



MOMIT
**MULTI-SCALE OBSERVATION
AND MONITORING OF RAILWAY
INFRASTRUCTURE THREATS**

Grant Agreement No: 777630

Project Acronym: MOMIT

Project Title:

**Multi-scale Observation and Monitoring of railway Infrastructure
Threats**

This project has received funding from the Shift2Rail Joint Undertaking under the European Union's Horizon 2020 research and innovation programme under grant agreement No 777630

**Technical note:
THEXfinder - A tool to automate the
detection of ADAs produced by expansivity
or thermal phenomena**

This Document is composed by 37 pages, including attachments

Last revision date: 2019-07-11

Dissemination Level: CO = Confidential

Deliverable information

Document Configuration

Programme Name:	Shift2Rail
Call:	H2020-S2RJU-2017
Topic:	S2R-OC-IP3-03-2017
Grant Agreement Number:	777630
Project Acronym:	MOMIT
Project Title:	Multi-scale Observation and Monitoring of railway Infrastructure Threats
Template Version:	v1.2
Deliverable Number:	Technical note
Title of the Deliverable:	THEXfinder - A tool to automate the detection of ADAs produced by expansivity or thermal phenomena
Work-Package:	WP3
Due Date:	-
Issue Date:	2019-07-11
Revision of the document:	v1.0
Dissemination Level:	CO = Confidential
Responsible Beneficiary:	CTTC
Prepared by:	José Navarro (CTTC)
Reviewed by:	
Contributors:	-

Revision History

Revision	Date	Author	Organisation	Description
v1.0	2019-07-11	José Navarro	CTTC	First issue

Acknowledgement

This project has received funding from the Shift2Rail Joint Undertaking under the European Union's Horizon 2020 research and innovation programme under grant agreement No 777630.

Statement of originality

This document contains original unpublished work except where clearly indicated otherwise. Acknowledgement of previously published material and of the work of others has been made through appropriate citation, quotation or both.

Disclaimer

This publication reflects the author's views. The Joint Undertaking is not responsible for any use that may be made of the information it contains.

The information contained in this document and any other information linked therein is confidential, privileged and it remains the property of its respective owner(s). As such, and under the conditions settled in the MOMIT Grant Agreement and the MOMIT Consortium Agreement, it is disclosed for the information of the intended recipients within the MOMIT Consortium and the European Commission / Shift2Rail Joint Undertaking according to its "Dissemination Level" and may not be used, published or redistributed without the prior written consent of its owner(s).*

** PU = Public; CO = Confidential, only for members of the consortium (including the JU and Commission Services) EU-R/R-UE = Classified, as referred to in Commission Decision 2001/844/EC.*

Table of Contents

LIST OF TABLES	5
LIST OF FIGURES	6
ACRONYMS AND ABBREVIATIONS	7
1 EXECUTIVE SUMMARY	9
2 DISCLAIMER	10
3 THE THEXFINDER TOOL	11
3.1 THE PROCESS	11
3.2 THE GUI.....	13
3.3 THE DEFAULTS FILE.....	17
3.3.1 <i>The available options</i>	18
3.4 THE MAP FILES	26
3.4.1 <i>The points read / write map file</i>	26
3.4.2 <i>The geologic map read map file</i>	29
3.5 THE INPUT (DATA) FILES	31
3.5.1 <i>The PS (points) file</i>	31
3.5.2 <i>The optional boundaries file</i>	31
3.5.3 <i>The optional infrastructures inventory file</i>	32
3.5.4 <i>The optional geologic inventory file</i>	32
3.6 THE OUTPUT FILES.....	32
3.6.1 <i>The ADA file</i>	32
3.6.2 <i>The points file</i>	33
3.7 INSTALLING THE TOOLS.....	34
3.8 USING THE TOOL	34
4 CONCLUSIONS	35
REFERENCE DOCUMENTS	36

List of Tables

Table 1: The available options in an THEXfinder defaults file	23
Table 2: Arrangement of a QI table	24
Table 3: Example of QI classes assignment.....	25
Table 4: The attributes in the read map file	28
Table 5: The attributes in the geology map read map file	30
Table 6: The attributes in the output ADAs file.....	33
Table 7: The attributes in the output points shapefile.....	34

List of Figures

Figure 1: Flowchart depicting the algorithm used by THEXfinder..... 13

Figure 2: THEXfinderr's GUI (options tab)..... 15

Figure 3: THEXfinderr's GUI (files tab)..... 15

Figure 4: Drawing ADAs: areas of influence (a) versus convex hulls (b)..... 16

Figure 5: QGIS showing a map with ADAs (blue areas) and the points making them (in red). 35

Acronyms and Abbreviations

ADA	Active Deformation Area
AOI	Area of Interest
CRF	Coordinate Reference Frame
GUI	Graphical User Interface
PS	Persistent Scatterers

Technical note:
THEXfinder - A tool to automate the
detection of ADAS produced by expansivity
or thermal phenomena

1 Executive Summary

This technical note presents THEXfinder, a tool to identify automatically Active Deformation Areas (ADA) that are caused by either expansive soils (expansivity) or thermal phenomena. The basics of the algorithm used are briefly described, as well as the kind and characteristics of the input, output and configuration files. The Graphical User Interface (GUI) and command line tools, although very concisely, are also explained.

2 Disclaimer

This document is the user's guide of the THEXfinder tool. It is neither a description nor a tutorial of the methodology it implements, which is described partially in [RD1]. Other parts of the methodology, devised by the University of Alicante, are not documented yet. Therefore, previous, solid knowledge of such methodology is required to fully exploit the potential of THEXfinder. A thorough reading of [RD1] is consequently recommended.

3 The THEXfinder tool

The THEXfinder tool is targeted at the automatic detection of Active Deformation Areas (ADA) that have been caused exclusively by expansive soils (expansivity) or thermal phenomena.

The application has been developed in C++ for efficiency reasons. Both a command line and a Graphical User Interface (GUI) are offered to the user. The GUI simplifies noticeably the operational procedure; the command line version may be used to integrate this tool in a batch production line, if necessary.

The tool accepts and produces Esri shapefiles as input and output for compatibility reasons. These files are easily managed by a wide number of Geographic Information System (GIS) tools. Among these, ArcGIS and QGIS. For a description of the shapefile format, please see [RD2].

Some options controlling the behaviour of the application are not shown in the GUI for simplicity reasons; normally, these extra options may be considered as project-wide ones so it would make no sense changing each value each time the tool is executed (in the context of some given project). Not appearing in the GUI does not mean that these options cannot be modified; these are stored in a defaults file that the tool opens each time it is started. In fact, the default values shown in the GUI when the tool is started (see Figure 2 and Figure 3 on page 15) are read from the aforementioned defaults file. This defaults file is, at the same time, the file where all the options are specified for the batch version of the tool.

THEXfinder is able to adapt itself to (slightly) different input shapefile formats. There is a minimal set of data that an input PS shapefile must contain to be processed by this application. Providing that such data is present in the shapefile, their positions in the file (“column”) may differ depending on the origin of the data. A configuration file (“read map”) may be used therefore to define where to find the necessary data without having to modify the application.

On output, THEXfinder creates two files: the first one contains the polygons defining the areas covered by the ADAs identified by the application. The second one stores a subset of the input points, those that meet some criteria.

The following sections describe THEXfinder in more detail. This encompasses the GUI, the formats of the configuration, input and output files as well as guidelines about how to use it.

3.1 The process

As already stated, THEXfinder implements the methodology described below.

The first set of steps is targeted at obtaining the PSs to process.

- Read the input files. These are (1) an Esri shapefile containing the Point Scatterers (PS) to analyse, *optionally* (2) another Esri shapefile including one or more polygons defining the Areas of Interest (AOI) where the actual process must take place and the inventory mapping the features of interest for each process (a geology map for the expansivity phenomenon, an inventory of infrastructures for thermal phenomena).

- If a shapefile containing the boundaries of the AOI has been provided, then clip the point dataset to remove those PS falling outside the requested limits.

The second part takes care of selecting only those points that meet the requirements set by the methodology on which THEXfinder relies. This means filtering the PSs according to the following criteria:

- Remove those PS not included in the appropriate areas of the inventory (expansive lithologies for the expansivity, buildings or man-built infrastructures in the case of thermal effects).
- Check, for each PS, if their deformation time series match a sinusoidal function. Remove these that do not match.

At this point, the first filtering phase is over. The next steps are based on the algorithm described in [RD1]. The goal is to group the surviving points into ADAs. This includes extra filtering as well, which takes place based on proximity and isolation criteria only.

- Remove isolated PS. A PS is an isolated one when no other PS exist within a distance defined by the user.
- Look for clusters of points that are close enough (the distance is set by the user). Those clusters with a number of PS below a given threshold (set by the user) are discarded. This is the second isolation filter.
- Look for ADAs. An ADA is a set of PS that are close enough to each other. Again, the distance defining the meaning of “close enough”, that is, the radius of influence, aka the clustering radius, is set by the user.
- Compute ADA statistics.
- Compute polygons representing ADA’s areas (concave hulls / influence areas).
- Write output ADA and PS shapefiles.

For more information about the details concerning the last part of the algorithm above, please refer to [RD1]. A flowchart depicting the whole algorithm may be found in Figure 1.

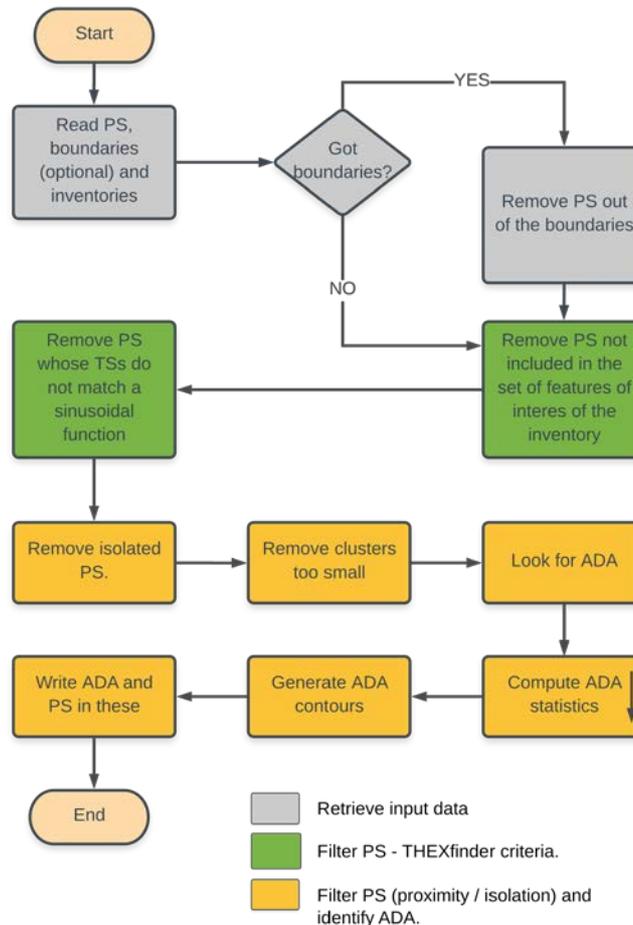


Figure 1: Flowchart depicting the algorithm used by THEXfinder.

3.2 The GUI

Figure 2 and Figure 3 depict the graphical interface of the THEXfinder (GUI version) tool. Figure 2 focuses on the options controlling the behaviour of the application. These options are:

- Couples (OP01, OP02) and (OP03 and OP04). These options let the user provide (1) the r^2 coefficient used as a threshold to accept or reject the match between the values in each PS' time series and a sinusoidal function and (2) the period (in days) of the sinusoidal function to match. The first couple is used for the expansivity process, while the second controls the thermal one.
- OP05. The distance (in meters) to consider than a PS is isolated, and thus, to be removed from the input data set.
- OP06. Minimum number of PS in a cluster. Clusters with this number of PSs or less will be removed in the filtering phase.
- OP07 and OP08. The radius of influence (OP08, meters) to consider the points surviving the previous filtering steps as belonging to the same cluster. That is, the points that, directly or indirectly are within the "ADA clustering radius" are going to be considered as members of

an ADA, but only (OP07) if the number of PS in such ADA is greater than or equal to the “Minimum ADA size (points)”.

- OP09. This option tells the application how many decimal positions the decimal comma must be shifted to convert the units of the coordinates of the points to, at least, centimetres. For instance, if these units are meters, then the value of this option should be 2, since shifting the decimal comma two positions to the right, the units become centimetres. This rather weird option is needed to compute the contours of the ADAS – the library used to do it works with integer instead of double values, so to avoid losing precision, all coordinates are converted internally by THEXfinder to centimetres and the rounded.
- OP10. When the ADAs are drawn as circular influence areas (see OP11) circles must be drawn. These are, in fact, approximated by a series of straight segments. This parameter controls the number of points over the theoretical circle that will be used to draw such segments. The minimum value is 3. Experience tells that 40 is a good number, not affecting the overall performance of the application.
- OP11. Selection of the way ADAS are drawn. Two approaches have been implemented: the first option is to draw “influence areas”. Each point in the ADA defines an influence area, limited by the circle centred at the point and with a radius measuring the amount set in the field “radius of influence” (OP08) above. The superposition of the influence areas of all the points in the ADA define its influence area. The second approach is to draw the convex hull containing the whole set of points in the ADA. Figure 4 show the differences of these two approaches when drawing the same set of ADAs. Note that purists consider the areas of influence a much more realistic representation of the area covered by ADAs – that is, affected by the deformation process.
- OP12. Velocity threshold to consider a PS as belonging to class 1 or 2.
- OP13 Number of elements in the deformation time series of each PS to take into account to compute the mean of the deformation. Given a number n , then the last n elements in each time series are used to compute the statistic.

For more details on the meaning of the several fields described above (especially for fields OP05 to OP13), please refer to [RD1].

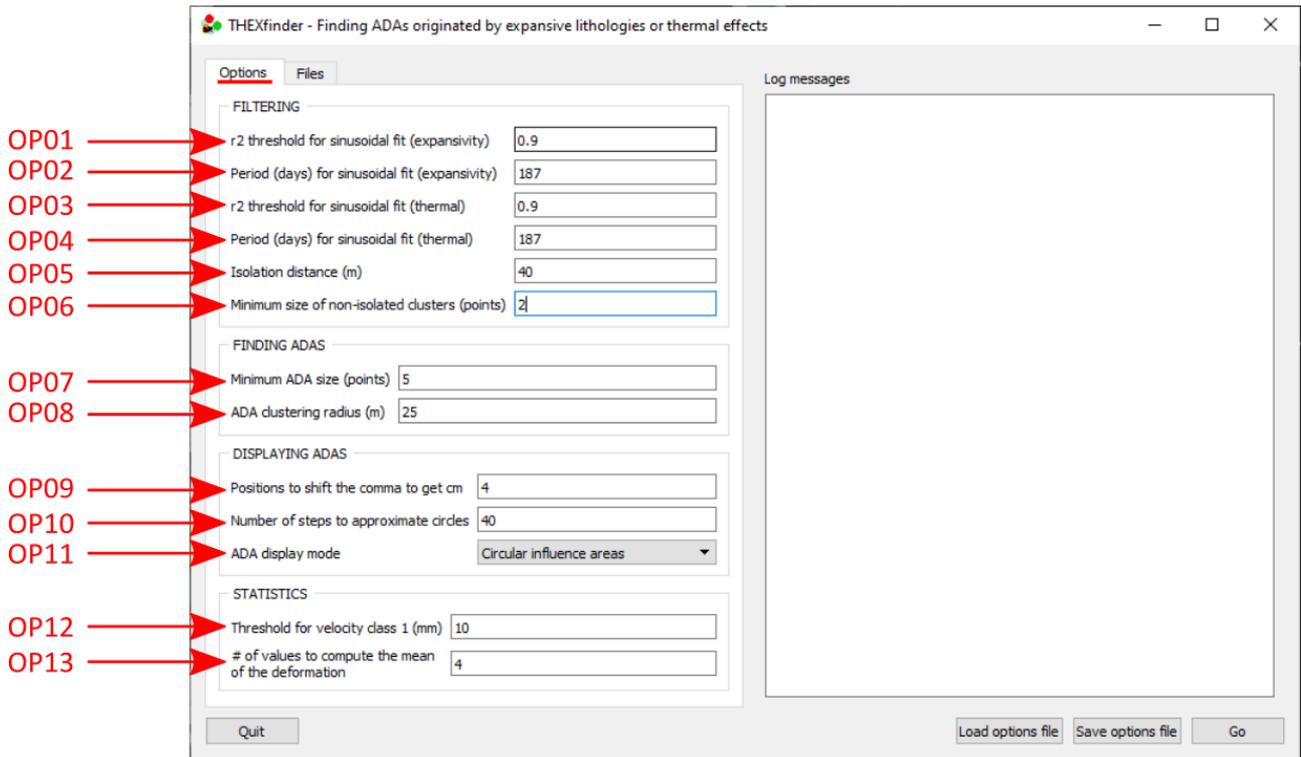


Figure 2: THEXfinderr's GUI (options tab).

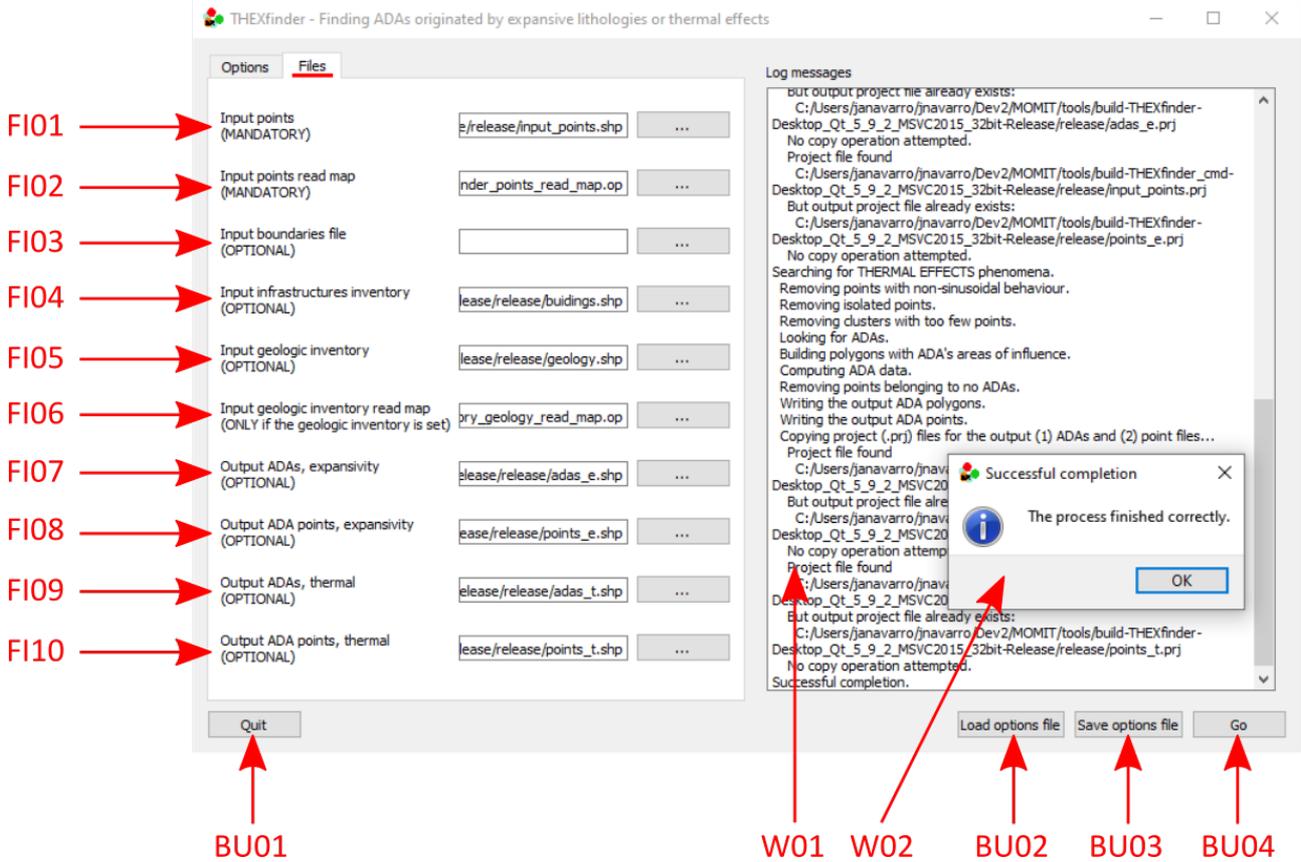


Figure 3: THEXfinderr's GUI (files tab).

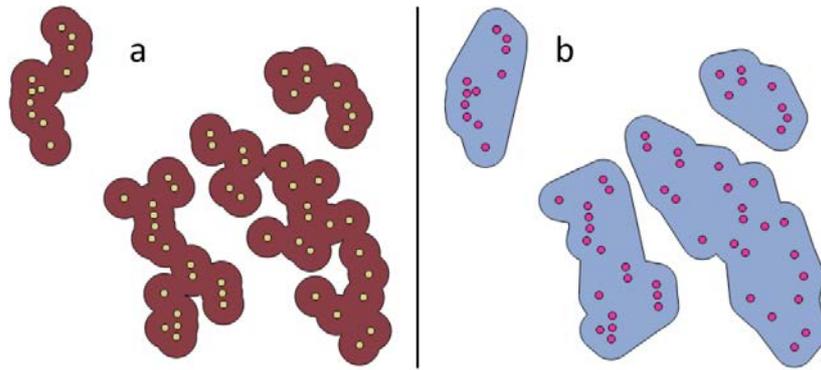


Figure 4: Drawing ADAs: areas of influence (a) versus convex hulls (b)

Concerning the Files tab (Figure 3), these are the meanings of the several fields there included:

- FI01. Name of the ESRI shapefile with the input PSs. Input. Must contain points. Mandatory.
- FI02. Name of the file containing the read map for FI01. Input. Mandatory. See section 3.4.1 for further details.
- FI03. Name of the ESRI file containing the boundaries delimiting the areas of interest (masks). Input. Must contain polygons. Optional.
- FI04. Name of the ESRI shapefile with the infrastructures inventory. Input. Must contain polygons. Optional.
- FI05. Name of the ESRI shapefile with the geologic inventory (aka geologic map). Input. Must contain polygons. Optional.
- FI06. Name of the geologic inventory (geologic map) read map (read map for FI05). May be omitted if FI05 is not set. Conversely, it is mandatory when FI05 is set. See section 3.4.2 for details.
- FI07. Name of the ESRI shapefile containing the ADAs detected (expansivity only). Output. It will contain polygons. Optional.
- FI08. Name of the ESRI shapefile containing the points included in the ADAs included in FI07. Output. It will contain points. May be omitted if FI07 is not set. Conversely, it is mandatory if FI07 is set.
- FI09. Name of the ESRI shapefile containing the ADAs detected (thermal effects only). Output. It will contain polygons. Optional.
- FI10. Name of the ESRI shapefile containing the points included in the ADAs included in FI09. Output. It will contain points. May be omitted if FI09 is not set. Conversely, it is mandatory if FI09 is set.

Note that there are many optional files. Concerning the inventories (FI04 and FI05), it is true that these are optional, but at least one of the must be specified or no detection process will be possible. Of course, setting both of them will enable the whole processing (expansivity, thermal effects) of data. In the same way, the output files (the couples FI07+FI08, FI09+FI10) will become mandatory depending on the presence / absence of the aforementioned inventories. If the infrastructures inventory exist (FI04) then the names of the output files for the thermal effects detection process

(FI09+FI10) must be provided. The same happens when the geological map (FI05) exist: the names of the output files for the expansivity detection process (FI07+FI08) must be set.

Concerning the read map files, that of the input points (FI02) is always mandatory (since the input points, FI01, are always required). On the contrary, the geologic inventory (geologic map) read map file (FI06) need to be specified only when a geologic map (FI05) exist.

Finally, Figure 3 show some other elements of the inportance, namely buttons and windows or window areas. These are:

- BU01. Quit the application.
- BU02. Load a THEXfinder options file previously saved. This may cause errors if such file does not adhere to the syntax defined for this kind of files (see section 3.3 for details). All options are loaded, including the names of the files involved in the process.
- BU03. Save the current set of options (including the fields in both tabs, that is, options and files) to disk. This may cause errors if invalid values have been typed or required information is missing. When successful, a THEXfinder options file that must be loaded later on by means of BU02 is created on disk.
- BU04. Start the actual processing of data. All the mandatory fields (options & file tabs) must have been provided.
- W01. This is the log area. The messages issued by the application will be shown here.
- W02. Popup windows. Used to report the status of the process.

3.3 The defaults file

Defaults (aka options) files are used to store the set of file names and options controlling the behaviour of the application.

When using **the GUI version** of the tool, a defaults file must reside in the same directory than the application's executable. A defaults file is just a plain text file with a set of defaults options that the GUI application will read on start-up and will show to the user. When processing numerous datasets for which the parameters to set are the same or almost the same, providing the appropriate defaults file may help to save time and reduce the number of human errors. Additionally, these defaults may suggest reasonable values for some of the parameters.

Remember: when using the GUI version of the tool, the defaults file must reside in the same directory than the tool's executable and its name must be exactly THEXfinder_default_configuration.op. If you are using an automated installer, a copy of this file will be deposited in such directory automatically.

The **command-line tool** accepts a full path and name to this defaults (options) file as an input parameter. In this case, the defaults file may reside in any folder, not necessarily in the application’s directory.

When working with the GUI version of the tool, **only a subset of the options included in this file may be modified by means of the graphical interface** of the application. Some others are considered either too complex to be offered to the regular user or too immutable as to be changed each time the application is run, so these are not shown in the GUI and may be changed in the defaults file only. By immutable parameters the reader must understand, essentially, “project wide parameters”, that is, those whose values will not change when the tool is used in some specific project. Not showing these in the GUI makes it clearer and less burdensome. The parameters not shown in THEXfinder GUI are those related to the definition of SNI and TNI thresholds and classes as well as those defining QI classes.

Note, however, that the values loaded from the defaults file may be overridden by loading (button BU02 in Figure 3) another options file. Such options file adhere to the same format that defaults files (in fact, these are exactly the same).

The **syntax** of the THEXfinder defaults (or user-saved options) file is **described in [RD3]**.

3.3.1 The available options

The options that may be present in an THEXfinder’s defaults file are summarized in Table 1. Note that THEXfinder expects to find **all** the options listed in the aforementioned table. Missing parameters or incorrect values of these will make the application complain about the error. Note also that in the description of each option, references to the GUI labels (such as OP03 or FI04) in Figure 2Figure 3 are made, so it is possible to link the names of such options in the options file and their counterparts in the GUI. Those parameters not shown in the GUI (that is, those that, therefore, must be modified manually editing the defaults / options text file) are labelled as “Not shown in the GUI”.

Label	Description	Comments
INPUT_SHAPEFILE_POINTS	(Full path and) filename of the input file containing the points to process. FI01.	Valid (full path and) file name. It must point to an ESRI points shapefile.
INPUT_SHAPEFILE_POINTS_READMAP	(Full path and) file name of the read map file defining how the input file with points must be read. See section 3.4.1. FI02.	Valid (full path and) file name. It must point to a points read map file (section 3.4.1)

Label	Description	Comments
INPUT_SHAPEFILE_BOUNDARIES	<p>This file is optional. It must contain either the (full path and) name of the input file defining the boundaries of the areas of interest or the character “-“ (quotes not included). In the second case, the dash indicates that there is no boundaries file to clip the input data.</p> <p>FI03.</p>	<p>Valid (full path and) file name. It must be an ESRI shapefile containing one or more polygons (the boundaries).</p>
INPUT_INVENTORY_INFRASTRUCTURES	<p>This is an optional file. It must contain either the (full path and) filename of the input file containing the inventory of infrastructures or the character “-“ (quotes not included). In the second case, the dash indicates that there is no boundaries file to clip the input data.</p> <p>FI04.</p>	<p>Valid (full path and) file name. It must be an ESRI shapefile containing polygons (the infrastructures being inventoried).</p>
INPUT_INVENTORY_GEOLOGY	<p>This is an optional file. It must contain either the (full path and) filename of the input file containing the geologic inventory (aka geologic map) or the character “-“ (quotes not included). In the second case, the dash indicates that there is no boundaries file to clip the input data.</p> <p>FI05.</p>	<p>Valid (full path and) file name. It must be an ESRI shapefile containing polygons (the kinds of soil being inventoried).</p>
INPUT_INVENTORY_GEOLOGY_READ_MAP	<p>This parameter needs to be set only when INPUT_INVENTORY_GEOLOGY is set too. Otherwise, it is optional (which means that its value should be set to the character “-“). When set, it contains the (full path and) name of the read map explaining how to read and interpret the geologic inventory.</p> <p>FI06.</p>	<p>Valid (full path and) file name. It must point to a geologic inventory read map (section 3.4.2)</p>
OUTPUT_ADA_POLYGONS_SOILS	<p>This parameter needs to be set when INPUT_INVENTORY_GEOLOGY is set too, otherwise it is optional (which means that its value must be the character “-“). When set, it contains the (full path and) file name of the output file containing the ADAs portraying expansive soils (aka expansivity).</p> <p>FI07.</p>	<p>Valid (full path and) file name. It must be an ESRI shapefile containing polygons (the ADAs produced by expansivity).</p>

Label	Description	Comments
OUTPUT_ADA_POINTS_SOILS	This parameter needs to be set when INPUT_INVENTORY_GEOLOGY is set too, otherwise it is optional (which means that its value must be the character "-"). When set, it contains the (full path and) file name of the output file containing the points included in the ADAs portraying expansive soils (aka expansivity). FI08.	Valid (full path and) file name. It must be an ESRI shapefile containing points (those included in the ADAs produced by expansivity).
OUTPUT_ADA_POLYGONS_TEMPERATURE	This parameter needs to be set when INPUT_INVENTORY_INFRASTRUCTURES is set too, otherwise it is optional (which means that its value must be the character "-"). When set, it contains the (full path and) file name of the output file containing the ADAs portraying thermal effects. FI09.	Valid (full path and) file name. It must be an ESRI shapefile containing polygons (the ADAs produced by thermal effects).
OUTPUT_ADA_POINTS_TEMPERATURE	This parameter needs to be set when INPUT_INVENTORY_INFRASTRUCTURES is set too, otherwise it is optional (which means that its value must be the character "-"). When set, it contains the (full path and) file name of the output file containing the points included in the ADAs portraying thermal effects. FI10.	Valid (full path and) file name. It must be an ESRI shapefile containing points (those included in the ADAs produced by thermal effects).
THRESHOLD_SINUSOIDAL_FIT_SOILS	Threshold to accept / reject the match between a sinusoidal function and the deformation time series of the points. Used by the expansive soils detection process. OP01.	Floating point. It must be in the range [-1,1].
SINUSOIDAL_FIT_PERIOD_SOILS	Period (in days) of the sinusoidal function used in the match with the deformation time series of the points. Used by the expansive soils detection process. OP02.	Floating point number. It must be greater than 0.
THRESHOLD_SINUSOIDAL_FIT_TEMPERATURE	Threshold to accept / reject the match between a sinusoidal function and the deformation time series of the points. Used by the thermal effects detection process. OP03.	Floating point number. It must be in the range [-1,1].

Label	Description	Comments
SINUSOIDAL_FIT_PERIOD_TEMPERATURE	Period (in days) of the sinusoidal function used in the match with the deformation time series of the points. Used by the thermal effects detection process. OP04.	Floating point number. It must be greater than 0.
ISOLATION_DISTANCE	Distance (in meters) to consider a point as isolated because of the lack of neighbors. Usually, it is two times the longitude of the longest pixel side. OP05.	Floating point number. It must be greater than 0.
ISOLATION_CLUSTER_SIZE	Minimum number of points in a cluster that makes it "not isolated". That is, even when a PS has a certain number of points around it (see ISOLATION_DISTANCE) , this point will be considered as isolated (and therefore, removed) if the total number of points is not greater than or equal than this value. OP06.	Integer value. It must be greater than 0.
ADA_MINIMUM_CLUSTER_SIZE	Minimum number of points in a cluster to make it an ADA. Clusters with less points will be discarded. OP07.	Integer value. It must be greater than 2.
ADA_RADIUS	Radius of influence (in meters) to build the ADA (usually, 1.3 times the longest pixel side). To be part of an ADA, a point must be close enough to at least another point. This distance sets the threshold for this proximity criterion. OP08.	Floating point number. It must be greater than 0.
ADA_SHIFT_TO_CM	Number of decimal places to shift the decimal point in such a way that the units of the projected coordinates of the input points become, at least, cm. This is necessary for the buffering operation applied to the ADA polygons; the library used to perform the buffering works with integer coordinates only, so scaling these to finer units is a must to avoid losing meaningful precision. OP09.	Integer value. It must be greater than or equal to 0.

Label	Description	Comments
ADA_CIRCLE_STEPS	<p>Number of steps to approximate the circle representing the area of influence of every point in an ADA cluster. The higher this number, the better circle approximation obtained, but also the more CPU time to compute it. Note that this parameter makes sense only when the ADA_DISPLAY_MODE is set to 1 ("influence area"). When ADA_DISPLAY_MODE is set to 2 ("concave hull") this parameter is ignored, although it must be present in this defaults file.</p> <p>OP10.</p>	<p>Integer value. It must be greater than 3.</p>
ADA_DISPLAY_MODE	<p>Display mode. ADAs may be represented as a series of circles whose centers are located at the points making the ADA (namely, the "influence area") or building the (buffered) concave hull surrounding these points. For influence areas, set the ADA_DISPLAY_MODE to 1. For concave hulls, set it to 2. Other values are illegal.</p> <p>OP11.</p>	<p>Integer value. It must be either 1 or 2.</p>
THRESHOLD_FOR_VELOCITY_CLASS_1	<p>Threshold (mm) to consider a velocity as belonging to class one. (if $v >$ this value) then velocity class = 1.</p> <p>OP12.</p>	<p>Floating point number. It must be greater than 0.</p>
N_VALUES_MEAN_DEFORMATION	<p>Number of values in each time series that will be used to compute the accumulated mean of the deformation. This number may never be greater than the actual number of values in the time series. If n is the value of this parameter, then only the LAST n values in each time series are used.</p> <p>OP13.</p>	<p>Integer value. It must be greater than 1 and less than or equal to the actual number of values in the time series in the input points file.</p>
TNITABLE_N_VALUES	<p>Number of threshold values in the TNI (Temporal Noise Index) table. See the explanations in the current section for more details about tables.</p> <p>Not shown in the GUI.</p>	<p>Integer value. It must be greater than 1.</p>

Label	Description	Comments
TNITABLE_THRESHOLDS	The TNI threshold values themselves. Note that there must be as many values as stated in parameter <code>TNITABLE_N_VALUES</code> above. Note that these thresholds are used to check autocorrelation values; therefore, these must be in the range (-1, 1) and sorted in ascending order. See the explanations in the current section for more details about tables. Not shown in the GUI.	List of floating point values. Each value must be comprised in the range (-1, 1). The list must be sorted in ascending order.
SNITABLE_N_VALUES	Number of threshold values in the SNI (Spatial Noise Index) table. See the explanations in the current section for more details about tables. Not shown in the GUI.	Integer value. It must be greater than 1.
SNITABLE_THRESHOLDS	The SNI threshold values themselves. Note that there must be as many values as stated in parameter <code>SNITABLE_N_VALUES</code> above. Note that these thresholds are used to check correlation values; therefore, these must be in the range (-1, 1) and sorted in ascending order. See the explanations in the current section for more details about tables. Not shown in the GUI.	List of floating point values. Each value must be comprised in the range (-1, 1). The list must be sorted in ascending order.
QITABLE_CLASSES	The Quality Index (QI) class table. This table must have a total of $(\text{TNITABLE_N_VALUES} + 1) * (\text{SNITABLE_N_VALUES} + 1)$ values. See the explanations in the current section for more details about tables. Not shown in the GUI.	List of integer values.

Table 1: The available options in an THEXfinder defaults file

For a full understanding of the parameters OP05 to OP13 as well as those labelled as “Not shown in the GUI” please refer to [RD1].

The parameters describing **tables** (temporal & spatial noise indices, quality index class) deserve a special discussion.

The temporal and spatial noise index tables work exactly in the same way. Therefore, only the first case (spatial noise index table) will be described. The rationale behind the explanation below may then be applied literally to the temporal noise index table.

The temporal noise index is based on the value of the autocorrelation of the deformation time series. The possible range for the autocorrelation value (-1, 1) is split in several segments, each of these defined by a threshold value. Thus, if for instance x thresholds are defined, this implies $x+1$ segments in the autocorrelation range.

Assuming that the threshold values are -0.4, 0.5 and 0.9 (three thresholds) then four segments exist: [-1, -0.4], (-0.4, 0.5], (0.5, 0.9], (0.9, 1]. Each of these segments correspond to a temporal noise class. The identifier of these classes starts at 1 for the first segment and ends at $x + 1$ (the number of segments), 4 in this example. Thus, the classes would be 1,2,3 and 4.

The parameters `TNITABLE_N_VALUES` and `TNITABLE_THRESHOLDS` are used to define the thresholds, and, indirectly, the segments and classes used to classify the spatial noise. In the example above, `TNITABLE_N_VALUES` should be set to 3 (there are 3 thresholds) and the list of values in `TNITABLE_THRESHOLDS` should be -0.4, 0.5 and 0.9. The correct way to write this in the defaults options file would be:

```
TNITABLE_N_VALUES      = 3
TNITABLE_THRESHOLDS  = -0.4 0.5 0.9
```

The same discussion applies to `SNITABLE_N_VALUES` and `SNITABLE_THRESHOLDS`.

Finally, the quality index class (QI) table explains how the combination of the classes derived from the spatial and temporal quality indices produce another class classification. For clarity reasons, an example will be used to describe how the QI table is built and stored in the defaults file.

The assumption is that there exist four classes for the temporal noise index (namely, 1, 2, 3 and 4) and only three for the spatial noise index (that is, 1, 2 and 3). Then, the QI table must be arranged as shown in Table 2. Note that in this table the spatial noise classes are arranged from left to right while the temporal ones are deployed from top to bottom.

In this example, up to $3 \times 4 = 12$ QI classes are possible. At any rate, the number of classes is decided by the user of the application; however, the maximum number of these is limited by the product of the number of spatial and temporal noise classes. Of course, it is also the user who assigns a QI class to each combination of spatial and temporal classes.

		Spatial Noise Class		
		1	2	3
Temporal Noise Class	1			
	2			
	3			
	4			

Table 2: Arrangement of a QI table

In this example, 6 QI classes will be assumed. Table 3 shows how the user decided to assign these classes depending on the temporal and spatial ones.

		Spatial Noise Class		
		1	2	3
Temporal Noise Class	1	1	2	3
	2	2	3	4
	3	3	4	5
	4	4	5	6

Table 3: Example of QI classes assignment

This table must be written in the defaults file. To do it, parameter `QITABLE_CLASSES` must be used. The class data in the matrix must be written row-wise, starting at the top of the table and then going downwards. The example below shows how Table 3 must be written to the defaults file:

```
QITABLE_CLASSES = 1 2 3 2 3 4 3 4 5 4 5 6
```

Note that the number of elements (12) in `QITABLE_CLASSES` is directly derived from the product `TNITABLE_N_VALUES` × `SNITABLE_N_VALUES`, so there is no need for a parameter in the defaults file specifying its size.

A full example of an THEXfinder defaults file is shown below. Note that no comment lines have been written in this example.

```
INPUT_SHAPEFILE_POINTS = input_points.shp
INPUT_SHAPEFILE_POINTS_READMAP = points_read_map.op
INPUT_SHAPEFILE_BOUNDARIES = -
INPUT_INVENTORY_INFRASTRUCTURES = buidings.shp
INPUT_INVENTORY_GEOLOGY = geology.shp
INPUT_INVENTORY_GEOLOGY_READ_MAP = geology_read_map.op
OUTPUT_ADA_POLYGONS_SOILS = ADA_polygons_expansive_soils.shp
OUTPUT_ADA_POINTS_SOILS = ADA_points_expansive_soils.shp
OUTPUT_ADA_POLYGONS_TEMPERATURE = ADA_polygons_thermal_effects.shp
OUTPUT_ADA_POINTS_TEMPERATURE = ADA_points_thermal_effects.shp
THRESHOLD_SINUSOIDAL_FIT_SOILS = 0.90
SINUSOIDAL_FIT_PERIOD_SOILS = 187.0
THRESHOLD_SINUSOIDAL_FIT_TEMPERATURE = 0.90
SINUSOIDAL_FIT_PERIOD_TEMPERATURE = 187.0
ISOLATION_DISTANCE = 40.0
ISOLATION_CLUSTER_SIZE = 2
ADA_MINIMUM_CLUSTER_SIZE = 5
ADA_RADIUS = 25.0
ADA_SHIFT_TO_CM = 4
ADA_CIRCLE_STEPS = 40
ADA_DISPLAY_MODE = 1
THRESHOLD_FOR_VELOCITY_CLASS_1 = 10.0
N_VALUES_MEAN_DEFORMATION = 4
TNITABLE_N_VALUES = 3
TNITABLE_THRESHOLDS = 0.53 0.70 0.84
```

```

SNITABLE_N_VALUES           = 3
SNITABLE_THRESHOLDS        = 0.53 0.70 0.84
QITABLE_CLASSES             = 1 1 2 4 1 2 3 4 2 3 3 4 4 4 4 4

```

3.4 The map files

Reading shapefiles imply reading attribute tables too. Some of these attributes, the ones required by the THEXfinder tool, may be located at different positions in shapefiles created by different tools. To deal with such variety of sources, THEXfinder includes a couple of so-called “map files” that define where to find the information sought.. These are the points read/write map file, and the geologic inventory read map file.

3.4.1 The points read / write map file

The points that THEXfinder will process must be stored in an ESRI shapefile (FI01 in sections 3.2 and 3.3). Although the shapefiles (.shp, .shx) containing points are always the same, this is not so for the attribute table (.dbf file) accompanying the former.

The set of attributes in the .dbf file may vary. THEXfinder needs a minimal set of attributes to be present in order to carry out its work. If at least one of these attributes does not exist, THEXfinder may not use the file.

The mandatory attributes are:

- X coordinate (easting).
- Y coordinate (northing).
- Velocity (in mm / year).
- Deformation time series (mm / year).

The X & Y coordinates must be projected coordinates. The number of values in the deformation may vary from one file to other, but must be the same for all the points in the same file.

Additionally, two more optional fields may be present for each point:

- Lambda coordinate (deg) and
- Fi coordinate (deg).

Assuming that a points shapefile’s attribute table contains at least the mandatory ones (and, eventually, the optional ones too) the positions where these appear may vary between different datasets (because, for instance, have been produced by several providers in different projects). Obviously, this variability poses a problem that must be solved if THEXfinder must be able to read these different shapefiles.

This is why the tool includes a mechanism to define where to find the mandatory (and, eventually optional) fields listed above. By providing such information, THEXfinder will be able to read many different point shapefiles with no changes in the software itself.

The way to do this is to assign, to each field, the position in the shapefile where the field sought may

be found. For instance, assuming a shapefile whose attribute table is defined as follows:

```
identifier X Y lambda fi gamma row column velocity def01 def02 ... def20
```

then the positions for the X, Y, velocity and deformation attributes would be, respectively, 1, 2, 8 and 9 (note that the first column, identifier in the example, uses index 0). Since the number of values in the deformation time series is also user-definable, it is necessary to state that such number; in this example, it is 20. The positions of the optional lambda and fi attributes would be 3 & 4.

To define such positions, a simple options file, whose syntax is defined in [RD3], is used (in fact, is exactly the same kind of file used by the defaults file).

Finally, when writing the output file(s) with points the names of the (variable) number of fields storing the values of the time series with deformations must be defined. These fields may have any name(s) the user wish to assign, although the usual practice is to label these including the date of the measurement.

For instance, a series consisting of four values measured the 30th of April of 2010, 2011, 2012 and 2013 could use the strings “20100430”, “20110430”, “20120430” and “20130430” to label the corresponding deformation fields in the output points file, where the pattern to build these identifier adheres to the schema “YYYYMMDD”, being YYYY the year, MM the month and DD the day.

In fact, the usual convention prepends a “D” in front of these dates to build the field names – where the literal letter “D” stands for “deformation”; thus, the previous example would mutate to “D20100430”, “D20110430”, “D20120430” and “D20130430” respectively.

Note, however, that this is an accepted convention and not a rule; the user of THEXfinder may decide to name these fields at his / her will.

What is true is that it is necessary to provide names for the deformation fields. The way to do it is the label named OUTPUT_TS_FIELD_NAMES.

All the labels to use (for both read and write maps) are defined in Table 4.

Label	Description	Comments
POSITION_X	Position (column number) of the X projected coordinate.	Mandatory, 0.based index.
POSITION_Y	Position (column number) of the Y projected coordinate.	Mandatory, 0.based index.
POSITION_VELOCITY	Position (column number) of the velocity.	Mandatory, 0.based index.
POSITION_TIME_SERIES	Position (column number) of the first value of the deformation time series.	Mandatory, 0.based index.
N_VALUES_TIME_SERIES	Number of values in the deformation time series.	Mandatory, It must be greater than 0.

Label	Description	Comments
HAVE_LAMBDA_FI	Flag indicating whether the optional lambda and fi coordinates are available.	Mandatory. Either 0 (no lambda and fi exist) or 1 (lambda and fi are available).
POSITION_LAMBDA	Position (column number) of the lambda (longitude) coordinate.	Optional if HAVE_LAMBDA_FI is zero. Mandatory otherwise. 0 based index.
POSITION_FI	Position (column number) of the fi (latitude) coordinate.	Optional if HAVE_LAMBDA_FI is zero. Mandatory otherwise. 0 based index.
OUTPUT_TS_FIELD_NAMES	Names of the fields (in the output points file) containing the time series with the deformations.	Mandatory. List of string values. As many values in the list as indicated by label N_VALUES_TIME_SERIES.

Table 4: The attributes in the read map file

The following is an actual example of a read map file, including a few comments.

```
# =====
#
# This is and THEXfinder's points read map file.
#
# =====

# -----
#
# MANDATORY attributes. Positions (0 is the origin) for:
# - X
# - Y
# - Velocity
# - First value of the deformation time series.
#
# Additionally, number of values in the deformation time series.
#
# -----

POSITION_X           = 5
POSITION_Y           = 6
POSITION_VELOCITY    = 9
POSITION_TIME_SERIES = 11
N_VALUES_TIME_SERIES = 26

# -----
#
```

```

# OPTIONAL attributes. Flag indicating the availability of the lambda and
# fi coordinates and the positions of the coordinates themselves.
#
# -----

HAVE_LAMBDA_FI      = 1
POSITION_LAMBDA     = 4
POSITION_FI         = 3

# -----
#
# The names for each of the fields containing a deformation series value.
#
# Remember that the number of values in this option must match the value
# given in label N_VALUES_TIME_SERIES.
#
# -----

OUTPUT_TS_FIELD_NAMES = D20141105 D20141117 D20141129 D20141211 D20141223 \
                        D20150104 D20150116 D20150128 D20150209 D20150221 \
                        D20150305 D20150317 D20150329 D20150609 D20150703 \
                        D20150715 D20150727 D20150808 D20150820 D20150901 \
                        D20150913 D20150925 D20151007 D20151019 D20151031 \
                        D20151112

# End of file.

```

3.4.2 The geologic map read map file

The case of the geologic inventory (or map) is very similar to that of the points shapefiles. In this case, the only attribute needed by THEXfinder to perform its work is the one describing the kind of lithologies corresponding to the polygon to which the aforementioned attribute is related. In other words, THEXfinder needs to know the kind of soil corresponding to each polygon in the shapefile containing the geologic map.

For the sake of simplicity, this document will call such attribute as the “code”. THEXfinder needs to know the position where such code attribute is stored in the attribute’s file.

Furthermore, THEXfinder, in its current version, is only interested in potentially expansive lithologies. There may be many kinds of lithologies that are potentially expansive, so the other thing that THEXfinder needs to know is the list of code **values** that correspond to these.

Therefore, assuming (1) that the input geologic map contains an attribute telling apart the different kinds of lithologies (the “code”) and that it is possible to determine de set of values of such code

pointing to expansive lithologies, then THEXfinder will be able to use the geologic map.

Table 5 depicts the labels used to provide the aforementioned data (the position of the code and the values to consider as expansive lithologies).

Label	Description	Comments
POSITION_FOR_CODE	Position (column number) of the attribute containing the code telling apart the different kinds of lithologies.	Mandatory, integer, 0-based index.
CODE_LIST_FOR_POTENTIALLY_EXPANSIVE_LITHOLOGIES	The list of values pointing to lithologies in the inventory that may be considered as expansive ones.	Mandatory. List of unique integer values. There may be as many values as needed.

Table 5: The attributes in the geology map read map file

The following is an actual example of a geologic map read map file. There, the code to tell apart the different geologic features is located at position 0 (that is, it is the very first attribute in the list). Moreover, there are two different codes that may be considered as potentially expansive lithologies, namely those with values 16 and 71.

```
# =====
#
# This is the THEXfinder's file defining how the fields in the geology
# inventory must be read.
#
# -----
#
# POSITIONS for:
# - Code
#
# -----

POSITION_FOR_CODE = 0

# -----
#
# LISTS OF CODES for the geological phenomenon of interest.
#
# -----

CODE_LIST_FOR_POTENTIALLY_EXPANSIVE_LITHOLOGIES = 16 71
```

3.5 The input (data) files

THEXfinder's input files are the following:

1. the defaults (options) file,
2. the points read map file,
3. the geologic map read map file.
4. the PS (points) file,
5. the optional boundaries file, setting the AOI to actually process.
6. the optional infrastructures inventory.
7. the optional geologic map.

The first three (defaults, point read map and geologic map read map files) have been thoroughly described in sections 3.3, 3.4.1 and 3.4.2 respectively. Next sections talk about the remaining files.

3.5.1 The PS (points) file

Input PS (point) files must be an ESRI shapefile (see [RD2]). Its type, since it will contain points, must be 1 ("Point"). The coordinates of the points must be projected (x & y, that is, easting & northing).

Points in the input shapefile must have a minimum set of attributes for ADAFinder being able to carry out its work. These attributes, stored in the attribute table (.dbf file in the shapefile file set) are:

- X coordinate (easting).
- Y coordinate (northing).
- Velocity (in mm / year).
- Deformation time series (mm / year).

Additionally, the following optional attributes may also be present, but their absence will not prevent the normal operation of the tool.

- Lambda coordinate (deg) and
- Fi coordinate (deg).

Both the mandatory and optional attributes may appear in any order in the attribute table. The read file map (see Section 3.4.1) will be used by THEXfinder to ascertain the actual position of every attribute in the input .dbf file.

3.5.2 The optional boundaries file

Again, the input boundaries file is an ESRI shapefile (see, once more, [RD2]). In this case, however, it must contain one or more polygons defining the AOIs where the actual process must take place. This means that the type of the shapefile must be 5 ("Polygon").

The coordinates defining the aforementioned polygons must be projected (that is, x or easting and y

or northing), using the same coordinate reference frame (CRF) than the projected coordinates used in the PS file.

No table of attributes (.dbf file) is expected. If present, THEXfinder will ignore it.

3.5.3 The optional infrastructures inventory file

This inventory is once more an ESRI shapefile containing polygons defining the areas where several infrastructures (buildings, bridges, etc.) are located. The type of the shapefile must be, therefore, 5 (“Polygon”).

The coordinates defining the aforementioned polygons must be projected (that is, x or easting and y or northing), using the same coordinate reference frame (CRF) than the projected coordinates used in the PS file.

No table of attributes (.dbf file) is expected. If present, THEXfinder will ignore it.

3.5.4 The optional geologic inventory file

This inventory is an ESRI shapefile too, containing polygons as well, in this case defining the shapes of areas with different kind of lithologies. The type of the shapefile must be 5 (“Polygon”) too.

The coordinates defining the aforementioned polygons must be projected (that is, x or easting and y or northing), using the same coordinate reference frame (CRF) than the projected coordinates used in the PS file.

There must be at least one attribute (the “code”, see section 3.4.2 for details) in the table of attributes.

3.6 The output files

THEXfinder creates up to four output files, grouped into two couples: the first file in each couple will store the polygons defining the surface covered by the ADAs; the second one will include those points from the input PS file that belong to the ADAs. Both are shapefiles, so refer to [RD2] for details on the format of these ESRI files.

The two aforementioned couples are those corresponding to the expansivity and thermal effects detection processes. Since the format of these files (ADAs, points) is exactly the same, only one kind of file per couple will be described in the next sections.

Note, however, that each couple holds data referring to the related process from which each are derived. That is, the couple output of the expansivity process keeps ADAs and points corresponding to this phenomenon alone. The other couple will store the information (ADAs and points again) obtained from the thermal effects detection process.

3.6.1 The ADA file

The output ADA file is an ESRI shapefile containing polygons, that is, its type is 5 (“Polygon”). There are as many polygons in the file as ADAs has been detected by the tool.

The coordinates used to define the polygons are projected (x & y, that is, easting & northing) and refer to the same CRF than the one used in the input points file.

The fields included in the attribute table (related .dbf file) are shown in Table 6.

Field	Description
ADA_ID	ADA identifier. This is a sequential number starting at 1 for the first ADA found. Used to link points in the output points file with the ADAs in this one.
N_APS	Number of active points that are included in the ADA.
X_MEAN	Mean X coordinate for the ADA. Computed as the mean of the X coordinates of the active points included in it.
Y_MEAN	Mean Y coordinate for the ADA. Computed as the mean of the Y coordinates of the active points included in it.
VEL_MIN	Absolute value of the minimum velocity of the active points included in the ADA.
VEL_MEAN	Absolute value of the mean velocity of the active points included in the ADA.
VEL_MAX	Absolute value of the maximum velocity of the active points included in the ADA.
VEL_CLASS	Velocity class.
DEF_MEAN	Mean of the last n values of the deformation time series of the active points included in the ADA.
TNI_CLASS	Temporal Noise Index Class. The number (and labels) of the classes depend on the number of temporal noise thresholds set by label <code>TNITABLE_THRESHOLDS</code> in the application's defaults file.
SNI_CLASS	Spatial Noise Index Class. The number (and labels) of the classes depend on the number of spatial noise thresholds set by label <code>SNITABLE_THRESHOLDS</code> in the application's defaults file.
QI_CLASS	Quality Index Class. The values of the classes are defined by label <code>QITABLE_CLASSES</code> in the tool's defaults file.
SINE_R2	r ² coefficient measuring the goodness of the fit of the averaged time series of the points included in the ADA and a sinusoidal function.

Table 6: The attributes in the output ADAs file.

3.6.2 The points file

The output points file is, as usual, an ESRI shapefile (type 1, "Point").

A point included in the input file will be saved in the output file providing that it is part of an ADA.

The coordinates of the points are projected ones (x & y, that is, easting and northing) and refer to the same CRF than the one used by the input points file.

The fields included in the attribute table (.dbf file) are described in Table 7.

Field	Description
E	Easting (x coordinate).
N	Northing (y coordinate)
LAMBDA	Longitude. WARNING: this field is optional, and will only appear when it is present in the attribute table of the input points file.

Field	Description
FI	Latitude. WARNING: this field is optional, and will only appear when it is present in the attribute table of the input points file.
ADA_ID	Identifier of the ADA the point belongs to. When a point belongs to no ADA, this identifier is set to -1. On the contrary, it is set to an strict positive value (1 or greater) when the point belongs to one of the ADAs found.
DEF_MEAN	Mean value of the deformation time series.
Deformation value X.	There are as many fields like this as values in the deformation time series. That is, if the time series include, for instance, 20 values, then there will be 20 of these fields. Each field contains one of the values in the deformation time series. The names of these files are provided in the read / write map file, using label OUTPUT_TS_FIELD_NAMES (see section 3.4.1 and Table 4)
SINE_R2	r ² coefficient measuring the goodness of the fit of the time series of the point and a sinusoidal function.

Table 7: The attributes in the output points shapefile

3.7 Installing the tools

To install the tool please refer to [RD4].

3.8 Using the tool

The tool has two incarnations that must be used in different ways.

To use **the GUI version** it is enough to start the application by whatever means at hand (start menu, desktop shortcut...), fill the form shown by the tool (see section 3.2 and Figure 2 and Figure 3) and click on the “Go” button. Once the process is over, a message box warning about the result (success, failure and its reason) will show up. The process may be interrupted at any moment by closing the GUI window.

Since some of the options may not be changed using the GUI, it is important to remember that these may be modified editing the application’s default file located in the same folder than the tool’s executable. See section 3.3 for details about the contents of defaults (options) files.

To use the **batch version**, a command line must be opened. To start it, type the following command:

```
THEXfinder_cmd <defaults_file>
```

where <defaults_file> stands for the full path and name of the defaults file. See again section 3.3 for details.

Note that in the case of the command line version, the defaults file does not need to stay in the application’s directory. This is so because it is the way to control the application and it is much more convenient let the users manage it at his / her will.

4 Conclusions

THEXfinder is a tool that computes Active Deformation Areas for expansivity and thermal effects phenomena exclusively, using two methodologies, the one defined by the University of Alicante (no references yet) and that reported by It may be integrated as a toolbox (ArcGIS) or plugin (QGIS). Moreover, the kind of inputs and outputs managed may be obtained from GIS tools as well, being therefore no obstacle in a “production” workflow based on GIS tools. The same may be said about the outputs.

The application is configurable, this meaning that it may be adapted to read different kind of input shapefiles, providing these include the minimum set of attributes needed to operate.

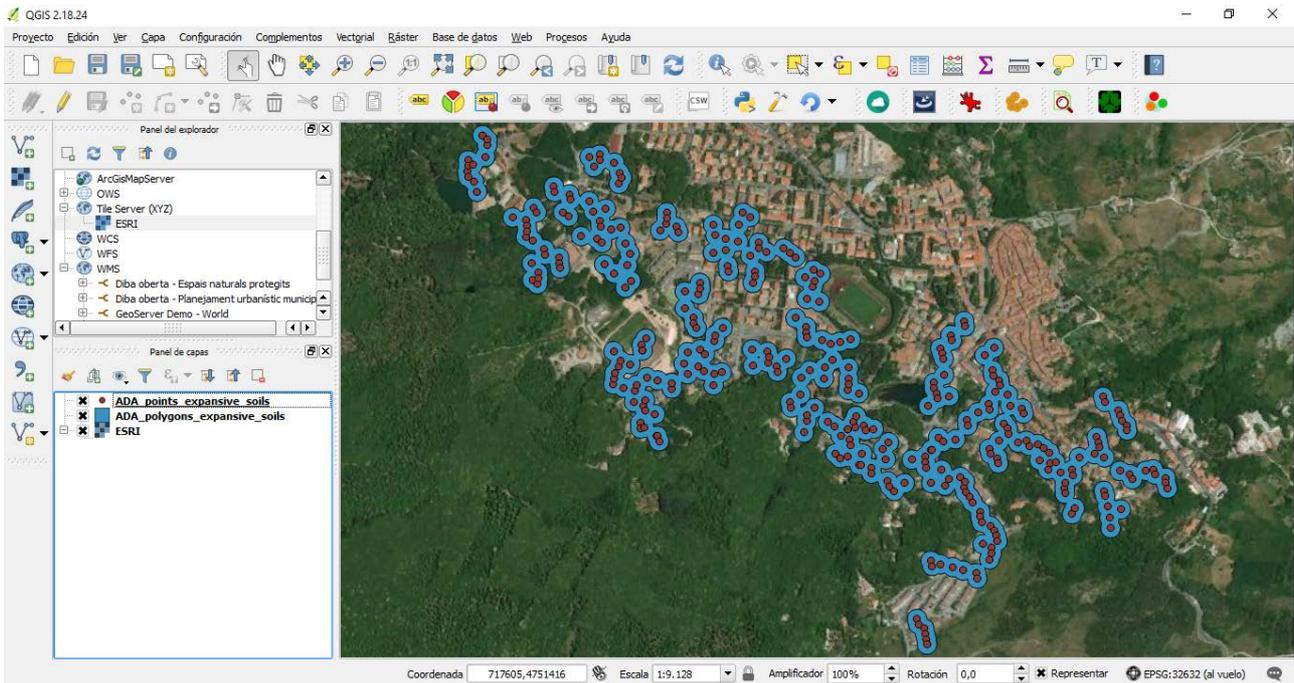


Figure 5: QGIS showing a map with ADAs (blue areas) and the points making them (in red).

Reference Documents

- [RD1] Barra, A.; Solari, L.; Béjar-Pizarro, M.; Monserrat, O.; Bianchini, S.; Herrera, G.; Crosetto, M.; Sarro, R.; González-Alonso, E.; Mateos, R.M.; Ligüerzana, S.; López, C.; Moretti, S. *A Methodology to Detect and Update Active Deformation Areas Based on Sentinel-1 SAR Images*. *Remote Sensing*. 2017, 9, 1002.
- [RD2] Esri. Esri shapefile description. 1999. <https://www.esri.com/library/whitepapers/pdfs/shapefile.pdf>. Last visited: March 13th, 2018.
- [RD3] Navarro, J.; Cuevas, M. MOMIT technical note – Simple options file: technical description.
- [RD4] Navarro, J.; Cuevas, M. MOMIT technical note – Installing the software.

End of the document
