Geoinformatica (2008) 12:435–453
DOI 10.1007/s10707-007-0040-1

Matching Networks with Different Levels of Detail

Sébastien Mustière · Thomas Devogele

Received: 27 February 2007 / Revised: 20 September 2007 /
Accepted: 4 October 2007 / Published online: 20 October 2007

Abstract This paper deals with the issue of automatically matching networks with different levels of details. We first present why this issue is complex through an analysis of the differences that can be encountered between networks. We also present different criteria, tools and approaches used for network matching. We then propose a matching process, named *NetMatcher*. This process is a several steps process looking for potential candidates and then analysing them in order to determine the final results. It relies on the comparison of geometrical, attributive, and topological properties of objects. It determines one-to-many links between networks: in particular a node of the less detailed network can be matched to several arcs and nodes forming a complex junction in the most detailed network. An important strength of the process is to self-evaluate its results through the comparison of topological organisation of networks. This paves the way to an interactive editing of the results. The *NetMatcher* process has been intensively tested on a wide range of actual datasets, thus highlighting its effectiveness as well as its limits.

Keywords data matching · integration · network · network matching · conflation · level of detail

1 Introduction

Nowadays, more and more geographic data are captured, updated and manipulated. An increasing need is thus appearing: integrating possibly heterogeneous databases. Integrating shall here be understood in the very wide sense of making explicit the relations existing between the data. A first need concerns the combination of information. If databases are integrated, data users can perform efficient geographic analysis taking into account several points of view [27],

S. Mustière (✉)
IGN/COGIT Lab, 2 Av. Pasteur, 94160 St-Mandé, France
e-mail: sebastien.mustiere@ign.fr

T. Devogele
Naval Academy Research Institute (IRENav), Lanvéoc, BP 600, 29240 Brest Naval, France
e-mail: devogele@ecole-navale.fr

and data producers can transfer information from one database to another one [12]. Another need relates to the propagation of updates from one database to another one: if data are linked together, the propagation of updates is facilitated [2]. This concerns data producers that need to simultaneously maintain several databases, but also data users that need to easily update their own thematic data from updates provided by reference data providers. A last need relates to the comparison of different datasets. If databases are linked together, data producers can compare them and thus perform some quality analysis and detect inconsistencies [25].

Database integration requires some works at the schema level and some works at the data level, especially in the context of geographic data [10]. Schema integration is necessary to identify corresponding classes in the schemas [10], [21], [26], [28]. Then data matching is necessary to actually identify homologous objects in the data [9], [12], [14], [30], [31]. Data matching is an easy process when a common identifier is defined for the data. Unfortunately, when such identifiers do not exist, data matching is often a complex problem that must rely on properties of objects. This is the case for geographic data: matching them requires comparing their geometrical as well as non-spatial properties.

A lot of geographic information is modelled as networks (roads, railways, rivers…). They are usually complex and associated with different properties according to the underlying point of view of the database (Fig. 1): a traffic manager may model the speed limits of roads while a maintenance department may model their physical composition for example. Matching networks received some attention in the literature. Some approaches concentrate on matching networks coming from the same database, but at different dates, in order to detect updates [2], [11]. Other approaches concentrate on matching data with a relatively similar level of detail, but different viewpoints [24], [29], [30]. However, as far as we know few works do concentrate on matching networks with different levels of detail [12]. We propose an approach to do so in this paper.

The remainder of this paper is as follows. Section 2 describes the differences between networks relatively to the level of detail, and existing criteria, tools and strategies used to match them are introduced. In Section 4, we propose a global matching process developed to match such networks. Before concluding, Section 5 illustrates the results of this matching process by some experiments.

2 Networks matching and levels of detail

2.1 Differences between representations of networks

Geographic objects are obtained by abstracting real world features, through pre-defined rules defining which objects are represented and how they are represented by means of

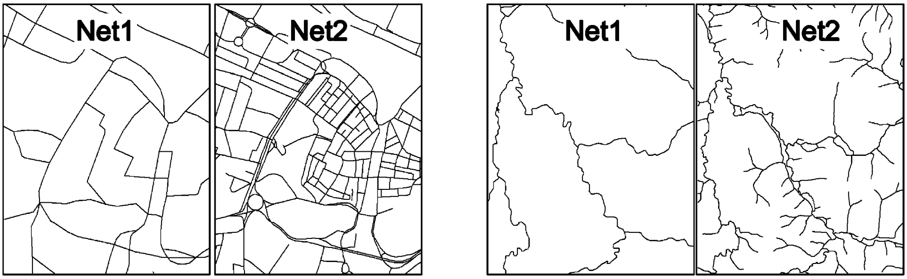


Fig. 1 Examples of networks with different levels of detail: roads (*left*) and rivers (*right*)

attributes. The set of these rules guiding the data capture process are called specifications. First, two different data captures may lead to different datasets consistent with the same specification as illustrated in the left of Fig. 2. Most importantly, differences between specifications, acquisition dates and acquisition processes infer differences between databases, named conflicts in the database community. Parent et al. [19] provides a taxonomy of conflicts between geographic databases and the right of Fig. 2 exemplifies some of them commonly encountered when matching networks. This figure shows several representations of a freeway containing several lanes in each direction. For the first and most precise representation (a), the specifications take into account the traffic separation: four objects of class "road sections" are defined, with an attribute "number of lanes" on each object. For the second representation (b), only three road sections are captured because specifications do not segment freeways into several road sections for each direction. For the last representation (c), only one road section is distinguished, because thresholds on minimal lengths necessary to create new objects are different.

2.2 Types of data matching relationships

The different conflicts between databases lead to different types of data matching results. When matching one network with another one more detailed, the most current cases are: unmatched data (1–0 or 0–1), one-to-one relationships (1–1), one-to-many relationships (1–N), and many-to-many relationships (N–M). First, some objects may appear in one database but have no homologous counterpart in the other one. This is of course the case for numerous objects of the most detailed database, but the reverse cases may also appear. These latter cases originate from errors, mismatch between updates or differences between specifications. Second, one object of one database may be matched to only one single object of the other database. Two main cases appear: one node matches with one node of the other database, and similarly one arc matches with one arc of the other database. More currently, one-to-many relationships may be encountered, where one object of the less detailed network is matched to several objects of the other one detailing more the representation of routes and connections. One-to-many relationships regularly appear if a single node of the less detailed network corresponds to a set of connected arcs and nodes forming a complete sub-graph for representing a connection in the other network (see for example the roundabout in Fig. 3, left). A one-to-many relationship also exists when one arc of the less detailed database is matched with one or two paths of the other one: for example, a segmentation conflict can bring about a 1–N relationship between one arc and two consecutive arcs (Fig. 3, middle, case a), or one arc and two parallel arcs (case b), or with a combination of consecutive and parallel arcs (case c). Finally, the more complex

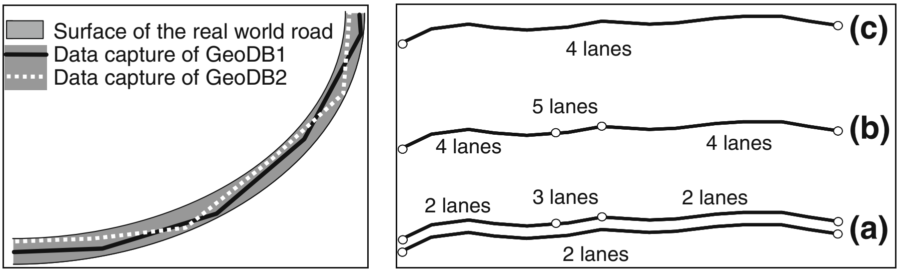


Fig. 2 Different valid captures of a road, either with the same (*left*) or different (*right*) specifications

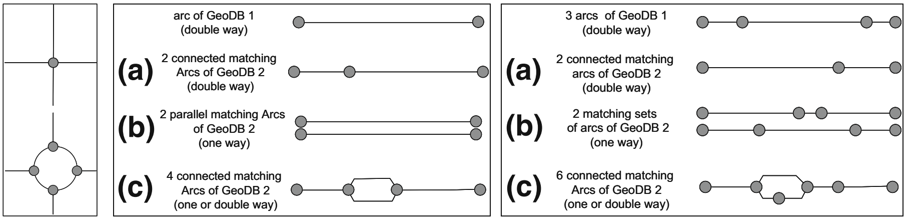


Fig. 3 Complex results of matching: 1–N relationship for a node (*left*) or for an arc (*middle*) matched to a subgraph; or even N–M relationships (*right*)

relationships are many-to-many relationships; They are relatively rare but may appear when specifications are very different (Fig. 3, right).

The ratio between matched and unmatched data depends on the differences between specifications and levels of detail. The closer they are, the more objects are matched. In the same way, the closer the levels of detail and specifications are, the simpler most relationships are. Practically, an efficient data matching process should be able to deal with all the different types of relationships, and this makes it a difficult process. It thus requires to take advantage of all kind of available information, as explained in the next section.

2.3 Criteria guiding network matching

Matching geographical objects may require comparing all their properties, be it geometrical internal properties, spatial organization among objects, and non-spatial internal properties. First, The geometrical position of objects is often the main criterion analysed to bypass the lack of common identifiers in geographic databases: homologous objects are first of all objects closely located. For points, the position is the only geometrical criterion to be analysed. For linear objects, a richer analysis of geometries can also be performed: shapes or orientations can for example be compared [6], [16], [20]. Second, a geographic data matching process also gains from comparing the spatial organisations of objects, and in particular the topological organisations of networks (how nodes and arcs are connected). This is exemplified in Fig. 4 that shows two actual railway networks: the sole analysis of positions could led to match node 1 to node B, while a careful analysis of topological organisations shows that these networks are actually shifted and that node 1 should be matched to node A. Finally, comparing non-spatial objects properties is also full of information to match networks, because homologous objects should have similar properties.

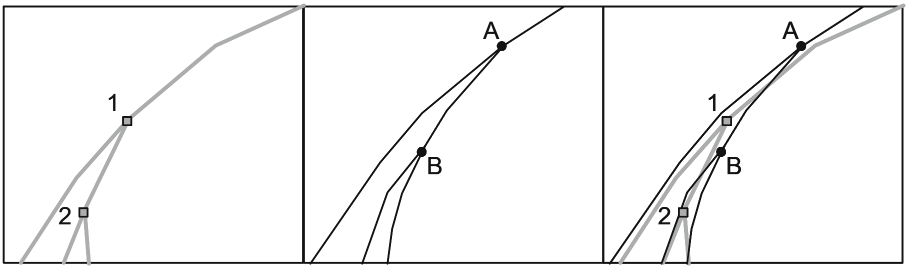


Fig. 4 Two railway networks and their superposing, showing the importance of topology when matching

2.4 Basic tools for matching

Comparing all these geometric and non-geometric properties of objects must be done through some tools that quantify how much objects are similar. Some of existing tools are described in this section.

2.4.1 Comparing geometries of nodes and arcs

Different tools exist to evaluate the spatial distance between geographical objects. For comparing the location of two points, the Euclidean distance is of course the most used. For comparing the location of two lines, different maximal and average distances may be used [8], [16]. The Hausdorff distance (d_H) that is the most popular maximal distance between two lines L_1 and L_2 [1], [7]: it is equal to the maximum of distances from any point of one of the two lines to the other line (Fig. 5). When one compare arcs of networks with different levels of detail, it is more valuable to consider the Hausdorff "semi-distance" from the lines of the most detailed network to the other ones (d_2 in Fig. 5). It is defined as the maximum of distances from any points of the detailed line to the other one. Indeed, the most detailed network is more split than the other one, and the full Hausdorff distance between one line and another line corresponding only to a part of the first one has no real meaning, as exemplified in Fig. 5. Other distances between lines may be defined to go further. Measures based on the Fréchet distance are for example particularly pertinent to compare very sinuous lines like coastlines [1], [8], [15], but detailing these measures is out of the scope of this paper.

The shape and orientations of lines may also be compared, in particular by means of the definition of a bounding area of the line. For example, the length and the width of the minimal bounding box are useful to evaluate how much the global shape is elongated [6]. These measures are full of information, but are more especially used to compare networks with similar levels of details, and thus with similar arcs.

2.4.2 Comparing spatial organisations

Some tools also exist to analyse the spatial organisation of objects. For the particular context of network matching, emphasis usually goes on the topological organisation of arcs and nodes. A simple approach is to consider topological characteristics as any other characteristic. For example, nodes may be characterised by the number of arcs connected to them [30]. The analysis of the network topology can also be thought of as the key information guiding the matching process. For example, nodes could be matched first, and arcs are subsequently matched in order to match arcs that are connected to matched nodes. Topological criteria are then more thought of as a matching strategy (see Section 3.3) rather than a characteristic at the same level than other characteristics.

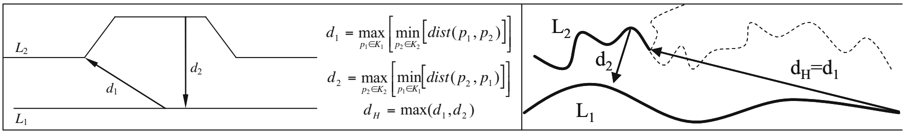

L_2
d_2
d_1
L_1
$d_1 = \max_{p_1 \in K_1}\left[\min_{p_2 \in K_2}\left[dist(p_1, p_2)\right]\right]$
$d_2 = \max_{p_2 \in K_2}\left[\min_{p_1 \in K_1}\left[dist(p_2, p_1)\right]\right]$
$d_H = \max(d_1, d_2)$
L_2
d_2
$d_H = d_1$
L_1

Fig. 5 Hausdorff distance and semi-distance between two lines

2.4.3 Comparing non-spatial attribute values

Comparing the nature of objects requires developing semantic distances between them. Almost all the approaches comparing the nature of objects rely on an ontology that inventory and organise the different concepts used to define the nature of the objects [21], [28]. These approaches are very promising but raise complex issues, such as the conception of ontologies and their alignment, and the development of semantic distances between concepts of the ontology. The difficulty is that the ontology may be either too complex and thus hard to define, or conversely too simple and thus hard to use to compare databases.

Comparing names is also very useful, because geographical names are pseudo-identifiers for geographical objects when they exist, even if differences may exist between names of homologous objects in different databases. Comparing names usually relies on the so-called edit distance or Levenshtein distance that determines how close two words are [13]. Some refinements of this distance also exist to compare full strings rather than simple words, which has been proved to be necessary for geographical names [24].

How to compare other characteristics of objects depends on the scale of measurement of the attributes. In particular, non-spatial attributes can be quantitative or qualitative. Comparing qualitative attributes, like the values of the attribute “number of lanes,” is rather direct and relies on distances between numbers. Comparing qualitative attributes, like the “traffic restriction” of roads, requires a semantic analysis of the attribute values similar to the one necessary for comparing the nature of the objects.

2.5 Some existing approaches for matching networks

Many criteria can be used to match data, and each one can be measured with different tools. The key issue for developing a matching process is then to choose and combine these tools, which raises several questions. On which criteria shall we mostly rely? Non-spatial criteria are full of information, but they usually require ad-hoc tools dedicated to specific attributes. Second, how to combine the criteria? Shall they be analysed one by one, and in which order? Shall they be considered holistically at once, and how to effectively combine them?

Walter and Fritsch [30] match networks with relatively similar scales. First, they propose to bypass systematic positional shifts in the data with a global and interactive stretching of one database on the other one. Then, they compare arcs with many different criteria such as their location, size, orientation and number of topological connections. Their approach relies on two principles. The first principle guides how thresholds are determined. Indeed, for each considered criterion, a threshold must be determined to fix which differences between homologous objects are authorised (for example the difference of length between homologous arcs should always be less than a given threshold). These thresholds are determined by a statistical analysis of some existing previously matched data. The second principle guides how the different criteria are combined. Measures used in the information theory paradigm are used to do that. The best matching between arcs is searched in order to optimise the measure of “mutual information” between networks. Based on these measures, the proposed approach also self-detects the more doubtful matched objects that need to be interactively checked, which is considered as an important point.

Whereas the approach of Walter and Frisch mainly compares arcs, the approach of Volz [29] also compares nodes, with the same goal of matching networks at close scales. This approach has the particularity of being progressive. Nodes and arcs that are the most similar are matched, and then the approach takes advantage of this first matching to define how other objects should be matched. This approach authorises to match slightly more different networks than the previous one, and does not require manually matched data to set up the thresholds, but it is less holistic.

In another approach, Zhang et al. [31] compare the locations of arcs and then subsequently refine the results through comparing the topological organisations of nodes supposed to match together, through counting and comparing their number of connected arcs. More drastically, Safra et al. [23] propose to compare only nodes when matching networks. This latter approach principally aims to speed up the process in contexts where the computational time is crucial, and mainly apply to databases with very close levels of detail.

Once arcs of networks have been matched, it may be useful to refine the matching results and to derive from them a matching between vertices of the arcs. This is particularly useful when one typically wants to transfer objects from one database to the other one, with rubber sheeting techniques [12], [22].

Beyond these approaches especially dedicated to match networks, let us also mention useful insights learnt from approaches developed to match isolated data represented by points or polygons like buildings. Most of approaches dealing with data with similar levels of details look for one-to-one matching links [4], [24], whereas approaches dedicated to data with different levels of detail also look for one-to-many or even many-to-many links [5]. Most approaches compare only two databases at a time, but some other shows that one can take advantage of comparing simultaneously several databases [24]. Some authors show that matching may require comparing databases in both directions [4]: in their approach, objects of DB_1 are first matched to objects of DB_2 on some criteria, but objects of DB_2 are also matched to object of DB_1 with the same criteria, and they combine those two mono-directional processes in order to finally decide the actual matching of data. Finally, some authors show that it may be possible to automatically look for systematic shifts in the data [24]: once some of the objects have been first matched, other objects may be matched while taking into account systematic shifts between the matched objects in the first step.

3 *NetMatcher*: a matching process for networks at different levels of detail

In this section, we describe the *NetMatcher* process, developed for matching networks with different levels of detail. This process has been originally developed by Devogele [9] and then enhanced and extensively tested by Mustière [17]. For the sake of clarity, we will name Net_1 the less detailed network and Net_2 the most detailed, and subsequently A_1 an arc of Net_1, A_2 an arc of Net_2, N_1 a node of Net_1, N_2 a node of Net_2…

The *NetMatcher* process relies on several principles. The first one is the complementarities of arcs and nodes: matching nodes and matching arcs are two useful complementary approaches for network matching, and thus both points of views are consistently considered during the process. The second principle is to follow an approach roughing-out and focusing: the first roughing-out step searches for potential matching candidates, and the second focusing step searches among candidates the actual matching links. Another principle is to bear in mind the dominance of one-to-many links: we consider that matching networks with different levels of detail results in numerous one-to-many relationships from the less detailed network to the other one. Finally, the process is designed in order to reduce as much as possible its sensitivity to the thresholds, because determining these thresholds is key issue in practice [30].

The overall process is described in Fig. 6. The first step is the preparation of networks in order to facilitate their comparison (see Section 3.1). Then nodes are roughly compared: for

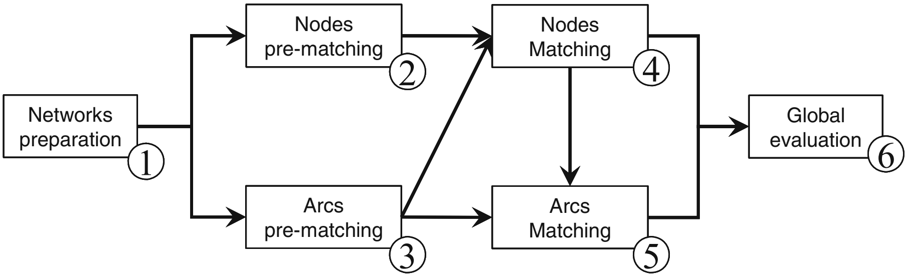


Fig. 6 The NetMatcher matching process

each node of Net_2, a first selection is made among nodes of Net_1 to look for potential candidates for matching (Section 3.2), and a similar step if performed for arcs (Section 3.3). Based on the previous two pre-matching of arcs and nodes, the final matching decision is made for each node of Net_1 (Section 3.4), and then for each arc of Net_1 (Section 3.5). The final step is a global evaluation of the results (Section 3.6).

3.1 From geographic networks to graphs

The first step of the *NetMatcher* process is to transform initial geographic features into a common graph structure. Whatever the original networks look like, the matching process requires input graphs with the basic following arcs and nodes structure: an arc links an initial and a final node; the geometry of an arc is an oriented line and arcs may be qualified as ‘one-way’ of ‘two-ways’ if this information is available and pertinent (e.g. for roads); the geometry of a node is a point and nodes may be qualified with a ‘size’ representing the importance of the node on the real ground (e.g. a roundabout is know bigger than a simple junction). Different pre-treatments may be performed to transform the initial features into the generic graph structure. Typically, in our test data the electric network is originally represented by electric lines (with a line for geometry) and transforming stations (with a surface for geometry). In this case, a node of the graph is created for each transforming station and located at the barycentre of the surface, and one arc is created for each electric line with a geometry distorted in order to connect to the nodes.

3.2 Looking for candidates: Pre-matching of nodes

The second step of the *NetMatcher* process is a *pre-matching of nodes*: we look for couples of nodes (N_1, N_2) that are potential candidates for matching. This selection is based on a simple distance criterion between positions of nodes. Determining how far we consider close nodes as candidates for matching may be defined by a fixed threshold related to the overall metric precision of the two networks. Some attribute based criteria may also easily be added at this step for refining it: in the roads example, the threshold may be quite big if the node is known representing a complex interchange, but may be quite small if the node is know representing a simple junction. These criteria are really data dependent for the time being, but works on geographic ontologies are a step towards a general approach to that point [21].

Determining such distance thresholds is usually a key issue for the effectiveness of the process [30]. In our case, the purpose of this pre-matching step is to make an over-selection of candidates. It thus has a minor sensitivity to the distance threshold(s), as long as they are pessimistic: thresholds are better over-evaluated than under-evaluated.

3.3 Looking for candidates: Pre-matching of arcs

Similarly to the pre-matching of nodes, the next step of the *NetMatcher* process is a pre-matching of arcs: for each arc of Net_2, we look for close arcs in Net_1 candidate for matching. Here we only consider how the most detailed arcs are far from the less detailed one, but not conversely. Still, this raises two questions: how to quantify the closeness, and what is the maximal distance authorized between two candidates?

First, the distance between arcs is defined as the Hausdorff semi-distance d_2 (A_2, A_1) introduced in Section 2.4.1. Second, the selection principle is once again designed in order to reduce the sensitivity to the thresholds. Given an arc A_2 of Net_2, we first search for the closest arc $A_{1\mathrm{closest}}$ in Net_1. We then consider as candidates for matching with A_2 all the arcs of Net_1 at a distance to A_2 lesser than $\mathrm{Min}(D_{\max}, d_2(\mathrm{A}_2, A_{1\mathrm{closest}})+D_{\mathrm{res}})$, where $D_{\max}$ is a threshold representing the maximum expected position shifts between the networks, and D_{res} is another threshold expressing that all arcs of Net_1 at a distance relatively similar to A_2 should be equally considered. Like when pre-matching nodes, some semantic criteria may also be introduced in this pre-matching step: for example we may define different thresholds when dealing with large roads like highways and narrow local roads.

3.4 Matching nodes

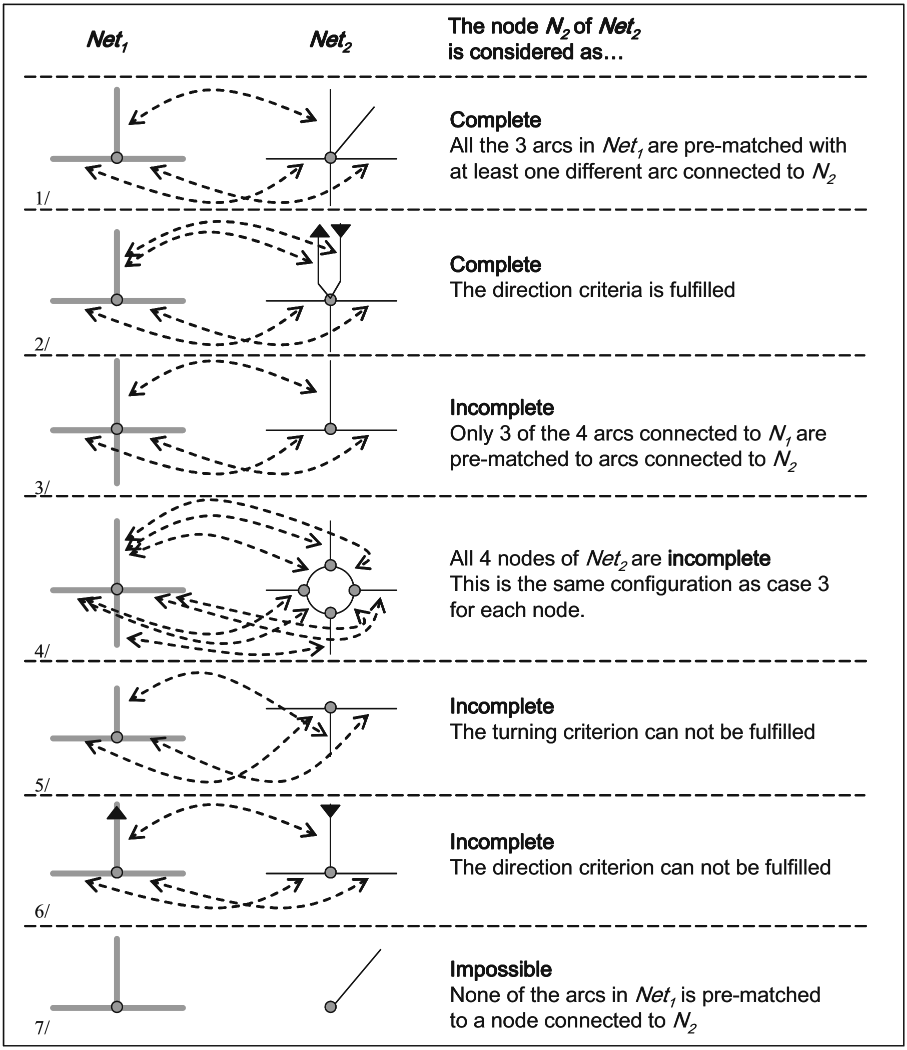
The two previous steps determined pairs of candidates for matching. The purpose of this third and most complex step of the *NetMatcher* process is to determine the actual matching links for nodes of Net_1. The main principle of this step is to analyse the consistency between arcs and nodes pre-matching.

For a given node N_1 in Net_1, its matching process is fully based on the analysis of the candidate nodes N_2, defined as follow (examples in Fig. 7, where dashed lines represent some pre-matching links between arcs):

> N_2 is said to be a *complete* candidate if we can find a correspondence in the pre-matching of arcs between *all* the arcs connected to N_1 and some of the arcs connected to N_2.
>
> N_2 is said to be an *incomplete* candidate if the pre-matching of arcs exhibits at least *one* correspondence between one arc connected to N_1 and one arc connected to N_2.
>
> N_2 is said to be an *impossible* candidate if the pre-matching of arcs exhibits no correspondence at all between arcs connected to N_1 and arcs connected to N_2.

Let us precise that during this study of correspondences, a special care is given to two criteria. The ‘turning criterion’ checks that the correspondences between arcs are consistent with the sequence of the arcs around the node. For example, let say that N_1 has three connected arcs A_1, B_1 and C_1, and N_2 has four connected arcs W_2, X_2, Y_2, Z_2. If A_1 is pre-matched with W_2, B_1 with Y_2, C_1 with Z_2; the turning criterion checks that W_2, Y_2, Z_2 are organised around N_2 (e.g. clockwise) the same way that A_1, B_1 and C_1 are organised around N_1. Then, the ‘direction criterion’ compares the directions of the arcs regarding the “one-way” attribute. Only arcs similarly oriented can be put into correspondence. More, an arc in both ways must be must be put into correspondence with either one arc in both ways connected to N_2, or with two arcs in one-way but in the opposite direction connected to N_2. If these criteria are not fulfilled, the node is considered as incomplete.

Based on these definitions, the actual decision process for matching nodes is described in Fig. 8.



Net1
Net2
The node N2 of Net2 is considered as...
1/
Complete
All the 3 arcs in Net1 are pre-matched with at least one different arc connected to N2
2/
Complete
The direction criteria is fulfilled
3/
Incomplete
Only 3 of the 4 arcs connected to N1 are pre-matched to arcs connected to N2
4/
All 4 nodes of Net2 are incomplete
This is the same configuration as case 3 for each node.
5/
Incomplete
The turning criterion can not be fulfilled
6/
Incomplete
The direction criterion can not be fulfilled
7/
Impossible
None of the arcs in Net1 is pre-matched to a node connected to N2

Fig. 7 Consistency of nodes and arcs pre-matching (for the sake of clarity the two networks are shifted but should be considered approximately at the same place)

The overall principle is first to consider nodes N_1 of Net_1 one by one. Then, we first look for 1–1 matching links: if one "complete" candidate node exists, i.e. with a full consistency between nodes and arcs pre-matching, it is considered as the actual homologous object of N_1, and the matching link is marked as certain (Fig. 9, case b). Else, only one incomplete candidate node exists, it is also considered as the actual homologous object, but this matching link is marked as uncertain (case c). In the other cases, the node of Net_1 is likely to correspond to several nodes in Net_2, one then look for 1–N matching links. In order to do that, connected subgraphs of Net_2 linking candidate nodes are built and wholly considered as groups candidate for matching with N_1 (they are thought of as hypernodes of the network). These groups are then characterized one by one as *complete*, *incomplete* or *impossible* candidate, and actual matching links are built and marked as certain or not,

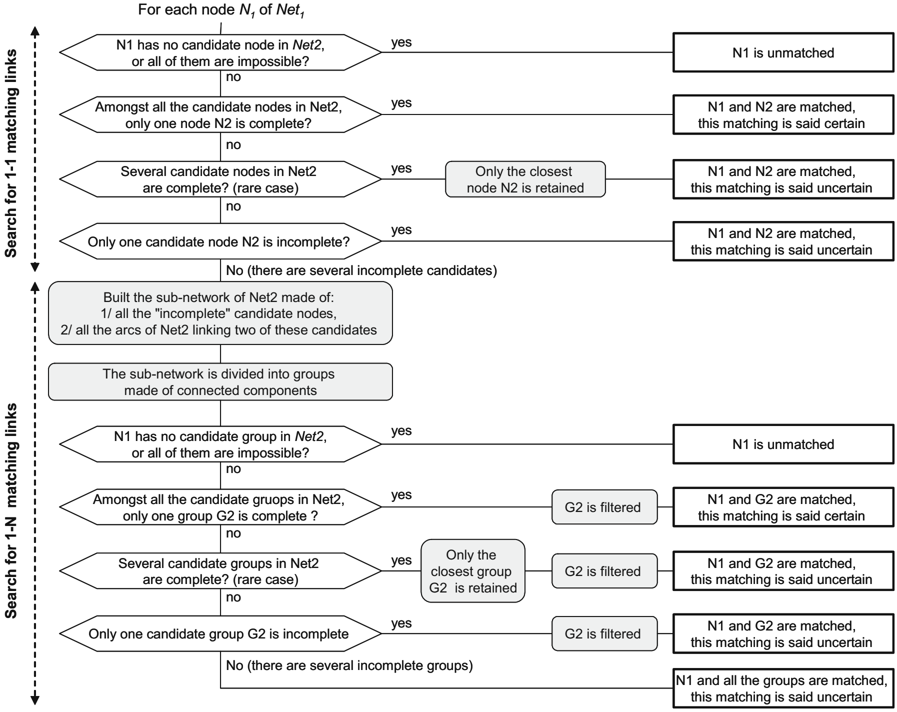

For each node N_1 of Net_1
Search for 1-1 matching links
N1 has no candidate node in Net2, or all of them are impossible?
yes
N1 is unmatched
no
Amongst all the candidate nodes in Net2, only one node N2 is complete?
yes
N1 and N2 are matched, this matching is said certain
no
Several candidate nodes in Net2 are complete? (rare case)
yes
Only the closest node N2 is retained
N1 and N2 are matched, this matching is said uncertain
no
Only one candidate node N2 is incomplete?
yes
N1 and N2 are matched, this matching is said uncertain
No (there are several incomplete candidates)
Search for 1-N matching links
Built the sub-network of Net2 made of: 1/ all the "incomplete" candidate nodes, 2/ all the arcs of Net2 linking two of these candidates
The sub-network is divided into groups made of connected components
N1 has no candidate group in Net2, or all of them are impossible?
yes
N1 is unmatched
no
Amongst all the candidate gruops in Net2, only one group G2 is complete ?
yes
G2 is filtered
N1 and G2 are matched, this matching is said certain
no
Several candidate groups in Net2 are complete? (rare case)
yes
Only the closest group G2 is retained
G2 is filtered
N1 and G2 are matched, this matching is said uncertain
no
Only one candidate group G2 is incomplete
yes
G2 is filtered
N1 and G2 are matched, this matching is said uncertain
No (there are several incomplete groups)
N1 and all the groups are matched, this matching is said uncertain

Fig. 8 The overall process of node matching

exactly with the same principles than the one used for nodes (case d). Finally, the retained group may contain extra-objects that should not be matched (case e), a final filtering of the groups is then performed to remove parasitic extensions (case f): without detailing this step in detail, let us say that it intends to remove arcs of the group without loosing its completeness: a complete (respectively incomplete) group before filtering should not become incomplete (respectively unmatched) after.

3.5 Matching arcs

Once nodes of Net_1 have been matched, the *NetMatcher* process matches arcs. Similarly to node matching, this step considers arcs A_1 of Net_1 one by one. The principle of this step is when to search in Net_2 the path that, (1) is made of arcs candidate for matching with A_1, (2) links the matched nodes of A_1 and, (3) is the closest as possible to A_1. We define the notion of closest path to A_1 as the path minimizing the surface between it and A_1 (the grey surface in Fig. 10).

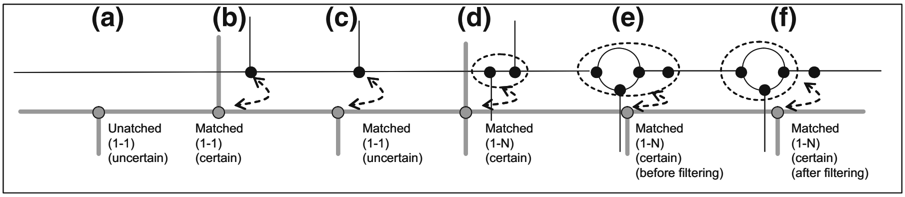

(a)
(b)
(c)
(d)
(e)
(f)
Unatched (1-1) (uncertain)
Matched (1-1) (certain)
Matched (1-1) (uncertain)
Matched (1-N) (certain)
Matched (1-N) (certain) (before filtering)
Matched (1-N) (certain) (after filtering)

Fig. 9 Typical results of node matching

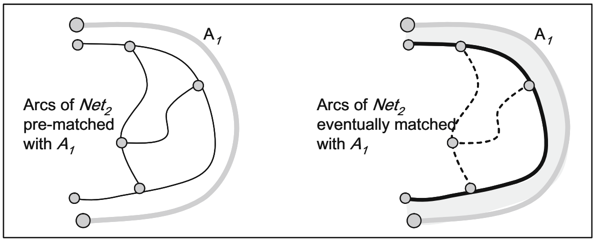

A1
Arcs of Net2 pre-matched with A1
A1
Arcs of Net2 eventually matched with A1

Fig. 10 The closest path to an arc minimises the surface separating them

Computationally speaking, a 'shortest' path is computed with weights on arcs A_2 being equals to the surface defined when projecting A_2 on A_1 (rather than the length of A_2). One may notice that while the pre-matching of arcs relies on a maximal distance between arcs, this step relies on an average distance between arcs. This is consistent with the idea of roughing out with rough measures eliminating worst candidates and focusing with finer measures balancing several considerations.

More, if the arc A_1 is known to be a double-way arc, this search for closest path is done in both directions, taking into account the direction of arcs in *Net*$_2$. Figure 11 illustrates this: the two nodes at the extremities of A_1 have been previously matched to, respectively, several and one node in *Net*$_2$. The final arc matching retains as matched with A_1 the closest paths in *Net*$_2$ linking these matched nodes in both directions.

Some semantic criteria may also be easily introduced in this step. Typically, when matching uncovered road of *Net*$_1$, we favor paths going through secondary roads in *Net*$_2$ rather than paths going through primary roads. This can be easily done through the modification of weights associated to arcs in the closest path computation: the more the semantic difference between A_1 and A_2 is important, the more the arc A_2 is weighted. Similarly, if arcs correspond to roads that have the same identifying numbers, the weight may be significantly decreased.

Finally, like for node matching, an evaluation made at this step: if the extremities of the arc are matched without doubt, the arc is also considered as matched without doubt; otherwise it is considered doubtful.

In practice, one may notice that the key step of *NetMatcher* is the node matching: the effectiveness of the arc matching highly depends on the effectiveness of the node matching. *NetMatcher* should be thought of as a process matching first of all connections of networks and from which arc matching is derived.

3.6 Global evaluation

During the matching process, the result of a node or an arc matching has been marked as certain or not. However, this evaluation is based on local criteria. Actually, the *NetMatcher* process considers matching nodes (respectively arcs) of *Net*$_1$ one by one, independently

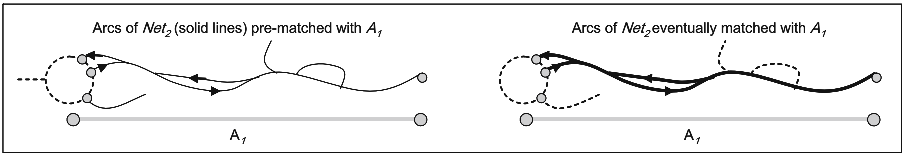

Arcs of Net2 (solid lines) pre-matched with A1
A1
Arcs of Net2 eventually matched with A1
A1

Fig. 11 All the arcs in *Net*$_2$ pre-matched with an arc in *Net*$_1$ (*left*) and the final choice of matching (*right*)

from the others. Thus a final evaluation step is performed to detect arcs or nodes of Net_2 matched with two different elements in Net_1. This inconsistent situation should not appear if Net_1 is less detailed than Net_2, and is thus marked as doubtful.

4 Experiments

4.1 Test data and experimental settings

Our main experiments concern two databases from IGN-France, made from different sources and with different purposes. The less detailed database used (Net_1) is named BD CARTO® (Fig. 12, left). Its precision is from one to several decameters according to the considered themes. It precisely describes the topological organization of the networks. It is typically used to make maps at 1:100,000 or 1:250,000 scale, or for geographical studies on large areas. The most detailed database (Net_2) is named BD TOPO®, and its precision is close to one meter (Fig. 12, right). It concentrates on the precise topographic description of networks. It is typically used to make maps at 1:25,000 scale and some local studies. Matching these two databases interests IGN for several reasons. First, the matching is necessary to automatically propagate updates from BD TOPO to BD CARTO. Second, the matching allows highlighting inconsistencies between databases [25] and enriching them each other. Third, the matching paves the way to BD CARTO users to propagate their own data enrichment into BD TOPO, which is an increasing need.

In order to evaluate the *NetMatcher* process, we matched the two databases in an area of 150×90 km, chosen for the heterogeneity of its landscape containing coastal and inland areas, mountainous and flat areas, rural and urban areas [17]. The experiments reported here concern the following themes: roads, rivers, electric lines, and railways that are represented in both databases, plus hiking routes of BD CARTO that have been matched to paths and roads of BD TOPO (Fig. 12). The studied networks are very diverse: one is natural while others are manmade; they are differently dense and complex (from simple 200 electric lines to complex roads with 100,000 road sections in BD TOPO); some have very different levels of detail in the databases while other are relatively similar; and finally, some networks

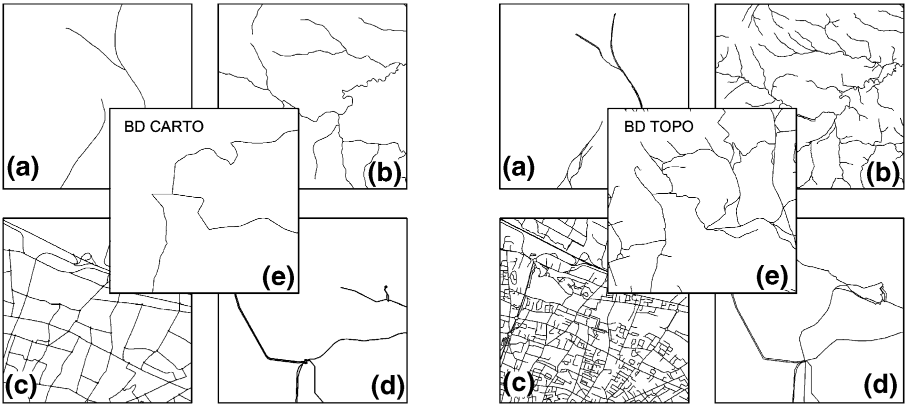


Fig. 12 Test data (BD CARTO on left, BD TOPO on right): railways (a), rivers (b), roads (c), electric lines (d), and hiking routes of BD CARTO compared to roads and paths of BD TOPO (e)

contain information on nodes while others not. Let us notice that other networks from diverse producers (IGN-Belgium, TeleAtlas) have also been tested with similar results, but not detailed here.

Practically, the *NetMatcher* process has been coded in Java, in the open source system GeOxygene[1] [3]. The *NetMatcher* code itself is supposed to be released in open source shortly as well. The visualisation and checking of results is done in Jump (http://www.jump-project.org/), another open source GIS. In the previous section we presented where semantic criteria may be introduced in the process. The actual choices made for these experiments are not detailed here because they are too data dependant. We may just notice that we globally did not make an intensive use of semantic information, in order to evaluate a relatively general process.

4.2 Evaluation principles

In order to quantify the results, we defined several indicators. The first indicator is a classical one in the field of information retrieval: *Precision* (adapted from [4]). The global precision of the process is the percentage of the length of Net_1 effectively matched or effectively not matched when compared to a manual matching. This indicator can be refined: the *Precision of matching* is the percentage of the length of Net_1 effectively matched (i.e. matched to the right corresponding objects in Net_2); and the *precision of not-matching* that is the percentage of the length of Net_1 effectively not matched (i.e. objects of Net_1 not matched by the matching process and that actually have no corresponding object in Net_2).

The own internal evaluation of *NetMatcher* is also evaluated through the *optimism* and *pessimism* indicators. *Pessimism* is the percentage of the length of Net_1 that is effectively matched or effectively not-matched compared to a manual matching but is internally evaluated by the algorithm as doubtful. Conversely, *Optimism* is the percentage of the length of Net_1 that is not effectively matched or not effectively not-matched compared to a manual matching but is internally evaluated by the algorithm as certain. These indicators are very important in practice, when an interactive editing will be done to correct errors of the process: checking the whole dataset is most of the time intractable and thus only objects considered as doubtful by the process itself will be checked and possibly corrected. An over-pessimistic process will thus be expensive: it will lead to a useless important work of interactive editing. Conversely, an over-optimistic process will be cheap but ineffective: too many errors will not be interactively edited.

Theoretically, calculating these indicators requires some reference matching results, which rarely exist in practice and are very expensive to obtain. Thus, for our experiments all these indicators have been evaluated through an interactive counting of errors. While this task is very long, it has been performed on an excerpt of the data, and then extrapolated to the entire dataset. Our evaluations are thus just rough estimations.

4.3 Quantitative results

The matching process has been applied to the whole test area in acceptable runtime for such a batch process: it took from a few minutes for simple networks to two hours for the most complex one, in a standard personal computer. The qualitative evaluation of the effectiveness is summarized in the following Table 1.

These qualitative results require some comments. First the global precision illustrates the effectiveness of the *NetMatcher* process: from 90% to almost 100% of the networks are rightly matched. As it was expected, the less complex are the networks, the more effective

Table 1 Qualitative evaluation of the matching

Global precision (%)	Precision of matching (%)	Precision of not-matching (%)	Optimism (%)	Pessimism (%)
90	95	20	1	15
95	95	50	1	15
98	100	90	0	25
99	98	90	1	10
97	97	95	2	5

is the process. If these results are encouraging for most networks, they must be moderated for the road network: 10% of errors can be thought of as a good result compared to the complexity of the networks, but in practice it still represents an important work for an interactive editing. Second, we can notice that the precision of matching is more important than the precision of not-matching. In other words: when *NetMatcher* matches an object it is often rightly, but when it does not match an object this is sometimes wrong (the corresponding object in Net_2 has not been found). Finally, the process is much more pessimistic than optimistic. This can be explained by the fact that, for the time being, whenever the process does not match an object it considers that as doubtful, which is actually an over-pessimistic approach. Nevertheless, this means that the criteria determining the quality of the results in the process are effective but must still be tuned. As we said before, this is very important to reduce the interactive editing and then lead to an actual use of such a process.

4.4 Qualitative results

Figure 13 illustrates typical results obtained by the *NetMatcher* process. In all screenshots, BD CARTO (Net_1) is solely displayed on the upper right, BD TOPO (Net_2) is solely displayed on the upper left, and both databases as well as the links representing the matching results are displayed in the bottom: green segments connect matched arcs, and the node or sub-graph of Net_2 corresponding to a node of Net_1 is circled in green. Case 1 of Fig. 13 shows a typical result on roads: the node A_1 is matched to the node A_2, the node B_1 is matched to the full roundabout B_2.

A first interest of the process exemplified here is to search for one-to-many links. When one node of Net_1 corresponds to a detailed but relatively simple connection in Net_2 the result is generally effective. Cases 1 to 3 illustrate this: one node of Net_1 may for example correspond to a full roundabout (case 1), a "shifted connection" made of two nodes and one arc (case 2), or a "square" junction (case 3). One-to-many matching also occurs for arcs. One arc of Net_1 is usually matched to several consecutive arcs of Net_2, as illustrated in cases 4 and 5. One arc may also be matched to several parallel arcs like in case 6. The process has proved to be rather effective in these latter cases [9], but it requires knowing the direction of arcs (one-way, double-way), which was unfortunately missing in our test data presented here.

The results also show that the process does not necessary match together the closest objects. Cases 7 and 8 display some nodes and arcs of Net_1 that are effectively matched to nodes and arcs that are not the closest objects in Net_2. Case 7 corresponds to the case already mentioned in Fig. 4. In case 8 the two electric networks are shifted, resulting in a complex matching where closest arcs are not the right one to be matched.

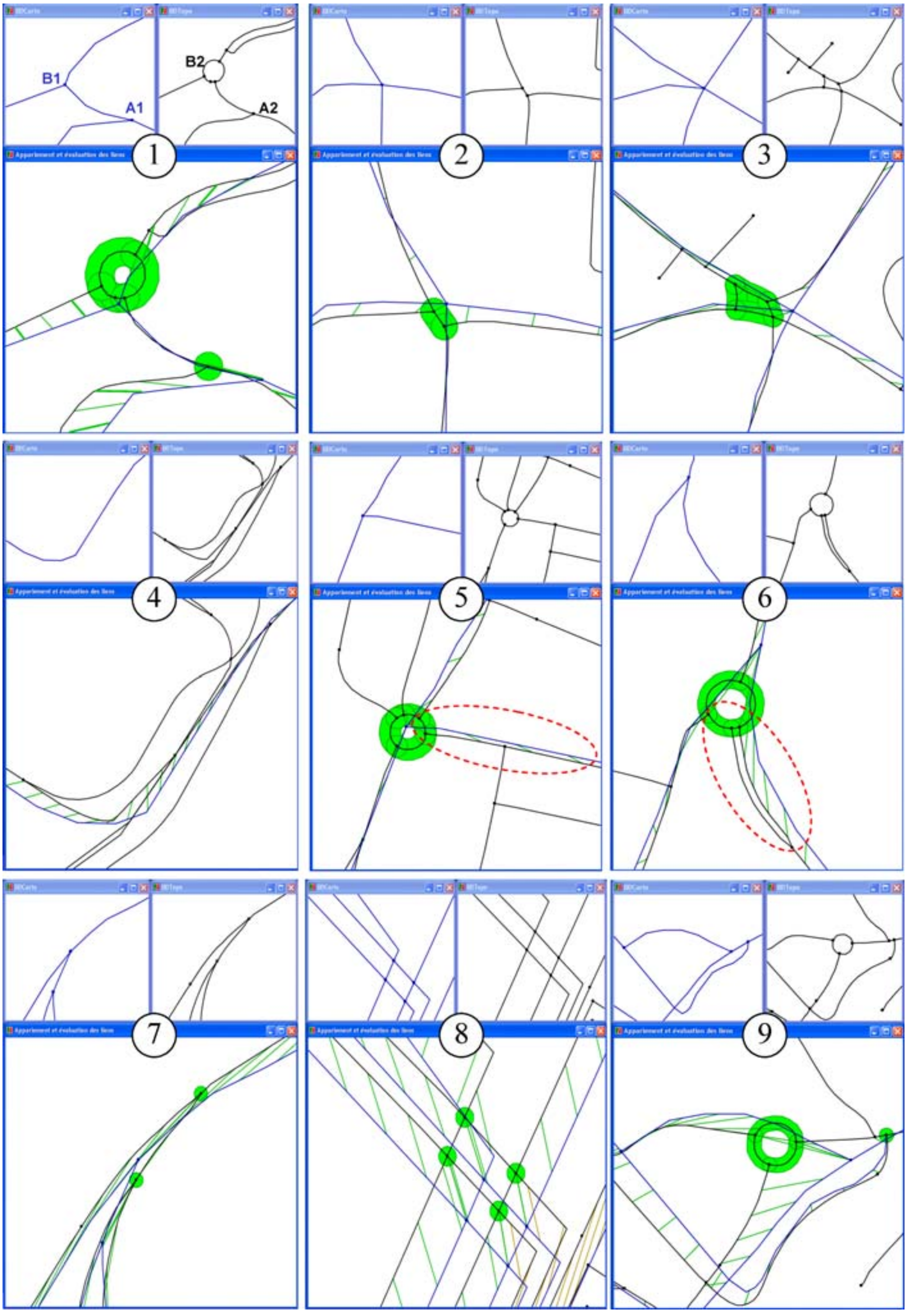


Fig. 13 Typical matching results

The process also fails to match some parts of the networks, revealing very interesting cases of inconsistencies between data, as exemplified in case 9 where one of the databases has not been updated. Detecting such inconsistencies is of significant interest for data producer that intend to detect errors and improve their data. Finally, the limits of the process appear near complex connections where most of the errors are located. This subsequently raises an important issue: visual checking and editing is very difficult in these cases, and even impossible without more interactive tools. Ergonomics is thus an important issue in practice.

5 Conclusion

We presented an approach to match networks with different levels of details, named *NetMatcher*. This approach is a several steps process. The first steps determine candidates for matching; they mainly rely on geometric criteria even if they can also make use of attribute values. The next steps determine the actual matching results; based on the previous steps, they compare the topological organisation of networks. One of the main advantage of the proposed process is to determine one-to-many links between networks: a node of the less detailed network can be matched to several arcs and nodes forming a complex junction in the most detailed network; an arc of the less detailed network can also be matched to several serial or parallel arcs and nodes in the less detailed network. A strength of *NetMatcher* is to minimise its sensitivity to parameters. Another strength is to self-evaluate its results through the comparison of topological organisations of networks, thus paving the way to an interactive editing of the results.

The process can still be improved, and several directions may be explored. The first one is the improvement of measures used to compare geometric as well as other characteristics of objects. For example, the distance between arcs could gain in relying on the Fréchet distance, which may be more efficient especially for natural objects [8]. A second research direction concerns the combination of the matching criteria. The process may take more advantage of the comparison of attribute values, but this requires an efficient combination of geometric, topologic and semantic criteria. This could be done through the use of theories managing uncertainty, like fuzzy logic or theory of evidence [18].

References

1. H. Alt and M. Godau. “Computing the Fréchet distance between two polygonal curves”, *International Journal of Computational Geometry and Applications*, Vol. Vol. 5(1/2):75–91, 1995.
2. T. Badard and C. Lemarié. “Propagating updates between geographic databases with different scales,” *in Proc. of GeoComputation ’2000: Innovations in GIS VII*, pp. 134–146, London, UK: Taylor & Francis, 2000.
3. T. Badard and A. Braun. “Oxygene: a platform for the development of interoperable geographic applications and web services.” *In Proc. of the 15th International Workshop on Database and Expert Systems Applications (DEXA'04)*, pp. 888–892, IEEE Press: Zaragoza, Spain, 2004.
4. C. Beeri, Y. Kanza, E. Safra, and Y. Sagiv. “Object fusion in Geographic Information Systems,” *in Proc. of the 30th VLDB conference*, pp. 816–827, Toronto, Canada, 2004.
5. A. Bel Hadj Ali, and F. Vauglin. “Geometric Matching of Polygons in GISs and assessment of Geometrical Quality of Polygons,” *in Proc. of Int. Symp. on Spatial Data Quality (ISSDQ’99)*, pp. 33–43, Hong Kong, 1999.
6. B. Buttenfield. “Line Structure in Graphic and Geographic Space,” PhD thesis, University of Washington, USA, 1984.
7. M. Deng, X. Chen and Z. Li. “A Generalized Hausdorff Distance for Spatial Objects in GIS,” *in Proc. of the 4th ISPRS Workshop on Dynamic and Multi-dimensional GIS*, pp. 10–15, Pontypridd, UK, 2005.
8. T. Devogele. “A new Merging process for data integration based on the discrete Fréchet distance,” *in Proc. of the 10th Int. Symposium on Spatial Data Handling (SDH)*, pp. 167–181, Ottawa, Canada: Springer, 2002.

9. T. Devogele. “Processus d'intégration et d'appariement des bases de données géographiques—Application à une base de données routière multi-échelles,” PhD thesis, Université de Versailles, France, 1997.
10. T. Devogele, C. Parent, and S. Spaccapietra. “On spatial database integration”, *International Journal of Geographical Information Science*, Vol. Vol. 12(4):335–352, 1998.
11. M. Gombosi, B. Zalik, and S. Krivograd. “Comparing two sets of polygons”, *International Journal of Geographical Information Science*, Vol. Vol. 17(5):431–443, 2003.
12. J.–H. Haunert. “Link based conflation of geographic datasets,” *in Proc. of the 8th ICA workshop on Generalisation and Multiple Representation*, 7 p,, A Coruña, Spain, 2005.
13. V.I. Levenshtein. “Binary codes capable of correcting deletions, insertions, and reversals,” *Soviet Physics-Doklady*, Vol. 10(8):707–710, 1966. *Translated from Doklady Akademii Nauk SSSR*, Vol. 163 (4): 845–848, 1965.
14. D. Mantel and U.W. Lipeck. “Matching Cartographic Objects in Spatial Databases”, *in Proc. of ISPRS’2004, Archives of ISPRS*, Vol. Vol. 35(B4):172–176, 2004.
15. A. Mascret, T. Devogele, I. Le Berre and A. Hénaff. “Coastline matching process based on the discrete Fréchet distance,” *in Proc. of the 12th International Symposium on Spatial Data Handling (SDH)*, pp. 383–400, Springer: Vienna, Austria, 2006.
16. R.B. McMaster. “A statistical analysis of mathematical measures for linear simplification,” PhD thesis, University of Kansas, KS, 1983.
17. S. Mustière. “Results of experiments on automated matching of networks,” *in Proc. of the ISPRS Workshop on Multiple Representation and Interoperability of Spatial Data*, pp. 92–100, Hanover, Germany, 2006.
18. A.-M. Olteanu. “A multi-criteria fusion approach for geographical data matching,” *in Proc. of the 5th International Symposium On Spatial Data Quality (ISSDQ)*, Enschede, Netherlands, 2007.
19. C. Parent, S. Spaccapietra, and T. Devogele. “Conflicts in Spatial Database Integration,” *in Proc. Of the 9th Int. Conf. on Parallel and Distributed Computing Systems, PDCS ‘96*, pp. 772–778, Dijon, France, 1996.
20. C. Plazanet. “Enrichissement des bases de données géographiques: analyse de la géométrie des objets linéaires pour la généralisation cartographique (application au routes),” PhD thesis, Univ. de Marne-la-Vallée (F), 2006.
21. M.A. Rodriguez and M.J. Egenhofer. “Determining semantic similarity among entity classes from different ontologies”, *IEEE Transactions on Knowledge and Data Engineering*, Vol. Vol. 15(2):442–456, 2003.
22. A. Saalfeld. “Conflation: automated map compilation”, *International Journal of GIS*, Vol. Vol. 2(3):217–228, 1988.
23. E. Safra, Y. Kanza, Y. Sagiv, and Y. Doytsher. “Efficient integration of road maps,” *in Proc. of the 14th annual ACM international symposium on Advances in geographic information systems*, pp. 59–66, ACM Press: Arlington, VA, 2006.
24. A. Samal, S.C. Seth, and K. Cueto. “A feature-based approach to conflation of geospatial sources”, *International Journal of Geographical Information Science*, Vol. Vol. 18(5):459–489, 2004.
25. D. Sheeren, S. Mustière, and J.-D. Zucker. “Consistency assessment between multiple representations of geographical databases: a specification-based approach,” *in Proc. of 11th International Symposium on Spatial Data Handling (SDH)*, pp. 617–627, Leicester, United Kingdom: Springer, July 2004.
26. A. Stonykova. “Semantic validation in spatio-temporal schema integration,” PhD thesis, Ecole Polytechnique Fédérale de Lausanne, Switzerland, 2006.
27. S. Timpf, G. Volta, D. Pollock, and M.J. Egenhofer. “A conceptual model of wayfinding using multiple levels of abstraction,” in Goos et Hartmanis (Eds.), *Theories and Methods of Spatio-Temporal Reasoning in Geographic Space*, Springer, pp. 348–367, 1992.
28. H. Uitermark. “Ontology-based geographic data set integration,” PhD thesis, Universiteit Twente, The Netherlands, 2001.
29. S. Volz. “An iterative approach for matching multiple representations of street data,” *in Proc. of the ISPRS workshop on Multiple Representation and Interoperability of Spatial Data*, pp. 101–110, Hanover (G), 2006.
30. V. Walter and D. Fritsch. “Matching spatial data sets: a statistical approach”, *International Journal of Geographical Information Science*, Vol. Vol. 13(5):445–473, 1999.
31. M. Zhang, W. Shi, and L. Meng. “A generic matching algorithm for line networks of different resolutions,” *in Proc. of 8th ICA workshop on Generalisation and Multiple Representation*, 8 p, A Coruña, Spain, 2005.

Dr. Sébastien Mustière is a researcher in GIS at the COGIT Laboratory of IGN-France since 1997. He received a PhD in 2001 in Computer Science and Artificial Intelligence, on the subject of "Machine Learning for Cartographic Generalisation." He first worked on the field of automatic generalisation. After a PostDoc position in Laval University in Quebec, he works since 2002 on the field of geographic databases integration. His main current research subjects are, on the one hand, data matching and, on the other hand, building and alignment of geographic ontologies.

Dr. Thomas Devogele is an Assistant Professor in computer science at the French Naval Academy Research Institute. His research interests include spatial databases, computational geometry, navigation systems and digital elevation models. Dr. Thomas Devogele received his PhD in computer science in 1997 from the University of Versailles and the French National Geographic Institute. His thesis was oriented towards spatial database integration, data matching and multi-scale representation. His current research interests involve moving objects, integration of topographic and bathymetric data, and maritime GIS. He is one of the leaders of the French national working group on mobility and real time GIS (SIGMA research network).