge Graph for Crime Matching through Graph Theory," 2017 European Intelligence and Security Informatics Conference (EISIC), Athens, pp. 143-146, 2017.
- [21] A. Sharma, "Ontology Matching Using Weighted Graphs," 2006 1st International Conference on Digital Information Management, Bangalore, pp. 121-124, 2007.
- [22] B. Banerjee, F. Bovolo, A. Bhattacharya, L. Bruzzone, S. Chaudhuri and K. M. Buddhiraju, "A Novel Graph-Matching-Based Approach for Domain Adaptation in Classification of Remote Sensing Image Pair," in IEEE Transactions on Geoscience and Remote Sensing, vol. 53, no. 7, pp. 4045-4062, July 2015.
- [23] R. Wen, W. Tang and Z. Su, "Topology based 2D engineering drawing and 3D model matching for process plant," Graphical Models, V. 92, PP. 1-15, 2017.

- [24] G. S. Martínez, T. A. Karhela, R. J. Ruusu, S. A. Sierla and V. Vyatkin, "An Integrated Implementation Methodology of a Lifecycle-Wide Tracking Simulation Architecture," in IEEE Access, vol. 6, pp. 15391-15407, 2018.
- [25] G. S. Martínez, S. A. Sierla, T. A. Karhela, J. Lappalainen and V. Vyatkin, "Automatic Generation of a High-Fidelity Dynamic Thermal-Hydraulic Process Simulation Model From a 3D Plant Model," in IEEE Access, vol. 6, pp. 45217-45232, 2018.
- [26] S. A. Sierla, T. A. Karhela and V. Vyatkin, "Automatic Generation of Pipelines Into a 3D Industrial Process Model," in IEEE Access, vol. 5, pp. 26591-26603, 2017.