



“Gheorghe Asachi” Technical University of Iasi, Romania



A METHODOLOGY FOR SUPPORTING EXPLORATORY RESEARCH ON THE ROLE OF GEO-LOCATION AND BOUNDARIES IN SPATIO-TEMPORAL AND ENVIRONMENTAL STUDIES

Daniel Homocianu¹, Aurelian-Petruș Plopeanu^{2*}, Alin Adrian Mihăilă³,
Emil Lucian Crișan³, Răzvan Liviu Nistor³, Dinu Airinei¹

¹Department of Accounting, Business Information Systems and Statistics, Faculty of Economics and Business Administration (FEAA), Alexandru Ioan Cuza University of Iași (UAIC), Carol I Blvd., No.22, 700505, Iași, Romania

²Institute of Interdisciplinary Research, Department of Social Sciences and Humanities, UAIC, Lascăr Catargi Street, No.54, 700107, Iasi, Romania

³Department of Management, Faculty of Economics and Business Administration (FSEGA), Babes-Bolyai University of Cluj-Napoca (UBB), Teodor Mihali Street, No.58-60, 400591, Cluj-Napoca, Romania

Abstract

The paper brings a methodological contribution useful for supporting the development of predictive models of phenomena and human behaviours in the area of environmental protection with respect for history, geo-location, environment boundaries and considering further applications in defining corresponding stimulating policies by decision makers and management teams. It proposes a handy and easy-to-use technique of deriving spatial variables (e.g. aerial distances to a custom border and belonging or not to a chosen area) starting from real data including respondents' residencies from a survey of Romanian students. The authors used Quantum Geographical Information System (QGIS) and a historical map for recreating the coordinates of this border two programming languages and two categories of algorithms: first for computing a minimum aerial distance to this custom border based on Haversine (HAV) formula and a proposed Point to Points (P2P) approach compared with another one based on the Pythagorean Theorem with corrections depending on the average latitudes using both P2P and P2S (Point to Segments); second for finding the belonging to a chosen / defined area considering the corresponding polygon and the Crossing-Number (CN) approach of solving Point-in-Polygon (PIP) problems. Aerial distance scale derivations for statistical purposes, online query-able map examples for demonstrating the expected spatial alignment of the inputs based on this methodology and execution benchmarks for efficiency reasons have been also provided.

Keywords: border studies, GIS, Haversine and Pythagoras, P2P and P2S, PIP

Received: September, 2019; Revised final: February, 2020; Accepted: April, 2020; Published in final edited form: July, 2020

1. Introduction and literature review

This study proposes a handy and useful technique involving minimum technical abilities, in the context of many existing methods for computing fields related to border proximity.

Geography was probably the first discipline to study borders, one of the possible outcomes of initial border research being related to the justification of

territorial claims (Kolossoy, 2005). However, researchers from various fields are now interested in studying borders for different problems: ecologists for finding the role of borders in determining the human impact upon ecosystems (Graybill, 2014); economists for discovering links between previous and current economic conditions and current and historical borders (Becker et al., 2015); cultural researchers for observing effects of migration, movement of people,

* Author to whom all correspondence should be addressed: e-mail: aplopeanu@gmail.com; Phone: +40232201413; Fax: +40232217000

and the spread of cultural ideas; political and geographical ones for analysing them in a more interdisciplinary approach to observe changing societal power relations in terms of territoriality; anthropologists for finding different human patterns in relation to social categories such as ethnicity (Paasi, 2009); linguists and others for studying people, places, objects, goods, ideas and experiences that exists in a particular space and time (Lünen and Travis, 2012). Borders are recognized as a complicated social phenomenon, one which influence fundamental basis of the organization of society and human psychology.

Research on borders is also achieved using Historical Geographical Information Systems (HGIS), especially due to the informational capabilities they have. They can store attribute and spatial data related to any point that exist on Earth. This approach is not accepted by all researchers since it involves complex methodologies. These methodologies need a major time investment for being mastered by new researchers. Considering this problem, we have developed two more accessible categories of algorithms, which can be used by all investigators in performing research related to historical and current borders. Both are developed and implemented using a well-known data oriented Integrated Development Environment (IDE), namely Visual Basic for Applications (VBA) under Microsoft Excel: (1) the first one is built for the biaxial computation of the minimum aerial distance, from each pair of coordinates from a set of points to a border as a pre-established or regenerated set of coordinates mainly based on the "half versed sine" (HAV) technique; (2) the second is useful for determining the belonging of each pair to a larger polygon, including the border as a polyline, based on the CN approach of the PIP problem.

The paper validates the core hypothesis that for reasonable aerial distances to custom borders (up to hundreds of Kilometres, as resulting from our data) and detailed sets of coordinates generated for them, both the P2P and P2S approaches and also both the Haversine formula and the Pythagorean Theorem are appropriate to use and bring almost similar results. The results provided for both categories of algorithms are based on the case of the historical border between the former Romanian Kingdom and that part of Romania under the occupation of the long-gone Austro-Hungarian Empire (AHE). This former border existed more than one hundred years ago and divided Romania into two parts with a balanced coverage in the structure of all 3051 responses collected - a ratio of almost 8 to 7 in favour of the respondent's residences belonging to the former AHE. Both categories of algorithms are proved as very intuitive, easy to use and understand and they deliver good results even with few processing resources. They can be further used to build interactive and query-able representations under the form of pushpin-based maps of corresponding coordinates and support studies focused on the influence of proximity.

Border related research has evolved in the last

years due to the appearance of advanced technologies and increased processing capabilities. In an investigation that analyses the historical perspective on border studies and mapping, Kolossov (Kolossov, 2005) identifies multiple changes regarding border research paradigms and also the term of limology for the study of borders as an interdisciplinary field. Consequently, different specialists pursue different goals and the tools they are using are very diverse. More, the number of borders increased with the collapse of the Soviet Union, Yugoslavia and some other countries (Paasi, 2009) and multiple territorial changes continue to influence the international relations while the focus is no longer only on linear borders, but also on multiple theoretical borders influenced by world's dynamics as structural changes.

The global paradigm concerning border research is now more complex covering different aspects raised by the replacement of virtual lines with economic, cultural and other boundaries. Therefore, this paradigm may become a separate interdisciplinary field (Kolossov, 2005). Border research has evolved from the analysis of border as a barrier line to that of "borderlands" as a social construct, reflecting the pronounced phenomena of economic, social and cultural globalization (Arieli, 2016). Such elements are very stable, despite frequent changes of boundaries (Kolossov, 2006). Boundaries can also be regarded as diverging sets of contextual performances (Paasi, 2005).

The usage of GIS in border research has extended also in other fields. Two researchers (Gregory and Healey, 2007) observe that for historical geography. Although mapping is one key capability of GIS, they can also store a lot of data (text, images and movies) linked to coordinates on Earth and connect spatial (point, line or area) with attribute data (categories, ranks, counts, amounts, ratios etc.) to be used in multiple ways for mapping events and spatial analysis with different variables (Bailey and Schick, 2009). Their growing sophistication allowed them to combine the modern computing power with the ability to integrate all three components of data: attribute, time and space (Gregory, 2002). Many European countries / regions created GIS tools for their changing administrative boundaries as first step to understand the spatial and temporal complexities of long-term socio-economic change (Gregory and Ell, 2006).

The use of historical maps in GIS is challenging, as long as an increased level of accuracy is required (Levin et al., 2010). Some authors (Bailey and Schick, 2009) have performed a study that analyses GIS effectiveness in historical research and observed the same huge potential of GIS, which is able to expand spatial perspectives in other disciplines and incorporate the spatial visual insight. Others (McLeman et al., 2010) use a complex qualitative model based on GIS to analyse drought adaptation and migration impact in time. Some mentioned above (Lünen and Travis, 2012) provide evidence that the old GIS usage of spatial and attribute data for mainly quantitative research is surpassed by new

developments in GIS-related science: spatial multimedia, GIS-enabled web services, geo-visualization, cyber-geography, exploratory spatial data analysis, and virtual reality, all of them providing capabilities that exceed the abilities of GIS on its own. Stanev (Stanev, 2013) presents the impact that history and geography had on railway development in Balkans between 1870–2001 based on data from more than 60 statistical publications from 14 different states later incorporated within HGIS and grouped into different layers related to: population of towns and cities, main railways lines constructed and a geographical database that included all boundary changes that had taken place. Another author (Knowles, 2014) argues that HGIS is technically demanding for historians and few main-stream historians use GIS and HGIS.

The pros identified in literature for HGIS are mainly related to their power to identify historical patterns and relationships that otherwise remain invisible and suitability for both quantitative and qualitative variables while the cons mainly refer to their rigidly structured database projects which could determine a limited use.

From another perspective, namely the Environmental GIS (EGIS), according with Clifford et. al (2016), EGIS refers to the use of Geographic Information Systems in environmental science and management, one of the fundamental aspects of EGIS being to collect, analyse and create new layers with environmental data and test hypotheses in science. A good example of applying GIS principles and methods in environmental research is provided in a scientific paper (Buta et al., 2019) dedicated to assessing the spatial variability of the apparent electrical conductivity of saline soils. These authors also mention the spatial variability maps they generated by using a geo-statistical method. Another such example is available in a new scientific paper (Sestraş et al., 2019a) dedicated to the presentation of a GIS based digitization for a fruit research station benefiting from the great advantage of a fully integrated land information system. The same authors above argument some benefits of using both statistical and GIS methods in environmental research through a study (Sestraş et al., 2019b) dedicated to assess the dynamic potential of a territory. In another research related with environmental issues, the authors (Houessou-Dossou et al., 2019) determined the return periods of major floods in a particular region using rainfall frequency analysis and a Participatory Geographic Information System (PGIS) as a tool able to collect data and support decision making in this particular field. More, Nishitateno and Burke (2020) assessed the role of registration restrictions in improving the quality of the urban air in Japan based on a major data source provided by the Environmental GIS and compiled by the National Institute for Environmental Studies (NIES) and analyses using Difference-in-Differences (DD method) statistical approach.

Our paper empowers further studies regarding

borders by providing simple ways of performing custom border definition, determination of belonging to resulting areas and calculations of aerial distances to it. These results and the entire method can be further used to analyse the impact of the closeness to a border on other variables (environmental, economic, cultural, political, etc.) and identify different boundaries and cross-boundary interactions. This paper provides to researchers in ecosystems, history, geography, economics, social sciences and humanities and many others, an accessible alternative of how GIS data can be used in different investigations. This way, it fills an already identified gap (Gregory and Healey, 2007), contributing to the increasing usage of GIS in historical and environmental geography and other fields. By improving accessibility to GIS through our simple algorithms, users will be capable of performing more research and increase their trust in related methods and tools.

2. Material and methods

We have started this study from a dataset as result of a survey designed considering the influence of the aforementioned historical border on today Romanian students' various intentions and attitudes. One of those was that corresponding to a question asking about the clear intention (*clear intent*) to involve in actions concerning environmental protection and renewal in the near future. This question corresponds to a variable which could be analyzed depending on both the rest of the questionnaire items and afferent variables, acting as input ones, and the additional / external / derived ones, including those related with geo-location. The corresponding dataset we started from comprises 3051 unique responses, obtained through a printed questionnaire distributed during the previous university year to students in economics, who studied at that time at Alexandru Ioan Cuza University of Iasi (UAIC), Babes-Bolyai University of Cluj-Napoca (UBB) and Lucian Blaga University of Sibiu (ULBS), and who came from different cities and villages, their residencies being wide spread over Romania (geographical positions are indicated in the next section - Fig. 5a).

We paid respect to a set of scientific principles including: 1) the one of transparency and support for replication of results because of combining highly suggestive visual representations with online support as files containing all the data, source code and configurations we have used; 2) support for triangulation in research by paying respect to the use of multiple approaches and instruments to address the same scientific question and get similar results across different methods which are less likely to generate a false image regarding the specific purpose of the research investigation (Munafò and Smith, 2018); 3) support for interactivity by emphasizing many processing states in the same representation including all together tabular data and invoked functions in some cases and maps together with various filters and

numerical results in others (Plopeanu et al., 2018); 4) the well-known principle which states that a good picture paints a thousand words with examples starting from the impact of visual mining when building reporting solutions, or the one of distribution maps when assessing the statistical representativeness of different data samples.

Our primary goal for this research was to setup a technique to determine whether students' residencies belong to a selected historical area or not and, next, to determine the aerial distance to a related particular border. In another research, Plopeanu and Homocianu (2017) already demonstrated a simple way to automatically compute road distances from the full residences of each respondent (1400 cases) as short text blocks (e.g., "Paşcani, Iaşi, Romania" as city/town or village, county or district and country, stored in the column MD, from MD2 to MD1401) to a certain point of interest such as the city where a certain university centre is located (e.g. "Cluj-Napoca, Cluj, Romania" to store in a single cell such as ME\$2), using the *importXml* powerful function from Google Sheets and an Application Programming Interface (API) from Google, namely Directions which belongs to the Google Maps Platform, e.g. `=importXml(http://maps.googleapis.com/maps/api/directions/xml?origin="&MD2&"&destination="&ME$2&"&sensor=false&alternatives=false",//length/distance/value"/1000`. Consecutive auto fill, copy followed by paste special as values and sort options corresponding to the formula above and its results have been also used to ensure persistence and avoid duplicate requests and waiting times from Google servers for the same input. Moreover, in another research paper, Homocianu and Airinei (2017) improved a cloud-based generator of portable, auto-guided and interactive virtual road routes, originally called GPS4GISVR - General Purpose System for Generating Interactive Street View Routes (Homocianu and Airinei, 2016). The results of using GPS4GISVR were in fact portable documents with a lot of pages (animation sensation at scroll in the presentation mode of most .pdf viewers), each having info about the current location (e.g. the nearest address in the form of city, street and street number and the road distance traveled up to the current point expressed in Km), a 2D map indicating the current point on the entire route (also highlighted) and a 3D image captured from and linked to the interactive Street View navigation mode of Google Maps. As technique, such 3D images have been extracted with respect for a simple rule, namely always look ahead and progressively enter curves in order to minimize the user's loss of visual perspective mostly when meeting wide angle turns (e.g. a 90 degrees one).

Our current general technique involves firstly to transform respondents' residencies into corresponding pairs of coordinates, then regenerate a custom polygon (e.g. the one afferent to the historical area that previously belonged to the no longer existing selected area above, including the custom aforementioned historical border) also as pairs of

coordinates, and finally derive at least two additional fields: first for determining if a given respondent's location belongs or not to the interior of the resulting polygon (new generated dummy variable) and second for computing the aerial distance to the considered border as a custom polyline (new generated continuous variable to be further translated into a scale-based one).

To automatically extract the precise coordinates of the respondents' residencies, we have used the same aforementioned function (*importXml*) and another (API) from Google, namely Geocoding, also belonging to the same Google Maps Platform, e.g. `=importXml("http://maps.googleapis.com/maps/api/geocode/xml?address="&C2,"/GeocodeResponse/location")`. The input cells being referenced (all 3051 starting from C2) have been used to store the full residences for each respondent (the aforementioned format above as city / town / village, county / district and country). No API key was required at the moment of extracting these coordinates (Plopeanu et al., 2018).

For regenerating the coordinates of this particular border as a polyline, we have used QGIS 3.4.1 Madeira, together with the Geospatial Data Abstraction Library (GDAL) plugin for importing, calibrating and converting the historical raster map (image provided by the Ministry for Culture and Heritage from New Zealand, Map of the Kingdom of Romania in 1916, updated on August 28, 2014, <https://nzhistory.govt.nz/media/photo/map-kingdom-romania-1916>) and the Open Street Maps plugin for handling geospatial data. To form a polygon that contains the part of Romania that belonged to the former AHE until one hundred years ago, we have added the coordinates corresponding to a part of the current Romania's border (almost the entire Northern, North-Western and South-Western extremities of Romania - NNWSW) to the ones of the related historical border's polyline. Those corresponding to the current Romania's border were collected from an official source (Crăciunescu, "Polygon for the Romanian border", updated in 2016, <http://www.geospatial.org/download/romania-seturi-vectoriale>).

Given the large computing power available today, and the critical need for intuitiveness and ease of understanding of the logic of any proposed implementation, by any researcher interested in spatial analysis, we have chosen to use VBA under Microsoft Excel for deriving those two additional fields. Excel is well-known as an easy-to-learn and user-friendly environment, while VBA is an IDE with increased capabilities to process worksheet data. The VBA's usage offers many advantages related to Excel's wider applications and well-known environment (Afzal et al., 2011). For a clear comparison with other IDEs, parallel speed tests were performed using implementations in C++ under Microsoft Visual Studio Ultimate (C++MVSU).

Two categories of algorithms have been built to obtain the above-mentioned derived fields. For the first category, represented by computing the minimum aerial distances from given data points (respondents'

location) to a border translated into the concept of complex polyline, we used HAV (van Brummelen, 2012) - Eqs. (1-2) and the P2P measuring approach (points as the most atomic parts of the border's polyline), with better results for polylines with a reasonable accuracy (e.g. at least 1/100 or from Km to Km for a polyline corresponding to a distance of 100 Km).

$$HAV = \sin^2(\Delta\varphi/2) + \cos\varphi_1 * \cos\varphi_2 * \sin^2(\Delta\lambda/2) \quad (1)$$

$$d_{HVS} = R * 2 * a \sin(\sqrt{HAV}) \quad (2)$$

where, *HAV* is the Haversine formula; d_{HVS} is the aerial distance computed with the HAV formula; *R* is the Earth's Radius, $R=6367$ Km; φ_1 and φ_2 are the latitudes of the points 1 and 2 converted to Radians; $\Delta\varphi$ is the difference in latitudes: $\Delta\varphi = \varphi_2 - \varphi_1$; $\Delta\lambda$ represents the difference in longitude.

Finally, we compared HAV's results (P2P) with the ones resulting after using Pythagorean Theorem, and both P2P and P2S. The latter is actually a point to segments measuring approach. It considers segments as the smaller parts of border's polyline. It has good results for both high and low accuracy polylines (Stojanovic and Stojanovic, 2014). This is mainly due to permanently aiming the calculation of aerial distance corresponding to the perpendicular (Kimberling, 2002) from a given point to the segment's line - Eqs. (4,6), when this perpendicular intersects the current segment (intersection point belongs to segment's interior / two acute angles at the bottom of the segment, where an angle is acute if an only if the squared opposite edge (Eq. 3), as provided by Daniel Zwillinger in 2012 is less than the sum of the remaining squared edges). When the condition above is not met (intersection point belongs to the outside of the segment / one obtuse angle at the bottom of one of the two segment's extremes, where an angle is obtuse if and only if the squared opposite edge is greater than the sum of the remaining squared edges) the minimum of the two sides is considered - Eqs. (5-6), as the minimum value of the two corresponding aerial distances from the respondents's point to the segment's extremes. In fact, this means that for an obtuse angle - *C* between $\pi/2$ and π (Eq. 3), the cosine is negative and consequently the squared opposite edge (c^2) becomes greater than the sum of the remaining squared edges ($a^2 + b^2$) and vice versa.

$$c^2 = a^2 + b^2 - 2*a*b*\cos(C) \Rightarrow \cos(C) = (a^2 + b^2 - c^2) / 2*a*b \\ \Rightarrow \text{sign}(\cos(C)) = \text{sign}(a^2 + b^2 - c^2) \quad (3)$$

$$d_{ppdc}(\varphi_1, \lambda_1, \varphi_2, \lambda_2, \varphi_3, \lambda_3) = \frac{(\Delta\varphi_{32} * \Delta\lambda_{21} * \cos(\overline{\varphi_{21}}) - \Delta\varphi_{21} * \Delta\lambda_{32} * \cos(\overline{\varphi_{32}}))}{\sqrt{\Delta^2\varphi_{32} + \Delta^2\lambda_{32} * \cos^2(\overline{\varphi_{32}})}} \quad (4)$$

$$d_{Pitagora}(\varphi_1, \lambda_1, \varphi_2, \lambda_2) = \sqrt{\Delta^2\varphi_{21} + \Delta^2\lambda_{21} * \cos^2(\overline{\varphi_{21}})}, \quad (5)$$

$$d = \begin{cases} d_{ppdc}(\varphi_1, \lambda_1, \varphi_2, \lambda_2, \varphi_3, \lambda_3), & \text{for 2 acute angles at segment extremes} \\ \text{Min}(d_{Pitagora}(\varphi_1, \lambda_1, \varphi_2, \lambda_2), d_{Pitagora}(\varphi_1, \lambda_1, \varphi_3, \lambda_3)), & \text{for one obtuse angle} \end{cases} \quad (6)$$

where *c*, *a*, *b* are the edges of a triangle (they obviously represent positive values) and *C* is the angle opposite to the *c* edge ($C < 180$ degrees or π as sum of those three angles in a triangle - Eq. 3); d_{ppdc} is the perpendicular distance; $d_{Pitagora}$ is the distance computed by applying the Pythagorean Theorem and a correction depending on the average latitudes; *Min* stands for the function computing the minimum value of the two arguments; 1 indicates the respondents' location point and both 2 and 3 indicate segment's extreme points; $\Delta\varphi_{ij}$ is the difference in latitudes between points *i* and *j*: $\Delta\varphi_{ij} = \varphi_i - \varphi_j$; $\Delta\lambda_{ij}$ is the difference in longitudes between points *i* and *j*: $\Delta\lambda_{ij} = \lambda_i - \lambda_j$; $\overline{\varphi_{ij}}$ is the average latitude of points *i* and *j*, namely $\overline{\varphi_{ij}} = \frac{\varphi_i + \varphi_j}{2}$ involved in computing the

corrections depending on the latitudes using the cosine function and this average indicator as argument.

When testing with Pythagorean Theorem (both P2P and P2S approaches), we have performed only the correction (L. McNish, "Latitude and Longitude", updated on November 5, 2018, Calgary Centre of the Royal Astronomical Society of Canada (RASC), <http://calgary.rasc.ca/latlong.htm>) of the differences of longitudes with the cosine of the average latitude. The equations - Eqs. (1-2) behind HAV are used to calculate the distance between two points, not as a linear distance, but the shortest aerial distance on the surface of a sphere. Actually, HAV gives the great-circle/orthodromic distances between two points on a sphere from their longitudes and latitudes, which are slightly different from the minimum distances as lines, especially for long distances (over the order of hundreds of Km). It assumes a spherical Earth and ignores ellipsoidal effects, but remains particularly well-conditioned for numerical computation even at small distances (Manzano-Agugliaro et al., 2013).

The second category is dedicated to find out the belonging to the considered former area, with a direct connection with the concept of complex polygon. The CN technique applied to the PIP problem was considered as a special case of the CN of a graph, given the minimum number of edges crossings in any drawing of the graph in the plane (Buchheim et al., 2008). The PIP problem is emphasized (Kumar and Bangi, 2018) as one of extreme importance in many fields, including geo-science or computational geometry. We formalized it below (Eq. 7) using the CN approach.

$$\text{pip} = \begin{cases} 1(\text{yes/inside}), & \text{when } i_no \text{ is an odd number} \\ 0(\text{no/outside}), & \text{when } i_no \text{ is an even number} \end{cases} \quad (7)$$

where i no stands for the number of intersections for a ray or the number of times a ray starting from a point to be checked (P) crosses polygon's boundary edges delimiting its inside and outside.

Although it is considered a simple geometric problem, the need to find out whether a point is inside or outside a polygon is extremely useful and has received an increased attention during time. We have chosen the CN approach (Eq. 7) because it is usually faster. Other algorithms solving the same problem could encounter difficulties, like singularity conditions, the duration for completing these procedures usually taking very long time (Kumar and Bangi, 2018).

To derive additional scale variables starting from both the distance to the border and the belonging to the area under investigation, we further used the built-in functions and operators.

For highlighting the geographical distribution of original responses, both as they were initially, and separated considering distances to the reconstructed historical border, we have used an query-able pushpin-based map of coordinates corresponding to respondents' location of origin, using the Google Fusion Tables (GFT) experimental tool and following an interactivity-oriented approach. Because we aimed for more persistence in time, we also captured the entire functionality of this interactive map as a short demo (Homocianu, 2018).

3. Results and discussion

Our main prerequisite was to regenerate this

former historical border, considering Romania's nowadays map. We have used a historical map (Fig. 1a) and a specialised GIS tool (QGIS 3.4.1 Madeira - Fig.1b) for retrieving the coordinates for many points indicated on the tool's map. We aimed for a good accuracy due to statistical significance needs. Therefore, we have redrawn this border crossing Romania from North-East (Mihăileni) to South-West (Orşova), by creating 3192 pairs of coordinates, whereas the shortest road distance between these two points is 850 Km (e.g. www.distanta.ro/mihaileni-botosani/ghimes/brasov/orsova). This means, on average, the generation of different coordinate pairs from 266 meters to 266 meters, and still a high accuracy for the custom area reconstruction.

In the GIS tool, we have firstly calibrated the imported raster map (image - Fig. 1a), using the GDAL Geo-reference module and the coordinates of six outer points available on this raster map (the centre of six foreign cities: Niš, Sofia, Debrecen, Czernowitz/Cernăuți, Kishinev/Chişinău and Varna, as provided by the search on www.openstreetmap.org). The transformation type chosen for the raster map was "Thin Plate Spline", and the resampling method used was "Nearest neighbour". Next, we added the Open Street Map Standard layer and we set the level of transparency of the imported map to 50%. When capturing the coordinates and editing the polyline (Fig. 1b), we ensured more accuracy by zooming-in inside the tool's map.

For the geographical coordinates of the NNWSW extremities of the current Romanian border, we have used public data (www.geo-spatial.org) as a filter on the polygon, corresponding to Romania's country border, including only 12109 pairs of coordinates from Mihăileni (North-East) to Orşova (South-West).



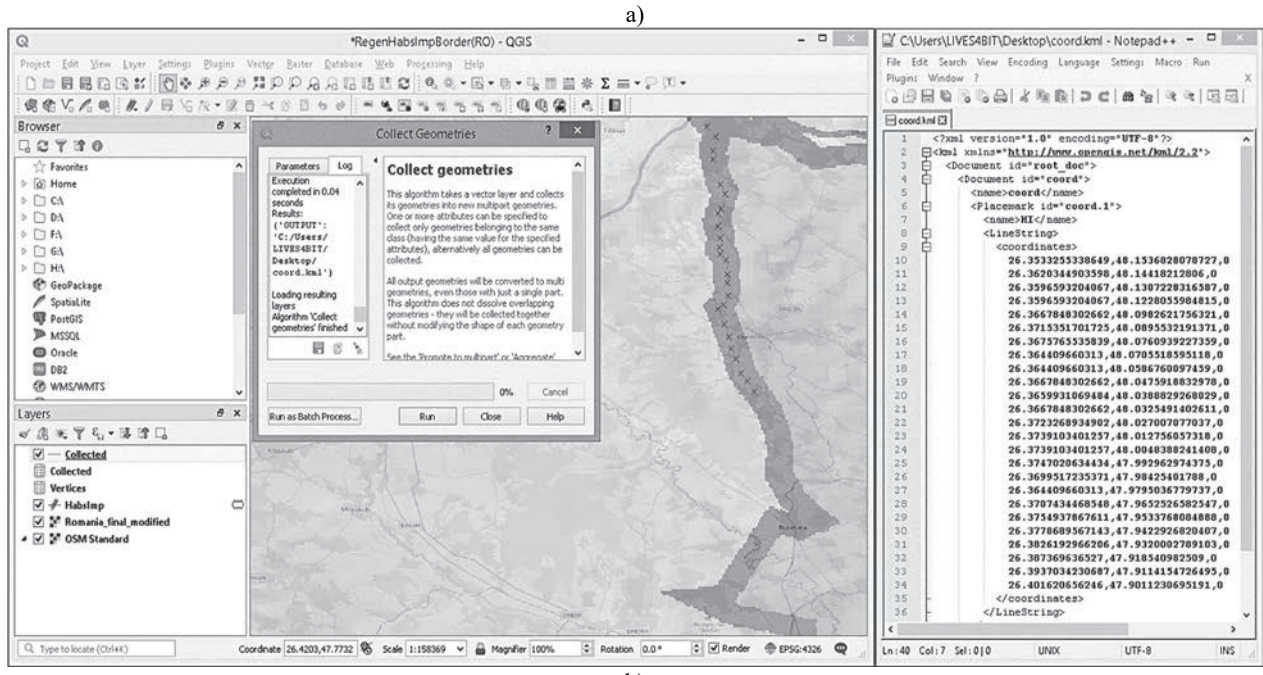


Fig. 1. Using QGIS 3.4.1 Madeira to regenerate coordinates (.kml file) of the former border - main layers: (a) the scanned historical map of Romania and (b) the current Open Street Map

To indicate whether students' residencies belong to the former area under investigation (value of 1) or not (0), a derived dummy field was introduced using the CN approach of the PIP problem (Fig. 3b). Because of our interest in studying the impact of the closeness to historical boundaries, another field was needed to store the minimum aerial distances from residence locations to the former historical border (polyline), using the HAV formula (Nordin, et al., 2012) - Fig. 2b, and not the minimum road distances (many changes in the transportation infrastructure over decades) which are easy to extract using various APIs (e.g. Google Maps JavaScript API- The Geometry Library, <https://developers.google.com/maps/documentation/javascript/geometry>).

To perform the actual computations using the Haversine technique (GetDistKM function in Fig. 2a and 2b), we have benefited from the biaxial design of spreadsheets:

- 3192 pairs of coordinates for the historical border from the .kml file resulting from the GIS export (Fig.1b), initially copied as they are (columns) and then pasted using the transposition option (two rows, namely D1:DRW2);

- 3051 pairs for respondents residencies (A4:B3054) extracted using the IMPORTXML function from Google Sheets;

- a matrix of 9.738.792 cells (3192 * 3051), populated with values generated by invocations of the GetDistKM custom function (Fig. 2a - starting from D4);

- a column (C4:C3054) populated with values generated by invocations of the MIN function (Fig. 2b).

For the calculation of minimum aerial distances, using the custom (GetDistKM - Fig. 2a and Fig. 2b) and the default (MIN) formula-based results in 9741843 cells (those 9.738.792 above in the matrix at the intersection of coordinates and those 3051 in column C for computing the minimum distance), we tested two hardware configurations. We used Windows 8.1 Pro-64 bit and Microsoft Office 2013 Professional Plus 64-bit in the first configuration (<https://erris.gov.ro/ccsirfc>, Equipment's section, Asus ROG G750JY laptop) and Windows 7 Pro-64 bit and Microsoft Office 2010 Professional Plus 64-bit in the second one (<https://erris.gov.ro/ccsirfc>, Equipment's section, Acer Aspire One D255 laptop).

responses & border (each point in polyline)					
	A	B	C	D	E
1		3192 pairs - polyline	border coord.	47.9852111	47.980615
2			D1:DRW2	26.1278915	26.1279798
3		3051 pairs - responses coord. A4:B3054	min.dist.		
4	47.1584549	27.6014418	=MIN(D4:DRW4)	=GetDistKm(\$A4,\$B4,D\$1,D\$1)	=GetDistKm(\$A4,\$B4,E\$1,E\$2)
5	45.6964745	27.184043	=MIN(D5:DRW5)	=GetDistKm(\$A5,\$B5,D\$1,D\$2)	=GetDistKm(\$A5,\$B5,E\$1,E\$2)
6	47.1584549	27.6014418	=MIN(D6:DRW6)	=GetDistKm(\$A6,\$B6,D\$1,D\$2)	=GetDistKm(\$A6,\$B6,E\$1,E\$2)

a)

```

Function GetDistKM(lat1Deg As Double, long1Deg As Double, lat2Deg As Double, long2Deg As Double)
Const Pi = 3.14159265358979
Const R = 6367 'AvgEarthSphereRadiusKM
Dim lat1Rad As Double
Dim long1Rad As Double
Dim lat2Rad As Double
Dim long2Rad As Double
Dim hvs As Double 'Half_versed_sine
Dim inv_hvs As Double
lat1Rad = (lat1Deg / 180) * Pi
long1Rad = (long1Deg / 180) * Pi
lat2Rad = (lat2Deg / 180) * Pi
long2Rad = (long2Deg / 180) * Pi
hvs = Sin((lat1Rad - lat2Rad) / 2) ^ 2
hvs = hvs + Cos(lat1Rad) * Cos(lat2Rad) * Sin((long1Rad - long2Rad) / 2) ^ 2
inv_hvs = 2 * Application.WorksheetFunction.Asin(Sqr(hvs)) 'Arcsin(X)=Atn(X/Sqr(-X*X+1))
GetDistKM = Round(R * inv_hvs, 2)
End Function
    
```

b)

Fig. 2. Applying a custom function (GetDistKM) for computing (P2P approach) minimum aerial distances from respondents' residences to the historical border using the HAV formula: (a) actual use and (b) source code definition

	A	B	C	D	E	F	G	H	I	J
3043	45.59371	20.76972469		45.239101	24.3398608	0				
3044	45.5952	20.76942602		45.7139464	24.1548523	=IF(DotInPoly(D3044,E3044,\$A\$1:\$B\$15301),1,0)				
3045	45.59771	20.76886656		45.0996753	24.3693179	=IF(logical_test, [value_if_true], [value_if_false])				
3046	45.59934	20.76852933		45.7983273	24.1255826	1				
3047	45.60099	20.76815155		45.6696594	24.2626411	1				
3048	45.60285	20.76775783		45.9519939	23.7960039	1				
3049	45.60385	20.76760406		45.7983273	24.1255826	1				
3050	45.60438	20.76756394		45.7983273	24.1255826	1				
3051	45.60579	20.76724343		44.6631799	24.2608711	0				
3052	45.60717	20.76691539								
3053	45.60959	20.7664114								
3054	45.61093	20.7661149								

a)

```

Public Function DotInPoly(X As Double, Y As Double, Poly As Variant) As Variant
Dim dot As Long, HowManyCrossedEdges As Long, slope As Double, intercept As Double, P As Variant
P = Poly
For dot = LBound(P) To UBound(P) - 1
If P(dot, 1) > X Xor P(dot + 1, 1) > X Then
slope = (P(dot + 1, 2) - P(dot, 2)) / (P(dot + 1, 1) - P(dot, 1))
intercept = (P(dot, 2) * P(dot + 1, 1) - P(dot, 1) * P(dot + 1, 2)) / (P(dot + 1, 1) - P(dot, 1))
If slope * X + intercept > Y Then HowManyCrossedEdges = HowManyCrossedEdges + 1
End If
Next
DotInPoly = CBool(HowManyCrossedEdges Mod 2)
End Function
    
```

b)

Fig. 3. Applying a custom function (DotInPoly) to test the belonging of a residence-PIP: (a) use and (b) source code definition

The computation of the minimum aerial distances in VBA under Excel took 13.5 minutes in the first configuration and 40 minutes in the second one. The Excel file containing only the sheet dedicated for computing the minimum such distances and the source code with the function above has grown from 25.4 MB (one invocation of MIN and GetDistKM) to 102 MB (entire column of MIN and matrix of GetDistKM

updated with the actual values). These long processing sessions are compensated by the advantages of further using Excel, which means above all more transparency and almost instant access to all resulting atomic values, in case of exceptions – e.g. the copy and paste special (values) option, followed by automatic searches for a certain value. In addition, the user will benefit at least from three other categories of

advantages when using this environment: a large collection of default powerful functions for various purposes, rich formatting rules for cell values with intuitive visual feedback together with different type of charts and trend extraction possibilities and many add-ins for optimization, representation and data analysis purposes such as: Solver, Power View, Power Map, Power Query, Power Pivot, Data Mining and Analysis ToolPack for statistics.

An additional test made using the first configuration and C++MVSU (source code in Fig. 4)

instead of VBA under Excel, delivered better execution time (45 seconds). In this scenario, we have started from two text files for both respondents' location and borders' coordinates (two tab separated values for each record) and we saved the results (3051 records with the minimum aerial distances) into a text file. However, the approach is more complicated and less intuitive mostly because it does not benefit from visual advantages and common user's experience (e.g. VBA under Excel).

```
#include "stdafx.h"
#include <vector>
#include <iostream>
#include <fstream>
#include <string>
#include <math.h>
#include <cmath>
#define earthRadiusKm 6367.0
#define pi 3.14159265358979
using namespace std;
double deg2rad(double deg) {return (deg * pi / 180);}
double rad2deg(double rad) {return (rad * 180 / pi);}
double GetDistKM(double lat1d, double lon1d, double lat2d, double lon2d)
{double lat1r, lon1r, lat2r, lon2r, u, v;
 lat1r = deg2rad(lat1d);
 lon1r = deg2rad(lon1d);
 lat2r = deg2rad(lat2d);
 lon2r = deg2rad(lon2d);
 u = sin((lat2r - lat1r) / 2);
 v = sin((lon2r - lon1r) / 2);
 return 2.0 * earthRadiusKm * asin(sqrt(u * u + cos(lat1r) * cos(lat2r) * v * v));}
int main()
{int i = 1;
 string Xr[3051];
 string Xb[3192];
 string x, y, d;
 string Yr[3051];
 string Yb[3192];
 double min_d, atomic_d;
 std::ifstream infile1r("D:/3051responses.txt");
 std::getline(infile1r, x, '\t');
 std::getline(infile1r, y, '\n');
 while (infile1r)
 {Xr[i] = x;
 Yr[i] = y;
 std::getline(infile1r, x, '\t');
 std::getline(infile1r, y, '\n');
 i++;}
 i = 1;
 std::ifstream infile2b("D:/3192border.txt");
 std::getline(infile2b, x, '\t');
 std::getline(infile2b, y, '\n');
 while (infile2b)
 {Xb[i] = x;
 Yb[i] = y;
 std::getline(infile2b, x, '\t');
 std::getline(infile2b, y, '\n');
 i++;}
 ofstream outfile3d;
 outfile3d.open("D:/out.txt");
 for (int i = 1; i < 3052; i = i + 1)
 {min_d = GetDistKM(atof(Xr[i].c_str()), atof(Yr[i].c_str()), atof(Xb[1].c_str()), atof(Yb[1].c_str()));
 for (int k = 1; k < 3193; k = k + 1)
 {atomic_d = GetDistKM(atof(Xr[i].c_str()), atof(Yr[i].c_str()), atof(Xb[k].c_str()), atof(Yb[k].c_str()));
 if (atomic_d < min_d) {min_d = atomic_d;}}
 outfile3d << to_string(min_d) << endl;}
 outfile3d.close();
 return 0;}
```

Fig. 4. Applying a similar custom C++ function to compute minimum aerial distances (Eq. 2 – Haversine) from respondents' residences to the historical border (data in text files)

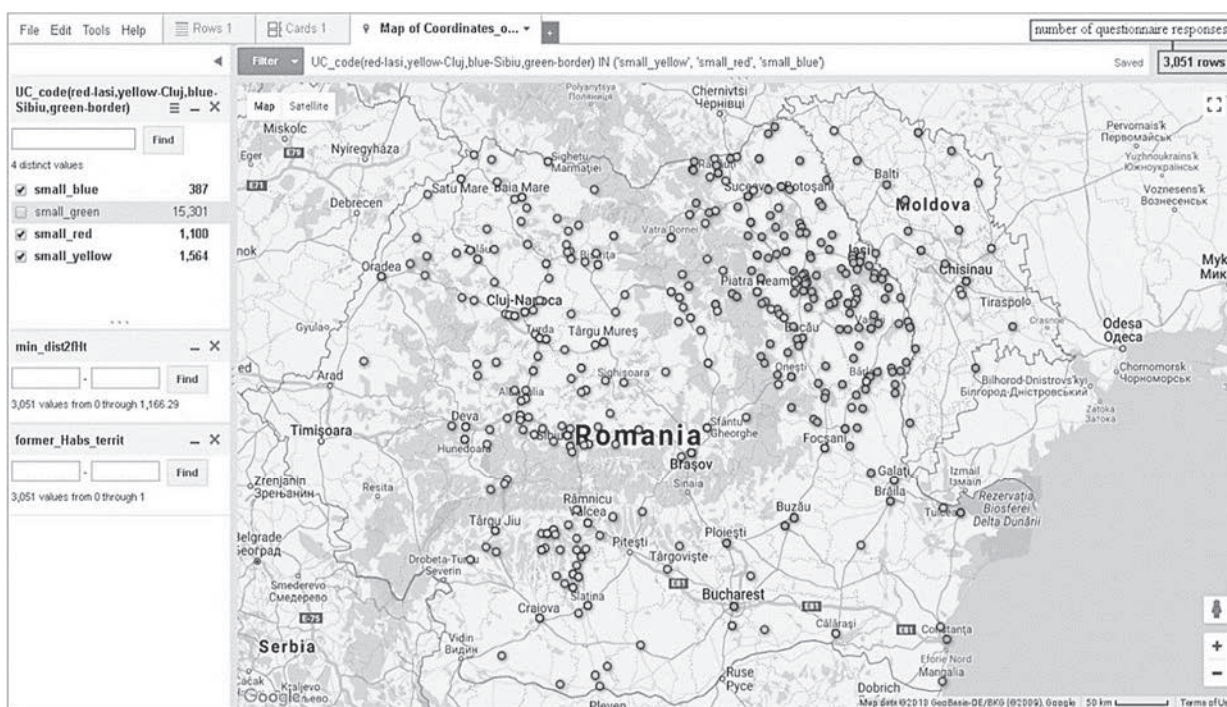
For computing the belonging of respondents' residencies to this former area (inside the corresponding polygon – Figs. 3a and 5c), 15,301 pairs of coordinates have been involved, including the regenerated historical border (3192 pairs) and the ones for the present NNWSW Romanian border. The calculations in VBA took 22 seconds in the first configuration and 3.5 minutes in the second one. The Excel file containing only the sheet dedicated for computing the belonging to the polygon and the source code of DotInPoly has grown from 484 KB (one invocation of DotInPoly) to 524 KB (entire column).

In all tests with VBA under Excel, the latter seemed frozen, showing a progress of execution message in the status bar at the bottom of the window, namely “Calculating: (8/4 processors(s)): n %”. In Task Manager (Performance tab), the in-use memory did not exceed 4 GB (first configuration) and 1.9 GB (the second one).

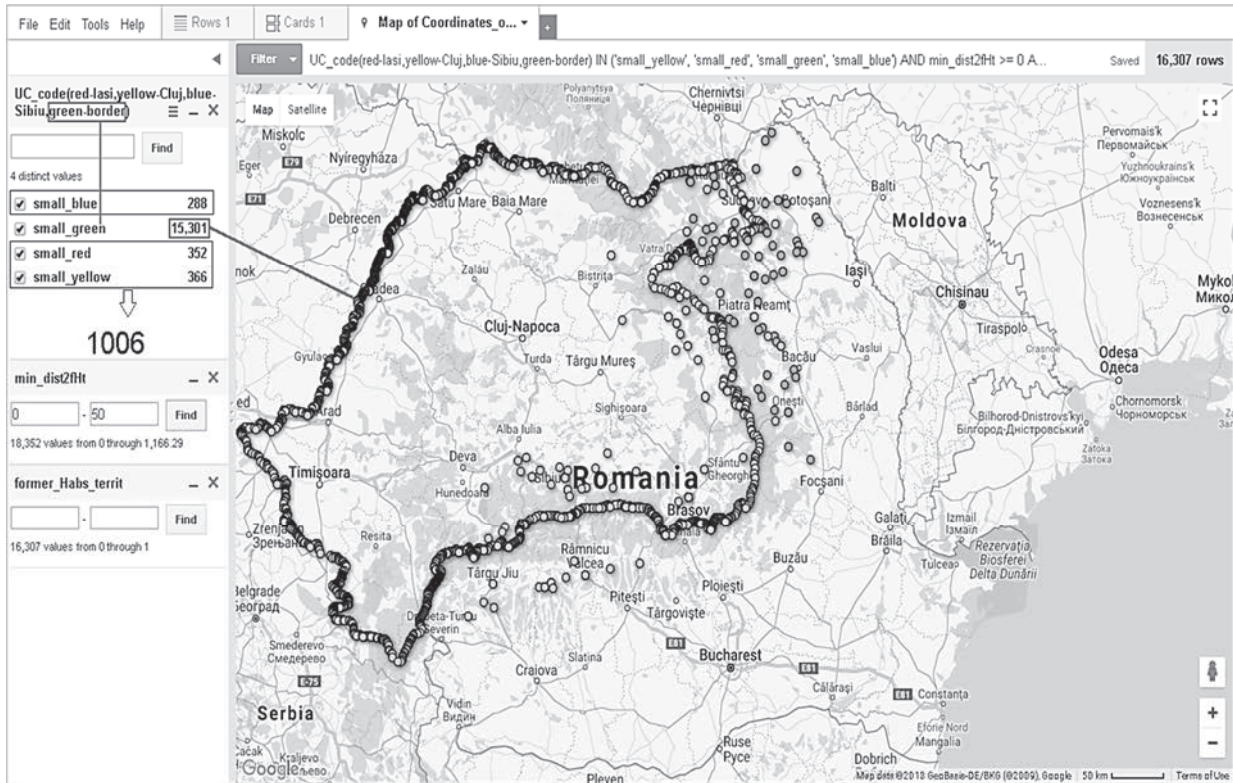
After storing the results as two separate columns (e.g. G for belonging and H for aerial distances if previously applying the ABS function to obtain an absolute value) in addition to the original dataset (3051 records), we can benefit from the use (e.g. in column I) of Excel's intuitive functions to support analyses focused on scales from aerial distance intervals: e.g. $=IF(G2=0,-1,1)*IF(H2=0,0,1)*IF(H2<10,1,IF(H2<20,2,IF(H2<30,3,IF(H2<40,4,IF(H2<50,5,IF(H2<60,6,IF(H2<70,7,IF(H2<80,8,9)))))))))$ in the cell referred as I2, for a symmetric resulting scale (from -9 to 9, with 0 for points on the border) and $=I2+10$ (in another cell – J2) for a scale comprising only positive values (from 1 to 19, and 10 for points on the border).

For representing the results (Fig. 5) obtained after applying the algorithms above (Fig. 2 and Fig. 3) in GFT, we started from a demo value (Fig. 5 – 50 Km) as an aerial distance threshold between respondent's location and the historical border. To be able to visualise in GFT both the coordinates of the questionnaire responses (3051 pairs – Fig. 5a) and the ones of the polygon (15301 pairs – Figs. 5b, 5c and 5d) in most scenarios, we have considered the following representation trade-off. In fact, we temporarily extended the domain of those two generated fields by allocating the intermediate value of 0.5 (former_Habs_territ – only 0 and 1 for further statistical analysis) and the one of 50 Km (min_dist2fHt) for coordinates belonging to the polygon (former border plus the rest of the present Romania's border: NNWSW). In the latter case, the indicated value in Km acted as left or right edge for our particular simulations and we reserved it with no doubt because it was not effectively obtained as it is (two-digit integer) when computing the minimum aerial distances.

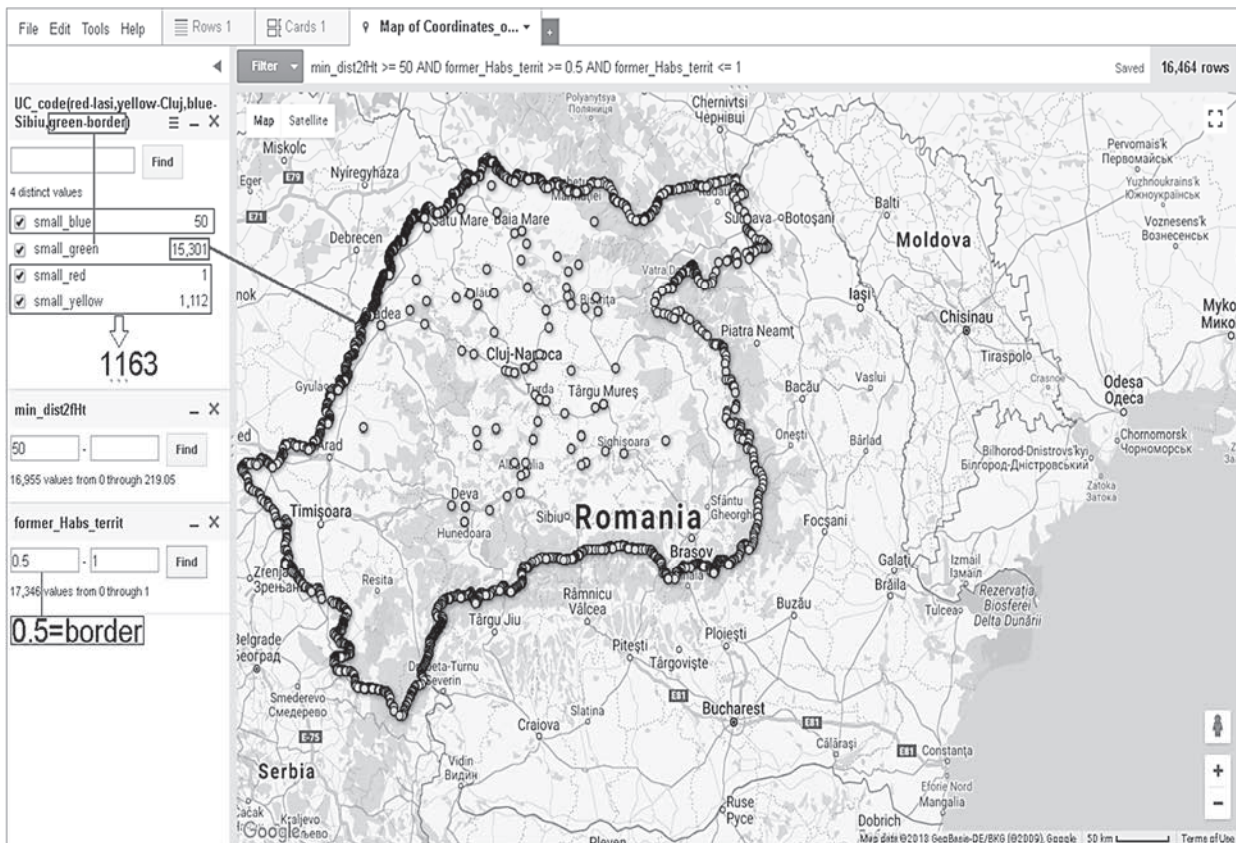
Although the visual feedback obtained when using GFT (Figs.5a-d) shows a good alignment of points for different representation criteria, the HAV-based P2P aerial distance computation approach still has the main limitation of being sensitive to the number and precision of the coordinates extracted with the GIS tool for the custom borders' polyline. Even with 3192 pairs extracted for this particular case we admit there is room for more points and better precision. This is the main reason we also included what we considered to be a better overall approach, namely P2S, less influenced by both categories of limitations above (quantitative and qualitative).



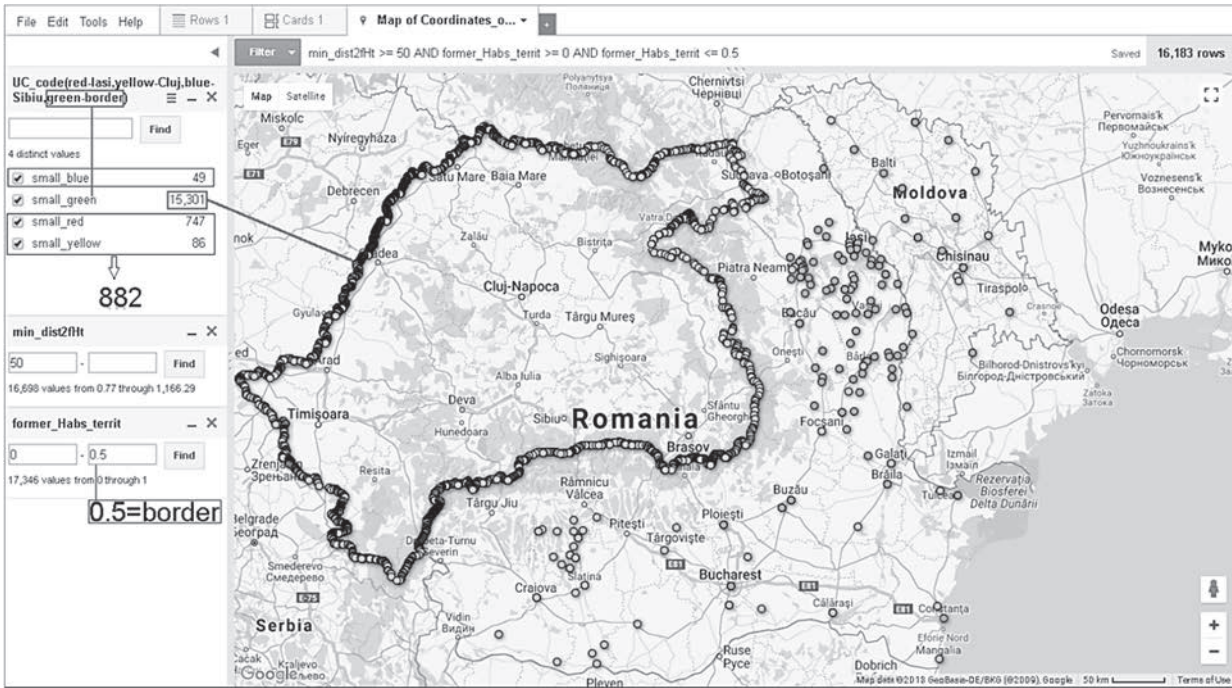
a)



b)



c)



d)

Fig. 5. Geographic distribution of responses (all 3051) using the GFT online interactive tool (Homocianu, 2018) (Homocianu, 2019): (a) all places; (b) dual band near the border with points both inside and outside the polygon and aerial distance to border, ad (in Km)<50; (c) inside and ad>=50; (d) outside and ad>=50

The execution time measured when using VBA under Excel (first configuration) for P2P and P2S (both by following the Pythagoras theorem with corrections depending on the average latitudes – Fig. 6b, lines 14-16) was comparable (almost 14 minutes in both cases) with the one when using HAV. In the case of P2S, we split the border’s polyline into segments connected by two points and performed

simple computations depending on the type of triangle – Eqs. (3,6) and Fig. 6a, generated in combination with each respondent’s coordinates as already proposed by Stojanovic and Stojanovic (2014), who have already done valuable research based on an implementation that benefits from high parallel processing power using the Compute Unified Device Architecture (CUDA) from Nvidia.

point_i a point_{i+1} ...
polyline segment
perpendicular falls inside the current segment
Eq. 3

two acute angles at the base of the segment => responses & border (each segment as 2 neighboring pairs in polyline)
min.dist. = the perpendicular
Eq. (4,6)
Point to Segments (P2S) approach – Stojanovic & Stojanovic
X ✓ f_x =GetDistKm 2nd(\$A4,\$B4,D\$1,D\$2,E\$1,E\$2)

Y^A Y=cosine(angle) Eq. 3
1
0
-1
π/2 π 3π/2 2π
angle

	A	B	C	D	E	F	G
1			3191 segments	47.98521	47.98062	47.97648	47.97315 ...
2		3051 pairs-responses	- polyline	26.12789	26.12978	26.13064	26.13029 ...
3			min.dist. D:DRV				
4	47.158455	27.601442	93.77	=GetDistKm 2nd(\$A4,\$B4,D\$1,D\$2,E\$1,E\$2)...			
5	45.696475	27.184043	62.17	266.18	265.72	265.37	264.79 ...
6	47.158455	27.601442	93.77	143.25	142.92	142.7	142.01 ...
7	47.241124	26.723383	38.77	93.44	93	92.69	92.05 ...
8	47.253314	27.239765	64.85	115.95	115.59	115.35	114.66 ...
...

point_i a point_{i+1} ...
polyline segment
perpendicular falls outside the current segment
Eq. 3

an obtuse angle at the base of the segment => min.dist. = min.length of the two sides (the perpendicular will be / was considered for the next / previous segment)
Eq. (5,6)

a)

```

Function GetDistKM2ND(la1D As Double, lo1D As Double, la2D As Double, lo2D As Double, la3D As Double, lo3D As Double)
Const Pi = 3.14159265358979
Const R = 6367
Dim d, dxy21, dxy31, dxy32, dx21, dx31, dx32, dy21, dy31, dy32, la1R, lo1R, la2R, lo2R, la3R, lo3R As Double
la1R = (la1D / 180) * Pi
lo1R = (lo1D / 180) * Pi
la2R = (la2D / 180) * Pi
lo2R = (lo2D / 180) * Pi
la3R = (la3D / 180) * Pi
lo3R = (lo3D / 180) * Pi
dx21 = (la2R - la1R) * R
dx31 = (la3R - la1R) * R
dx32 = (la3R - la2R) * R
dy21 = (lo2R - lo1R) * R * Cos((la2R + la1R) / 2)
dy31 = (lo3R - lo1R) * R * Cos((la3R + la1R) / 2)
dy32 = (lo3R - lo2R) * R * Cos((la3R + la2R) / 2)
dxy21 = Sqr(dx21 * dx21 + dy21 * dy21)
dxy31 = Sqr(dx31 * dx31 + dy31 * dy31)
dxy32 = Sqr(dx32 * dx32 + dy32 * dy32)
If (dxy21 * dxy21 < dxy32 * dxy32 + dxy31 * dxy31) And (dxy31 * dxy31 < dxy32 * dxy32 + dxy21 * dxy21) Then
GetDistKM2ND = Round(Abs(dx32 * dy21 - dx21 * dy32) / dxy32, 2)
Else
GetDistKM2ND = Round(Application.WorksheetFunction.Min(dxy31, dxy21), 2)
End If
End Function
    
```

b)

responses & border (each point in polyline)
Point to Points (P2P) approach - Pythagoras

	A	B	C	D	E	F	G
1			3192 pairs - polyline	47.98521	47.98062	47.97648	47.97315 ...
2		3051 pairs - responses		26.12789	26.12978	26.13064	26.13029 ...
3			min. dist. D:DRW				
4	47.158455	27.601442	93.77	=GetDistKm3P(\$A4,\$B4,D\$1,D\$2)			...
5	45.696475	27.184043	62.17	266.71	266.18	265.72	265.37 ...
6	47.158455	27.601442	93.77	143.69	143.25	142.92	142.7 ...
7	47.241124	26.723383	38.77	93.95	93.44	93	92.69 ...
8	47.253314	27.239765	64.85	116.41	115.95	115.59	115.35 ...
...

c)

```

Function GetDistKM3P(la1D As Double, lo1D As Double, la2D As Double, lo2D As Double)
Const Pi = 3.14159265358979
Const R = 6367
Dim d, dxy21, dx21, dy21, la1R, lo1R, la2R, lo2R As Double
la1R = (la1D / 180) * Pi
lo1R = (lo1D / 180) * Pi
la2R = (la2D / 180) * Pi
lo2R = (lo2D / 180) * Pi
dx21 = (la2R - la1R) * R
dy21 = (lo2R - lo1R) * R * Cos((la2R + la1R) / 2)
dxy21 = Sqr(dx21 * dx21 + dy21 * dy21)
GetDistKM3P = Round(dxy21, 2)
End Function
    
```

d)

Fig. 6. Alternative algorithm (Pythagoras theorem) with corrections depending on the latitudes (line10) to find minimum aerial distances: (a) use and (b) definition for P2S, when aiming for the perpendicular distance and (c) use and (d) definition for P2P

Most of the differences in our results, synthesized in the COMP_RES.xlsx file (Homocianu, 2020) as aerial distances in Km computed using all three approaches described in Figs. 2 and 6 were negligible, suggesting that any of the presented methods and even a mixture of them are suitable when dealing with custom borders defined with high precision and reasonable distances from given data

points (e.g. residences from questionnaire responses) to them. Consequently, the core hypothesis of this study is validated. The .xlsm (macro-enabled Excel) files hosting both the VBA source code and the geo-location data used in this methodological research paper are available inside two .zip archives at the same location above (Homocianu, 2020) to serve for further investigations and studies in the light of the attention

paid to the aforementioned scientific principle of transparency and support for replication.

4. Conclusions

We have used Microsoft Excel as a user-friendly data-oriented tabular environment, as well as simple definitions of custom functions under VBA, for developing an intuitive technique suitable for deriving basic spatial data required for historical and environmental border-related research.

Two categories of algorithms were successfully implemented. The first one was used for the biaxial computation of minimum aerial distances from each pair of coordinates from a set to a custom border, for which we also provided a way of regenerating the geographical coordinates. Alternative approaches have been also provided with negligible differences in terms of generated distances and processing time required. Therefore, the core hypothesis of this study is validated, both the P2P and P2S approaches being appropriate for computing distances to custom borders, with the condition of reasonable such distances (up to hundreds of Kilometres) and detailed sets of coordinates generated when reconstructing them. More, similar results have been obtained by taking into account the consecrated Haversine formula and the Pythagorean Theorem.

The second category of algorithms was used to identify the belonging of each pair of coordinates to a particular area (polygon), including the regenerated custom border (polyline). These algorithms have been also tested using different hardware configurations and IDEs.

Additionally, we have also provided aerial distance scale derivations for statistical analysis purposes, examples of representations using on-line interactive and query-able maps of coordinates for proving the appropriate spatial alignment of the inputs based on the entire methodology, together with computation benchmarks useful in terms of efficiency.

This study demonstrates the effectiveness of using spreadsheets and user-friendly programming languages in border related research and improves the access of researchers to GIS, HGIS and EGIS related computation.

Furthermore, we included and scientifically described technical details regarding three additional techniques dedicated to automatically extract the precise pairs of coordinates of the respondents' residencies, easily compute the road distances from such pairs to the ones of a certain point of interest and generating portable, auto-guided and interactive virtual road routes using different Google APIs.

In extenso, the entire methodology presented in this paper could be used in many other fields which deal with exploring and identifying consistent influences and patterns. Hence, we consider this methodological contribution useful for interdisciplinary studies about environmental management, engineering and applications of

advanced tools and techniques when analyzing complex phenomena.

References

- Afzal P., Alghalandis Y.F., Khakzad A., Moarefvand P., Omran N.R., (2011), Delineation of mineralization zones in porphyry Cu deposits by fractal concentration-volume modeling, *Journal of Geochemical Exploration*, **108**, 220-232.
- Arieli T., (2016), Borders, conflict and security, *International Journal of Conflict Management*, **27**, 487-504.
- Bailey T.J., Schick J.B., (2009), Historical GIS: enabling the collision of history and geography, *Social Science Computer Review*, **27**, 291-296.
- Becker S.O., Boeckh K., Hainz C., Woessmann L., (2015), The Empire is dead, long live the empire! Long-run persistence of trust and corruption in the bureaucracy, *The Economic Journal*, **126**, 40-74.
- Buchheim C., Chimani M., Ebner D., Gutwenger C., Jünger M., Klau G.W., Mutzel P., Weiskircher R., (2008), A branch-and-cut approach to the crossing number problem, *Discrete Optimization*, **5**, 373-388.
- Buta M., Paulette L., Man T., Bartha I., Negruşier C., Bordea C., (2019), Spatial assessment of soil salinity by Electromagnetic induction survey, *Environmental Engineering and Management Journal*, **18**, 2073-2081.
- Clifford N., Cope, M., Gillespie T.W., French S., (2016), *Key Methods in Geography*, Sage, Los Angeles, 456-469.
- Graybill A.R., (2014), *Boundless Nature: Borders and the Environment in North America and Beyond*, The Oxford Handbook of Environmental History, Oxford, USA.
- Gregory I.N., Healey R.G., (2007), Historical GIS: structuring, mapping and analysing geographies of the past, *Progress in Human Geography*, **31**, 638-653.
- Gregory I.N., Ell P.S., (2006), Error-sensitive historical GIS: Identifying areal interpolation errors in time-series data, *International Journal of Geographical Information Science*, **20**, 135-152.
- Gregory I.N., (2002), Time-variant GIS databases of changing historical administrative boundaries: a European comparison, *Transactions in GIS*, **6**, 161-178.
- Homocianu D., Airinei D., (2016), *GPS4GISVR - General Purpose System for Generating Interactive Street View Routes*, 15h Int. Conf. on Informatics in Economy, Cluj-Napoca, vol. 1, 1-6.
- Homocianu D., Airinei D., (2017), *Improving a cloud based generator of portable, auto-guided and interactive virtual routes*, 16h Int. Conf. on Informatics in Economy, Bucharest, vol. 1, 21-26.
- Homocianu D., (2018), Demo on using Google Fusion Tables for borders related studies (e.g. the former Habsburg Empire), On line at: <http://y2u.be/FU1VF28rjbE>.
- Homocianu D., (2019), GFT for custom borders including belonging & distances - e.g. the one with the former Habsburg Empire, On line at: <http://y2u.be/wRXFiLtsUPo>.
- Homocianu D., (2020), Materials uploaded to support a methodological research paper on the role of geo-location and boundaries in spatio-temporal and environmental studies, On line at: <https://drive.google.com/folderview?id=18DTEUJF6Z3amxA860M9-wKMTzFQAibe>.

- Houessou-Dossou E., Gathenya J., Njuguna M., Gariy Z., (2019), Flood frequency analysis using participatory GIS and rainfall data for two stations in Narok Town, Kenya, *Journal of Hydrology*, **6**, 90.
- Kimberling C., (2002), Enumerative triangle geometry part 1: The primary system, S, *Rocky Mountain Journal of Mathematics*, **32**, 201-225.
- Knowles A., (2014), The contested nature of historical GIS, *International Journal of Geographical Information Science*, **28**, 206-211.
- Kolossov V., (2005), Border studies: changing perspectives and theoretical approaches, *Geopolitics*, **10**, 606-632.
- Kolossov V., (2006), Theoretical limology: Postmodern analytical approaches, *Diogenes*, **53**, 11-22.
- Kumar G.N., Bangi M., (2018), An extension to winding number and Point-in-Polygon Algorithm, *IFAC-PapersOnLine*, **51**, 548-553.
- Levin N., Kark R., Galilee E., (2010), Maps and the settlement of southern Palestine, 1799-1948: An historical/GIS analysis, *Journal of Historical Geography*, **36**, 1-18.
- Lünen A., Travis C., (2012), *History and GIS: Epistemologies, considerations and reflections*, Springer Science and Business Media, Dordrecht, Netherlands.
- Manzano-Agugliaro F., San-Antonio-Gómez C., López S., Montoya F.G., Gil C., (2013), Pareto-based evolutionary algorithms for the calculation of transformation parameters and accuracy assessment of historical maps, *Computers and Geosciences*, **57**, 124-132.
- McLeman R., Herold S., Reljic Z., Sawada M., McKenney D., (2010), GIS-based modelling of drought and historical population change on the Canadian Prairies, *Journal of Historical Geography*, **36**, 43-56.
- Munafò M.R., Smith G.D., (2018), Robust research needs many lines of evidence, *Nature*, **553**, 399-401.
- Nishitateno S., Burke P.J., (2020), Have vehicle registration restrictions improved urban air quality in Japan? *Contemporary Economic Policy*, doi:10.1111/coep.12457.
- Nordin N.A.M., Zaharudin Z.A., Maasar M.A., Nordin N.A., (2012), *Finding shortest path of the ambulance routing: Interface of A* algorithm using C# programming*, in Humanities, Science and Engineering Research (SHUSER), IEEE Symp. on Humanities, Science and Engineering Research, 1569-1573.
- Paasi A., (2009), Bounded spaces in a 'borderless world': border studies, power and the anatomy of territory, *Journal of Power*, **2**, 213-234.
- Paasi A., (2005), Generations and the 'development' of border studies, *Geopolitics*, **10**, 663-671.
- Plopeanu A.P., Homocianu D., (2017), A Romanian university students' profile regarding their migration intentions. A cross-sectional study based on questionnaire survey, *Revista Romaneasca pentru Educatie Multidimensionala*, **9**, 61-88.
- Plopeanu A.P., Homocianu D., Mihăilă A.A., Crişan E.L., Bodea G., Bratu R.D., Airinei D., (2018), Exploring the influence of personal motivations, beliefs and attitudes on students' post-graduation migration intentions: evidence from three major Romanian Universities, *Applied Sciences*, **8**, doi.org/10.3390/app8112121.
- Sestraş P., Sălăgean T., Bilaşco Ş., Bondrea M.V., Naş S., Fountas S., Spalevic V., Cîmpeanu S.M., (2019a), Prospect of a GIS based digitization and 3D model for a better management and land use in a specific micro-area for crop trees, *Environmental Engineering and Management Journal*, **18**, 1269-1277.
- Sestraş P., Bilaşco Ş., Roşca S., Naş S., Bondrea M.V., Gâlgău R., Vereş I., Sălăgean T., Spalevic V., Cîmpeanu S.M., (2019b), Landslides susceptibility assessment based on GIS statistical bivariate analysis in the hills surrounding a metropolitan area, *Sustainability*, **11**, doi.org/10.3390/su11051362.
- Stanev K., (2013), A historical GIS approach to studying the evolution of the railway and urban networks: The Balkans, 1870–2001, *Historical Methods: A Journal of Quantitative and Interdisciplinary History*, **46**, 192-201.
- Stojanovic N., Stojanovic D., (2014), High performance processing and analysis of geospatial data using CUDA on GPU, *Advances in Electrical and Computer Engineering*, **14**, 109-114.
- van Brummelen G., (2012), *Heavenly mathematics: The forgotten art of spherical trigonometry*, Princeton University Press, New Jersey, USA.