Introduction to AZTool software

Hello, I am David Martin and in this series of short videos I'm providing an overview of the principles of automated zone design and in this third video I'm going to provide an overview of automated zone design using the AZTool software.

AZTool is software for the aggregation of a set of building block polygons into a set of tract polygons in a way that's optimized to meet a set of design criteria and these are principles which have been discussed in the previous video. What AZTool does is to run an iterative recombination of those building blocks from many different random starting points in order to produce an optimal solution, given a specified number of iterations. And its actually in a series of software implementations that deal with this type of zoning problem and in the literature over the years you may find that you can see reference to software called Sage, to ZDES, to ZDK, and to a forerunner of the AZTool software called AZM and they all have all been used in different research applications and studies. AZTool itself was developed by myself and colleagues Samantha Cockings and Andrew Harfoot at the University of Southampton and it uses Openshaw's automated zoning procedure as the underlying basic algorithm. Some of the functionality was previously available in a Visual Basic program which has been published in various papers called AZM, but AZTool is now freely downloadable and available for use. It is programmed in a .NET environment which means that it should run on any modern Windows PC and for any particular version for software it is advisable to read the version notes. So a typical download package for AZTool will include the application file and some example data and probably you would need to look for a readme file and for a set of version notes which will explain any other requirements for where files should be located and the directory structures to be used. But this is software which is downloadable as an executable. It doesn't need explicit installation before it can be run.

What I'm going to do is to talk through the input and control files then illustrate an example of what happens when we run a simple design problem using AZTool and then show the output files which we would expect to get. So the first of the input files are a pair which describe the building blocks and the associated data and they are specified by the .aat and .pat file extensions. They're actually plain text files but they describe the arc attributes and the polygon attributes of the building block data. The arc attribute file explains which building blocks are next to which. So in the left hand file here we can see a series of records, each of which describes the boundary between two building blocks and the three fields are building block ID which is on the left, the building block ID which is on the right and the length of the boundary. In this example you'll see that most of those boundaries are in arbitrary units: they're neat lengths of a hundred or two hundred but also there are some records, such as the third one in this example,where the length is zero and that indicates that those two zones touch at the corner; they come together at a point but the boundary between them is of zero length. You'll also note that there are no geographical coordinates in these files, so it's not necessary to have the full map data in order to use AZTool. Typically, these will have been data which are derived from a map in a Geographic Information System. I'll come back to that in a couple of minutes time. On the right hand side here we see the polygon attribute table and this comprises one record for each polygon, polygons here being the building blocks, with a header line which tells us which each field contains and one final record at the bottom, in this case it's the zone with ID 10, which describes the outside of the map. So in this particular example the outer area, if you like the world outside the map, is zone 10; all of its fields are 0 apart from that last one which is -1 and looking back on the left at the arc attribute table you'll see that some of the arcs are the boundaries between building block polygons and the outside of the map. Those will be the ones where one or other of the first two columns is that 10 representing the outside, and in the map itself – here we have the simple map – we've just set up a configuration with ten coloured polygons. They are slightly irregular, but broadly square shapes and one of them is a surrounding polygon which is the white outside area.

Now the attributes that we might find in one of those polygon attributes tables are going to need to contain all the data which we want to use in the zone design. So, for example, in that example the fifth column was population and population may be one of the variables that's part of our target. For example we might wish every zone to be greater than the population threshold; you might want it to be smaller than a population ceiling; it may have a target size; we would like all the zones to be as close as possible to that size. Therefore we need to know, and provide in this file, the population for each of the building block. We may contain information on the region and that would allow us to subdivide the map so that we only deal with all the zones falling in one region at a time, with no crossover. That's not used in this example but it's an example of the sorts of things that we could include in the polygon attribute table, and we may also have some variables which we want to use for example for homogeneity. So in the toy example here we've got three variables relating to housing tenure and three relating to dwelling type and those could be used if were seeking to design zones which are is internally as homogeneous as possible with respect to those variables. So all our input data is described in those two files. It is possible to use your own software or even manually create those input files but in most circumstances the data will come from a GIS and the AZTImporter utility which is available with AZTool will take widely used ESRI shapefile GIS format and create these two files and we see here the sole interface to AZTImporter. We give it the identity and location of the shapefile it will create the arc and polygon attribute tables.

So if we return to look at the data which I'm going to use in this example we have the map and we have the polygon attributes and we can see that we've got population as one of those and clearly there are many different ways we could set up the constraints for this example and so I'm just going to concentrate on one for the purposes of this video and that's the population column. If we were to map those same data and map the population column associated with each building block, we can see here that the five building blocks at the top of the map have populations of one hundred each. The building blocks down at the bottom of the map have got lower populations, slightly varying sizes, so broadly speaking we might think of this as a as a more densely settled area at the top and a slightly less densely settled area towards the bottom.

The second important input to AZTool is the parameter file and all of the program control is exercised through this XML file which contains all the program parameters. The advantage of working in this way is that it allows us to exactly save the syntax which we've used for a particular run. We can then edit it reuse it. It contains all of the necessary program control parameters, including the locations of the files, and that means that this is quite an effective way of running the program in batch mode so that we may be able to use a Windows batch file to setup multiple program runs with just small variations between them. Maybe we want to move on to another region, want to try zoning different target population size, we've got quite a lot of flexibility by using the XML file and that XML file will open in most text editors or word processing software without problem. So here we see the entire XML file. There's a block at the top which describes the locations of the files to be used; a block in the middle which tells us the AZTool software about the way in which we want to set up our zone design run and then a block at the bottom which has got some of the output characteristics which were seeking to achieve. And if I just look in the detail at the population target block, we see here that form in which its used. I'm setting some target threshold variables and the one that I'm going to work with here is population – it's the text in black. That's the fifth field in the file that were going to use, and I'm actually setting this target as one that I want to use. Now on that map that we looked at, the total population in Toytown here is actually 700 but we'll set the target population as 300, so what we're wanting to do is to find the solution in which each of the output zone is as close to 300 as its possible to get. I have set the tolerance here to a very large number which effectively means that we're not worried about the configuration of the initial random aggregation and we can let it use any values that are wanted. We're not going to worry about that tolerance in this example. And I've also used the default weight of 100 and the weighting here is useful if we set up the problem so that we've got several constraints. It gives us a metric for explaining their relative importance so if I were, for example, to be looking at shape and population total I will give them both the same weight of 100. If I wanted one of them to be twice as important as the other I can adjust those weight values. And then we have a few more key terms: this is going to be set as a minimum threshold, so were going to say all my zones have to be bigger than the minimum threshold value which here we see is 100 and I'm also setting a maximum threshold, set that true and that's 625. So the solution were looking for should be as close to output areas with 300 people in them as possible, never less than 100, never more than 625 and very similar syntax is used as we define the criteria for the different types of parameters available in the software, and even as additional functions have been added this syntax file maintains the same format.

So when we run AZTool effectively all we need to do is to provide the identity and location of that parameter file and the location where we want to write the output, because everything is contained in the parameter file there are no other options or anything else that needs to be done in the program interface, and because we may wish to zone very large problems there's no graphical interface to AZTool either. What it will do is produce a rolling log on the screen as in the right-hand window here which describes the progress, telling us a little about what's going on in each iteration and we see here a report of the way in which the initial random aggregation has been set, how many initial tracts were created and then we see their modification and a record of which was the best of the permutations that was achieved in that iteration. There is a little bit of summary information at the end. This is clearly a very, very simple problem that took 1.36 seconds to solve, so there is an indication of the time that is run and again that's very useful for solving running larger problems.

We get two output files, the first of those is simply a text format log file which is written to the output directory and contains that same progress reporting, so that if we were to leave this running and go home or go and have a coffee we'll be able to come back and see precisely what's been going on and how the zonation been improved. The actual zoning results are written to a CSV format file (comma-separated values) in the output directory and that's the key information for us because it shows, for each of the building blocks, which output tract it's been assigned into and we could reimport those into GIS and then progress to map the results. So here we have the two output files illustrated, one which is a mirror of the on-screen log and the other, here opened in a spreadsheet program, with those two columns, one of which is the building block ID so it's the identity of that small building block that we started with and the second column is the tract ID which tells us where it's been assigned in the final solution. None of the statistical information is written to this file and in this particular instance it's worthy of note that we've got two tracts and they are number 4 and 7.

They are not sequential because these will have been the result of a process of aggregating any number of zones. They will be unique but they may not be sequential numbers.

So in most GIS software what we would do that point, as here, would be to bring the spreadsheet and merg it back by adding it, joining it to the original attribute table. Over on the right hand side in this example we've got the tract IDs – those 4s and 7s added on to the data that we started with – and we can then map that identity and we see the way in which our original building blocks have been aggregated into the optimal tract solution using this very simple configuration which we started with. We've got four of those building blocks in green here being allocated to tract 7 and the remainder to tract 4. There they are if we dissolve the boundaries together (that would be the term that would probably be the name of the function in most GIS software) where all of the building blocks which have got a common value on either side of boundary be dissolved away, and there are the final populations and in this toy example we've been able to find a perfect solution because both of the zones neatly come out with 350 as the population at the output, so they are both as close as we could get to the 300 target. They're both above the 100 threshold; they are both below the 625 threshold. And there they were, the populations in the original configuration just so that we could see where they came from.

So AZTool can be used with a variety of design constraints and one of the features of successive versions of the software has been that additional design constraints have become available and they will always be described in the version notes. But certainly in current versions we can implement constraints within high-level regions, quite a variety of configurations of population targets and thresholds, standard shape compactness which is using circularity index perimeter squared over area, and the set of intra area correlation measures which allow us to control for homogeneity and in the versions most recently available at the time of making this video there are some new accessibility and network connectivity measures.

So in summary, AZTool is a freely downloadable Windows software application which can be used to implement the types of automated zone design which we've been talking about in this series of videos. It aggregates building block polygons into output tracts to best meet a set of zone design criteria. It's supplied with the arc and polygons attributes and controlled by an XML parameter file. Usually those will be data which would have come from a geographic information system. The output log files and tract composition files are the principal outputs and we would usually take the data and re-merge them into GIS software in order to map the results.