GRAPH MATCHING FOR RECONCILING SCADA AND GIS OF A DISTRIBUTION NETWORK

Bertrand CORNÉLUSSE
ULg - Belgium
bertrand.cornelusse@ulg.ac.be

Amandine LEROUX
RESA - Belgium
amandine.leroux@resa.be

Mevludin GLAVIC
ULg - Belgium
mevludin.glavic@ulg.ac.be

Damien ERNST
ULg - Belgium
dernst@ulg.ac.be

ABSTRACT

This article deals with the problem of automatically establishing a correspondence between two databases populated independently over the years by a distribution company, for instance a SCADA system and a geographical information system. This problem is abstracted as a graph matching problem, well known in the combinatorial optimisation community. It is then casted as an integer quadratic program. An idea of achievable results on a real system is provided, and needs for approximation or decomposition algorithms are discussed.

INTRODUCTION

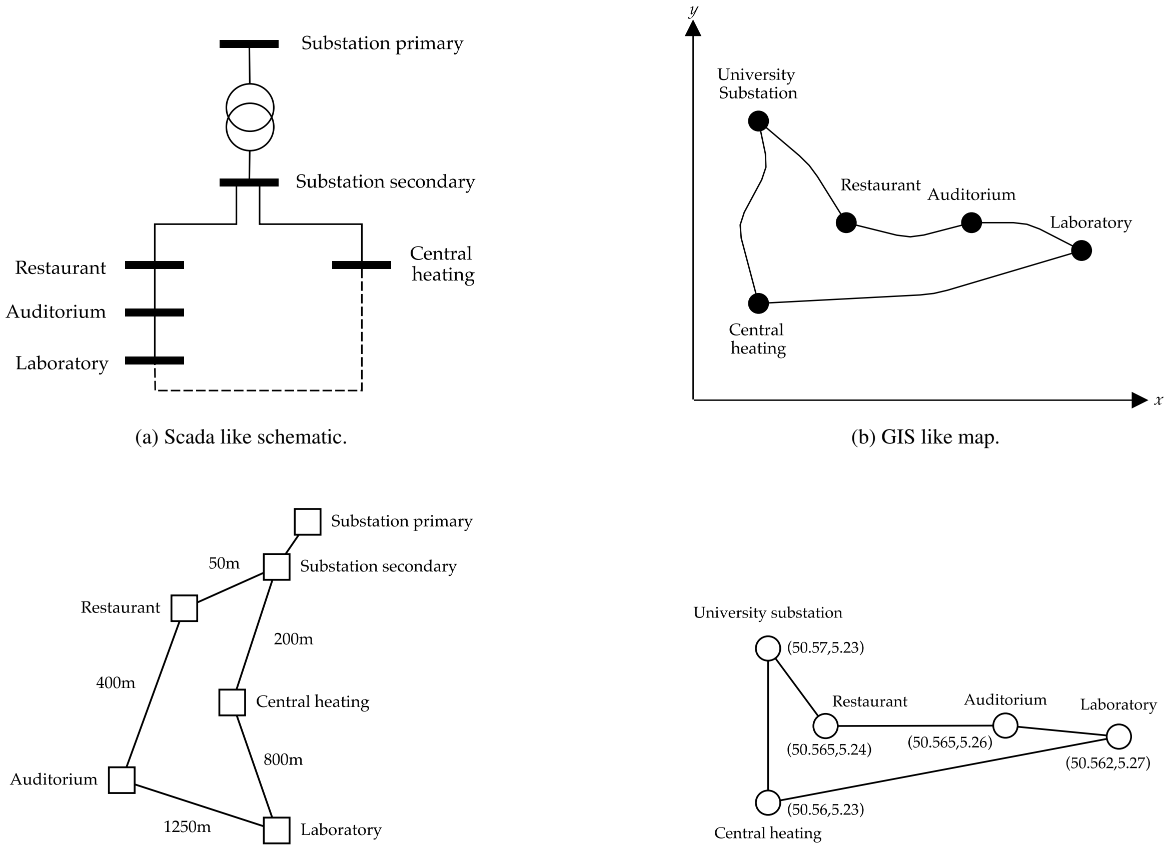
Many electricity distribution companies around the world maintain a SCADA type database and a Geographical Information System (GIS) since information they contain is essential for efficient system planning, operation, and control [1]. For historical reasons these two databases are maintained independently, although they reference nearly the same physical elements of the medium voltage distribution network. The SCADA database contains topological (connectivity) and electrical information, i.e. impedance of lines, the model of the transformers, etc. (cf. Figure 1a). The GIS database contains the coordinates of buses (cf. Figure 1b), the type and name of devices attached to them, and some line characteristics (their path, sometimes composed of several chunks, and cable properties). Both databases contain labels and numeric identifiers for a subset of buses. There is little information available to make a direct mapping between elements of the two databases. Furthermore there might be errors and missing data. This work aims to establish this mapping, i.e. to reconcile the two databases in an automated way, since doing this work fully manually would be tedious. A solution to these problems would be applications of standards for data exchange formats (CIM/XML) [2]. However, the evolution of these standards in practical use revealed to be much slower than expected and there are needs for other automated solutions anyway.

In this paper, we show how this problem can be cast and solved as a graph matching problem [3], and we discuss the peculiarities of this application with respect to usual graph matching applications. The graph matching problem has received a lot of attention in the pattern recognition community, for instance for fingerprint comparison [4]. However to the best of our knowledge, it has never been applied neither in the field of power systems nor to such a large scale application (several thousands vertices and edges). The graph matching problem is then formulated as a particular case of a quadratic assignment problem, which is a long studied problem in the Operations Research community. We provide results of the approach on the full MV network of RESA, one DSO of the province of Liège, Belgium. We think this contribution is of general interest especially in the era of smart grids, where putting together information from several information management systems could enable more efficient use of the grid. The application studied in this work may for instance be used to show the real-time electrical state of the network represented on a geographical map, on mobile devices. Another application is to filter out inconsistent values in both systems, to have a better representation of the system for operation, maintenance, investment planning, or asset valuation.

1 FORMULATION AS A GRAPH MATCHING PROBLEM

Each database can be represented in an abstract way as an *attributed undirected graph*, as shown in Figure 1. An attributed undirected graph $\mathscr{G}$ is made of a set of vertices V connected by edges gathered in a set E. Edges and vertices have attached information, or attributes, depending on the source database, collected in a structure A. Let $\mathscr{S} = (V_S, E_S, A_S)$ be the SCADA graph and $\mathscr{G} = (V_G, E_G, A_G)$ be the GIS graph. These two graphs do not necessarily have the same number of vertices and edges, as explained below.

The weighted graph matching problem consists in mapping vertices of V_S to vertices of V_G and edges of E_S to edges of E_G, taking into account the similarity between the attributes of the vertices and edges, and the topology of the two graphs. The similarity can be established based on the attributes attached to edges and vertices. Let us consider the two graphs of Figure 2, which are among the simplest we can imagine. Without taking into account the topology, there are $4! = 24$ solutions, since there are 4 ways to map vertex 1, then 3 remaining candidate vertices for vertex 2 and finally 2 candidate vertices for vertex 3. However, if the similarity measure between vertices is good, the number of optimal solutions is probably much smaller than the number of feasible solutions. When the topology is accounted for some of these solutions are no more feasible. For the two graphs of Figure 2, accounting for



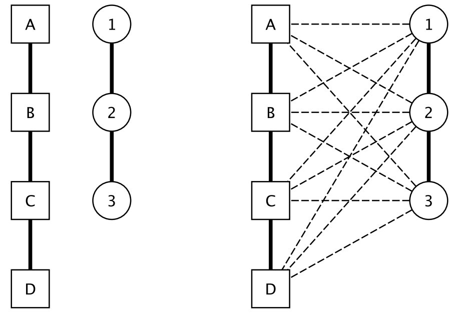
(a) Scada like schematic.

(b) GIS like map.

(c) Graph $\mathscr{S}$ corresponding to Figure 1a.

(d) Graph $\mathscr{G}$ corresponding to Figure 1b.

Figure 1: Distribution system representations.



(a) Two graphs.

(b) Dashed lignes represent possible assignments of vertices without taking into account the topology.

Figure 2: Graph matching example.

the topology leaves only 4 possibilities, since when vertex 1 is assigned to vertex A, vertex 2 must be assigned to vertex B and vertex 3 to vertex C. The other possible matches are $\{(1,B),(2,C),(3,D)\}$, $\{(1,C),(2,B),(3,A)\}$, and $\{(1,D),(2,C),(3,B)\}$. Hence, even partial information on similarity between edges and between vertices could help finding out the actual correspondence.

In this particular application, a vertex of $\mathscr{S}$ represents an electrical bus in the SCADA system. Ideally, vertices extracted from the GIS should represent the same buses as those extracted from the SCADA. However, as illustrated in Figures 1a and 1b, a single GIS vertex may represent two electrical buses, such as the two sides of the transformer, because they are geographically very close. Imposing a strict topological correspondence is thus likely to yield poor results, since many such divergences of representation and encoding errors are by nature present in both systems, especially the GIS system, in which connectivity is not the main focus. This also suggests that assigning several vertices of one graph to one vertex of the other graph should be allowed, especially several vertices of $\mathscr{S}$ to one vertex of $\mathscr{G}$. Similarly, $\mathscr{G}$ is likely to contain vertices that do not correspond to electrical buses in the SCADA, for instance physical junctions between cables that must be tracked in the GIS. In the sequel, we assume that a vertex of V_G can be assigned to at most one vertex of V_S, since an electrical bus cannot hold two physical positions, and one physical position may contain no SCADA bus. However, the number of vertices of V_S that could be assigned to a vertex V_G is not known a priori and is function of the particular vertex of V_G. As explained below, we make some prior assumptions based on vertex attributes.

Finally, although the GIS graph is not complete, i.e. does not contain an edge between every pair of vertices, it is possible to compute the geographical distance between any pair of vertices, and filter out inconsistent matches a posteriori.

In the next section we show how this graph matching problem can be solved or approximated using Integer Programming.

2 ALGORITHMS

2.1 Linear assignment problem (LAP) formulation

A first possibility is to look for a correspondence between vertices only, without taking into account the topology of the graph, i.e. the edges, nor the vertices position in the GIS. This is thus a relaxation of the actual problem. If there is enough information in the attributes of vertices, a solution to this problem may generate a solution having edges corresponding with a high accuracy. The problem is then a (linear) assignment problem which can be formulated as the integer program (1)–(4).

$$\max_{x \in X} \quad \sum_{i \in V_S} \sum_{j \in V_G} x_{i,j} S^V_{i,j} \tag{1}$$

$$s.t. \quad \sum_{j \in V_G} x_{i,j} \leq 1, \ \forall i \in V_S \tag{2}$$

$$\sum_{i \in V_S} x_{i,j} \leq M_j, \ \forall j \in V_G \tag{3}$$

$$x_{i,j} \in \{0,1\}, \ \forall i \in V_S, \ \forall j \in V_G \tag{4}$$

Variables and constraints. The variable $x_{i,j}$ indicates whether vertex $i \in V_S$ is associated to vertex $j \in V_G$. All variables are collected in the vector of variables x belonging to the set $X = \{0,1\}^{|V_S||V_G|}$. The set of constraints (2) indicates that a SCADA bus can be mapped to at most one GIS vertex. The set of constraints (2) states that a GIS vertex j can contain at most M_j SCADA buses. This parameter is estimated beforehand based on some prior knowledge. It should not be set to a too high value, or the solution will tend to map too many SCADA buses at some GIS vertices although they share little similarity. In our application this parameter is estimated according to a particular feature of RESA databases.

Objective and similarity measure. We must define a similarity measure between a vertex of the SCADA and a vertex of the GIS in order to map the most similar vertices together. This similarity measure is denoted by $S^V_{i,j}$. A specific measure was used in this application. $S^V_{ij} = 0$ if SCADA vertex i and GIS vertex j do not belong to the same township, the only geographical information available in the SCADA for this case. Else, S^V_{ij} is function of the similarity between the labels of the vertices and of a partial numbering that is available for some vertices, and which is assumed more important than the vertex label.

Computational complexity. The matrix of constraints is totally unimodular. Finding an optimal basic feasible solution of a continuous relaxation of this problem provides a solution to the integer program. This program can thus be solved efficiently even if there are many possible matches, i.e. the similarity matrix S^V contains a lot of non-zero elements.

2.2 Quadratic assignment problem formulation

Once SCADA vertices are mapped onto the GIS, we know their position. We can thus determine whether the distance between them is compliant with the length of the edge linking them in the SCADA, if any. This is not expressed in the mathematical formulation (1)–(4). Several options are available to take into account matching between edges. The application under consideration is particularly noisy: it is not because an edge exists in SCADA that the same edge exists in the GIS, and *vice versa*. Edge information is thus introduced through a penalisation of the objective function, rather than through hard constraints. Here the sole difference with respect to formulation (1)–(4) is the addition of

$$\sum_{(i,l) \in E_L} \sum_{(j,k) \in E_G} x_{i,j} x_{l,k} S^E_{(i,l),(j,k)} \tag{5}$$

to the objective. Note that this is a sum of quadratic expressions. We do not introduce any new optimisation variable. Any time a SCADA bus i is mapped to a GIS vertex j and a SCADA bus l is mapped to a GIS vertex k, if an edge (i,l) exists in the SCADA and an edge (j,k) exists in the GIS, a constant $S^E_{(i,l),(j,k)}$ measuring the similarity of the edges is added to the objective value. The problem thus becomes a quadratic assignment problem (QAP).

Computational complexity. This problem is in theory much harder than the linear assignment relaxation. It is manageable to solve it only if the matrix S^E is sufficiently sparse. It is thus not manageable to consider matches between a SCADA edge and two vertices in the GIS distant of approximately the SCADA edge length if no corresponding edge exists in the GIS. A number of preprocessing actions are performed to carry in the problem prior information and to reduce the size of the problem. The size of

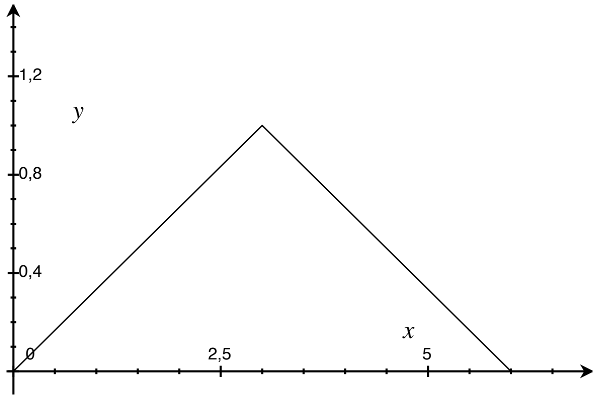


Figure 3: Illustration of the similarity measure $S^E_{(i,l),(j,k)}$ as a function of the length of (i,l) assuming the length of (j,k) is 3.

the problem must be reduced, else it could contain billions of variables and be unsolvable with current computers and state of the art algorithms. However it must not be too drastically preprocessed, else we would lose the interest of solving the optimisation problem to optimality.

Similarity between edges. We must define a similarity measure between an edge of the SCADA and an edge of the GIS in order to map the most similar vertices together. This similarity measure is denoted by $S^E_{(i,l),(j,k)}$. In this application we have used the following measure:

- $S^E_{(i,l),(j,k)} = 0$ if SCADA and GIS edges have end vertices that do not belong to the same township. Both orientations of the edges are taken into account.
- else
$$S^E_{(i,l),(j,k)} = \max\left(0, 1 - \frac{|x-y|}{x}\right)$$
where x is the length of edge (j,k) in the GIS and y is the length of edge (i,l) in the SCADA.

The similarity measure is symmetrical and takes value in $[0,1]$. It is equal to 100% when both edges have exactly the same length. It is illustrated in Figure 3.

3 RESULTS

The proposed methodology is applied to the data of RESA. The results are reported in Table 1 for the two methods described above. The "SCADA edges mapped" line represents the number of pairs of SCADA vertices matched connected by an edge in the SCADA. There are 11074 edges in the SCADA system and 12288 in the GIS. In each database the number of vertices is approximately equal to the number of edges, since both graphs are almost radial.

Table 1: Results.

	LAP	QAP
Vertex matches	4754	6700
Edge matches	163	907
SCADA edges mapped	3581	5724

LAP. With this formulation the number of SCADA buses a priori mapped to GIS vertices is 5221 (all variables x_{ij} such that $S^V_{ij} = 0$ are not created). We thus already know that it will not be possible to map all SCADA buses to GIS vertices by using solely the attributes of the vertices. The correspondences identified seem pretty good. However there are also some SCADA buses that are probably erroneously attached to the same vertex. Repeating the same simulation but filtering on SCADA edge length yields 2595 Vertex matches, 79 Edge matches: 79, and 583 SCADA Edges mapped. This is a bad news since filtering on length removes a lot of vertex matches although those matches seem good. It tends to mean that SCADA edge length information is not so reliable, which can be verified on some particular cases.

QAP. As foreseen, this approach turned out to be very impractical from a computational point of view. It takes more than 10 minutes to obtain a feasible solution restricting the possible matches to edges with a similarity above 0.95. Avenues of improvement are discussed in the conclusion.

4 CONCLUSION

It is very difficult to assess the solution quality because both databases contain errors. Apart from the computational issues, it is very important to have a good similarity measure for vertices, and more importantly for edges. As mentioned in [3], "The main research focus in pattern recognition is about designing efficient algorithms for approximately solving the quadratic assignment problem since it is NP-hard. In this paper, we turn our attention to a different question: how to estimate compatibility functions such that the solution of the resulting graph matching problem best matches the expected solution that a human would manually provide." This work allowed to obtain an approximation of the true matching and to highlight some issues with the databases that must be improved in order to move towards a solution of better quality. This work allows preprocessing the matching work, which can then be used to show the SCADA topology on a map to evaluate the quality of the results. This would be a good tool to resolve issues in the databases. Indeed, the information on the length of segments in the SCADA seems very imperfect, many lines having a very small length, and some vertices in GIS are wrongly positioned. These issues prevent the algorithm from finding more edge matches. With more insight into the database content, some clean up within the GIS database to filter out incoherent vertex positions, discussions with experts from GIS and SCADA at RESA, and little additional work, results could be largely improved.

Regarding the tractability of this problem from a computational point of view, insight can be gained from previous work in the pattern recognition field. An attempt to solve a similar problem using Linear Programming was performed in [5]. For two graphs containing the same number of vertices, the authors define an optimisation problem over permutation matrices. A permutation matrix contains only ones and zeros and its lines and columns sum to one. Maximising a matching based on edge similarity is achieved by minimising the norm 1 of the difference of the adjacency matrix of one graph and the permuted adjacency matrix of the other graph. This formulation does not encode similarity between vertices, but it can be readily added. The advantage of this formulation is to turn the problem into a mixed integer but linear problem. Several other papers contain promising ideas to improve the tractability of this problem formulation: Lagrangian decomposition [6], the graduated assignment method which solves a series of nonconvex but continuous problems [7], spectral methods [8], or formulation based on semi-definite programming [9].

ACKNOWLEDGMENTS

This research is supported by the public service of Wallonia, Department of Energy and Sustainable Building within the framework of the GREDOR project.

REFERENCES

[1] T. Taylor and H. Kazemzadeh, “Integrating SCADA/DMS/OMS: increasing distribution operations efficiency,” *Electric Energy T & D Magazine*, pp. 31–34, 2009.

[2] C.-W. Ten, E. Wuergler, H.-J. Diehl, and H. B. Gooi, “Extraction of geospatial topology and graphics for distribution automation framework,” *Power Systems, IEEE Transactions on*, vol. 23, no. 4, pp. 1776–1782, 2008.

[3] T. S. Caetano, J. J. McAuley, L. Cheng, Q. V. Le, and A. J. Smola, “Learning graph matching,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 31, no. 6, pp. 1048–1058, 2009.

[4] D. Maltoni, D. Maio, A. K. Jain, and S. Prabhakar, *Handbook of fingerprint recognition*. springer, 2009.

[5] H. Almohamad and S. O. Duffuaa, “A linear programming approach for the weighted graph matching problem,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 15, no. 5, pp. 522–525, 1993.

[6] A. Rangarajan and B. Mjolsness, “A lagrangian relaxation network for graph matching,” *Neural Networks, IEEE Transactions on*, vol. 7, no. 6, pp. 1365–1381, 1996.

[7] S. Gold and A. Rangarajan, “A graduated assignment algorithm for graph matching,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 18, no. 4, pp. 377–388, 1996.

[8] T. Cour, P. Srinivasan, and J. Shi, “Balanced graph matching,” *Advances in Neural Information Processing Systems*, vol. 19, p. 313, 2007.

[9] C. Schellewald and C. Schnörr, “Probabilistic subgraph matching based on convex relaxation,” in *Energy minimization methods in computer vision and pattern recognition*, Springer, 2005, pp. 171–186.