

DEVELOPMENT AND TESTING OF GEO-PROCESSING MODELS FOR THE AUTOMATIC GENERATION OF REMEDIATION PLAN AND NAVIGATION DATA TO USE IN INDUSTRIAL DISASTER REMEDIATION

G. Lucas^{a, b, *}, Cs. Lénárt^c, J. Solymosi^d,

^a Research Institute of Remote Sensing and Rural Development, Károly Róbert College, Gyöngyös, Hungary –

gregory.luc4s@gmail.com

^b Doctoral School of Military Engineering, National University of Public Service, Budapest, Hungary

^c Research Institute of Remote Sensing and Rural Development, lenart.dr@gmail.com

^d National University of Public Service, Budapest, Hungary - jozsef.solymosi@somos.hu

Commission IV, WG IV/7

KEY WORDS: Remediation, Clean-up, Industrial Disaster, Automatic Planning, Geo-processing Models, GIS, Guidance, Navigation, Python, ArcPy.

ABSTRACT:

This paper introduces research done on the automatic preparation of remediation plans and navigation data for the precise guidance of heavy machinery in clean-up work after an industrial disaster. The input test data consists of a pollution extent shapefile derived from the processing of hyperspectral aerial survey data from the Kolontár red mud disaster. Three algorithms were developed and the respective scripts were written in Python. The first model aims at drawing a parcel clean-up plan. The model tests four different parcel orientations (0, 90, 45 and 135 degree) and keeps the plan where clean-up parcels are less numerous considering it is an optimal spatial configuration. The second model drifts the clean-up parcel of a work plan both vertically and horizontally following a grid pattern with sampling distance of a fifth of a parcel width and keep the most optimal drifted version; here also with the belief to reduce the final number of parcel features. The last model aims at drawing a navigation line in the middle of each clean-up parcel. The models work efficiently and achieve automatic optimized plan generation (parcels and navigation lines). Applying the first model we demonstrated that depending on the size and geometry of the features of the contaminated area layer, the number of clean-up parcels generated by the model varies in a range of 4% to 38% from plan to plan. Such a significant variation with the resulting feature numbers shows that the optimal orientation identification can result in saving work, time and money in remediation. The various tests demonstrated that the model gains efficiency when 1/ the individual features of contaminated area present a significant orientation with their geometry (features are long), 2/ the size of pollution extent features becomes closer to the size of the parcels (scale effect). The second model shows only 1% difference with the variation of feature number; so this last is less interesting for planning optimization applications. Last model rather simply fulfils the task it was designed for by drawing navigation lines.

1. INTRODUCTION

On October 4th, 2010 Hungary faced the worst environmental disaster in its history when the embankment of a toxic waste reservoir failed and released a mixture of 600,000 to 700,000 m³ of red mud and water. Lower parts of the settlements of Kolontár, Devescer, and Somlóvásárhely were flooded. Ten people died and another 120 people were injured. The red mud flooded 4 km² of the surrounding area.

The idea motivating this research work came after considering the clean-up work done on the impacted area of Kolontár (to the north of Balaton). Whereas digital maps figuring the contour of the contaminated areas and the pollution thickness were available¹ (Burai et al., 2011), the excavation work was performed in a traditional way, without the support of positioning and navigation technologies. So accurate and detailed information produced in the early stage of the remediation process was not efficiently exploited.

In a broader context, our research work aims at developing methodologies and tools to assure a continuum with the geographic information exploitation/support through a precise remediation process. The GI gathered during the disaster

assessment phase should be adapted and used in the planning phase; this would provide plans and navigation data for the clean-up phase. Additionally technologies integration (remote sensing (detection), GIS (planning), positioning and navigation (clean-up)) should also be researched.

Our bibliographic research demonstrated that the use of geoinformation technologies in remediation is mainly done during the early stage of remediation for the detection and mapping of the pollution. Aerial survey (Burai et al., 2011) and soil sampling are used for data acquisition. GIS, geo-statistical analysis and 3D modelling (Guyard, 2013; Mathieu et al., 2009; Hellawell et al., 2001; Franco et al., 2004; Webster et al.) are then employed for visualizing the pollution extents, estimating the volume to process (project dimensioning and costs) and planning /monitoring the remediation work (at site level). In contrast, the use of geographic information technologies during the clean-up stage seems quite limited. In the case of in-situ remediation, injection and recovery wells can be precisely positioned with GPS based on planning optimized with geo statistic calculations. In the case of ex-situ² remediation,

¹ from aerial survey and remote sensing processing methods

² ex-situ remediation is opposed to on-site remediation and requires the excavation of soil and its transportation out of the site.

* Corresponding author

literature does not mention the use of navigation and positioning technologies for the excavation work done by heavy machineries. As positioning technologies are routinely employed in civil engineering for the guidance of heavy equipment for precise and efficient work it seems the shortcomings in the case of ex-situ remediation lays in the capacity to generate adequate remediation plans and in the lack of adapted GIS tools, models, methods and practice [1]. In response, this work develops models in order to be able to produce a plan containing “clean-up parcels” and derived navigation data.

“Clean-up parcel” is a central concept and the geographic feature of interest in this work. Clean-up parcel in the real world corresponds with the surface covered by a dozer shovel until it gets filled to capacity (in other words the dozer’s maximum work footprint). In the GIS model a clean-up parcel consist of a rectangular feature in a polygon feature class. Its width is equal with the dozer’s blade width. Its length ($length_{Max}$) is derived from the bulldozer characteristics and the thickness of pollution to collect (1).

$$Volume_blade_{Max} = width_{dozer} \times length_{Max} \times thickness \quad (1)$$

The area of interest (contaminated area) is presented in figure 1. It is a polygon shapefile which was created from classified hyperspectral imagery (Burai et al., 2011). The area covers 4 km², is 16 km long in longitude and 5 km long in latitude. Because the catastrophe was a flood, the polygon features of the contaminated area have an orientation that generally follows the direction of the flood.

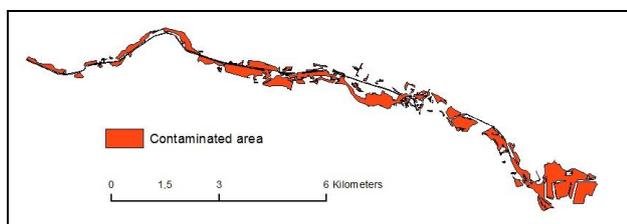


Figure 1. Overview of the source dataset “contaminated_area”

This paper focuses exclusively on precisely describing the conception of the algorithms, architecture and how geo-processing is done rather than providing line per line calculation details and scripts. The latter can be requested from the author using the email details provided.

Readers should notice that this exploratory work is relevant for ex-situ remediation (remediation where excavation is done) on extended areas where industrial disaster took place (red mud, nuclear, chemical, etc.). In such cases heavy machinery is used and it makes sense to try to plan their moves precisely in order to save effort, time and money, in a similar way as precision agriculture or civil engineering do.

The research firstly develops models through the design of algorithms and their transcription in Python scripts. Secondly the models are tested with a test dataset derived from the red mud disaster impact assessment. The first test control if geo-processing is done without errors. The second test control if the model shows efficiency in its tasks consisting of optimizing the clean-up parcel plan (i.e. reducing the number of parcels). Last, the efficiency of the model is assessed in regard to time efficiency. Based on the results of the tests diverse proposals are

formulated for the development of the final version of the models.

2. CLEAN-UP PARCELS MODEL DEVELOPMENT

2.1 Description of the objectives

This model generates a polygon feature class, containing rectangular features with a unique shape that represents the clean-up parcels. The parcel’s width is inherited from the bulldozer’s blade width. The parcel’s length is derived from the blade capacity. Dividing the contaminated area into clean-up parcels should be done automatically. The parcels should properly cover the whole contaminated area. The pattern designed should be optimal, meaning that technically it ensures the proper removal of pollution and economically it ensures the highest efficiency.

Considering those requirements it appears that rectangular grid pattern model is optimal.

2.2 Algorithm’s raw architecture

The feature class will be similar to a grid with rectangular polygons. The algorithm could be divided into two parts:

- the first part makes the calculations in order to point out to locations organised in a grid pattern with the appropriate orientation.
- the second part calculates the parcel corners’ coordinates and draws rectangular polygons.

Iterations (done with loops implementing repeat/while commands) will succeed ranges calculations deriving from the geographic extent of “Contaminated_area”. This calculation can be separated in a function.

As the process will be automatized, it could be useful to test different plans with different orientations of the parcels. An algorithm with four different orientations (0°, 90°, 45° and 135°) was drafted. The best result was selected can be done by counting the number of features in each feature class created and selecting the one with the fewest parcels.

2.3 Data requirement (input)

- A polygon feature class where features’ geometry represents the polluted areas.
- Width (in meter), length (in meter), orientation (in degree).

2.4 Algorithm architecture

Procedure createRectangleAtPoint(x, y, length, width, orientation, layer)

This procedure draws one rectangle according to the coordinates of a corner starting point, the orientation, the width and length of the rectangle. The vertices of the rectangle are attributed in the clockwise direction (figure 2).

Function extent(fc)

This function extracts the geographical extent (xmax, xmin, ymax, ymin) from a reference layer; i.e. the contaminated area layer. It is used later in the calculation of the maximal limit for the iteration in the loops building the grids. This function already existed and we have simply re-used it. [2]

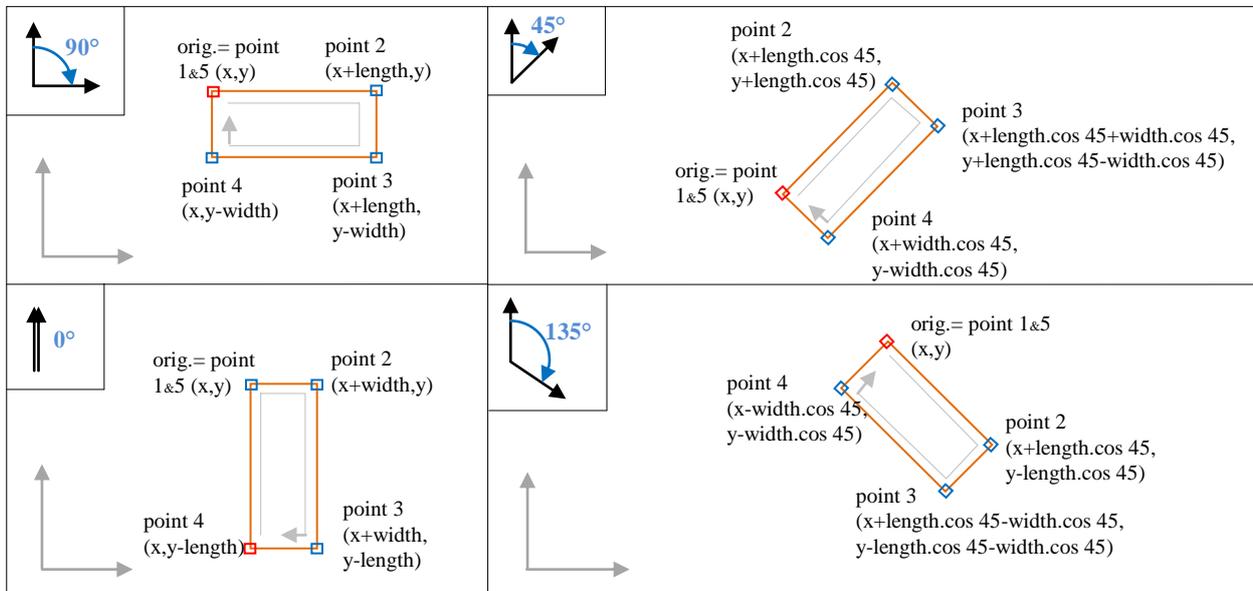


Figure 2. Details of the coordinate calculations with vertices of the parcel and drawing method in createRectangleAtPoint procedure with the 4 orientations cases

Procedure Make_Grid(length, width, layer_name, grid_orientation)

This procedure primarily draws a grid pattern taking into consideration the orientation, the width and length provided as parameters. For each point of the grid the procedure calls the createRectangleAtPoint procedure which draws a rectangle. With the 0° and 90° orientations the procedure loops top-down with the lines and left-right inside line. With the 135° and 45° patterns the procedure proceeds in two steps (step 1 is presented in blue colour, step 2 in green on Figure 3.). Step 1: loop creates features in diagonal starting from the top left corner moving towards the bottom right corner and a second loop control jumping one line down under the start of previous line using a backup of previous line start coordinates. Then in a second step, it moves diagonally going down (loop 2) but the second loop's implementation positions the next line on top of the previous one so that the grid can cover the second half of the area (above the step one). As many features are created out of the area of interest, a clean-up is necessary at the end. Selection is done on the features that intersect the "polluted_area" layer. They are copied in a new layer and all temporary layers are deleted at the end.

2.5. Script body

The algorithm uses the Procedure_Make_Grid and Procedure_CreateRectangleAtPoint in order to create four feature classes with 0°, 90°, 45° and 135° orientations. Finally a "get count" method is used to retrieve the number of features from each feature class. The feature class containing the smallest number of features is selected and saved; the other feature classes are deleted from the map document.

3. OFFSET EFFECT TESTING: MODEL DEVELOPMENT

3.1 Description of the objectives

The model should move the features of clean-up parcel altogether following a grid pattern (so both in vertical and

horizontal direction). The grid is oriented in the same way as the clean-up parcel feature class and the sampling distance of the grid is equal with a fifth of the parcel width. Each time the feature class is drifted the model counts how many features are located in the area of interest. The "get count" result with the smallest number of features shows the best offset to be applied.

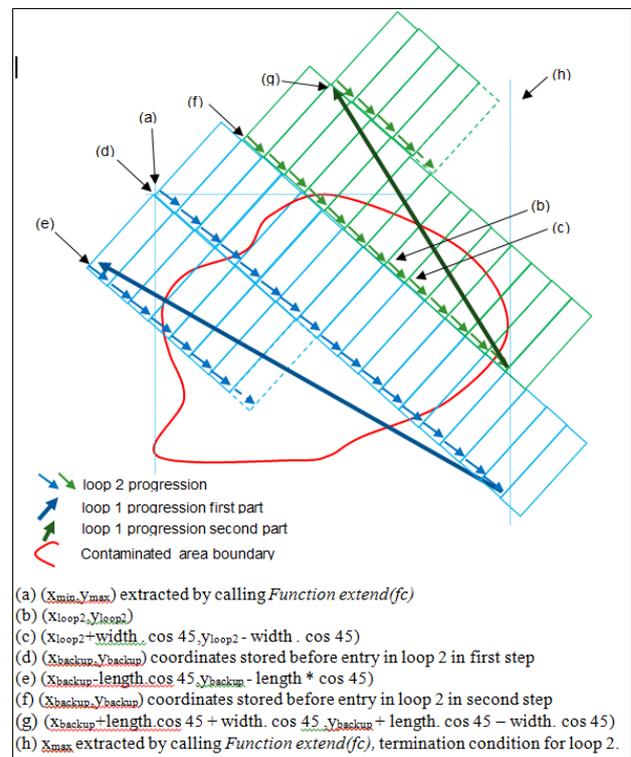


Figure 3. Conceptual representation of clean-up parcel model in the 45° case

3.2 Data requirement (input)

- the original area of interest is required to perform a selection based on intersection.
- a new clean-up parcel feature class is necessary. It is similar to the one generated in model 1 with optimal orientation but the reference area of interest differs (extended).
- The extended reference area of interest is the original area of interest extended with a buffer zone of the parcel width. If this precaution is not implemented, the clean-up parcel extent is too limited and for example an empty area appears on the left when x receives a positive drift.

3.3 Algorithm's raw architecture

1. The model should generate a new area of interest with a buffer of "length" size around the original area of interest.
2. Clean-up parcel feature class should be recreated based on the new target area (this is done in order not to have an empty area when the features will be shifted (maximal shift will be equal to parcel length)).
3. Calculate the shift values based on parcel width, length, orientation and store them in a three dimensional list.
3. All the features of this new clean-up feature class are shifted applying the offset values stored in the matrix (x,y). The grid x and y range are fixed at one-fifth of the parcel width.
4. Each time the feature class is shifted, a selection of the features intersecting with the original target area is done and the result of "getcount" is stored in a two dimensional list.
5. Unselect all features
6. Inverted shift is applied to set the feature back in place.
7. Next shift is applied, etc.
8. When all the shifting x,y values are passed, a search in the list value returns the smallest getcount.
9. From the minimal getcount, to retrieve the optimal x,y shift values.
10. Apply a final shift with the optimal x,y shift values.

3.4 Algorithm architecture

Function_calculate_drift_matrix(length, width, orientation)
This function returns a three dimensional matrix containing the shift coordinates corresponding to each point of the grid. The grid is oriented according to the parameter "orientation". The step of the grid is width/5 both with "rows" and "columns". For example if parcels are 30mx3m at 90°, there are 50 columns and 5 rows in the grid and the step is 3/5m. This function has four parts for the four different orientations.

Funtion_shift_features(in_features, x_shift=None, y_shift=None)
This function uses the arcpy.da module's UpdateCursor. By modifying the SHAPE@XY token, it modifies the centroid of the feature and shifts the rest of the feature to match. This function was available online and usable without changes, so it was simply copied [3].

Main procedure

The procedure deals with the creation of the extended area of interest; appeals the two functions described above and deals with the searches in the list "getcount".

4. NAVIGATION LINES MODEL DEVELOPMENT

4.1 Description of the objectives

The model should create a polyline feature class with navigation lines. The navigation lines should:

- be located in the middle of parcels,
- follow their length

Input: "clean-up parcel" shape file

Output: "Navigation_lines" shape file

4.2 Algorithm's structure

Function_ExtractVerticesCoordinateFromFeature(input_feature_class)

This function extracts the vertices' coordinates from a polygon feature class geometries and returns a two dimensional list storing the coordinates. SearchCursor method is employed on each row of the feature class. The result is appended to the list. Most of the script derives from the example of Reading polyline or polygon geometries of ESRI resources help [4].

Function_CalculateMiddlePoints(list_corners)

This function receives the coordinates of the four corners of a rectangle and returns the values of the coordinates of the two points located in the middle of the shortest sides.

Procedure_WriteaLine(point_1, point_2, layer)

This procedure writes a polyline feature between two given points (coming from function_CalculateMiddlePoints) in the given layer. SearchCursor method is applied to enter new geometry.

Procedure_DrawNavigationLines(Output_Navigation_Lines, Source_feature_class)

This procedure makes use of the functions and procedures above to draw a new polyline feature class with the navigation lines. Createfeatureclass_management method is used to create the output feature class.

The algorithms were successfully converted into scripts in Python language and models tested first with a subset of the pollution thickness layer derived from the processing of hyperspectral aerial survey data of Kolontar red mud disaster. (Burai et al., 2011)

5. RESULTS AND DISCUSSION

5.1 Clean-up parcel model

During its development the script was tested on a small feature extracted from the "Contaminated_area" shapefile.

Figure 4 shows an example of result with the four intermediary feature classes generated by the clean-up parcels model with 0°, 90°, 45° and 135° orientation, 3 meter width and 30 meter length on a sample of the contaminated area.

After correcting mistakes in the script the geo-processing model was applied to the whole "contaminated_area" shapefile. It resulted in very long geo-processing (more than 3 days to generate 0° and only a part of 45° orientation clean-up parcels layers).

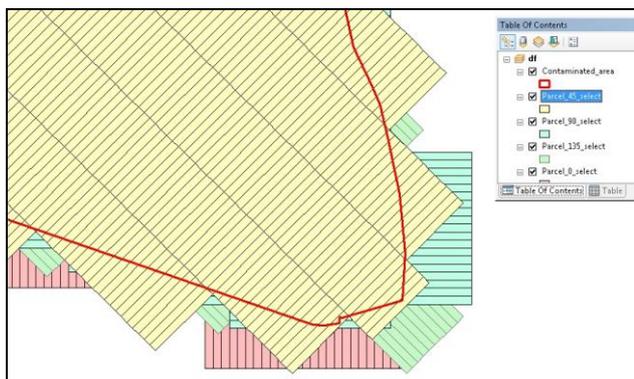


Figure 4. Intermediary results of clean-up parcel model with 0°, 90°, 45° and 135° orientation plan overlay

Processing was voluntarily stopped before geo-processing was completed. This long calculation was caused:

- 1/ by the extent and geometry of the target area (containing a lot of empty space where it was useless to have the geo-processing run),
- 2/ by the huge number of parcels to generate (around 60,000); a direct effect of the extent of contaminated_area,
- 3/ by the procedure_Make_grid which is not efficient with geo-processing (a lot of unnecessary geo-processing is done during iteration outside of the area of interest).

To cope with these various problems the second test was run on the same data but split into 8 zones (11 shapefiles as zone 7 was split in four).

The number of features generated per zone with the four orientations is summarized in the table 1. Smallest values are highlighted in green and highest in red background.

Orientation	0	45	135	90
Zone_1	13048	13690	13151	13062
Zone_2	13133	13869	12514	13112
Zone_3	25442	25522	24358	23416
Zone_4	1887	1938	1695	1614
Zone_5	5033	4431	5089	4486
Zone_6	2489	2795	2276	2370
Zone_7a	19	61	52	52
Zone_7b	147	165	205	203
Zone_7c	112	127	151	138
Zone_7d	22	65	61	64
Zone_8	297	457	359	387

Table 1. Number of features with the different orientations within the 8 zones.

At first we can observe a significant difference in the number of features obtained after geo-processing with different orientations. It appears the orientation of the parcel pattern is an important parameter to consider in optimizing planning.

Table 2 provides statistics per zone. First we calculated a classic measure of deviation of σ/\bar{x} (standard deviation divided by mean). As the number of entities vary significantly per sample (zone) and in order to have values of the same order it was necessary to divide deviation by mean. The deviation varies from 2% to 20%. The second value provided in the table is more relevant in our opinion because it better expresses the important difference between the extremes and better pulls out the efficiency of the algorithm (subtraction of the maximum feature number with the minimum feature number divided by the maximum feature number and expressed in percent). This

value can be interpreted as the ability of the algorithm to “reduce” the number of parcels by x%. The feature number reduction ranges from 4% to 38%.

Zone	σ	\bar{x}	σ/\bar{x}	(Max-Min)/Max (in %)
1	305	13238	2%	5%
2	555	13157	4%	10%
3	998	24685	4%	8%
4	154	1784	9%	17%
5	349	4760	7%	13%
6	226	2483	9%	19%
7	81	411	20%	38%
8	66	375	18%	35%

Table 2. Statistics with the 8 zones.

As a second conclusion the variations in the results can be very high (up to 38 %). This is definitely significant information for the planning strategy. Last, such a difference should be investigated and explained.

Figure 5 shows the geometry and size of the 8 zones in order to be able to cross the statistical results from table 2 with spatial information. The following observation can be formulated: the smallest zones show bigger variances (reported to the mean) than the biggest zones.

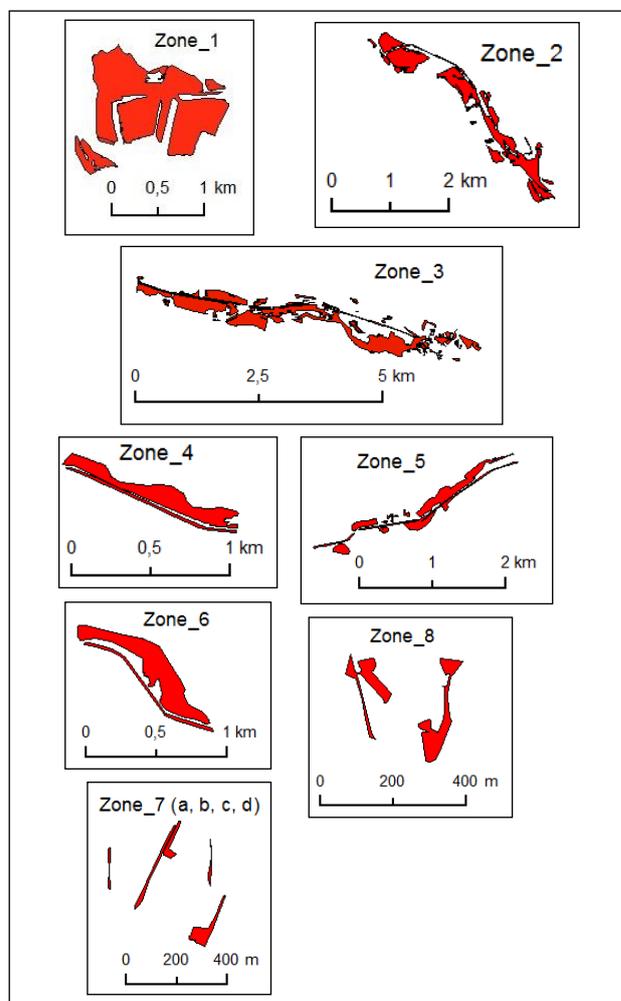


Figure 5. Overview of the 8 zones

Hypothesis 1: the reduced number of features is the cause of the bigger variance. Orientation matches more efficiently with smaller number of features because much of them are oriented in the same way. On the contrary when there are more features, their orientation varies more and the efficiency of the model decreases.

Hypothesis 2: the cause for important variance is a scale effect because the model efficiency works with a border effect. On smaller areas the features could be smaller; the ratio boundary/area is more in favour of the boundary compared to massive area and orientation becomes much more important.

After additional tests we could conclude that both hypotheses seem valid. When comparing the results between zone 4 that has two oriented features in the same direction and zone 7 a, b, c, d with small and long features; feature number decrease by 38 % with zone 7 whereas the feature number is only decreased by 17% for zone 4.

In terms of practice with the preparation of “Contaminated_area”; in order to optimize the geo-processing, the user should pay attention to three things:

- 1/ to prepare zones as small as possible in order to reduce empty areas (time consideration).
- 2/ to the extent possible have features with the same orientation inside one zone. If necessary, a zone should be split into several parts in order to ensure the features’ general orientation is as similar as possible (example is 7 a, b, c, d).
- 3/ to split feature if their geometry is complex. The result should be the creation of sub-features with simpler and oriented geometries.

In order to validate the presumptions mentioned above, the method was implemented on Zone 1 (figure 6) (where the algorithm showed the lowest efficiency) which was divided following the above recommendations. New results are summarized in table 3.

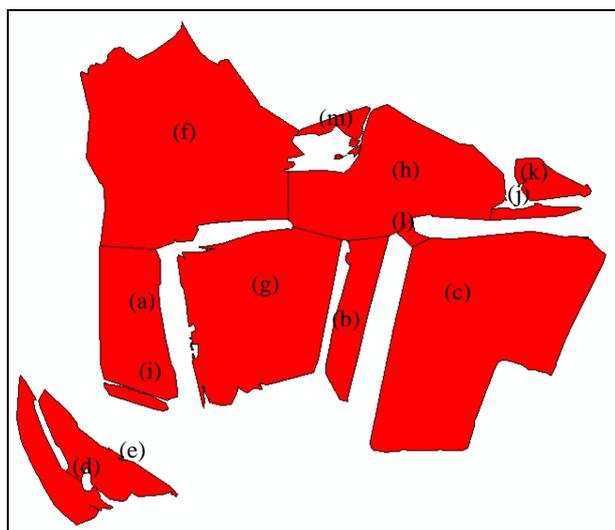


Figure 6: splitting of zone 1 into several parts

An additional reduction of 3.7% could be reached by applying an appropriate cut with zone 1 compared to the previous result.

Orientation	0	45	90	135
Zone_1a	887	1065	913	956
Zone_1b	507	605	588	593
Zone_1c	3076	3240	3089	3166
Zone_1d	459	527	483	453
Zone_1e	572	756	554	534
Zone_1f	3183	3449	3197	3216
Zone_1g	2037	2224	2052	2113
Zone_1h	1760	1836	1691	1770
Zone_1i	123	171	92	102
Zone_1j	166	170	102	161
Zone_1k	224	303	210	237
Zone_1l	71	84	61	73
Zone_1m	164	170	148	184
Sum min	12562		Sum max	14043

Table 3. Counting of the number of features with the different orientation and the different sub-zones.

Further developments

Regarding the reduction of geo-processing time, a test will be added inside the scripts implementing iteration. Before calling createRectangleAtPoint procedure an “IF” condition will be applied to check if the corner point (x,y) of the rectangle to be drawn falls into the area of interest (extended with a buffer zone of the parcel length). If x,y falls out no action will be taken, if it falls in then the rectangle will be written.

The orientation clearly appeared as a key parameter to control in order to optimize the remediation plan design. In our approach (which was exploratory) we decided to limit the number of orientation to 4 (with 0°, 45°, 90° and 135°). In order to increase the efficiency of the model the optimal orientation could be identified with 1° accuracy. This means the algorithm should be improved to take the following actions:

- 1/ isolate each individual polygon
- 2/ calculate polygon’s orientation (with 1° accuracy)
- 3/ apply a modified version of the clean-up parcel algorithm in order to design a clean-up plan with x° orientation for the feature considered.

With such an implementation, the optimized clean-up parcels are designed directly and it is no longer necessary to run the same script (clean-up parcel) four times with the four different orientations. So it would solve two issues: reducing the time processing and improving algorithm efficiency while reducing the number of parcel.

Dividing the contamination-area into subparts with homogenous orientation and limited geographic extents is the task of the user.

5.2 Shift testing results

Model 2 testing showed very limited results with the reduction of feature number. Only 1% percent difference with the number of features could be modelled. After further considerations, it seems that due to the irregular shape of the area of interest and the scale ratio, a shift is useless because on average as parcels disappear on one border others appear on the opposite border. If the AOI is regular (rectangle for example) and the scale ratio much smaller (AOI area compared to parcel area), this tool could achieve significant results. In our case -a large scale industrial disaster with relatively large irregular areas- the tool shows limited efficiency; consequently we decided not to go further with the development.

5.3 Draw navigation line model

Figure 7 and figure 8 below show the result of the Navigation lines model.

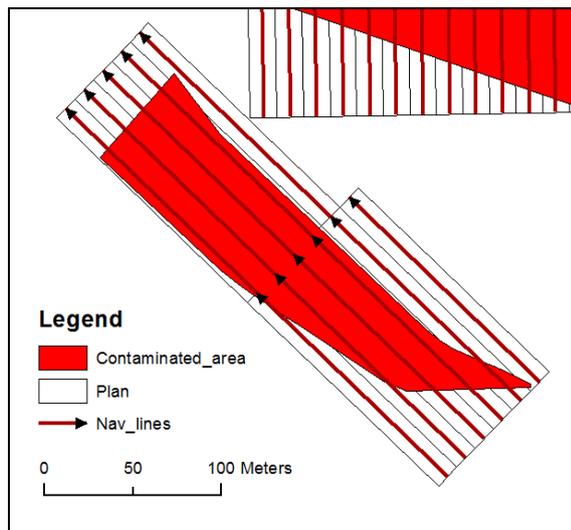


Fig.7. Navigation lines feature class generated by the navigation line model overlaying the clean-up parcels feature class

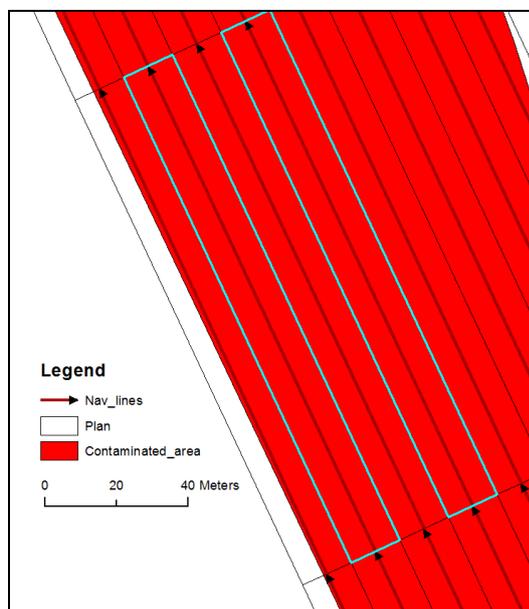


Fig.8. Zoom in the navigation lines

The algorithm orients all the geometries of the lines in the same direction by default. Some additional algorithms development could be foreseen if it turns out that navigation requirement would need a pre-planned navigation direction and computation of the optimal order of visit. As the operation in the field constantly changes, this kind of development is not considered a priority at the moment.

REFERENCES

Burai P., Smailbegovic A., Lénárt Cs., Berke J., Tomor T., Bíró T., 2011. Preliminary analysis of red mud spill based on aerial imagery. *Acta Geographica Debrecina, Landscape & Environment Series*, Volume 5, I: pp. 47-57.

Franco, C., Delgado, J., Soares, A., 2004. Impact Analysis and Sampling Design in The Pollution monitoring Process of The Aznalcollar Accident using Geostatistical Methods. In: *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Istanbul, Turkey, Vol. XXXV, Part B7, pp.373-378.

Guyard, C., 2013. Dépollution des sols et nappes : un marché sous pression. *L'eau, l'industrie, les nuisances*, n°359, pp.23-40. http://www.ianesco.fr/web/revue%20de%20presse/eau_industrie_nuisances_d%C3%A9pollution_sols_et_nappes_f%C3%A9v_2013_n%C2%B0359.pdf (accessed: 01 Jul. 2015)

Hellawell, E.E, Kemp, A.C, Nancarrow, D.J, 2001. A GIS raster technique to optimize contaminated soil removal. *Engineering Geology*, Vol. 60, Issues 1–4, pp. 107-116

Mathieu J.B., Garcia, V., Garcia, M., Rabaute, A., 2009. SoilRemediation: a plugin and workflow in Gocad for managing environmental data and modeling contaminated sites. Gocad Meeting, Nancy June 2-5, 2009.

Lindsay, J., Simon, T., Graettinger, G., 2001. Application of USEPA FIELDS GIS technology to support Remediation of Petroleum Contaminated Soils on the Pribilof Islands, Alaska. 2nd Biennial Coastal GeoTools Conference, Charleston, SC, January 8-11. <http://proceedings.esri.com/library/userconf/proc00/professional/papers/PAP789/p789.htm> (accessed: 01 Jul. 2015)

Webster, I., Ciccolini, L. Solving contaminated site problems cost effectively: plan, use geographical systems (GIS) and execute. http://www.projectnavigator.com/downloads/webster_ciccolini_solving_contaminated_sites_04.09.02.pdf (accessed: 01 Jul. 2015)

[1] Global remediation technologies. MAPPING & MODELING <http://grtusa.com/services/mapping-modeling/> (accessed: 01 Jul. 2015)

[2] ArcGIS Help 10.1 / Extent (Arcpy) <http://resources.arcgis.com/en/help/main/10.1/index.html#/018z00000072000000> (accessed: 01 Jul. 2015)

[3] ArcPy Café / Shifting features <https://arcpy.wordpress.com/2012/11/15/shifting-features/> (accessed: 01 Jul. 2015)

[4] ArcGIS Help 10.1 / Reading geometries <http://resources.arcgis.com/en/help/main/10.1/index.html#/002z0000001t000000> (accessed: 01 Jul. 2015)

ACKNOWLEDGEMENTS

This research was supported by the TÁMOP programme to the Károly Robert College (TÁMOP-4.2.2.D-15/1/ KONV-2015-0010).