On-the-Fly Edge-Matching for Cascaded Service Content

Pekka Latvala
Finnish Geospatial Research Institute
PL 84
Helsinki, Finland
pekka.latvala@nls.fi

Lassi Lehto
Finnish Geospatial Research Institute
Geodeetinrinne 2
Masala, Finland
lassi.lehto@nls.fi

Samuli Kytö
Finnish Geospatial Research Institute
PL 84
Helsinki, Finland
samuli.kyto@nls.fi

Abstract

Cascading download services combine various background services together so that their contents can be queried via a single service. A common problem with cascading services in a multinational setting is that the features coming from the services of different countries often do not match across border areas. Usually, edge-matching is executed iteratively as an off-line process between the neighbouring countries where adjustments and tests can be made during the matching process. On-the-fly edge-matching process is carried out during the service request and it leaves no room for modifications or negotiations. We present in this paper a method for performing on-the-fly edge-matching for linear features in a multinational cascading service environment. The proposed algorithm uses a country boundary lines data set and a connecting feature points data set that contains pre-determined locations on the boundary lines where the features should be matched. The results show that the proposed approach is suitable for most edge-matching situations where the connecting feature points are available. Problematic cases include situations, where the features are alternating across the border and cases where the features don't reach near enough to the border line. In the proposed approach, the edge-matching is executed simply by moving selected line end points to the locations of connecting feature points. In future, the process could be improved by adopting conflation methods that produce visually smoother results and extending the edge-matching functionality to cover also polygonal features.
Keywords: Edge-matching, linear features, cascading services

1 Introduction

Cascading download services combine different services together so that their contents can be queried through a single service. When cascading services are used, the data coming from various background services should be harmonized and edge-matched so that the border-crossing features fit seamlessly together. This is often not the case and the features may overlap at the border areas or they might not connect at the border.

The edge-matching process is usually performed as an iterative off-line process between the mapping agencies of the countries that are sharing the border. The iterative approach offers a possibility for testing and correcting the edge-matching results.

On-the-fly edge-matching provides a different setting, where the edge-matching process is executed during the service request. The on-the-fly matching process doesn't leave any room for checking or adjusting the results

1.1 Previous work

Edge-Matching can be categorized into horizontal conflation methods that aim at removing differences between maps in their shared area (Yuan & Tao, 1999). (Ruiz et al, 2011) have created a comprehensive review and classification of different conflation processes. (Wiemann & Bernard, 2010) have studied methods for conflation in a web service environment. (Samal, Seth & Cueto, 2004) have studied feature matching in a multiple data source setting, where they used a graph-based approach for conflation.

In the European Location Framework (ELF) project, (Brühl, 2015) has described an edge-matching process that uses connecting feature points that are located on the agreed international boundary lines. The specifications for the ELF international boundaries and for the ELF connecting feature points have been defined in (Brühl 2013a) and (Brühl 2013b).

Different edge-matching tools were also tested during the ELF project (Warner et al, 2017). The tested tools include 2 commercial products that utilize the connecting feature points data and one open-source product that carries out edge-matching for polygonal features. (Kruse, 2017) has provided a description on the implementation of edge-matching process and tools in the ELF project.

This paper is organized as follows: the chapter 2 presents the data sets that were used in the work and introduces the developed edge-matching process together with the operational environment where a demonstration service was set up. The chapter 3 presents the results of the work. The paper ends with discussion.

2 Data and Methods

2.1 Data

The on-the-fly edge-matching process was implemented as a component within the ELF Cascading Web Feature Service (WFS) (Lehto, 2017). The Cascading WFS combines European download services from 13 countries and more than 120 feature types. At the time when the work was executed, the service coverage of the ELF Cascading WFS was not yet complete and there were only a few cases where data was available from the same feature type in neighboring countries.

The developed edge-matching approach is based on the ELF international boundaries and the ELF connecting feature points data sets. The international boundaries data set consists of the agreed border lines between various European countries and it is available in two levels of detail, master and regional.

The connecting feature points data contains the locations on the international border lines where the features, that are coming from the neighboring countries, should be matched. The connecting feature points data is also available in master and regional levels of detail. The master level points were marginally off the international border lines and they were moved to the borders as a pre-processing step with the PostGIS database's snapping functionality.

The on-the-fly edge-matching implementation work was decided to be restricted to the handling of linear features. Feature types: RoadLink and Watercourse were selected for test feature types from the Cascading WFS contents because they both are linear and potentially cross the borders between countries.

Two test areas were selected for the edge-matching. The main selection criterion was the availability of the data from the selected feature classes from neighbouring countries. First test area included Finland and Norway and contained master level international boundaries and master level connecting feature points data from the Watercourse feature type. The second test area included France and Spain and comprised of regional level international boundaries and regional level connecting feature points data from the Watercourse and RoadLink feature types.

For executing point-in-polygon operations in the edge-matching process, we created country polygons from the geometries of the ELF international boundaries dataset. The country polygons were created manually with the QGIS application for Finland, France, Norway and Spain.

2.2 On-the-Fly Edge-Matching Process

The edge-matching process is executed on-the-fly during the processing of the ELF Cascading WFS's GetFeature request. The Cascading service gathers the features that are within the query's bounding box window from the background services in the Geography Markup Language (GML) format and forwards them to the edge-matching component.

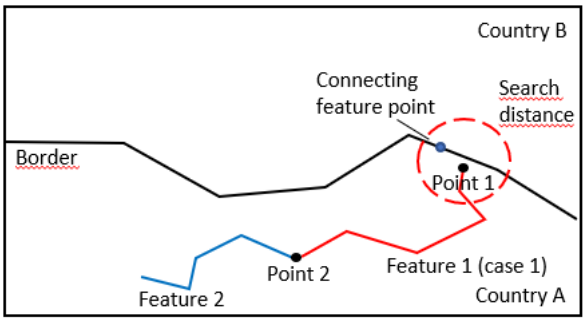
The edge-matching process is performed only for geometries, not for any other feature attributes. The various calculations that are executed during the edge-matching are performed with functionalities provided by the PostgreSQL database's spatial extension PostGIS. After the edge-matching has been completed, the response of the cascading WFS consists of the same GML features that were retrieved from the background services, with possibly modified geometries.

The edge-matching process can be fine-tuned by setting a custom value for the connecting feature point search distance. The value of 25 meters was used in this work.

In the beginning of the edge-matching process, the country polygons are used for calculating for the both end points of each line feature, whether they are inside the polygon of the country where the dataset originates from. Further processing is separated into 4 cases, whose execution is based on the results of the point-in-polygon operation.

2.2.1 Case 1

Figure 1: Edge-matching situation in case 1.



The case 1 processing (Figure 1) is executed for features that have both end points inside the country, where the dataset originates from (Country A). The processing is carried out with the following sequence (example processing for Feature 1):

- Check for feature's both end points (Points 1 and 2), whether the line continues from that point with another line that originates from the same country.
- If the feature continues from the point in question (Point 2), with another line (Feature 2) the point will not be edge-matched because it could break the connectivity in the data.
- For the line end points that passed the previous test (Point 1), search if any connecting feature points are within the specified connecting feature point search distance.
- If connecting feature points are found, match the point to the nearest connecting feature point by moving the point to its location.
- If both line end points are to be matched to the same connecting feature point, match only the point that is closer to the connecting feature point. If the distances are equal, match only the line's start point.

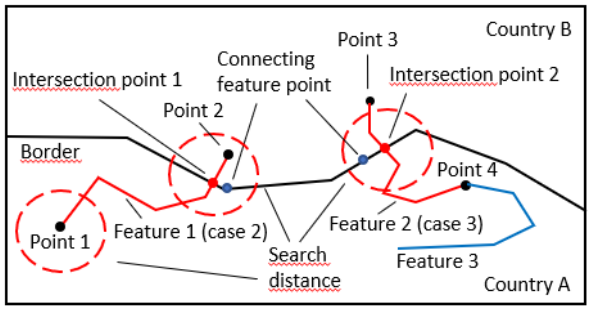
2.2.2 Case 2

The case 2 processing (Figure 2) is executed for features that have the start point inside the country, where the dataset originates from (Country A) and the end point in another country (Country B). The processing is carried out with the following sequence (example processing for Feature 1):

- Find the intersection points between the feature and the country border line.
- Shorten the last segment of the feature that is in a foreign country to the border line (from Point 2 to Intersection point 1).
- For the line's new end point (Intersection point 1), search if any connecting feature points are within the specified connecting feature point search distance.
- If connecting feature points are found, match the point to the nearest connecting feature point by moving the point to its location.

- Check for the line's start point (Point 1), whether the line continues from that point with another line that originates from the same country.
- If the line continues from the point in question with another line, the point will not be processed further because it could break the connectivity in the data.
- If the line's start point passed the previous test, search if any connecting feature points are within the specified connecting feature point search distance.
- If connecting feature points are found, match the feature's start point to the nearest connecting feature point by moving the start point to its location if it will be matched to a different connecting feature point than the line's end point.

Figure 2: Edge-matching situations in cases 2 and 3.



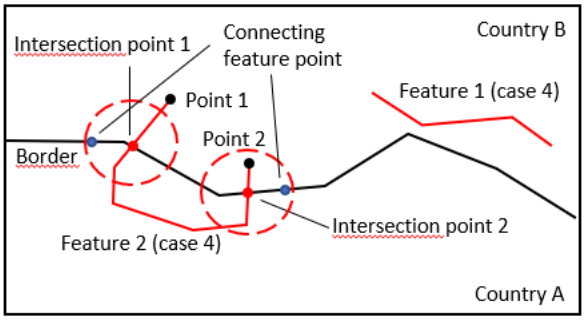
2.2.3 Case 3

The case 3 processing (Figure 2) is executed for features that have the end point inside the country where the dataset originates from (Country A) and the start point in another country (Country B). The processing is carried out with the following sequence (example processing for Feature 2):

- Find the intersection points between the feature and the country border line.
- Shorten the first segment of the feature that is in a foreign country to the border line (from Point 3 to Intersection point 2).
- For the line's new start point (Intersection point 2), search if any connecting feature points are within the specified connecting feature point search distance.
- If connecting feature points are found, match the point to the nearest connecting feature point by moving the point to its location.
- Check for the line's end point (Point 4) whether the line continues from that point with another line that originates from the same country.
- If the line continues from the point in question with another line (Feature 3), the point will not be processed further because it could break the connectivity in the data.
- If the line's end point passed the previous test, search if any connecting feature points are within the specified connecting feature point search distance.
- If connecting feature points are found, match the feature's end point to the nearest connecting feature point by moving the end point to its location if it will be matched to a different connecting feature point than the line's start point

2.2.4 Case 4

Figure 3: Edge-matching situations in case 4.



The case 4 processing (Figure 3) is executed for features that have both end points outside the country, where the dataset originates from (Country A). The processing is carried out with the following sequence:

- Find the intersection points between the feature and the country border line.
- If there are no intersection points (Feature 1), modify the feature's geometry element to be empty. This removes the geometries that are totally outside the country of origin from the Cascading WFS's output.
- If intersection points exist (Feature 2):
- Shorten the line's start segment that is in a foreign country to the border line (from Point 1 to Intersection point 1).
- For the line's new start point (Intersection point 1), search if any connecting feature points are within the specified connecting feature point search distance.
- If connecting feature points are found, match the new start point to the nearest connecting feature point by moving the point to its location.
- Shorten the line's end segment that is in a foreign country to the border line (from Point 2 to Intersection point 2).
- For the line's new end point (Intersection point 2), search if any connecting feature points are within the specified connecting feature point search distance.
- If connecting feature points are found, match the new end point to the nearest connecting feature point by moving the point to its location.

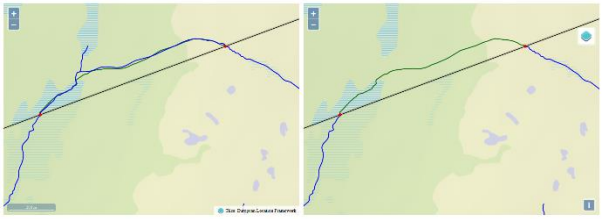
2.3 Demonstration Client

A demonstration client (Figure 4) (OpenELS, 2018) was implemented in the project with the OpenLayers JavaScript library to visualize the results of the on-the-fly edge-matching process. The client contains several edge-matching situations from the study areas and it includes two map windows: the left-hand side window shows the cascading WFS output

without edge-matching and the right-hand side window the output where the edge-matching process has been executed.

The demonstration client contains the ELF Basemap layer (Laurent, 2016) as a background layer and the international boundaries and the connecting feature points layers as overlays.

Figure 4: Demonstration client.



2.4 Operational Environment

The components used in the demonstration client, including database, web application server and web server were installed to a virtual server running on an OpenStack-based cloud service hosted by the Finnish IT Center for Science.

Two versions of the cascading service were installed, one with and one without edge-matching functionality. We used an extension parameter "EM" in the GetFeature query to indicate, whether the edge-matching should be executed or not. We also created a front-end servlet module that interprets the service queries and forwards them to the specific cascading service depending on the value of the "EM" parameter.

3 Results

The presented edge-matching algorithm is strongly reliant on the availability of the connecting feature points data. The algorithm can match the line features with the connecting feature points in most cases where the points are available.

The connecting feature points data used in the project was limited both in its coverage and completeness. There were many border-crossing features that didn't have any connecting feature points near them. In contrast, there were points available that didn't have corresponding features in the national services.

The principle of cutting the line ends that are in the foreign country to the border, removes most of the data overlapping from the edge-matched output. Some overlapping remains for certain features, such as rivers that cross the border line multiple times in a single line feature. We decided to not cut these features completely with the border line and only shorten the line ends that are in the foreign country to the border. The complete cutting could break the features, if there are not corresponding features available in the neighbouring country's data set.

The cutting of the ends of the features that are following along the border and crossing it multiple times may also lead to problem where the line may pass near a connecting feature point and not be matched into it. Problems arise also in situations where the data in the national download services don't reach close enough to the border line.

4 Discussion

The presented edge-matching process is performed by moving the selected end points of linear features to the locations of the connecting feature points. This approach creates the connectivity between the features that come from different services but sometimes leaves sharp angles to the processed features. In the future, the process could be improved by using methods that produce more gradual changes to the edge-matched features.

The operational use of the edge-matching process would require that international boundaries and connecting feature points data are available between all countries that share a border and have services connected to the cascading WFS. Another requirement would be the creation of country polygons from these countries. Also, the performance of the edge-matching solution should be improved for operational use.

Currently, the edge-matching is performed only for linear features. Another possible future development would be to extend the edge-matching functionality to handle also polygonal features.

References

Brühl, M. (2013a) ELF D 2.7: ELF International Boundaries. [Online] Available from: http://www.elfproject.eu/sites/default/files/D2.7%20ELF%20InternationalBoundaries%20v1.1.pdf [Accessed 20th March 2019].

Brühl, M. (2013b) ELF D2.7-1: ELF International Boundaries - Data Specification. [Online] Available from: http://www.elfproject.eu/sites/default/files/D2.7%20ELF%20InternationalBoundaries%20v1.1.zip [Accessed 20th March 2019].

Brühl, M. (ed.) (2015) ELF D2.3.3 Edge Matching. [Online] Available from: http://elfproject.eu/sites/default/files/D2.3.3_EdgeMatching.pdf [Accessed 20th March 2019].

Kruse, D. (ed.) (2017) D4.10: Edge Matching tools and Edge Matching process Implementation Report. [Online] Available from: http://elfproject.eu/sites/default/files/D4.10%20Implementation%20report%20on%20Edge%20matching%20tool.pdf [Accessed 20th March 2019].

Laurent, D. (ed.) (2016) ELF D 2.4: E.L.F. Basemap. [Online] Available from: http://elfproject.eu/sites/default/files/D2.4%20Product_Service%20specification%20for%20ELF%20basemap%20v2.0.pdf [Accessed 20th March 2019].

Lehto, L. (2017) Cascading Geospatial Content Services CASE: European Location Framework. In: *GEOProcessing 2017: The Ninth International Conference on Advanced Geographic Information Systems, Applications and Services.*

OpenELS (2018) OpenELS 2.5 "On-the-fly Edge-Matching Service" Demonstration. Available from: http://193.166.25.14/OpenELS-EM-Demo.html [Accessed 2nd April 2019].

Ruiz, J. J., Ariza, F. J., Ureña, M. A. & Blásquez, E. B. (2011) Digital Map conflation: a review of the process and a proposal for classification. *International Journal of Geographical Information Science, 25:9, 1439-1466, DOI: 10.1080/13658816.2010.519707*

Samal, A., Seth, S. & Cueto, K. (2004) A feature-based approach to conflation of geospatial sources. *International Journal of Geographical Information Science, 18:5, 459-489, DOI: 10.1080/13658810410001658076*

Warner, D., Sunderland, R., Hardy, P., Ledoux, H. & Kruse, D. (2017) ELF D4.4: Edgematching. [Online] Available from: http://elfproject.eu/sites/default/files/D4.4%20Edgematching%20tools.pdf [Accessed 20th March 2019].

Wiemann, S. & Bernard, L. (2010) Conflation Services within Spatial Data Infrastructures. In: *13th AGILE International Conference on Geographic Information Science 2010, Guimarães, Portugal.*

Yuan, S. & Tao, C. (1999) Development of Conflation Components. In: *Proceedings of Geoinformatics, Ann Arbor, 1-13.*