Department of Spatial Sciences

A Rule Based System for Assisting the Spatial Adjustment Process

Valerie Allanah Greville Macduff

This thesis is presented for the Degree of

Doctor of Philosophy

of

Curtin University

December 2018

Declaration

To the best of my knowledge and belief, this thesis contains no material previously published by any other person except where due acknowledgement has been made.

This thesis contains no material which has been accepted for the award of any other degree or diploma in any university.

Signature: VAG Macduff ……………………………………………

Date: …01/12/2018……………………...

ABSTRACT

Many vector-based digital datasets such as planning zones, census blocks and utilities are captured in the first instance from a digital cadastre. Following a cadastral update, these datasets would require spatial adjustment to bring them into line with the upgraded cadastre. Spatial adjustment processes require, as input data, control points or links connecting corresponding points between the original and the upgraded cadastre. Existing software solutions require an operator to provide one or more input parameters defining the maximum distance between control points that can be considered for matching. This research describes an automated rule-based system for identifying control points for both urban, rural, and mixed urban and rural datasets with no requirement for operator supplied search distance parameters.

The research has shown that by taking the spatial characteristics of a cadastre into account when matching two vector-based cadastral layers, even those covering rural areas with complex parcel boundaries, purely spatial techniques employing readily available spatial search tools can achieve automated matching of more than 90% of the points defining a cadastral dataset. It presents a novel hierarchical method for matching cadastral points based on first matching cadastral blocks, then the individual parcels and then parcel boundaries before attempting to match the points. New methods for matching points on rural road and riparian boundaries, and for distinguishing between the two, have also resulted from this research.

Automated solutions to the spatial adjustment problem necessarily involve a manual checking and correction stage for the generated links. A further outcome of the research is the automated production of error checking layers to guide an operator to areas that requiring correction.

ACKNOWLEDGEMENTS

The author thanks the following people and organisations for their assistance with this project:

- The Western Australian Department of Planning for the provision of cadastral and dependent datasets at the start of this research project and Matt Devlin for facilitating this provision.
- Landgate Western Australia for permission to use their cadastral data.
- Dr Roger Merritt for the loan of the Spatial Tapestry Workbench software and unstinted assistance with the use thereof, and for his ongoing support at every stage of the research. Thanks also go to Dr Merritt for use of the IntraGIS Map Manager, with which all the maps in this thesis were produced.
- Professor Geoff West, my principal supervisor for most of the duration of this research, and Professor Bert Veenendaal, latterly my principal supervisor, for their critiques, assistance, support, and encouragement.
- Dr Janet Baldwin for her meticulous proof-reading.
- My partner, Michael Marjot, for his proof-reading and helpful suggestions.

The author would also like to acknowledge the contribution of an Australian Government Research Training Program Scholarship in supporting this research.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

GLOSSARY

In this glossary, where no reference is given, the definition refers to the use of the term for the purposes of this thesis. Italicised words in the glossary refer to the glossary entries.

Adjusted layer	An *adjusted layer* is a map layer created by applying the *ArcGIS rubber-sheeting* tools to a layer using either *shift vectors* created by the software developed for this research or *links* created by the *ArcGIS* link generation tool (Esri, 2017a).
ArcGIS	*ArcGIS* is a commercial Geographical Information System (GIS)a package from the Environmental Systems Research Institute (Esri) (Esri, 2018)
Bisector azimuth	A *bisector azimuth* is the azimuth of a line bisecting the *segment angle*
Block	The word *block* is used to refer to any group of contiguous polygons enclosed by the world polygon. In a *cadastral* dataset, such a block is sometimes referred to as a superblock (Merritt, 2005).
Boundary	The word *boundary* can have the general meaning: “the line defining the outer extent of a polygon”, for example, a planning zone or cadastral parcel. The word is used in this thesis to refer specifically to a line connecting adjacent *corners* and/or *nodes* in a parcel (also known as “edges” (Esri, n.d.-b). A boundary between *nodes* defines a boundary between two neighbouring parcels. *Boundaries* are modelled as an ordered string of *coordinate pairs*.
Cadastre	A *cadastre* is a “register of property titles, usually managed by government agencies” (ANZLIC Committee on Surveying and Mapping - ICSM, 2018). The register includes a detailed description of the location of the land *parcel*, the ownership of the land and may hold other details such as the street address, land use, valuation for tax purposes, and others. The location details are typically derived from a survey “produced by a registered/licensed surveyor who accurately measures and records the boundaries of each property” (ANZLIC Committee on Surveying and Mapping - ICSM, 2018).
Centroid	A *centroid* is the point at the geometric centre of a *polygon feature* or a *line feature*. The point may, or may not, fall inside the polygon (Deakin, Bird, & Grenfell, 2002).
Control points	The points of correspondence between two map layers needed for any spatial adjustment process. The ArcGIS rubber-sheeting tool used for this research requires the identification of control points (Esri, 2016a).

Coordinate pair	A *coordinate pair* is a pair of numbers that indicate the location of a point, on the Earth's surface (GISGeography, 2018). All the map layers used for this research were stored in the Australian Map Grid of Australia (MGA) coordinate system (Geoscience Australia, n.d.-c).
Corner	The term *corner* is used to refer to locations on *parcel* boundaries with a close to 90° angle where there is no corresponding *node*, for example, on a corner property at a street intersection.
Difficult-area dataset	A *difficult-area dataset* (DA) is a dataset covering part of a complete *LGA dataset* where the *point matching* algorithms were initially delivering incorrect results.
Densification	*Densification* is the process of adding additional *vertices* to any digitised line at defined intervals without altering its shape (Esri, n.d.-a).
Dependent dataset	A *dependant dataset* is a spatial dataset whose spatial features coincide with cadastral *boundaries* or are offset from those boundaries by known values. *Dependent datasets* can model polygons, lines, or points.
Feature identification number	A *Feature Identification Number* (FID), also known as an Object ID, is a number that uniquely identifies each spatial object in a dataset. The value is usually assigned by the software used to create the data.
Genetic algorithm	A *Genetic Algorithm* (GA) is a software algorithm designed to emulate natural selection to arrive at a solution to a complex problem (Mitchell, 1996).
Identity point	An *identity point* is a point where a pair of matched *control points* are at the same location; the length of the resulting *link* or *shift-vector* is zero.
Identical reverse shift vectors or links	*Identical reverse shift vectors* are vectors where the same pair of *control points* have been identified regardless of whether the *shift vector* generation process was attempting to match old *cadastre* points to new *cadastre* points or vice versa (Siriba, Dalyot, & Sester, 2013).
Island	An *island* is any parcel surrounded by the *world polygon* or by another polygon such as a polygonised lake, river or creek.
Isolated point	An *isolated point* is a *vertex* on a multipoint *boundary* that is a significant distance from the *vertices* on either side of it.
LGA dataset	An *LGA dataset* is a dataset covering a complete Local Government Area (LGA)
Line feature	A *line feature* in a digital vector-based *map layer* is a one-dimensional feature defined by two or more *vertices* (The GIS Encyclopedia, 2011b). A *line feature* consists of one or more *segments. Line features* are typically used to model road-centrelines, power lines, pipelines, and other one-dimensional features.

Link	See *Shift vector*. *Link* is the term used by the Esri ArcGIS software for vectors constructed between matched *control points*. The term *link* has been used in this thesis instead of the term *shift vector* wherever reference is being made to vectors generated by ArcGIS.
Map layer	A *map layer* is a digital representation of a set of geographical entities having a common shape type (polygon, line, or point) and attribute set.
Node	A *node* is a location where two lines meet in a line layer or where the boundaries of three or more polygons (including the *world polygon*) meet (GISGeography, n.d.-b). *Nodes* are not modelled in the *shapefile* format used in this research and *vertices* at these locations may or may not be exactly coincident, depending on the accuracy with which the data were captured.
Object ID	See *Feature Identification Number*.
Parcel	The term *parcel* refers to individual properties in a *cadastral* dataset (Lateș, Luca, Chirica, & Dumitrașcu, 2017). In this thesis, the word is also used for any other enclosed polygon found in the dataset such as polygonised roads or creeks.
Point feature	A *point feature* in a digital vector-based *map layer* is a single *coordinate pair* used to model a single point feature such a lamppost, well, or a drainage inspection hole (The GIS Encyclopedia, 2011c).
Point matching	*Point matching* is the process used in this research to locate points of correspondence between two *map layers* modelling *cadastral* datasets; i.e. pairs of points which represent the same location on the ground. The points of correspondence between two *map layers* are also known as *control points* (GISGeography, n.d.-a).
Polygon feature	A *polygon feature* in a digital vector-based *map layer* is a two-dimensional area bounded by line *segments* forming a closed loop; the point where the edges meet are the *vertices* (Kui Liu, Qiang Wang, Zhe Bao, Gomboši, & Žalik, 2007). Each polygon is defined by three or more *vertices*, usually ordered in a clockwise direction. A *polygon feature* may have one or more holes representing, for example, a lake with one or more *islands*, in which case the *islands*, in the *shapefile* format, would be represented by *vertices* ordered in the opposite direction from the outer polygon (Esri, 1998).
Reverse shift vectors	*Reverse shift vectors* are *shift vectors* created by attempting to match the updated-*cadastre* points to points from the original *cadastre*.
Rubber sheeting	*Rubber sheeting* is a process by which one spatial dataset is warped in such a way as to coincide with another (White & Griffin, 2013)
Salient point	A *salient point* is any point that it is important to match, for example, one having a significant *segment angle* or an *isolated point* on a long *boundary*.

Search distance	The *search distance* is the value used to limit the extent of a spatial search for matching features between two *map layers* modelling the same data – the maximum allowable distance between matched features.
Segment angle	A s*egment angle* is the acute angle formed at the centre *vertex* of three adjacent *vertices* on a single parcel *boundary*.
Segment	The term *segment* or *line segment* refers to the notional straight line connecting two adjacent *vertices* defining a *parcel boundary* (Esri, 2016f). In a vector-based *map layer*, the lines themselves are created at display time when the map is drawn.
Shapefile	A *shapefile* is a digital format for geospatial data originally developed by Esri (Esri, 1998). Each separate geographical feature in a shapefile representing a set of similar features (for example, planning zones, rivers, lampposts) is defined by one or more vertices. *Shapefiles* model *map layers.*
Shift vector	A *shift vector* is a vector that connects the *control points* identified by the algorithms developed for this research. *Shift vectors* are also referred to as just “shifts” in many of the map legends.
ShiftGen	*ShiftGen* is the name of the single program incorporating all the research algorithms.
Slivers	*Slivers* are small *polygon features* that can result from a polygon overlay operation where polygon *boundaries* do not exactly coincide (The University of British Columbia, n.d.). In a *cadastral map layer*, they can also represent easements (Gray & Gray, 2009).
Source point	A *source point* is the current point from the old *cadastre* to be matched – if successfully matched it becomes the “from” end of a *shift vector*. When creating *reverse shift vectors*, the *source point* comes from the new *cadastre*.
Target point	A *target point* is the expected location of a matching point in the new *cadastre*.
Turning angle	A *turning angle* is the clockwise angle formed by three adjacent *vertices* on a single *parcel bou*ndary.
Unique identification number	A *Unique Identification Number* (UID) is a numeric or alpha-numeric attribute that uniquely identifies a *parcel*. In the software developed for this research, UIDs are allocated by the program.
Vertex	A *vertex* is one of an ordered set *coordinate pairs* defining the shape of a polygon or line in a spatial dataset.
Weeding	A process used by the software developed for this research to remove redundant or erroneous shift vectors.
World polygon	The *world polygon* is a notional polygon representing the rest of the world not specifically modelled in a polygon-based map layer (Longley, Goodchild, Maguire, & Rhind, 2011). In the case of a *cadastre*, the *world polygon* may include roads, rivers, and oceans surrounding the property *parcels*. The world polygon typically has no corresponding record in a spatial dataset.

1 INTRODUCTION

This research was conducted to determine methods to aid the process of spatial adjustment that is needed when a cadastre or a portion of a cadastre is resurveyed, upgraded to fit orthophotos, or otherwise amended. The need for a spatial adjustment process arises because vector-based digital datasets such as planning zones, census blocks, electricity supply lines, drainage, and many others are frequently captured in the first instance from, or with reference to, a digital cadastre.

Whenever a cadastre is upgraded after, say, development of a new subdivision or recapture of an area, the datasets captured from the cadastre (the dependent datasets) will also need upgrading. Rather than re-digitising an affected dataset, the dataset can be brought into line with the upgraded cadastre by applying an automated spatial adjustment process. This process would be simple if the apparent movement of an upgraded cadastre's points exhibited a consistent direction and distance, in which case a simple linear transformation could be applied to the points of the dependent datasets. This is rarely the case, however, and different areas of an upgraded cadastre can exhibit large non-linear differences in apparent vertex movements after an upgrade. Where this is the case, spatial adjustment processes require, as input data, many control point pairs indicating the points of correspondence between the original and the upgraded cadastre.

In recent years, several commercial solutions, for example, the Spatial Tapestry Workbench (Spatial Tapestry, n.d.), ArcGIS (Esri, 2017a), Envitia (Envitia, n.d.), that automatically generate control points, have become available but checking and correcting these is a time-consuming process and often requires the trial-and-error determination of input-parameter values, such as search distances, to improve the results. Automated methods that removed the manual intervention currently needed for the determination of suitable search-distance parameter values would be of value (Merritt, 2009).

The research conducted for this thesis aims to improve on existing solutions by removing the need for an operator to provide input-parameter values and to maximise, wherever possible, the number of old-cadastre points correctly matched to points or locations in the upgraded cadastre, even for the most difficult and complex cadastral areas.

1.1 What is a spatial adjustment process?

For the purposes of this thesis, a spatial adjustment process is defined as the process of adjusting one vector-based digital map to bring it into alignment with another.

The process consists of two separate operations. The first operation consists of identifying as many corresponding points as possible (control points) between two vector maps modelling the same data (Walter & Fritsch, 1999). In this research, the control points are the matching points from an old and an upgraded cadastre covering the same area. The output from this process is a set of links (ArcGIS terminology (Esri, 2017a)) or shift vectors (Merritt, 2005) connecting the control points. Each shift vector indicates the direction and distance of real or apparent movement between a matched pair of control points.

The second operation is the actual spatial adjustment. To achieve this, the links or shift vectors connecting the control points are supplied as input to a spatial adjustment algorithm. The final solution described in this thesis makes use of non-linear warping of one dataset to match the other using the *RubbersheetFeatures* tool from Esri's ArcGIS (Esri, 2016b) to carry out the adjustment portion of the complete process. *RubbersheetFeatures* operates on links between control points using the links to adjust each source point to the target point, interpolating unmatched points where necessary. The rubber-sheeting process forms no part of this research which is concerned solely with the first operation – the control point identification process.

Unless otherwise stated, the term "spatial adjustment process" has been used in this thesis to denote the combination of both the control point identification operation and the application of the *RubbersheetFeatures* tool.

Once the shift vectors have been created and corrected where necessary, the *RubbersheetFeatures* tool can be applied to the original cadastre or to any dataset which was derived from it. The output from the *RubbersheetFeatures* tool is a new digital map layer in which each individual matched point defining the spatial features has been moved by the distances and direction indicated by the shift vectors. If every shift vector links corresponding points correctly and all points have been matched, the result of using the shift vectors to adjust a dependant dataset is a new map layer in which it is expected that all the features are in the correct locational relationship with the upgraded cadastre. If the shift vectors are used to adjust the old cadastre, rather than the dependent datasets, it is to be expected that the adjusted old cadastre boundaries will exactly overlay the new cadastre boundaries if, and only if, all the identified control points are correct and all old cadastre points have been matched; this fact has been extensively used to verify the research results.

1.2 Why is spatial adjustment necessary?

Before the advent of computerised mapping systems in the 1960s (Tomlinson, 1974), (Kemp, 2008), the location data for the cadastre would be recorded on paper maps; now the location data is more usually held in a spatial database in a computer (Hashim, Omar, Ramli, Omar, & Din, 2017). A cadastral database holds all the coordinate pairs (vertices) needed to accurately define the location and shape of a property boundary as well as any associated information such as ownership (Effenberg, 2001).

Once such a spatial database exists, for example, the digital cadastre maintained by Landgate in Western Australia (Landgate, 2015), it forms a useful digital basis for the automated or semi-automated capture of other spatial information layers such as road centrelines, planning scheme boundaries, census districts and mesh-blocks, postcode boundaries, drainage, pipelines and many others. For example, the town-planning scheme maps for Western Australia were originally captured using an Arc/INFO application (Western Australian Planning Commission, 1996) developed by this author that allowed an operator to digitally select a group of parcels from the cadastre and assign them to one of a predefined list of planning zones. The application would then dissolve the internal property boundaries for all the selected parcels, thus allowing a thematic map of the planning zones, suitably coloured by zone, to be viewed or printed. Figure 1.1 shows a small area from a Western Australia (WA) town planning scheme as an example.

The precise locations modelled in spatially dependent layers may have legal as well as practical implications, for example, electoral district boundaries, census districts, building density codes or planning zones. Their spatial accuracy is, therefore, important.

A cadastral database is not a static object; each time a property is subdivided or created via a consolidation, an update must be made to the digital cadastre. Also, from time-to-time, a large area of the property boundaries may be re-digitized for reasons such as the availability of higher resolution orthophotography, improved survey accuracy, or redevelopment of an area. The coordinate pairs defining the boundaries of the cadastral parcels can move for a variety of real and apparent reasons, for example, tectonic plate movement (Australia moves about 7 centimetres per annum in a roughly north-easterly direction (Geoscience Australia, n.d.-a)), seismic activity, coastal erosion and accretion, datum changes (for example, AGD84 to GDA94 or GDA94 to GDA2020 in Australia), and improvements in survey mathematics and surveying accuracy. Boundaries on parts of an older cadastre may have been captured by manually digitising from paper maps or orthophotography. Newly digitised data updated into the digital cadastre can result in real or apparent movement of parcel boundaries such that lines and points derived from or offset from the original cadastre are no longer correctly located when overlaid on the new cadastre.

Figure 1.1 shows an example of the type of situation that can arise after a cadastral update. The figure shows a thematic map of a town-planning scheme with the old and the upgraded cadastre drawn on top. The shift vectors (in blue), created by the algorithms developed during this research, connect corresponding points (the control points) on the original and the new cadastres. In this area, there has been an apparent movement of approximately 10 metres, but not in a single consistent direction, between the two versions of the cadastre so that the planning zone boundaries no longer coincide with the new cadastre boundaries drawn in red. Figure 1.2 shows the same area after applying the *RubbersheetFeatures* tool to the town-planning scheme map, using the shift vectors shown in Figure 1.1. The planning zone boundaries now coincide exactly with the new-cadastre boundaries.

Figure 1.1 Map of part of a town-planning scheme before adjustment

The small area covered by this pair of maps has been selected to clearly show the problem and exhibits a perfect solution. In this case, the solution is perfect because every cadastral point in this area has been correctly matched to the corresponding point in the upgraded cadastre and, therefore, the rubber-sheeting algorithm used to carry out the adjustment of the town-planning scheme dataset had no need to interpolate the movement of unmatched points. Automated solutions, however, are seldom perfect across a large dataset; this will become clear from the research described in this thesis.

Figure 1.2 The same town-planning scheme after adjustment

Correct adjustment of spatially dependent datasets is required for more than purely cosmetic reasons. Planning zones, for example, have legal implications so that it is important that their boundaries coincide with the parcel boundaries originally used to define them. In Figure 1.1, the unadjusted road reserve in grey appears, incorrectly, to cut through several new-cadastre parcels to the north-west. Using a GIS viewer, inspection of the available test datasets has shown that the degree of real or apparent movement can be much greater in rural areas than in urban areas, up to several hundred metres in some cases where the course of a creek has changed. Coastal erosion and accretion, too, can result in significant movement of a coastline.

Underground assets also need to be accurately represented on a map so that expensive and potentially dangerous excavations are not undertaken in the wrong place. When a digital cadastre or road centreline dataset is updated, it may occur that the spatially dependent datasets no longer appear to be in the correct places on any printed map that also shows the upgraded cadastre. Serious problems can arise if this discrepancy is not amended. For example, any road works involving the repair of underground utilities could waste time, effort, and money by undertaking excavations in the wrong place. Indeed, excavation in the region of underground cables can also be hazardous; deaths by electrocution are not unknown (Quinlan, 1993). Accurate mapping of underground cables is essential in the prevention of such accidents.

Figure 1.3 shows the cadastre and two drainage inspection holes in an urban area. The old-cadastre parcels are shaded in green and the new-cadastre parcel boundaries are drawn in red. The apparent shift in the new cadastre means that if the inspection-hole dataset was not adjusted before being printed on an updated parcel map the more westerly inspection hole would appear to be wrongly placed on the adjacent property. In this case, the inaccuracy of the new map would result in inspection personnel wasting time visiting a property in the wrong street because the more westerly of the two holes now appears to be located on a different property.

At the time this research commenced, when an updated version of the Western Australian cadastre was released, the Department of Planning was faced with the need to spatially update all their cadastrally dependent layers for every Local Government Area in the state (of which there are 139) comprising more than one million parcels (Devlin, 2009). Any spatial adjustment process that can improve on current solutions will, therefore, be highly valuable.

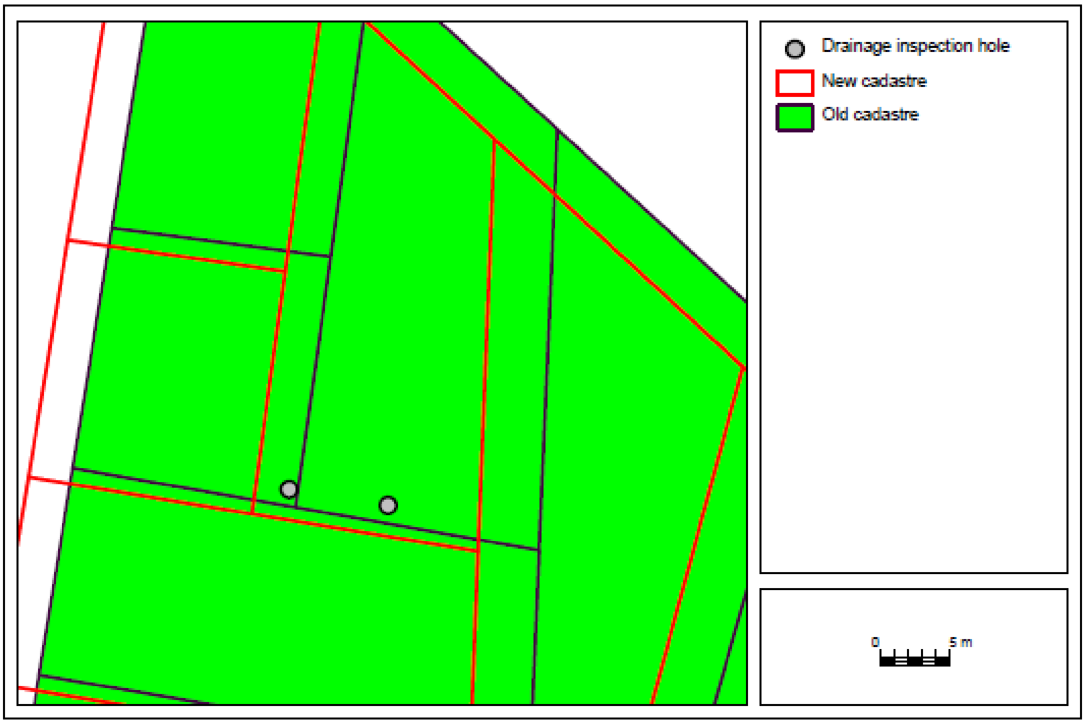

Figure 1.3 Illustration of a problem arising from unadjusted data

A closely related problem not specifically addressed in this research is that of adjusting the cadastre itself to fit with a newly created and accurately surveyed subdivision, i.e. “cutting in” the new subdivision data and adjusting the old cadastre to fit around the edges. Figure 1.4 shows a new subdivision whose outer boundaries do not coincide exactly with the parcel boundaries in the old cadastre. When updating the cadastre to include the new data, it would be necessary to perform an adjustment (upgrade) around the edges. This is, of course, one of the reasons that spatially dependent datasets may also need to be adjusted after the subdivision.

Sometimes the opposite problem occurs: a new survey may indicate that there are areas of inaccuracy in the existing cadastre (as shown in the same figure) but it may not be convenient, for a variety of reasons, to adjust the existing cadastre to fit the new data; the new subdivision data would, therefore, need to be "downgraded" to fit the cadastre (Merritt, 2009).

Both problems involve feature matching, so that the techniques developed during this research will be of value here.

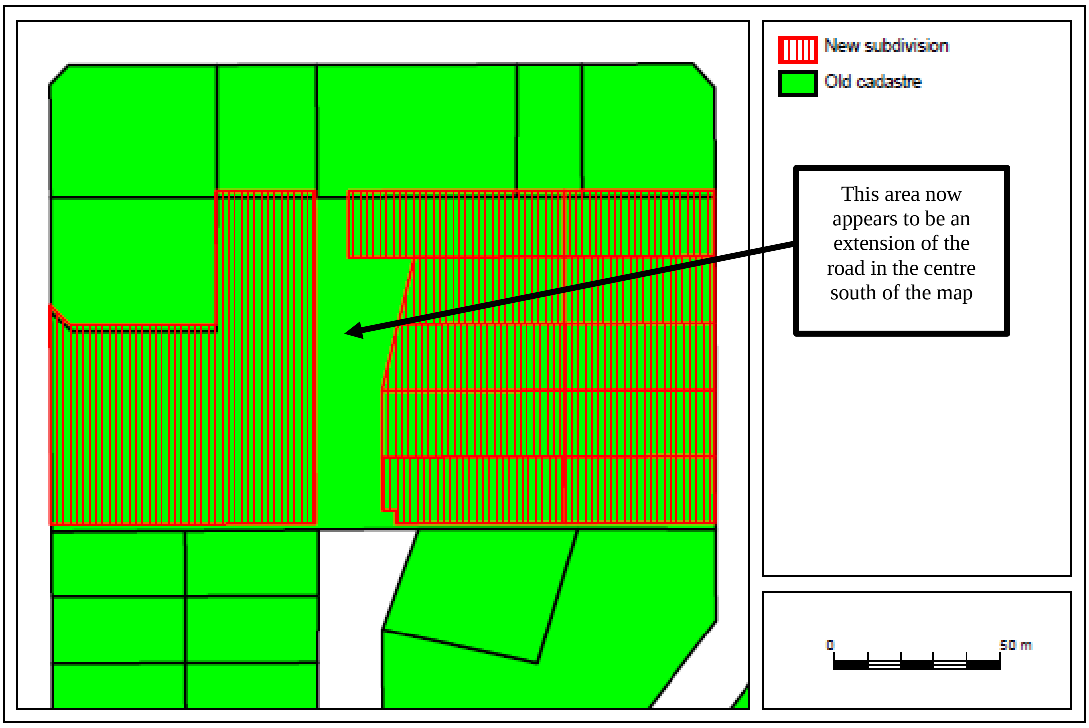

Figure 1.4 A new subdivision

1.3 Existing solutions

Several commercial software packages offer solutions to the spatial adjustment problem (Spatial Tapestry, n.d.), (Civica, n.d.), (Envitia, n.d.), (Esri, 2017a). The solutions involve the automated identification of points of correspondence between the two datasets and the creation of shift vectors (or links) between the original and updated points, followed by a least-squares adjustment or a rubber-sheeting process that makes use of the shift vectors to adjust the spatially dependent data. However, these packages are not always easy to use and require the operator to input one or more parameter values. The simplest solutions, from an operator perspective, typically require a single parameter, usually the maximum search distance, to constrain the search for a match between the original and updated map points. The ArcGIS *GenerateRubbersheetLinks* tool from Esri is an example (Esri, 2016b). A more sophisticated offering such as used in the Spatial Tapestry Workbench requires more parameter values (Merritt, 2009). Both approaches give rise to problems:

(a) The one parameter solution works well on largely homeomorphic areas such as inner-city areas where the positional accuracy was good in the first instance and the apparent movements are small. If a dataset covers both urban and rural areas, a maximum search distance suitable for urban areas can often be too small to allow the correct identification of the control points in rural areas; rural parcels are typically much larger than urban parcels and often exhibit poorer coordinate accuracy. Conversely, a suitably large maximum search distance for rural areas may result in many incorrect matches in the urban areas.

(b) The multi-parameter approach employed by the Workbench software requires a highly skilled operator who is sufficiently experienced to undertake a trial-and-error process to arrive at the optimal parameter values (Merritt, 2009).

(c) The Workbench software can give improved results if there are matching Unique Identifiers (UIDs) on both the original and the updated parcels. Depending on the origins of the two cadastral datasets, there may or may not be matching UIDs. Eight of the twelve cadastral datasets available for this research did not have matching UIDs.

The single parameter problems arise on datasets that cover both urban and rural parcels; the characteristics of urban and rural parcels differ enormously. Inner city parcels are often simple rectangles, sometimes needing only four corner points for their complete spatial definition, whilst rural parcels such as pastoral properties may require many hundreds or even thousands of coordinate pairs to define their boundaries. Now, positional accuracy in urban areas is sometimes so good that an updated version of the digital cadastre will show almost no apparent movement of the property boundaries. Figure 1.5 shows an urban area from a largely rural dataset where the apparent boundary movement is about one metre. Figure 1.6 shows another area from the same dataset (drawn at the same scale) where the apparent north-westerly movement of the boundary is of the order of 25 metres. Both maps show the old-cadastre boundaries drawn in black with the new-cadastre boundaries drawn on top in red.

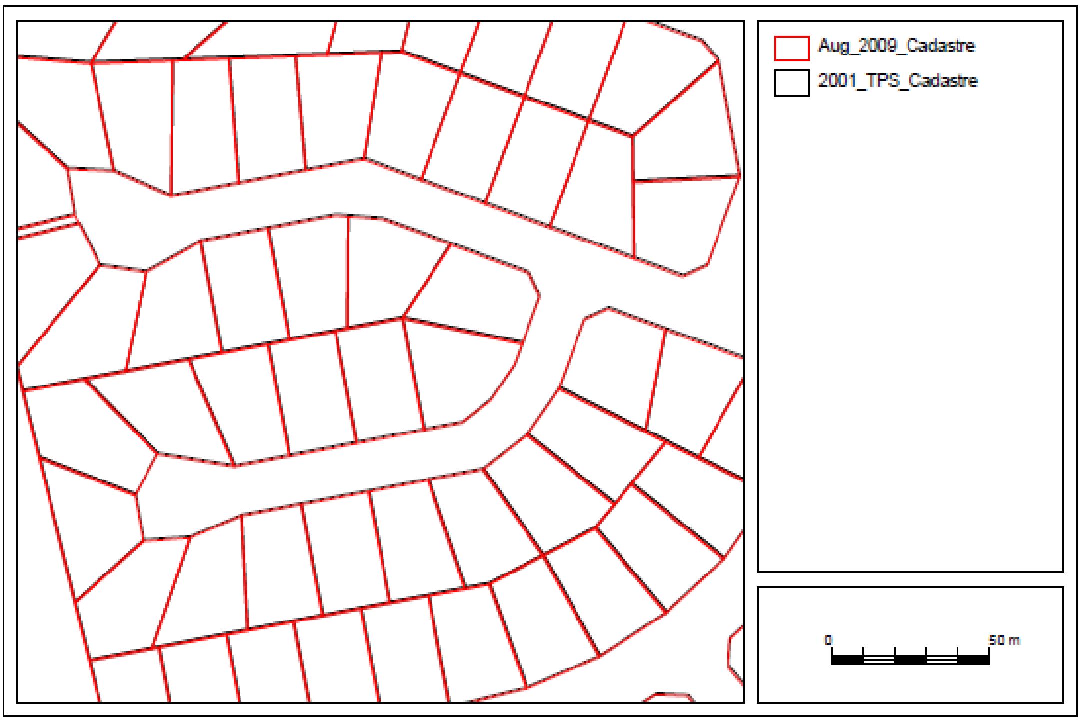

Figure 1.5 An urban area showing little apparent movement of boundaries

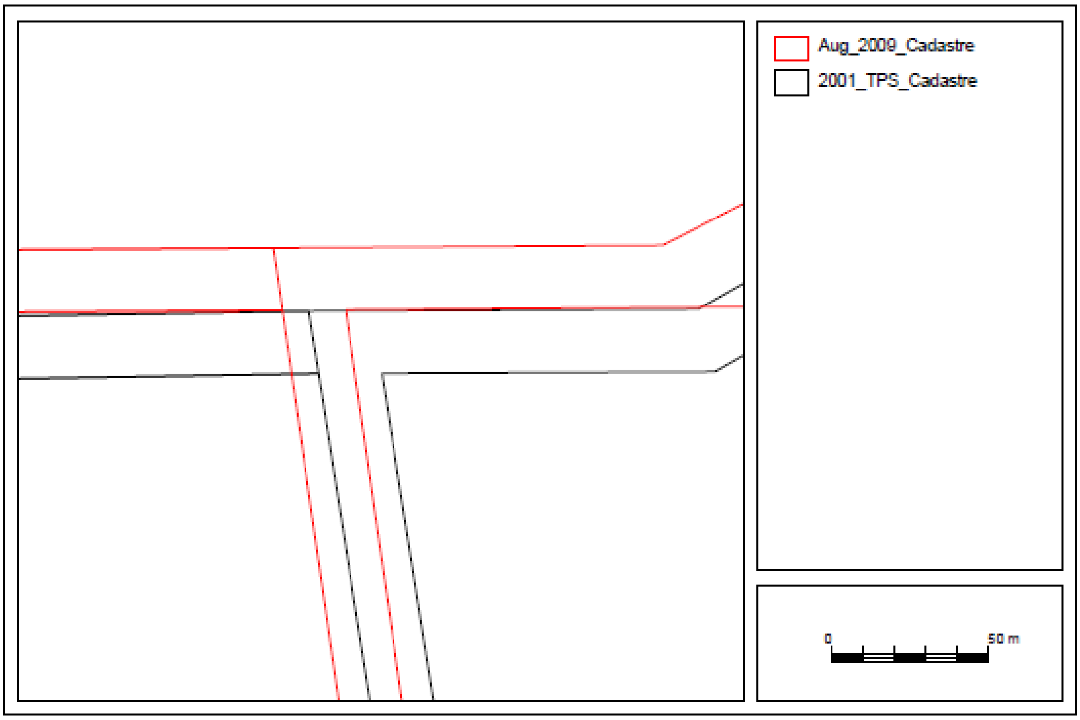

Figure 1.6 A rural area showing large apparent boundary movement

The research conducted for this thesis is justified by the desirability of removing the existing solution limitations.

Much research has been conducted into the correspondence problem and methods of aligning spatial data sets (Gielsdorf, Gruendig, & Aschoff, 2004), (Merritt, 2005), (Guo, Lv, Wang, Zhang, & Cui, 2010). This has been realised in the commercial packages identified by the author, such as those mentioned above. These solutions each require an operator to provide one or more parameters as input to the package. Figure 1.7 illustrates the workflow involved in carrying out a spatial adjustment process using the Spatial Tapestry Workbench software (Merritt, 2009) and Figure 1.8 shows the workflow needed when carrying out the same process using ArcGIS. The upper grey box in each figure encapsulates the shift vector or link creation stage of the spatial adjustment workflow. The lower grey box encapsulates workflow steps common to all spatial adjustment solutions, i.e. the manual checking of the results delivered by the processes in the upper grey box and the application of the generated links or shift vectors to the adjustment of the spatially dependent layers.

The Spatial Tapestry Workbench is a product specifically developed for the spatial adjustment of cadastrally dependant layers. This package requires a skilled operator to undertake an iterative process to determine optimal values for five input parameters that define constraints for the point matching process; these are the processes shown in the upper grey box.

The Esri ArcGIS tools (Esri, 2016b) are generic tools, i.e. tools not specifically designed to adjust cadastrally related data. In the case of the ArcGIS software, the only required parameter is a search distance for the link (shift vector) creation process. Although not suggested in the ArcGIS documentation, it may be necessary to repeat the process with different search distance values, depending on the results obtained.

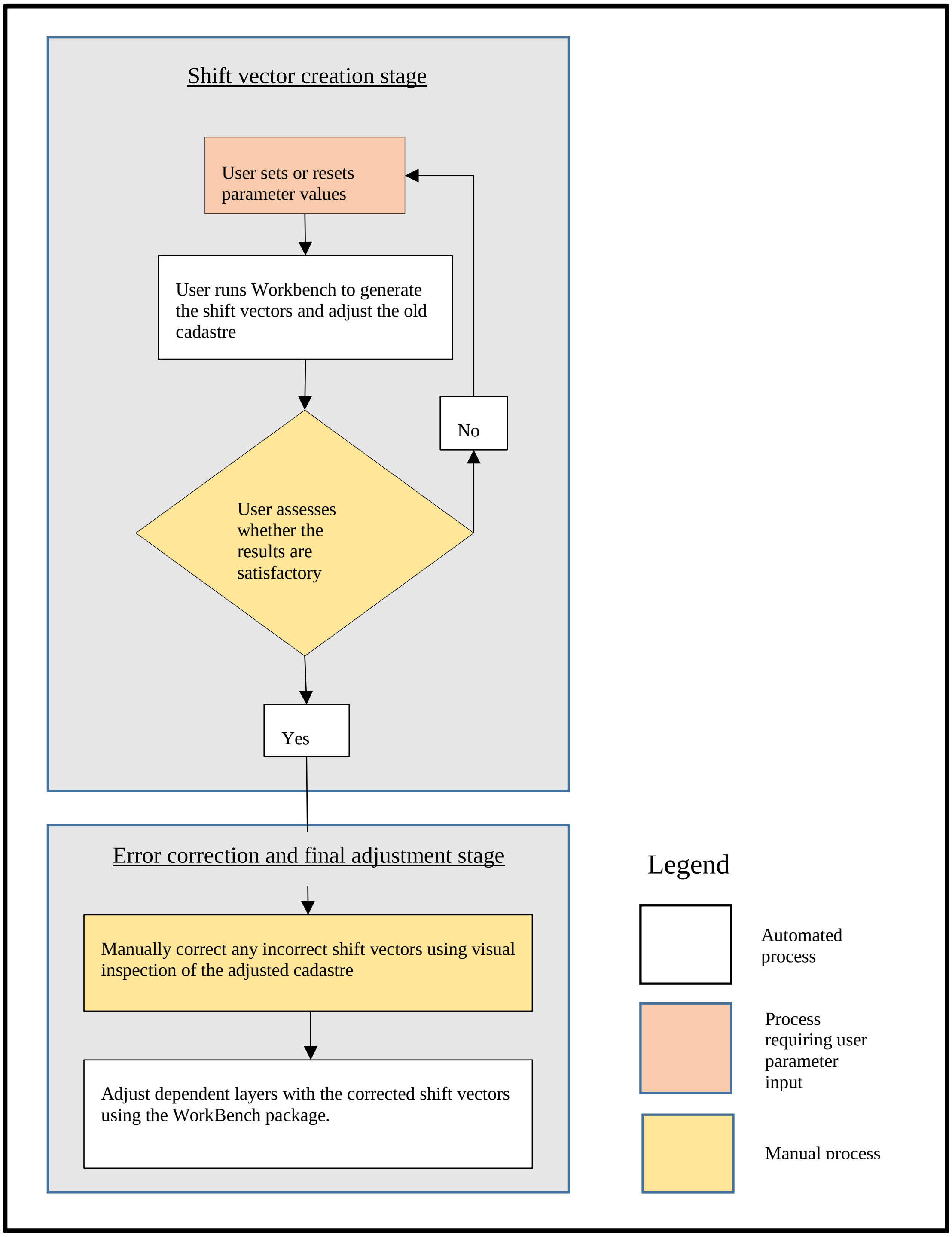

Figure 1.7 The Spatial Tapestry Workbench process

Investigation of existing commercial solutions to the spatial adjustment problem, where the information was publicly available, showed that they also require the operator to enter at least one search-distance parameter to constrain the search for matching control points. These parameters indicate the maximum distance between an old cadastre point and any new-cadastre point that could be a potential match. As this research will demonstrate, where the cadastral datasets cover a heterogeneous area comprising urban and rural parcels and the survey accuracy varies, a single search-distance parameter value cannot result in a satisfactory set of control point matches. Even for a homogeneous urban dataset, it may be necessary to undertake a trial-and-error process to determine the optimum value for the search distance. In the Workbench solution illustrated in Figure 1.7, this is known to be the case (Merritt, 2009).

Whichever tools or products are adopted to achieve the spatial adjustment of the dependant layers, the steps listed in the lower box in each diagram are always needed to eliminate any errors that have arisen from an automated shift-generation process. As stated in the ArcGIS *GenerateRubberSheetLinks* documentation (Esri, 2017a), "The result of GenerateRubbersheetLinks may contain errors […]. **Inspection and editing may be necessary to ensure correct links before using them for rubber sheeting** (author's emphasis)**.**" This manual inspection process by the operator is advisable, no matter what software is used to create the links. This research will show, however, that a software solution developed specifically for cadastral matching can result in several ancillary datasets that can be helpful in guiding the operator to possible-error locations.

In the upper grey box of Figure 1.8, the upper four automatic processes could be encapsulated into an ArcGIS Python script, as could the lower two processes, thus reducing the workflow in the upper box to just three steps. However, the manual intersecting-links removal process cannot currently be automated using standard ArcGIS tools and the scripts would not remove the need for a user supplied search-distance parameter which would need to be determined by several iterations of the process.

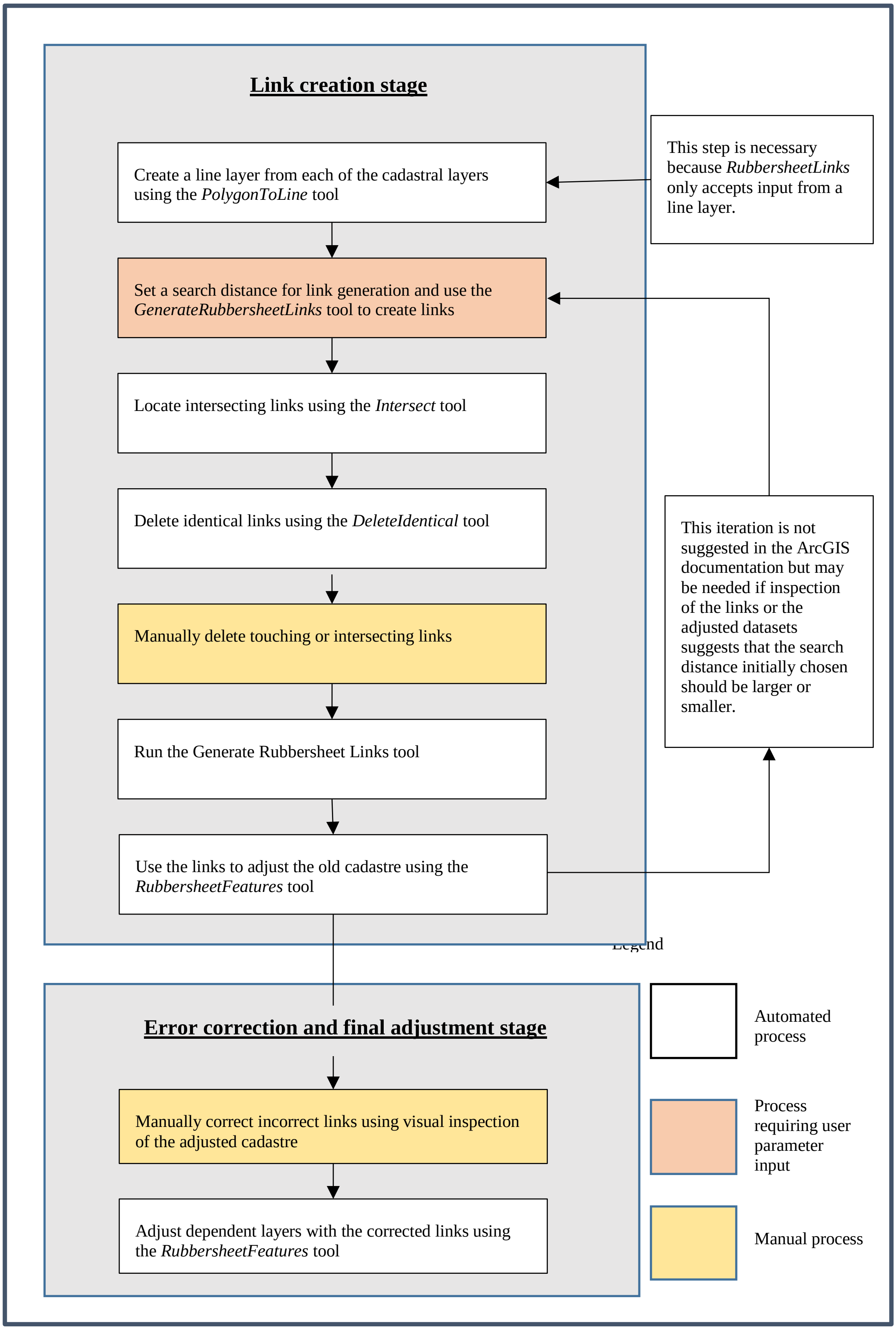

Figure 1.8 The ArcGIS process

1.4 Research objectives

The aim of this research is to simplify the spatial adjustment workflow described in Section 1.3.

The initial approach to the research was to achieve this aim by automatically computing the parameter values needed by the Workbench software and thus remove the user-judgement and iterative aspects of that solution (see the upper grey box in Figure 1.7). When this approach failed (see Chapter 4), the following research objectives were addressed:

1. Remove the need for user supplied search distance parameters required by existing solutions,
2. improve on existing solutions in the correct matching of control points and the generation of shift vectors and
3. determine how to add automation to the current manual error correction processes.

These objectives aim to obviate the need for the manual processes such as iteratively selecting search distance maxima and manually correcting all control point matching errors rather than just those remaining after an automated error removal process.

1.5 Research strategy

Initially, access to a commercial package, the Spatial Tapestry Workbench, was obtained. The Workbench application has the capability to automatically generate shift vectors and also carry out the spatial adjustment of the dependent layers, i.e. it provides a complete solution to the spatial adjustment problem. However, the process is iterative and requires operation by an experienced GIS operator (Merritt, 2009).

The first approach to simplifying the workflow aimed to develop a Genetic Algorithm (GA) to entirely replace the processes shown in the upper grey box in Figure 1.7, i.e. to remove the need for an operator to iteratively refine the parameter values required by Workbench. For reasons described in Chapter 4, this approach to achieving the aim was found to be unsuitable. However, insights gained while attempting to implement this solution were fundamental to the subsequent trajectory of the research (see Section 4.2). Full details of the GA research can be found in Appendix D.

When the GA approach was found to be inappropriate for the achievement of the primary goal, research then concentrated on developing an accurate control point identification process without the need for user supplied search-distance parameters thus simplifying the workflow in accordance with the aim.

The Workbench software, as has been mentioned, comprises a complete spatial adjustment process, i.e. it implements both the control point matching process and the spatial adjustment of the dependent datasets. Once the initial approach to a solution using a GA was discontinued and the Workbench no longer formed part of the solution, it became necessary to use a different tool to execute the actual spatial adjustment process; developing an adjustment algorithm was never one of the research aims. The ArcGIS *RubbersheetFeatures* tool can accept shift-vectors from any source and was therefore selected for use whenever trial adjustments were required.

The change of approach led to the research and development of algorithms to solve several specific problems, the solutions to which were required to achieve the objectives listed in Section 1.4. These problems were:

(a) cadastral parcel matching;

(b) parcel boundary matching; and

(c) parcel boundary classification.

This thesis describes why and how these problems were addressed and how they were solved, with a detailed description of the algorithms finally used in each case.

Figure 1.9 illustrates the final workflow resulting from this research. The software developed requires no search-distance parameters from the user so that no trial-and-error attempts are required; erroneous shift vectors, such as crossing and touching vectors, are automatically removed wherever possible before the final manual error checking stage. The upper grey box showing multiple steps in Figure 1.7 and in Figure 1.8 has been replaced, in the final solution, by the single process developed as a result of this research.

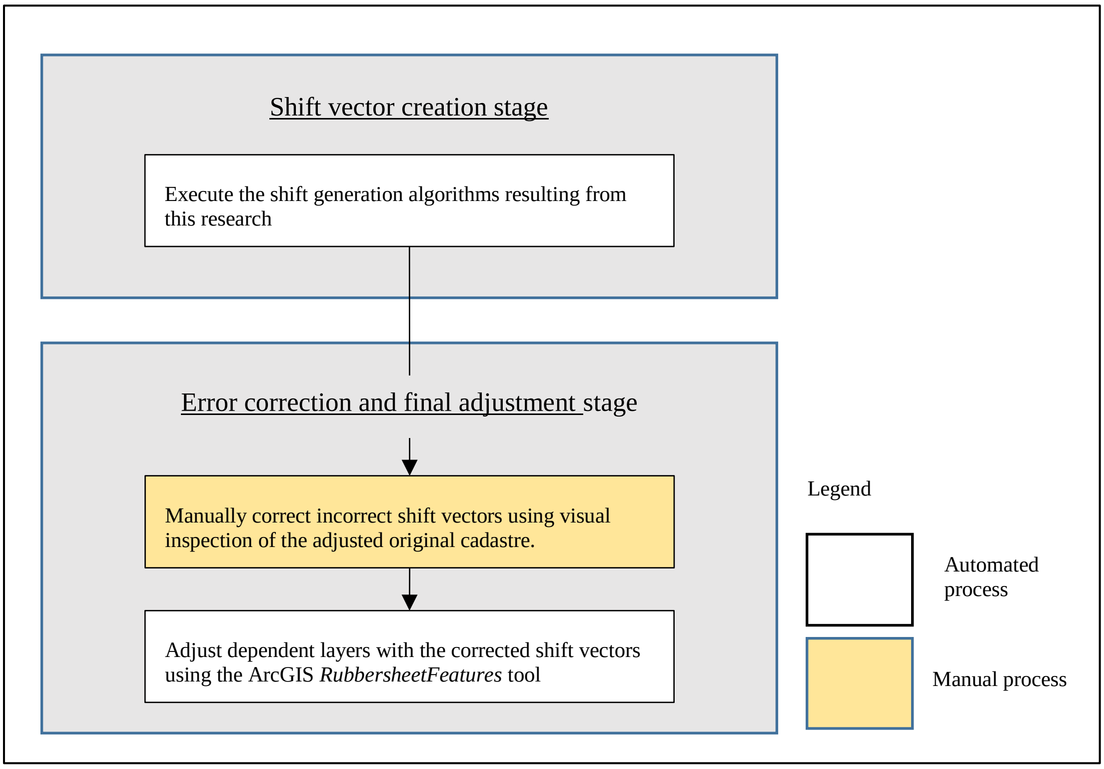

Figure 1.9 Final workflow achieved by this research

It is important to note that this research did not attempt to develop a rubber-sheeting algorithm; the adjustment of the old cadastre incorporated into the software solution was achieved with a call to an ArcGIS Python script that executed the *RubbersheetFeatures* tool, using, as input, the shift vectors resulting from the algorithms developed for this research. The solution resulting from this research matches control points and generates shift-vectors that could be input to any spatial adjustment software, for example, National Geodetic Survey (2017) or DynAdjust, originally developed by Dr Frank Leahy (Leahy, n.d.) of Melbourne University and available at GitHub (2018). The solution is in no way limited to the use of the ArcGIS *RubbersheetFeatures* tool.

1.6 Test data

Twelve cadastral dataset pairs captured some years apart were acquired for this research. They covered both urban and rural areas and exhibited large differences in apparent movement in some areas. They are described in detail in Chapter 3. Some of the datasets were accompanied by spatially dependant datasets such as planning zones, building density zones, pipelines, and drainage,

1.7 Significance of the research

When this research commenced there appeared to be little research activity on the specific problem of the spatial adjustment of cadastrally dependant data although there has been a great deal of research on the problem of feature matching (Lupien & Moreland, 1987), (Saalfeld, 1993), (Yuan & Tao, 1999), (Safra, Kanza, Sagiv, Beeri, & Doytsher, 2010), (Guo et al., 2010), (Chávez, Chávez-Cáliz, & López-López, 2013), (Yanling, Wenjing, & Yuxin, 2008), a process which is a necessary preliminary to carrying out a complete spatial adjustment process. More recently there has been an upsurge in research on cadastral matching and adjustment (Hashim et al., 2017), and polygon matching in general (Ruiz-Lendínez, Ureña-Cámara, & Ariza-López, 2017), (Kim, Yu, & Bang, 2018). Kim et al. (2018) observe that research on "directly matching polygons is not common"; polygon matching is a major component of this research.

Version 10 of Esri's ArcGIS, released in 2010, was their first version to incorporate a tool to automatically generate shift-vectors or links matching individual points, the *GenerateRubbersheetLinks* tool in addition to the spatial adjustment tool, *RubbersheetFeatures,* present in earlier versions. Prior to version 10, users of ArcGIS needed to create the links by identifying matching points manually (Esri, 2009). Commercial companies, of course, do not reveal details of their internal algorithms. An outcome of this research, therefore, is that algorithms developed here, which implement an improved version of the functionality encapsulated in the ArcGIS *GenerateRubberSheetLinks* tool, will be publicly available to all developers in the field, including both open-source and commercial developers. As will be shown in this thesis, the algorithms developed have resulted in a solution to the spatial adjustment problem that removes the requirement for the GIS operator to determine suitable search distance parameter values for the control point identification process whilst also producing large numbers of correct shift vectors and several additional outputs to aid with manual error-checking.

Feature matching (or conflation) alone is an important process in some organisations using GIS (Schuurman, Grund, Hayes, & Dragicevic, 2006), (Kang, 2002) and, as stated by Cobb et al. (1998), "Accurate feature matching results are imperative for rubber-sheeting". The conflation process is often used to transfer or combine attributes from two digital datasets or to improve the spatial accuracy of one set (Song, Keller, Haithcoat, & Davis, 2011). It is envisaged that the methods developed during this research will also assist others attempting to improve the polygon conflation process.

The problem addressed by this research is a common and ongoing one in many jurisdictions (Merritt, 2009). Any solution that can improve on existing commercial solutions will, therefore, be valuable.

1.8 Thesis roadmap

Chapter 2 provides the history of the spatial adjustment process and cadastral data structures and goes on to provide an overview of existing research on topics and terminology underpinning the research in this thesis. Chapter 3 describes the datasets that were acquired for this thesis and the manual and statistical methods employed to analyse the data to gain an understanding of the problems likely to be encountered. The chapter also describes the research methods used to attain the research objectives and outlines the major stages of the research, i.e. the parcel matching, boundary matching, and point matching stages, and the automated error removal stage. Finally, the methods used to test and check the complete solution are outlined.

Chapter 4 details the research undertaken to achieve, first, cadastral superblock matching and, second, parcel matching. It explains the reasons why the parcel matching research was initially undertaken as part of the GA research and why the GA research was discontinued. It also explains the reasons for matching the blocks before attempting to match the parcels.

Chapter 5 explains how point layers were created from the parcel vertices and details the many attributes that were stored for each created point for use in the later point matching algorithms. It describes the way in which the point layers were then reduced to the single most salient point at each location. In addition, the chapter describes the way in which corners were identified on urban parcels to facilitate the boundary matching process described in Chapter 6.

Chapter 6 describes the rationale for creating a line layer from the old-cadastre parcel boundaries and the algorithm used to locate matching boundaries in the new cadastre. It also describes how the riparian boundaries were classified by type as either rural roads or creek banks.

Chapter 7 describes the point matching algorithms used to identify the control points and create the shift vectors needed for input to the ArcGIS *RubbersheetFeatures* tool.

It explains the target point concept and how target points arise from the parcel and boundary matching results and furthermore explains the search distance concept and how this distance, used to constrain the area searched for a matching point in the upgraded cadastre, is computed from the various attributes stored for each point. It describes the way in which three different point matching algorithms were employed, depending on the nature of the point, and the way in which two of the point matching algorithms enable the matching of old-cadastre points to locations on boundaries rather than to existing points in the new cadastre.

Chapter 8 describes the way in which some shift vector errors can be automatically corrected and the methods by which other errors can be efficiently located manually. It describes situations in which correct adjustment results cannot be expected for a variety of reasons.

Chapter 9 provides a broad outline of the final solution and goes on to provide numerical results from the several research stages. The chapter includes many example maps showing successful results in difficult areas. It also includes a section comparing the results from this research with the results arrived at by applying just the ArcGIS spatial adjustment tools to the same datasets. Comparative maps are also illustrated. The following sections summarise how the research objectives listed in Section 1.4 and the specific problems listed in Section 1.5 have been achieved. The last sections describe some unexpected outcomes and discuss problems that remained unsolved when this research concluded.

Chapter 10 summarises the many insights regarding the processing of cadastral data and dependant datasets. Chapter11 discusses potential areas for future research and goes on to summarise the conclusions drawn from each stage of the research.

The next chapter provides a literature review of research relevant to aspects of this thesis.

Note: The legends produced by the software used to draw the maps for this thesis indicate the drawing order of the layers shown on the map. The first layer to be drawn is always shown at the bottom of the legend and the last layer at the top. Where there is no legend, maps or insets showing the results of adjusting the old cadastre using the generated shift vectors have always been created by drawing the new cadastre first in red and old cadastre on top in black. Adjustment errors or topology changes such as subdivisions are thereby indicated by locations where the red new cadastre parcel boundaries are exposed.

2 BACKGROUND

This chapter provides historical information regarding the spatial adjustment process and references earlier research relevant to the research carried out for this thesis.

2.1 Overview

Section 2.2 briefly outlines the history of spatial adjustment and why the need arose. Section 2.3 covers the background material on conflation and feature matching research. Section 2.4 describes research into feature matching. Section 2.5 describes research on shape analysis of polygons and boundary lines. Section 2.6 mentions some applications of machine learning to GIS problems. Section 2.7 covers dynamic segmentation research. Section 2.8 describes research undertaken on the various methods used to automate the spatial adjustment process and Section 2.10 briefly outlines the use of genetic algorithms in GIS research.

2.2 Spatial adjustment history

In the early days of GIS, it soon became apparent that digital datasets were frequently captured by different organisations for different purposes but often represented the same features. There is obviously a potential for conflict between these different datasets (Kang, 2002). The original need for the development of a spatial adjustment process arose in the United States of America where census maps comprising census tracts and blocks were freely available from the American Bureau of Census, but other digital maps modelling some of the same features were captured by municipalities (Kang, 2002), (Saalfeld, 1993). Because municipal maps were usually more accurate than Census Bureau maps, it was deemed desirable to improve the accuracy of the census maps by matching each vertex of the linear features represented, wherever possible, to the corresponding vertex in the municipal map, and then spatially adjusting the census data using a rubber-sheeting algorithm; rubber-sheeting is a map transformation process that preserves topology (Saalfeld, 1988). Saalfeld (1985) was already conducting rubber-sheeting research in 1985 and references an internal U.S.A. Census Bureau document dated 1981; research into constructing Delaunay Triangulations to aid the rubber-sheeting process were being undertaken as early as 1980 (Lee & Schachter, 1980).

More recently, the topic has become of interest because of the development GPS-based navigation systems, vehicle tracking systems, and the rise in driverless vehicle research (Velaga, Quddus, & Bristow, 2012), (Kanchanabhan, Abbas Mohaideen, Srinivasan, & Sundaram, 2011), (Goodchild, 2018).

Several spatial adjustment algorithms such as Affine Transformation (Lembo, 1997), (The GIS Encyclopedia, 2011a), Delaunay Triangulation (Preparata, 1985), and Rubber Sheeting (Lembo, O'Rourke, & Moses, 2003) have been developed to facilitate the spatial adjustment of one dataset to bring it into coincidence with another. However, all the spatial adjustment methods require the initial creation of control points or links indicating the points of spatial correspondence between two maps. In the case of affine transformation, a minimum of three pairs of control points indicating matching locations are required but other methods require more and results improve as the number of correct control points is increased. Geoscience Australia, n.d.-c

Over the years, a great deal of research has been carried out into automating the process of feature matching to generate links between the control points; an overview of this research is provided in Section 2.3.

More recently much of the spatial adjustment research has been undertaken by commercial organisations so that the methods used are not publicly available. Currently, several commercial packages, as mentioned in Section 1.3, offer automated or semi-automated solutions to the complete spatial adjustment process albeit with limitations, described in detail in Section 1.3, which give rise to the need for this research.

2.3 Cadastral data structure history

Spatial database structure has been the subject of a considerable amount of research since the early days of digital mapping, for example, Astrahan et al. (1976), Borges, Davis, and Laender (2001)) and, because of the potentially very large size of cadastral datasets, in particular, it has been important to research and develop ways of storing and indexing the data to facilitate rapid retrieval. This has been an extensive area of research since the early days of digital mapping, for example, Gargantini (1982), Guttman (1984), Arge, Hinrichs, Vahrenhold, and Vitter (2002), Hussain, Krishna Prasad, and Kumar (2011), up until the present day Yan and Igi Global (2016)} and Roumelis, Vassilakopoulos, Corral, and Manolopoulos (2018).

Much of the spatial data storage and retrieval research has been conducted by commercial organisations. Oracle, one of the early relational database products, first incorporated spatial capabilities in the Oracle 4 release, working in conjunction with the Canadian Hydrographic Service, and later developed the Spatial Data Option (SDO) which was released with Oracle 7 in 1992 (Wikipedia, 2018b). Other relational database systems, for example, SQL Server (Microsoft, 2017) first released in 1989, and PostGIS (PostGIS, n.d.) first released in 2001, also offer spatial data storage capabilities.

Esri's early approach to handling such large datasets was the Arc/Info Map Librarian which allowed the database manager to tile the data into suitably sized rectangular areas which could be seamlessly viewed in the ArcPlot GIS viewer (Esri, 1996). Each tile comprised a set of individual computer files holding the individual spatial features, attributes, and topological relationships – "a coverage" (Esri, 2016f). Later, the company acquired the Spatial DataBase Engine (SDBE), a relational database product, from Salamanca Software and rebranded it ArcSDE (Esri, n.d.-c). ArcSDE was first announced at the GIS'95 conference in Vancouver, British Columbia (Wikipedia, 2018a). In the relational format each spatial feature is modelled in a single field in the relational table.

Of late, research into large cadastral databases has concerned itself with 3D representations (Stoter, Ploeger, & van Oosterom, 2013), (Abduhl Rahman et al., 2012), (Vandysheva et al., 2011).

2.4 Feature matching and conflation

Feature matching as a topic of research has, in the past, usually been addressed as the first stage of the map conflation process. Map conflation is a process whereby two or more existing digital maps are combined into a new product that is "better" than either of the originals (Cobb et al., 1998), (Blasby, Davis, Kim, & Ramsey, 2003). The meaning of "better" depends on the ultimate purpose of the conflation process which might be to improve spatial accuracy or to upgrade attribute information.

In the early days of digital mapping, datasets were often captured by a process of manual digitisation from a paper map. Frequently, multiple copies covering the same area and feature content would exist but with different attributes and degrees of accuracy (Song et al., 2011). A map conflation process was therefore needed to allow the creation of new datasets combining the attributes of two datasets and/or to improve spatial accuracy (Siriba et al., 2013). In either case, the first step is to match corresponding features.

The feature matching process can be achieved by a process using spatial techniques alone; Safra et al. (2010), for example, describe three algorithms for matching point-based datasets using two different nearest neighbour algorithms and a *normalized weights* algorithm. Cobb et al. (1998) describe a rule-based system that combines the use of semantic attributes, such as road surface type, with spatial attributes, such as the shape similarity of linear features, to find the correct matching pairs.

If the aim of the conflation process is simply to combine or upgrade the attributes, then the completion of the process is trivial once the features have been correctly matched. However, if the aim is the improvement of spatial accuracy, then one or other of the matched datasets must be spatially adjusted (see Section 2.8).

Siriba et al. (2013) suggest a conflation method for spatial adjustment of a map layer that had been created by digitising the cadastre from a paper map. Their method involves the extraction of road centre-lines from the digital cadastral layer and then matching the nodes and vertices to a more accurate road network. Siriba et al. (2013) also report that the technique is effective but may not result in a large enough set of control points to accurately adjust all the parcel boundaries, particularly in areas far from the control points. Also suggested is that matching accuracy can be verified by a bijective matching of points, i.e. where the two-way match connects the same two points, the match can be considered to be correct, although, as this research will show, that is not always the case.

The map conflation process involves two separate components: the matching of the features followed either by attribute transfer, i.e. the transfer of attribute values from a source feature to the corresponding target feature, or by the spatial adjustment of one of the input layers (Ruiz, Ariza, Ureña, & Blázquez, 2011). To carry out the spatial adjustment, it is necessary to provide the adjustment software with a set of links indicating the points of correspondence between the two layers. In the early days of digital mapping, some GIS software vendors such as Esri provided the tools to automate the spatial adjustment. In the case of earlier versions of the Esri software, it was left to the user to create the necessary links, either as output from a specially developed application or manually; the links would then be input to a rubber-sheeting tool to adjust the dataset (Lupien & Moreland, 1987), (Esri, 2016e).

Much of the early research into map conflation was conducted in the United States (US). A pilot project run by the US Bureau of Census between 1983 and 1985 resulted in “a number of computer science techniques and mathematical generalisations” (Saalfeld, 1993). Much of this early research made use of planar graph matching techniques to automate the point matching component of the conflation process (Saalfeld, 1993) (Kang, 2002). These techniques frequently made use of data held in the Vector Product Format (VPF) mandated by the US Department of Defense (Department of Defense, 1996), for example, the work described by Cobb et al. (1998). In addition, much of the data used as input to the research was based on linear features rather than areal features (polygons). This came about because a large amount of government data available in the US in the early days was linear based, for example, the Topologically Integrated Geographic Encoding and Referencing (TIGER) files available from the Census Bureau (Gakstatter, 2009), (Zandbergen, 2008). VPF is a topological data structure wherein the relationships between points, lines, nodes, areas, and edges are easily accessible to computer code (Department of Defense, 1996).

In recent times, the VPF format and similar topological data structures such as the Esri coverage format (Esri, 2008) have been abandoned by GIS vendors in favour of the format used in spatial database products such as PostGIS and Oracle Spatial. In these products, each separate spatial entity such as a lamppost, a road, or a parcel is held as a coordinate pair (point features) or a set of points (linear and areal features) in a single field in the database table; relationships between the spatial features are not explicitly held in the database but can be obtained using spatial operators (PostGIS, n.d.). The result of this change is that the techniques involving graph theory for point matching are no longer appropriate unless the data are pre-processed to create the topological structure. Even if topological structures were available, enabling graph theory to be employed, the large number of changes that are typically present in different versions of the cadastre issued at separate times, such as new subdivisions, amalgamations, land transfers between adjacent parcels, coastal erosion or accretion, and changes introduced or corrected during the upgrade process, means that there may still be many matching problems to be solved.

Of relevance to the matching processes needed to accomplish the goals of this thesis is research on polygon matching. In a paper by Huang, Wang, Ye, Wang, and Wu (2010) on the topic of cadastral matching, the point was made that most previous research had been on the topic of point and linear matching and that little had been written on the topic of polygon matching even though a large proportion of digital matching is concerned with polygon based datasets. Huang et al. (2010) makes the point that past results have shown that starting with point matching does not result in satisfactory polygon matching. Arkin, Chew, Huttenlocher, Kedem, and Mitchell (1991) working in the field of pattern matching have described a method of polygon matching based on the turning function for the polygon. Kim et al. (2018) also describe an automated geometric method for polygon matching in a GIS using the Hausdorff distance (Hausdorff, 1919), the turning-function distance (Arkin et al., 1991), intersection ratio, and a fusion of the shape similarities. They also point out that GIS polygon matching problems may involve 1:M, M:1 and M:N situations; their solution required a training dataset in order to determine a threshold value for the shape similarity calculated from the spatial characteristics listed above. The work by Huang et al. (2010) has been the basis for the polygon matching algorithms developed for this research.

Topology matching has recently become a major topic of research in the field of transportation because of the rise of GPS in-car navigation systems and automatic driving systems – see Velaga et al. (2012). However, that research is concerned with matching an individual point – the vehicle – to the correct road and is not directly relevant to matching polygons.

Yuan and Tao (1999) has observed that, because of the many different situations that can arise in the map conflation process, a single system to solve all the problems would be difficult to achieve. This has certainly been found to be the case during this research and heuristics have been used to address the many special cases that arise when attempting to conflate two cadastres. A literature search has failed to discover any prior research using heuristics to solve the conflation problem but the method has been extensively applied to the solution of route planning problems see, for example, Deepak (2013), Maier (1995), and Wiener, Ehbauer, and Mallot (2009). Both Wiener et al. (2009) and Deepak (2013) have used heuristics in combination with a GIS.

Research conducted for this thesis into ways to match polygons was extensive (See Chapter 4). Ruiz et al. (2011) discuss the distance measures that can be used when carrying out a matching process for linear features such as Hausdorff distance (Banič & Taranenko, 2017) and Frechet discrete distance (Har-Peled & Raichel, 2014) but do not consider them appropriate for this purpose. These methods assume no correspondence between datasets and matching and error determination are computed iteratively. They assume that each iteration refines the match and reduces the error; then better correspondences are achieved and the match improves. A perfect result occurs when the correspondences are correct and the error is zero. Use of Hausdorff distance was considered in this research for the polygon matching process but the high degree of iteration required to converge on a solution suggested that this approach would be inefficient on large datasets (the largest dataset available for this research (LGA12) comprised over 21,000 parcels with more than 160,000 vertices). The considerable number of almost identical parcels to be found in urban areas also mitigated against this approach. The research, therefore, proceeded using more direct spatial matching techniques requiring only a small number of potentially matching new-cadastre parcels to be tested for each old-cadastre parcel.

2.5 Shape analysis

The analysis of land parcel shapes has been the subject of extensive research although Ai, Cheng, Liu, and Yang (2013) observe that most of this research has been applied to image processing rather than to shape analysis applied to vector data. Demetriou, Stillwell, and See (2013) propose using the multi-attribute decision-making (MADM) method to classify the land parcels according to their shape. They demonstrate that their method is superior to previous methods used by Akkaya Aslan, Gundogdu, and Arici (2007) and Gonzalez, Alvarez, and Crecente (2004). Figure 2.1 shows the results from the Parcel Shape Index (PSI) algorithm developed by Demetriou et al. (2013). The PSI algorithm aims to classify parcels according to their suitability for land consolidation

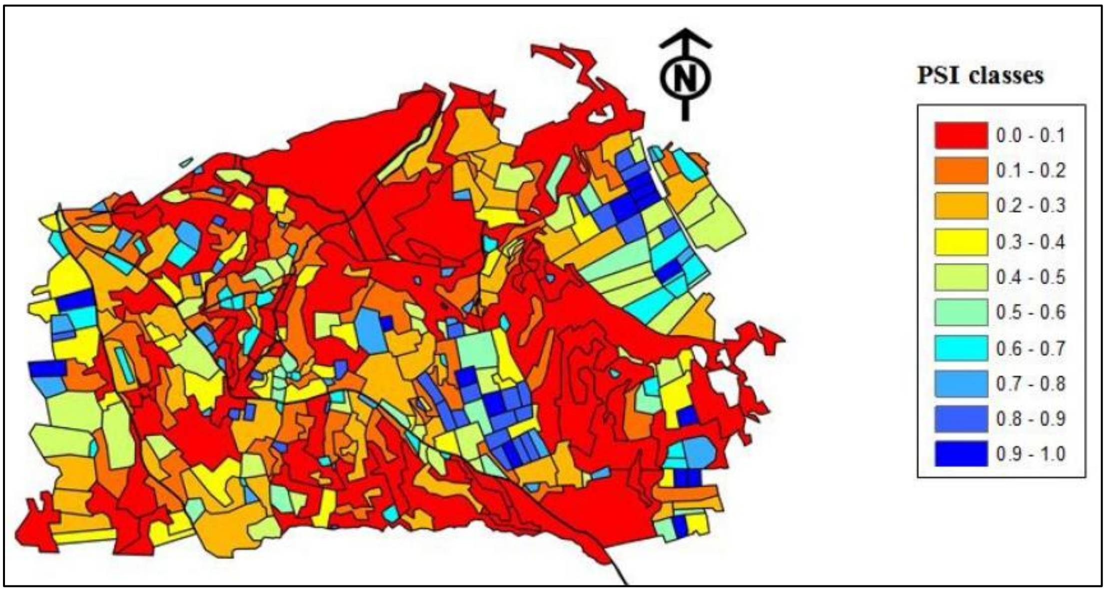

Figure 2.1 Results from the PSI algorithm (Demetriou et al., 2013)

Shape classification was undertaken as part of the research for this thesis but the PSI algorithm was not suitable for the purposes required here; because the algorithm does not incorporate any perimeter to area comparison, it does not distinguish between long narrow irregular parcels such as creeks and other irregularly shaped land parcels – a distinction found to be necessary for the block matching process described in Chapter 4. Instead, use has been made of the circularity index (Tomlin, 1990) along with other factors, in the classification of shapes.

In a vector-based digital cadastre held in a shapefile, each parcel is modelled as a string of vertices. However, in a topologically structured version of the same data, the parcel would be modelled as a series of separate boundaries, each comprising at least two vertices, connecting nodes (locations where three or more parcel boundaries meet). Figure 2.2 illustrates the three boundaries (numbered) that make up the shaded parcel.

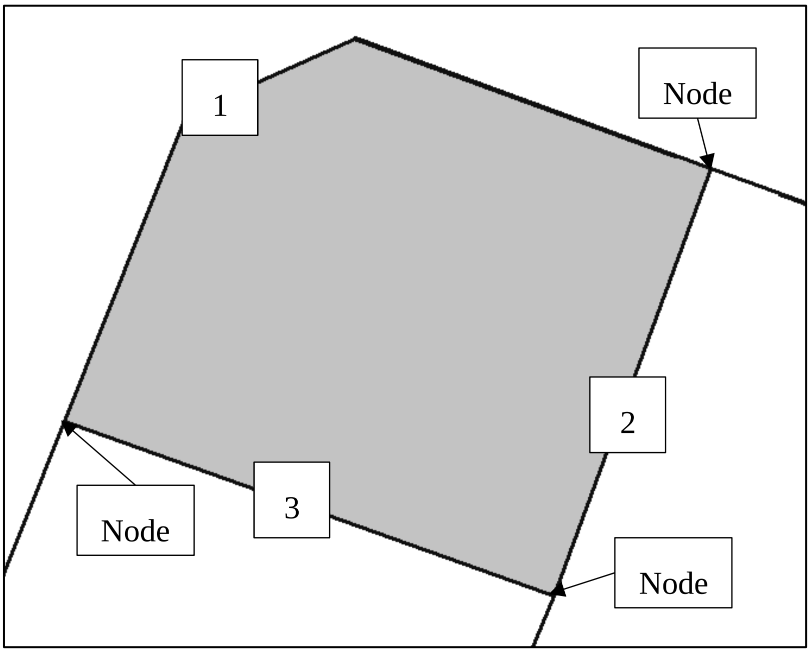

Figure 2.2 Parcel boundaries

When implementing control point matching, research undertaken for this thesis has shown that not all points are equal; the matching of points on rural boundaries may need a different approach from those defining urban parcel boundaries. It was found during this work that the match results would be improved by distinguishing between distinct types of boundaries such as urban parcel boundaries, rural road casements, or creek, river, and coastal boundaries (riparian boundaries).

In general, the shape of riparian boundaries is much more complex than the shape of urban parcel boundaries; rural road casement boundaries tend to have a more complex shape than urban parcel boundaries, but less complex than riparian boundaries. Past research conducted in the field of cartographic line generalisation has proposed several methods for measuring the complexity (or "wiggliness") of linear features, including measurement of the fractal dimension (Carstensen, 1989), and: the mean and variance in angularity between segments; the mean and variance in segment length; and the curvilinearity ratio (Jasinski, 1990). Some of these approaches were investigated in the attempt to classify boundaries into distinct types. However, most of the research concerned with line complexity referenced by Jasinski (1990) was carried out in connection with line generalisation or simplification algorithms and was not found to be directly relevant to the feature matching processes addressed in this thesis.

2.6 Machine learning

The polygon matching research undertaken for this thesis initially attempted to use a machine learning process to discover a rule for polygon shape matching, using the spatial characteristics of each polygon as suggested by Huang et al. (2010). The machine learning software utilised was the open source program – WEKA (Hall et al., 2009). There are many examples of machine learning applied in the field of GIS, for example, Peng, Wong, Nichol, and Chan (2016), Chen et al. (2018), Fanos, Pradhan, Mansor, Yusoff, and Abdullah (2018), and Kobler and Adamic (2000). Some recent example of the application of machine learning to the polygon matching problem are to be found in papers by (Ruiz-Lendínez et al., 2017) and (Kim et al., 2018).

2.7 Dynamic Segmentation (now known as Linear Referencing)

The method developed for this research to match cadastral points that fall on matched boundaries (described in section 7.6) is based on research carried out by the author in the late 1980s to solve the problem of modelling linear data related to transport systems such as highways, rivers and railways (Macduff, 1987). The Esri Arc/INFO software available at that time allowed attribute values to be attached to linear features between nodes but the nature of transport system data meant that many of the data values changed frequently between nodes, for example, skid resistance, surface roughness, and surface material, in the case of highways. These data are collected by odometer rather than by latitude and longitude. Adding pseudo nodes (a node connecting only two line segments) at every point where a data value changed so that their values could be stored against a single linear feature was not an option; it would result in numerous small linear features with many duplicated attribute values (Macduff, 1987), (Fletcher, n.d.), (Dueker, 1992). The solution was to use the odometer reading to dynamically segment the linear features at the map production stage by symbolising the line for distances calculated from the Euclidean distances between the vertices defining the line, i.e. by mapping the odometer distance to a distance along a line feature. At the time, the odometer readings were read in from a table in a text file or database. Although the details of the implementation designed by the author have changed, (odometer readings are now modelled in the *z* coordinates of the line), the functionality is still incorporated in the Esri software (Esri, 2016g).

Figure 2.3 is a screen shot from the author's M.Sc. project report, Macduff (1988), illustrating a dynamically segmented road. The 2.5 Km length of road ends in a node at either end and is modelled in the digital database as a single linear feature. The symbolisation represents the values for the attributes shown in the boxes.

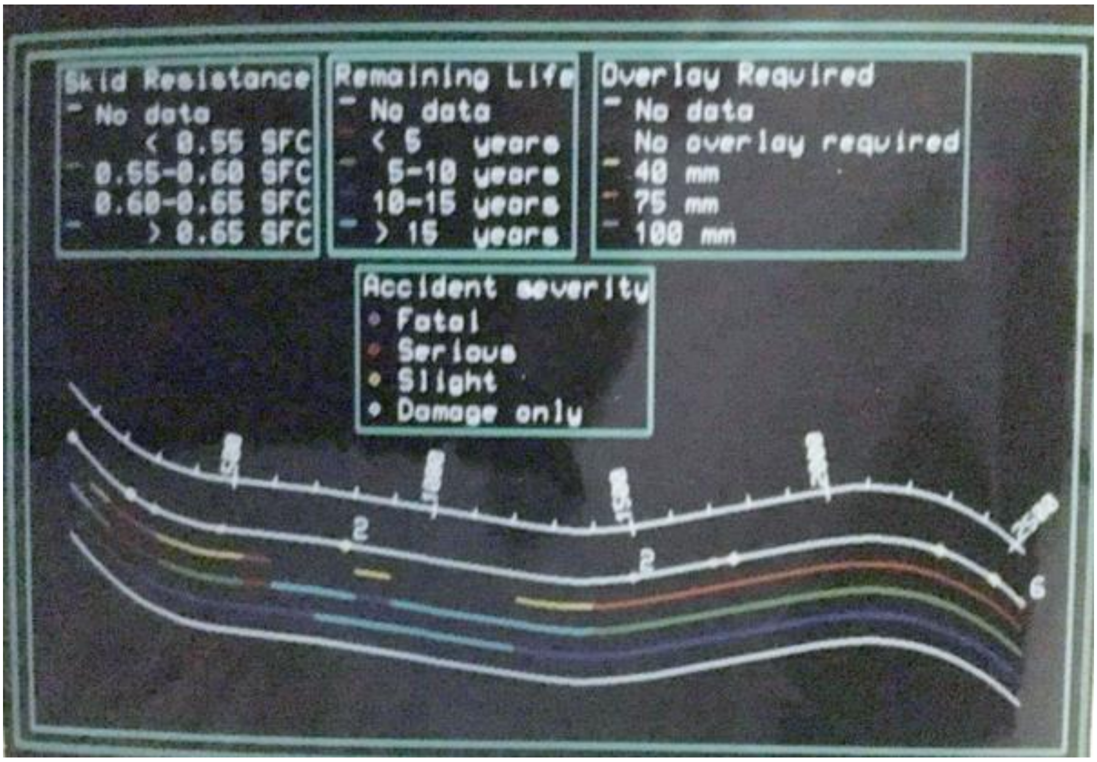

Figure 2.3 A screen shot of a dynamic segmentation display (Macduff, 1988)

A method based on dynamic segmentation was used in this research to match points on complex boundaries.

2.8 Visual inspection of adjustment results

Throughout this research, visual inspection of mapped results had been employed to assess the correctness or otherwise of the results and to improve the algorithms where errors were detected. This technique has been employed by Siriba et al. (2013) to count the number of nodes matched by their algorithm. Kim et al. (2018) have also used visual processes to evaluate their results. Figure 2.4 shows an image from their paper. Velaga et al. (2012) used a GIS viewer (ArcGIS Explorer) to discover matching errors by “visual inspection”. Song et al. (2011) do not specifically say that they used visual methods to evaluate their results but their paper and the illustrations in it strongly suggest so.

Figure 2.4 A figure from Kim et al. (2018)

2.9 Spatial adjustment algorithms

Various techniques have been explored by researchers to automate the spatial adjustment component of map conflation. Some have used a simple affine transformation where "three common point coordinates are sufficient for the calculation of the transformation parameters" (Sisman, 2014). This method is most appropriate for adjusting scanned maps where a degree of rotation is required to bring the map into alignment with a coordinate system. Sisman (2014) also discusses various alternative techniques based on variations of the least-squares adjustment algorithm. Merritt (2005) provides an in-depth analysis of the reasons why spatial adjustment techniques are so important in the field of cadastral management and a comparative analysis of different least-squares adjustment algorithms. Tong, Liang, Xu, and Zhang (2011) also provides a comparative analysis of three Positional Accuracy

Improvement (PAI) algorithms. López-Vázquez (2012) proposes a method using empirical analytic functions using some ideas borrowed from fluid mechanics.

Many of the papers on map conflation already cited in this chapter also cover details of the adjustment algorithms; Kang (2002), and Saalfeld (1993) for example, all discuss the Delaunay triangulation method used in rubber-sheeting. Gillman (1985) gives a detailed description of the triangulation method. Gielsdorf et al. (2004) describes using Delaunay triangulation methods to improve the positional accuracy of survey data in Hamburg. Further information is also available from the Esri ArcGIS help files (Esri, 2016b).

Rubber-sheeting algorithms can also be applied to the adjustment of raster-based information. Shimizu and Fuse (2003) discus a method for rubber-sheeting scanned historic maps using the locations of historic features such as temples, shrines, and castles to identify control points.

The ArcGIS rubber-sheeting process makes use of a triangulated irregular network to effect the adjustment. The method ensures that straight lines defined by only two points in the original dataset will map to straight lines in the adjusted dataset, however, points on lines with multiple intermediate points will not necessarily be adjusted correctly where the points are not matched to a corresponding point in the new cadastre and the algorithm can result in distortions (Merritt, 2005). This aspect of rubber-sheeting algorithms makes it important that as many original points from the old cadastre as possible are matched.

All the different spatial adjustment methods require the identification of control points before they can be applied (Lembo, 1997). In the case of affine transformation, it is only necessary to identify three points of correspondence (Encyclopedia of Mathematics, n.d.). For the other methods, as many correspondence points as possible must be created (Merritt, 2005).

The method developed for this thesis, which matches control points by first matching parcels, has also been used by Ruiz-Lendínez et al. (2017), albeit to match building-footprints rather than parcels. Their method employs a genetic algorithm for the polygon matching stage of the process.

The research conducted to achieve the objectives of this thesis was concentrated on the control point identification component of the spatial adjustment process. Esri's

RubbersheetFeatures spatial adjustment tool was employed wherever an adjustment process was needed for evaluation of the feature matching results or to adjust dependent datasets. Thus, the spatial adjustment algorithms themselves form no part of this research.

The software selected for the adjustment process was from ArcGIS which is one of the most popular GIS packages available and had the advantage that the *RubbersheetFeatures* tool could accept, as input, the shift vectors resulting from this research. The shift vectors could alternatively have been supplied to a least-squares adjustment algorithm, as demonstrated in the analysis by Merritt (2005). This would be applicable if the control points were not correctly identified or the errors between a pair of datasets could not be accommodated by a triangulated irregular network. However, this research has established that ArcGIS rubber-sheeting results can be highly accurate on boundaries where every point has been correctly matched, either to a matching point from the new cadastre or to a location along a matching new-cadastre boundary.

2.10 Genetic algorithms and spatial adjustment

A Genetic Algorithm (GA) is a computer algorithm that endeavours to arrive at a solution to a complex problem by emulating the process of natural selection. GAs require a means of testing the “fitness” of the result at each iteration of the algorithm. Several researchers have used genetic algorithms in the field of feature matching for spatial adjustment. Ruiz-Lendínez et al. (2017) made use of a GA for matching building-footprint polygons using a manually created training database on which to train the GA, i.e. polygon matches resulting from the GA could be evaluated for correctness using the training database. Shnaidman, Shoshani, and Doytsher (2013) and González-Matesanz and Malpica (2006) describe the use of GAs to improve the accuracy of a cadastre. The usefulness of a GA in feature matching and spatial adjustment research is still an open area although its use in this research was discontinued for reasons described in Chapter 4

2.11 Summary

This chapter supplied information regarding how and why spatial adjustment processes were first found to be necessary. It has also provided information on prior research on several disciplines relevant to this research, i.e. feature matching, shape analysis, boundary classification, spatial adjustment, dynamic segmentation., and Genetic Algorithms.

The next chapter will describe the datasets acquired to facilitate this research and how they were initially analysed. The chapter goes on to describe the methods used to execute the research including a description of the four distinct research stages: polygon matching, boundary matching, point matching, and automated error correction and the way in which each iteration of the solution was tested.

3 DATA ACQUISITION AND RESEARCH METHODS

The problem this research is endeavouring to address is, primarily, how the spatial adjustment process can be automated in such a way that an operator has no need to supply any search distance parameters, the value of which may need to be determined by trial-and-error. Initially, the plan was to develop a GA to automatically determine search distance parameter values. When the GA approach was discontinued for reasons described in Section 4.2 and Appendix D, the research approach was then focussed upon finding methods for automating control point identification, again without the need for user supplied parameters. Because the GA approach was discontinued, the research methods described in this chapter all pertain to the subsequent research.

3.1 Overview

To aid the research, twelve pairs of cadastral datasets captured on different dates were acquired. Section 3.2 describes these datasets and the analyses that were carried out. Section 3.3 outlines the research methods adopted and outlines each of the different research stages: polygon matching, boundary matching, point matching, automated error correction, and manual error checking.

3.2 Data acquisition and analysis

Because this research aims to address a real-world problem, it was desirable to gather sizeable datasets covering a range of rural and urban areas in the hope that many or most of the typical control-point identification problems would be encountered. Twelve cadastral dataset pairs issued on different dates up to eight years apart and with different spatial characteristics were, therefore, acquired. They covered a wide range of urban, rural, and mixed urban and rural datasets. Some of these datasets exhibit extreme apparent movements between corresponding parcels in some areas. Spatially dependent datasets such as planning zones, building density zones, and drainage were also obtained for some of the cadastral pairs.

The cadastral datasets acquired each covered a single Local Government Authority Area (LGA). Using LGAs to carry out the research has the advantage that their outer boundaries do not usually intersect cadastral parcels; incomplete parcels would complicate a matching process between two cadastral datasets. The datasets were all stored in the Map Grid of Australia (MGA) coordinate system (Geoscience Australia, n.d.-c); the coordinates represent metres on the ground, hence, all computations concerning distances were carried out using metres (m).

In four of the datasets, comprising a total of more than 21,000 parcels, matching Unique Identifier (UID) attributes were available to check the accuracy of the parcel conflation algorithms.

A visual inspection and statistical analysis of the datasets was undertaken prior to the start of the project.

3.2.1 Manual inspection of the cadastral layers

Using a GIS viewer, each cadastral layer was inspected to assess whether maximum search-distance values could be automatically computed from the data. The inspection was conducted with both the old and new cadastral layers overlaid in the viewer to evaluate the discrepancy between parcel boundaries. The available datasets clearly showed that there is a strong tendency for data from the old and new cadastres to differ more widely in rural areas than in urban areas (see Figure 3.1 and Figure 3.2). These two images alone (from LGA07), suggest that any solution involving a single maximum search distance is unlikely to be effective in a mixed rural and urban area. In Figure 3.1, the apparent movement of parcels boundaries is less than one metre. In Figure 3.2, the apparent movement is more than 50 metres in some places.

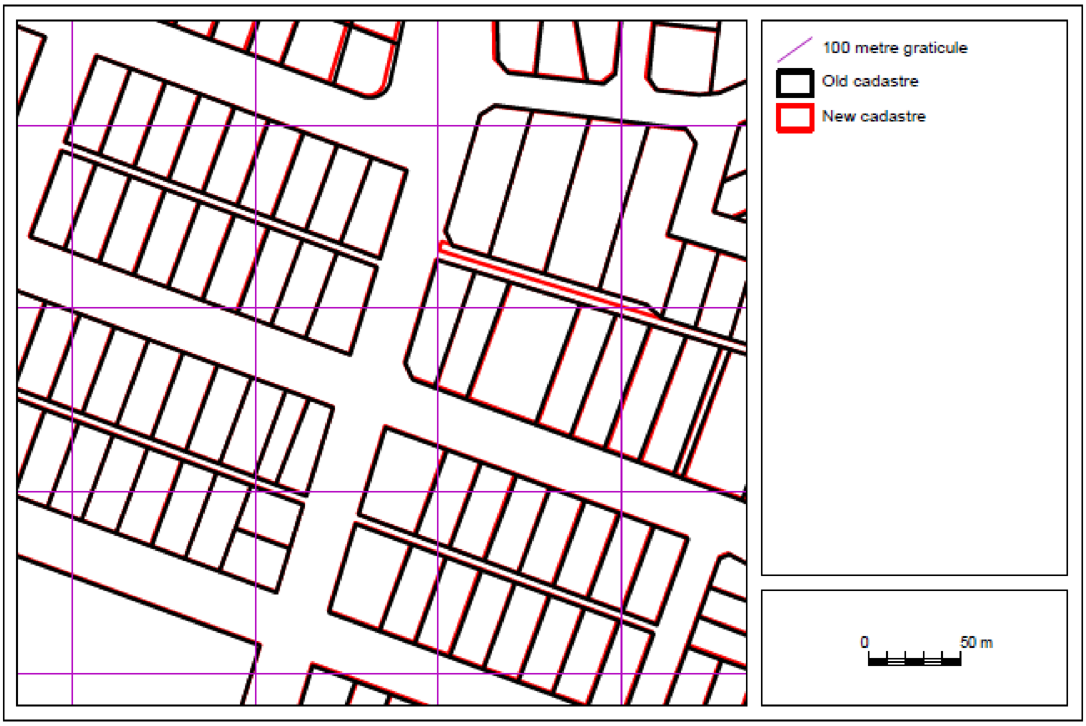

Figure 3.1 An urban area showing little apparent movement

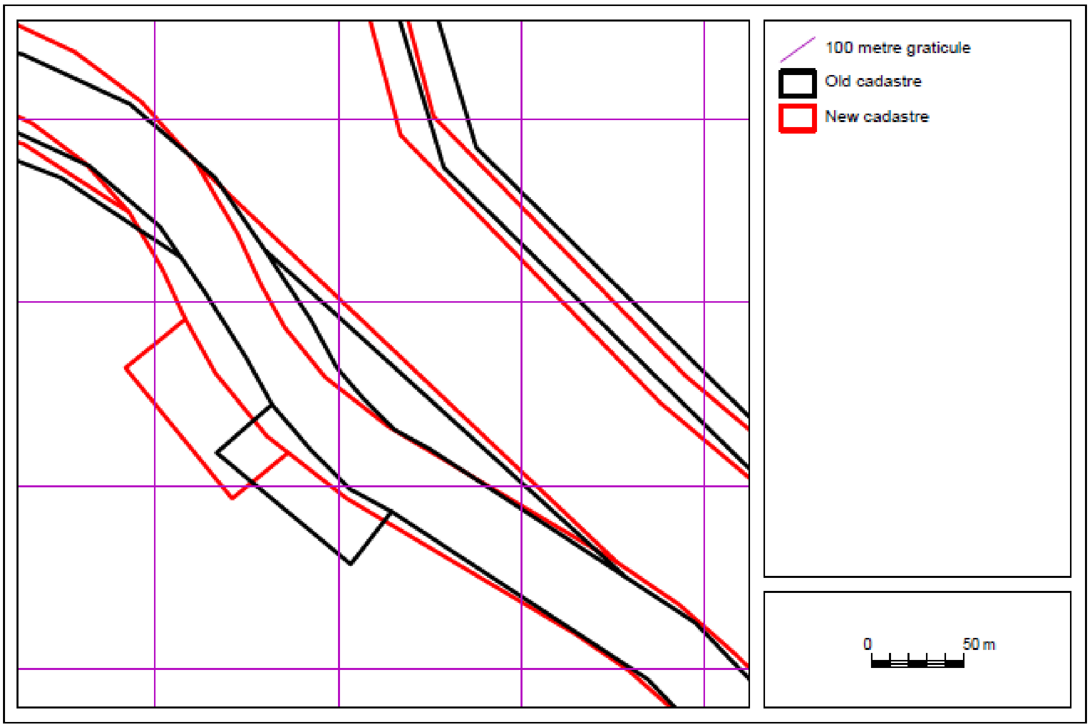

Figure 3.2 A rural area showing large apparent movement

3.2.2 Statistical analyses of the datasets

Inspection of the test datasets revealed, as one would expect, that in the rural datasets most of the parcels were large with a few small towns where parcel areas were tiny in comparison. The urban datasets comprised many small parcels with just a few large ones.

Table 3.1 shows the raw statistics for each of the old cadastral datasets, including the total area of the dataset, the number of parcels, the number of vertices, whether the dataset has a unique identifier or not, and the predominant type of parcel (urban or rural) present in the dataset.

Table 3.1 Raw dataset statistics

LGA ID	Total area Hectares	Parcel Count	Vertex Count	Has UID	Predominant parcel type
LGA01	87.57	655	5,810	TRUE	Urban
LGA02	408.13	1,026	8,519	TRUE	Urban
LGA03	181,029.45	2,258	37,710	FALSE	Rural
LGA04	378.59	2,353	20,574	FALSE	Urban
LGA05	579,666.77	2,785	26,555	FALSE	Rural
LGA06	315,333.16	5,198	68,213	FALSE	Rural
LGA07	79,994.43	5,630	99,443	FALSE	Rural
LGA08	2,364.10	7,262	53,839	TRUE	Urban
LGA09	1,031.81	8,679	98,403	FALSE	Urban
LGA10	789.85	12,239	84,550	FALSE	Urban
LGA11	157,056.16	12,597	146,154	TRUE	Rural
LGA12	2,594.33	21,514	162,467	FALSE	Urban

In addition to the accumulation of the above statistics, computer code was also developed to accumulate statistics for the areas of parcels in each input cadastral layer, both for subsets of the data defined by area ranges and for the complete layer. Area ranges were arbitrarily defined for this analysis. The values were:

0 > Area <= 1 ha (Small area)
1ha > Area <= 10 ha (Medium area)
Area > 10 ha (Large area)

Figure 3.3 shows the total area of parcels in each area range as a percentage of the total area of all parcels in each old cadastre layer.

Figure 3.4 shows the number of parcels in each area range as a percentage of the total number of parcels. The datasets for LGAs 3, 5, 6, 7 and 11 cover predominantly rural areas. The remainder cover predominantly urban areas.

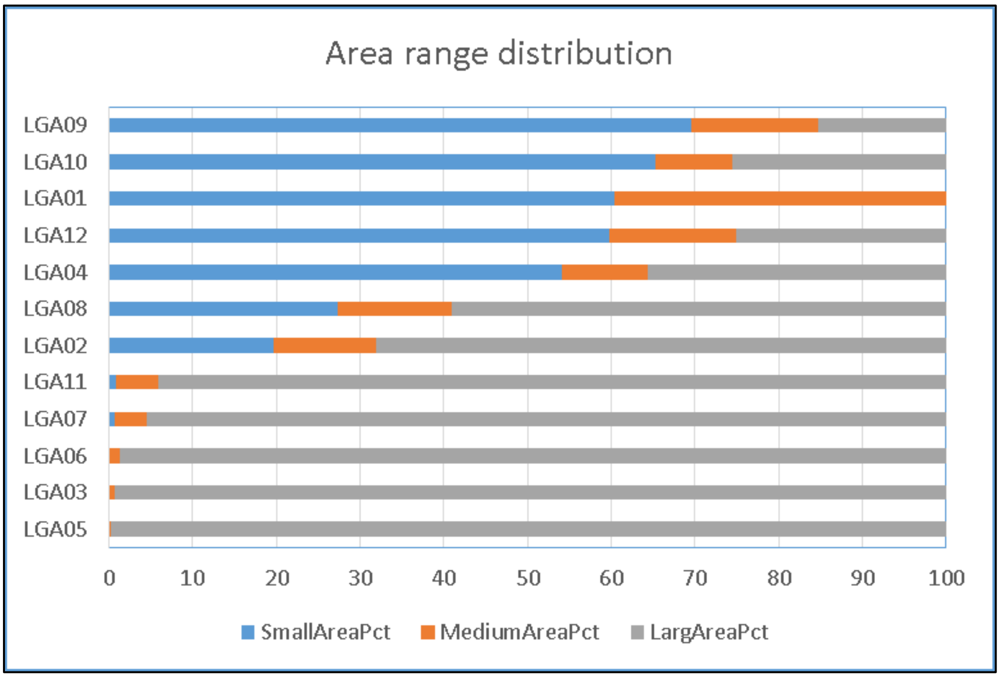

Figure 3.3 Parcel area range percentages

It was immediately obvious, from inspection of the results shown in the charts displayed in this section, that there is a significant difference in the range of parcel sizes found in urban areas and the range of parcel sizes to be found in the rural areas.

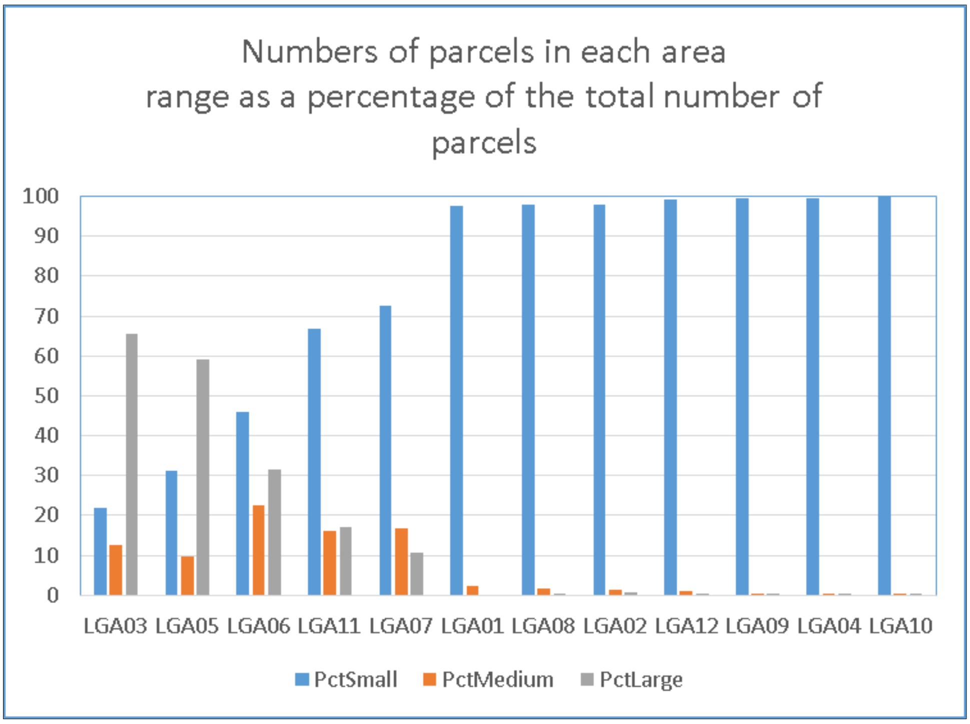

Figure 3.4 Parcel counts for each old cadastral dataset

The results of the area analysis for all parcels in each LGA are shown in Table 3.2. The results show a very distinct difference in the mean and variance in parcel size between urban (the first seven in the list) and predominantly rural datasets (the last five in the list).

Table 3.2 Statistics for all parcels in each test LGA ordered by increasing standard deviation of area

LGA ID	Dataset type	Parcel count	Minimum area (ha)	Maximum area (ha)	Mean area (ha)	Area standard deviation (ha)
LGA10	Urban	12239	0.00	15.36	0.06	0.23
LGA01	Urban	655	0.00	8.43	0.13	0.44
LGA12	Urban	21514	0.00	36.43	0.12	0.49
LGA09	Urban	8679	0.00	90.50	0.12	1.08
LGA04	Urban	2353	0.00	70.10	0.16	1.99
LGA08	Urban	7262	0.00	128.27	0.33	3.31
LGA02	Urban	1026	0.00	91.31	0.40	4.32
LGA03	Rural	2258	0.01	1,661.32	80.17	121.51
LGA07	Rural	5630	0.00	10,116.24	14.21	169.27
LGA11	Rural	12597	0.00	12,438.19	12.47	172.34
LGA06	Rural	5198	0.00	8,337.48	60.66	303.67
LGA05	Rural	2785	0.00	13,616.60	208.14	446.43

From these analyses and inspection of the data in a GIS viewer, it became clear that a single set of the search-distance parameter values required for existing spatial adjustment solutions would not be appropriate for all the data in any one dataset. The rural datasets showed larger differences in apparent movement in the rural areas than in the town areas of the map. Search distances would, therefore, need to have larger values in the rural areas if points are to be successfully matched.

3.3 Research methods

To achieve control point identification in any pair of vector maps covering the same area, it is first necessary to carry out a conflation or feature matching process as a preliminary step (Song, 2011). In the case of a cadastre, the features to be matched by the conflation process are the parcels and their boundaries. This section describes the methods adopted for each of the conflation operations undertaken and for the control point identification process.

Ideally, a solution to the spatial adjustment problem requires the matching of every unique point from an old cadastre to a corresponding point or a location in the upgraded cadastre. Once this matching has been achieved, shift vectors can be created from the old point to the new point, and the resulting vectors can be used to drive any commercial software package, such as ArcGIS, that can accept these vectors as input and use them to carry out the adjustment of the spatially dependent datasets.

In practice, it is highly unlikely that any solution could produce a 100% correct match from every source point because cadastral updates and upgrades can result in large differences in apparent movements in different areas and there may also be topology changes to complicate the process. Coastlines, rivers, and streams, where sandbanks can appear and disappear between versions, present particular difficulties. The research has therefore employed heuristic techniques as it is not to be expected that one simple algorithm could be developed to completely solve all the complex problems likely to be encountered (Yuan & Tao, 1999).

The first task undertaken (initially as part of the GA research) was to attempt to match as many parcels as possible between the old and new versions of the cadastre (Stage 1). Where available, UIDs were employed to check the correctness of the matches. Where they were not, the results were thematically mapped in a GIS viewer and inspected for correctness. Once an acceptable matching algorithm was developed, the research moved on to attempt to match all the individual points (Stage 3). However, inspection of the results from rubber-sheeting the old cadastre using the generated shift vectors suggested that improved results would be obtained by matching the individual parcel boundaries first and so research was then concentrated on achieving boundary matching (Stage 2). In this case, manual inspection using a GIS viewer was the only method available for evaluating the correctness of a match. Once a solution to boundary matching that was satisfactory to the author was achieved, research thereafter was concentrated on point matching and shift vector generation (Stage 3). Each individual stage underwent iterative improvement of the algorithms. From time-to-time, inspection of results suggested that improvements could be made to earlier research stages so that research then reverted to that stage.

Automatic shift vector generation can result in crossing and touching shift vectors; if these are not removed before using them for rubber-sheeting, the result can be topological errors in the adjusted layers. Stage 4, therefore, was concentrated on developing algorithms to automatically correct as many of the crossing and touching shift vectors as possible whilst retaining those most likely to be correct.

At each stage, for particularly difficult areas where the portion of the solution being tested gave poor or incorrect results, a smaller dataset encompassing the problem area was extracted for further testing. In total, 33 particularly difficult areas from several different datasets were extracted for algorithm testing purposes – the Difficult Area (DA) datasets.

At each stage, a detailed inspection of the mapped results from these DA datasets was undertaken to understand why an existing algorithm was failing and to determine whether the algorithm could be improved in a way that would be applicable to all similar areas of difficulty and without adversely affecting other areas.

Figure 3.5 shows an outline of the research methods employed.

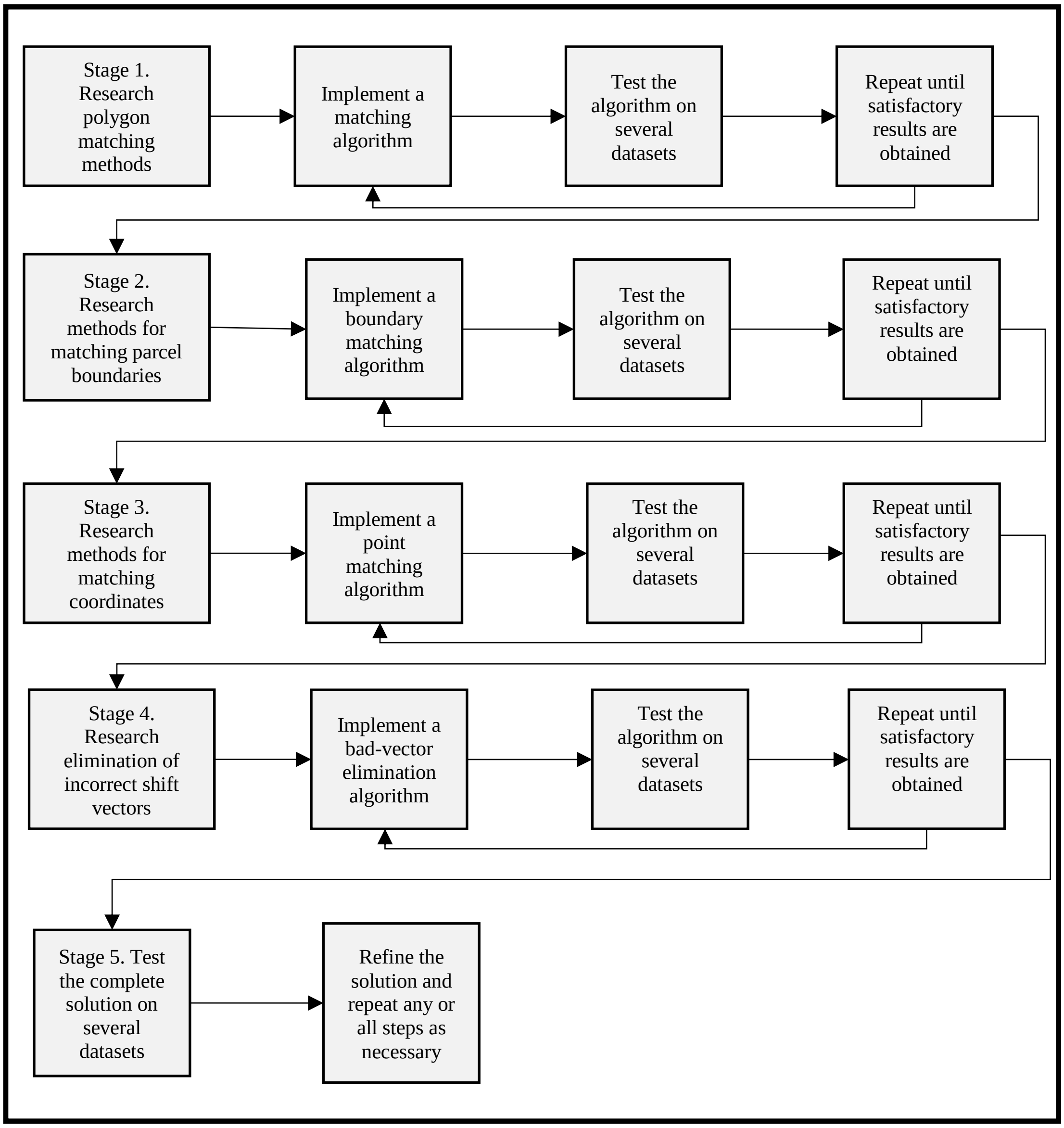

Figure 3.5 Outline of research methods

3.3.1 Methodology used for evaluating results

Only for the parcel matching stage of the research (Stage 1, detailed in Chapter 4) was an objective method discovered for checking the correctness of the match results, i.e. the parcel UIDs available for four of the LGA datasets could be used to compare the match created by the parcel matching algorithm to the correct result. In the later stages of the research, after the point matching and shift vector generation process described in Chapter 7 had been implemented, it was possible to carry out a trial adjustment on the old cadastre following each test run. The trial adjustment could then be inspected in a GIS viewer to assess the effectiveness of any algorithm change.

Using the DA datasets, by inspection of the adjusted old cadastre overlaid on the new cadastre in a GIS viewer, it was possible to efficiently locate areas where the results were poor and consequently identify the need for algorithm improvement. The early inspections proceeded by drawing the adjusted old-cadastre boundaries in black over the new-cadastre boundaries in red. This method clearly reveals any adjustment errors because the red boundaries can only be seen where the adjustment is poor. Figure 3.6 shows an area where most of the old-cadastre adjustment is correct but there is a problem with the parcel in the northeast. In all, over 3000 tests of this type were conducted on the DA and full LGA datasets.

Later inspections used the same method but were guided by secondary outputs, described in Chapter 9, that could pinpoint areas of potential error very rapidly.

Whenever a poor adjustment result was observed, an attempt would be made to refine and retest the algorithms until the mapped results appeared satisfactory in the subjective opinion of the author.

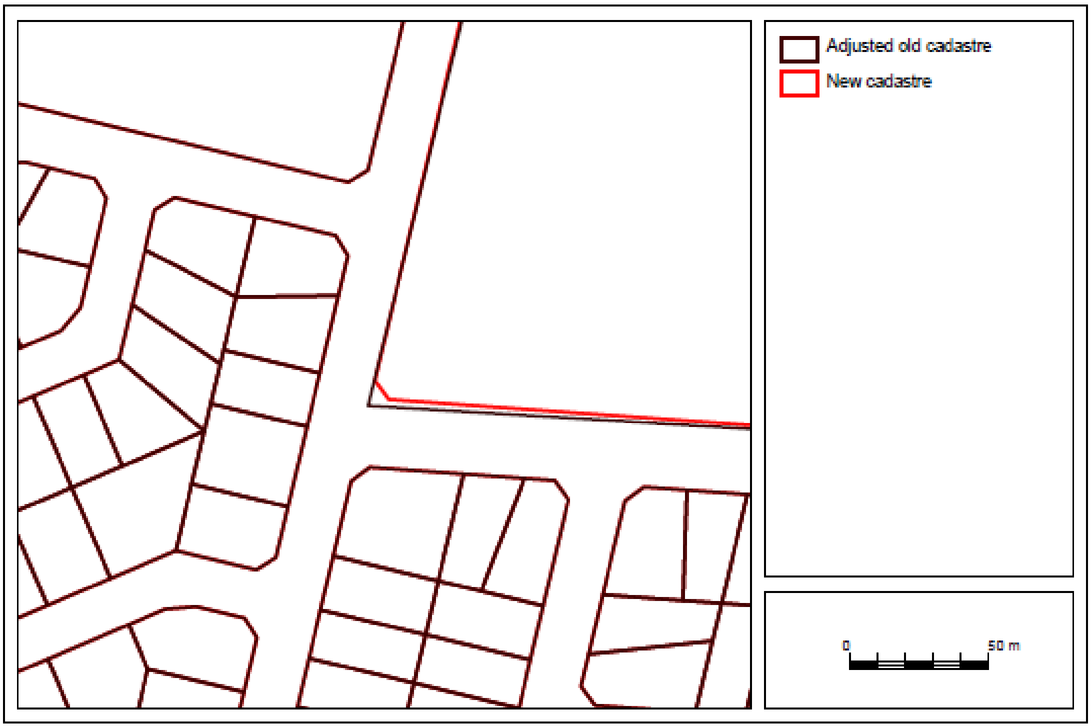

Figure 3.6 Viewing the results of an adjustment

3.3.2 Stage 1. Polygon (block and parcel) matching

The parcel-matching algorithm was initially undertaken using the datasets with available matching UIDs, i.e. where semantic matching was possible. Using the UIDs, it was possible to determine whether the algorithm had correctly matched a parcel or not. By drawing a thematic map using the results of the UID match check, it was possible to locate areas of real or apparent errors. Figure 3.7 illustrates an area from LGA11 where a few parcels are shown as “Wrongly matched” and one as “Should have been matched”, i.e. the parcels had matching UIDs but the software had failed to identify them as a match.

The map also shows the generated parcel centroid shift vectors drawn in blue. These indicate graphically which pairs of parcels have been matched to each other; the angle of these vectors also helps to locate areas where the parcel-matching algorithm had failed, for example, on the wrongly matched parcel (red) towards the south-east of the map. The two wrongly matched parcels in the northwest are, in fact, correctly matched as can be clearly seen from the centroid shift vectors; the parcel UIDs, in this case, were incorrect.

The results of the inspections were used to refine the parcel-matching algorithm, proceeding iteratively until a result satisfactory to the author was obtained.

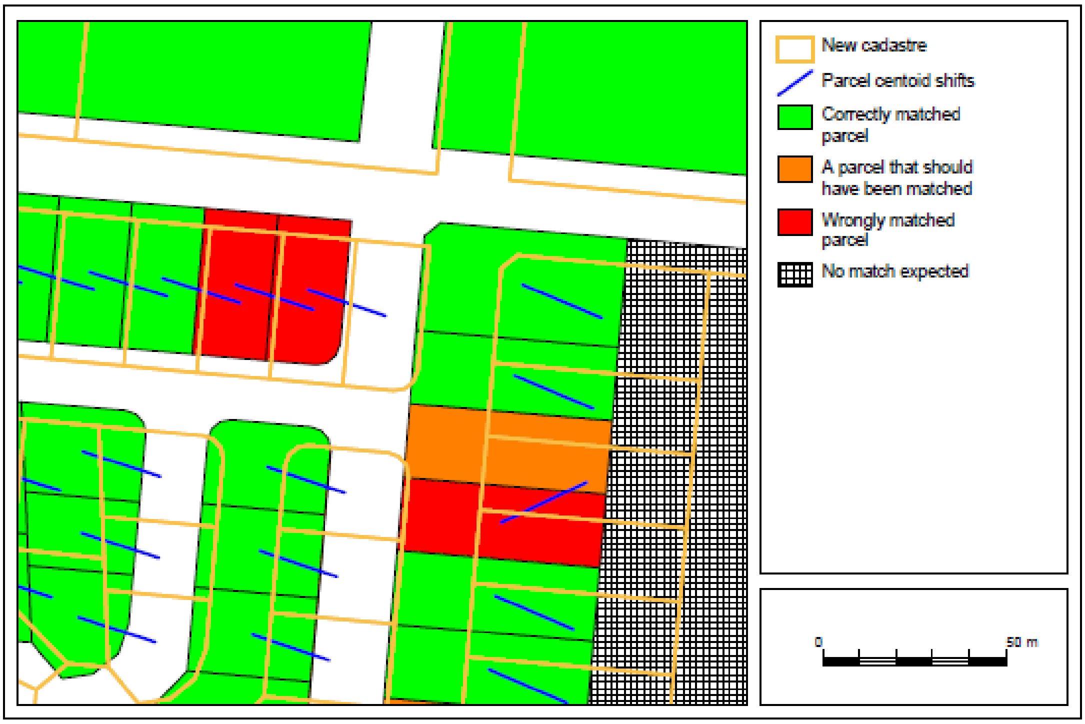

Figure 3.7 Parcels symbolised by match type

The parcel matching process is described in detail in Chapter 4. The chapter also discusses the reasons why the parcel matching process was preceded by a block matching process.

The creation of shift vectors for input to a rubber-sheeting tool must proceed from the points defining the old-cadastre polygons, i.e. the aim is to match each unique location point to a corresponding point in the new cadastre. After the parcels were matched, the vertices, therefore, were extracted into a point layer which was then processed to reduce the layer to just one point at each location. Many attributes were assigned to each point. The nature and purpose of each of these attributes and the ways in which they were used to guide the elimination of duplicate points at a location are described in Chapter 5.

3.3.3 Stage 2. Boundary matching

In the case of boundary matching, no objective process such as checking for UID matches was available. Therefore, the only method available in the preliminary stages of the research was to inspect the results using thematic mapping showing matched and un-matched boundaries in different colours and, later, by inspecting the adjustment results in the area. As with the parcel matching process, the matching algorithm was refined iteratively until a result satisfactory to the author was obtained.

The boundary-matching process is described in detail in Chapter 6.

3.3.4 Stages 3. Point matching

Once the first version of the point matching algorithms was implemented, it was possible to feed the shift vectors generated from the point matching process into the ArcGIS *RubbersheetFeatures* tool to adjust the old cadastre. There is no automated method, known to the author, for checking that control points have been matched correctly but, where the matches are correct, the adjusted old-cadastre parcel boundaries should exactly coincide with the new-cadastre boundaries (see Figure 3.6). Inspection of the adjusted old-cadastre layer overlaid on the new-cadastre layer in a GIS viewer was, therefore, the primary method used to discover adjustment failures. Several other methods to aid the location of potential point matching errors were developed during this research. They are summarised in Section 3.3.5.

In general, if inspection of the results in a GIS viewer indicated that a human operator could have created a correct link without difficulty, then it should, in principle, be possible to develop an algorithm to create the same correct link. The point matching algorithms were iteratively refined until the results were, in the opinion of the author, satisfactory or no significant improvement was achieved.

As the research progressed it became known which areas of each LGA map were likely to present difficulties. Therefore, when the research was sufficiently advanced to test the software on entire LGA datasets, it was possible to zoom in on these areas to ascertain whether the algorithms that had proved successful on the DA datasets were equally successful when applied to a complete LGA.

Once the point matching process was operational, at each iteration in the development of any of the matching algorithms, previous shift-vector and deleted-vector layers were retained so that the changed-algorithm results could be spatially compared with the results from the previous version. This enabled the discovery of locations where the new algorithm had delivered different results and provided a more objective way for deciding whether a change in the algorithm should be retained or not.

The point matching processes are described in Chapter 7.

3.3.5 Stage 4. Removing or locating erroneous shift vectors

The shift vector generation processes can result in touching and crossing shift vectors. In the ArcGIS solution, these vectors must be manually processed to remove those that are found to be incorrect. At this stage in the research, algorithms were developed to remove the erroneous vectors automatically. As with all the later stages of the research, the algorithms were iteratively refined until inspection of the old-cadastre adjustment appeared to be satisfactory. In addition to the automated removal of erroneous shift vectors wherever possible, techniques were developed to allow the author to rapidly locate areas of probable error, for example,

(a) Areas where intersecting or touching shift vectors had been deleted.

(a) Places where the areas of matched parcel no longer matched after adjustment.

(b) Locations of unmatched nodes or points.

(c) Locations of unexpectedly long shift vectors.

(d) Locations where the point match was not identical in both directions: old-to-new and new-to-old; it can be assumed that a lack of a reverse match suggests an incorrect point match in one direction or the other Siriba et al. (2013).

(e) Areas where block boundaries do not coincide after adjusting the old cadastre blocks.

Chapter 4 describes this research stage in detail.

3.3.6 Testing the complete solution

The processes described above were initially executed on the 33 small datasets covering difficult areas. Wherever poor adjustment results were detected, a decision was made regarding whether a human operator, tasked with manually identifying control points, would be able to correctly do so. Whenever it seemed to the author that an operator could easily identify a correct match, it was assumed that the software could also be refined to do so. An attempt would then be made to refine the algorithms to improve the results in that area. At each matching stage, it was often necessary to refine a previous stage to further improve results.

When satisfactory results on the small datasets were obtained, the complete LGA datasets were re-processed to ensure that the latest changes had not caused those results to deteriorate. Sometimes this process resulted in additional problems being detected and new difficult areas being located, in which case the entire process of algorithm refinement would be repeated. This iterative process was carried out more than 3000 times in a process of gradual convergence on an optimal solution.

It should be noted that the iterative refinement approach to developing the solution led to expending much effort on addressing pathological situations, for example, elimination of rarely occurring mismatches. However, given the potentially huge size of cadastral datasets, even pathological situations are likely to occur many times. Throughout the research, efforts were made to automatically avoid mismatches and thus minimise the possibility that the resulting shift vectors, when used with the ArcGIS *RubbersheetFeatures* tool, would result in incorrect topology in the adjusted layers. Efforts were also made to minimise the number of errors needing manual correction.

3.4 Summary

This chapter has described the stages and methods of research employed after the GA research was discontinued. The next chapter describes the parcel matching research in detail and the reasons why parcel matching research was initially undertaken as part of the GA research. It also describes the reasons for matching cadastral superblocks before matching the parcels. The matching algorithms used for matching the blocks and the parcels are described in detail.

4 BLOCK AND PARCEL MATCHING

The research described in this section was initially undertaken to automatically provide maxima for the search distance parameters (required by the Workbench software) when developing the genetic algorithm (see Section 1.5). This research resulted in the realisation that the information arising from the parcel matching could be used to guide a control point matching process and that, therefore, the Workbench software would no longer be required as a component of a complete spatial adjustment solution.

This chapter describes the reasons why the initial approach to the research was discontinued but how this initial approach resulted in some particularly important insights that affected the subsequent direction of the research. It goes on to detail the research undertaken into block and polygon matching and the rules developed for the matching algorithms resulting from that research.

4.1 Overview

Section 4.2 encompasses all the preliminary research stages executed for this thesis, including sections on: the importance of the genetic algorithm research (Section 4.2.1); the insights gained into problems specific to cadastral matching (Section 4.2.2); the research undertaken to discover a way of spatially limiting searches for matching parcels (Section 4.2.3); and the attempts to use machine learning techniques to formulate rules for matching parcels using their spatial characteristics (Section 4.2.4). Importantly, Section 4.2.3.3 describes how the decision was taken to use apparent block movement to constrain the search for matching parcels.

Section 4.3 provides a detailed description of the algorithms that were developed for block and parcel matching. It describes the way in which parcels were classified into different spatial types and how the blocks were then created from the classified parcels (Section 4.3.2) and how a preliminary parcel matching process was undertaken to establish the nature of the dataset – urban or rural (Section 4.3.3). Section 4.3.4 describes the way in which the information gained from the preliminary parcel matching process allowed identification of subdivisions and amalgamations and how these were then removed to improve the parcel match rate. Section 4.3.5 documents the different rules used to match urban or rural parcels.

Section 4.4 summarises the results obtained using the parcel matching algorithms documented in Section 4.3, and Section 4.5 discusses the benefits of the block matching process to the parcel matching results.

4.2 Preliminary research

This section describes the various research processes and investigations undertaken before the parcel matching process was developed.

4.2.1 Importance of the genetic algorithm research

The Workbench spatial adjustment program acquired for this research requires the operator to supply the five parameter values needed to constrain the search for control points. The parameters are: the maximum allowable distance between parcel centroids; the maximum allowable difference in length between parcel boundary lines; the maximum allowable difference in angle between boundary bearings; the maximum allowable distance between nodes; and another distance parameter used to cull intersections that are too close together for reliable matching.

At the start of this research, a Genetic Algorithm (GA) program was developed to automatically optimise the five parameter values. However, four of the five values are used to constrain search distances and, to make sure that the GA would not explore dangerously high values for these parameters, it was found to be necessary to provide upper limits for the GA to explore; values that are too high can result in large numbers of incorrect control point matches (Merritt, 2009). Because the GA was programmed to attempt to maximise the number of cadastral points matched, it was important to determine suitable maxima for the parameter values.

In addition to this requirement, it was known that the Workbench software operated more efficiently if parcels had matching UIDs but UIDs were not available for eight of the twelve LGA datasets (see Table 3.1). The parcel matching research was therefore conducted to assign software generated matching UIDs to the parcels, and to determine maxima for the distance parameter values.

As a result of this research, it was realised that no one set of maxima would be suitable for every area of an LGA; inspection of the cadastral datasets using a GIS viewer had already revealed that the apparent movement of parcel boundaries tended to be very much greater in rural areas than in urban areas (see Section 3.2.1) and the parcel matching research confirmed that. Research, therefore, proceeded with an attempt to overcome this problem by separating each dataset into large, medium and small parcel-size groups to be processed separately by the GA; this attempted solution resulted in yet more problems (described in detail in Appendix D). However, the parcel matching, initially undertaken only to improve the efficiency of the Workbench software, gave rise to the crucial insight that the parcel matching information would facilitate point matching without the use of the GA and without the need for user supplied search-distance parameters. Once two parcels are correctly matched, by constructing a vector connecting their centroids, computation of the bearing and distance of the apparent parcel movement becomes possible. It was realised that, for matched parcels, the apparent movement metrics could be used to predict the expected location – a target point – for all a parcel's vertices in the new cadastre.

Figure 4.1 illustrates the computed target locations (black triangles) for old-cadastre points (black dots) on matched parcels from LGA11. The location of each target point has been arrived at by applying a linear transformation, derived from the apparent movement between the matched parcel centroids, to each parcel vertex. The map shows how close these target points can be to the correct new-cadastre point, especially for small urban parcels; in this area, the parcels have all moved approximately 14 metres in a westerly direction. The conclusion drawn from these results was that the point matching process could best be constrained by using search criteria determined from the apparent movement of the parcel and that the constraints would be different for each parcel. All further research was based on these assumptions.

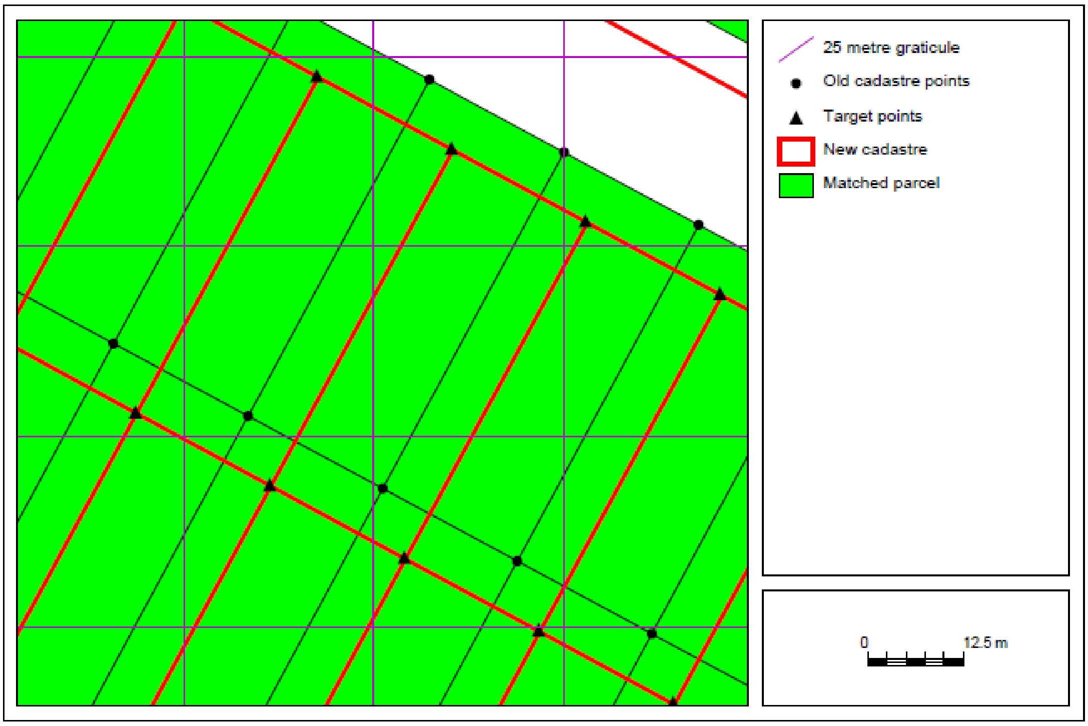

Figure 4.1 Expected-location points for new-cadastre points on correctly matched parcels

Most of the published research on the map conflation problem, referenced in Section 2.3, has been based around a planar graph solution. It is possible, if the GA research had not been undertaken, that the parcel matching research would not have been commenced and that the value of using the apparent distance and direction of parcel movement to locate point matches would never have been realised.

To summarise, the GA research formed no part of the eventual solution but resulted in the significant insight into the value of parcel matching for identifying control points. Details of the GA research are to be found in Appendix D.

4.2.2 Problems arising from the nature of cadastral data

For the reasons outlined in Section 4.2, matching parcels became an early step in this research project. Some of the problems associated with parcel matching are outlined in this section.

Where two cadastral datasets have matching Unique Identifier attributes (UIDs), a parcel matching process could easily be automated using the UIDs alone without the use of spatial matching techniques. However, even where UIDs exist, there may be pairs of parcels that do not have corresponding UIDs where it would be correct to match them. In addition, UIDs can sometimes be in error.

The parcel matching problem becomes more difficult when there are no matching UIDs, as was the case for eight of the twelve available cadastral datasets. In this case, a spatial method is essential. For the purposes of this research, spatial parcel matching was conducted for all the datasets and the UIDS were only employed to test the accuracy of the match results.

The following sections will outline several general problems that can cause difficulties for the parcel matching process and the specific problems that must be solved for the distinct types of parcel (urban or rural).

4.2.2.1 Poor data quality

Cadastral datasets, because of the enormous volume of data involved, are seldom one hundred percent accurate. Problems that can make the parcel matching process more difficult are listed here:

4.2.2.1.1 Polygonised roads

Within a single dataset, some roads may be represented as polygons and some as voids within the dataset. Figure 4.2 shows an area from LGA07 where several roads are modelled as polygons in the new cadastre but not in the old. (The blue colour shading of the new cadastre is revealed only where there is no overlapping old-cadastral parcel.) The single new-cadastre block left of centre covers two separate blocks (contiguous parcels surrounded by roads, rivers, or coastlines) in the old cadastre; the road void in the old cadastre is now represented by a closed polygon in the new cadastre.

Figure 4.2 An area where several roads are modelled as polygons in the new cadastre but not in the old.

4.2.2.1.2 Incorrect UIDs

Even when the parcel attributes include matching UIDs that could be used to match the parcels, these may not always be correct, sometimes implying an incorrect match between two parcels and sometimes failing to indicate a match between two parcels that should be matched. For example, the UID of a parcel may have been changed between versions, even though the parcel itself has not changed.

A component of the parcel matching algorithm, described in Section 4.3.3, was a UID test implemented to identify errors of this type. Figure 4.3 shows an example from LGA08 where the UID values have been exchanged between versions; the parcels highlighted in red have been correctly matched by the parcel matching algorithm but the UID test flagged them as "wrongly matched" because the UIDs did not match, thus enabling the discovery of the error.

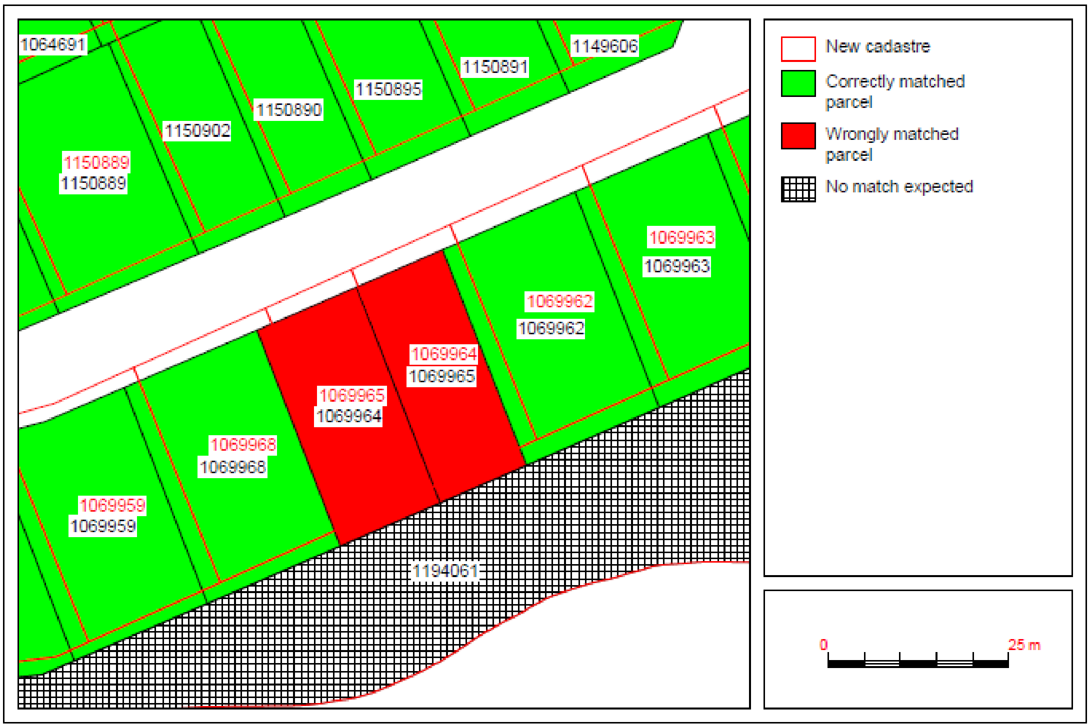

Figure 4.3 An area where two adjacent parcels have exchanged their unique identifiers in the interval between versions.

4.2.2.1.3 Slivers

There may be slivers in one or other of the datasets. Figure 4.4 shows a sliver from LGA11 (indicated only by the thickness of the boundary line adjacent to the shaded region); the sliver has an area of less than one square meter along the road frontage of the old-cadastre parcel. The sliver does not appear in the new cadastre.

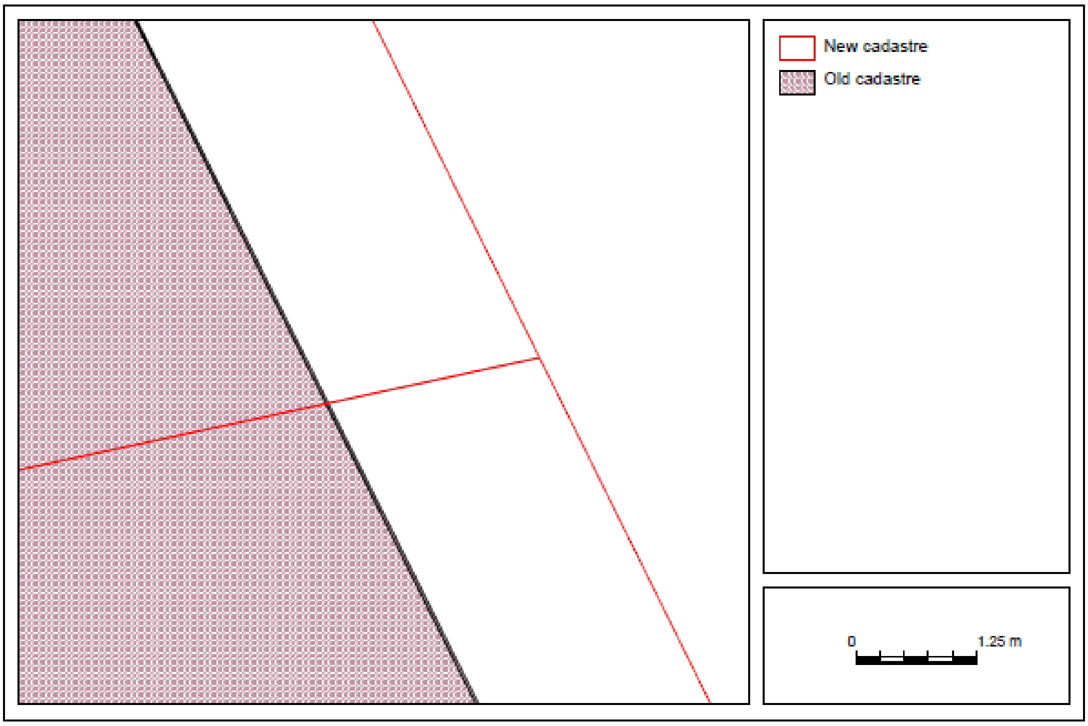

Figure 4.4 A sliver along a parcel road frontage

4.2.2.1.4 Incorrect topology

The topology may be incorrect, for example, duplicated polygons, overlapping polygons or non-contiguous parcel boundaries; the latter can often occur in data modelled as independent polygons as in the shapefile format. Figure 4.5 shows an area from LGA07 where there is an overlapping parcel in the new-cadastre layer (dark grey in this image) and some of the parcel boundaries are not contiguous although the latter is only visible at a large scale.

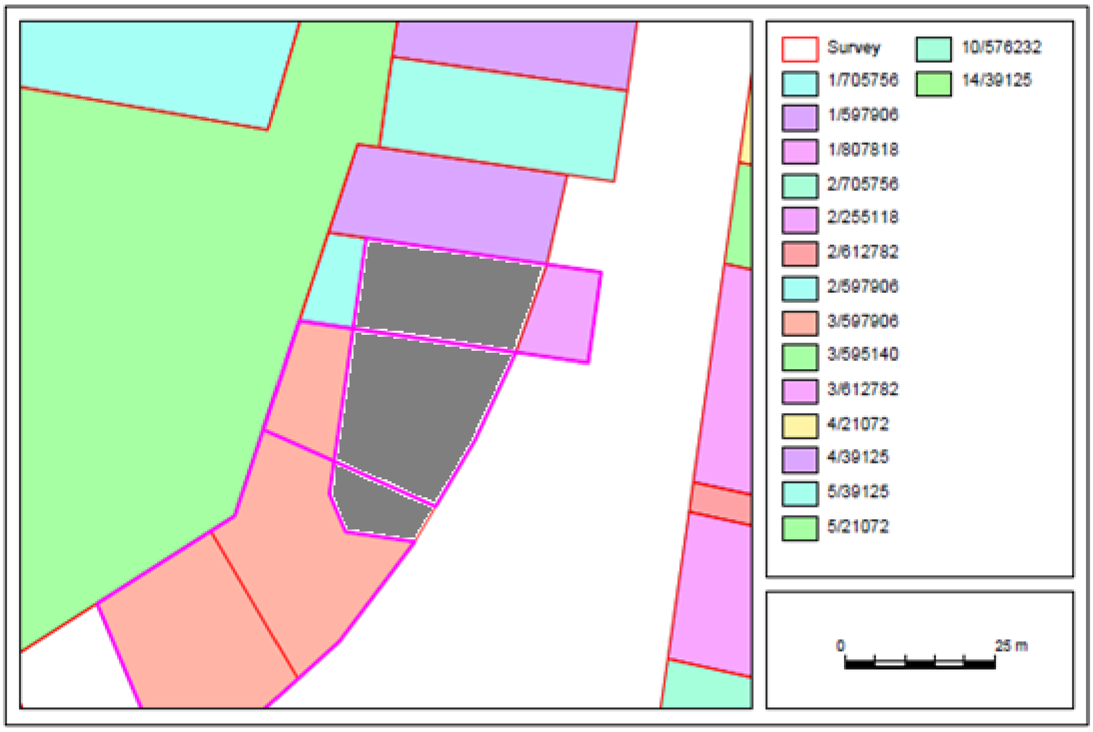

Figure 4.5 There are overlapping parcels in the grey area

4.2.2.2 Subdivisions and amalgamations

In the interval between two cadastral versions:

(a) Parcels may have been subdivided, for example, in an area where a rural property has been developed for residential purposes. For an example see Figure 4.24.

(b) Boundaries may have appeared, disappeared, or moved, for example, where an area of one property has been transferred to a neighbouring property. Figure 4.6 from LGA07 shows an area where complex boundary changes have occurred between versions. The four grey shaded parcels at the centre (labelled 297, 298, 370 and 372) have been merged and then split into five quite differently shaped parcels (outlined in blue).

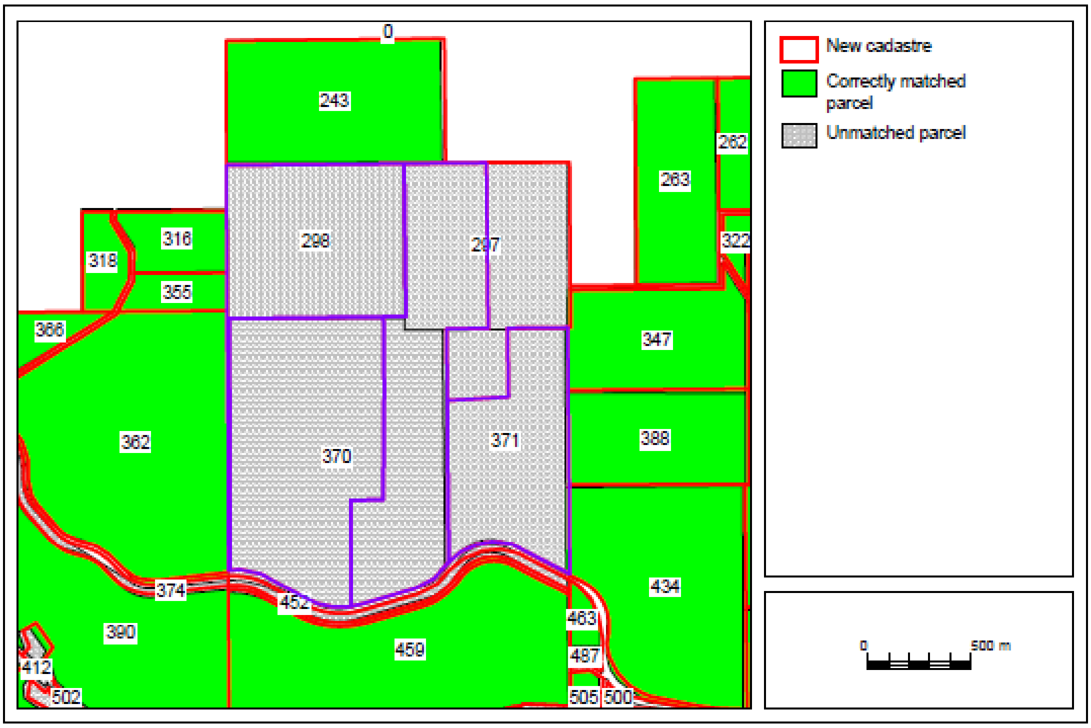

Figure 4.6 An area showing complex boundary changes

(c) Parcels may have been amalgamated to form a larger property. This appears to have happened in the area shown in Figure 4.6.

(d) The effect of subdivisions and amalgamations between versions is that unmatchable points occur in one cadastre or the other with the potential to give rise to incorrect shift vectors.

4.2.2.3 Problems specific to urban areas

In urban areas, where parcels tend to be small, there may be rows of identically shaped parcels all having a similar area. If the apparent shift between the old and the new cadastre is larger than the width (street frontage) of the parcel, it can be difficult for an automated solution to match up the correct parcels when there are no matching UIDs.

If the apparent shift is very large it is even possible that the parcels that should be matched have no area of overlap to aid in the matching process. Figure 4.7 from LGA11 shows just such an area.

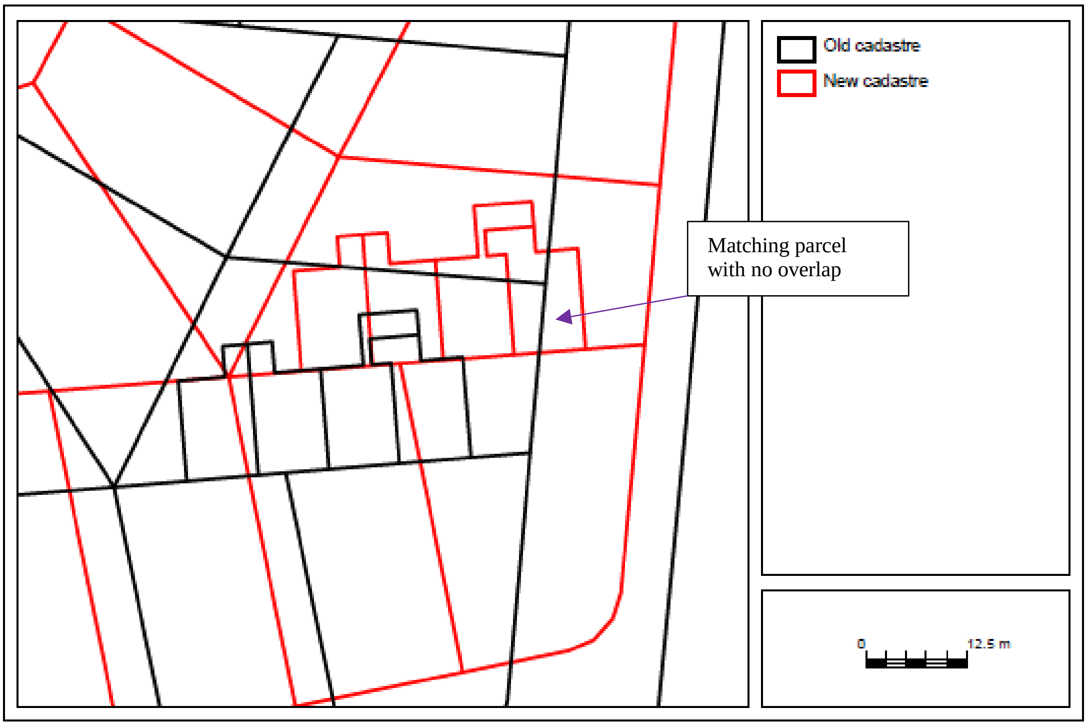

Figure 4.7 An area where several small matching parcels have no area of overlap

4.2.2.4 Problems specific to rural areas

A rural area dataset can exhibit all the above problems, but the very inhomogeneous nature of the rural data also gives rise to others, for example, the banks of creeks, streams, rivers, and coastlines (collectively known as riparian boundaries) may have changed significantly between versions and different versions may or may not include sandbanks and small islands. Figure 4.9 shows a section of creek exhibiting, in places, more than 100 metres movement between the two versions.

Matching points on riparian boundaries may sometimes be unimportant for the adjustment of spatially dependent datasets; it is unlikely that highways asset data, such as street lights or signposts, for example, would fall along creek lines or other riparian boundaries, although planning zones and other administrative boundaries may do so. Nevertheless, if point matching is poor in these areas, bad shift vectors can result in poor adjustment of the surrounding parcels necessitating additional GIS operator effort to correct.

Parcel boundaries may, or may not, follow creek boundaries. In the area shown in Figure 4.8 from a rural area in LGA07, the parcel boundaries do not follow the creek boundaries; the creek banks have been modelled as separate parcels. However, in the case of large rural parcels, parcel boundaries typically do follow the banks. Figure 4.9

from LGA11 shows an area where the parcel boundaries are coincident with the creek boundaries.

Figure 4.8 An area that includes a section of creek where the parcel boundaries do not follow the creek boundaries

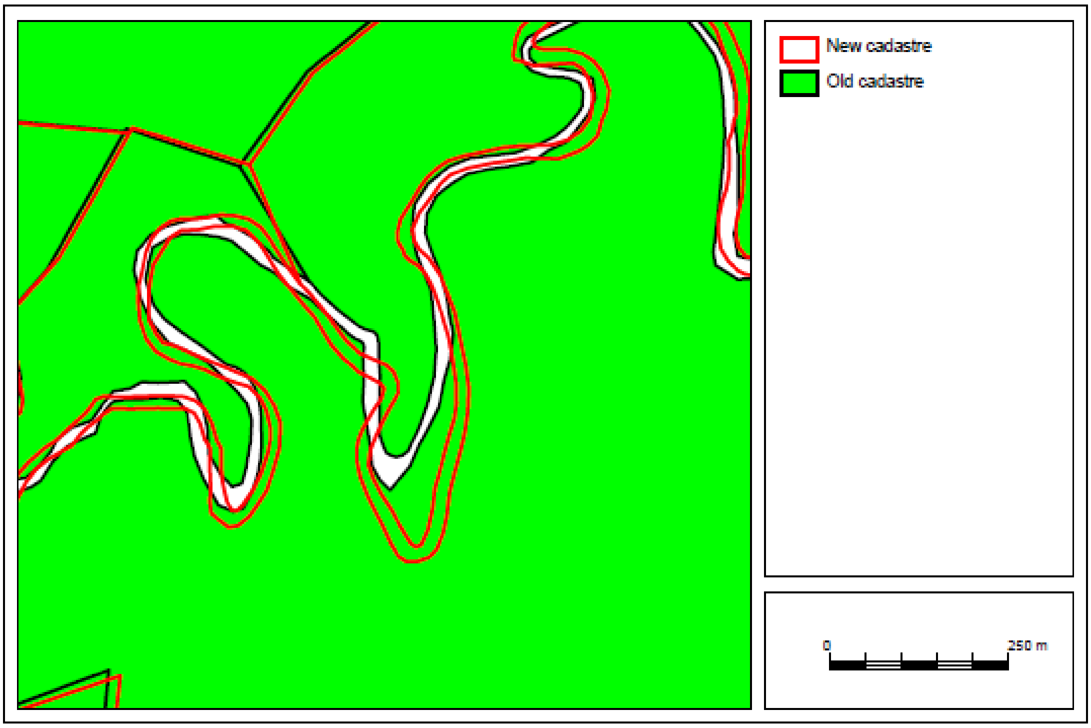

Figure 4.9 Large movement of creek boundaries that also form parcel boundaries

In pastoral areas with long, complex boundaries there is seldom a one-to-one correspondence between points in the two cadastres. There may also be greater point density in one or other of the datasets as illustrated in Figure 4.10 from LGA11 where the riparian boundary in the new cadastre has far more points (red dots) than the corresponding boundary in the old cadastre (black dots). These differences can result in differences in perimeter lengths.

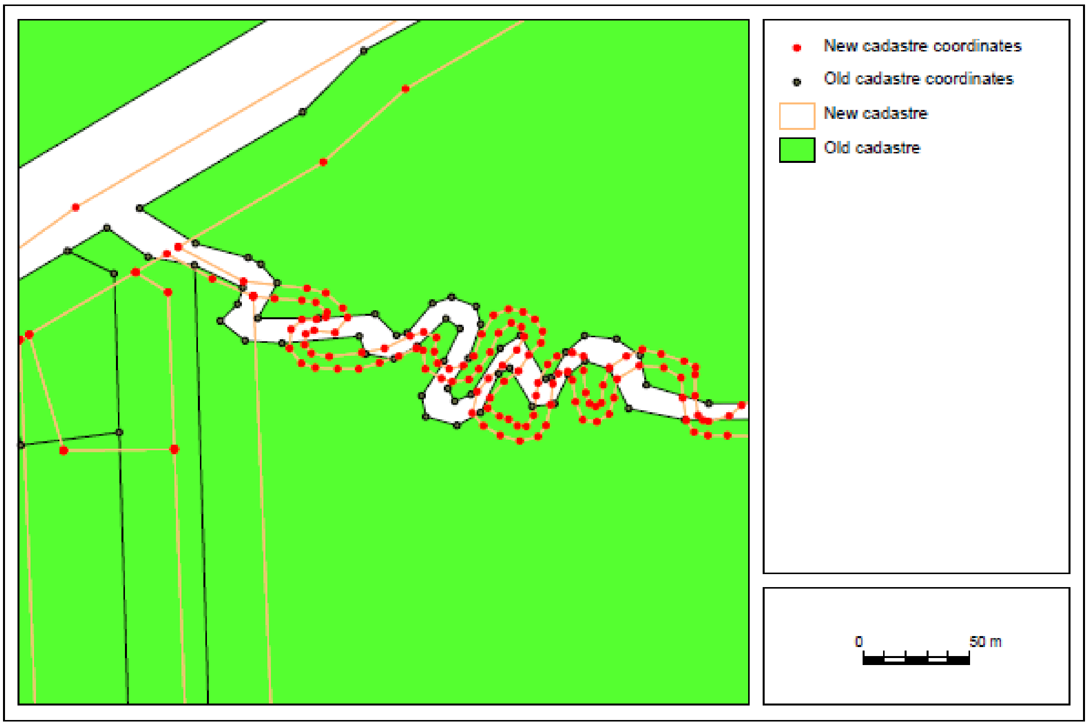

Figure 4.10 An area where the points are denser in the new cadastre than in the old

4.2.3 Determining search distances for polygon matching

To undertake an automated parcel matching process between two polygon layers, prior information about search distance and direction would ideally be available, i.e. "where should the software look for a match". A polygon matching algorithm would need to search for parcels of similar shape and size but in urban areas there are often long rows of almost identical rectangular parcels. An unconstrained search could easily result in many incorrect matches. Figure 4.30, for example, shows mismatches of this type that occurred in the early stages of the research; the blue vectors connect the centroids of matched old and new cadastral parcels. Inspection of the resulting parcel centroid-shift vectors in a GIS viewer clearly shows that most of the matches outlined in the centre of the map are incorrect.

In the same way that existing solutions to the spatial adjustment problem require a search distance to constrain the search for matching control points, it is also desirable to apply a similar constraint to the search for matching parcels. However, as the aim of this research was to automate the entire spatial adjustment process as far as possible, it was necessary to conduct research into ways to set the search distance for parcel matches automatically, and, if possible, a search direction as well. Several ideas for establishing these values were explored and are described below.

4.2.3.1 Using the extent of the map

Each cadastral dataset theoretically covered a complete LGA. Hence, consideration was given to using the difference between the centroids of the extent (bounding box) of each layer (old and new) to determine a suitable search direction and maximum distance for the parcel matching process. However, inspection of the data immediately showed that this approach was not viable as the extents of the available datasets were not always exactly coincident.

Figure 4.11 shows the north-west corner of LGA07, selected because it illustrates the problem. The orange shaded parcels are those present in the old cadastre and the yellow parcels are those from the new cadastre. The image clearly shows that the northern boundary of the bounding boxes (extents) of the two datasets are not coincident meaning that any apparent movement between the centroids of those boxes would be of no value in deciding values for search distance or search direction constraints.

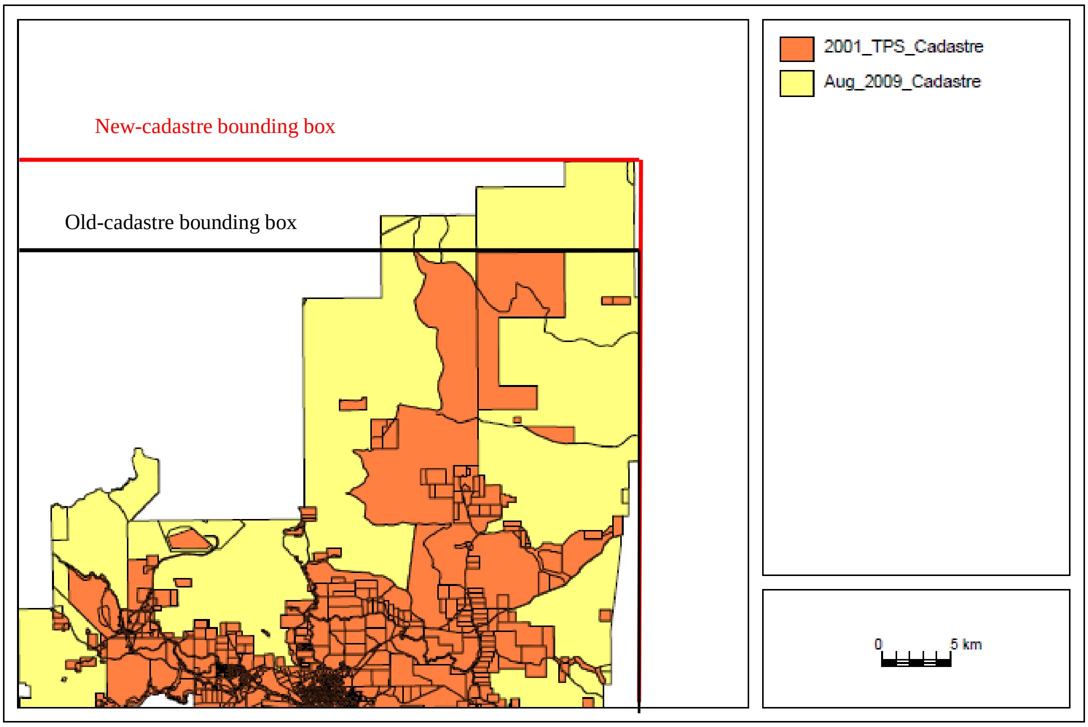

Figure 4.11 Different areas covered by the old and new cadastre in an LGA

Although the easterly boundaries of the bounding boxes are coincident, the northerly boundary of the bounding box of the new cadastre is more than seven kilometres north of the northerly extent of the old cadastre.

4.2.3.2 Using an average of apparent movements of overlapping parcels

Vectors were constructed between the centroids of every overlapping pair of parcels by processing the centroids from the old cadastral layer and locating a new-cadastre parcel by point-in-polygon search (the simplest possible form of matching). The mean length and direction of these vectors were calculated with the intention of using these values to arrive at upper limits for the direction and search distance constraints. Inspection of the results showed that mean apparent movement direction would not be useful because the apparent shift can be different in different areas of the dataset. Figure 4.12 from LGA01 shows apparent movements in different areas of the map that are almost orthogonal to each other.

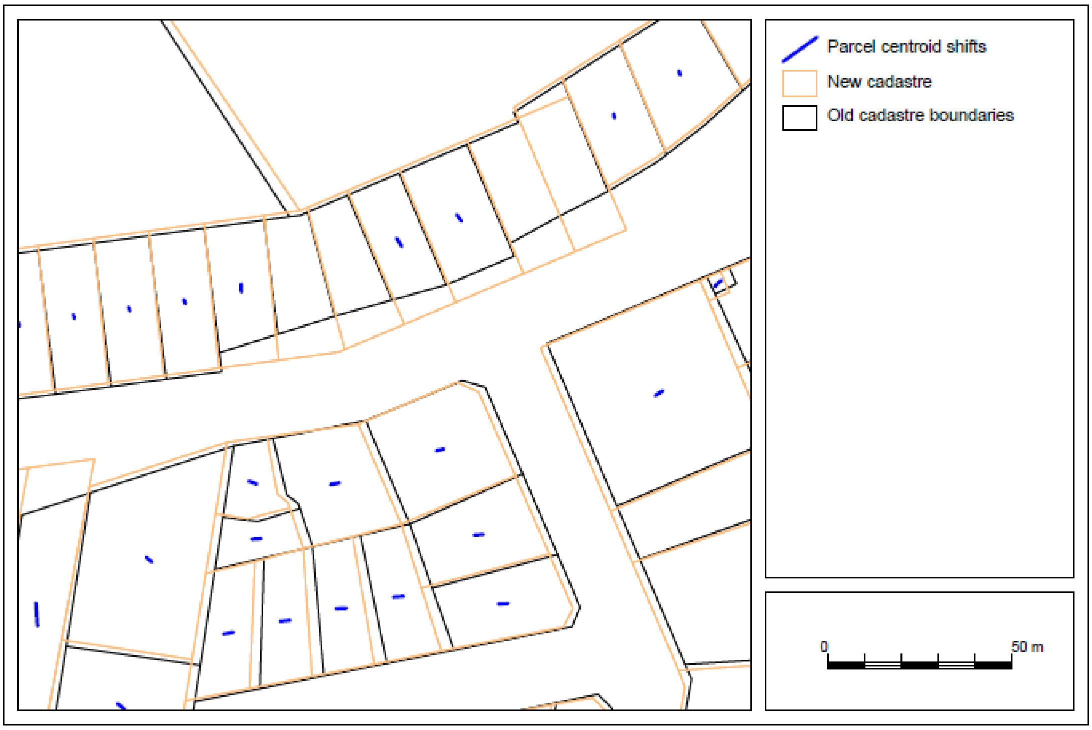

Figure 4.12 An area where parcels have apparently moved in different directions

4.2.3.3 Using apparent block movement

Using a GIS viewer, inspection of the available datasets led to the hypothesis that, if blocks of contiguous parcels contained within a road void (also known as cadastral superblocks) could be matched between the cadastral layers, the apparent movement of the blocks could be used to constrain the search directions and distances for the parcels within that block. Figure 4.13 fromLGA11 shows blocks only and is an example of an area where the blocks appear to have moved in different directions in the east and west of the area; the apparent movement is up to 40 metres in some cases. This can occur because of low positional accuracy in the original digitising, whether the source was town maps, parish maps or low-resolution orthophotography, and the small scale of those sources. The observation of these large apparent movements was fundamental to the realisation that apparent block movements would be important for the achievement of correct parcel matching, particularly in urban areas where many identical parcels may be adjacent to each other, or there may be small or zero overlap between matching parcels.

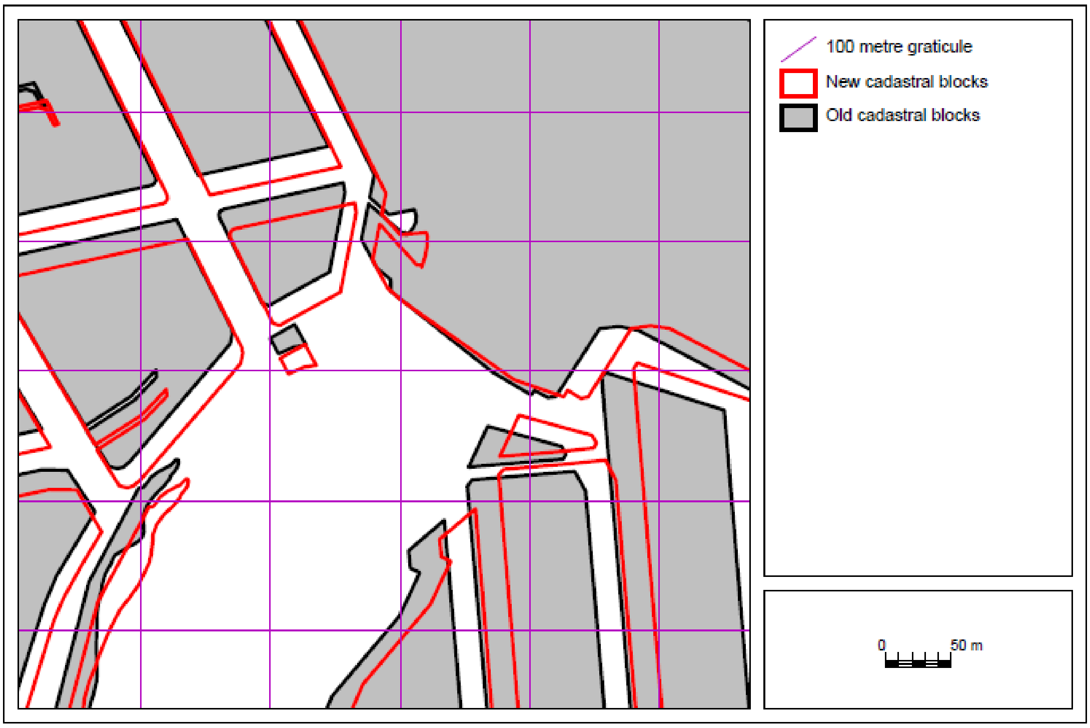

Figure 4.13 Apparent block movements

Ascertaining the apparent movement of the blocks was the method finally adopted for setting direction and distance constraints for the parcel matching process but, before arriving at this solution, consideration was given to creating blocks by separating contiguous urban parcels from contiguous rural parcels. It soon became apparent that this process, in many cases, did not deliver matchable blocks because the number of urban parcels in a superblock may have changed in the interval between the release of the two cadastres thus changing the size of the contiguous urban parcel block and the contiguous rural parcel block. Figure 4.14 from LGA11 shows an area where additional urban parcels have been created within a rural block.

After this approach was rejected, all further research made use of the apparent superblock movement. Roads tend to change much less between cadastral versions suggesting that attempts to match these blocks between two cadastres would be more likely to be successful. Once this decision was made, the advantages of the method became clear.

(a) A high degree of block matching can be achieved in areas where the road network has not significantly changed because a large area of overlap between matching blocks can be expected in most cases.

(b) Corresponding parcels within a block are likely to have apparently moved in the same direction and over the same distance as the apparent block movement.

(c) Incorrectly matching a parcel in the old cadastre to one in a different block in the new cadastre can be avoided where the old block has been matched.

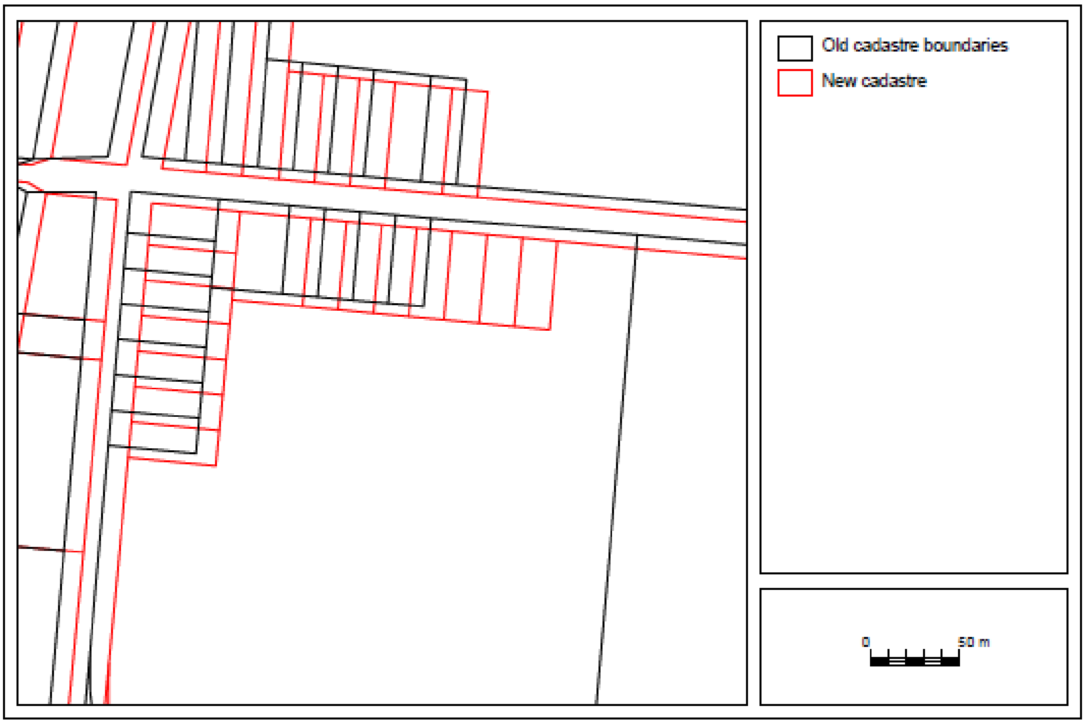

Figure 4.14 An area where additional urban sized parcels have been added within a rural block

4.2.4 The machine learning attempt to formulate parcel matching rules

For the spatial characteristics approach to polygon matching, as suggested by Huang et al. (2010), several different shape attributes were investigated for their ability, when compared to the attributes of a potentially matching polygon, to identify the best match. The four datasets with UIDs, comprising more than 53,000 parcels, were used to assess the quality of the results. The shape attributes investigated were:

(a) Polygon area differences.

(b) Polygon perimeter differences.

(c) Diagonal bearing differences using the bounding rectangle diagonals.

(d) Distance between polygon centroids.

(e) Differences between the ratios of polygon area to minimum-bounding rectangle area.

(f) Area of overlap of the current and candidate polygon.

When matching the parcels (rather than the blocks), in addition to those shape attributes suggested by Huang et al. (2010), it was possible to include further discriminatory match parameters, i.e. the information arising from the block shift process:

(g) The difference between the length of a vector joining the corresponding centroids of the candidate and the current parcels (centroid shift) and the length of the current block centroid shift.

(h) The difference between the azimuth of the centroid shift and the azimuth of the current block centroid shift.

Given the number of potentially discriminatory variables and the availability of the four datasets where the correct result was known, it seemed possible that a machine learning algorithm, using some of these datasets as training material, could discover rules for the use of the spatial-characteristics to determine the correctness of any potential match. The freely-available WEKA machine learning software (Witten, Frank, Hall, & Pal, 2017) was used for this purpose.

Two experiments using pairs of cadastral layers were conducted using the WEKA software. The cadastral pairs were the old and new cadastre from the LGA01 dataset and the old and new cadastre from the LGA11 dataset. These datasets were used to create the training data. The datasets both have existing matching UID attributes so that the correctness of a match can be validated. The smaller dataset, LGA01, consists of mostly urban sized parcels, 655 parcels in all. The larger dataset (LGA11) has more than 12,000 parcels consisting of mostly rural parcels with one medium sized town and several smaller ones. The two datasets do not overlap.

To create the data for the machine learning algorithm it was necessary to identify all potentially matching parcel pairs and output the values for all the discriminatory variables together with a Boolean variable to indicate whether the parcel pair were a correct match or not. A potential matching parcel in the new cadastre was identified as any parcel that overlaps a buffer created using the apparent block-movement distance around the current old-cadastre parcel. For each parcel in the old cadastre all the overlapping parcels were processed and the potentially discriminatory attributes were output for each pair. The WEKA classification tool J48, a machine learning algorithm, was used to create a decision tree from the training data; J48 is a JavaScript version of the C4.5 algorithm developed by Quinlan (1993).

The attributes used in the training data are listed below. The variable names in brackets below are those that appear in the WEKA results in Appendix A.

(a) The difference between the block shift azimuth and the azimuth of the line joining the centroids of the potentially matching parcel pair (DiffAngle).

(b) The difference between the block shift vector length and the length of the line joining the centroids of the potentially matching parcel pair (DiffLength).

(c) The difference between the areas of the two potentially matching parcels as a percentage of the area of the larger parcel (AreaRatio).

(d) The difference between the perimeters of the two potentially matching parcels as a percentage of the perimeter of the longer perimeter of the two parcels (PerimRatio).

(e) The overlap area as a percentage of the larger of the two parcels (OverlapRatio).

(f) The absolute difference between the circularity indices (see Section 4.3.2.3) of the two parcels (Similarity).

(g) The angular difference between the diagonal of the minimum containing rectangles of the two parcels (DirectionDiff).

(h) A Boolean variable indicating whether two potentially matching parcels also shared matching UIDs (UIDsMatch), i.e. the variable indicating whether a match was true or false.

For each test dataset, a record was output for each potentially matching pair of parcels with the Boolean variable indicating whether the match was correct according to the UID values. A J48 decision tree was produced from this output.

The smaller dataset, LGA01, yielded a simple tree indicating that OverlapRatio was the only significant discriminator (see Appendix A.1). This suggested that, for urban datasets like LGA01, it could be assumed that using the old-cadastre centroid for a point-in-polygon search on the new cadastre and then checking for a large overlap ratio would give a correct result in most cases, especially where the parcels are regularly shaped rectangles as most urban parcels are. Figure 4.15 shows an urban area from LGA07 where a centroid search and overlap area check would correctly identify all matching parcels.

It has been observed that apparent movement in these areas are typically small, at least in the urban datasets available for this thesis. This may be because urban areas have been initially surveyed to a high degree of accuracy. The point-in-polygon search with overlap-ratio check was the method finally selected for matching parcel pairs in datasets that were deemed to be urban only. The algorithm will be described in detail in Section 4.3.5.

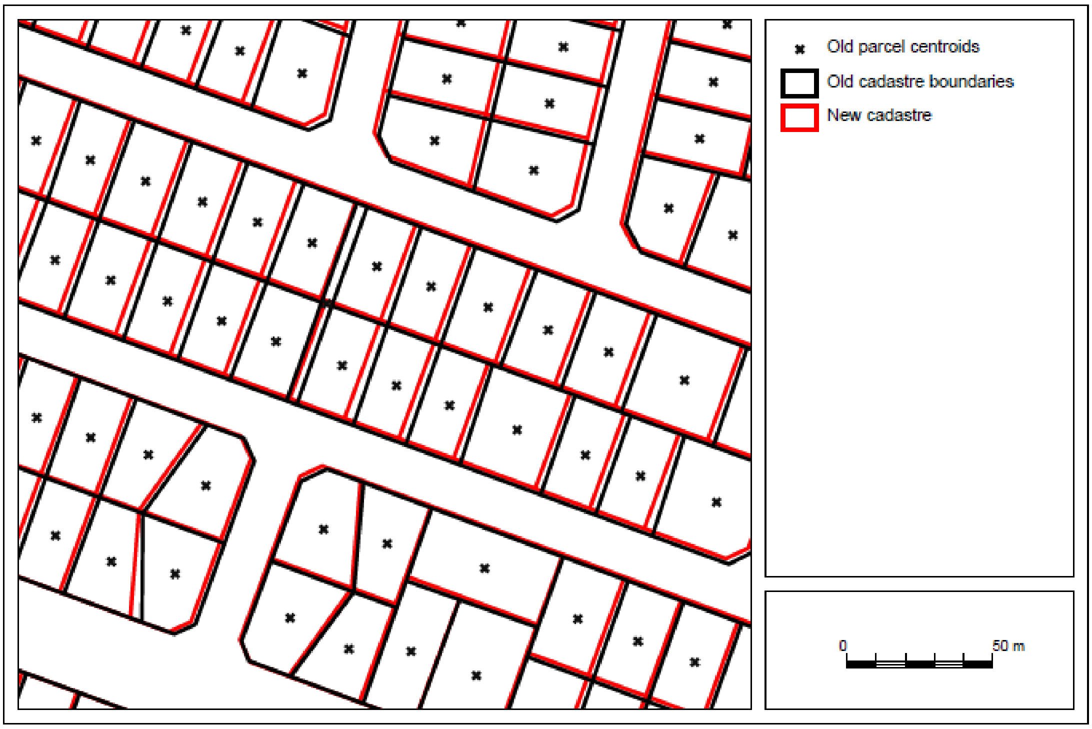

Figure 4.15 Urban parcels with a high degree of overlap

Examination of the extremely complex decision tree resulting from the rural dataset, (LGA11), suggested that the tree was highly specific to that dataset and that the results were unlikely to be applicable to any other (see Appendix A.2).

In machine learning, Occam's razor prevails and simple solutions are more valid and general than complex ones. A complex decision tree means that the feature space is partitioned into many regions with each class-split represented by many regions. Ideally, one region per class is preferred.

When the WEKA experiment failed to find a simple rule set for matching parcels in rural areas, the training data from LGA11 was exported to Microsoft Excel. It was hoped that Excel's multiple regression functions could be used to develop a regression equation that could then be encoded in the parcel matching algorithm. This method uses a combination of values to compute the results, rather than the orthogonal decisions used to create a WEKA decision tree. However, it was found that, although the coefficients resulting from the regression algorithm would result in many correct matches for the specific dataset used to develop the equation, applying the same equation to other datasets resulted in unsatisfactory results, i.e. the equation did not generalise.

The failure of these approaches is unsurprising. From inspection of the cadastral datasets, it is apparent that, for example, area and perimeter comparisons are not likely to be discriminatory in urban areas with small highly similar rectangular blocks, but that in rural areas, these comparisons are more likely to be of value. The option of employing heuristic techniques to arrive at a solution was therefore adopted, i.e. several different discriminatory expressions combining the values of potential discriminators were tested and refined until an acceptable solution was arrived at. These techniques were used to develop a match score (MS) based on some of the above-listed criteria. The polygon with the lowest match score was selected as the most likely match. Table 4.4 shows the results obtained by using the resulting algorithm to match the parcels from datasets having UIDs.

Different algorithms were developed for matching blocks as opposed to matching parcels. This was because, when matching blocks, there is no prior knowledge about search direction and distance but most blocks have a large area of overlap between the two datasets (see Figure 4.13) that can be used in addition to other spatial attributes to guide the matching process. Correct results for block matching were achieved using a similar algorithm to that developed for the parcels, but without the apparent block movement, of course. The algorithm used is described in detail in Section 4.3.

When matching parcels rather than blocks, if the block in which a parcel lies has been matched and, therefore, has a known apparent distance and angle of movement, these values can be used to guide the parcel matching process in addition to other spatial attributes of the parcel. In testing the different expressions used to identify matched parcel pairs, extensive use was made of the four LGA datasets with existing matched UIDs. For the block matching, and parcel matching on datasets with no UIDs, inspection of the thematically mapped results in a GIS view was used to check the validity of the matches. In each case, the algorithms were iteratively modified until results satisfactory to the author, or validated by matching UIDs where possible, were obtained.

Details of the algorithms finally developed for matching first blocks and then parcels are given in the following sections.

4.3 Polygon matching research

This section documents details of the algorithms finally adopted for the parcel matching process.

4.3.1 Outline of the parcel matching process

Given that this research was specifically designed to address the real-world problem of adjusting cadastrally dependent datasets, it seemed appropriate to make use of the unique spatial aspect of cadastral layers, i.e. the fact that roads, rivers and coastlines can subdivide the data into discrete blocks and that matching the blocks would be a simple process and result in information that could guide subsequent parcel matching (see Section 4.2.3.3).

Figure 4.16 outlines the processes used in the block and parcel matching stages of the solution (the numbers in parentheses refer to the section describing the process).

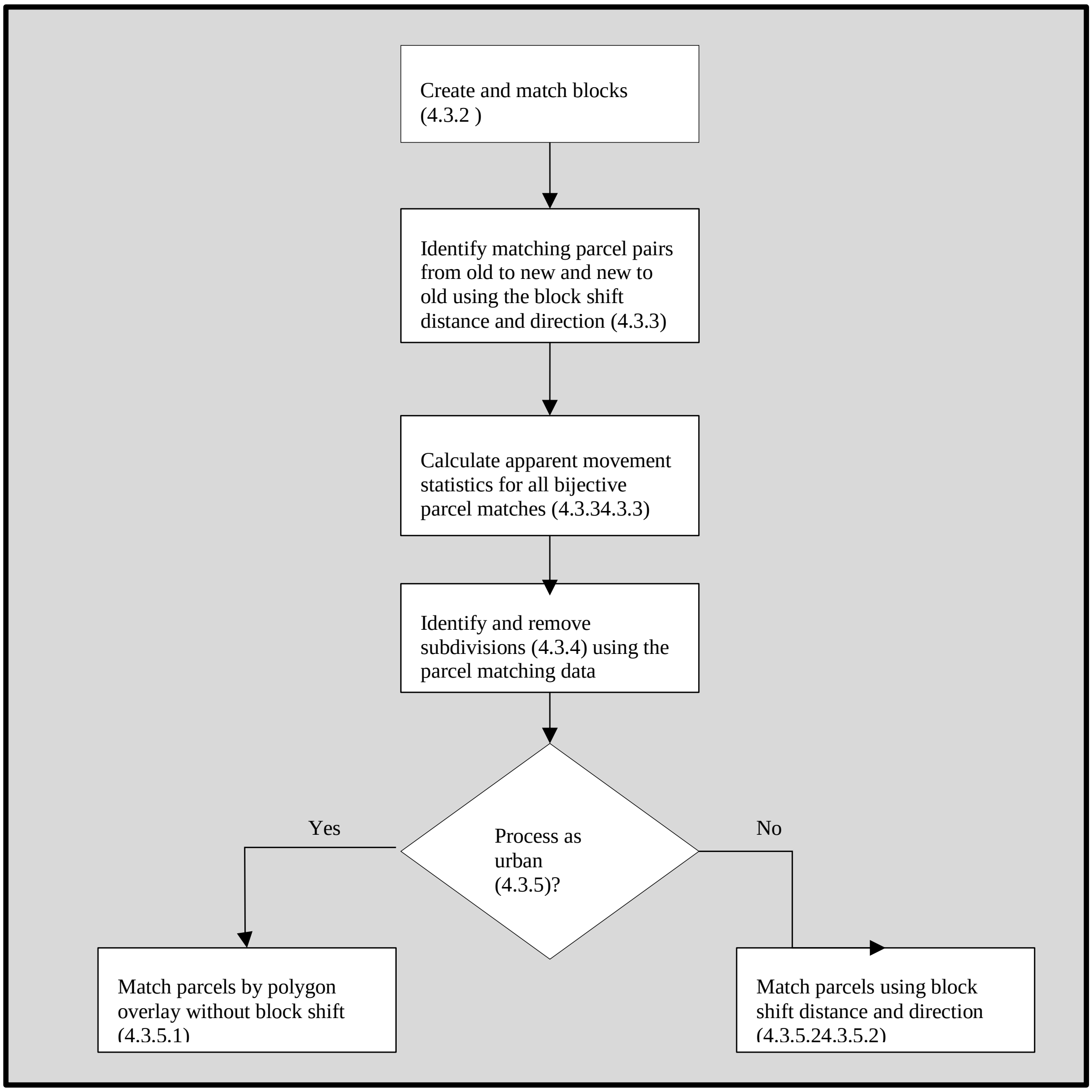

Figure 4.16 Block and parcel matching process outline

4.3.2 Extracting and matching blocks

Matching blocks in a clean urban dataset where all roads are correctly modelled as voids is very straightforward, a process of finding an overlap between the blocks from the two layers and checking that the shapes and areas of the blocks match within a reasonable tolerance is sufficient. However, datasets are seldom completely clean and, in several of the LGA datasets, there was no consistency between the two layers as to how roads were modelled, i.e. a road can appear as a void in one dataset and a closed polygon in the other (see Figure 4.2 which shows road polygons in the LGA07 new cadastre where there are none in the old). This can lead to two separate blocks in one dataset appearing as a single block in the other resulting in matching failure.

Because the intention of matching blocks is to improve the subsequent parcel matching process by determining a search distance and direction to guide that process, it was deemed important to match blocks wherever possible. However, it became clear during the development of the matching algorithm that the simple overlap method sometimes resulted in a match between blocks with slightly different shapes. In these cases, the apparent movement of the block centroid did not accurately indicate the block movement. This was particularly true for large rural blocks where the apparent movement of the centroid may not be at all representative of the apparent movement of the individual parcels within the block. Figure 4.17 shows an old and a new irregular block where the centroid shift does not accurately indicate the apparent shift of the block. The inset shows that the western ends of the two blocks are not coincident; the western ends are separated by approximately half a kilometre.

Polygons that enclose very highly irregular areas such as those bordering rivers, creeks and coastlines can also give rise to misleading apparent movements; centroids in these areas tend to be unreliable for measuring apparent movement. In the case illustrated in Figure 4.17, the new block is truncated in the northwest compared to the old block but the match in area and perimeter was close enough for the algorithm to record them as a matched pair.

For these reasons it was determined that parcels that were deemed to be highly irregular in shape or to be polygonised roads would be handled separately from all other parcels when creating the blocks and their apparent movement would be computed from the mean apparent movement of surrounding blocks. This decision gave rise to the need to devise a classification scheme for parcels based on their shape. The classification process is described in Section 4.3.2.1

Figure 4.17 Matched blocks where the centroid shift vector is not useful

4.3.2.1 Classifying the parcels by shape

Before creating a new spatial layer to hold the blocks, the cadastral layers were pre-processed to classify the parcels into four spatial types: regular parcels, irregular parcels, urban roads, and slivers. The parcel types were determined using the shape and size properties of the polygon.

Slivers may arise from easements, i.e. a legal right for a landowner to enter another's land (Gray & Gray, 2009) or from errors in the dataset. The decision was taken to exclude them from the point matching processes; because of their tiny size their points make no significant difference to the final spatial adjustment process.

Urban roads were created as separate blocks so that urban blocks were more likely to be correctly matched (see Section 4.3).

Irregular parcels were created as separate blocks because their apparent movement was deemed to be misleading (see Figure 4.17 and its accompanying text).

Figure 4.18 outlines the processes used to create the blocks.

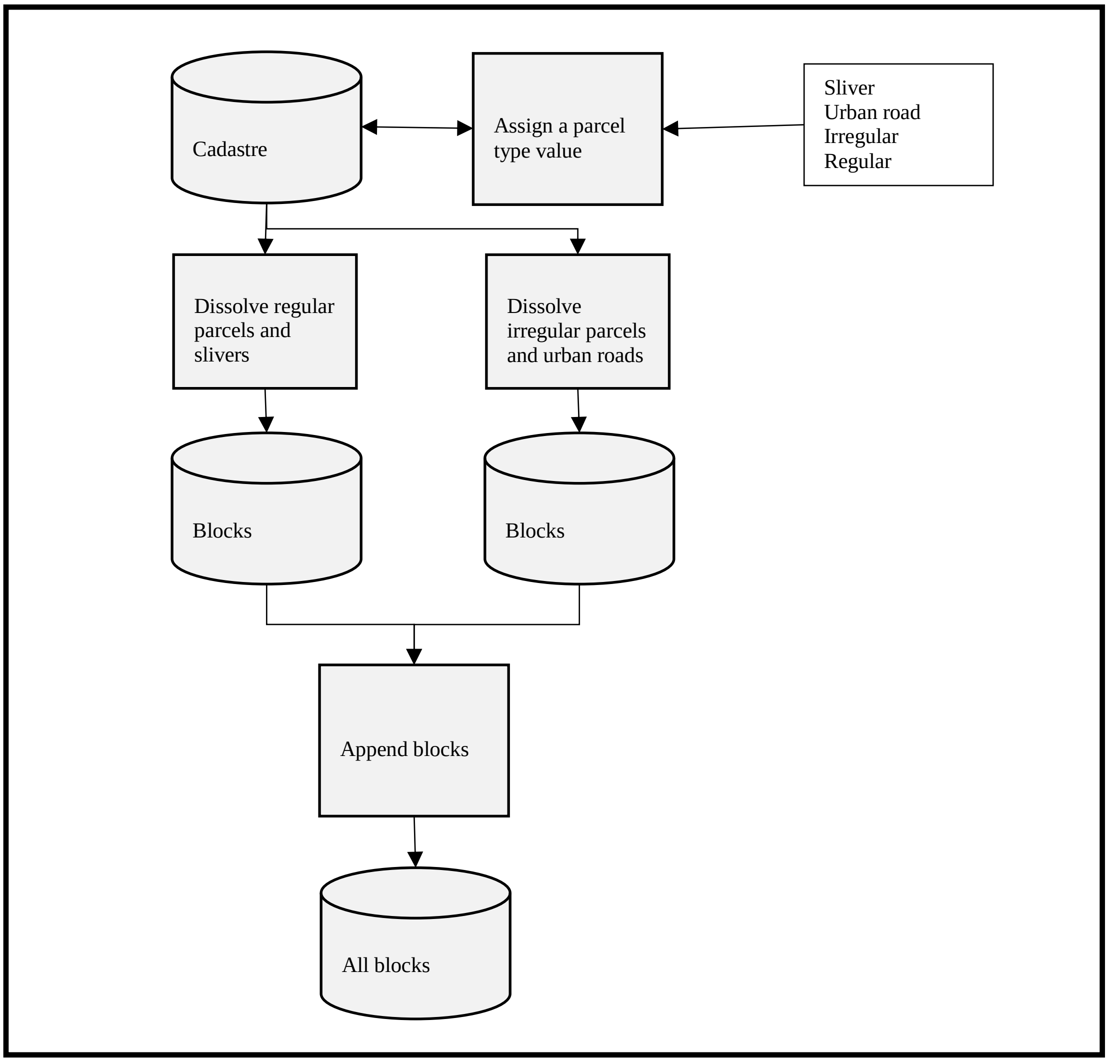

Figure 4.18 The block creation process

The spatial type of each parcel was assigned using rules that were arrived at by tests conducted on urban and rural dataset types using different threshold values and selecting those which best discriminated between the different parcel types. No automated method was discovered for evaluating the results, therefore, after each test run with varied selection expressions the results were thematically mapped and examined to evaluate the success of the expression. The judgement as to whether a given classification expression was successful was, of necessity, subjective. The tests were repeated on many different test data subsets with known areas of complexity. The rules finally implemented for each parcel type are described below:

Slivers: Any polygon with an area of less than one meter was classified as a sliver.

Urban roads: Any parcel where the ratio of the perimeter to the square root of the area is greater than 10 and the point count is less than or equal to 20 was classified as an urban road. This expression locates polygons with exceptionally long boundaries compared to a square with the same area. The point count is limited to less than or equal 20 to exclude the possibility of including rural roads and creeks in this category; separately classifying these features was found to improve the probability of correct point matching.

Irregular parcels: Any parcel where the ratio of the perimeter to the square root of the area is greater than 10 and the point count is greater than 20 was classified as an irregular parcel, i.e. all parcels with unexpectedly long boundaries other than urban roads were in included in this category.

Regular parcels: Any parcel not falling into one of the above categories was classified as a regular parcel.

The boundary conditions for the classifications are, of course, arbitrary but have been found to be useful in improving final control point identification results. The classification rules have been refined iteratively throughout the course of the research, their sole purpose was to improve the point matching algorithms.

Table 4.1 shows the number of parcels of each type identified in the twelve test datasets.

Table 4.1 Parcel type counts in the old cadastre datasets

Parcel type counts					
LGA ID	**Parcel Count**	**Regular Parcel Count**	**Road Count**	**Sliver Count**	**Irregular Count**
LGA01	655	654	0	0	1
LGA02	2,235	2,153	65	0	17
LGA03	2,273	2,245	20	3	5
LGA04	2,763	2,689	61	1	12
LGA05	5,166	5,045	45	0	76
LGA06	5,591	5,465	48	0	78
LGA07	7,263	7,228	14	0	21
LGA08	8,616	8,510	51	4	51
LGA09	11,783	11,323	317	40	103
LGA10	12,500	12,241	158	1	100
LGA11	21,349	20,778	214	296	61
LGA12	21,346	20,917	73	296	60

As with every other stage of this research, the ultimate test of all the individual algorithms comprising the final solution is the correctness of the control point matches. The parcel classification process and the separate dissolve processes using the classification have been found to improve the rate of correct block matching which, in turn, affects the correctness of the parcel matching, which, in its turn, affects the correctness of the point matching.

Figure 4.19 from LGA07 shows an example of the thematic map used to evaluate the classification results. The large map shows that there are two polygonised roads in the new cadastre; the inset at right shows part of the same area of the old cadastre where the same two roads are not polygonised – white areas in both maps are part of the world polygon. Had the polygonised roads not been modelled as separate blocks, the block matching algorithm would not have matched the old and new cadastral bocks in that area.

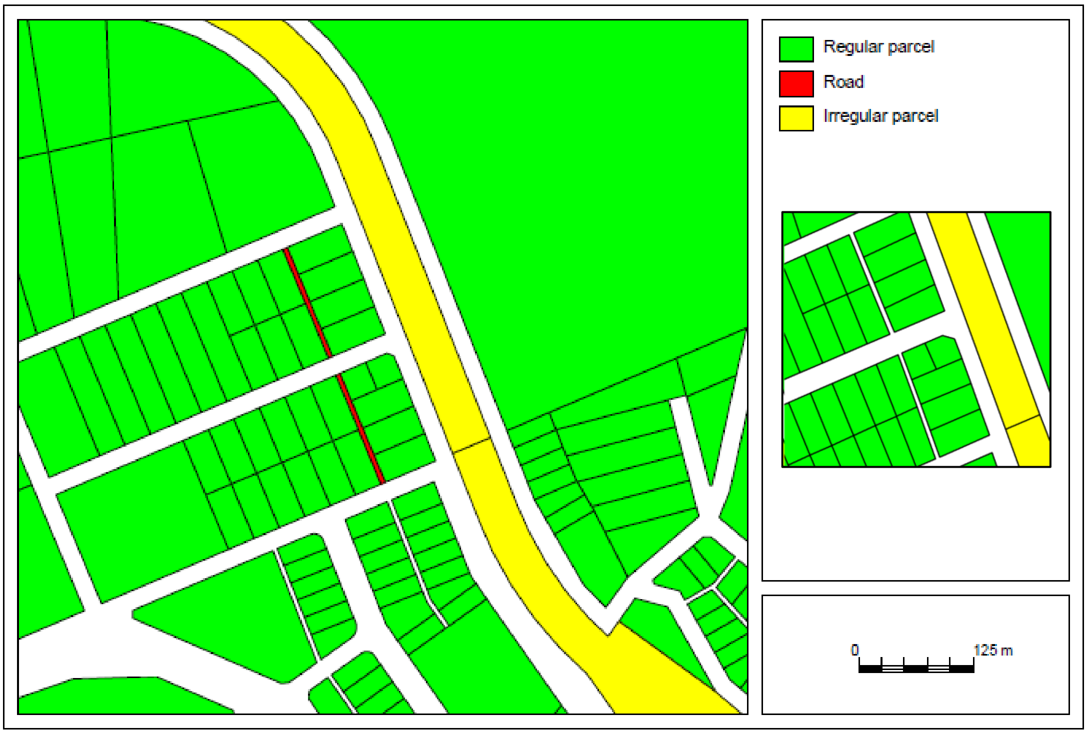

Figure 4.19 Parcel types

It has not been possible to quantify the value of including the parcel classification process; the only method available for evaluation of the results was to evaluate the correctness of the old-cadastre adjustment.

A test on the area shown in Figure 4.19 showed no significant difference in adjustment results when the parcel classification step was omitted but Figure 4.20, also from LGA07, shows an area where the adjustment results were affected. The map shows the adjusted old cadastre drawn in black over the new cadastre drawn in red. The inset on the right shows the old-cadastre adjustment results without parcel classification; they are noticeably poorer. The inset top left shows the classification used to process the main map; the regular parcels are shaded in green and the irregular parcels are shaded in yellow. As with many of the algorithms developed for the final solution, the effects are greater in some areas than others.

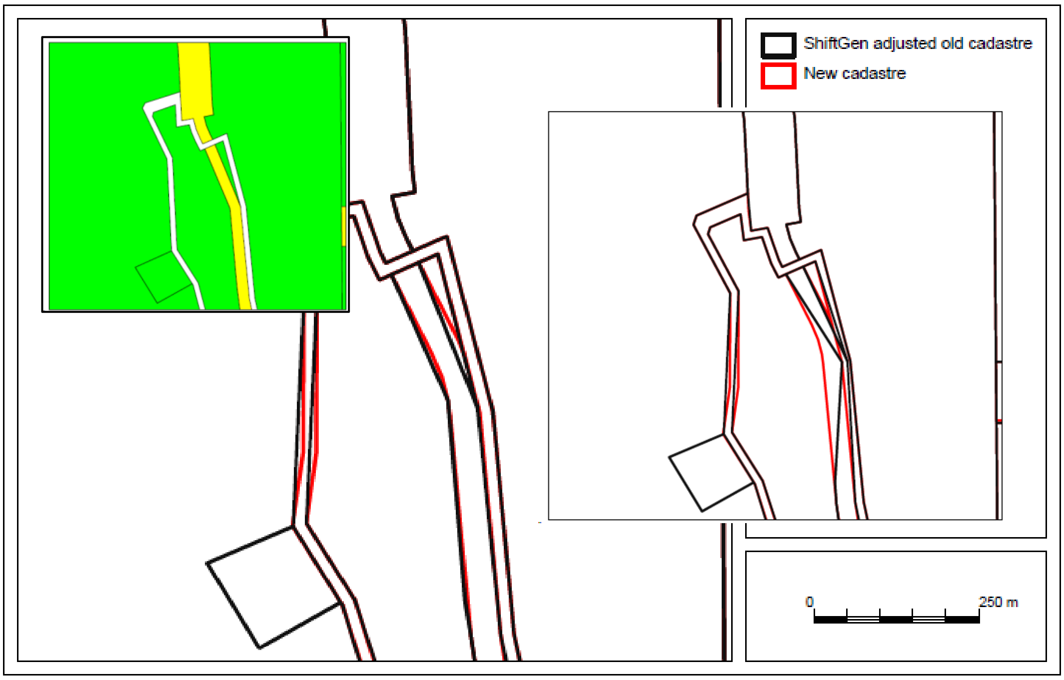

Figure 4.20 Adjustment results with and without parcel classification

In addition to the above classification, each parcel was also assigned a circularity index (see Section 4.3.2.3). The value of this index provides a means of checking the degree of shape similarity between two blocks or parcels.

The classification process was applied to both the old and the new cadastres prior to creating the blocks.

4.3.2.2 Creating the blocks

The Esri ArcGIS *Dissolve* tool (Esri, 2016d) was used to dissolve the parcel boundaries using the parcel type, regular parcels, urban roads, irregular parcels, and slivers, to control the output as illustrated in Figure 4.18. Because roads and creeks can sometimes occur as voids in one layer and not in the other, the urban roads and all irregular parcels were dissolved into one layer and the regular parcels and slivers were dissolved into another. The two layers were then appended.

This separation of the roads and irregular parcels allows for the blocks to be matched between layers where corresponding blocks are present in both layers but ensures that incorrectly polygonised roads do not distort the block shapes. Figure 4.21 from LGA07 shows an area where several roads have been polygonised in the new cadastre (outlined in green in map B) but not in the old cadastre (map A). Map C shows the blocks created from the new cadastre when the polygonised roads were not removed. Maps D and E show the results of block creation on the old cadastre and the new cadastre respectively after the polygonised road removal. They visually demonstrate the likelihood that more blocks will be successfully matched using the strategy described above; this is confirmed by the matching block ID numbers assigned by the block matching algorithm and displayed in the centre of each parcel; unmatched block have ID number 0. In these images, the old-cadastral blocks are drawn in black and the new-cadastral blocks are drawn in red.

Figure 4.21 Polygonised roads and resulting blocks

4.3.2.3 The matching algorithm for blocks

Polygon overlay was used on each block in the old cadastre to locate all the overlapping blocks in the new cadastre. The circularity index, the overlap area, and the area and perimeter similarity were all used to establish the most likely match as follows.

For each old cadastral block:

(a) Calculate a circularity index (*CI*) for the old block

$$CI = 4\pi A/P^2$$

where *P* is the perimeter of the shape and *A* is the area. For a circle CI = 1.

(b) Select all the blocks from the new cadastre that overlap the old block.

(c) For each overlapping block

(i) Calculate the circularity index of the candidate block,

(ii) Calculate an area ratio (*AR*) as follows:

$$AR = \min(A1, A2)/\max(A1, A2) * 100$$

where *A1* is the area of the current block and *A2* is the area of the candidate matching block.

(iii) Calculate and perimeter ratio (*PR*) as follows:

$$PR = \min(P1, P2)/\max(P1, P2) * 100$$

where *P1* is the perimeter of the current block and *P2* is the perimeter of the candidate matching block.

(iv) Calculate a circularity difference (*CD*) as follows:

$$CD = \text{abs}\,(CI1 - CI2) * 100$$

where *CI1* is the circularity index of the current block and *CI2* is the circularity index of the candidate matching block.

(v) Calculate the match score (*MS*) as follows:

$$MS = (100 - AR) + (100 - PR) + CD$$

(d) Choose the block with the lowest value of *MS* as the best match.

(e) Assign a unique identifier (BlockID) to each matched pair.

A vector was then constructed linking the centroids of each matched pair. Figure 4.22 illustrates some results from LGA11. The direction and length of each centroid shift vector were stored as attributes of the block, to be used later in the parcel matching process. Wherever a block was not matched, the direction and length attributes were calculated using the mean values from all nearby matched blocks.

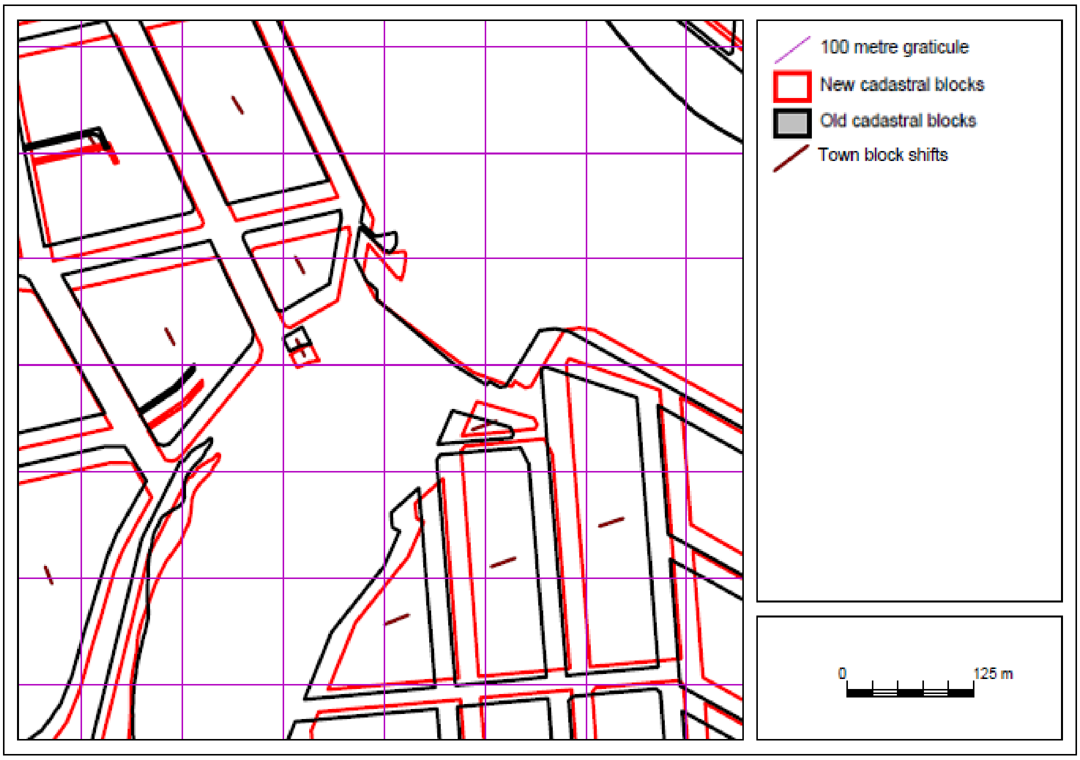

Figure 4.22 Vectors generated between matching block centroids.

No attempt was made to match blocks with areas greater than 100 hectares. This is because test runs on the rural datasets showed that centroids generated from very large matched pairs of blocks were unreliable for use when matching small residential parcels contained within the block. For example, Figure 4.23 from LGA11 shows a small town in a large rural block of about 2300 ha (shown in the inset map) where most of the parcels have been incorrectly matched; this dataset had matching UID attributes on the parcels so that the correctness of the matching process could be checked. By calculating this block's apparent movement from the mean movement of surrounding matched blocks, correct matching of all the parcels in this area was achieved.

In the early stages of the research, a process was developed to remove statistical outliers from the set of block centroid shift vectors. However, as the research progressed and the block matching algorithms were improved, this process was found to be unnecessary.

Figure 4.23 Incorrectly matched parcels in a large rural block

4.3.3 Matching the parcels – first pass

Before carrying out a final parcel matching process, a simple parcel-matching algorithm was executed. The centroid of each parcel was projected using the block centroid shift angle and distance and this projected point was then used to locate a parcel in the other layer by the point-in-polygon method. A vector was created between the centroids of the matched parcels. This process was repeated using each layer in turn, i.e. old cadastral parcels were first matched to new cadastral parcels and then new cadastral parcels were matched to old cadastral parcels.

Polygons that enclose very highly irregular areas such as rivers and creeks can potentially give rise to misleading apparent shift vectors (see Figure 4.17). However, such parcels can safely be ignored during the parcel matching process because, in datasets formatted as shapefiles, each vertex on the riparian boundary will also be present on the adjacent parcel boundary and these can be used for the later shift generation process in place of those on the riparian boundary (see Section 5.7 on the selection of points to be used in the final shift generation process).

In urban areas, the centroid shift generation process typically results in a correct bijective match between two urban parcels. Where this is the case, two vectors of identical length and opposite angle are created, one from each of the two cadastre layers, and it can be assumed that, if their areas are similar, the two parcels are a match to each other, i.e. they both represent the same property. However, where there have been subdivisions and amalgamations, two or more vectors of different lengths may terminate at the same point as shown in Figure 4.24 from LGA07. These vectors can then be used to locate many of the subdivisions and amalgamations in order to eliminate them before attempting to match points (see Section 4.3.4).

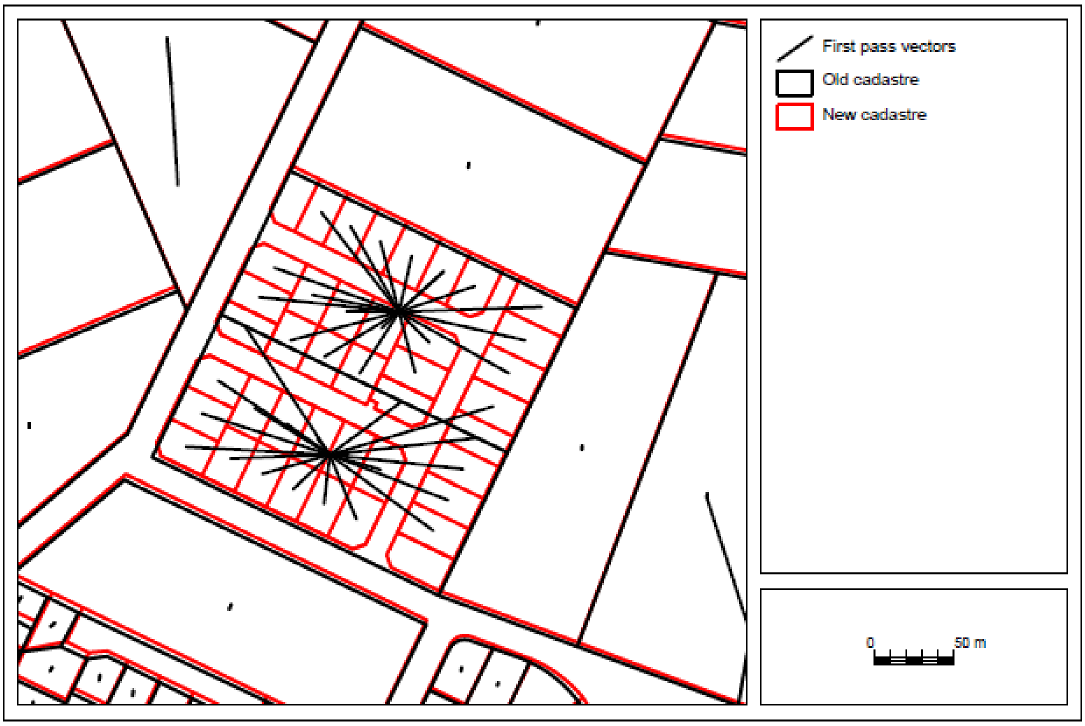

Figure 4.24 An area with a new subdivision

In detail, the process used to match the parcels is, for each cadastre in turn:

(a) For each "regular" parcel in turn:

- (i) Use the angle and distance of the centroid shift vector of the overlapping block to project the parcel centroid.
- (ii) Select the parcel from the other cadastre in which the projected point falls.
- (iii)Construct a vector between the centroids of the two parcels.

Using the generated vectors, for all duplicated vectors, i.e. all vectors that joined the same pair of parcels in each pass, calculate the mean and standard deviation of vector lengths.

4.3.4 Identifying subdivisions and amalgamations

In the preliminary stages of this research, no attempt was made to discover subdivisions and amalgamations in the cadastral layers. Later, however, once the code to generate shift vectors (described in Chapter 7) had been developed, it became apparent from the discovery of poorly adjusted locations in the adjusted old cadastre, that the results could be improved by adding a process to identify as many subdivisions as possible.

To identify subdivisions and amalgamations the non-duplicated touching centroid shift-vectors from the first-pass parcel matching process described in Section 4.3.3 were used. Where several parcel-centroid shift vectors all touched at a single centroid point, the total area and outer perimeter of all the origin parcels were compared to the area of the destination parcel at the central point and, where the areas and outer perimeters were sufficiently close, the origin parcel boundaries were dissolved. This was carried out for each cadastral layer in turn. Figure 4.25 from LGA04 shows an urban area where several subdivision boundaries (shown in red) have been removed by this process. The process ensures that, at the point matching stage described in Chapter 7, points that cannot be expected to match to a point in the other layer will be removed, i.e. the points from the internal subdivision or amalgamation boundaries will never take part in the point matching process. Before this process was implemented, the research undertaken for this thesis had shown that such points could result in incorrect shift vectors which then caused poor adjustment around the point and increased the number of areas an operator would need to check.

Figure 4.25 Cadastre showing removed subdivision boundaries in red

In detail, the process used to identify the subdivisions and amalgamations was, for each cadastre in turn:

(a) First, remove all vectors that touch another at both ends. This can occur when a row of urban parcels all appear to have moved a significant distance so that a parcel has a centroid that falls on the neighbouring parcel in the other cadastre. Figure 4.26 from LGA01 shows examples.

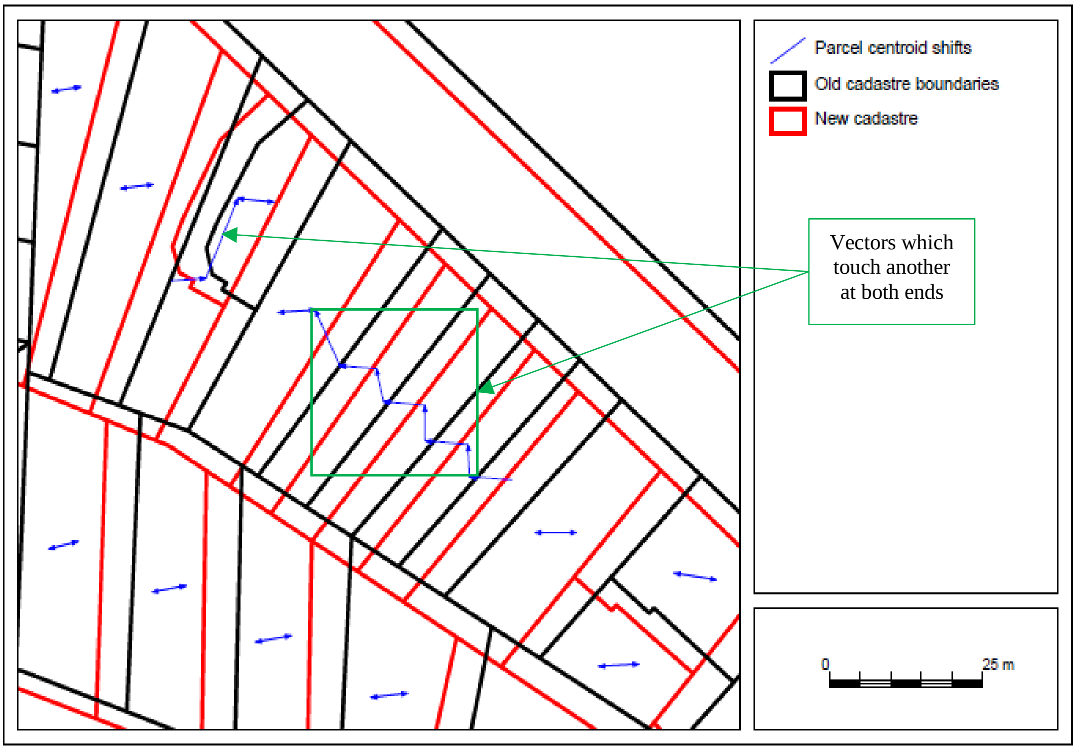

Figure 4.26 Chained shift vectors

(b) Where more than one vector terminates in the same parcel, if the total area of the multiple vector-origin parcels is within 10% of the area of the single destination parcel, and the outer perimeter of the origin parcels is within 10% of the perimeter of the destination parcel, flag the destination parcel as a subdivision/amalgamation. The 10% area and perimeter tolerance are used to allow for the fact that the cadastral updates are unlikely to have resulted in areas and perimeters that match exactly.

(c) Make new layers from the two cadastres to contain the groups of parcels which have matched to a single parcel in the other layer.

(d) Dissolve all internal boundaries in the new layers.

(e) Eliminate all polygons in the dissolved layers where the dissolve has resulted in more than one shape for the supposedly same amalgamation.

(f) Remove the remaining valid subdivisions from the cadastral layers and merge in the dissolved shapes.

The area and perimeter tolerances were found to work well in urban areas where inspection of the mapped results, such as that shown in Figure 4.25 from LGA)7, showed few errors. However, the algorithm required further refinement in rural areas where parcels of land may be exchanged between neighbouring properties and two or more touching centroid shift vectors do not always indicate a simple subdivision. Figure 4.27 shows an area where three centroid shift vectors are touching. The inset shows an enlarged area of the meeting point. In this case, the centroid of the triangular new-cadastre parcel (highlighted in magenta) happens to fall on the large old-cadastre parcel at the centre of the map rather than on the correct matching triangular parcel. This has resulted in the longest of the three vectors. However, the centre of the large new-cadastre parcel to the east of the triangular parcel also falls on the same old-cadastre parcel. In this example, the change appears to be due to a correction to the original version of the cadastre.

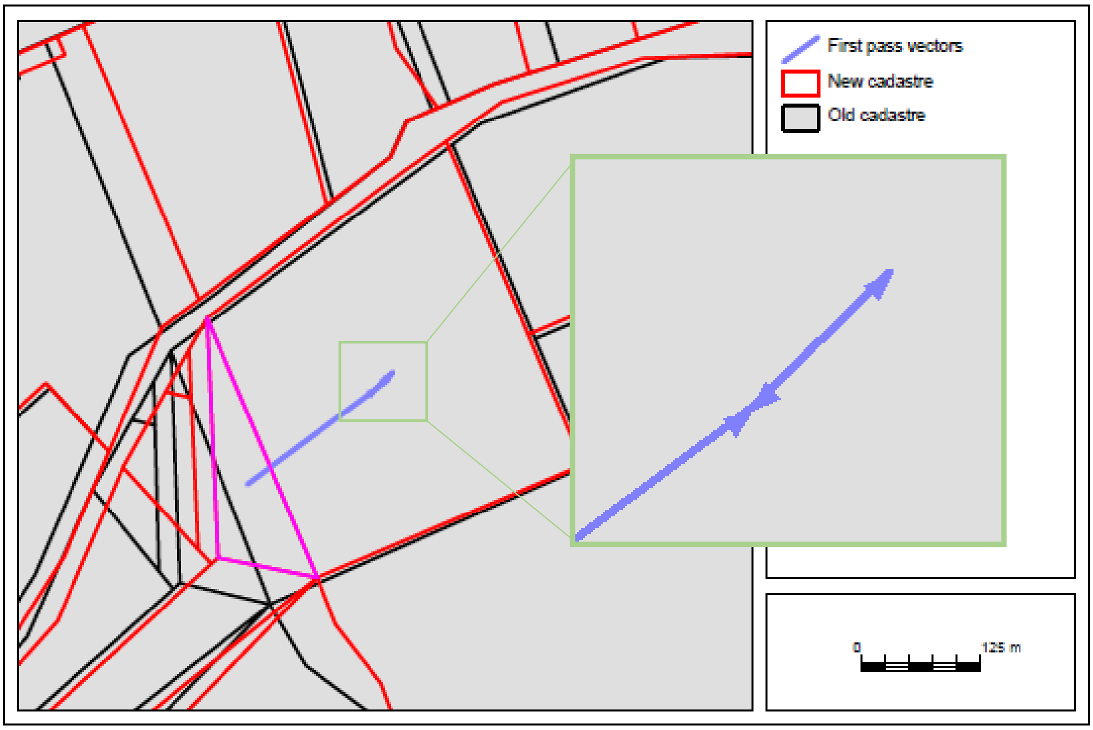

Figure 4.27 Three touching centroid shift vectors

Using the algorithm described above, using the 10% area and perimeter comparison rule, the two linked new-cadastre parcels would have been identified as a subdivision of the old parcel but this is clearly not the case. An additional rule was therefore applied:

(g) Where three or more centroid shift vectors touch and two of the vectors join the same two parcels, for example, old-parcel 3 links to new-parcel 5 and new-parcel 5 links to old-parcel 3, the parcel group shall not be included in the subdivision process unless the combined area of parcels in one layer is within 99% of the area of the single parcel in the other layer, i.e. 1% difference in area, rather than the 10% difference used in rule (b) above.

Figure 4.28 shows an area where a subdivision has been correctly identified using this rule. Here four parcel centroid shift vectors (blue) touch and the easterly pair are duplicated. However, the rule ensured that the parcels shown in Figure 4.27 were not incorrectly identified as part of a subdivision. In this figure the single hatched parcel in the old cadastre has been split into the three yellow shaded parcels in the new cadastre.

Many experimental tests were executed on datasets where subdivisions or amalgamations had been identified by locating areas with multiply touching centroid-shift vectors, for example, as shown in Figure 4.24 from LGA07. Threshold values for the rules were varied until results satisfactory to the author were obtained; no objective method for evaluating the results has been identified. The values are a compromise between identifying too many incorrect subdivisions or amalgamations and identifying too few.

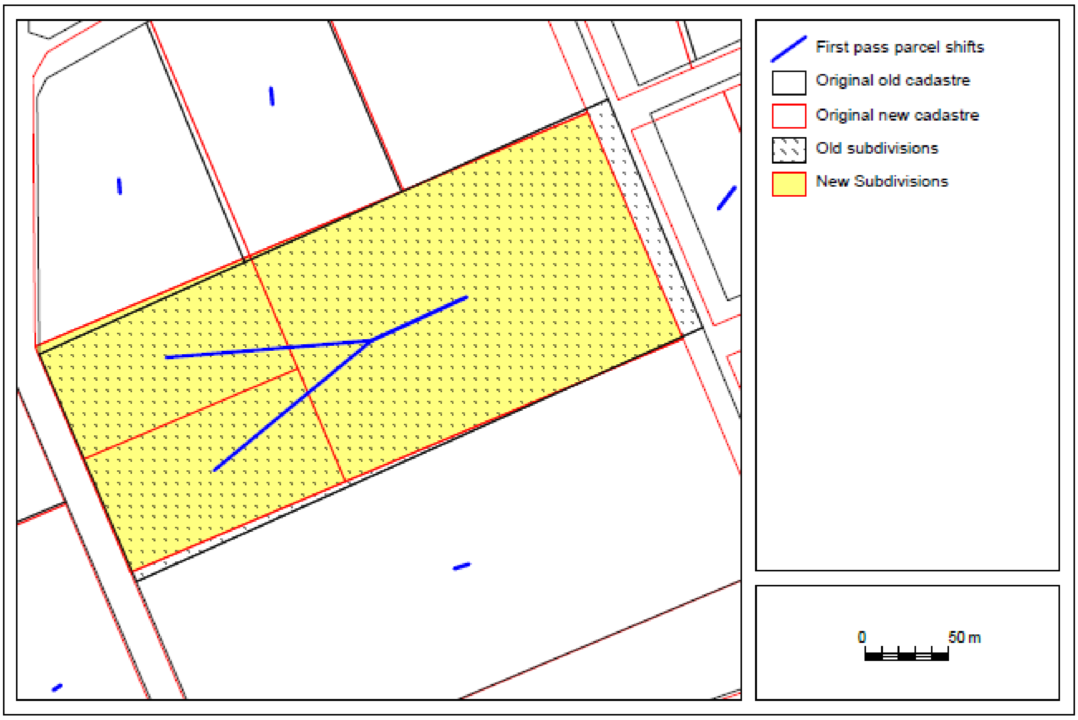

Figure 4.28 A correctly identified subdivision.

Table 4.2 shows the number of subdivisions identified in each LGA dataset. The rightmost column shows the count of potential subdivisions that were rejected by the application of the above rule. The considerable number rejected in LGA08 was found to be due to incorrect topology over a large area where all the parcels in the new cadastre were duplicated. This resulted in three centroid shift vectors for each parcel. They were correctly rejected, i.e. they did not identify areas where subdivisions or amalgamations had occurred.

Table 4.2 Subdivision counts

LGA ID	Parcel count	Subdivision count	Not a Subdivision
LGA01	655	2	9
LGA02	1026	2	246
LGA03	2258	34	152
LGA04	2353	165	165
LGA05	2785	39	114
LGA06	5198	132	235
LGA07	5630	105	73
LGA08	7262	2	1071
LGA09	8679	248	32
LGA10	12239	483	660
LGA11	12597	103	256
LGA12	21514	1102	37

Once this process has been completed for both layers it can be expected that it should be possible to match the merged polygons in the parcel matching process. Point matching can be more accurately achieved on matched parcels and removing the subdivisions and amalgamations makes a match more likely. As expected, the process resulted in improved results at the point matching stage described in Chapter 7 thus reducing the number of incorrect shift vectors that would need to be manually removed or replaced. In addition to allowing a greater number of parcels to be matched, the process removes all the internal points defining the subdivision boundaries thus preventing incorrect shift vectors from originating or ending on those points.

From examination of thematically mapped results, it has been determined that the algorithm works well in many cases but that it does not, and cannot in its present form, correctly identify all subdivisions and amalgamations. For example, the area match will fail when the subdivided area contains a new road that has become part of the world polygon. Figure 4.29 from LGA07 shows an area where several subdivisions have been correctly identified but the two old-cadastre parcels (outlined in black) in the centre have not been so identified because the road void in the new subdivision caused the area-match check to fail. Where subdivisions have been missed or wrongly identified, the final adjustment tends to be poor; fortunately, these can be easily identified by viewing the results in a GIS viewer (see Figure 4.36) at the error correction stage described in Section 8.3.

In general, correctly identifying subdivisions and amalgamations before attempting a parcel matching process improves the homotopy of the two layers and can be expected to result in a greater number of matched polygons.

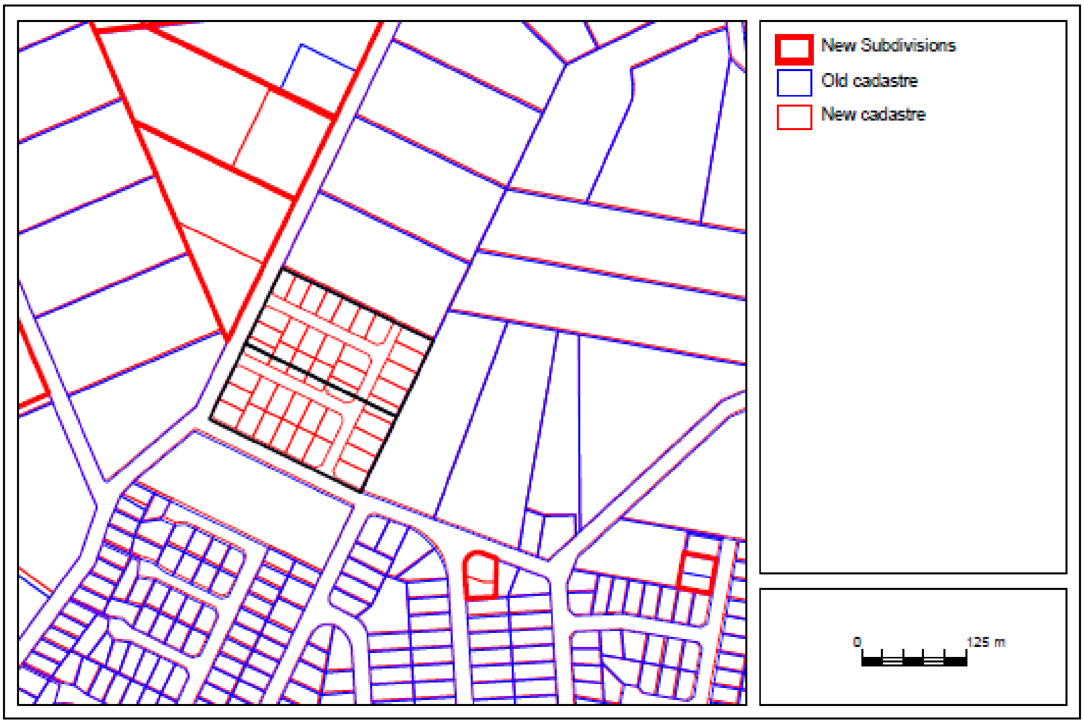

Figure 4.29 An unidentified subdivision

4.3.5 Matching the parcels – the second pass

Using the statistical analysis resulting from the first-pass centroid shift vector generation it has been found possible to determine whether a more sophisticated method of parcel matching than that used in the first pass would be needed. In urban areas, where most parcels are small and the simple overlap method results in correct matches in the majority of cases, inspection of the statistical results showed, not unexpectedly, that the variance in the centroid shift vector lengths was very small. This is because the apparent shift between versions, in these areas, is often very small. For example, the mean length of centroid shift vectors was less than three metres and the standard deviation was under one metre for several urban datasets whereas the corresponding values for the rural datasets were considerably larger. Table 4.3 shows the statistical results from the first pass centroid shift vectors. The table is ordered by increasing standard deviation of shift length.

Table 4.3 Centroid shift vector length statistics

LGA ID	Dataset Type	Centroid shift count	Min. Length	Max. Length	Mean Length	Length Sigma
LGA12	Urban	38894	0.00	43.60	0.06	0.83
LGA04	Urban	4130	0.00	18.53	0.40	0.65
LGA09	Urban	16394	0.00	22.26	0.64	0.55
LGA10	Urban	20628	0.00	26.19	0.68	1.21
LGA08	Urban	13852	0.08	11.34	2.48	1.48
LGA02	Urban	1970	00.1	9.43	2.52	2.27
LGA01	Urban	1234	0.14	24.99	2.65	2.69
LGA03	Rural	4178	0.00	116.42	9.49	8.32
LGA11	Rural	22010	0.01	258.60	13.26	8.35
LGA07	Rural	10350	0.00	251.89	4.93	9.44
LGA05	Rural	5240	0.00	227.80	4.88	11.32
LGA06	Rural	9446	0.00	741.37	8.70	23.31

Where the mean shift vector length was greater than 3 metres or the standard deviation was greater than 1.5 metres, the more complex second-pass processing described in Section 4.3.5.2 was used for that dataset. Note that datasets LGA01 and LGA02, even though they comprised largely urban sized parcels, were selected for processing by the more complex method because they each exhibited large apparent movement in some areas.

The processing method distinction was made purely for efficiency, the second-pass method would work equally well on urban datasets but the extra processing time is unnecessary. Processing the urban datasets using the simpler parcel matching method described in Section 4.3.5.1 allowed testing to proceed more rapidly. The values used to decide which method should be chosen were initially arbitrary but tests showed that using the simpler method did not cause degradation of results for the urban datasets.

The methods adopted for the different dataset types are described below. Note that LGA01 and LGA02 are both largely urban areas but each has a significant number of rural sized parcels which resulted in them being selected for processing by the more complex matching algorithm described in Section 4.3.5.2.

4.3.5.1 Second pass – urban datasets

The first pass parcel matching process (Section 4.3.3) was repeated one way only, i.e. using the centroid of each old cadastral parcel to find a parcel in the new cadastre by point-in-polygon search. Matching from the old cadastre to the new cadastre was carried out after subdivision/amalgamation removal. Unlike the process used for rural and mixed datasets, described in the next section, block shift vectors were not used in the matching process because the statistical results indicated that block shift vectors were unlikely to be necessary for the current dataset, i.e. the mean length and standard deviation of the parcel centroid shift vectors was very small implying that the simple point in polygon matching algorithm described in Section 4.3.3 would have a high probability of matching the parcels correctly.

4.3.5.2 Second pass – rural datasets

In the initial stages of the parcel matching research, a single algorithm was developed for all regular parcels. That matching process involved the selection by overlap of candidate parcels in the new cadastre that were potential matches to the current parcel from the old cadastre and selection of the best match from this set. The algorithm was as follows:

For each regular cadastral parcel in the old cadastre select all overlapping parcels in the new cadastre, then, for each overlapping parcel:

(a) Calculate the area ratio AR as for blocks (see Section 4.3.2.3).

(b) Calculate the absolute angular difference between the azimuth of a line drawn between the centroids of the current and candidate parcel and the azimuth of the centroid shift vector of the overlapping block as a percentage of 180 degrees (dA).

(c) Calculate the absolute difference in length between the length of a line drawn between the centroids of the current and candidate parcel and the length of the centroid shift vector of the overlapping block as a percentage of the block shift vector length (dL).

(d) Calculate the match score (MS) as follows:

$$MS = dA + dL + (100 - AR).$$

This algorithm was found to work very well in most cases but, as the research progressed, areas were discovered where the algorithm did not work at all well. In particular, a small group of almost identical urban parcels in an otherwise largely rural block would be unduly influenced by the block centroid-shift distance resulting in incorrect matching as shown in Figure 4.30 from LGA07. Most of the parcels in the outlined area have been wrongly matched as can be seen from the vectors joining the centroids of the matched parcels. Because there were no matching UIDs in this dataset, there was no way to check the correctness of the match except by inspecting the centroid vectors in a GIS viewer.

The solution to the problem was to classify the parcels into three distinct types, each type being processed slightly differently. These types are:

(a) Regular parcels occupying a small area in a large rural block.

(b) Other regular parcels.

(c) Highly convoluted parcels. A parcel was deemed to be highly convoluted if its circularity index (see Section 4.3.2.3) was less than 0.2.

The convoluted parcels were matched by the algorithm described above. The matching methods for the other types are described below.

Figure 4.30 Incorrectly matched urban parcels in a rural block

4.3.5.3 Matching regularly shaped parcels in largely rural blocks

Parcels were processed by the method described in this section if all the following conditions were satisfied:

(a) The parcel falls in a block where the total area of urban parcels is less than or equal to 20% of the block area.

(b) The block area is greater than 100,000 m^2.

(c) The block centroid shift vector length as a percentage of the square root of the block area is greater than 10.

(d) The parcel area is less than or equal to 10,000m^2.

The values used in these expressions were determined by varying the threshold values in the above expressions and observing the results in areas which exhibited this problem; the values finally adopted were chosen because inspections showed that they produced correct parcel matching in those areas.

For parcels selected using the above criteria, using the centroid of the parcel, a circular buffer having a radius of one fifth of the square root of the parcel area was then created. The reasoning behind this choice of buffer distance is as follows; regular parcels of this size are likely to be rectangular and, typically, rectangular parcels have a width of somewhere between one half and one third of their length; the buffer distance is, therefore, likely to be less than half the width of the rectangle. Limiting the search area in this way makes it less likely that a parcel will be incorrectly matched to its neighbour in the new cadastre; this is what has occurred in the rural block shown in Figure 4.30.

The buffered centroid was then used to locate all parcels in the new cadastral layer that overlap the buffer shape. The overlapping parcels were processed one-by-one and the best match selected with the same algorithm as used for block matching (see Section 4.3.2.3).

Figure 4.31 shows the same area as in Figure 4.30 after the dataset was processed using the amended rules described in this section. All the parcels in the outlined area have now been correctly matched.

Figure 4.31 Correctly matched urban parcels in a largely rural block

4.3.5.4 Matching other regular parcels

Regular parcels were matched by projecting the centroid of the parcel by the angle and distance of the block containing the parcel and selecting the parcel in which the projected point falls in the new cadastre. The following tests were then applied to determine whether the new parcel is a match.

(a) There is less than 15% difference in the circularity indices (see Section 4.3.2.3).

(b) There is less than 15% difference in areas.

(c) There is less than 15% difference in perimeters.

If the selected new-cadastre parcel passes all the above tests it is deemed to be a match. The 15% tolerance was determined by varying the value and observing the number of correct and incorrect matches achieved on the datasets with UIDs until results satisfactory to the author were obtained.

4.3.6 Eliminating poor parcel centroid shift vectors

Section 4.3.5 described the different matching algorithms used to match parcels depending on whether the datasets was determined to be entirely urban or not. However, an automated parcel matching process can, inevitably, create incorrect matches. Where there were no existing UIDs available to automatically check the correctness of the match, a statistical analysis was carried out to determine the mean and standard deviation of the centroid shift vector lengths and then eliminate those falling outside of reasonable values.

The method used to eliminate potentially erroneous shift vectors differs depending on which parcel matching algorithm had initially been chosen. For the urban datasets processed using the algorithm described in 4.3.5.1, all centroid shift vectors longer than the mean centroid shift vector length plus five times the standard deviation were deleted. Inspection of results in a GIS viewer suggested that this check should be included and showed that the results for urban datasets were improved after the step was adopted.

The non-urban datasets were processed with different, more complex rules. They are described below.

4.3.6.1 The elimination rule applied to all parcels in a non-urban layer

For all parcels where there are more than 20 shift vectors in a block, a rule was developed to remove anomalous shift vectors as follows: eliminate shift vectors that cross more than one old-cadastre parcel when more than 80% of shift vectors in the block do not do so.

Figure 4.32 shows an example of an area from LGA01 where four shift vectors (green) have been deleted using this rule. The shift vectors drawn in blue are those remaining after the weeding process. In this dataset, UIDs were present and indicated that the parcels in question had been incorrectly matched.

The rule was arrived at by inspecting parcels where the UIDs indicated a mismatch and observing how the centroid shift vectors differed from those in the surrounding, correctly matched, areas, i.e. they cross parcel boundary lines whereas others in the same block do not.

Removing the incorrect vectors forces the point matching algorithms to use the mean of the apparent movement of surrounding parcels with the result that control point matches in the area are correct, as indicated by the correct old-cadastre adjustment results shown in the inset where the adjusted old cadastre in black has been overlaid on the new cadastre in black.

Figure 4.32 An area where four parcel centroid shift vectors have been deleted

4.3.6.2 Parcels where the UIDs do not match or UIDs are not available

The following rule was applied to effect the elimination of unwanted parcel-shift vectors in blocks where the block shift vector length was available:

Eliminate all shift vectors where the following expression is true:

$$PSL > \max(\min(5 * BSL, 10), 150)$$

PSL is the parcel shift vector length and *BSL* is the block shift vector length.

Where no block shift-vector length was available, all shift vectors were eliminated where the vector length was greater than the mean length of all the parcel centroid-shift vectors plus five times the standard deviation of the shift-vector lengths.

The values of the thresholds used in these expressions were arrived at by varying their values and inspecting the vectors removed by the algorithm. After the point-matching process described in Chapter 7 was implemented, the matching results for datasets without a UID, or unmatched parcels with UIDs, were evaluated by inspecting the adjustment results; if a parcel is incorrectly matched, the generated shift vectors and consequently the adjustment results in that area are sometimes poor; poor results become obvious when the adjusted old cadastre is drawn on top of the new cadastre in a GIS viewer, as can be seen in the lower inset in Figure 4.32.

4.3.6.3 Direction of apparent movement of blocks.

In earlier stages of the research, an attempt was made to include the block shift vector direction, in addition to the length, as a means of discovering incorrect parcel shift vectors. However, this approach was discontinued when it was discovered that the parcel-shift angle could vary quite considerably across a block.

Figure 4.33 shows an area from LGA01 where the apparent parcel movements differ widely in direction across a single block especially where the apparent parcel movement is small.

Figure 4.33 Varying parcel centroid shift vector directions

4.3.6.4 Updating the parcel attributes

After creating and weeding the parcel centroid shift vectors, the length and azimuth of the remaining vectors were added to the parcel attributes for use in the later point matching process. Also, a new UID was generated and assigned to each matching pair of parcels.

For unmatched and incorrectly matched parcels, the shift vector length and azimuth attributes were computed from the mean shift vector length and azimuth of the surrounding matched parcels.

4.4 Evaluating the parcel matching results

Where available, the existing UIDs were used to evaluate the correctness of the matching algorithms; the correctness or otherwise of the match can be inferred by comparing the new UID values assigned by the software (the generated UID) and the original UID values available in the attribute tables of the supplied data. Figure 4.3 showed an example of a location where there appear to be two wrongly matched parcels although, in that case, it seems that the UIDs supplied with the data are wrong in one of the two cadastres.

Where there are existing UIDs, it is possible to classify the results for each parcel into several types. A list of all the original UIDs that are present in both cadastres was prepared – the match list. It was then possible to use this list to assign an attribute to the cadastral table indicating the type of match achieved:

(a) Correctly matched; the original UID pairs and the generated UID pairs both match.

(b) Wrongly matched, i.e. a false positive; the original UIDs do not match but the generated UIDs do.

(c) Should have been matched, i.e. a false negative, i.e. a pair of parcels have matching original UIDs but the software has failed to match the old cadastral parcel to any parcel in the new cadastre.

(d) Unexpectedly matched, i.e. a possible mismatch where an old parcel has been matched to a new parcel but the original UID is not present in the UID match list.

(e) No match expected, i.e. the original UID on the old parcel is not present in the match list.

(f) Unmatched.

Results were evaluated by inspecting the generated centroid shift vectors in a GIS viewer and by thematically mapping the old cadastre parcels using their match type, concentrating particularly on areas where the matching process was expected to be difficult, for example, in the area shown in Figure 4.34 where the matching parcels have no overlap between the two layers.

The parcel shaded in orange was unmatched because it had been identified by the software as a highly irregular shape; the algorithm is designed to ignore irregular shapes. The parcel centroid shifts, in blue, indicate that all the parcels in this area have been correctly matched.

Where UIDs are available, the match results were easily evaluated by both the count of correctly matched parcels and by inspection of the thematically mapped data in a GIS viewer as can be seen in Figure 4.35 from LGA11 where the map has been created using the parcel match-type to control the colours. In addition to the objective results obtained by comparing the match results with the existing UID values, inspection of the thematic map allowed discovery if the individual wrongly match or unmatched parcels. These observations were then used to improve the matching algorithms wherever possible.

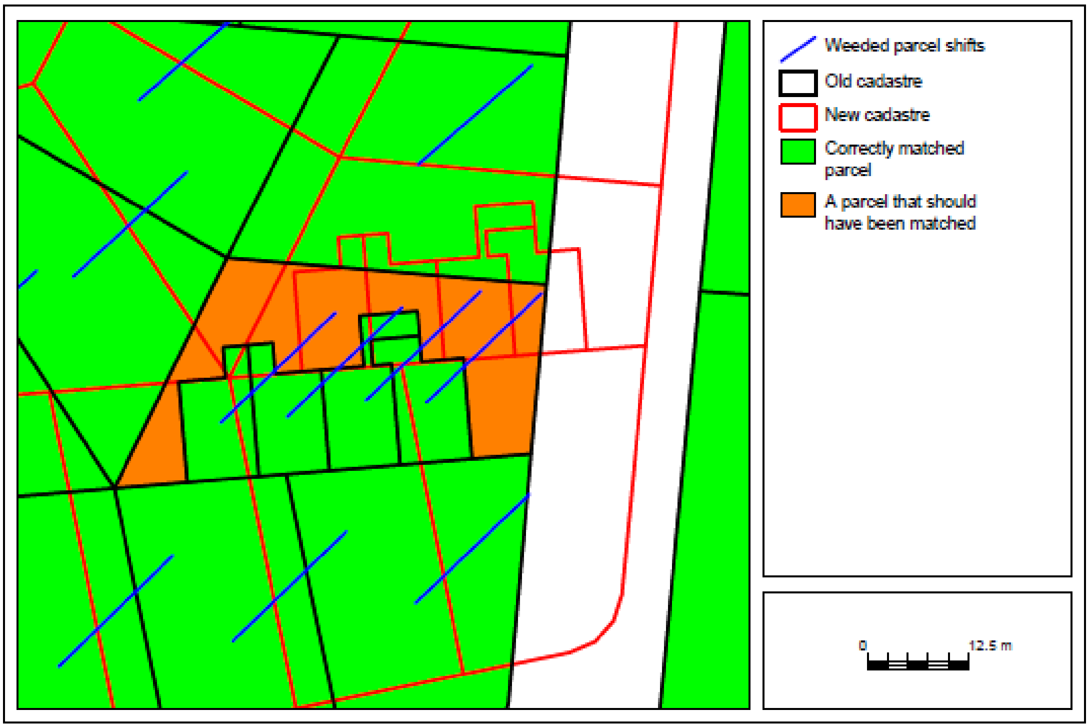

Figure 4.34 Non-overlapping parcels

In general, if original UIDs exist, it is anticipated that, after an adjustment has been carried out, inspection of areas where parcels are not correctly matched can be used during the final manual error checking stage that is always required (see Section 1.2) to guide an operator to places where the generated shift vectors may need correction. Figure 4.35 shows an area from LGA01 where several parcels have been unexpectedly matched (yellow). The three yellow shaded parcels fronting the north side of the road have been incorrectly matched because of the area and perimeter tolerances used in the parcel matching algorithm. It is not clear in this area how the drainage data (brown) should be adjusted; local knowledge would be needed, or recapture of that region of the drainage data.

It would, of course, be possible to omit the parcel-matching step where UIDs exist and just assume that all UID matches indicate correct parcel matches. However, this research has shown that in a few cases, such as the ones shown in Figure 3.7 Figure 4.3, this assumption may be incorrect. The parcel matching process on its own could, therefore, be utilised to discover UID errors and consequently improve data quality.

Figure 4.35 An area with several incorrectly matched parcels

Table 4.4 shows the match results for the datasets with UIDs. The percentage of correctly matched parcels in the right-hand column has been computed from the number of regular parcels where matches were expected, i.e. the analysis of the UIDs available as parcel attributes had shown that the same UID existed in both cadastres. The only rural dataset in this group is LGA11. All the correct matches have been validated using the existing parcel UIDs.

Table 4.4 Match type counts for datasets with UIDs

Thesis ID	Regular parcel count	Number of correctly matched parcels	Number of unexpectedly matched parcels	Number of parcels that should have been matched	Number of wrongly matched parcels	Number of parcels where no match is expected	% Correctly matched
LGA01	653	597	21	15	10	10	92.85
LGA02	1018	983	10	25	0	0	96.56
LGA08	7232	6877	24	43	12	276	98.86
LGA11	12345	11590	116	418	69	152	95.05
Totals	**21248**	**20047**	**171**	**501**	**91**	**438**	**96.33**

Table 4.5 shows the match results for the datasets without UIDs. In these datasets, it cannot be said with certainty that the matches are correct as there are no available existing UIDs to validate the match. Inspection of the thematically mapped results and the later control point identification process that relies on them has suggested that the majority are correct.

Table 4.5 Percentage of parcels matched for datasets without UIDs

LGA ID	Regular parcels	Matched parcels	Unmatched parcels	% matched parcels	Dataset type
LGA03	2178	2085	93	95.73	Rural
LGA04	2256	2211	45	98.01	Urban
LGA05	2710	2626	84	96.90	Rural
LGA06	5064	4756	308	93.92	Rural
LGA07	5487	5211	276	94.97	Rural
LGA09	8550	8395	155	98.19	Urban
LGA10	11764	10793	971	91.75	Urban
LGA12	20916	20441	475	97.73	Urban
Totals	**58925**	**56518**	**2407**	**95.92**	

Inspection of dataset LGA10, where the results were significantly poorer than those for the other datasets, showed that there were many parcels that had been subdivided between the original and the updated version. Many of these were of a type that could not be discovered by the rules described in Section 4.3.4. Figure 4.36 shows two such subdivision areas (outlined in green). The inset shows the same area after adjustment, showing how failure to detect a subdivision has given rise to incorrect results in the north-westerly of the two subdivisions; the adjusted old cadastre has been drawn in black over the new cadastre in red.

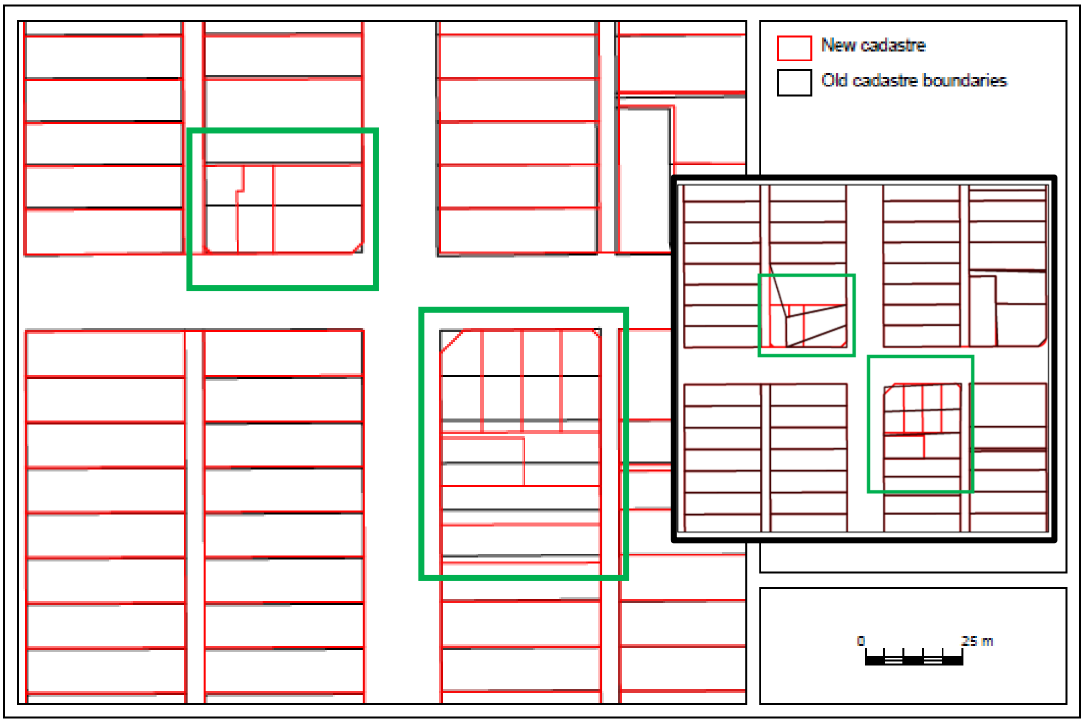

Figure 4.36 Undiscovered subdivisions

It is not possible to quantify the number of missed subdivisions but 1167 regular (see Section 4.3.2.1) parcels, were unmatched. In this urban dataset, inspection of some of the unmatched parcels suggested that most of these are undetected subdivisions.

Although there was no objective way to check the correctness of parcel matches where the dataset had no existing UIDs, at a later stage of the research, after the point matching research was operating, it was possible to check whether the areas of matched parcels still matched after the adjustment. The results of this test have been displayed in Table 9.2.

4.5 The benefits of block matching

The block and parcel matching research described in this chapter has shown that parcel matching between two cadastral versions issued on different dates can be straightforward in most urban areas where there is little apparent movement between cadastral versions and corresponding parcels overlap to a very large degree. In rural areas, however, the apparent parcel movement can be highly variable even in the urban areas of a mainly rural dataset. In the urban areas showing large apparent movement such as shown in Figure 4.13 from LGA11, it was observed that the parcels in a single block all shared a similar apparent direction and distance of movement whereas the blocks appeared to have moved in different directions. The conclusion was drawn, therefore, that a preliminary block matching process would guide the parcel matching process to an improved result, particularly where the initial analysis showed a larger range of apparent parcel movements.

In the research conducted into block matching, it was discovered that identification of roads and creeks and the creation of separate blocks from these improved the block matching results. This is because it was found that roads and creeks were sometimes polygonised and sometimes not in different cadastral versions and that there was no consistency between versions. This step ensured that a block of parcels was more likely to be matched to a similar block in the new cadastre.

The block and parcel matching processes have resulted in many correct parcel matches, over 90% in all LGA datasets with UIDs available for validation, and few obviously or identifiably wrong results in datasets without UIDs; the preliminary block matching process has proved essential in some cases, particularly in urban areas where there is a large apparent movement of the block and many parcels are of a similar size and shape.

The area shown in Figure 4.22, where block movement was particularly large, was extracted into a test dataset, DA28. This dataset has been used frequently in developing the parcel matching algorithms; it is an area which initially proved particularly difficult to match. To demonstrate the value of the block matching component of the solution, the parcel matching process has been executed without the initial block matching. Figure 4.37 shows the result of matching the parcels without the use of the block movement information. Large numbers of parcels which should have been matched according to their existing UIDs remain unmatched. The inset shows the results from matching the parcels when block movement was available, using the algorithm detailed in Section 4.3.5; the three unmatched parcels in the inset are irregular parcels – the algorithm does not attempt to match these.

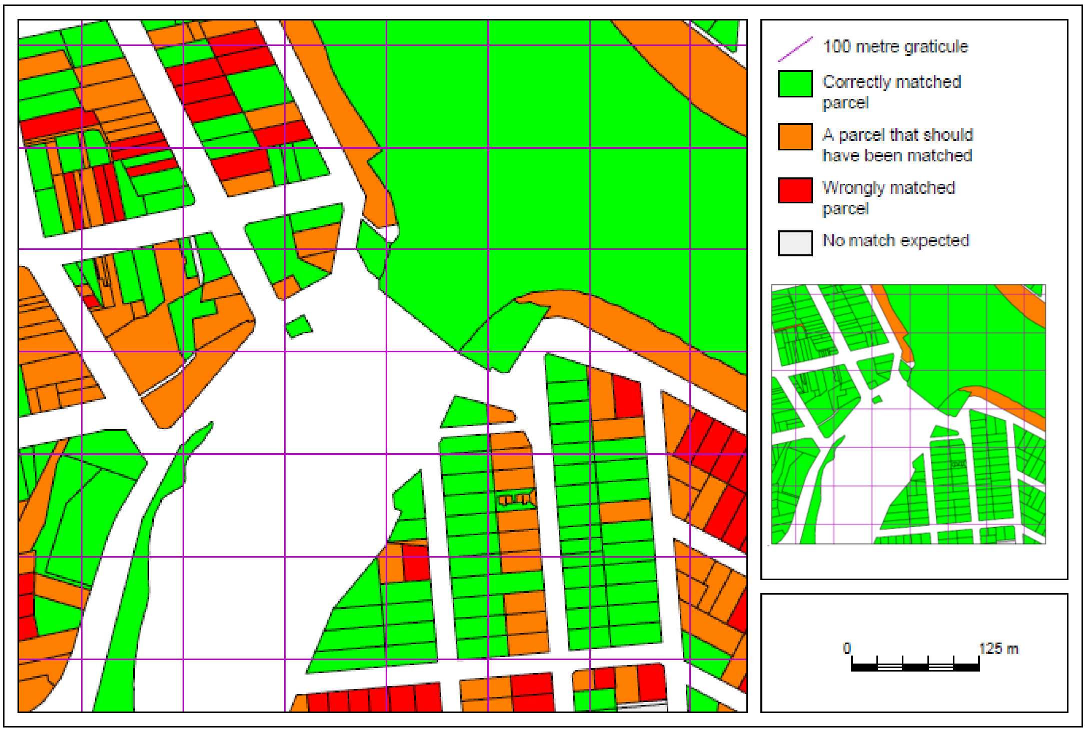

Figure 4.37 Parcel matches with and without block matching

4.6 Summary

The decision to use only the spatial characteristics of a parcel and its underlying block has been vindicated by the number of correct parcel matches achieved. The number of correct matches achieved for the all the parcels from the four LGA datasets where UIDS were available for validation, over 96%, (see Table 4.4), compares favourably with the number reported by Kim et al. (2018), 87.6%, which was achieved using areal Hausdorff distance combined with a turning-function distance and other spatial attributes. The polygons in the dataset used by Kim et al. (2018) represented building-footprints so it is not clear how well their methods would generalise to cadastral parcels, given the large number of identically shaped parcels in urban areas of cadastral datasets.

This chapter has described in detail the methods and algorithms used to achieve correct parcel matching – over 96% where the correctness of the match could be established from existing UIDs (see Table 4.4). For the datasets without UIDs, almost 96% have been matched although there was no objective way to check the correctness of those matches. (see Table 4.5) at the parcel matching stage of the research. Once the point matching algorithms were completed, it was possible to check that matched-parcel areas were close to the new-parcel areas after adjustment of the old cadastre. The results from this check are illustrated in Section 9.3.4.

The chapter has also described how the parcel matching methods were researched and implemented and how substantially correct parcel conflation between two cadastres was achieved using several shape and size measures to control the matching process. When it became clear that substantially correct match results were being obtained, the decision was taken to discontinue the GA approach. This was because, once effective parcel matching had been achieved, it seemed probable that the parcel centroid shift information could be applied to a parcel's constituent vertices to locate the correct matching point from the new cadastre and thus generate the required shift vectors thereby obviating the need for the GA and the Workbench software. Although this assumption would later prove to be correct for most regularly shaped, urban sized parcels, the later stages of the research described in this thesis has shown that the assumption is not always valid for rural parcels and, in particular, for points on riparian boundaries. The techniques developed to handle the more difficult areas is covered in detail in the succeeding chapters.

The next chapter describes the process developed to extract the individual parcel vertices into a point layer and the attributes assigned to each point to aid the point matching process.

5 CREATING THE POINTS LAYERS

The creation of shift vectors for spatial adjustment requires that each point in a source layer be matched, wherever possible, to the corresponding point in a target layer. Although it would be possible to carry out this process whilst reading all the vertices one-by-one from each parcel, this would be inefficient; when the polygons are stored in a shapefile format, multiple points can exist at a single location, for example, at the corner points of adjacent parcels. Attempting to match every vertex could be expected to result in multiple, potentially conflicting, shift vectors at each location. The decision was taken, therefore, to create point layers for each cadastre and eliminate, from the old-cadastre point layer only, all but one point at each unique location.

This approach has the added advantage that attributes concerning the spatial environment of each point can be assigned to each point feature created and the point layer and attributes can be used to ensure that, where more than one point exists at a location, the point most likely to result in a successful match is retained in the duplicate elimination process.

This chapter describes the processes used to create spatial point layers from all the points from both the old and the new cadastres. It further describes the attributes assigned to each point and the way these attributes were used to eliminate all but one point from the old cadastre at each unique location.

The chapter also describes the processes by which it was determined whether a point should be flagged as a node, a corner or neither. Identifying urban parcel corners became desirable when it was realised that matching the parcel boundaries could increase the number of points that could be matched.

5.1 Overview

Section 5.2 describes the way in which spatial point layers were created from the cadastral points, the attributes assigned to each point, and the way in which the attribute values were determined. Section 5.3 describes the way in which irrelevant points, i.e. points that would have no discernible effect on the adjustment results, were ignored when creating the point layers. Section 5.4 discusses the reasons why it was found to be desirable to identify urban parcel corners and the reasons why it was deemed necessary to distinguish riparian boundaries from other parcel boundaries before doing so. Section 5.5 describes the research carried out to distinguish riparian boundaries from other cadastral parcel boundaries and the rules finally defined to achieve the distinction.

Section 5.6 details the rules developed for identifying corners using the spatial characteristics of the point and the boundary type – regular or riparian. Section 5.7 describes the methodology used to eliminate all but one old cadastral point at each unique location and Section 5.8 describes the way in which the entire point creation process was reversed to permit the creation of reverse shift vectors. Section 5.8 summarises the lessons learned from this stage of the research.

5.2 Creating the points layers

For each parcel in turn from each cadastre in turn, the parcels were processed, firstly to establish that all the vertices on the outer boundary of the polygon were arranged in a clockwise direction and, secondly to transfer the vertices one-by-one to a separate point layer. In the process of creating the points, several attributes were assigned to each one. The number of attributes defined has changed over the course of the research as the need for additional ones became apparent or, in some cases, were no longer needed. (Some of the attributes have also been added for Run-Time Efficiency (RTE) only.) The uses of each attribute will be described in detail in this and later chapters but for clarity in this thesis, the final complete set is described below:

(a) The area of the current parcel (RTE).

(b) The BlockID (see Section 4.3.2.3) of the current parcel's block (RTE).

(c) Whether the point is from a road polygon (RTE).

(d) The shift vector length and azimuth from the current parcel attributes (RTE).

(e) The parcel Feature Identification Number (FID).

(f) The polygon part number and the point number within the part (used for software debugging purposes rather than the final matching process).

(g) Whether the point falls on a node. To establish whether a point is a node, a small search tolerance was used because of the lack of exact coincidence of points at nodes in a shapefile.

(h) The type of boundary on which the point falls, i.e. whether the point falls on the boundary between two matched parcels, two unmatched parcels or one of each.

This attribute was not recorded for nodes; points at nodes, by definition, touch three or more parcels which may each be matched or unmatched.

(i) The acute angle at the point (segment angle), i.e. the acute angle between the line segments joining the preceding and succeeding points to the current point.

(j) The clockwise turning angle at the point (the turn angle) – as above but the clockwise angle between the line segments rather than the absolute value of the acute angle.

(k) The azimuth of a line bisecting the acute angle at the point (the bisector angle – BA). Where the turn angle at the point is close to 180° the bisector angle was computed by adding 90° to the segment angle.

(l) Whether the point is a corner (see section 5.4). Any node with a segment angle of less than 115° is also defined as a corner.

(m) Whether the point is a salient point, i.e. any point having a segment angle of less than 170°. All nodes and corners are automatically defined as salient points.

(n) The sequence number of the boundary in the current parcel. A boundary is defined as the line joining two nodes, two corners or a node and a corner

(o) The distance of the point along the current boundary, computed after eliminating straight-line points (see Section 5.3).

(p) The percentage distance of the point along the current boundary computed as above.

(q) The FID of the boundary line created from this and adjacent points. This was assigned a value when the boundary lines were created rather than at the time the points were created (see Chapter 6).

The threshold values used to arrive at the above Boolean values such as point saliency and corner identification were arrived at by repeated testing on the difficult-area datasets until good point matching and shift generation results were obtained.

The identification of salient points was important because the angle between the segments is sufficiently distinct from a point on an almost straight boundary that it can be used to limit possible point matches in the new cadastre.

5.3 Identifying irrelevant points

Points that are not isolated points on a boundary, and do not fall on the world polygon, and have a close to 180° segment angle, even where they can be matched, have no significant effect on the rubber-sheeting-tool results. Such points were, therefore, not included in the point matching process; initially, these points were just flagged as “no match expected”. At a late stage in the research, however, it was discovered that removing these points altogether from the point layers rather than just flagging them would improve point matching results still further in some cases. The reasons for this will be discussed in Section 7.7.4.

During this research, it was observed that poor data quality adversely affected the shift vector generation process. In particular, where two points are almost, but not quite, coincident with each other, the computation of the bisector azimuth BA can produce spurious results.

Figure 5.1 shows a highly zoomed in screen shot of two points on a boundary where the distance between them is only a fraction of a millimetre. It was found necessary to eliminate such points from the matching process because the computed bisector azimuth at these points is not valid, i.e. it cannot be expected that a point with a similar bisector azimuth would be discovered in the upgraded cadastre layer. Although this elimination of extremely close points resulted in some improvement of results, it became apparent that the algorithms would deliver an even better result, i.e. one where the operator would be presented with fewer areas to check and correct, if the datasets were to be spatially cleaned with GIS editing software before attempting to create shift vectors. For this reason, the one dataset exhibiting this problem (LGA08) was pre-processed using the ArcGIS snapping tools before running the shift vector generation code (see Chapter 9 for further discussion on this subject.)

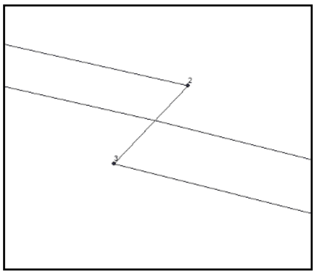

Figure 5.1 Almost coincident points on a boundary.

5.4 Identifying corners

The need for the identification of corners, i.e. locations on a parcel boundary where the segment angle is close to 90 °, arose because it was realised, during the later stages of the research, that many points falling at intermediate locations along a boundary between corners would be matched more accurately if the boundaries they fall on could be matched. In this context, a “boundary” has been defined as a section of a polygon perimeter between nodes or corners. Corner identification, therefore, became necessary to facilitate the creation of the boundary layer described in the next chapter and increase the number of boundaries that could potentially be matched.

Although a human operator can look at a digital map and state with a degree of certainty whether a point falls on a corner or not, automating their identification was not simple. Clearly, any point at a node which also has a significant segment angle is a corner, but intermediate corners are not so easy to define. The algorithm finally used underwent several refinements until the version described here was arrived at.

Initially, it was thought that corners could be identified by their segment angle and by inspecting the preceding and following vertex on the polygon to determine whether they themselves were corners. This works well for an urban parcel on a street corner where only four points have been used to define a more-or-less rectangular parcel. For example, in Figure 5.2 from LGA01, this simple algorithm would correctly identify the corner at the northeast of the highlighted parcel because the adjacent two vertices are nodes and also corners. However, identifying corners on rectangular parcels with just four vertices is not important in this context as there are no intermediate vertices to be matched; moreover, the algorithm fails when there are intermediate vertices between the nodes and the corner in question. Intermediate vertices can sometimes be included on straight-line boundaries where the map has been digitised from separate map sheets. The horizontal row of points near the centre of Figure 5.3 fromLGA01 is an example. The algorithm was therefore modified to ignore any vertex having a segment angle of close to 180° when seeking adjacent corners; this solved some problems but not all.

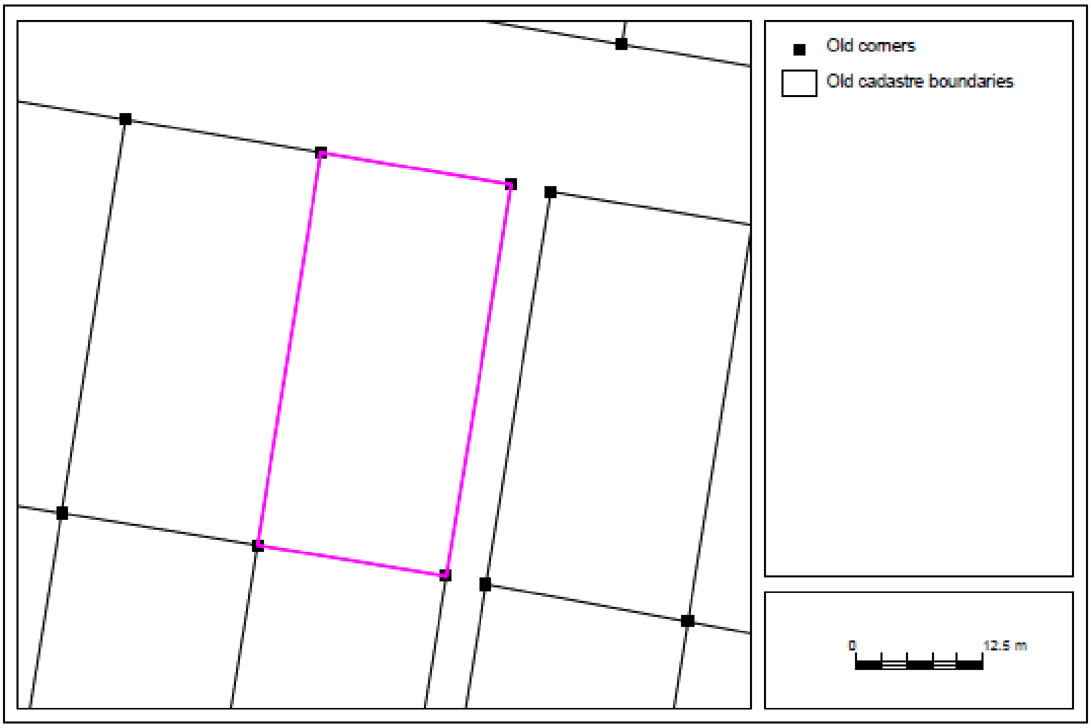

Figure 5.2 Easily identifiable corners

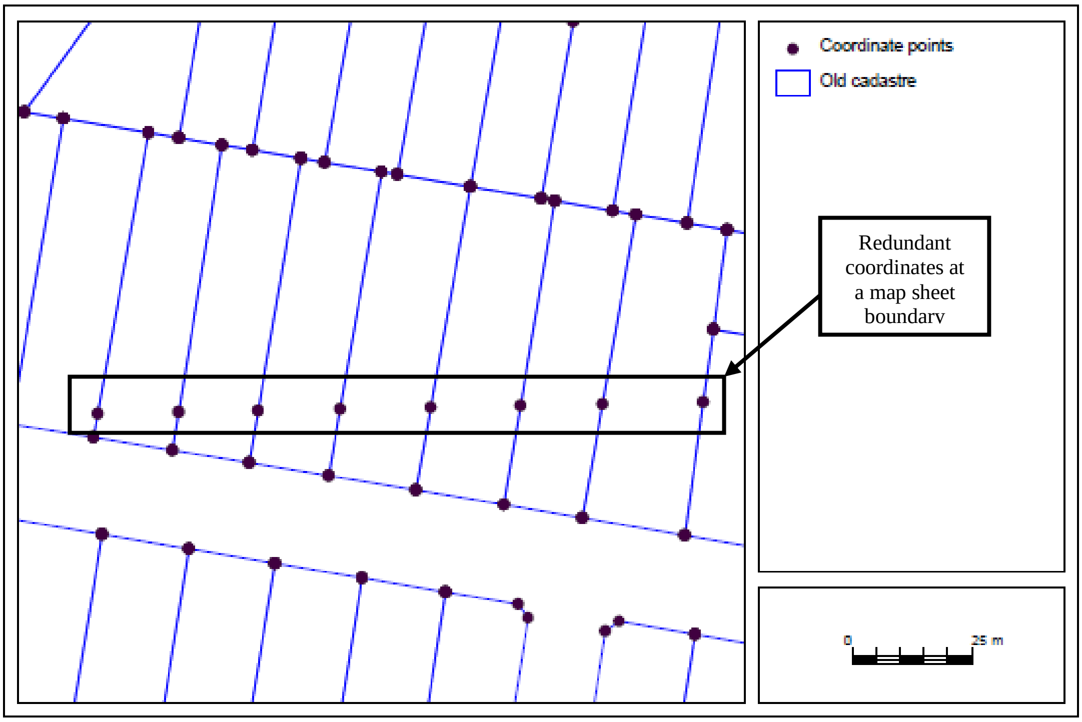

Figure 5.3 Redundant vertices along a map sheet boundary

Intermediate vertices between nodes or true corners can occur for many reasons. Typically riparian boundaries and road easements in rural areas will have many defining vertices but intermediate vertices can also occur in many other situations, for example, the map sheet boundary locations shown in Figure 5.3; other redundant vertices created during the digitisation process; truncated parcel corners as shown near the southern edge of Figure 5.3; curved road frontages on urban parcels; and complex parcel shapes such as shown in Figure 5.5 from an urban area of LGA11.

It became apparent during the point matching stage of the research that it would be necessary to distinguish riparian boundaries from regular boundaries so that vertices on those boundaries would not be defined as corners by the automated process. Figure 5.4 from LGA11 shows a riparian boundary where several vertices were wrongly identified as corners (small black squares) before the corner identification algorithm was amended to exclude such boundaries; in this case, there were no corresponding corners identified in the new cadastre so that the later point matching process produced incorrect results. It was therefore deemed necessary to automatically identify riparian boundaries prior to corner identification.

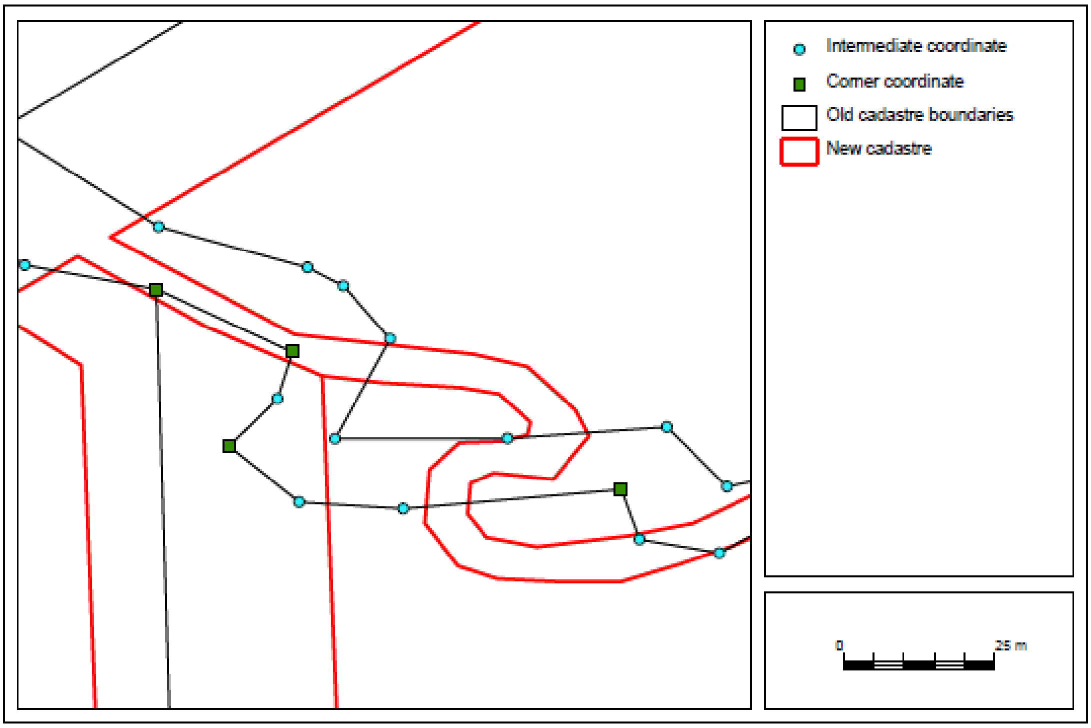

Figure 5.4 Vertices wrongly identified as corners on a riparian boundary

5.5 A rule for identifying riparian boundaries

The rules for identifying riparian boundaries were arrived out by formulating hypotheses based on intuitions arising from inspection of the data. This section describes the way in which the rule eventually selected was arrived at after testing each rule variation.

The various approaches considered were:

(a) Define a riparian boundary as any boundary between nodes where there are seven or more vertices defining the boundary. The number seven was chosen because an initial inspection had suggested that boundaries with more defining vertices than this were usually riparian.

The results from applying this rule were thematically mapped and inspected by the author. The inspection revealed that the rule correctly identified some riparian boundaries but also wrongly identified others as riparian in places where examination of the results indicated that they were not. Figure 5.5 shows an example of an urban parcel (outlined in magenta) where there are more than seven points on one boundary. Inspection of these wrong identifications suggested that the length of the line between nodes on riparian boundaries is typically much greater than the Euclidian distance between the same nodes. Rule (a) was therefore amended as follows.

(b) A riparian boundary is a boundary between nodes where there are seven or more vertices defining the boundary and the length of the boundary is 50% greater than the Euclidian distance between the nodes.

This rule produced satisfactory results in rural areas but wrongly identified some urban parcels as riparian where inspection again showed that they were not. This wrong identification can typically arise on urban parcels at street corners as illustrated in Figure 5.5. The north-western boundary of the central parcel has eight vertices, including a redundant one close to the southwest corner, and its length is more than 150% of the Euclidian distance between the nodes at each end of the boundary. If this boundary and the one on the parcel to the south were processed using rule (b) they would be wrongly identified as riparian boundaries, i.e. boundaries where most points should not be treated as corners. Inspection of the more southerly of the two parcels shaded in magenta in a GIS viewer clearly showed that there are seven true corners that are not also nodes on this parcel.

The algorithm also wrongly identifies all island parcels as riparian because the Euclidian distance between the first and last vertex on such parcels is zero.

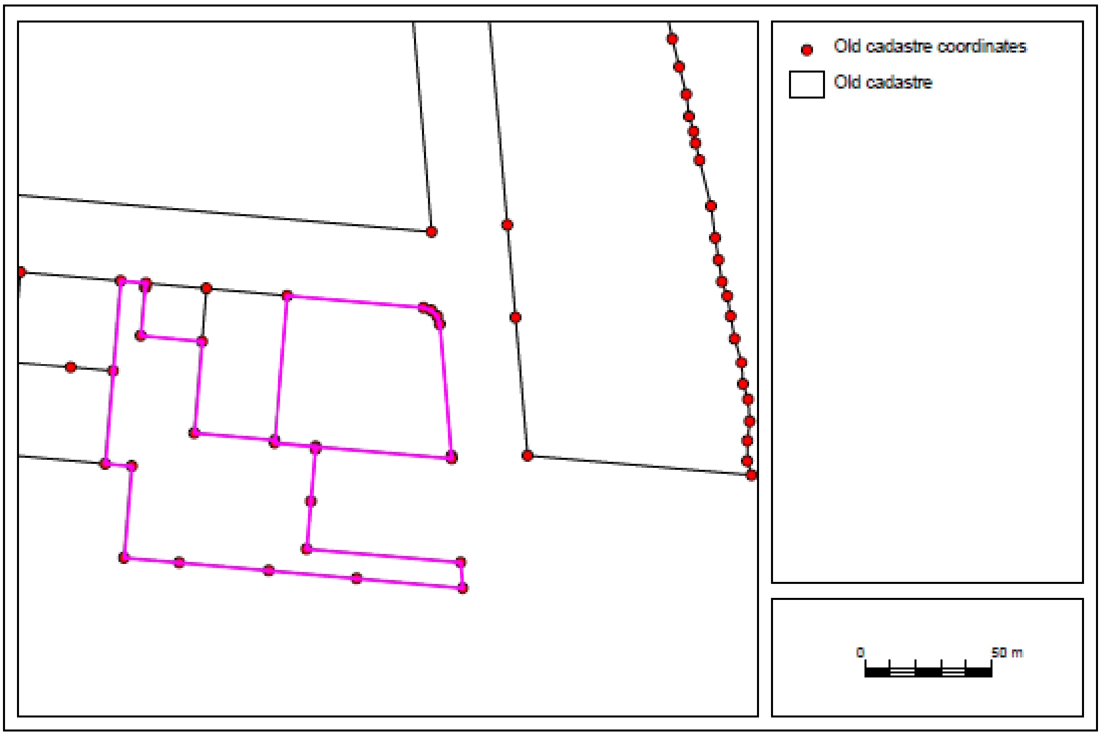

Figure 5.5 Two parcels (outlined in purple) each with many points on the boundary between two nodes

Rule (b) was therefore further amended to distinguish between true riparian boundaries and urban parcel boundaries defined by several vertices. From observation, it appeared that the principal difference between true rural parcels with riparian boundaries and urban parcels with many vertices is that urban parcels tend to have a larger proportion of segment angles that are corner angles. For example, in the more southerly of the two highlighted parcels in Figure 5.5 from LGA11, slightly more than 50% of the vertices on the long boundary have a segment angle of close to 90°.

In addition, it was observed that riparian boundaries are typically defined by vertices that are more densely clustered along the boundary than on non-riparian boundaries.

The final rule implemented for the detection of a riparian boundary, i.e. a section of the parcel perimeter between detected nodes, is as follows:

(a) Obtain the current-parcel perimeter (*PP*).

(b) Count the number of nodes on the parcel perimeter (*NCt*).

Then for each detected node on the parcel

(c) Obtain the length of the boundary between the current node and the subsequent node (*BL*).

(d) Find the number of points between the nodes (*CCt*).

(e) Calculate the Mean Point Distance (*MPD*) between points on the boundary:

$$MPD = BL/CCt.$$

(f) Calculate the Euclidian Node Distance between this node and the next one (*END*).

(g) Count the number of potential corners on the boundary, i.e. the number of points where the segment angle is less than or equal to 115° (*PCCt*).

(h) Then

WHERE

$CCt > 4$

AND

$PCCt \;/\; CCt * 100 < 75$

AND

$(MPD < \; 50 \text{ m OR } MPD \,/\; BL * 100 < 2)$

IF *END* = 0 (i.e. the current boundary is the perimeter of an island), the boundary is riparian if

$MPD \,/\, PP * 100 < 2$

ELSE (the boundary is not an island boundary) the boundary is riparian if

$MPD \,/\, PP * 100 < 2 \text{ OR } (NCt > 2$

AND

$BL > END * 1.5).$

This more complex rule has been developed as the result of tests on several different datasets, both urban and rural. The final value of the thresholds in the expressions and the form of the rule was arrived at by inspecting thematically mapped results and modifying the algorithms until the results were satisfactory in the subjective opinion of the author.

It should be noted that, to produce as few incorrect shift vectors as possible using the point matching algorithms described in Chapter 7, it is preferable that sometimes a real corner is not identified as such rather than that vertices on riparian boundaries are wrongly identified as corners. This is because corners have been treated differently from other points at the point matching stage described in Chapter 7.

5.6 Identifying corners, part two

During the development of the corner detection algorithm, it became clear that it would be necessary to compute a "corner distance" (CD) value taking the current spatial environment into account.

After developing the rule for the identification of riparian boundaries described in the previous section it was possible to refine the corner identification rule as follows:

> IF the boundary is riparian then set the corner distance (CD) to three times the mean distance between the vertices on that boundary
>
> ELSE set the corner distance to zero

Corners were thereafter defined as follows: a polygon vertex is a corner if

$$SA \leq 115$$

AND

$$FD > CD$$

AND

$$TD > CD$$

where *SA* is the segment angle, *FD* is the distance from the previous vertex in the polygon and *TD* is the distance to the next vertex in the polygon. In other words, a corner on a non-riparian boundary can be close to an adjacent corner but a vertex will only be identified as a corner on a riparian boundary if it lies at a significant distance from its nearest neighbour.

These expressions and threshold values have been arrived at by experiments conducted on the difficult-area datasets that each included some riparian boundaries. For each experiment, the expressions and/or threshold values were modified until inspection of the resulting old-cadastre adjustment showed that the results were satisfactory to the author, i.e. the algorithm correctly identified most urban parcel corners but did not identify sharp turns on riparian boundaries as corners.

Note on identifying riparian boundaries: At the stage of the research where this algorithm was developed, it was not yet appreciated that an assignment of boundary type (road casement or true riparian – see Section 6.5) to the boundary layer would be useful. Additionally, boundaries cannot be created until corners are identified because a boundary has been defined, in this research, as a vector starting and ending on a node **or** a corner. In other words, the boundaries cannot be created until the corners have been established, yet the corners cannot be identified until a boundary is identified as riparian or otherwise.

The assignment of a boundary type once the boundary linear spatial layers have been created is described in Section 6.5. The algorithm used there could not be used instead of the one described in Section 5.5 as it only handles boundaries with at least 20 vertices but, for the purpose of identifying corners, it was found to be important to recognise riparian boundaries with fewer points at the point-layer creation stage discussed in this chapter.

5.7 Eliminating duplicate old cadastral points

The point layer created from the old-cadastre parcel vertices was processed to remove all but one point from each unique location, retaining just the points at each location whose attributes best constrained the point matching process described in Chapter 7.

Although eliminating all but one point at each location would obviously improve run-time efficiency at the point matching stage, the principal reason for implementing this process is that, in the shapefile and similar formats, the points at an individual location may have quite different spatial characteristics. It is possible for there to be three points at a junction between three parcels where two of the points have close to a 90° turn angle and the third has an approximately 180° turn angle because it falls on a straight-line boundary of its parcel; Table 5.1 lists the attributes of the points created at one such location. Other characteristics of a point also vary, such as its distance from its neighbours, the type of boundary it falls on and many others.

The elimination process was achieved by assigning a unique location number to all the points created at a single location and then sorting the points using their attributes in descending order of importance. The first point at each unique location in the sorted list is retained; all other points created from the old cadastre layer are deleted.

In effect, the order chosen results in a set of points such that there will be the least amount of ambiguity when attempting to match a point. The following list shows the attributes used to sort the points in order of importance and the reasons why each attribute was used:

(a) Location number; all points at the same location share the same location number.

(b) Is a node. Nodes are preferred to all other point types because there is a high probability that there will be a corresponding node on the new cadastre and the azimuth of the bisector angle and the turning angle at the point constrain the match with a high degree of accuracy.

(c) Is a corner. Corners are preferred to non-corners. Points that have been identified as originating on parcel corners are preferred for the same reasons that nodes are preferred; the bisector azimuth and the turn angle can be used to constrain the match.

(d) Parcel type. Points from regular parcels are preferred to irregular parcels, roads, and slivers. Regular parcels typically comprise fewer vertices than other parcel types. (In the shapefile format, where a regular parcel is adjacent to an irregular parcel, the individual vertices are duplicated along the common boundary.)

(e) Whether the point is a salient one, i.e. whether the segment angle is less than or equal to 170°. The more acute the angle at the point, the more likely it is that it can be matched to a new point with a similar angle.

(f) Parcel match type. Where UIDs are present, matched parcels are preferred to others in the order: correctly matched, unexpectedly matched, should be matched, wrongly matched, and unmatched. Where no UIDs are present, matched parcels are preferred to unmatched parcels. For matched parcel pairs have matching UIDS the UID are used to limit the set of potential matching points from the new cadastre.

(g) Whether the boundaries have been matched. Points from matched boundaries are preferred. Where points fall on a matched boundary, potential matches can be constrained to those falling on the boundary with same boundary ID.

(h) The parcel area. Points from smaller parcels are preferred. Smaller parcels tend to exhibit a smaller apparent movement used to constrain the search area. Also, the area of the parcel on which a potential match falls can be compared to the area of the originating parcel and the comparison used to refine the matching process.

(i) The angle between line segments at the point. Points on the more acute angles are preferred, as mentioned above.

(j) The number of points on the boundary. Points from boundaries with fewer vertices are preferred.

(k) The boundary type. Points on a boundary between two matched parcels are preferred.

(l) The distance along the boundary, shorter distances are preferred.

The order was determined by altering the order of priority of the attributes and then executing test runs on the difficult-area datasets. A candidate precedence was adopted when the results of adjusting the old-cadastre using the generated shift vectors were satisfactory to the author. The candidate solution was then tested on all the full LGA datasets; only when these were also satisfactory was the candidate solution adopted.

Table 5.1 illustrates a few of the differing attributes of three points from a unique location (point “a” in Figure 5.6 below). Figure 5.6 from LGA07 illustrates the old-cadastre points remaining in a small area after the elimination process has taken place.

Table 5.1 Attributes of three points at a single location

AREA m^2	Is a node	Is a corner	From distance	To distance	Bisector azimuth	Turn angle
1,271	TRUE	TRUE	20.19	63.10	-111.90	90.17
1,213	TRUE	TRUE	63.10	19.46	-20.78	92.06
53,580	TRUE	FALSE	19.46	20.19	114.13	177.77

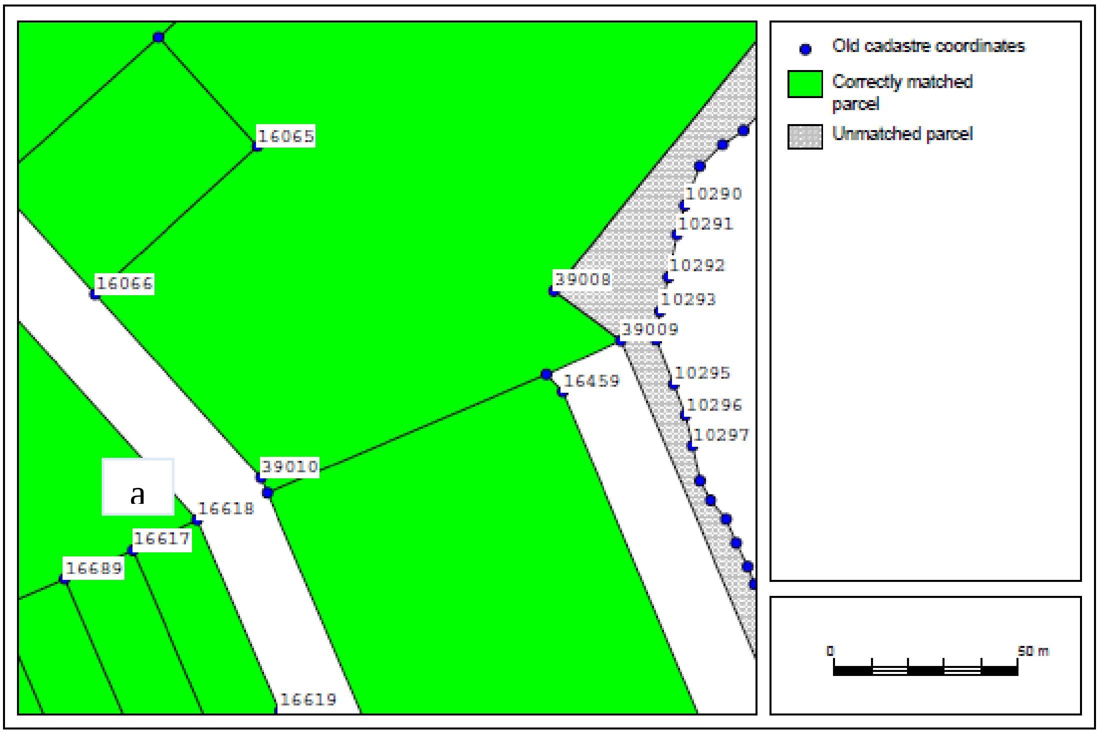

Figure 5.6 Different point types in a small area

In this example, the second point in Table 5.1 (highlighted) was selected for retention by the elimination process because it fell on a smaller parcel than the first point in the list. Any points not selected for retention are deleted from the old-cadastre point layer before commencing the point matching process.

Table 5.2 shows some of the attributes used in the matching process for the points shown in Figure 5.6. Points are labelled with the identifier shown in the left-hand column of the table. Point 16617, highlighted in the table, is the only point remaining at location “a” after the duplicate elimination process.

Table 5.2 Some point attributes

Point FID	Is a corner	Is a node	Bisector azimuth	From distance	To distance	Turn angle
10290	FALSE	FALSE	-17.97	11.44	8.21	-173.41
10291	FALSE	FALSE	-13.03	8.21	12.01	-176.71
10292	FALSE	FALSE	167.47	12.01	9.52	177.71
10293	FALSE	FALSE	-10.47	9.52	7.81	-173.61
10294	FALSE	FALSE	7.03	7.81	13.09	-151.39
10295	FALSE	FALSE	-159.00	13.09	9.06	179.34
10296	FALSE	FALSE	-163.23	9.06	8.48	172.20
10297	FALSE	FALSE	-167.55	8.48	9.93	179.16
10298	FALSE	FALSE	20.44	9.93	6.07	-163.19
10299	FALSE	FALSE	34.46	6.07	6.66	-168.77
10300	FALSE	FALSE	-148.49	6.66	7.08	162.87
10301	FALSE	FALSE	24.41	7.08	7.12	-177.06
10302	FALSE	FALSE	-156.87	7.12	5.31	174.50
10303	FALSE	FALSE	24.26	5.31	13.08	-172.23
16065	TRUE	FALSE	177.34	0.00	59.99	90.25
16066	TRUE	TRUE	87.27	0.00	40.93	89.61
16458	TRUE	TRUE	-102.00	0.00	6.51	110.03
16459	FALSE	FALSE	-147.13	6.51	114.02	159.73
16465	TRUE	TRUE	-21.98	0.00	82.99	89.92
16617	TRUE	TRUE	-20.78	0.00	19.46	92.06
16618	TRUE	TRUE	-110.76	0.00	57.13	87.99
16689	TRUE	TRUE	-111.86	0.00	63.11	90.24
39008	TRUE	FALSE	7.34	0.00	22.75	-87.28
39009	TRUE	TRUE	174.18	0.00	22.40	60.97

5.8 Reversing the point creation process

To allow the creation of reverse shift vectors to be used for checking the correctness of the results (see Section 8.3.6) the entire process described in this chapter was repeated using the vertices from the new cadastre so that they could be used as source points for the point matching process described in Chapter 7. In each case, the source point layer has only one point at each location but the target layer holds all the points originating from all the vertices.

5.9 Summary

This chapter has described the process of creating point layers from the cadastral parcel vertices and it has detailed the point attributes recorded for each point. It has also described the process adopted for the identification of probable corners on parcel boundaries and the reasons why it became necessary to first discover whether a boundary was a complex riparian boundary or just a normal parcel boundary with a substantial number of vertices. Observations include:

(a) The process of creating a point layer from all the vertices from a polygon layer is a straightforward spatial operation. However, the shapefile and similar formats now commonly adopted by GIS software vendors do not lend themselves to the efficient identification of nodes unlike the topological format of the old Esri Arc/INFO coverages or the VPF format mention in Section 2.3.

(b) Determining whether a given point should be modelled as a corner has proved to be complex and necessitates the determination of the type of parcel boundary, riparian or other, on which the point falls.

(c) Distinguishing between riparian and non-riparian boundaries has also proved to be complex.

Most of the processes described in this chapter were developed as the result of insights gained during research into methods for matching old-cadastre points to other points, or locations, in the new cadastre layer. The algorithms evolved over the period of the research. The research has been described here for reasons of clarity in describing the several stages of data processing needed to facilitate the point matching process.

The next chapter will describe the reasons for creating a boundary line layer from the point layers and the process used to create a boundary line layer from the point layer. The algorithm used to find matching boundaries between the layers are also described as is the algorithm used to assign a boundary type to each boundary, i.e. regular, rural road, or riparian.

6 BOUNDARY CREATION AND MATCHING

During this research, in the preliminary stages of attempting to match points between the old and new cadastre, it became apparent that the points from some parcel boundaries, points from riparian boundaries, in particular, could be matched more accurately to the upgraded cadastre points if the boundaries had themselves been matched wherever possible.

6.1 Overview

This chapter describes the reasons for creating the boundary line layer and the rules developed to control the matching process. Section 6.2 describes the reasons why the creation of this layer was found to be necessary. Section 6.3 describes the way in which the boundary lines were created and the attributes assigned to each feature. Section 6.4 describes the spatial search used to select a candidate set of matching boundaries and the way in which the best acceptable candidate, if any, was chosen from the list. Section 6.5 discusses the reasons for classifying the boundaries into distinct types and the algorithm used for the classification.

6.2 Rationale for creating a boundary layer

The initial attempts to match points between the two cadastral point layers focussed on methods for finding, for each point from the old cadastre, a single matching point from the upgraded cadastre using the apparent movement of the parcel to determine the expected location of the matching point and the spatial attributes of the point to limit the possible candidate points for the match.

In many cases, the expected locations – the target points – fall close to the correct matching point. Figure 6.1 from LGA07 shows an area where this is true for many of the points; the expected locations are marked by blue triangles. However, on the complex boundary near the centre of the map, there are no new-cadastre points (red dots) to which to match, perhaps because a surveyor or GIS data capture operator considered that section of the parcel boundary to be sufficiently straight that it did not need additional vertices.

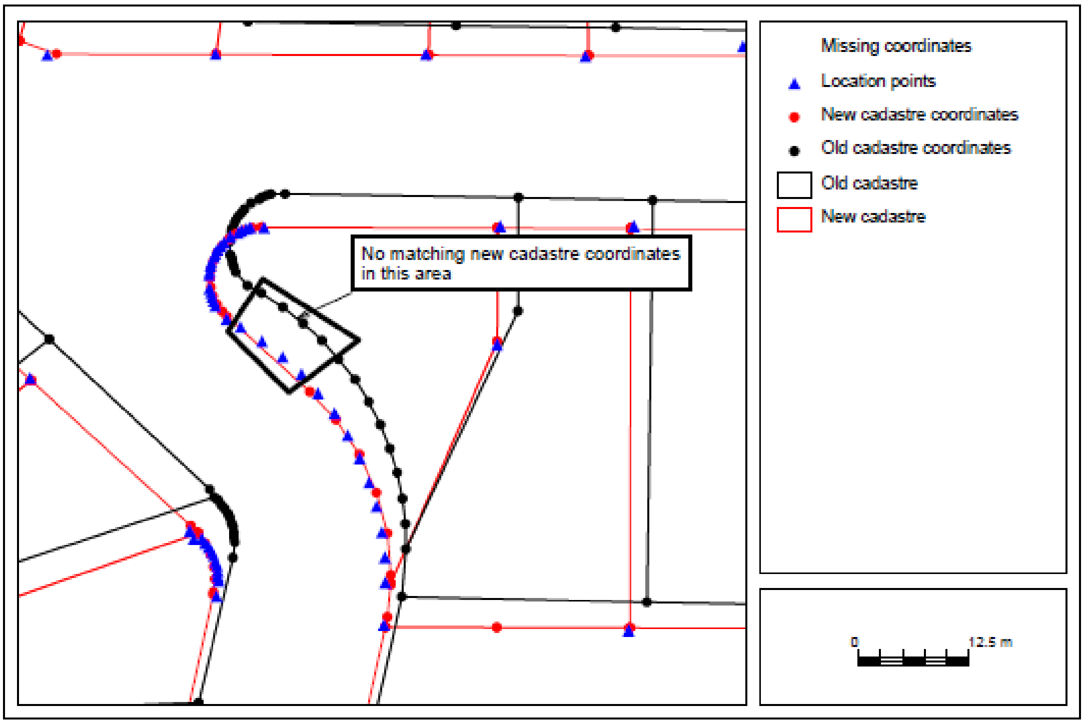

Figure 6.1 A boundary with no matching points

An additional problem arises from the movement of riparian boundaries that may be due to real movement of creek or river banks, or due to coastal erosion or accretion, or to real or apparent movement due to re-digitization from a higher resolution orthophoto. In the case of real movements, the expected location of a matching point may be a long distance from any of the new-cadastre points. Figure 6.2 from LGA11 shows just such an area along some creek banks. The target points (indicated by the blue triangles) along the creek banks do not coincide with the corresponding bank in the new cadastre although the computed locations of the target points on the non-riparian boundaries are accurate. This map also reveals that on a riparian boundary there may be no one-to-one correspondence between the old cadastre points (shown by the black dots) and the new points (shown by the red dots). Such a correspondence is not to be expected, as different surveyors or GIS operators would be unlikely to choose the same locations along the creek bank to record their data points.

For these reasons, it was determined that a different point matching algorithm should be used for points on complex boundaries. In these cases, it was decided that no attempt would be made to match the individual old points to corresponding ones in the new cadastre. Instead, where boundaries could be matched, points would be matched to locations on the matching boundary by proportion along the length, i.e. if the distance of a point along a matched boundary as a percentage of the length of the boundary is known, then an appropriate location for the other end of a generated shift vector can be computed at the same percentage distance along the matching new-cadastre boundary.

.

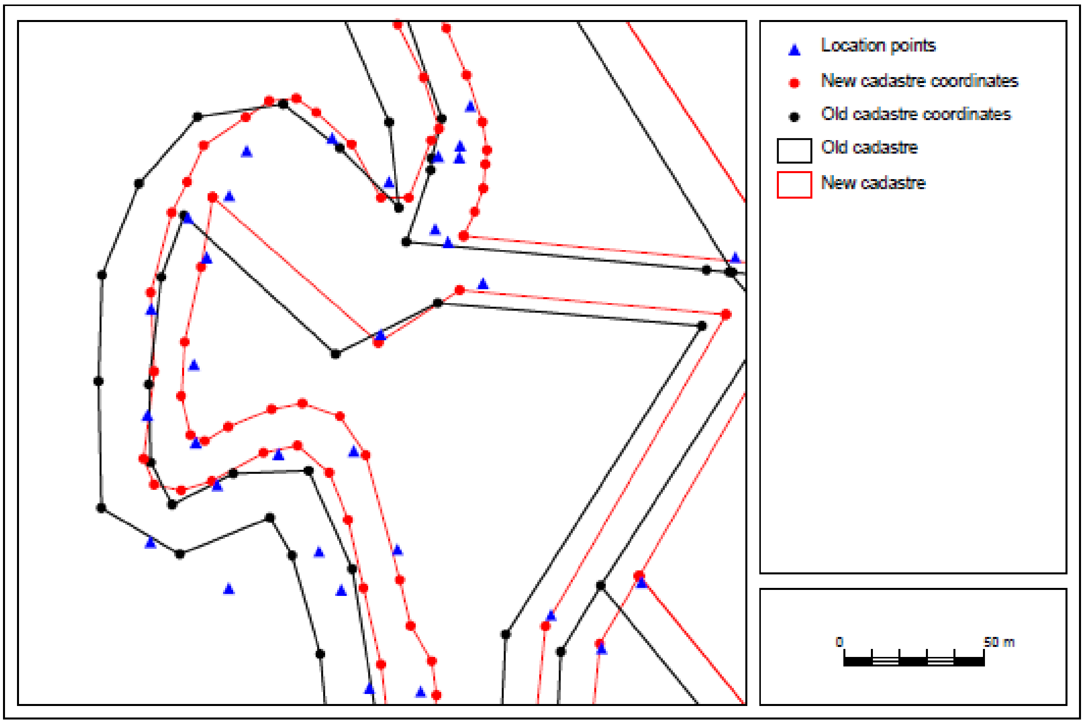

Figure 6.2 A riparian boundary with little correspondence between old and new points

6.3 Creating a line layer representing the parcel boundaries

The boundary line layers were created from the point layers described in Chapter 5. Each boundary line was created by processing the points in sequential order, beginning and ending each line at a corner or a node identified by the processes described in Section 5.4. The following attributes were stored against each line:

(a) The FID of the parcel from which the point came.

(b) The UID of the parcel from which the point came.

(c) The sequential number of the segment on its parcel.

(d) The azimuth of a line joining the end points of the boundary line.

(e) A Boolean attribute to record whether the boundary should be handled as a riparian boundary using the rules finally defined in Section 5.5.

(f) The type of boundary on which the boundary falls, i.e. between two matched parcels, two unmatched parcels or one matched and one unmatched.

(g) The UID of the block on which the parcel falls.

(h) A UID for the boundary if it is successfully matched to a boundary in the other layer, i.e. a UID assigned to both the old and new matched boundary pair.

(i) A Boolean attribute to record whether the boundary was successfully matched.

6.4 Matching the boundaries between the old and the new cadastre

When matching the boundaries, no attempt was made to match boundaries having only two points because the methods finally used to match the points did not require boundary information when matching nodes and corners. An attempt was made to match all the remaining old-cadastre boundaries as follows (Figure 6.3 illustrates aspects of this solution using a single boundary from a single parcel):

(a) Compute a distance tolerance for the end points of the segments (*distTol*) using the following equation.

$$distTol = \max(25\text{ m}, PSL * 2.5)$$

where *PSL* is the length of the parcel centroid shift vector (green).

(b) Compute a length tolerance for the current boundary (*lenTol*) using the following equation:

$$lenTol = \max(BL / 10, PSL)$$

where *BL* is the length of the boundary line between a1 and b1 and *PSL* is the length of the parcel centroid shift vector. In the illustrated case, *lenTol* is equal to the length of the parcel centroid shift, *PSL*, because *PSL* is longer than one tenth of the boundary length.

(c) Construct a search rectangle from the minimum-bounding rectangle of the current old-cadastre boundary line, increased in size by the value of *distTol* (BR in the illustration – not to scale).

(d) Select all new-cadastre boundaries that intersect the rectangle BR and whose parcel UID matches the UID of the current old cadastral parcel. All unmatched parcels were assigned the single UID value, -1, at the parcel matching stage ensuring that unmatched parcels were also processed where appropriate.

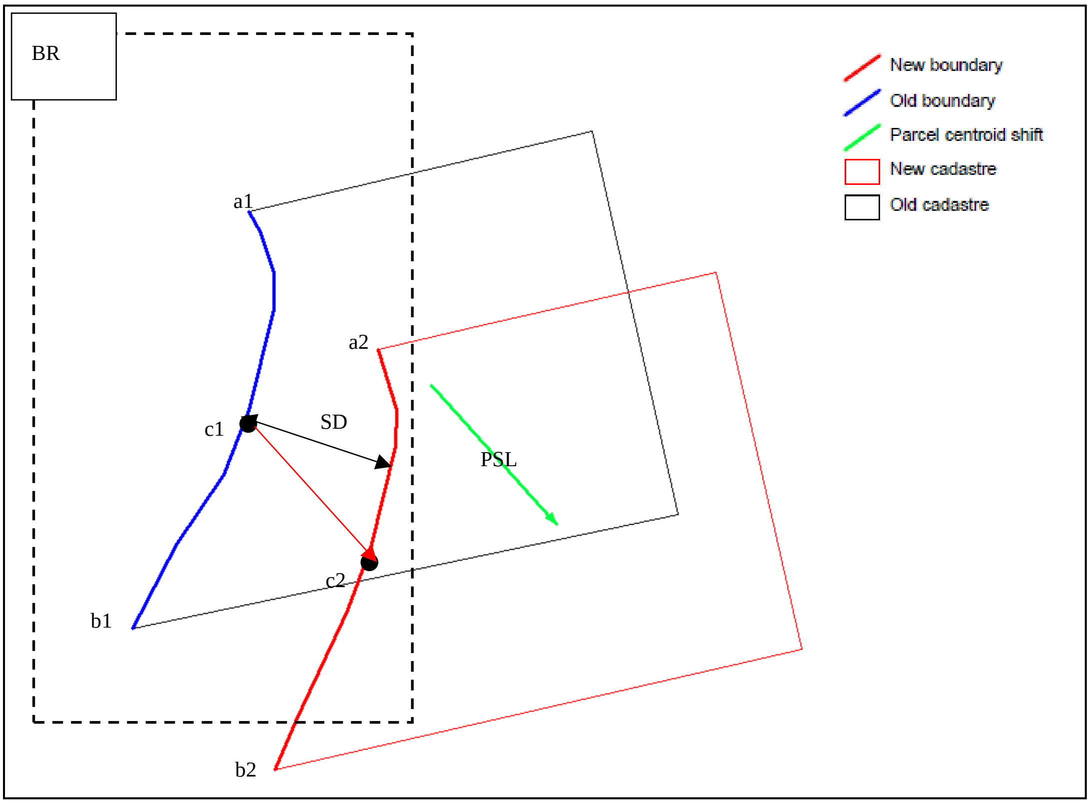

Figure 6.3 Boundary matching illustration

To find the best match from the set of potential matching boundaries, each new boundary located by the search described above was processed as follows:

(a) Compute the shortest distance between the centroid of the old boundary and the candidate new boundary line (SD) – the black double-headed arrow in the illustration.

(b) Compute the difference between the angle of a line connecting a1 and b1 and the angle of a line connecting a2 and b2 (dSA).

(c) For boundaries with five or fewer points, compute the difference between the angle of a line connecting c1 and c2 (the red arrow in the illustration) and the centroid shift angle (dCA). Boundaries with many points are typically riparian boundaries where the value of dCA is unlikely to be of value in matching the boundaries; in these cases, dCA is set to zero.

(d) If the two boundaries are of the same boundary type (BT) and dSA and dCA are each less than 30°:

(i) Compute the difference in length between the two boundaries (*dL*) and the difference in length as a percentage of the longer of the two lengths (*dL%*).

(ii) Compute the distance between the start points of the two boundaries (*distS*).

(iii) Compute the distance between the end points of the two boundaries (*distE*).

(iv) Save the new boundary-match candidate, replacing any previously saved candidate, if the following condition is satisfied:

$$distS < distTol$$

AND

$$distE < distTol$$

AND

$$(dL\% \leq 15 \text{ OR } dL < lenTOL)$$

AND

$$SD < minDist$$

where *minDist* is the smallest value of *SD* so far encountered amongst the set of potentially matching boundaries.

(e) When all the potentially matching boundaries have been processed, if a matched new-cadastre boundary has been found (see item (iv) above), then the two boundaries are assigned matching unique UIDs and flagged as having been matched.

In the absence of any objective way to evaluate boundary matching success, these expressions have been arrived at by iteratively modifying the expressions and altering the threshold values until the results were satisfactory to the author. Because of the method developed to match points from old-cadastre matched boundaries to locations on new-cadastre boundaries (see Section 7.6) it was important that matched boundaries were digitised in the same direction, all parcels were therefore processed at the point creation stage to ensure that their outer boundary vertices were ordered in a clockwise direction.

Figure 6.4 from LGA11 shows an area where most of multi-point boundaries have been successfully matched. Only the matched boundaries have been drawn. They have been drawn over the top of the old and new parcel boundaries (green and orange). The blue lines indicate old boundaries which have been matched to a corresponding boundary (drawn in red) from the new cadastre. Only a few multi-point boundaries remain unmatched, for example, the boundaries of the parcel at the bottom centre of the map.

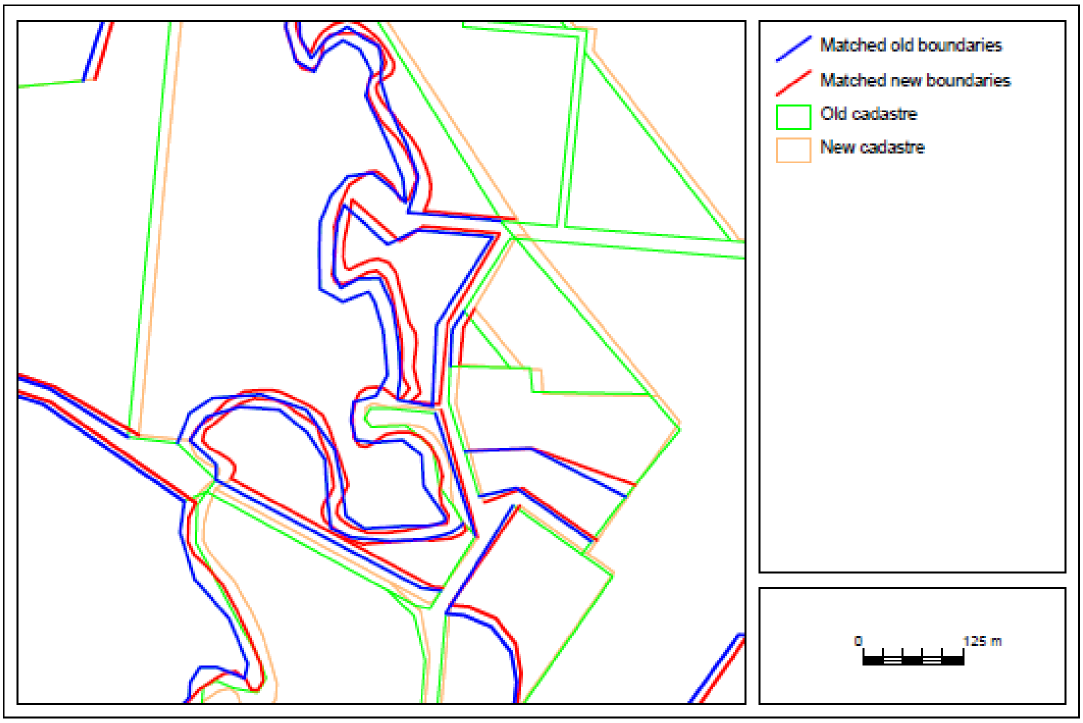

Figure 6.4 Matched parcel boundaries

In practice, it may or may not be important to adjust the riparian boundaries. For example, on a town-planning-scheme map or a local-authority boundary map, the adjustment may well be important where such a boundary follows a coast or river but, on an asset map modelling, for example, point data such as street lights or signposts, adjustment may not be required because these types of assets are seldom located along riparian boundaries. Figure 6.5 from LGA07 shows an example of a planning zone boundary along a creek boundary in an area where there has been considerable real and apparent movement between versions and where correct adjustment of the planning-zone layer would be important.

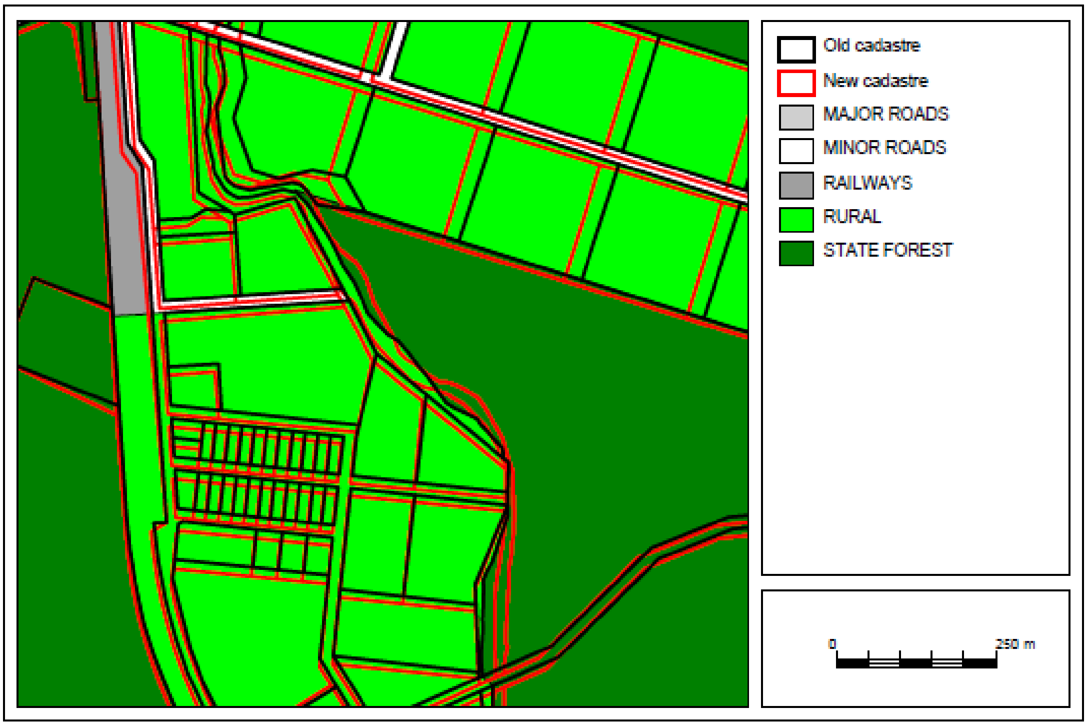

Figure 6.5 A planning zone following a creek boundary

6.5 Research methodology for boundary classification

The cadastral datasets acquired for this research are all vector-based polygon datasets. Boundaries in the datasets are not separately represented, therefore, there are no existing attributes stored against the boundaries. During the point matching stage of the research it was determined that classifying the boundaries into, regular boundaries, rural road boundaries, and riparian boundaries, would assist with the correct identification of matching points in the new cadastre or, in some cases, matching locations in the new cadastre. The boundary classification algorithms were tested by thematically mapping the results in a GIS viewer and making a subjective judgment of the success or otherwise of each version of the algorithm until the results were satisfactory in the opinion of the author.

6.6 Assigning a boundary type

As illustrated in Figure 3.5, the entire research process has been an iterative one. Once the shift-vector generation stage of the research described in the Chapter 7 was reached it was possible to test the results by adjusting the old-cadastre using the ArcGIS *RubbersheetFeatures* tool and to map those results in a GIS viewer to assess the success or otherwise of an algorithm change. Sometimes an inspection resulted in further insights into potential improvements to earlier stages. One such insight was the realisation that further point matching improvements could be made by distinguishing between rural-road boundaries and riparian boundaries such as creeks and coastlines.

Visually, it is often possible on a cadastral map to distinguish river and stream banks or coastlines from road casements with a moderate degree of certainty (except perhaps in mountainous areas where roads may more closely resemble streams).

Figure 6.6 from LGA11 shows, on the left, a small section of what appears to be a country road and, on the right, a small section of what appears to be a stream or river. The old cadastre is drawn in black and the new in red. In the case of the road, the apparent movement of the boundary is probably due to changes in the positional accuracy of the source data but, in the case of the river or stream, the course of the channel has physically changed. This suggested that there would be benefits in using a different process for matching the points depending on the boundary type and raises the question as to how they can be automatically distinguished.

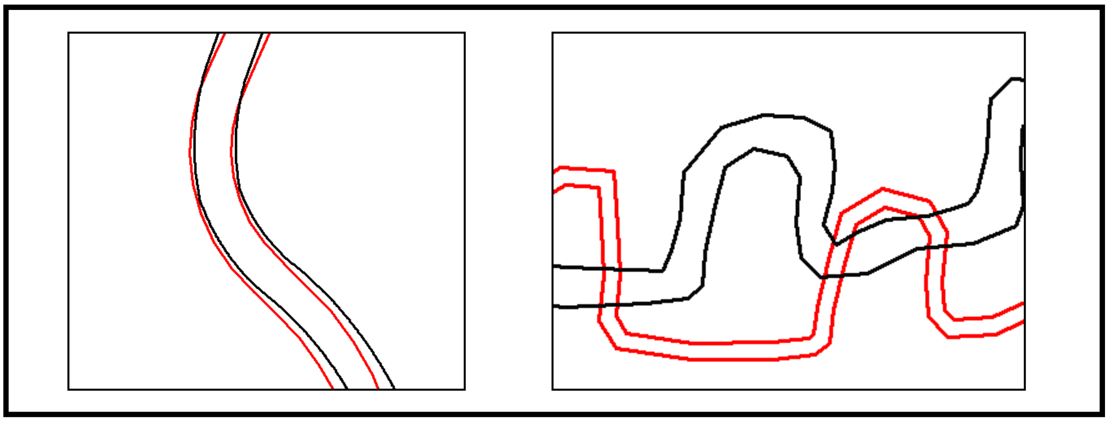

Figure 6.6 A road casement and a river boundary

After inspection of the cadastral datasets using a GIS view, it was decided that an attempt would be made to classify the boundaries into three types, regular boundaries, road boundaries, and riparian boundaries so that different strategies could be applied to the process of matching the points on those boundaries. The inspections suggested that adjustments along roads would be improved by matching a point at a place of significant direction change, i.e. a corner, to a matching point in the new cadastre whereas such an attempt was not appropriate for points on boundaries along creeks, rivers and coastlines, even where there are sharp direction changes.

A literature review revealed that several different methods have been proposed to solve the problem of determining the complexity or curvilinearity of a line (see Section 2.5). In particular, Jasinski (1990) mentions several methods involving estimation of fractal dimensions and one involving counting the number of times a line changes direction. The latter method was investigated as it seemed the least likely to impact processing times but was not found to be useful for making the distinction required to discriminate between road casements and riparian boundaries, possibly because in a large-scale dataset such as a cadastre, even a road casement boundary may change direction very slightly many times over a small distance.

Algorithms for assigning boundary type were coded and tested using various approaches including calculating the variance in turn angle between segments and the variance in the bisector angle at each point. Each version of the algorithm was run on several of the rural difficult-area datasets and in each case a thematic map, such as the one shown in Figure 6.7, was drawn. A visual assessment of the thematically mapped results was conducted to evaluate the success (in the subjective opinion of the author) or otherwise of the current version of the algorithm.

Ultimately, it was realised that the significant difference between roads and creeks, at least in the cadastral datasets available, was the rate of change in the turning angle along the line. The algorithm finally adopted computes the difference between the turn angle at each point and the turn angle at the previous point along the boundary and, wherever this angle difference was greater than 15°, a counter was incremented. For each boundary, the number of these significant direction changes was computed as a percentage of the total number of points on the boundary. By inspection of the thematically mapped results, it was concluded that, where this value was greater than 35%, the boundary was probably riparian whereas, below this value, the boundary was probably a road casement.

This computation was only applied where there were at least 20 points along a boundary; for smaller numbers, the results were found to be less reliable. There being no objective means of evaluating the results, the expressions were arrived at by conducting tests on the all the available rural and mainly-rural datasets. The threshold values, 15° and 35% were finalised when the results were deemed by the author to be satisfactory; it is not to be expected that results will be correct in every case.

This simple algorithm was found to be effective in distinguishing between the road and riparian boundaries as can be seen in Figure 6.7 illustrating a rural area from LGA07.

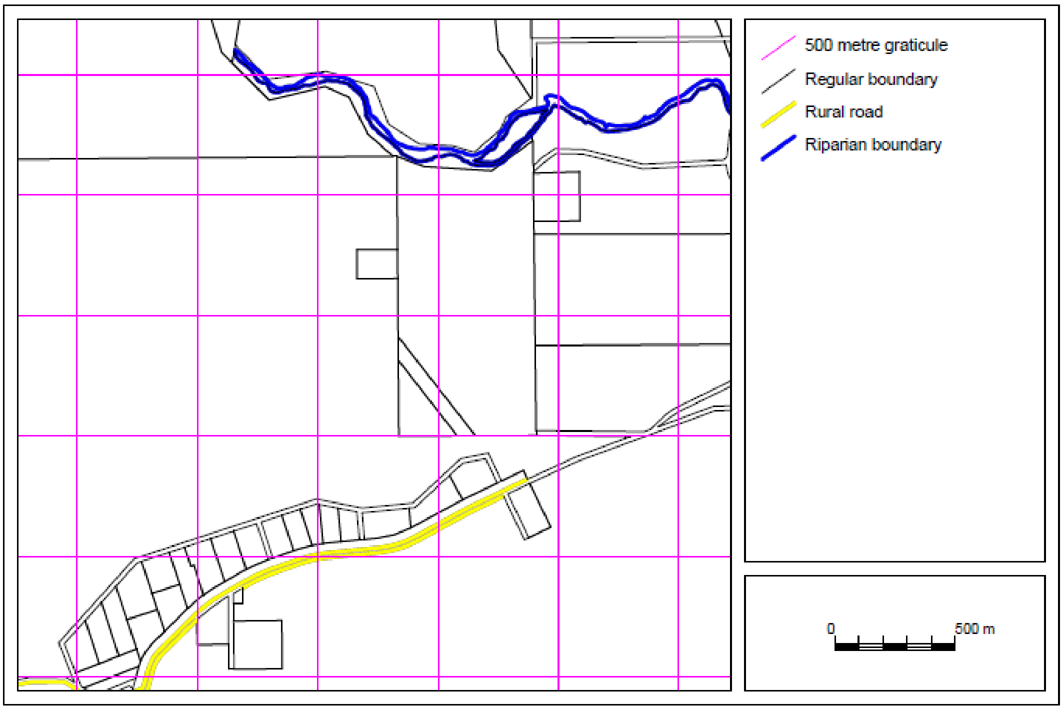

Figure 6.7 Boundary types

6.7 Checking the results

There is no straightforward way to check the correctness of the boundary matching even by visual inspection of the thematically mapped results. However, after the point matching and shift-vector generation processes described in Chapter 7 it became possible to use the shift-vectors to execute a trial adjustment of the old cadastre. It consequently became possible to discover the location of poor adjustment results and assess whether these were caused by incorrect or non-existent boundary matching or incorrect boundary classification. For example, the poor adjustment shown near the centre of the map in Figure 6.8 from LGA11 was discovered to be due to an incorrect boundary match. In this map, the old cadastre has been adjusted using the ArcGIS *RubbersheetFeatures* tool using the shift vectors generated by the matching algorithms as input links. The adjusted layer has then been drawn on top of the new cadastre. The exposed red line indicates a location where the adjustment was poor.

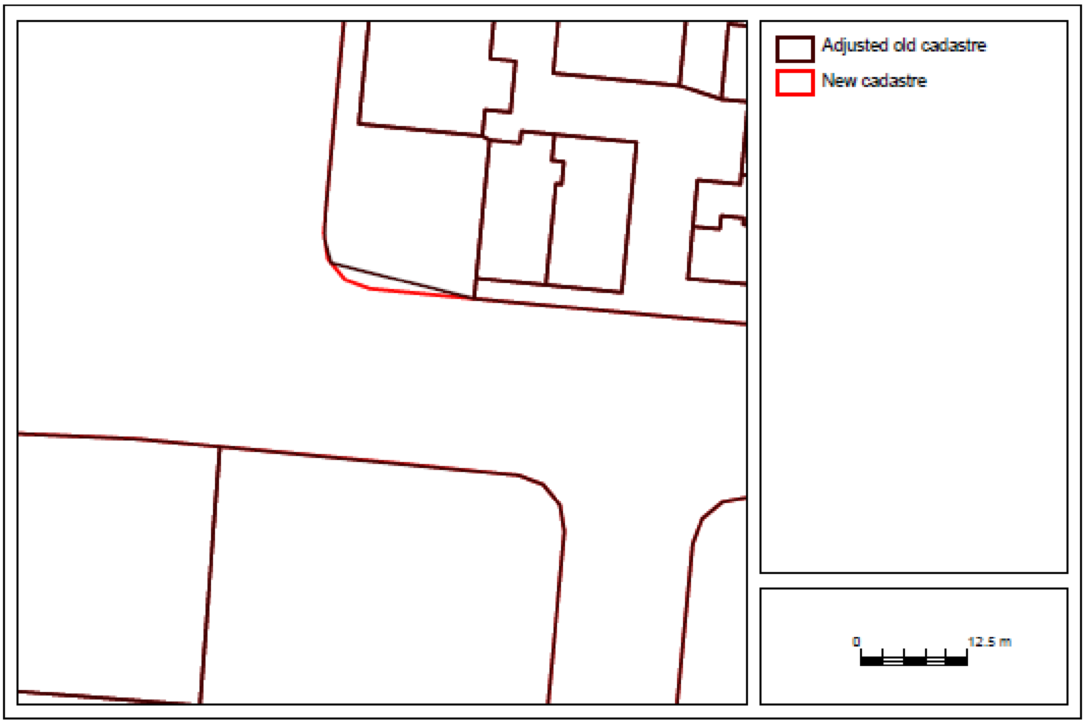

Figure 6.8 Poor adjustment due to an incorrect boundary match

The final expressions and threshold values used were the result of a compromise between finding too many incorrect boundary matches and boundary classifications and finding too few correct ones.

6.8 Summary

The results from the boundary matching research have shown that:

(a) It is possible, using the rules described in Section 6.4, to match many multipoint boundaries correctly.

(b) The algorithm will not always find a match when it should be possible.

(c) The algorithm may sometimes create an incorrect match.

(d) Correct boundary matches can improve shift vector generation.

(e) It is possible to distinguish riparian boundaries from road casements in many cases.

The next chapter will describe the algorithms that make use of the parcel matching and boundary-matching results to assist in the control point matching and the creation of shift vectors. The chapter will also describe how three different point matching algorithms were developed and how the attributes assigned to the points were used to determine the best algorithm to use for finding a matching point in the new cadastre.

7 POINT MATCHING

In order to create the shift vectors required for input to the ArcGIS *RubbersheetFeatures* tool, research was undertaken to match as many of the old cadastre points as possible, enabling shift vectors to be created between matched pairs of points. This chapter describes the rules developed to match each old-cadastre point to the corresponding point in the new cadastre, or to an appropriate location in the new cadastre in some cases.

Often there are several or even many new-cadastre points that are potential candidates for a correct match to an old-cadastre point. Points are also of diverse types, for example, nodes, corners, salient points, and others. Ultimately, three different Point Matching Algorithms (PMA1 to PMA3) were developed to automate the matching process, depending on the type of point and its attribute values (see Section 5.2). This chapter describes the criteria used to select the most appropriate algorithm for each point and details the rules encapsulated in each algorithm.

7.1 Overview

Section 7.2 describes the methodology used to check the shift vectors generated by the point matching algorithms described in the chapter.

Section 5.2 described the many attributes that were assigned to the parcel vertices when they were captured into the point layers. These were used at the point-matching/shift-generation stage of the process to create a target point, i.e. the expected location of the matching point in the new cadastre, around which a search for a matching new-cadastre point would be conducted; Section 7.3 below explains the target point concept. Other attributes described in Section 5.2 were used to determine a search distance around the target point; Section 7.4 describes the search distance concept and the method used to determine a maximum value for that distance. Yet other attributes were used to guide the selection of the most appropriate point matching algorithm. Section 7.5 describes the selection process. This section also includes a flow chart illustrating the way in which the different algorithms were selected and executed.

Sections 7.6 to 7.8 describe the three point-matching algorithms, PMA1 to PMA3. Each of these sections describes in detail how a search distance was computed depending on the attributes of the point and the algorithm used to match the point. Section 7.9 describes the use of PMA3 on riparian boundaries. Section 7.10 discusses considerations regarding further processing of unmatched points. Section 7.11 discusses special cases. Section 7.12 explains the importance of the individually computed search distances to the principal objective of this research. Section 7.13 summarises the point matching research.

7.2 Validating the point matching results

Once any point matching algorithm had been developed, the matched control points could be used to generate a shift vector connecting them. The ArcGIS *RubbersheetFeatures* tool can accept, as input, links or shift vectors from any source thus allowing this tool to be used with the output from the point matching algorithms developed for this research. This tool has been used to carry out a trial adjustment on the old cadastre after each test run of the research algorithms. Errors in the old-cadastre adjustment have been used to discover locations where the algorithms were performing badly and to evaluate whether a change to the algorithm has resulted in a greater number of correct shift vectors in the adjustment-error area.

Initially, the correctness of the point matching algorithms was assessed by inspecting the output from the *RubbersheetFeatures* tool using a GIS viewer. Locations where the adjusted old-cadastre parcel boundaries coincide exactly with the new-cadastre boundaries indicate areas where correct point matching has occurred. Later, a process was implemented to generate the vectors in the reverse direction, from the new to the old cadastre, and then identifying vectors duplicated in the reverse set. The number of identical vectors found could then be used for evaluation of the results because the presence of an identical vector suggests the correctness of a match with a high probability. Unfortunately, it is not possible to quantify the probability of identical incorrect vectors; there is no objective way to evaluate this – inspection of more than 350,000 individual shift vectors on the twelve LGA datasets was clearly not feasible and evaluation using one of the small DA datasets would be unlikely to be representative because of the extreme heterogeneity of the cadastral datasets.

7.3 The target point concept

Before attempting to match any point, other than those falling on matched boundaries, the expected location of the matched point was computed using the parcel centroid shift-vector length (*PSL*) and the parcel centroid shift-azimuth (*PSA*) and a new point generated (the target point) by projecting the current point location by these values.

Mapping the generated target points has shown that, for urban parcels that have been correctly matched, the points fall closer to the correct matching point than to other nearby points. In the area from LGA11 shown in Figure 7.1, 30 of the target points created in this urban area fall less than one metre from the correct matching point. Only for five points at the southern end of the long parcel in the east is the correspondence less accurate.

Target points are used in the point matching process for matching all salient points except those falling on matched boundaries.

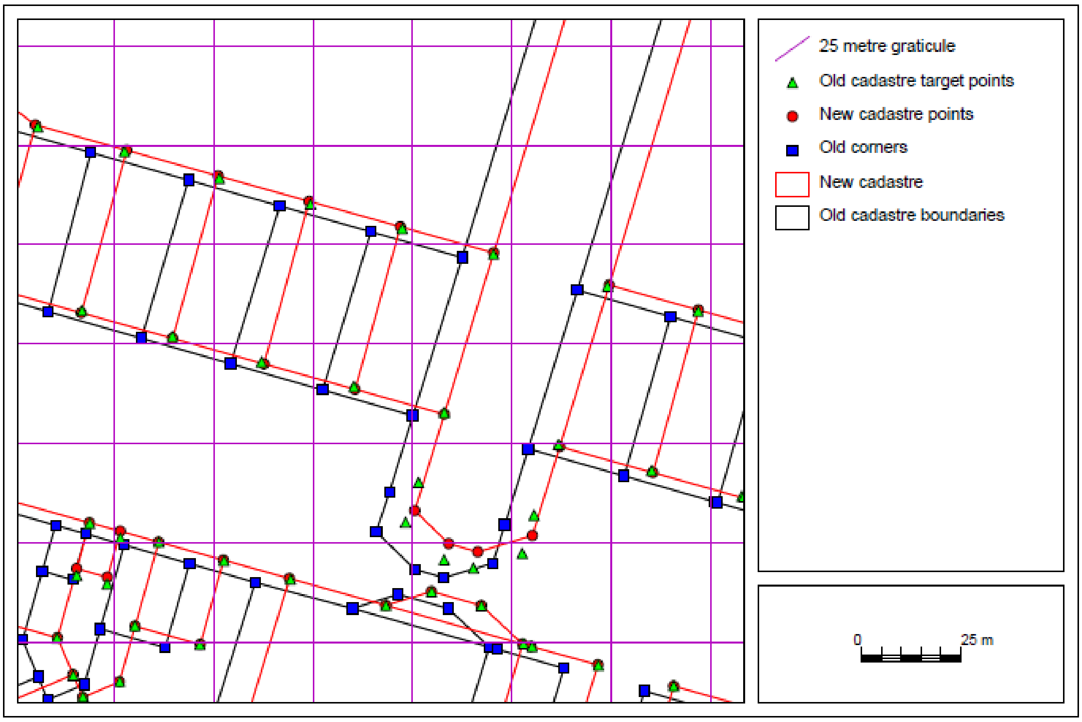

Figure 7.1 Target points

7.4 The search distance concept

The research carried out for this thesis has shown that the search distance most likely to achieve the best possible point match varies depending on the type of point, i.e. node, corner, salient point, or intermediate point, and the attributes of the point such as the segment angle and distance from neighbours. Figure 7.1 shows that the distance of the target point from the correct matching point can vary considerably. In this example, the distance varies between less than one metre to more than 12 metres. In rural areas, the apparent shift distances vary by much greater amounts.

The point matching algorithm for salient points makes use of a square constructed around the target point using a computed search distance. Any new-cadastre point falling within this square is then considered a potential match and evaluated using one of the algorithms described in this section. A circle could have been used for this purpose but is computationally more expensive both to create and to use for the spatial search and in practice makes little or no difference to the results.

For all points other than those on matched and riparian boundaries, a maximum search distance (*MaxSD*) was imposed. Before commencing the point-matching and shift-generation processes, length statistics were computed for the all the matched-parcel centroid shift vectors, i.e. those remaining after executing the process for the removal of bad centroid shift vectors described in Section 4.3.6. The maximum search distance was then set to the maximum parcel centroid shift vector length discovered (*MaxL*), using the equation:

$$MaxSD = \min(MaxL, 200\ m)$$

The maximum possible value for *MaxSD* was set at 200 metres because the initial visual inspection of the twelve full LGA datasets had failed to locate any real or apparent point movements on non-riparian boundaries greater than this value. There is no objective method for determining the maximum distance in apparent movement of parcel boundaries prior to attempting to match them.

This equation produced satisfactory results in the urban areas and eliminated several of the overlong shift vectors created in these areas at an earlier stage in the research when all the datasets were processed using a 200-metre maximum. However, results in rural areas showed that the apparent movement of points in a rural dataset could be much larger than the apparent movement of matched parcels. Figure 7.2 shows an area from LGA07 where the apparent point movement for several points from parcels in a rural area is more than 100 metres although the maximum matched-parcel centroid shift distance for this dataset was about 83 metres. Table 7.1 shows the maximum apparent parcel movement for all the LGA datasets.

A maximum search distance of 200 metres was used for rural datasets while the value computed from the equation above for MaxSD was used for the urban datasets.

Table 7.1 The maximum apparent shift of matched parcels

LGA ID	Maximum parcel shift distance	Dataset type
LGA01	7	Urban
LGA02	15	Urban
LGA03	40	Rural
LGA04	4	Urban
LGA05	60	Rural
LGA06	84	Rural
LGA07	83	Rural
LGA08	10	Urban
LGA09	3	Urban
LGA10	4	Urban
LGA11	59	Rural
LGA12	3	Urban

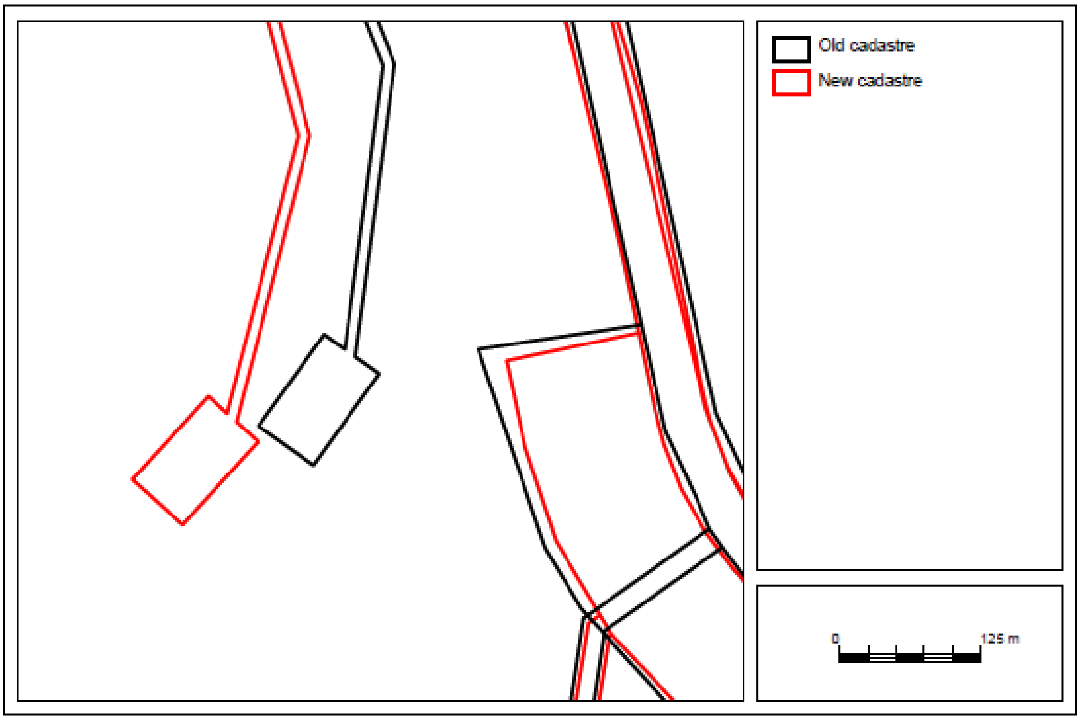

Figure 7.2 Large apparent vertex movement in a rural area

Experiments were conducted using a maximum value greater than 200 metres for *MaxSD*. In general, a few more shift vectors were created when greater values were used but there were also more vectors deleted by the elimination processes thus generating more locations needing to be checked by an operator. Since longer search distances can potentially result in a greater number of incorrect matches, particularly for points at locations where there is no corresponding point from the new cadastre, the method described above was adopted, i.e. to use a computed maximum for urban datasets but a fixed maximum value of 200 metres for unmatched boundaries in rural datasets. Note that this limit is not applied to points on matched boundaries which use a different point matching algorithm, PMA1, described in Section 7.6.

7.5 Choosing the point matching algorithm

During this research, it was determined that the most appropriate method for locating a match to a point depended on the individual characteristics of the point. Initially, the research focused on attempting to match every old-cadastre point to a corresponding point from the new cadastre. As the research progressed it was realised that this approach was only appropriate for certain types of points. The approach was often not useful for points from riparian and rural road boundaries where there is seldom a one-to-one correspondence between the vertices of two features captured on separate occasions. It became apparent that an accurate method of computing the correct new-cadastre location for intermediate boundary points, i.e. points lying between corners and nodes, could be implemented if the parcel boundaries could first be matched (see Chapter 6).

Because of these insights, three different Point Matching Algorithms (PMA) were implemented. Figure 7.3 is a diagram outlining the algorithm selection process.

PMA1 is the method used to generate shift vectors from source points on matched boundaries. The method uses the percentage distance of a point along the boundary to generate a point at the equivalent location along the matched boundary in the new cadastre; the method always results in a successful match (albeit, not to a point but just to a location) and can sometimes result in a match to an existing point if one falls close to the computed location on the matched boundary. PMA1 does not make use of MaxSD and can generate correct shift-vectors longer than 200 metres.

PMA2 is used for node, corners and other salient points. When successful, the method results in a match to a point from the new cadastre.

PMA3 is used for all non-salient points and any points not matched by PMA2. Like PMA1, the result is usually a location on a boundary in the new cadastre rather than an existing point although, as for PMA1, salient points may be matched to matching points if there is one close to the interpolated location.

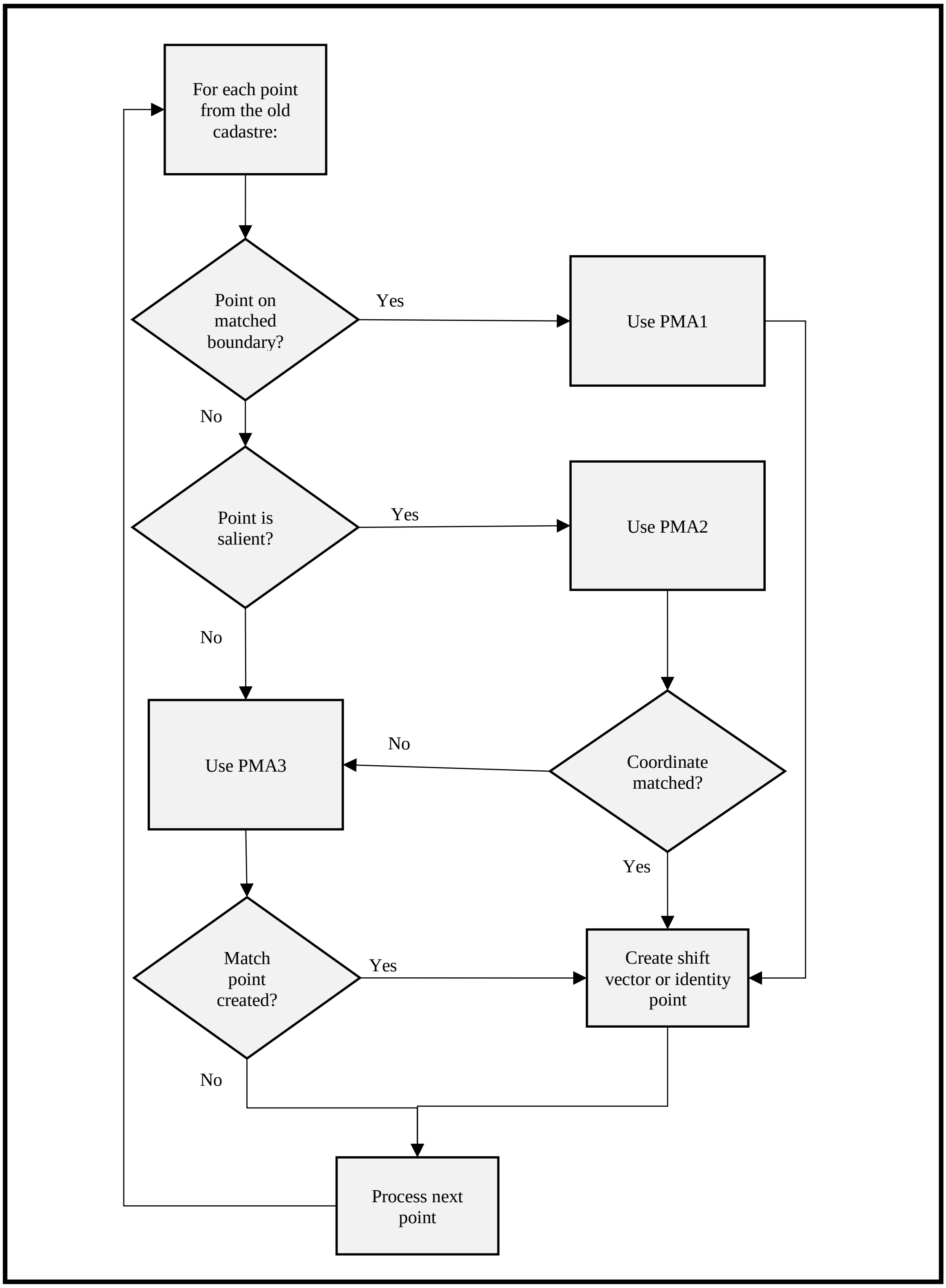

Figure 7.3 Outline of the PMA selection process

Each method, when successful, results in the creation of a shift vector from the source point to the matched point or to the interpolated location. Where the length of the shift vector is zero, an identity point is created. An identity point is a point that indicates to the ArcGIS rubber-sheeting process that this location is fixed, i.e. the correspondence between the two layers is exact at that point.

The resulting shift-vector layer is further processed to remove intersecting, touching, and over-long vectors. This process will be described in detail in the Chapter 8.

7.6 Processing points on matched boundaries – PMA1

Points located on matched parcel boundaries are the simplest of all to match. At the time of point creation, after the boundaries had been matched, each point was updated with the percentage distance of the point along the matched boundary and with the unique identifier attribute (UID) of the matched boundary; all matched boundaries were assigned matching UIDs at the boundary creation stage. Using this UID, the corresponding boundary in the new cadastre layer was selected and a point was created at the same percentage distance along its length. Initially, the generated location was then used as the endpoint of a shift vector starting at the old-cadastre point but, for the reasons described below, the algorithm was later upgraded to include a search around the intersection point for a matching new-cadastre point.

Where the parcel vertices are dense and the segment angles large, as occurs on the turning circle shown in Figure 7.4 from LGA01, the resulting shift vectors can deliver a perfect adjustment. However, inspection of the mapped results of adjustment trials revealed that this is not always the case on a boundary where some points have smaller segment angles, typically on boundaries where the vertices are sparse. In these cases, it becomes important to attempt to match salient points to a corresponding point from the matched boundary. A salient point was defined as any point having an acute angle between segments of less than 170°. The attempt to match a salient point from a riparian boundary to a point (rather than location) from the new cadastre was only carried out if the longer of the distances to adjacent points was greater than the parcel shift vector length, *PSL.*

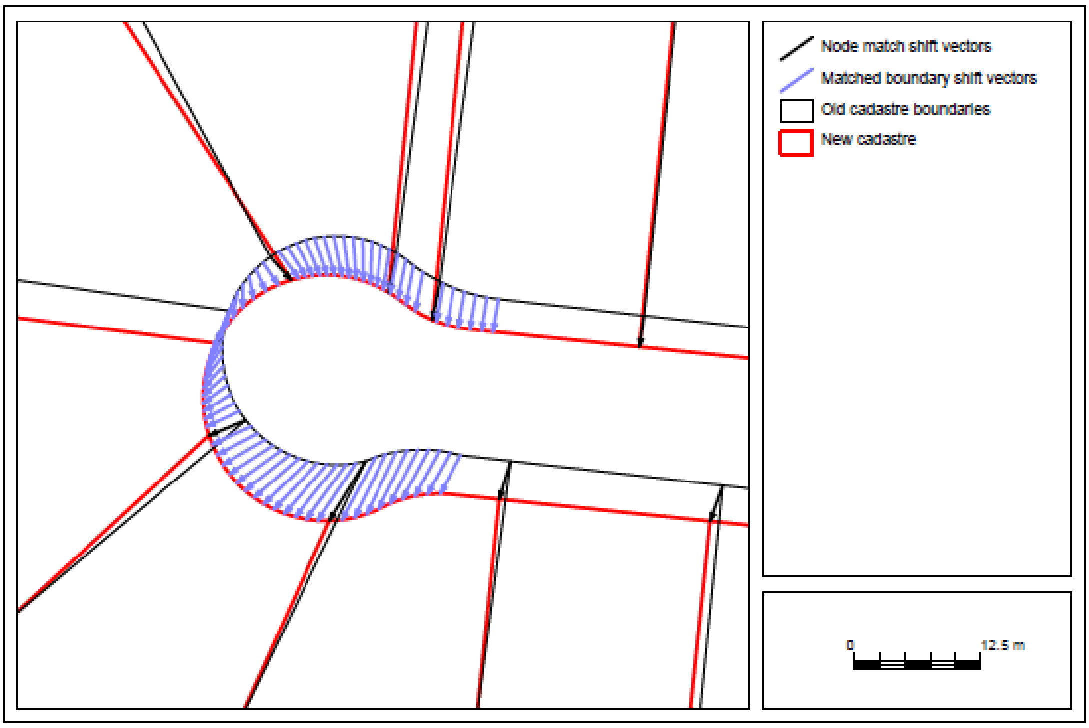

Figure 7.4 Vectors produced from matched boundaries on a turning circle.

The process used to match salient points on a matched boundary was as follows:

(a) Construct a search extent around the computed location (the intersect point) along the new-cadastre boundary. The width and height (D) of the extent were computed using the equation:

$$D = \max(\max(fromDist, toDist), 5 \text{ m})$$

where *fromDist* and *toDist* are the distances of the old point from its two nearest neighbours along the same boundary.

(b) Find the closest point to the intersect point that is inside the search extent, falls on the matched boundary, and whose bisector angle and turn angle (see Section 5.2) are each less than 10° different from those recorded for the old point.

(c) If a matching point is found, use that point to construct the shift vector rather than the interpolated location.

Figure 7.5 from LGA07 shows shift vectors on matched boundaries where some of the vectors (blue) have been constructed from a target point to a computed location but others (green) have been constructed between matched pairs of points. Figure 7.6 shows the same area after the shift vectors had been input to the ArcGIS *RubbersheetFeatures* tool to adjust the old-cadastre. The fact that the new-cadastre boundaries drawn in red cannot be seen indicates that the adjustment, in this case, was satisfactory; it is difficult to see any errors at this scale.

Figure 7.5 Shift vectors on matched boundaries

PMA1 was not implemented in the early stages of this research but inspection of old-cadastre adjustment results soon showed that simply attempting to match the old-cadastre points to existing new-cadastre points on complex boundaries, such as riparian boundaries, was delivering unacceptably poor results.

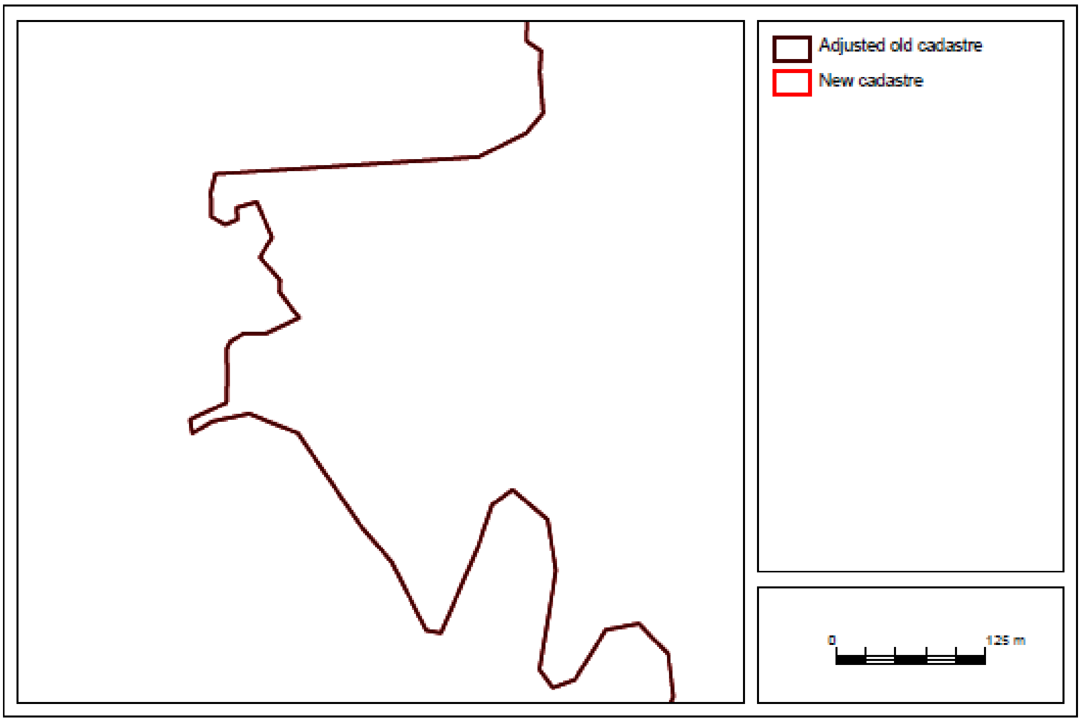

Figure 7.6 The same area as in Figure 7.5 after adjustment

Figure 7.7 shows a section of creek from LGA11 with and without the creation of PMA1 shift vectors. The area covers approximately 200 metres on the east-west axis. The two left-hand maps in the figure show the adjusted old cadastre in black drawn on top of the new cadastre in red. In the leftmost map, the shift vectors used to effect the adjustment were generated using PMA1. The poorer adjustment shown in the centre map used vectors created using PMA2 and PMA3 only. The rightmost map shows the unadjusted old-cadastre shaded green with black vertices and the new-cadastre vertices drawn in red suggesting that the results, using either point matching method would have been improved by densifying the vertices in the old cadastre along the riparian boundary, i.e. by adding more vertices; the advantages and disadvantages of this approach will be discussed in a Section 10.2.2.

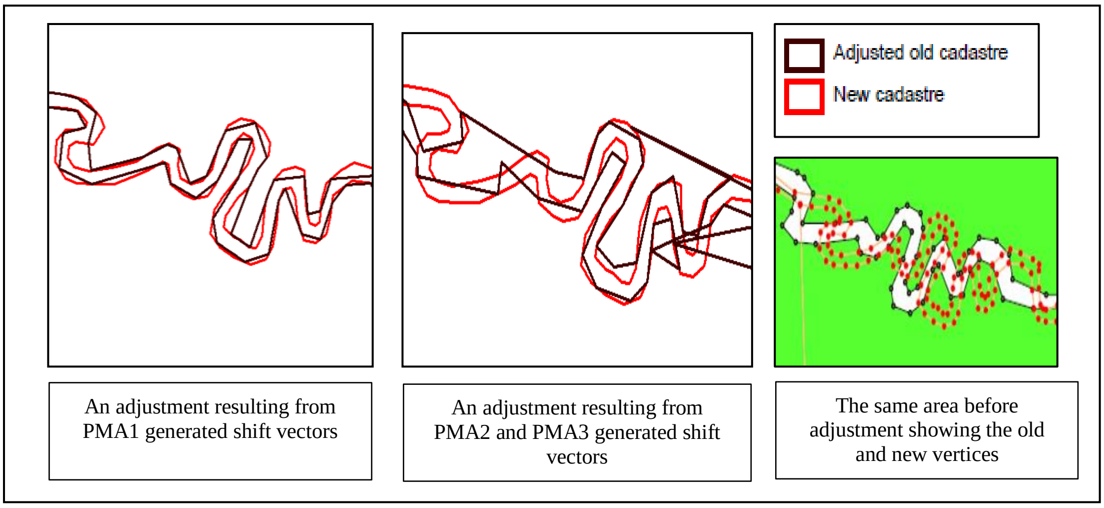

Figure 7.7 Adjusted creek with and without PMA1 shift vectors

The realisation that a perfect location match would be achieved for points from a correctly matched old-cadastre boundary arose from the work carried out by the author to implement the initial dynamic segmentation solution developed for Arc/INFO (Macduff, 1987).

7.7 Matching nodes, corners, and other salient points – PMA2

This matching method (PMA2) was used to match isolated points (see Section 7.11.1) and all salient points except for those on matched boundaries which are always processed by PMA1. For this purpose, a salient point was defined as any point falling on a node or a corner or any point with a segment angle of less than 170°.

Initially, salient points on riparian boundaries were not matched using PMA2. However, results suggested that where these points were widely separated, PMA2 could be successfully applied (see Section 7.7.4).

When matching salient points, all searches for a potential match are carried out by generating a search rectangle around the target point created by the process described in Section 7.4. This search rectangle is then used to select points for evaluation.

During this research, it became apparent that much larger search distances, used to locate potential matching points, could be used for some types of points than for others. For example, a corner on a matched parcel is much easier to match than most other points because, typically, each corner will have a different bisector azimuth so that potential matches can be limited to just those have a similar bisector angle. The research also suggested that search distances for salient points other than corners and nodes should additionally consider other attributes of the point.

Figure 7.8 shows an area from LGA11 with correctly matched parcels and one parcel with the bisector azimuths indicated. Locating the correct corresponding corners in the new cadastre only requires that the parcel's UIDs match and that the bisector azimuths are close in value. In this case, even the use of a large search distance, that could find many potential matching points, is unlikely to result in the wrong point from the new cadastre being selected. There are exceptions, however, which are discussed in Section 7.7.3.

The different search distance computations are described in the next two sections.

Figure 7.8 A matched parcel showing bisector azimuths

7.7.1 The search distance for corners and nodes

For corners and nodes, the following equation was used to set the search distance (*SD*):

$$SD = \min\left(\sqrt{A}/2, maxSD\right)$$

where *A* is the parcel area and *maxSD* has been set as described in Section 7.4. The reason for using half of the square-root of the parcel area to set the search distance is that this search rectangle will be large enough to capture all potential matches (the distance would be half the length of one side of a square parcel), even where the target point is not accurate, but not so large that large numbers of nearby points would also be selected for evaluation. This would not be a problem when processing points from matched parcels but where the parcel is unmatched there may be many nearby points with matching "UIDs" having similar bisector angles. All unmatched parcels were allocated the same "UID" value, -1, meaning that, where the parcels are not matched, this attribute cannot be used to distinguish between potentially correct point matches and incorrect ones.

7.7.2 The search distance for other salient points

For all other salient points, it became apparent during this research that the search distance would need to take account of the mean distance between vertices along the segment and the acute angle between segments at this point as well as the expected shift distance and shift angle. The following equation was used to calculate the search distance:

$$SD = \min(\max(BMD, PSL * (1 + (4((180 - SA) / 180)))), maxSD).$$

BMD is the mean of the distances between vertices on the current boundary, PSL is the centroid shift vector length of the current parcel and *SA* is the segment angle, i.e. the acute angle between the line segments either side of the vertex from which the current point was created. Using this equation, the search length varies depending on the significance of the angle at the point. For points that fall on an almost straight line, i.e.one with a segment angle close to 180°, the value will be equal to the parcel centroid-shift distance, *PSL*; for points with highly acute segment angles the value will be closer to five times *PSL*. This search distance, however, is not permitted to fall below the mean distance between vertices along the segment or to exceed the maximum search distance *maxSD*.

The rationale for this equation is as follows. Earlier in the research the equation

$$SD = \max(\sqrt{A}/4, maxSD)$$

was used to calculate the search distance for salient points other than corners and nodes. However, it was observed when examining the old-cadastre adjustment results, that this equation produced many incorrect matches. In many cases, points in rural areas were wrongly matched to a remote point from an incorrect boundary that happened to have a matching bisector angle. After the change to the new equation, the results showed that such errors were significantly reduced. Unfortunately, it is not possible to quantify the reduction in errors. Every amendment to the code results in a different number of points matched and a different number of shift vectors deleted. The numbers alone do not indicate whether a solution has been improved by the code change; a greater number of shift vectors does not necessarily indicate a better result unless the additional vectors are all correct. In the absence of any prior information as to which point-pairs should be matched, mapping the old-cadastre adjustment results in a GIS viewer and inspecting the known problem areas is the only method by which the results could be evaluated.

7.7.3 The point matching algorithm for PMA2

The algorithm for matching the points selected for matching by PMA2 is as follows.

First, select all the new-cadastre points that fall within the search extent (see Section 7.4) whose parcel UIDs match that of the current point and whose bisector angle and turn angle (see Section 5.2) are each less than 25° different from those recorded for the old point. Then, for each selected point, carry out an evaluation to determine the best match as follows:

(a) Calculate the difference in bisector angles as a percentage of 180° (*dBA%*).

(b) Calculate the azimuth of a line drawn between the current old-cadastre point and the candidate point (*SA*).

(c) Calculate the difference between *SA* and the parcel centroid shift azimuth (*PSA*) as a percentage of 180° (*dSA%*).

(d) Calculate the distance from the current point to the candidate point (*SL*).

(e) Calculate a value *dSL%* from *SL* and *PSL* using the equation:

$$dSL\% \;=\; 100 - (\,\min(SL, PSL)\;/\;\max(\,SL, PSL)\;*\;100\,).$$

(f) Calculate a value *dPA%* from the difference between the area of the parcel the current point came from (*PA1*) and the area of the parcel the candidate point came from (*PA2*) as a percentage of the larger of the two parcels using the equation:

$$dPA\% \;=\; (\min(PA1, PA2)\;/\;\max(PA1, PA2)\;*\;100\,).$$

(g) Calculate the difference in turn angles between the current and the candidate point as a percentage of 360° (*dTA%*). For points touching the world polygon, this value was calculated as the difference between the acute angles at the point to allow for the possibility that boundaries adjacent to the world polygon may be ordered in opposite directions (see Section 7.8.2.).

(h) Assign a Match Score (*MS*) to the candidate point as follows using the equations

$$AF = \sqrt{PA} / 10$$

$$MS = dBA\% + dTA\% + (dSA\% + dSL\% + (100 - dPA\%)) / AF.$$

PA is the area of the source parcel. The factor AF has been applied to just those elements of the equation that have been derived from the attributes of the originating parcel (dSA%, dSL%, and dPA%), rather than from attributes of the point such as bisector angle and turn angle (dBA% and dTA%). This has the effect of giving less weight to these components when they are derived from a large parcel. This factor has been applied because it has been noticed, by observation of the generated shift vectors, that the parcel centroid shift created for large rural parcels is often an inaccurate indication of the expected shift for all the vertices that define the parcel.

The match score has been designed, using the equations described in this section, to have a smaller value for more likely matches.

(i) Save the candidate point if its match score is smaller than the score for any previously saved point in the current selection set.

(j) When all candidate points have been processed, that last point saved (if any) is used to construct the shift vector.

During this research, it was observed that points on unmatched parcels could sometimes be matched to points on parcels in an adjacent block; unmatched parcels were all assigned the same "UID" number, therefore, the parcel UID could not be employed to validate the match. The above algorithm was therefore refined to ensure that, where the relevant old cadastre block was matched to one in the new cadastre, only points in the matching block would be considered.

It was also observed that parcels could sometimes match "by accident". Figure 7.9 from LGA05 shows a case where a parcel in the old cadastre, shaded green in the centre of the map and in the upper inset, has matched to the new-cadastre parcel shaded blue in the inset. The large new-cadastre parcel has had a new small parcel excised from the upper left of the original old-cadastre parcel and the old parcel has merged with the one shaded in grey in the inset indicating a simultaneous subdivision and amalgamation of the properties. The result of these two changes is a match between the larger of the parcel pairs leaving the two smaller ones of each pair unmatched; by chance, the areas and perimeters of the two wrongly matched large parcels fall within the software specified tolerance. The smaller-parcel points are flagged as being on an unmatched parcel but the corresponding points in the new cadastre are marked as being matched. To handle this situation, when matching points by method PMA2, if the search for points in the search area yields no hits with matching UIDs, then the search is repeated for possible matching points whose UIDs do not match that of the old-cadastre parcel.

Prior to making this change, the points from the unmatched small parcel remained unmatched.

Another error that can arise on incorrectly matched parcels is the creation of an overlong vector from a node to an incorrect new-cadastre point. This can arise because of the large search distance used for nodes. Usually, a node on a matched parcel should be matched to a node on the matching parcel in the new cadastre. Before an error of this type was discovered, the algorithm used to eliminate statistical outliers (see Section 8.2.4) only processed vectors on unmatched parcels. This algorithm was, therefore, amended to additionally check points on matched parcels where a node was matched to a non-node or vice versa. Inspections showed that the results were subsequently improved.

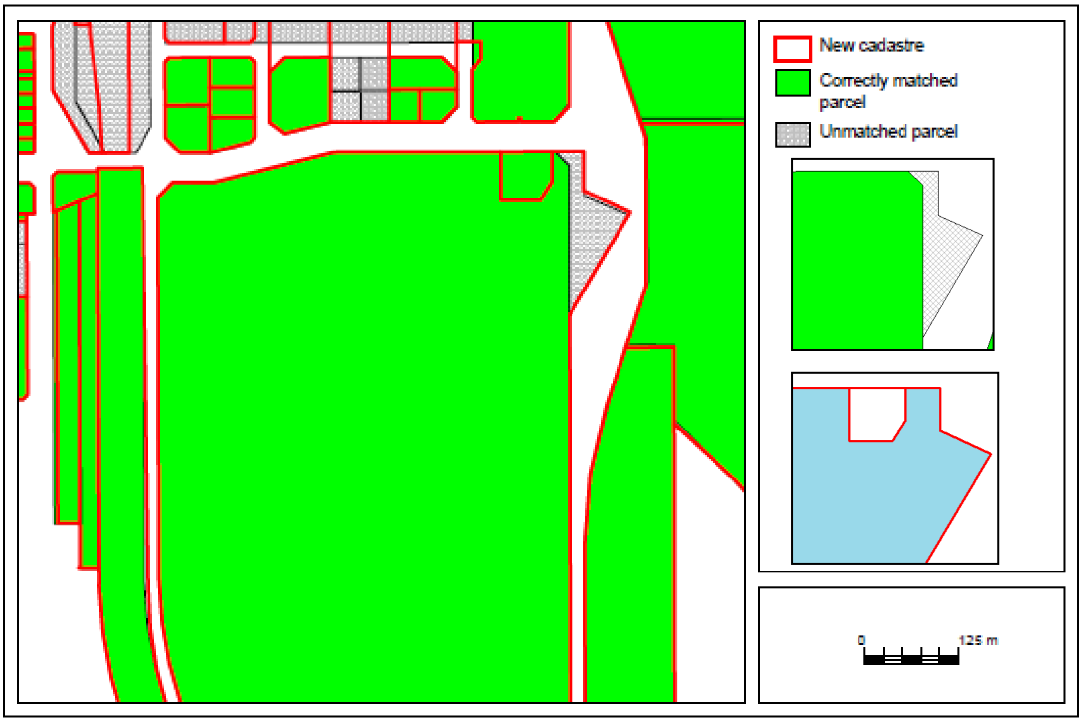

Figure 7.9 An “accidental” parcel match

Inspection of the results of the PMA2 process showed that, in a small number of cases, a node had remained unmatched when it would have been clear to a human operator that a match should have been created. This occurred because the target point fell beyond the search distance from the correct node match. The algorithm was therefore modified to iteratively increase the search distance in increments of 20% up to one-half of the square root of the parcel area or *maxSD*, whichever was the smaller, or until the node was matched to another node. This resulted in just 12 additional matched nodes in a large rural dataset with more than 50,000 points (LGA07). However, since every correct shift vector created may result in reduced operator effort at the manual error correction stage described in Section 8.3, the algorithm enhancement was retained.

7.7.4 Matching salient points on riparian boundaries.

Salient points on unmatched riparian boundaries were only matched using PMA2 if the greatest distance of the point from its nearest neighbours on the same boundary was greater than the apparent parcel shift distance, i.e. if

$$\max(fromDist, toDist) > PSL$$

For these points, the search distance was set to

$$\max(fromDist, toDist)$$

This addition to the code delivered improved results on long, sparsely digitised riparian boundaries.

It was found to be important for the success of this step that redundant points on straight sections of the parcel boundary were removed as described in Section 5.3. This was because, before the solution was upgraded to remove these points from the old-cadastre point layer, the distances of the salient point from its nearest neighbours could be artificially small and the point would not be selected for matching by PMA2 because the distances were smaller than *PSL*.

As with all the algorithms described in this chapter, the rules and threshold values documented in this section (7.8) have resulted from iterative refinement and repeated tests until the results were satisfactory to the author.

7.7.5 Match results on urban salient points

All the points shown in Figure 7.1 fall on urban parcels. Figure 7.10 shows that all the shift vectors created in the area of LGA11 have connected correctly matched control points. All the vectors have also been matched to an identical reverse vector except the two drawn in yellow in the south-east; these have no corresponding reverse vector because there is no corresponding new-cadastre point to act as a source point. For this difficult-area dataset (DA18), more than 98% of the point-to-point matches were duplicated in the reverse matching process. The identical reverse shift vector count for all the DA and LGA datasets will be listed in the Chapter 9.

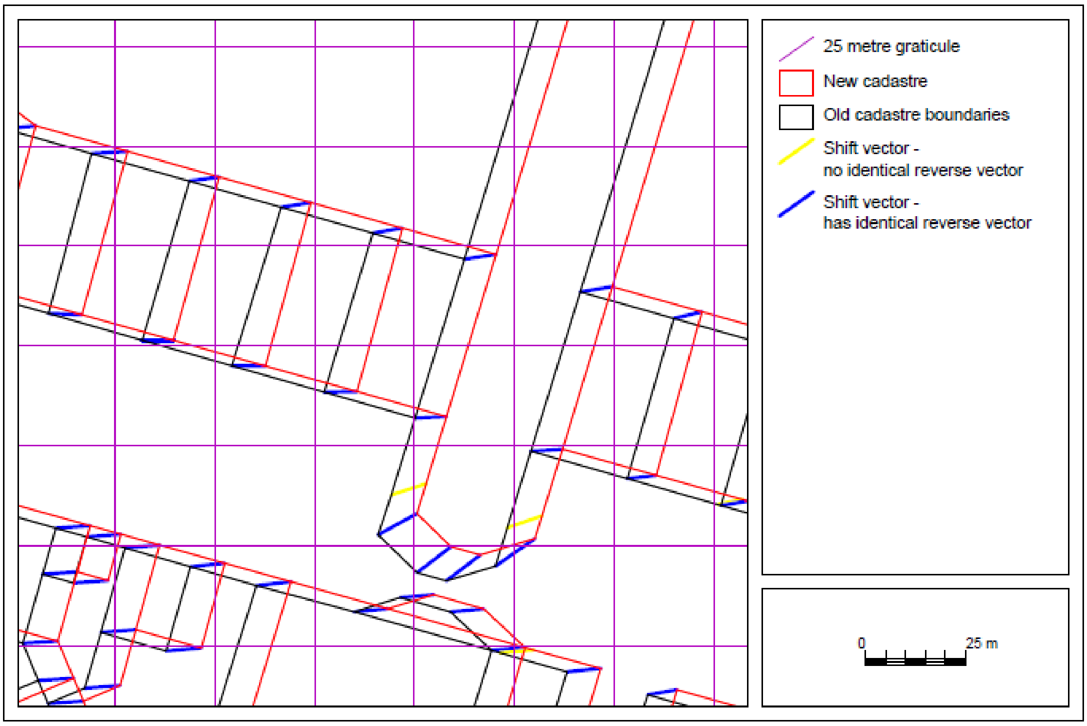

Figure 7.10 Shift vectors created on the parcels shown in Figure 7.1

7.8 Matching the remaining points – PMA3

PMA3 was used to create shift vectors for all the points not yet matched. Only points identified as corners were excluded because the use of PMA3 on these was found to produce unacceptable results. There is no objective way to evaluate the quality of the results from the PMA3 algorithms other than inspection of the adjusted old cadastre using a GIS viewer, seeking areas where the adjusted old-parcel boundaries do not overlay the new-parcel boundaries; reverse vectors are not expected to be identical because the PMA3 algorithms do not attempt to match to a new-cadastre-point, rather they attempt to identify an appropriate location on a matching parcel boundary to locate the other end of a shift vector. Of necessity, the evaluation of the results was arrived at by application of the subjective opinion of the author.

The PMA3 process is similar to PMA1 in that the process is not attempting to match the source point to a target point but to locate an endpoint for the shift vector on the most appropriate nearby parcel boundary. In outline, the method involves the creation of an intersect vector constructed either through the target point (intersect method one – IM1) or through the old-cadastre point to be matched (intersect method two – IM2). This line was then used to select all intersecting parcel boundaries in the new cadastre; each selected boundary was then evaluated to determine the best matching boundary. If the point was not salient, the point at which the search vector crosses that boundary is then used as the end-point of the shift vector. For salient points, an additional search for possible matching points was conducted around the intersect point.

Details of the algorithms used are described in the next sections.

7.8.1 The intersection distance

When constructing a vector to employ for the purpose of selecting potential matching parcel boundaries, it was first necessary to calculate a suitable vector length, the intersect distance (*ID*). Once again, it has been found that the most appropriate length for the search vector depends on the attributes and location of the source point. It is important that the vector is long enough to locate all possible matching boundaries but that it is not so long that a large number of very distant boundaries are selected for evaluation.

The maximum permitted value for search vector length was *maxSD* for all points not on a riparian boundary. A minimum distance of 20 metres was imposed for all points. The minimum distance was employed because it was discovered, by inspection of results, that creek boundary movements can be much larger than the adjacent parcel centroid movements. Since the algorithm employed to select the old-cadastre points to be matched (see Section 5.7) preferentially selects points from matched parcels, the parcel centroid shift distance would often be too small for the successful match of the points along the creek boundaries thus giving rise to the need to impose a minimum search distance.

7.8.2 The boundary-matching algorithm

The search vector described above will typically intersect more than one parcel boundary and, in complex areas, many more. The shapefile format employed by all the datasets used in this research stores boundaries twice where polygons are adjacent. Only in the case of island polygons, or polygons having one or more boundaries adjacent to the world polygon, is there the possibility that the search vector only intersects one boundary line. It is, therefore, necessary to determine which of the intersected boundaries best matches the originating boundary. At this stage of the point matching process all points from matched boundaries, except for riparian boundaries with more than 100 points on a parcel that is not an island parcel, have already been matched. On unmatched boundaries, it is possible that the lengths and other spatial attributes of the boundaries are quite different so that these attributes cannot be used to discriminate between possible matches. It is, therefore, necessary to evaluate the direction of a short section of the new-cadastre boundary line for a match to the direction of a short section of the old-cadastre boundary line around the source point.

Figure 7.11 shows an intersect vector (yellow) through a point from an old-cadastre road boundary. The vector crosses two boundaries in the new cadastre (red). The question now is: to which of the two new-cadastre boundaries crossed by the intersect vector should the old-cadastre point be matched?

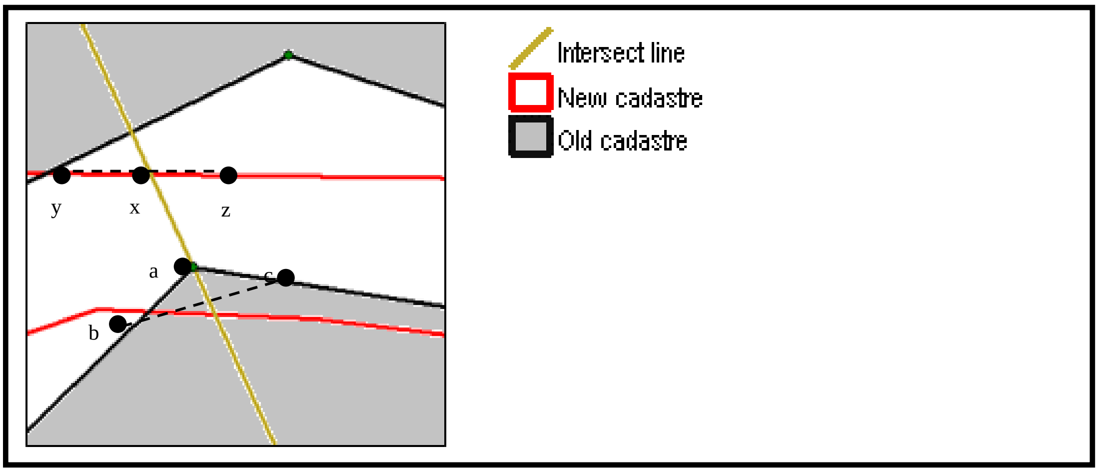

Figure 7.11 An intersect vector crossing two boundaries

The algorithm used to determine if an intersected boundary is a potential match to an originating boundary, i.e. a boundary upon which the source point falls, is as follows. For each boundary on which the source point falls:

(a) Locate two points (b and c in the diagram) at five metres either side of the source point from the current boundary. Calculate the azimuth of a line between these two points.

(b) Calculate the mean distance between vertices from the source boundary as a percentage of the boundary length ($\mu CD1\%$).

(c) Select all new-cadastre boundaries intersected by the search vector and whose boundary UIDs (see Section 6.4), boundary types and block IDs match. Unmatched boundaries and unmatched blocks all have the same "UID" value, 0.

(d) For each selected new-cadastre boundary, locate two points (y and z in the diagram) at five metres either side of the intersection point on the selected boundary. Calculate the azimuth of a line between these two points. Also, calculate the mean distance between vertices from the target boundary as for the source boundary ($\mu CD2\%$).

(e) If the source boundary is an unmatched boundary and there are more than five vertices on that boundary and the difference between $\mu CD1\%$ and $\mu CD2\%$ is greater than 35%, reject this match, otherwise, if the azimuths computed for the current pair of old and new boundaries are within 45° of each other, accept the boundary as a potential match.

All intersection locations on all possible matching boundaries undergo further evaluation to allow selection of the one most likely to be correct. Details of the further evaluation will be described in the following sections.

Note regarding the vertex mean-distance check: The inclusion of the vertex mean-distance component (e) ensured that points from boundaries with more than five vertices were not matched to locations on a boundary with only two or three vertices. For complex boundaries such as riparian boundaries, before this check was incorporated, it was observed that points from these boundaries would sometimes be matched to a nearby simple boundary that happened to match the azimuth of the source boundary.

Note regarding the azimuth test: To facilitate the identification of nodes for this research, a polygon representing the world was created from a minimum-bounding circle around each cadastral dataset. However, no boundary layers were created from these polygons. As the research progressed it became clear that these polygons could enclose quite different spaces in each cadastre. For example, Figure 7.12 shows an area from the LGA07 dataset where what appears to be a river is part of the world-polygon in the new cadastre but not in the old. The large grey area at the top of the map is part of the old world-polygon but not the new. The effect of this is that some single-boundary vertices that should match may be ordered in opposite directions in the two datasets. To improve the possibility of finding a suitable boundary crossed by the intersect vector, if there are two boundaries at the source point, both are checked. The azimuth test is of value around coastlines and islands in datasets where the world polygons cover approximately the same areas and both the source and the target boundaries are from an island parcel, and in dense urban areas where the intersect vector may cross several parcel boundaries.

Another potential solution to the problem of mismatched world polygons would be to create boundary lines for the world polygons. This has not been tested but could be the subject of further research.

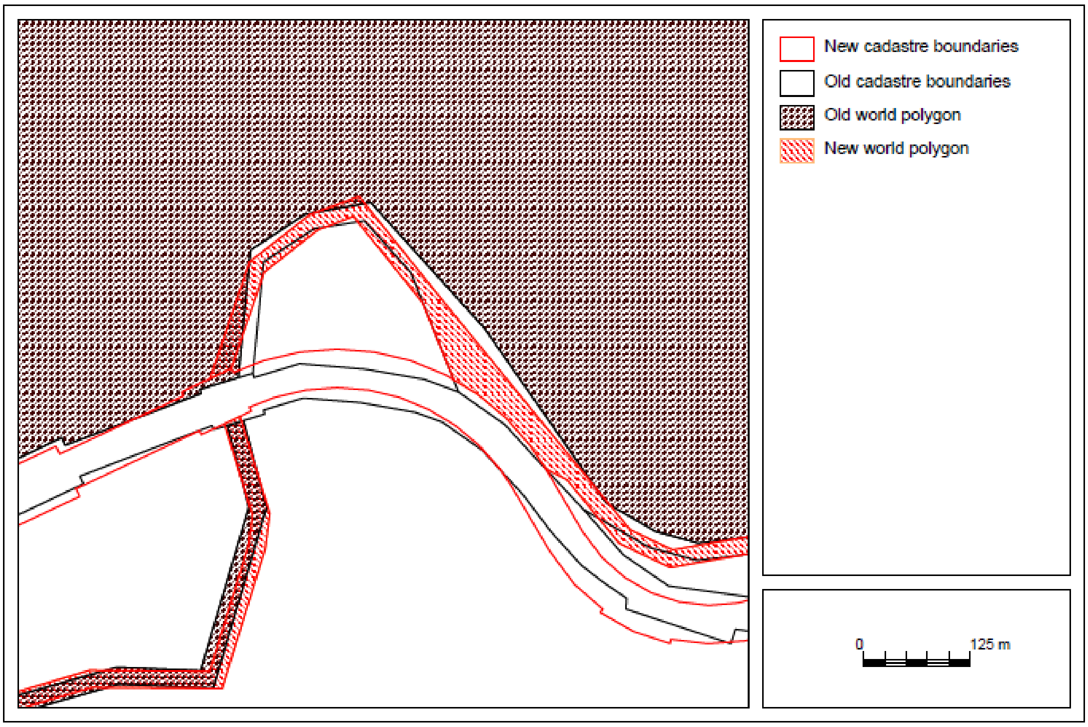

Figure 7.12 World polygons

Note on the minimum-bounding circle. A circle was used rather than a minimum-bounding rectangle because the ArcGIS software used to create the rectangle was observed to create small polygons at rounded corners; this gave rise to additional point matching problems.

7.8.3 Choosing the intersection match method

Depending on the attributes of the current point, different search distances and search methods were adopted. Figure 7.13 is a flowchart showing the processes involved in choosing the most appropriate Intersect Method (IM) to employ. The methods used are described in detail in the succeeding sections.

Only points from parcels with an area of less than one million square metres were selected for processing by IM1 and then only if the point falls on a regular parcel boundary (i.e. not a riparian or road boundary) or it falls on a road boundary and the parcel has an area of less than 20,000m². The threshold values arrived at are, of course, arbitrary; they have been determined as the result of iteratively testing and modifying them, initially on the difficult-area datasets, and, when the results were satisfactory to the author, on the complete LGA datasets. The version of the algorithm described here has delivered the best results so far observed.

Points from very large parcels (over 1,000,000m²) were excluded from the IM1 process because IM1 uses the parcel centroid shift vector length and azimuth to determine the most likely location of the matching point but these measures were found to be unreliable indicators of point shift distance and direction on very large parcels. Inspection of the results delivered by earlier methods of selection had also shown that IM1 seldom delivers satisfactory results on riparian boundaries; these were, therefore, also excluded from the IM1 process. Results using IM1 on rural road boundaries were acceptable but experiments comparing the IM1 results with the IM2 results suggested that IM2 was to be preferred unless the boundary identified as a road boundary falls on a small parcel, i.e. less than 20,000m².

The selection of the values to be used for parcel size in this selection process have been arrived at as the result of repeated tests on difficult areas until the inspection of the cadastral adjustment was satisfactory in the author’s opinion, i.e. when the adjusted old cadastre parcel boundaries exactly coincide with the corresponding boundaries in the new cadastre.

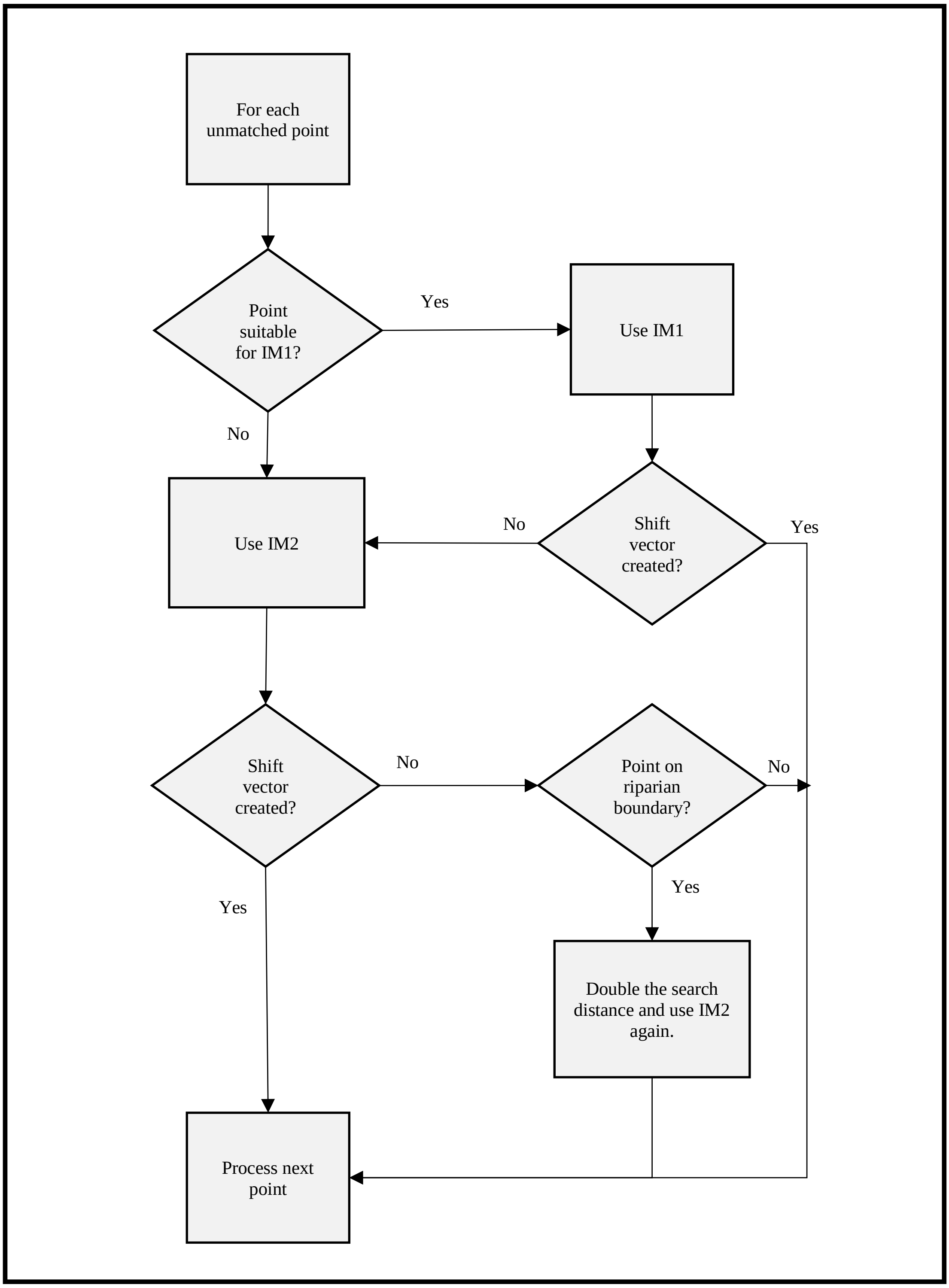

Figure 7.13 Intersect method choices

7.8.4 The intersection process when searching from the target point – IM1

Intersect method 1 uses an intersect search vector created around the target point with the angle used to construct the vector being derived from the parcel shift angle, *PSA*. Figure 7.14 from LGA07 shows intersect vectors centred on target points and using the parcel shift angle. The intersect vectors are drawn in brown.

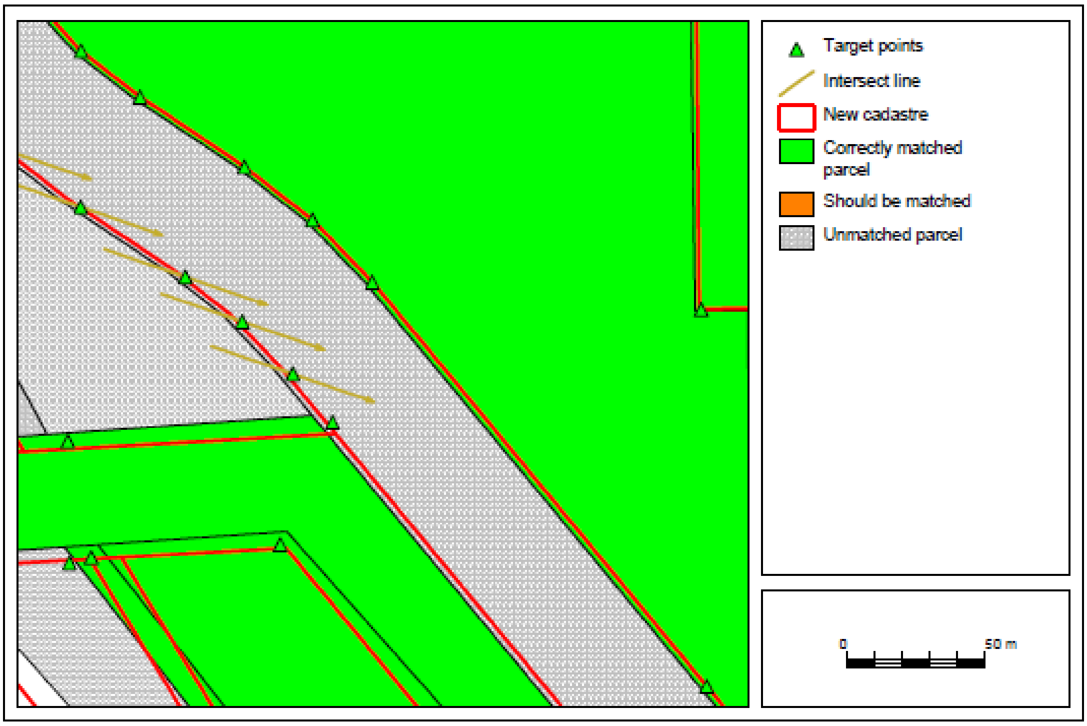

Figure 7.14 IM1 intersect vectors.

The algorithm used to create the shift vectors is as follows:

(a) Set the intersection distance *ID* using the equation

$$ID = \min(\max(PSL * 4, 20\text{ m}), MaxSD)$$

The multiplication factor, 4, was arrived at by testing different values on the difficult-area datasets and inspecting the mapped results. The value selected allows the algorithm to search a reasonably large area around the target point but is not so large that it would unacceptable increase the possibility of an incorrect match. The final value was chosen when the results were satisfactory to the author.

(b) Create a vector passing through the target point using the parcel shift azimuth *PSA* and having a length of twice the intersection distance *ID*, i.e. allow a search in both directions from the target point.

(c) Use the algorithm described in Section 7.8.2 to find all potentially good intersection locations and, for each location, determine the distance of the location from the target point and set a Boolean variable indicating whether the point lies in the expected direction, as indicated by the *PSA*, from the source point. Where the point is in a direction opposite to the expected direction, weight the distance by an additional 20%; the weighting causes the algorithm to prefer a location in the expected direction.

(d) Select the intersection point whose distance, possibly weighted, is the shortest distance from the target point.

7.8.5 The intersection process when searching from the source point – IM2

Intersect method 2 uses a vector created around the source point, rather than the target point, with the angle used to construct the vector being the bisector azimuth *BA* from the source point rather than the parcel shift angle *PSA*.

Figure 7.15 shows an area from LGA07 where intersect vectors created by this method (yellow) have resulted in the generation of correct shift vectors (red). The inset shows the same area after the application of the ArcGIS *RubbersheetFeatures* tool to the old cadastre. The old cadastre has been drawn in black on top of the new cadastre in red. The fact that the red lines of the new-cadastre boundaries cannot be seen indicates a correct adjustment.

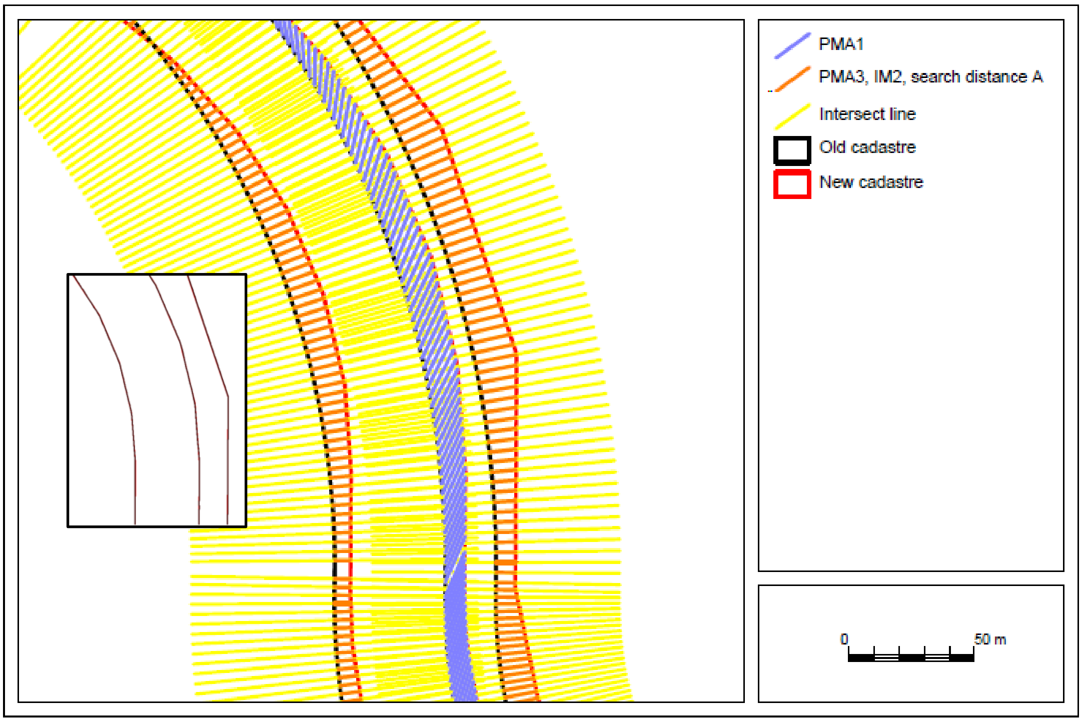

Figure 7.15 IM2 intersect vectors

This method may be called twice if the first pass does not produce a shift vector. On the first call, the intersect distance, ID, was set using the same equation as for IM1, i.e.

$$ID = \min(\max(PSL * 4, 20\text{ m}), MaxSD)$$

This distance is referred to as "Search distance A" in the legends.

The second call was made only for unmatched points on riparian boundaries. Here the search distance was doubled using the equation

$$ID = \min(ID * 2, 200 \text{ m})$$

This second pass was introduced after it was observed that creek lines in several areas had moved by larger distances than expected from the parcel centroid distance. Note that the value set for *MaxSD* was not used for the second pass; the upper limit is set to 200 metres. This distance is referred to as “Search distance C” in the legend.

Figure 7.16 from LGA11 shows three vectors just below the centre (dark brown) created on the second pass. In this case, the value of *PSL* was 15.5 metres but the second pass vectors were more than 65 metres long.

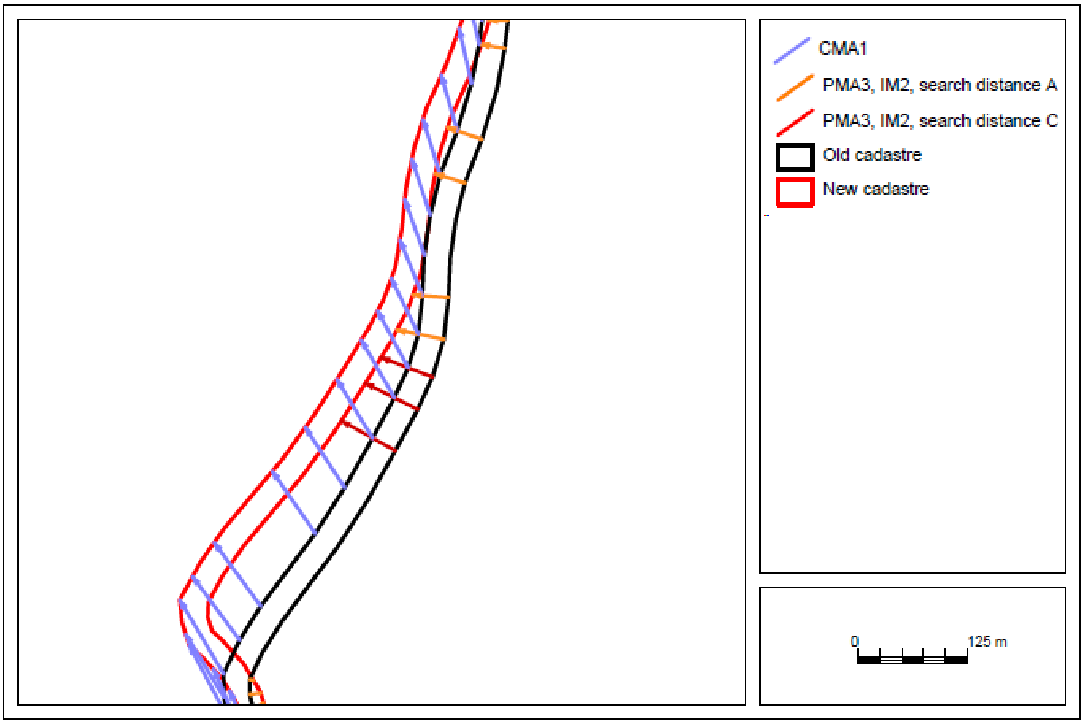

Figure 7.16 IM2 second pass intersect vectors

The algorithm used to create the search vectors is as follows:

(a) Create a vector passing through the source point using the source point bisector angle, *BA*, and having a length of twice the intersection distance *ID*.

(b) Use the algorithm described in Section 7.8.2 to find all potentially good intersection points and, for each point, determine the distance of the point from the source point and set a Boolean variable indicating whether the point lies in the expected direction (i.e. as indicated by the *PSA*) from the source point.

(c) Select the intersection point whose distance is the shortest distance from the target point. Where two points are equidistant from the target point but in opposite directions, choose the point that lies in the expected direction.

7.8.6 Searching for a matching point

For all salient points, i.e. having a segment angle of less than 170°, where an intersection point has been located, an attempt was made to match the point to a nearby one on the same new-cadastre boundary. A search rectangle was constructed around the intersect point on the new-cadastre boundary. The width and height (D) of the extent were computed using the equation:

$$D = \min(fromDist, toDist) * 0.8$$

where *fromDist* and *toDist* are the distances of the old point from its two nearest neighbours along the same boundary. The search extent, in this case, was limited to 80% of the shortest distance to an adjacent point in the old cadastre to reduce the risk of creating intersecting shift vectors along the boundary. The same algorithm as described in Section 7.6 was then used except that an angle check value of 25° instead of the 10° was used. This point in the processing is most often reached where a salient point from a riparian boundary has remained unmatched. This point matching process has been found in practice to improve the results for such points.

7.9 Iteration on riparian boundaries

From inspection of the results on several rural datasets, it was observed that matched riparian boundaries in the new cadastre could be a much greater distance from the original than computed by the PMA3 algorithm, even after the second pass. The algorithm was, therefore, modified to incrementally increase the search distance by 20% up to a maximum of the half the square root of the parcel area or 200 metres, whichever is the shorter. This resulted in a considerable number of additional shift vectors being created but, as many of them were incorrect and many others were deleted by the outlier removal process, the modification was removed.

7.10 Iteration on unmatched points

Consideration was given to the possibility of carrying out a second point matching process after adjustment of the old cadastre. It would be possible, using the shift vectors, to adjust the point layers and repeat the point matching process for all the unmatched points; these would now be closer to potential target points. However, examination of many unmatched points has shown that these almost always occur either at locations where an operator would be required to decide on the best match, for example, where two nodes have now become one or vice-versa, or areas where there was no possibility of a match. Figure 7.17 from LGA07 shows an area with an exceptionally large number of unmatched points; for the majority, there are no corresponding parcels in the new cadastre so that no match was possible. The inset shows the new cadastre only.

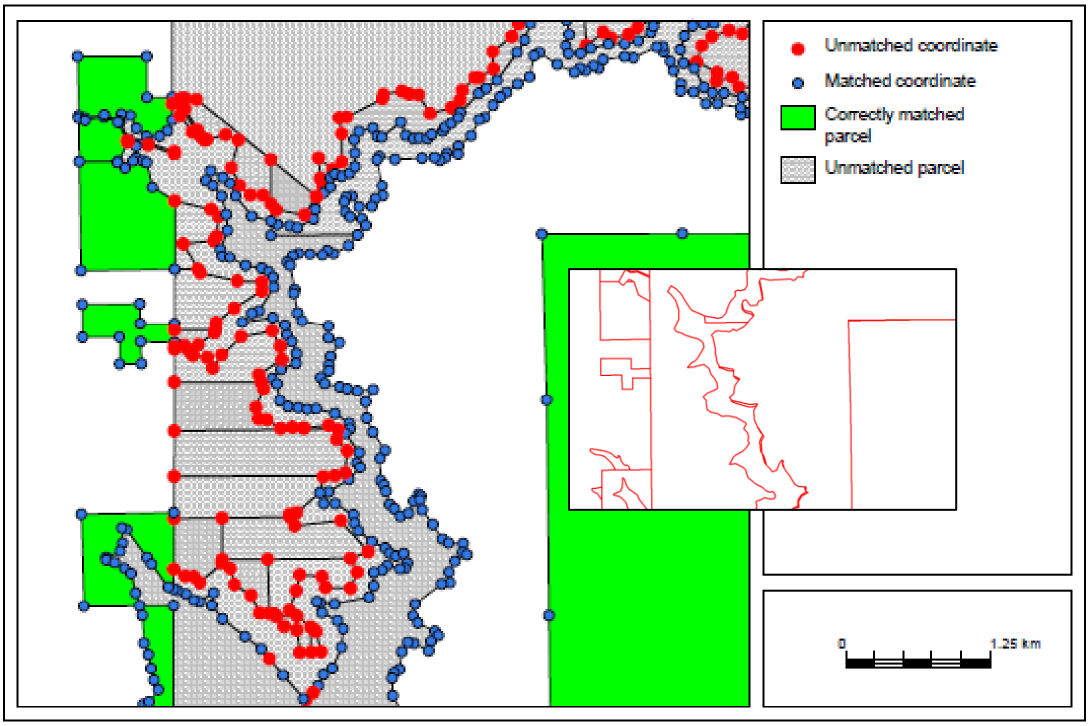

Figure 7.17 Unmatched points

7.11 Special cases

7.11.1 Isolated points

It is important to attempt to match isolated points on long boundaries for the obvious reason that even the smallest deviation from a straight line at the isolated point may represent a considerable distance on the ground. Isolated points have been defined as any point satisfying the following conditions:

$$\left(\begin{matrix} FromDist > 20 \text{ m AND } ToDist > 20 \text{ m AND} \\ SA < 180^{\circ} \end{matrix}\right) \text{ OR}$$

$$\left(\begin{matrix} FromDist > \max(MPD * 1.5, 20 \text{ m}) \text{ AND} \\ ToDist > \max(MPD * 1.5, 20 \text{ m}) \end{matrix}\right) \text{ OR}$$

$$(BPC <= 5 \text{ AND } SA < 170°).$$

MPD is the mean distance between points on the current boundary, *BPC* is the number of points on the current boundary, and *SA* is the segment angle at the point. Isolated points on unmatched boundaries that satisfied this condition were processed, in the first instance, by PMA2 and, if still unmatched, by PMA3

7.11.2 Islands

Islands are defined in this thesis as single parcels surrounded by roads or single parcels surrounded by water. PMA1 described above can deliver satisfactory results for such islands but only if the old and new digitising sessions commenced at the same vertex on the boundary. Where this is not the case the results can be very bad indeed if PMA1 is used unless steps are taken to ensure that the starting point for the boundary creation process is as similar as possible in the two cadastres. This problem was solved by locating the most south-westerly vertex on any island and starting the boundary creation process from that point, i.e. the vertex having the minimum value of $x + y$ where x and y are the coordinates of the point that was selected as the start point.

7.11.3 Long riparian boundaries

Whilst inspecting the results of adjustments on a particularly complex creek boundary, it was discovered that the use of PMA1 could give rise to a rubber-sheeting adjustment problem if only one bank of a creek was on a matched boundary and the other was not. Figure 7.18 shows an example from LGA11 where the southern bank of the creek has been matched between the old and the new cadastre but the northern bank has not. Figure 7.19 shows the shift vectors created by PMA1 on the southern bank and PMA3 on the northern bank. Figure 7.20 shows the rubber-sheeting results. Unsurprisingly, they are very poor.

This problem was addressed by excluding long riparian boundaries (more than 100 points) from the PMA1 match process. Figure 7.21 shows the rubber-sheeting results on the same area processed without the use of PMA1. This change resulted in adjustment improvements in many situations and some slight deterioration in others. On balance, the change was deemed advantageous. The use of the “100 points” number was arbitrary. A smaller value would result in fewer PMA1 matches which are typically excellent; a larger value results in fewer correct shift vectors. The value chosen is a compromise.

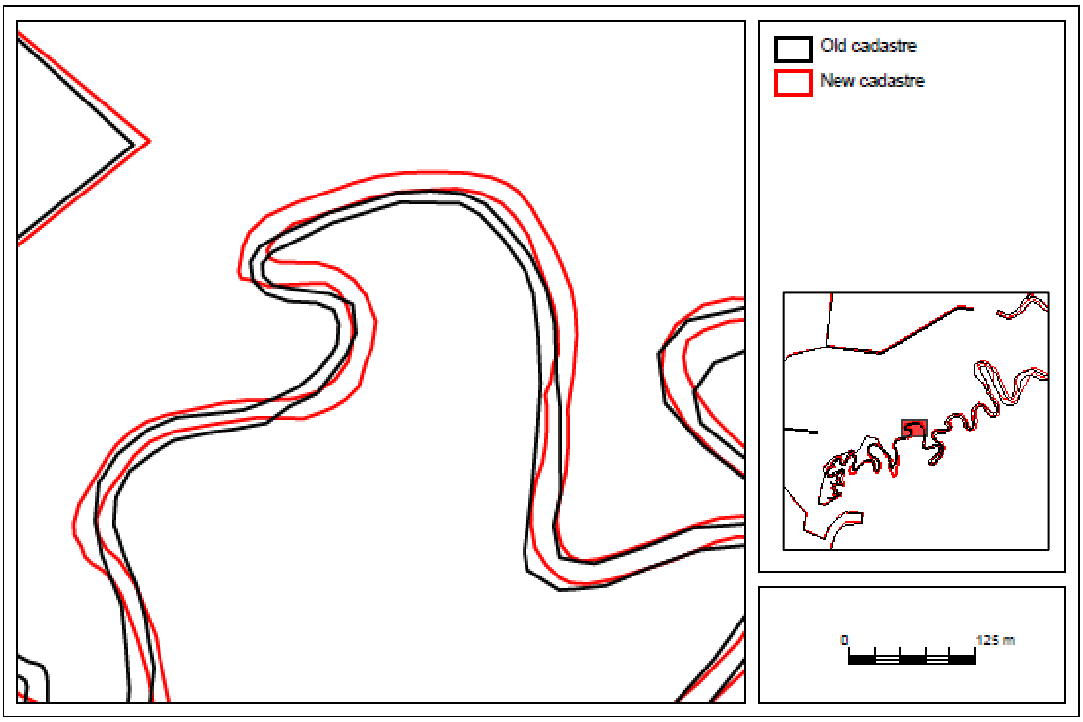

Figure 7.18 A long convoluted creek boundary

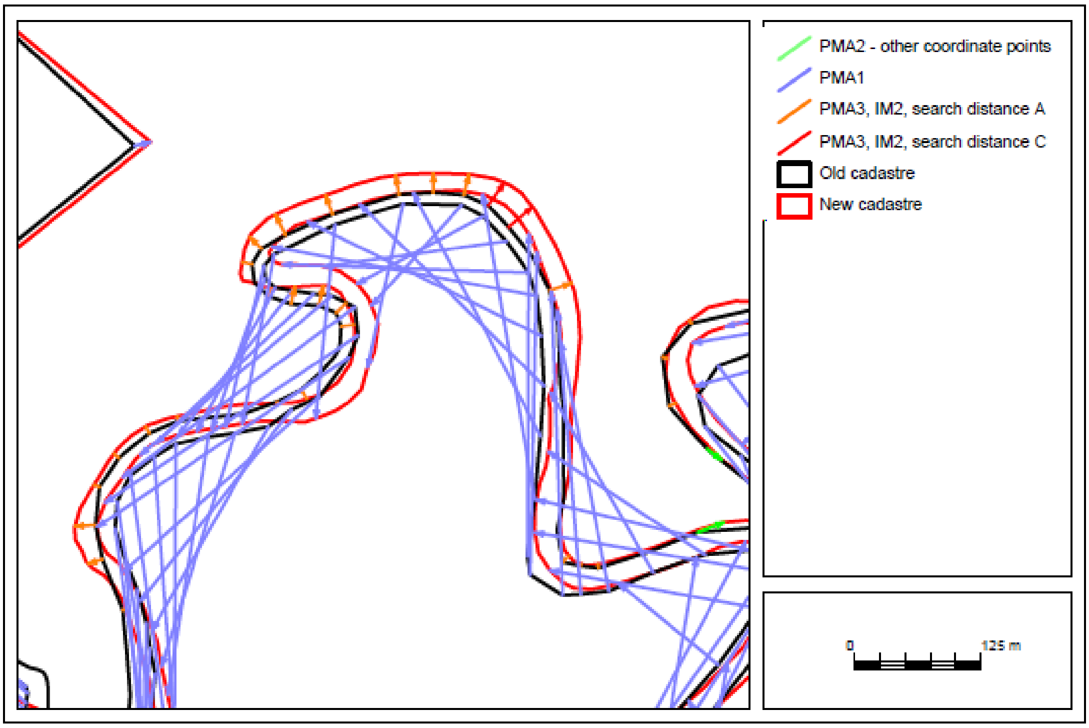

Figure 7.19 Shift vectors created by PMA1

Figure 7.20 The rubber-sheeting results using PMA1

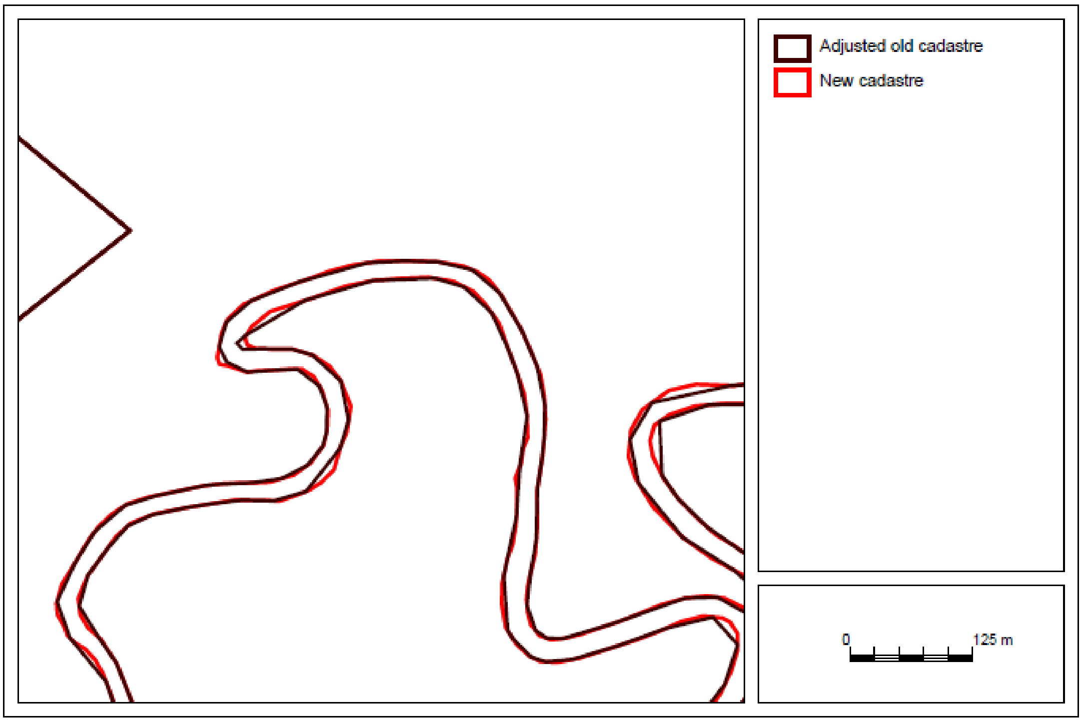

Figure 7.21 The results without PMA1

7.12 The significance of the search distance computations

Establishing that a suitable search distance for each point can be computed by considering the parcel and boundary matching information and the spatial characteristics of the point, such as the turning angle and distance from neighbours, has been instrumental in achieving the aim of this research – to simplify the spatial adjustment workflow; using these algorithms, user supplied search-distance parameters are no longer required. The tests conducted for this research have shown that large datasets covering rural and urban areas can be processed in their entirety with excellent overall results as can be seen from the quantitative results detailed in Chapter 9.

7.13 Summary

The results of the research described in this chapter have shown, to the satisfaction of the author, that a high degree of correct point matching can be achieved by considering the attributes of each point when selecting a point matching method. Point matching rules have been formulated and selectively applied depending on the attributes of the point. These rules have been finalised as the result of more than 3,600 test runs of the evolving software with each test being conducted after some aspect of a matching algorithm had been altered. The implementation of the reverse vector process allowed objective evaluation of the point-to-point matching results but, for the PMA1 and PMA3 algorithms, only the subjective opinion of the author was available for evaluation.

Observations conducted on the adjusted old cadastre have revealed that the algorithms detailed in this chapter can produce incorrect results where topology has changed and that they can sometimes result in touching, intersecting or overlong shift vectors, potentially leading to the incorrect topology of polygon layers after rubber-sheeting and incorrect adjustment of the dependent datasets. The next chapter will describe the automated methods developed to eliminate as many of these incorrect shift vectors as possible. It also describes the manual checking methods that were used in this research to locate any errors remaining after the automated error removal process.

8 CORRECTING ERRORS AND MANUAL CHECKING

Unsurprisingly, the processes described in Chapter 7 do not result in 100%-correct shift vectors although the number of incorrect vectors varies considerably depending on the type of dataset, urban or rural – see Table 8.1. It is, therefore, necessary to process the shift vectors to remove as many incorrect vectors as possible before proceeding to use them for the spatial adjustment of the dependent datasets. Error correction involves two process stages: automated and then manual.

The automated processes have been developed as part of this research project to maximise the number of correct vectors produced by eliminating as many incorrect vectors as possible. This process involves the location of all touching or crossing vectors and, in each case, removing the vector(s) most likely to be incorrect and retaining the correct ones, leaving no touching or crossing vectors.

To progress this research, it has been necessary, once the point matching algorithms were implemented, to use the generated shift vectors to adjust the old cadastre and inspect the results of that adjustment. Only by extensive use of this inspection has it been possible to gradually refine the algorithms to the stage now reached.

This chapter will describe the processes developed to accomplish the automatic removal of incorrect vectors such as intersecting or touching vectors or excessively long vectors and the manual processes used to discover the remaining errors.

8.1 Overview

Section 8.2 of this chapter describes the automated methods developed to remove erroneous shift vectors. Section 8.2.1 describes the statistical computations undertaken before running the removal algorithms. Section 8.2.2 describes the algorithm used to process sets of touching vectors and resulting in just one vector from each touching set being retained. Section 8.2.3 describes a similar process for processing vectors which cross others and Section 8.2.4 describes the processes used to identify and delete statistical outliers. Section 8.2.5 provides some additional information regarding this stage of the research and Section 8.2.6 provides statistics on the number of vectors automatically deleted by the error checking algorithms.

The remainder of this chapter discusses the manual techniques that have been be used to locate areas where errors are likely to have occurred. Section 8.3 describes the way in which the various manual error checking methods were used to iteratively improve the shift vector and adjustment results. Section 8.3.1 describes how shift vectors created by the point matching algorithms were used to rubbersheet the old cadastre and how the adjustment results were inspected using a GIS viewer in order to locate areas where the algorithms were not delivering correct results. Sections 8.3.2 to 8.3.7 describe the way in which output from the matching algorithms was used to locate areas where known errors had occurred. Section 8.3.8 describes the way in which a polygon layer was created showing all the different areas that should be checked.

8.2 Automated erroneous shift vector removal

Observation of the results from the point matching and shift vector generation process has shown that incorrect shift vectors can be identified from three distinct characteristics; vectors that cross one or more other vectors, vectors that touch one or more other vectors and vectors that are considerably longer than the mean length of other vectors in the surrounding area. If these are not removed before they are supplied to a rubber-sheeting tool, the result is likely to be a topologically incorrect adjusted layer. Algorithms were developed for each of these different situations; in the case of touching and intersecting vectors, algorithms were developed to attempt to select the best vector and eliminate the remainder. In the case of overlong vectors, the results of a statistical analysis were used to eliminate outliers. Vectors deleted during these processes were added to a new layer to be used at the manual error checking stage.

8.2.1 Preliminary processing

At the start of the shift vector removal process, the mean and standard deviation of vector lengths and azimuths were calculated for each individual block and for the entire dataset. Then, using spatial search tools, all vectors that crossed or touched another were identified and flagged. Where any vector crossed or touched a PMA1 vector, i.e. a vector created by the inferred intersect method used for points on matched boundaries (see Section 7.6), that vector was immediately removed. This is because, from inspection of results, it was observed that PMA1 vectors were rarely, if ever, incorrect. Indeed, PMA1 vectors can even be correct when they cross other PMA1 vectors on the same boundary as illustrated in Figure 8.1. For these reasons, PMA1 vectors were not removed by any of the algorithms described in the following sections except where the PMA1 match was to a new-cadastre point rather than to an interpolated point on a boundary.

The collected statistics are used extensively in all the algorithms described in the remaining sub-sections on Section 8.2.

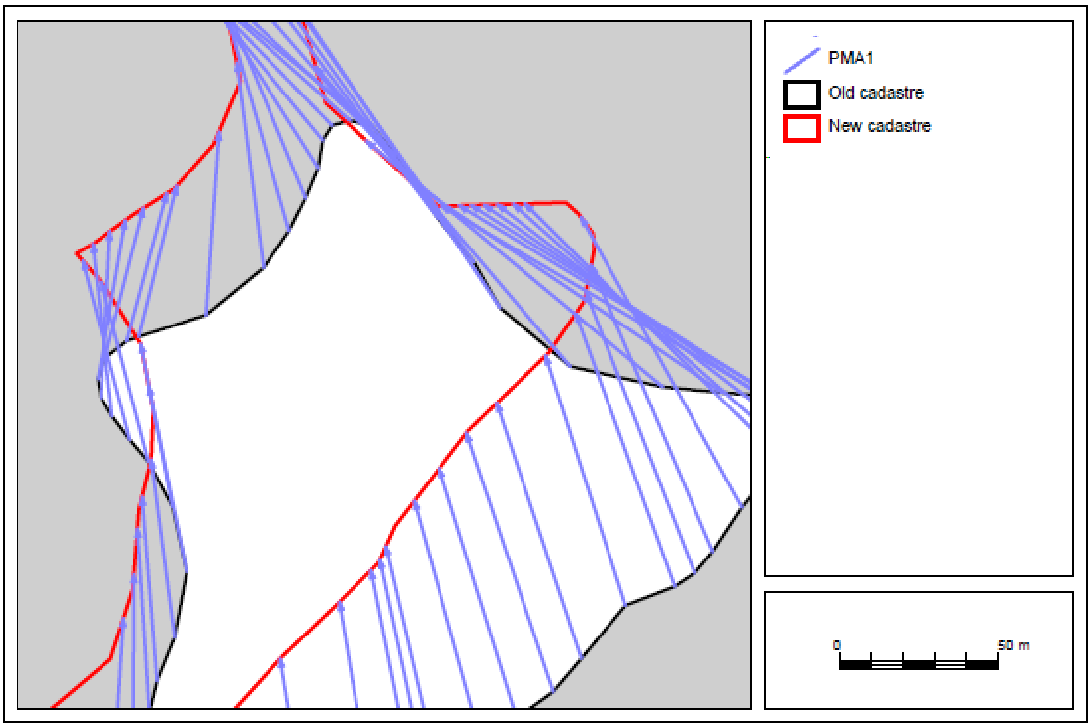

Figure 8.1 Intersecting PMA1 shift vectors on a complex coastline

8.2.2 Removing touching shift vectors

The aim of this process was to select the best, i.e. the most likely to be correct, shift vector from each group of two or more vectors that touch at a single point location in the new cadastre.

The method used to determine the best vector to retain is as follows:

(a) For each shift vector in a set of all vectors that touch at a single point:

(i) Find the difference between the azimuth of the vector and the mean azimuth for all vectors in the current old-cadastre block (*dA*) and *dA%* - *dA* as a percentage of 180°.

Find the difference between the length of the vector and the mean length of all vectors in the current old-cadastre block (*dL*) and *dL%* - *dl* as a percentage of μL where μL is the mean length of all the shift vectors created from points in the current old-cadastre block.

(ii) If *dL* is positive (i.e. the current vector is longer than the mean length of all the vectors in the block), compute a score for the current vector using the equation:

$$Score = dA\% + dL\%$$

else if dL is negative, compute the score using the equation:

$$Score = dA\%$$

(iii) Select the vector with the lowest $Score$ to be retained and eliminate the remainder.

8.2.3 Removing intersecting shift vectors

After completing the removal of touching shift vectors, the statistics for vector length and angle were re-computed as before.

The aim of the intersecting shift vector removal process was to determine which intersecting shift vectors should be retained and which discarded. In the case where one shift vector crosses one or more others and each of those crossed only crosses one, the first shift vector is always chosen for removal. This situation is illustrated in Figure 8.2 from LGA11. The blue deleted vector crosses two others, both of which have been retained.

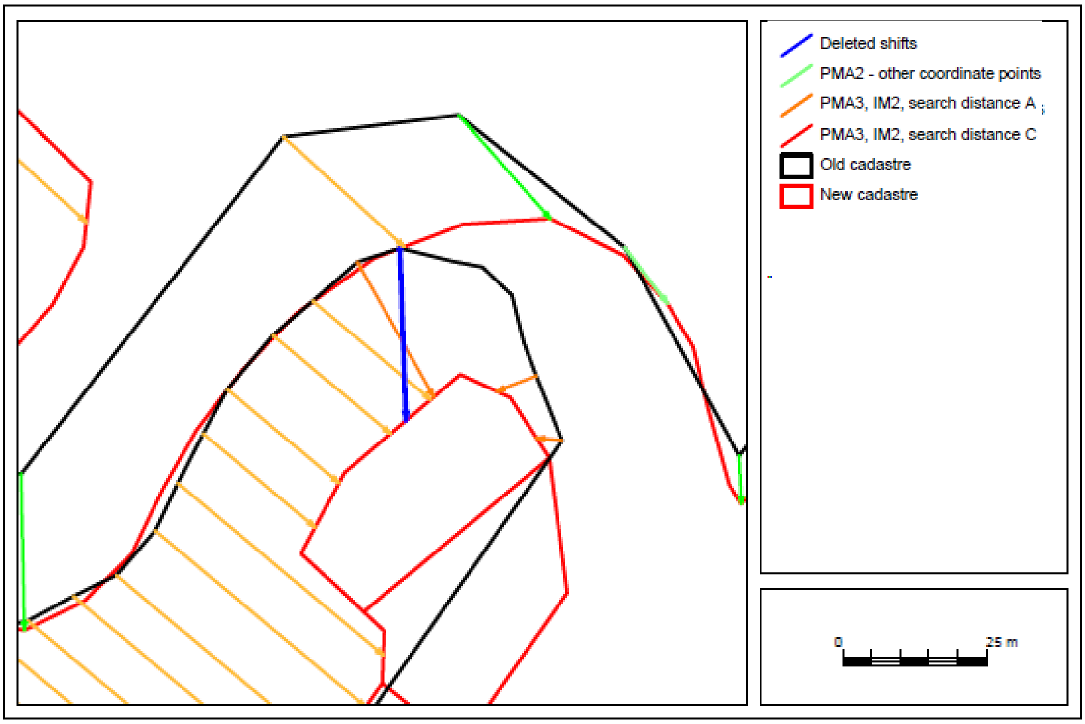

Figure 8.2 A single shift vector intersecting two others

In the case where one vector crosses several others and each of those crossed also crosses more than one other, the algorithm attempts to select the best vector from each intersecting pair for retention. Figure 8.3 from LGA11 illustrates this situation. The algorithm used to determine which vectors to delete is as follows:

(a) Where the current vector crosses several others which each cross only that vector, delete the current vector.

(b) For all other intersecting shift vectors, the algorithm described in Section 8.2.2 was used to choose which of each intersecting pair to delete.

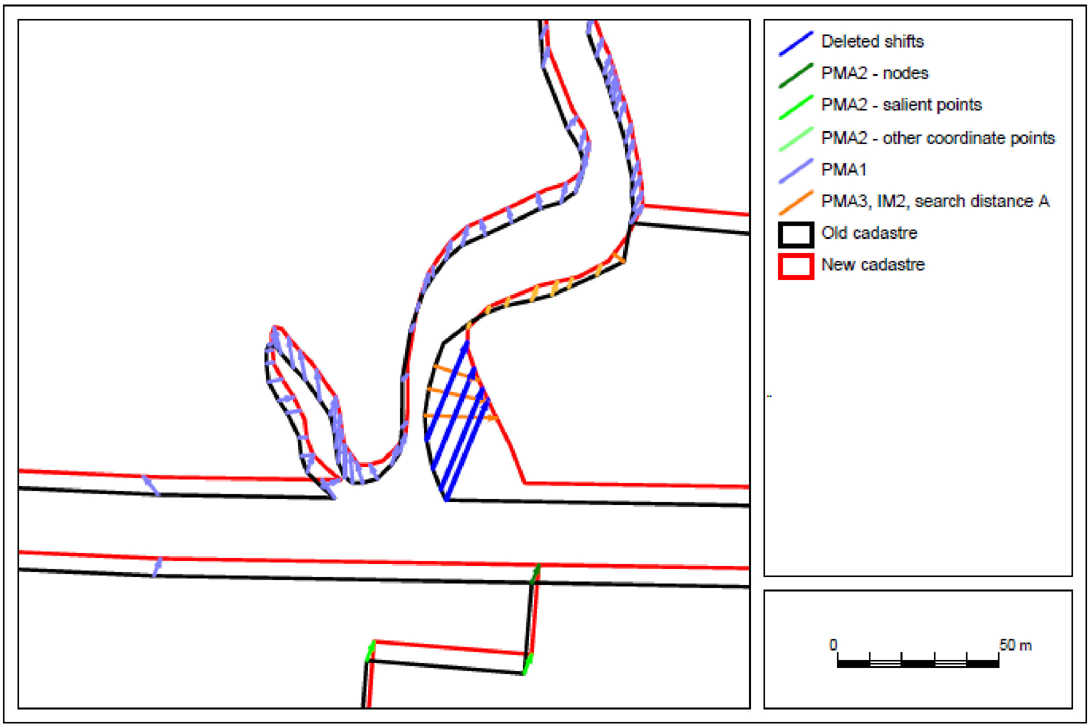

Figure 8.3 Multiply intersecting shift vectors

This process was repeated until no intersecting shift vectors remained.

8.2.4 Removing statistical outliers

The aim of this process was to discover any shift vectors which fall so far outside the mean lengths and angles for all vectors of the same type (i.e. created by the same PMA) in the same block that they are likely to be incorrect and should not be used as input to the adjustment tool. As before, the shift vector statistics for length and angle were recomputed before attempting to identify the outliers but for this process, statistics were also gathered for shift vectors created by each of the different point matching algorithms.

To identify outliers, the algorithm uses the mean length of all the shift vectors of similar type in the block if there are at least 20 vectors in that block. If there are not at least 20 vectors of the same type, the figures for all the vectors in the block are used. Where there are fewer than 20 vectors in the current block, the vector is checked against the mean length of all the vectors in the dataset. It was considered that the statistics were unlikely to be reliable where there are fewer than 20 vectors in a block. Only vectors from unmatched parcels or unmatched boundaries were checked; vectors originating from matched parcels and boundaries have been found during the many tests conducted during this research to be reliably correct.

Details of the algorithm are as follows:

(a) Calculate the difference in length between the length of the current vector and the mean length of all the vectors in the block that were created using the same PMA or, where there are fewer than 20 vectors created using the same PMA, the mean for all the vectors in the block, or where there are fewer than 20 vectors in the block, the mean for all vectors created using the same PMA in the dataset (dL).

(b) All vectors created using PMA1 are retained.

(c) A vector created using PMA2 is retained if it satisfies the following condition:

$$dL < \max(\sigma L * 3, \max(\sqrt{A}/10, 3))$$

where σL is the standard deviation of the vector lengths and A is the area of the parcel that the source point came from.

(d) A vector created using PMA3 with intersect method 1 (IM1) is retained if it satisfies the following condition:

$$dL < \max(\sigma L * 2.5, \sqrt{A}/5).$$

(e) A vector created using PMA3 with intersect method 2 (IM2) is retained if it satisfies the condition:

$$dL < \max(\sigma L * 1.5, \sqrt{A}/5).$$

Vectors not satisfying the above conditions are deleted.

8.2.5 Additional information regarding this stage of the research

As with all other algorithms described in this thesis, the thresholds used in the expressions described above have been arrived at by varying their values followed by inspecting a map of the results. The values used result from the compromise between removing too many good shift vectors and retaining too many bad ones. The final threshold values chosen have resulted from the author’s subjective evaluation of the mapped old-cadastre adjustment.

In the initial stages of the research, consideration was given to the use of Median Absolute Deviation (MAD) (Hoaglin, Mosteller, & Tukey, 1983) on the length values to discover outliers. However, this approach was not found to improve the results and it was therefore discontinued; its use significantly increased processing times.

Also, in the earlier stages of the research, the statistical-outliers algorithm described in Section 8.2.4 made use of the difference between the azimuth of the current vector and the mean azimuth of all similar vectors in the same block wherever there was a significant number of these. However, as the point matching algorithms described in the previous chapter were improved during this research, it was discovered that making use of the azimuth check for locating the outliers did not improve the results.

8.2.6 Statistics

Table 8.1 shows the number of shift vectors deleted by the automated error removal processes described in this section (Section 8.2). The **Shift vector count** column shows the count of all the non-zero length vectors after error removal. Identity points created by the point matching algorithms are not processed by the automated error removal process. The **Percentage of shift vectors deleted** column shows the number of deleted vectors as a percentage of the number of non-zero length vectors initially created by the point matching processes. The table is ordered in increasing percentages of deleted shift vectors. Unsurprisingly, in the rural areas where control point matching tends to be more difficult, even for a human operator (see Section 8.3 for examples), the percentage of shift vectors deleted tends to be greater.

Table 8.1 Counts of deleted shift vectors.

LGA ID	Shift vector count	Number of deleted shift vectors	Percentage of shift vectors deleted	Dataset type
LGA08	24609	83	0.34	Urban
LGA09	41354	378	0.91	Urban
LGA12	50138	486	0.96	Urban
LGA04	4927	52	1.04	Urban
LGA01	2270	40	1.73	Urban
LGA03	15876	315	1.95	Rural
LGA10	29766	618	2.03	Urban
LGA02	4179	90	2.11	Urban
LGA05	10323	326	3.06	Rural
LGA11	78146	3572	4.37	Rural
LGA07	36831	1790	4.63	Rural
LGA06	29220	1743	5.63	Rural
Totals	**327639**	**9493**	**2.90**	

8.3 Manual processes

The remainder of this chapter describes the various methods used by the author to locate incorrect shift vectors remaining after the processes described in the previous sections, and the way that inspection of these errors was used to improve the algorithms developed for this thesis.

It must be emphasised that whatever methods are used to carry out a spatial adjustment process – be they the use of commercial applications or the methods described here – it will always be necessary for an operator to check the results before proceeding to dependent layer adjustment. The processes described here make use of information gathered generated by the processes described in the previous chapters.

For this research, after each shift vector generation and error removal process was complete, the old cadastre for the current dataset was automatically adjusted using the ArcGIS *RubbersheetFeatures* tool. In the ideal case, the result of this adjustment would be an exact correspondence between all the old-cadastre parcel boundaries and the new ones, at least where the topology had not been changed between versions. By overlaying the adjusted old cadastre on the new cadastre in a GIS viewer, any adjustment errors become clearly visible (see Section 8.3.1).

This research has not discovered any automated methods for ascertaining that all the remaining generated shift vectors are correct. Chapter 7 has given the reasons why the PMA1 and PMA3 matches cannot be checked by the reverse vector process which is only of value on point-to-point matches. Several manual methods were therefore used; these are described in the sections below. The inspection methods described in these sections have been used extensively to iteratively improve the algorithms described in this and previous chapters.

In the initial stages of the research, areas which showed particularly poor results were extracted into smaller datasets (the DA datasets) so that these could be used for quickly checking the consequences of any algorithm changes. In all, 33 such data subsets were extracted.

In many cases, the observations described in this chapter suggested that the matching stages for blocks, boundaries or parcels needed refinement. In other cases, the observations also suggested improvements to the point matching algorithms.

The process of refining the algorithms and retesting them on the problem datasets and then the full LGA datasets numerous times has resulted in continued improvement of the algorithms up until the point where it was found that the remaining errors could not be eliminated without adversely affecting results in other areas. Overall, more than 6,000 test runs have been executed, not including tests run during the GA research. Each test was undertaken after a small algorithm change in order to ascertain whether the change improved the results from that algorithm. Six hundred of these tests, for evaluation of the parcel matching and boundary matching algorithms, were executed before the point matching component of the solution were implemented. Of the remainder, not all test runs resulted in shift vector creation, either due to coding errors or because a run was terminated early by the author, typically after observation of the behaviour of the code on a parcel, boundary or point that was being incorrectly matched. However, more than 3,600 tests, where the adjustment results could be inspected, ran to completion.

8.3.1 Inspecting results in a GIS viewer

The most effective shift vector validation method so far discovered has been to examine the result of carrying out an adjustment of the old cadastre and overlaying the adjustment output on the new cadastre in a GIS viewer. Saalfeld (1993) describes an early attempt to automate the map conflation process and comments on the effectiveness of visual review in all his test cases.

With a suitable choice of symbology, areas which have failed to adjust correctly can be revealed. Using an output scale appropriate to the area being examined, any incorrect adjustments show up clearly. In Figure 8.4 covering a rural area from LGA07, the visible new-cadastre parcel boundaries (red) reveal locations where the adjusted old-cadastre parcel boundaries (black) do not fall in same location; the inset shows the same area before adjustment.

Visual inspection of the adjusted old cadastre drawn on top of the new cadastre in a GIS viewer has been adopted as the primary method for the discovery of errors because, it is presumed, a set of shift vectors that delivers a perfect adjustment on the old cadastre will deliver a correct adjustment on all the spatially dependent layers except perhaps PMA1 adjustments on certain point based dependant layers – see 11.3.7.

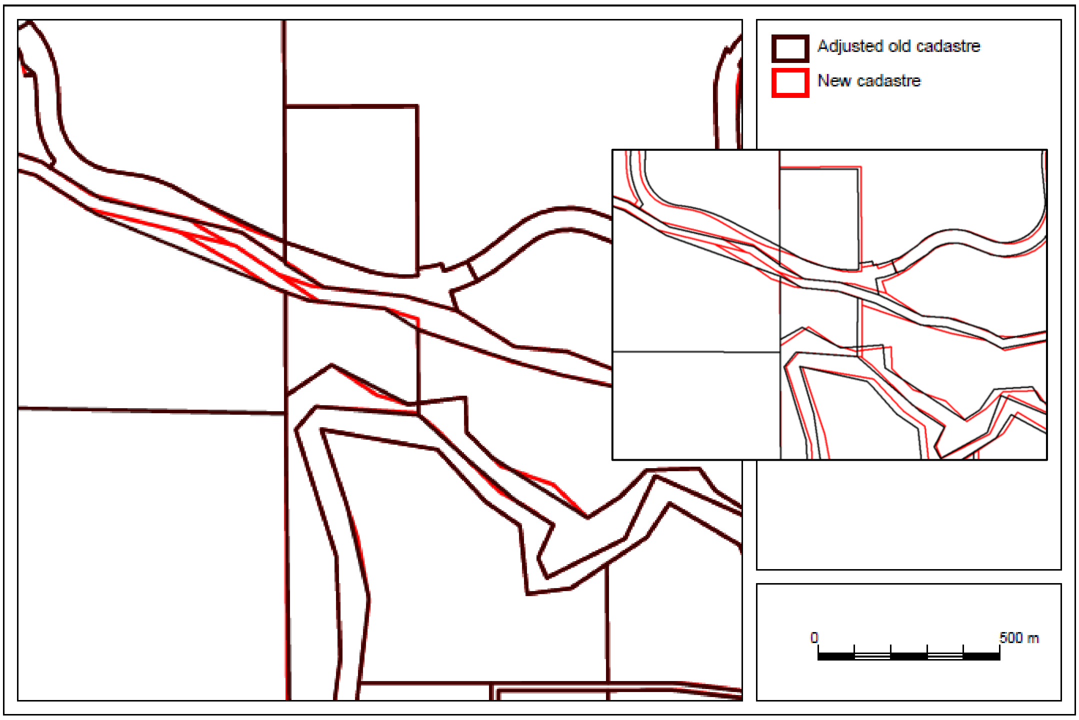

Figure 8.4 Several clearly revealed adjustment failures

Visually inspecting the entire dataset for badly adjusted areas is not practical unless the datasets are small. The methods outlined in the next sections, therefore, were used to locate areas where errors were expected. Throughout the research these techniques have been used to locate areas where the algorithms were performing poorly and to guide the gradual improvement of those algorithms. These same techniques could also be used to guide the operator of a developed solution to locations where shift vectors need to be removed, added, or replaced prior to adjusting the spatially dependent layers.

8.3.2 Inspection of locations where shift vectors have been deleted

Inspection of locations where shift vector deletions have occurred has shown that incorrect shift vectors usually arise in areas where there is a very poor match between the old and the new cadastre, for example, areas where there has been a subdivision undetected by the algorithm described in Section 4.3.4 or areas where the topology differs substantially. Figure 8.5 shows an undetected subdivision from LGA07; in this case the area test for matching subdivided or amalgamated areas failed because part of the area of the subdivision (shaded in grey) has now been absorbed into the road void so that this area was not included in the area-test calculation used to identify subdivisions. The deleted shift vectors drawn in blue have correctly been deleted.

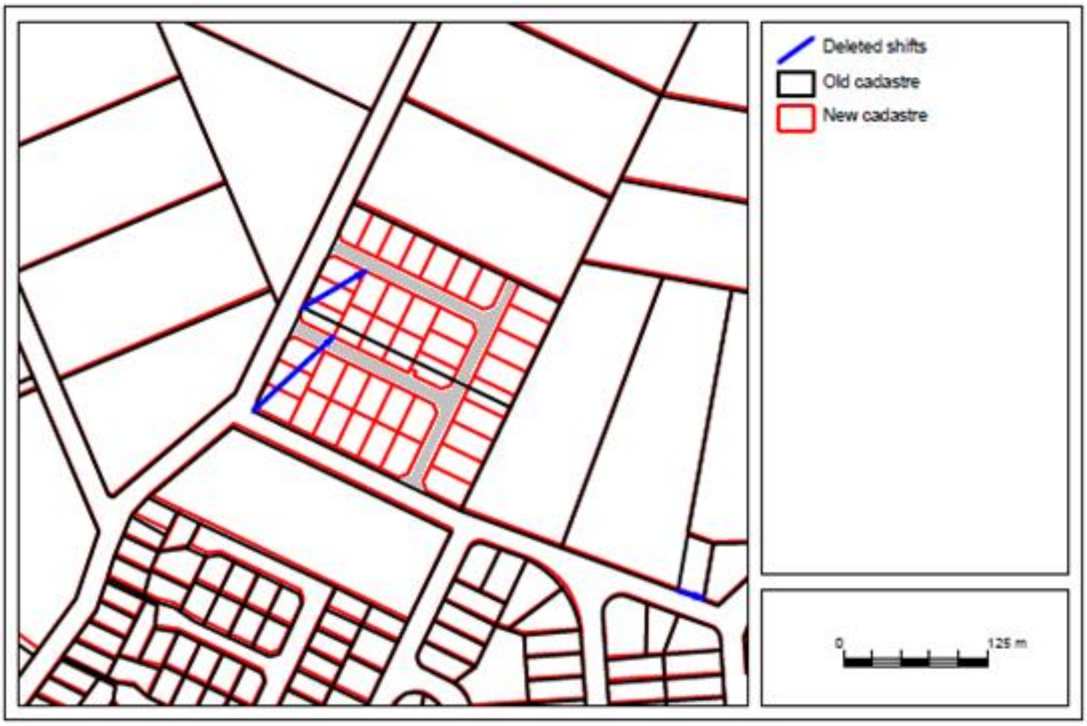

Figure 8.5 An area with an unrecognised subdivision

Figure 8.6 shows an area from LGA10 where the topology differs between the old and new cadastres. This area has been subjected to a complex process of amalgamation and subdivision between versions such that there is no precise correspondence between corners and nodes. The software has correctly identified and deleted several incorrect shift vectors (drawn in blue).

Figure 8.6 An area showing mismatched topology

Locations where shift vectors have been deleted by the processes described earlier in this chapter have usually been found by observation to be areas of especial difficulty, such as undiscovered subdivisions and parcel mismatches. Whenever possible, after examining areas where shift vectors had been deleted, the algorithms were further refined to eliminate the errors, although in the later stages of the research most of these areas were found to be areas where the author would also have difficulty. Figure 8.7 illustrates an example of one such area from LGA07 where there have been many boundary changes between the two versions. The deleted vectors were clearly incorrect, i.e. they have correctly been deleted but it is not at all clear to the author how most of the vertices should be matched. In this case, inspection of the deleted vectors did not rise to any new insights regarding algorithm improvement.

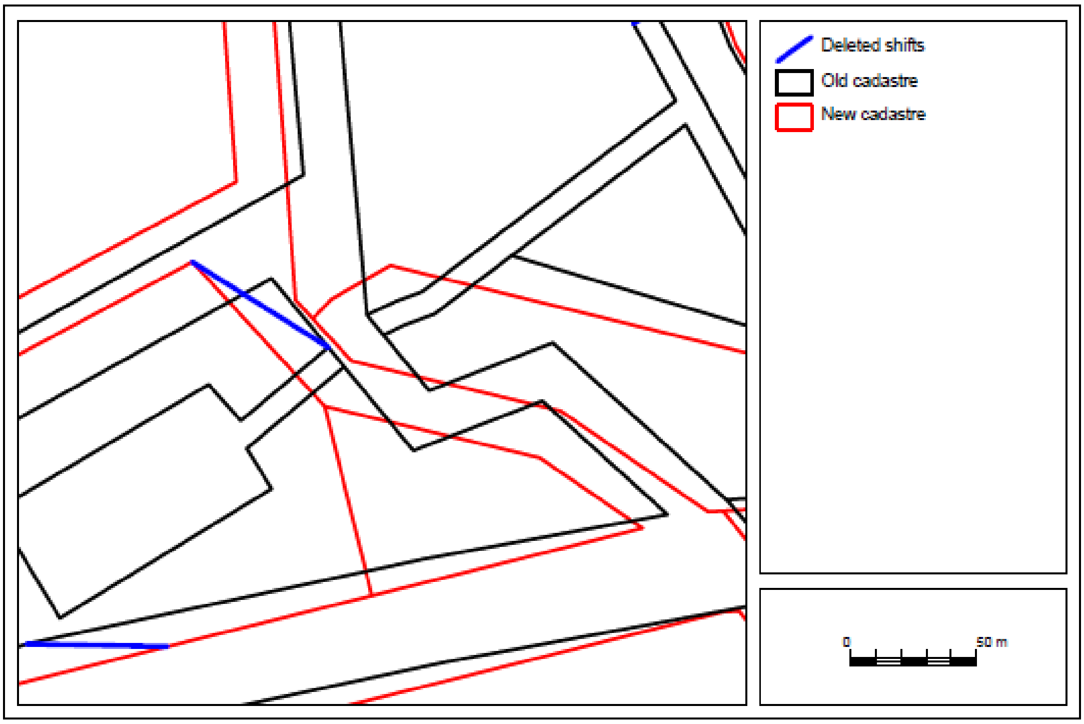

Figure 8.7 A difficult area

Figure 8.8 shows another area where shift vectors have been deleted. Here the two unlinked nodes highlighted in red have switched places between cadastral versions resulting in intersecting shift vectors; the inset shows a larger area around the point. This has been found to be a not-uncommon situation in urban areas and would not require correction by an operator unless there was an impact on a spatially dependent dataset. A similar point matching failure occurs when a single node in one cadastral layer appears as two in the other, again, in most cases, not a situation requiring correction. However, should correction be deemed necessary, any software developed to facilitate the correction process could provide a tool to allow a deleted vector to be returned to the set of correct vectors before readjusting the datasets.

Checking of the deleted shift vectors has proven to be of significant value in urban areas for the location of small topological changes, such as the one illustrated in Figure 8.8 below from LGA07, where either the new or the old parcel boundaries may be incorrect. As with the parcel matching process, this information could potentially be used to aid the quality improvement of the cadastral dataset. The inset shows the location of the problem in a larger area for context. Topological changes of this type have been found in many urban areas.

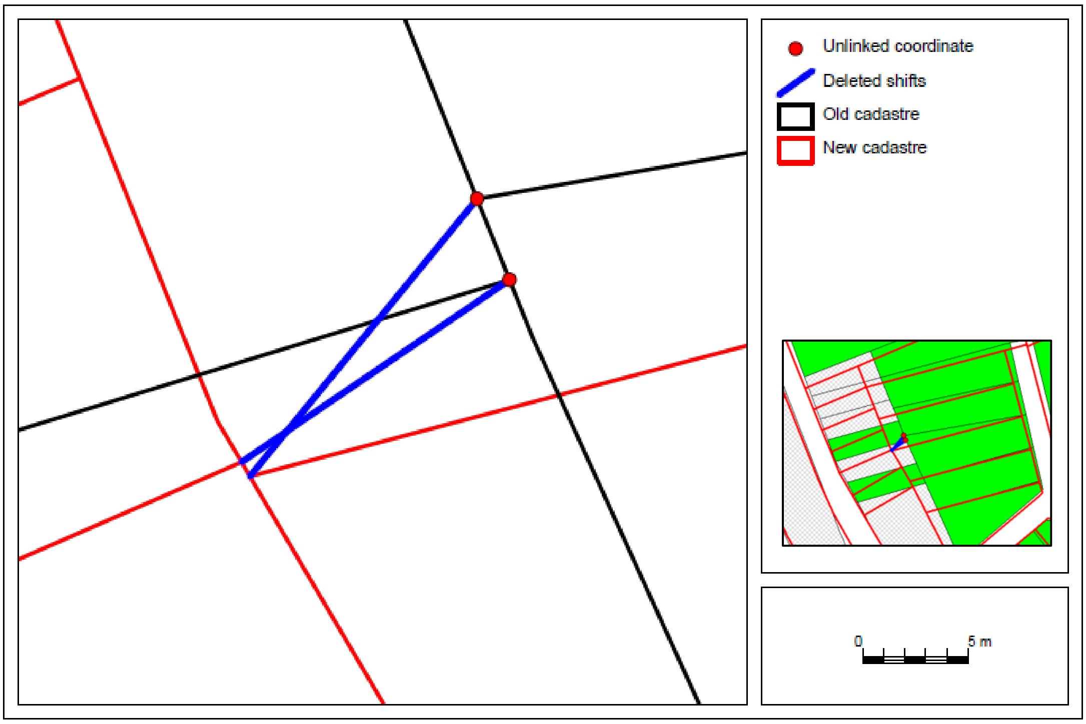

Figure 8.8 Topological changes at nodes

8.3.3 Inspecting parcels which should have been matched

Where there are existing parcel UIDs, the unmatched, wrongly matched, and unexpectedly matched parcels have often been found to indicate areas where incorrect shift vectors have been created. Inspection of these areas revealed that it was unsafe, where parcels were known to be wrongly matched, to use the generated parcel centroid shift vectors to guide the shift vector creation process. After this observation, the algorithms were altered to calculate the parcel shift vector length and azimuth for the wrongly matched and unmatched parcels from a mean value from surrounding matched parcels (see Section 4.3.6.4).

Where there are no parcel UIDs, all unmatched parcels were treated in the same way as the wrongly matched and unmatched parcels from a dataset containing UIDs. Figure 8.9 and Figure 8.10 show the same area (fromLGA01) before and after the algorithm change. In the second image, all the vectors are correct.

It is interesting to note that the visual 3D illusion created by many correct vectors in an area tends to draw the eye to vectors that are incorrect, especially in urban areas.

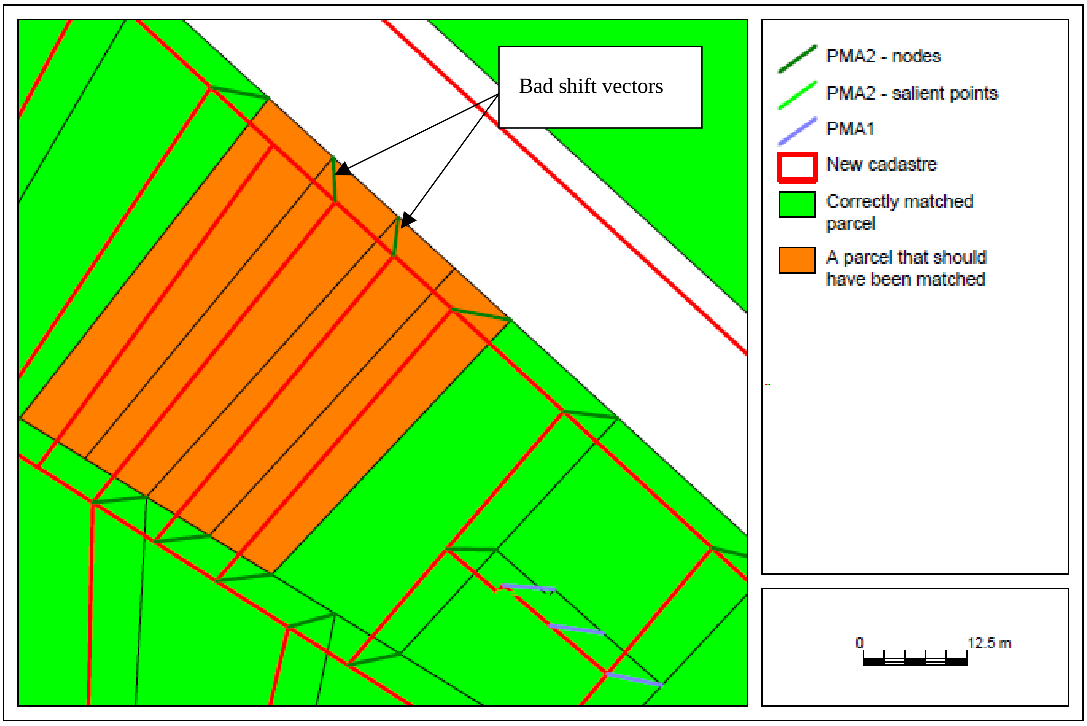

Figure 8.9 Incorrect shift vectors on unmatched parcels

Figure 8.10 Correct shift vectors on the unmatched parcels

8.3.4 Inspection of locations where the areas of matched parcels do not match after adjustment

Where parcels have been matched, it is to be expected that, after adjustment, the areas of matched parcel pairs should be almost identical. For this check, adjusted matched-parcels, where the adjusted area differed by more than 5% from the corresponding new-cadastre parcel, were selected for inspection. Figure 8.11 shows an area from LGA07 where such a parcel has been shaded in blue. The adjusted old-cadastre boundaries are drawn in black and the new-cadastre boundaries in red.

Figure 8.11 An area mismatch

Figure 8.12 shows the same area as above but also shows the unadjusted old-cadastre and the shift vectors. In this location, it would be necessary for a user to delete several of the shift vectors to improve the adjustment although it is not at all clear quite how the old and new parcels relate to each other. When adding and removing shift vectors it would be preferable for the operator to be able to display any spatially dependent dataset that is listed for adjustment. Only then would it become clear whether adjustment is needed at that location. The inset shows the new cadastre only, helping to illustrate why the results were poor in this area; many of the old-cadastre boundaries are missing in the new cadastre.

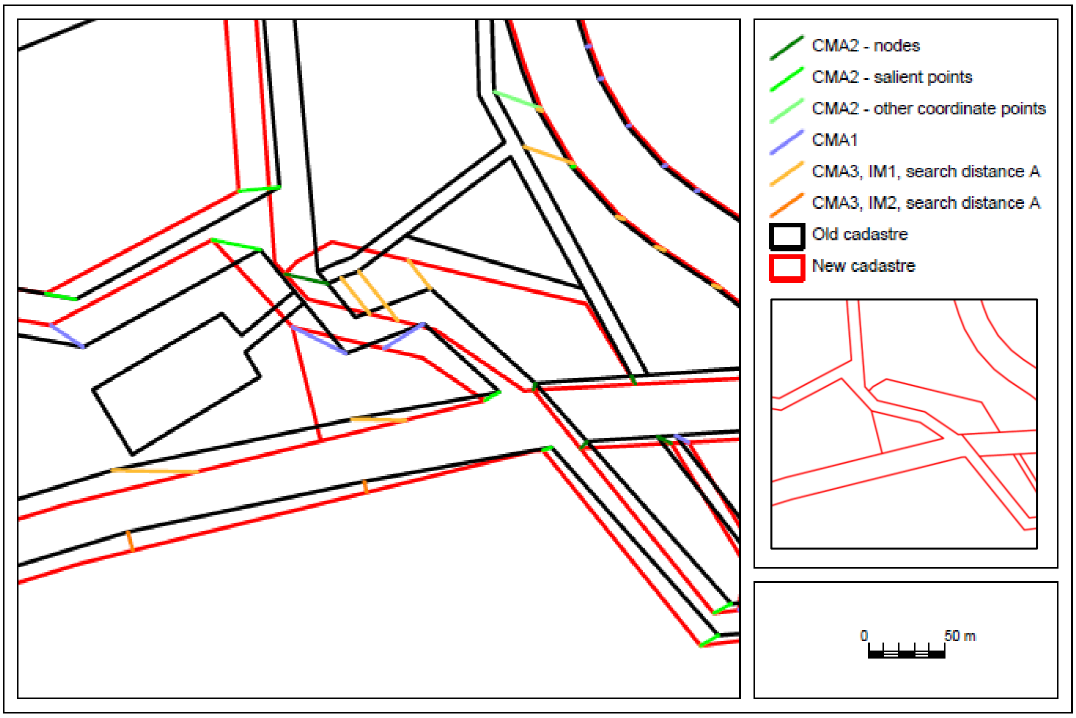

Figure 8.12 The same area (unadjusted) as the previous image showing shift vectors

Inspection of matched parcels where areas did not match after adjustment, though advisable in a developed application, did not give rise to many insights helpful in improving the algorithms. This is because most of the areas inspected proved to be especially difficult areas, such as the examples in this section, or areas where a matched parcel was adjacent to a creek. Because creeks can move a considerable distance over time, it is not surprising that the area of the adjusted parcel does not always match its pair very closely.

8.3.5 Inspection of areas where points have not been matched.

The point matching algorithm attempts to match all points with a segment angle of less than 198.8° and isolated points with a segment angle less than 180°. Inspections were therefore performed on areas with unmatched points. Figure 8.13 shows the same area as in Figure 8.7 with the unmatched points highlighted in red. The large number of missing point matches in this area is due to the topology changes illustrated in Figure 8.12 and the fact that several parcels no longer exist in the new cadastre.

Highlighting unmatched nodes has been particularly helpful during this research process. In general, it is expected that most nodes should be matched, especially in urban areas that have not been subjected to subdivisions. Inspection of unmatched nodes is therefore particularly important.

Inspection of unmatched points often resulted in ideas for improving the matching algorithms, for example, the selection of different search tolerances for different point types and the realisation that differentiation between point types could guide the choice of point matching method.

Figure 8.13 An area with many unmatched points

8.3.6 Inspection of areas where no identical reverse shift vector has been identified

As suggested by Siriba et al. (2013) a new-cadastre to old-cadastre point matching process was executed, followed by a process to identify all the old-to-new shift vectors that were not identical to one in the reverse shift-vector set. In principle, if the point matching algorithms are correct, every point-to-point match should result in an identical shift vector in the reverse direction. Flagging the identical reverse vectors allowed unflagged vectors to be located for identification of potential areas of error. Only unmatched vectors create by the PMA2 algorithm need to be checked in this way. PMA1 and PMA3 vectors do not link two points so cannot result in identical reverse vectors.

Figure 8.14 shows an area from LGA11 where several PMA2 shift vectors, drawn in yellow, have not resulted in an identical reverse vector, thus drawing attention to the fact that there has been a topological change in this area, i.e. the creek has been modelled as a single centreline in the new cadastre whereas, in the old cadastre, the creek banks have been digitised. The inset shows the resulting poor adjustment – the adjusted old cadastre (black) has been drawn on top of the new cadastre (red). This inset in this figure illustrates the importance of carrying out a manual check for shift vector errors before carrying out any adjustment of dependant datasets.

Figure 8.14 Non-identical PMA2 shift vectors

8.3.7 Checking the adjusted block boundaries

If an adjustment process were to be executed on the old-cadastre blocks, there would always be the possibility that the block boundaries would not coincide exactly with the new-cadastre block boundaries. To provide an additional means of error checking, an ArcGIS process was developed to adjust the old-cadastre blocks and highlight areas where the block boundary adjustment was poor. The process, described below, creates sliver polygons which vary in size according to the extent of the adjustment error. Identification of areas where large slivers have been created in this way has been used extensively in the refinement of the algorithms.

Using the ArcGIS *RubbersheetFeatures* tool, the block boundaries extracted from the old cadastre as described in Section 4.2.4 were adjusted using the shift vectors created by the research algorithms. If the shift vectors were 100% correct, the two boundaries should exactly coincide. To find where they did not, the following procedure, illustrated in Figure 8.15, was executed:

(a) Erase the new cadastral block layer using the adjusted old cadastral block layer to create a layer containing slivers where the block boundaries do not coincide.

(b) Erase the adjusted old cadastral block layer using the new cadastral block layer creating another layer containing slivers where the block boundaries do not coincide.

(c) Append the two layers of slivers.

The *Erase* tool acts like a "cookie cutter" so that each input block to the process creates a new layer containing any areas of the erased layer that fall outside the block boundaries. By repeating the process in reverse, all areas where there is a lack of correspondence between the old and the new block will be captured. The slivers can then be inspected to discover locations where the results are not perfect.

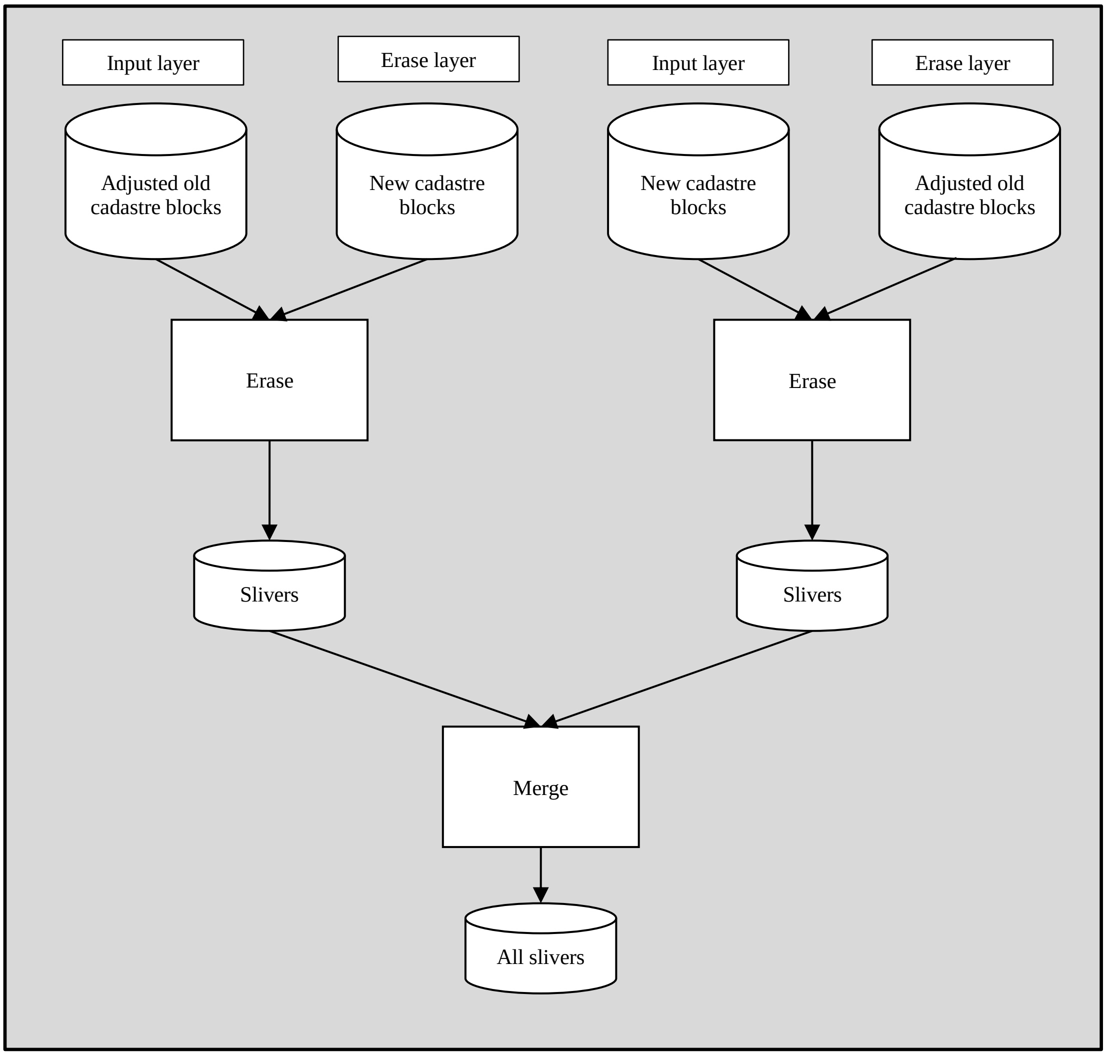

Figure 8.15 The sliver creation process

Figure 8.16 from LGA11 shows several slivers (yellow) created by using the processes described above. The location is a road junction in a rural area. The inset shows the old cadastral blocks before adjustment (shaded grey) with new-cadastre boundaries overlaid in red. Poor adjustment in this area is because the new cadastre has been captured at a higher resolution than the old, i.e. there is a paucity of vertices in the old cadastre to act as source points for the shift vectors. In the rural areas of the cadastral datasets, where boundaries have changed between cadastral versions, it was observed that it was uncommon for there to be similar numbers of vertices in the old and the new cadastre. See Section 11.3.9 for discussion about how this problem can be addressed.

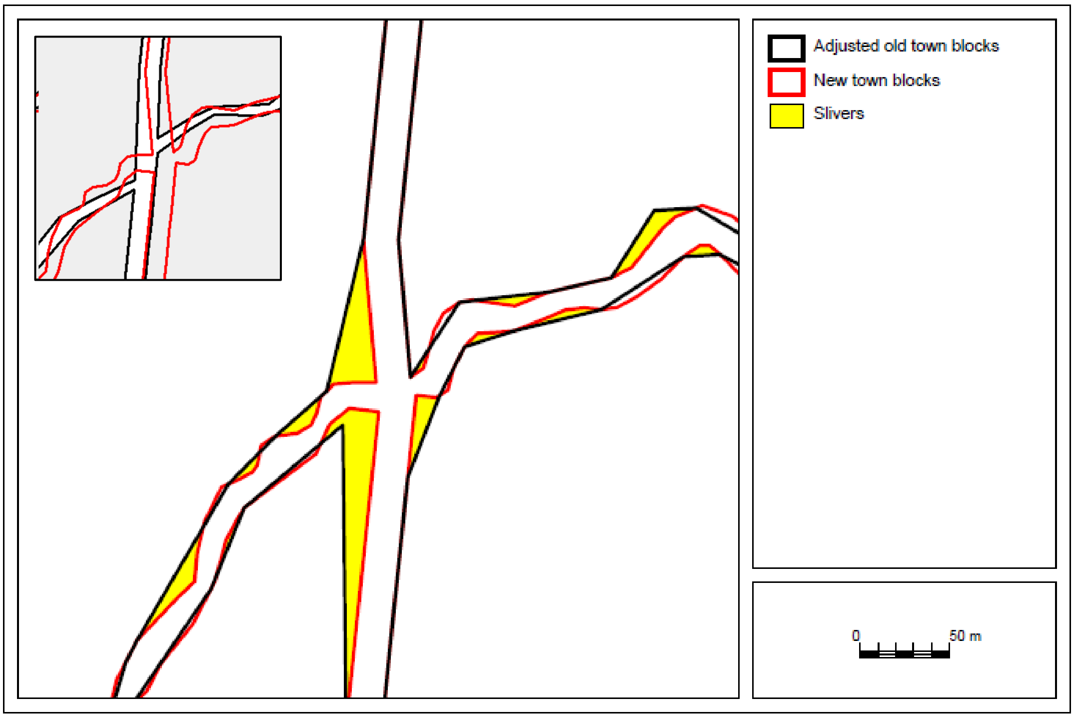

Figure 8.16 Slivers on block boundaries

Along riparian boundaries, exceedingly small slivers are often created; adjusted coastlines tend to result in very long, thin slivers. When used to aid the location of errors, slivers with a circularity index (see Section 4.3.2.3) of less than 0.05 were excluded from the checking process as were slivers with areas of less than one square metre.

Inspecting the slivers has shown, not unexpectedly, that riparian boundaries are the most difficult to match correctly. Fortunately, except where spatially dependent dataset boundaries need to follow them exactly, it may often be unnecessary to correct such errors.

Figure 8.17 however, does show an area from LGA07 where shift vector correction would be necessary. The planning zone is bounded on one side by a creek. Unfortunately, in the old cadastre, only one bank of the creek has been captured whereas in the new cadastre both banks are represented. The result of this is a topological inconsistency which the point matching algorithms cannot resolve. Shift vectors from the old-cadastre single boundary have sometimes been connected to the left bank and sometimes to the right in the new cadastre. It would be necessary, therefore, for the operator to determine which of the two banks in the new cadastre corresponds to the one in the old cadastre, and to add and remove shift vectors as appropriate. Without these corrections, the result of the adjustment of the planning zones would be as shown in the inset in the figure.

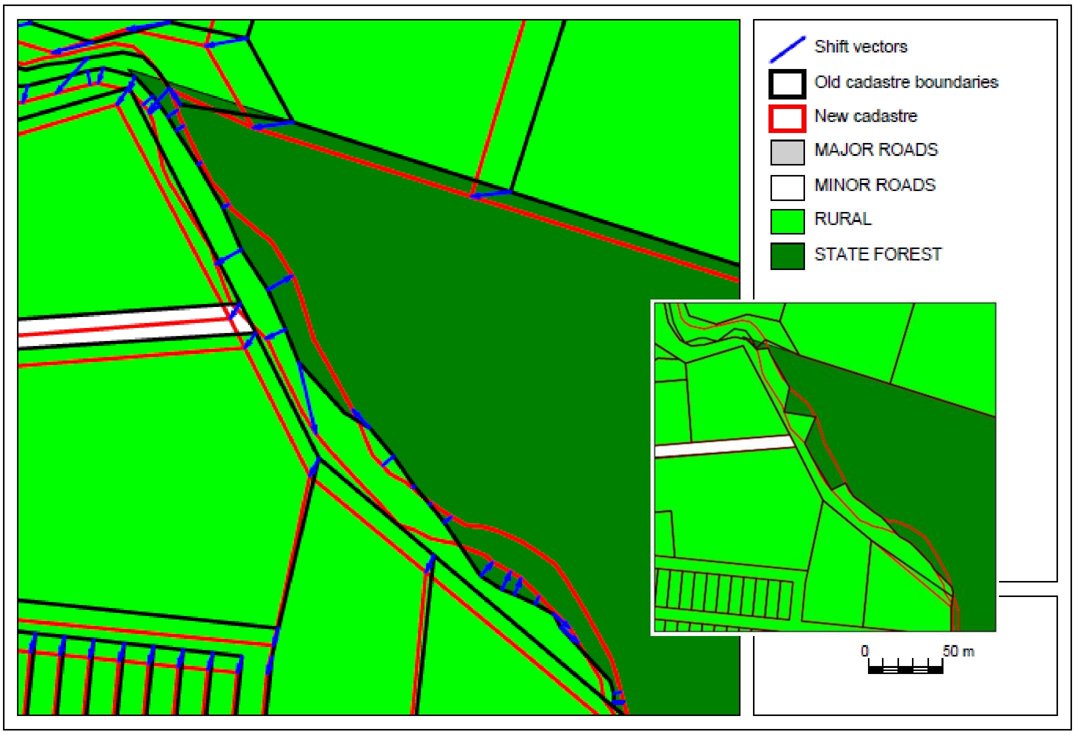

Figure 8.17 A planning zone bounded by a creek

During this research, the checking of errors on riparian boundaries has been important in the refinement of the point matching algorithms, particularly the PMA1 and PMA3 algorithms. The error checking process also gave rise to the realisation that, if rural road casements could be distinguished from other riparian boundaries, the algorithms could be further improved (see Section 6.5).

8.3.8 Combining the areas to check

As can be seen from many of the images in this chapter, difficult areas tend to have several distinct types of potential errors. If an operator were to be required to check all these areas individually, the workload would be heavy on a large dataset. For the purposes of this research, deleted shift vectors and unmatched points were buffered by 20 metres and all the areas of potential error were combined into a single layer for checking.

Figure 8.18 again shows an area from LGA07 around the difficult area shown in Figure 8.7. All the unmatched points in the area have been buffered, as have the deleted shift vectors. These have then been appended to unmatched parcels, area mismatches, and block slivers, and the entire layer then dissolved to remove internal boundaries. This method has been used extensively during this research. However, it was noticed that the combined layer was not so useful if exceptionally large rural parcels were included as the errors may fall in only a small part of the parcel. Subsequently, only unmatched points from unmatched parcels having fewer than 100 vertices were included in the layer.

Figure 8.18 Combined areas to be checked

The combined areas approach significantly reduced the number of individual areas that needed to be checked.

8.4 Summary

This chapter has detailed the algorithms developed for automatically locating and eliminating bad shift vectors. Where the deleted vectors have been inspected using a GIS viewer they have been determined by the author as having been correctly deleted. Several of the figures in this chapter, for example, Figures 8.2 and 8.3, and Figures 8.5 to 8.8, all illustrate examples of correctly deleted shift vectors. In view of the size of the datasets involved, the inspection has been limited to only some of the deleted vectors, particularly in the difficult-area datasets.

The chapter also described the methods used to manually inspect the adjusted old cadastre to locate possible areas of error using several products resulting from the algorithms, i.e. the deleted shift vectors, the unmatched points, the matched-parcel area mismatches, and the slivers resulting from incorrect block adjustment. The chapter has also discussed the ways in which the discovery of incorrect point matching results has been instrumental in the iterative improvement of all the algorithms used in the final solution.

The next chapter will discuss the results obtained using the algorithms described so far and several unsolved point matching problems.

9 RESEARCH RESULTS

This chapter presents the results achieved using the algorithms developed for this research and encapsulated into a single computer program – ShiftGen. The program has resulted in more than 90% of points of all types being matched to a point or appropriate location in the new cadastre.

9.1 Overview

Section 9.2 graphically presents a broad outline of the way in which the algorithms developed for this thesis have been encapsulated into the ShiftGen program. Section 9.3 provides quantitative and statistical results from the parcel matching, point matching and shift vector creation process, adjustment area checking, and a summary of the results from all the completed test runs.

Section 9.4 provides example maps from typical and difficult areas from the test datasets showing successful results in these areas. Section 9.5 compares the results obtained from ShiftGen with those resulting from the use of ArcGIS tools alone, including many comparative maps. Section 9.6 describes how each of the research objectives were accomplished and Section 9.7 summarises the solution to the specific problems listed in Section 1.5.

Section 9.8 briefly discusses the manual error checking stage of the research and Section 9.9 lists some unexpected outcomes that may be of interest to cadastral custodians. Section 9.10 discusses unsolved problems.

9.2 Solution outline

The algorithms detailed in Chapters 4 to 8 have been encapsulated into a single executable prototype named ShiftGen, developed solely for the purpose of testing the algorithms. Figure 9.1 illustrates a broad outline of the individual processes implemented in ShiftGen on the left and the resulting outputs on the right. The chapter in which each process is detailed is also shown.

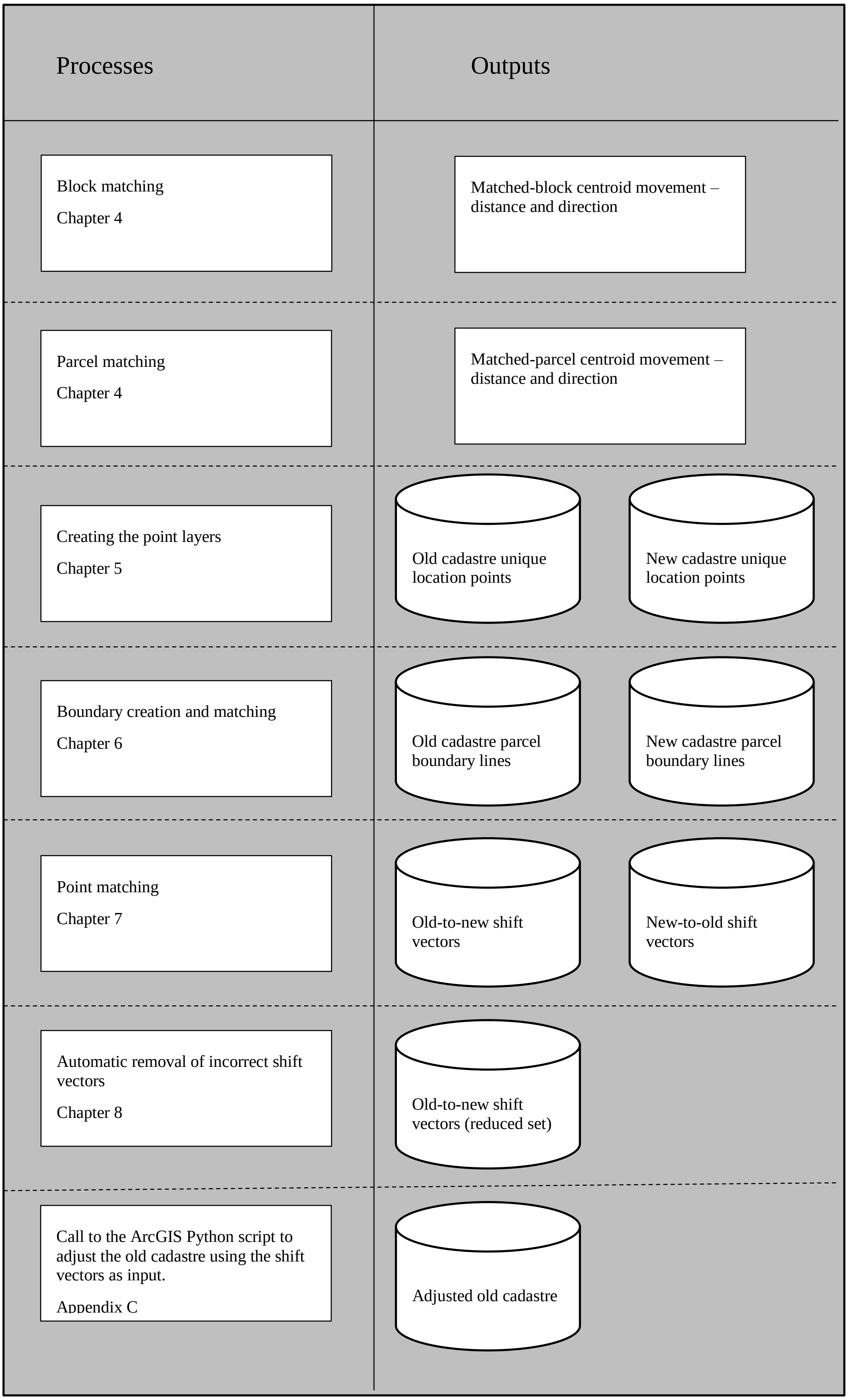

Figure 9.1 The software solution (ShiftGen) – broad outline

9.3 Quantitative results

9.3.1 Parcel matching results

The tables in this section have already been displayed in Chapter 4 but are included again here as a convenience for the reader.

Table 9.1 shows the parcel matching results of datasets with UIDs where the correctness of a match could be objectively evaluated from existing UID values stored in the dataset attribute tables. The percentage of parcels correctly matched was calculated from the number of correctly matched parcels as a percentage of the number of parcels that were expected to be matched, i.e. all regular parcels whose existing UID values were present in both cadastres - the regular parcel count minus the number of parcels where a match was not expected because there was no matching UID in the new cadastre.

Table 9.1 Match type counts for datasets with UIDs

Thesis ID	Regular parcel count	Number of correctly matched parcels	Number of unexpectedly matched parcels	Number of parcels that should have been matched	Number of wrongly matched parcels	Number of parcels where no match is expected	% Correctly matched
LGA01	653	597	21	15	10	10	92.85
LGA02	1018	983	10	25	0	0	96.56
LGA08	7232	6877	24	43	12	276	98.86
LGA11	12345	11590	116	418	69	152	95.05
Total	**21248**	**20047**	**171**	**501**	**91**	**438**	**96.33**

Table 9.2 shows the matching results for datasets without UIDs where only inspection of the thematically mapped results was available for checking the correctness of the matches. These results have been discussed in detail in Section 4.4.

Table 9.2 Percentage of parcels matched for datasets without UIDs

LGA ID	Regular parcels	Matched parcels	Unmatched parcels	% matched parcels	Dataset type
LGA03	2178	2085	93	95.73	Rural
LGA04	2256	2211	45	98.01	Urban
LGA05	2710	2626	84	96.90	Rural
LGA06	5064	4756	308	93.92	Rural
LGA07	5487	5211	276	94.97	Rural
LGA09	8550	8395	155	98.19	Urban
LGA10	11764	10793	971	91.75	Urban
LGA12	20916	20441	475	97.73	Urban
Totals	**58925**	**56518**	**2407**	**95.92**	

9.3.2 Point matching results

Table 9.3 shows the results obtained by running the ShiftGen program on all twelve LGA datasets (described in Section 3.2). Inspection of the results using a GIS viewer has shown that only very few point matches are incorrect and that these can usually be located by the methods described in Section 8.3. The table lists the number of points available for matching (Point Count), the number of those points that were matched and that number as a percentage of all points available for matching.

The slightly lower number of matched points in LGA06 was found to be caused by the large numbers of river and creek boundaries in the dataset; in many cases the movement was large and in many others the representation (one bank or two) was not the same in the two cadastral versions, i.e. there were many points that would be impossible to match, even using a manual process.

Table 9.3 Control point matching results

LGA ID	Point count	Number of matched points	Percentage of points matched	Dataset type
LGA01	2,298	2,287	99.52	Urban
LGA02	4,260	4,229	99.27	Urban
LGA03	16,328	16,043	98.25	Rural
LGA04	5,262	5,159	98.04	Urban
LGA05	12,749	12,494	98.00	Rural
LGA06	32,622	31,264	95.84	Rural
LGA07	39,301	38,073	96.88	Rural
LGA08	26,049	25,015	96.03	Urban
LGA09	41,725	41,388	99.19	Urban
LGA10	31,244	30,001	96.02	Urban
LGA11	88,918	87,331	98.22	Rural
LGA12	63,843	61,882	96.93	Urban
Totals	**364,599**	**355,166**	**97.41**	

Table 9.4 shows the number of points matched by point type, i.e. nodes, corners, other salient points, and all remaining points (including isolated points). Again, there is no objective way to prove that all these matches are correct but the high percentage of identical reverse shift vectors created from the matched points, detailed in Section 9.5, suggests that the majority are. Inspection of the resulting old-cadastre adjustment has supported this conclusion. Figure 9.2 shows the counts in a bar chart – the vertical scale minimum is 80% in this chart.

Table 9.4 Counts and percentages of points matched, by type

	Count of points of each point type				**Percentage of points matched for each point type**			
Dataset Name	**Nodes**	**Corners**	**Salient Points**	**Other Points**	**Nodes**	**Corners**	**Salient Points**	**Other Points**
LGA01	1153	134	166	834	99.52	99.91	97.10	100.00
LGA02	1909	283	718	1319	99.27	99.17	98.61	98.63
LGA03	3723	2074	1225	9021	98.25	98.10	97.74	98.39
LGA04	3660	351	616	532	98.04	98.76	94.61	96.86
LGA05	4519	2224	2114	3637	98.00	98.69	98.02	99.20
LGA06	8726	2613	7748	12177	95.84	97.09	92.92	96.63
LGA07	9486	2407	6791	19389	96.88	97.80	93.84	97.01
LGA08	12856	729	5949	5481	96.03	96.17	88.47	96.75
LGA09	14921	987	1459	24021	99.19	98.92	94.81	96.49
LGA10	20741	3017	1577	4666	96.02	96.69	94.93	95.63
LGA11	21924	6808	37120	21479	98.22	98.64	97.16	98.11
LGA12	36881	2107	7420	15474	96.93	98.16	92.05	99.32
Totals	**140499**	**23734**	**72903**	**118030**	**97.41**	**97.88**	**95.31**	**97.80**

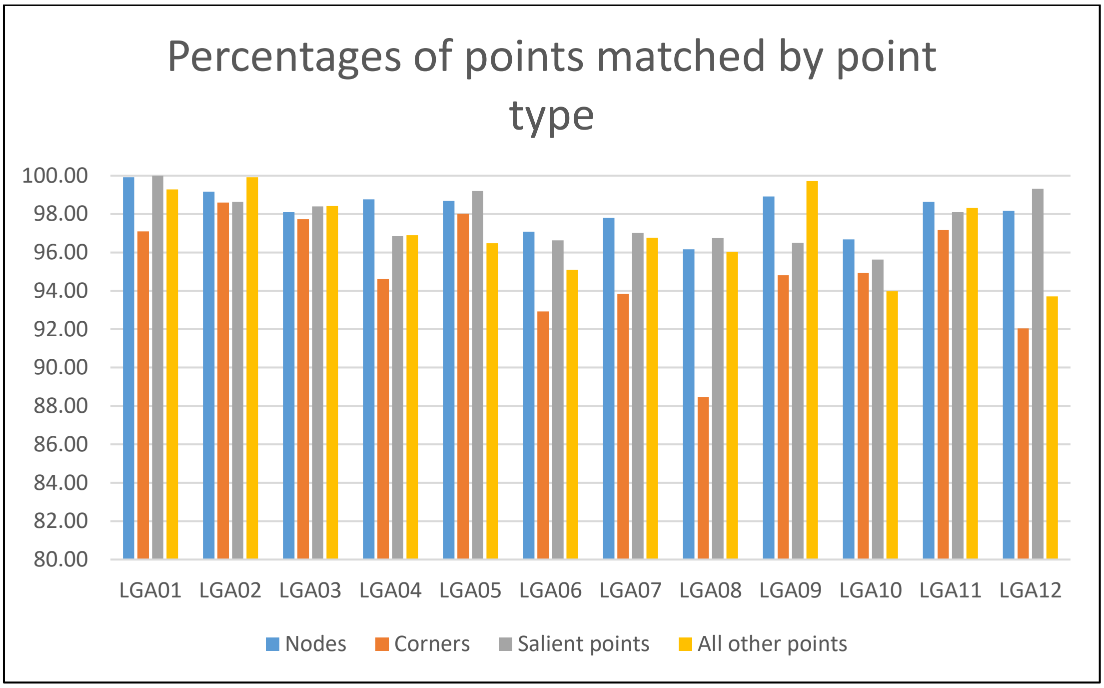

Figure 9.2 Percentage of points matched, by type

9.3.3 Shift vector results

To compute the results shown in this section, ten length-range groups were created using the equation

$$LI = \lceil (ML / 10) \rceil$$

where *LI* is the upper limit of the length of the first group and *ML* is the length of the longest shift vector. Length groups were created by rounding the length of the longest shift vector in the dataset to the next multiple of 10 and creating 10 equal sized groups. Lengths are in metres.

Table 9.5 shows the length distribution of the shift vectors created for each urban LGA. Column 1 (the shaded columns) for each LGA shows the maximum length in metres for the group to which the vectors were assigned; column 2 (the unshaded columns) shows the number of vectors that occurred in each length group.

Table 9.5 Shift vector length distribution in urban datasets

LGA01		LGA02		LGA04		LGA08		LGA09		LGA10		LGA12	
1	2	1	2	1	2	1	2	1	2	1	2	1	2
2	131	2	1484	1	3252	5	16611	2	31014	3	28469	2	60540
4	1377	4	1072	2	1611	10	7857	4	9943	6	1180	4	782
6	568	6	1144	3	158	15	296	6	122	9	100	6	126
8	166	8	259	4	73	20	1	8	39	12	9	8	140
10	14	10	165	5	24	25	1	10	40	15	16	10	0
12	1	12	32	6	11	30	1	12	9	18	12	12	86
14	1	14	16	7	7	35	2	14	28	21	17	14	0
16	0	16	6	8	1	40	1	16	10	24	7	16	0
18	0	18	1	9	0	45	1	18	60	27	0	18	2
20	0	20	1	10	1	50	2	20	0	30	0	20	0

Figure 9.3 shows the same data as in the above table with the y-axis showing the number of shift vectors in each length group as a percentage of the total number of shift vectors created.

Table 9.6 shows the length distribution results for rural datasets and illustrates the much greater apparent vertex movement that can occur in these areas. LGA05, for example, had a maximum shift vector length of almost 800 metres whereas the longest vector in any of the urban datasets was less than 50 metres; a vector longer than 200 metres can occur because PMA1 vectors are permitted by the algorithm to exceed the 200 metre limit employed elsewhere.

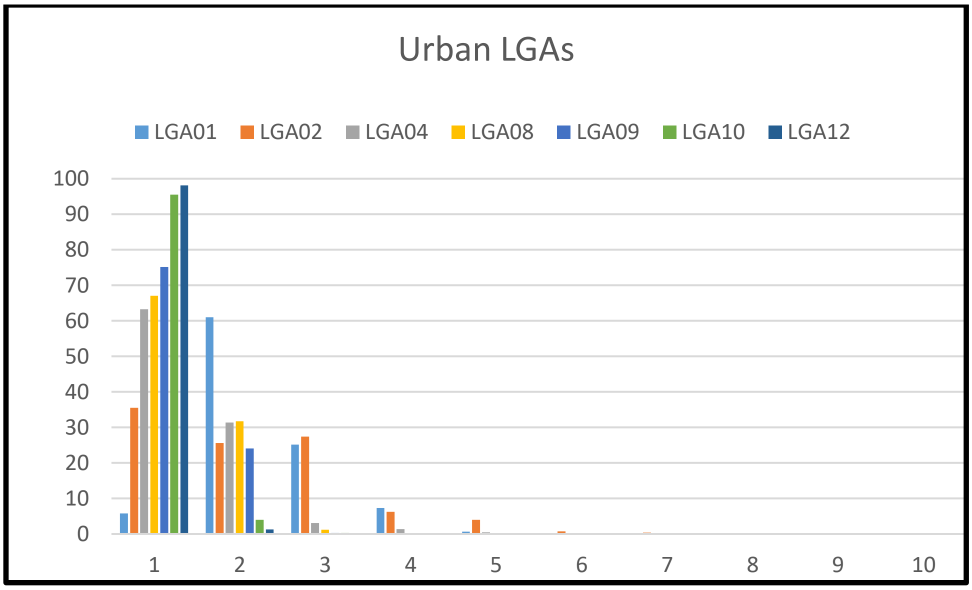

Figure 9.3 Shift vector length distribution in urban LGAs

Figure 9.4 shows the length distribution results for rural LGAs. Again, the y-axis shows the number of vectors in each group as a percentage of the total number of vectors created for the dataset.

Table 9.6 Shift vector length distribution in rural datasets

LGA03		LGA05		LGA06		LGA07		LGA11	
1	2	1	2	1	2	1	2	1	2
14	5653	80	11919	14	16221	12	20433	67	80858
28	7827	160	395	28	9467	24	10069	134	3908
42	2107	240	4	42	3252	36	3892	201	132
56	270	320	2	56	833	48	1357	268	49
70	33	400	3	70	316	60	457	335	42
84	7	480	0	84	140	72	248	402	7
98	2	560	0	98	89	84	83	469	5
112	0	640	3	112	26	96	328	536	7
126	1	720	0	126	15	108	257	603	10
140	1	800	3	140	3	120	63	670	18

Maps were inspected in areas where unexpectedly long shift vectors were located. In most of these cases, the longer vectors were created by the PMA1 match process and were therefore not eliminated by the automated error removal processes described in Section 8.2. The table above shows that the LGA11 dataset has a total of 270 shift vectors (highlighted in yellow) with a length greater than 200 metres. Inspection of these vectors showed that all but four resulted from the PMA1 algorithm. Of the remaining four, one was correct and the other three were not; these three did not have an identical reverse shift vector so could be found by the manual error checking process described in Section 8.3.6. The three shift vectors over 720 metres long in LGA05 were all found to be correct.

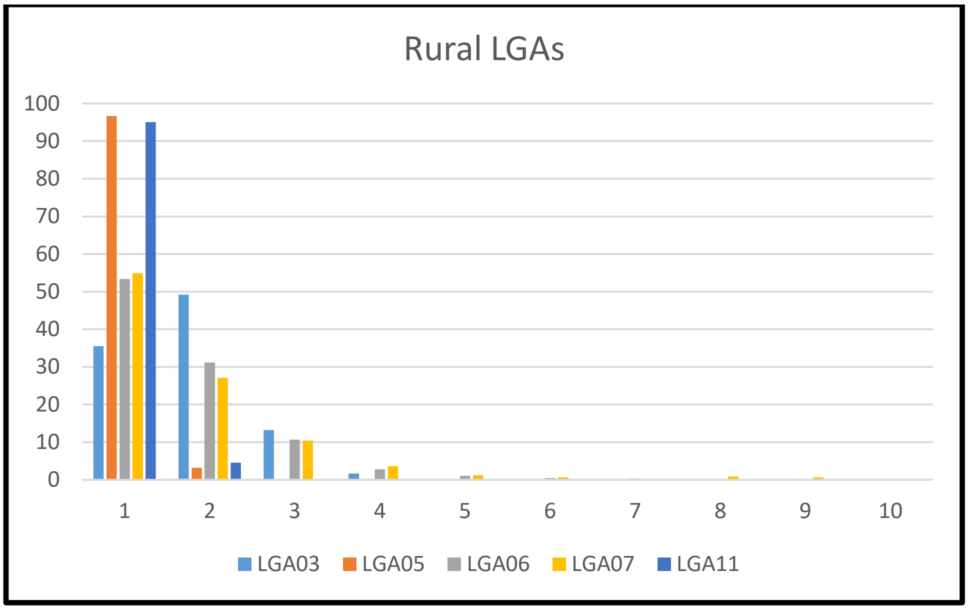

Figure 9.4 Shift vector length distribution in rural LGAs

9.3.4 Adjustment results

To check the accuracy of the ArcGIS rubber-sheeting process, the *RubbersheetFeatures* tool was applied to the old-cadastre points. For all points matched using PMA2, i.e. point to point, the distance between the adjusted old-cadastre point and the matched new-cadastre point was measured. This test was applied to one urban area dataset and one rural dataset. In no case was any significant distance between the adjusted old-cadastre point and the matching new-cadastre point found, indicating that the rubber-sheeting algorithm is operating as expected. The maximum distance of any point from its expected adjusted location was $3.74E^{-06}$. The vanishingly small values observed arise from the imprecision of real arithmetic in a computer.

A further test was conducted on the areas of matched parcels. The area of each matched parcel in the adjusted old cadastre was compared to the area of the matching parcel in the new cadastre. It is to be expected that, if the control point matching for that parcel's points was correct, the areas should match exactly. Parcels where the area discrepancy was greater than 5% were output to a separate layer for use in the error location process described in Section 8.3.4. Also, for each LGA dataset, the number of mismatched parcels were counted and the total areas of matched parcels in each cadastre computed. Table 9.7 shows the results of these computations. The rightmost column shows the difference between the total area of the old cadastre matched parcels and the new cadastre matched parcels as a percentage of the total new cadastre area of matched parcels. The maximum area discrepancy for any dataset is 0.07322% for LGA08 and the discrepancy for all the parcels in all twelve datasets is only 0.00333%.

Table 9.7 ShiftGen matched parcel area test results9.3.4

LGA ID	Number of parcels with greater than 5% area discrepancy	Total area of matched old cadastre parcels (ha)	Total area of matched new cadastre parcels (ha)	Area difference as a percentage of new cadastre area
LGA01	5	71.65	71.65	0.00805
LGA02	9	373.41	373.23	0.04929
LGA03	9	175,479.02	175,493.46	0.00823
LGA04	10	237.03	237.03	0.00003
LGA05	12	572,966.70	572,963.69	0.00053
LGA06	47	269,810.05	269,721.50	0.03283
LGA07	42	49,573.22	49,599.00	0.05196
LGA08	7	1,775.11	1,776.41	0.07322
LGA09	14	883.88	883.94	0.00672
LGA10	99	729.45	729.62	0.02290
LGA11	52	140,182.92	140,192.41	0.00677
LGA12	59	2,426.74	2,426.80	0.00232
Totals	**365**	**1,214,509.18**	**1214,468.73**	**0.00333**

It is interesting to note that results for the datasets with UIDs, LGA01, LGA02, LGA08 and LGA11, (highlighted), where the correctness of the match had been validated by the UIDs were not noticeably better than the datasets without, strongly suggesting that most of the parcel matches in the non-UID datasets were correct.

Only 365 parcels, from more than 70,000 matched parcels, exhibited a more then 5% area discrepancy after adjustment.

9.3.5 Statistics from all the completed test runs

More than 6,000 test runs were executed during the development and testing of the algorithms documented in this thesis. Several hundred of these tests were conducted in the parcel and boundary matching stages of the research before the point matching algorithms were implemented. Many more were terminated early by the author, sometimes because of a coding error or because the purpose of the run, to evaluate an algorithm change by examining the effect on a single parcel or point, had been achieved. The statistics in this section only report on the runs that resulted in a countable set of shift vectors – the completed test runs

During this research, most tests of the algorithms were concentrated on the areas of difficulty – the DA datasets. After identification and extraction of a difficult area, only when the results in that area were deemed by the author to be acceptable was a test run executed on the complete LGA dataset. Both Table 9.8 and Table 9.9 shows the results from the completed test runs. The tables show: the number of completed test runs; the number of unique points available for matching; the minimum and maximum number of shift vectors generated; and the minimum and maximum numbers generated as a percentage of the total number of points. The urban datasets are highlighted in red. Note that, although several maxima reach 100%, this value has not been reached in any of the final results shown in Section 9.2. The final versions of the algorithms have always been a compromise between matching too many incorrect control points and failing to match correct ones, and a greater number of matches may include many that are incorrect.

Table 9.8 shows the test run results for the DA datasets and Figure 9.5 shows a bar chart illustrating the minimum and maximum number of points matched as a percentage of the all the unique points from the dataset.

Table 9.8 Shift vector counts for completed test runs on DA datasets

LGA name	Number of test runs	Point count	Minimum shift vectors	Maximum shift vectors	Minimum percent matched	Maximum Percent matched
DA01	33	36	0	36	0.00	100.00
DA02	4	43	42	43	97.67	100.00
DA03	41	45	35	40	77.78	88.89
DA04	14	77	75	77	97.40	100.00
DA05	47	116	97	109	83.62	93.97
DA06	9	169	166	169	98.22	100.00
DA07	30	173	119	157	68.79	90.75
DA08	34	251	6	242	2.39	96.41
DA09	56	264	23	227	8.71	85.98
DA10	289	430	9	429	2.09	99.77
DA11	13	566	490	494	86.57	87.28
DA12	66	741	97	348	13.09	46.96
DA13	34	770	157	729	20.39	94.68
DA14	78	985	879	950	89.24	96.45
DA15	15	1103	874	1098	79.24	99.55
DA16	114	1146	59	1094	5.15	95.46
DA17	13	1162	798	1149	68.67	98.88
DA18	88	1145	195	1140	17.03	99.56
DA19	2	1243	1200	1231	96.54	99.03
DA20	97	1246	1051	1167	84.35	93.66
DA21	52	1421	1238	1408	87.12	99.09
DA22	20	2214	1551	2122	70.05	95.84
DA23	150	2341	0	1688	0.00	72.11
DA24	52	2377	519	2333	21.83	98.15
DA25	107	2703	2225	2497	82.32	92.38
DA26	104	3140	1223	2999	38.95	95.51
DA27	117	3562	0	3442	0.00	96.63
DA28	349	6110	3368	6012	55.12	98.40
DA29	3	6802	6644	6676	97.68	98.15
DA30	53	7219	4056	6987	56.19	96.79
DA31	14	7514	6523	6938	86.81	92.33
DA32	298	8111	944	7726	11.64	95.25
DA33	6	12157	3220	11596	26.49	95.39

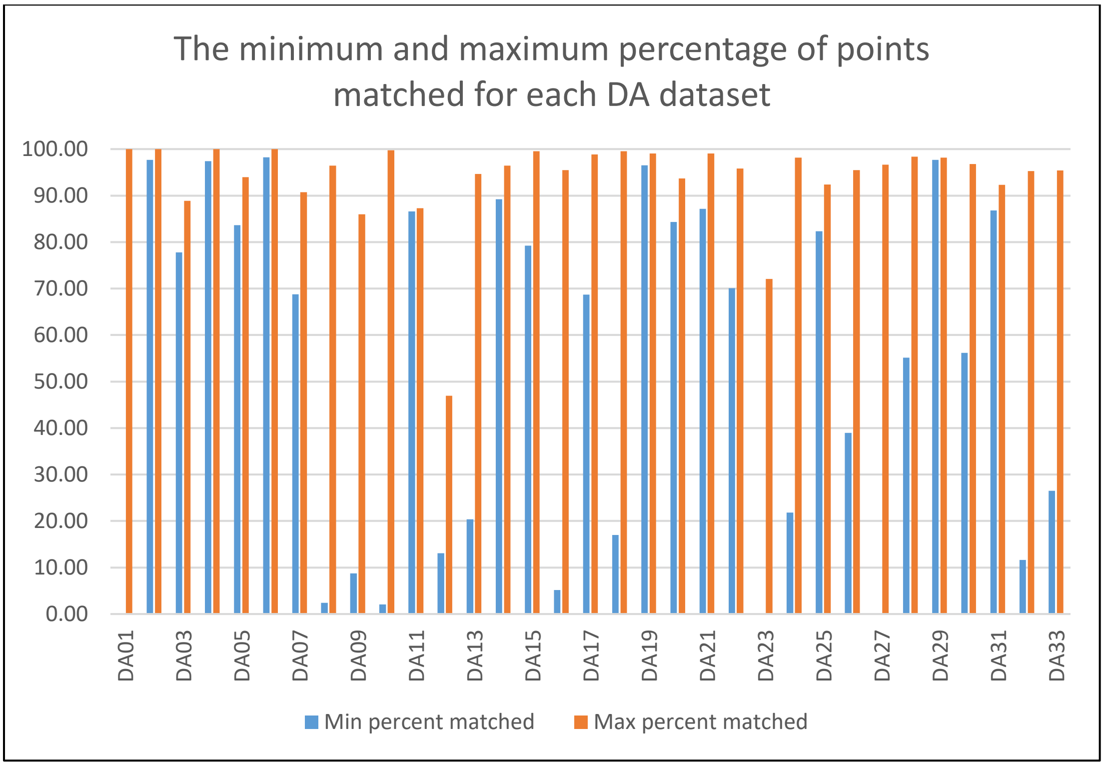

Figure 9.5 Minimum and maximum percentage of points matched for DA datasets

Table 9.9 shows the results for the full LGA datasets and Figure 9.6 shows the minimum and maximum percentage of points matched for each of 674 completed test runs executed on the full LGA datasets.

Table 9.9 Shift vector counts for completed test runs on full LGA datasets

LGA name	Number of test runs	Point count	Minimum shift vectors	Maximum shift vectors	Minimum percent matched	Maximum Percent matched
LGA01	280	2304	1733	2304	75.22	100.00
LGA02	38	4260	4007	4198	94.06	98.54
LGA03	16	16328	15268	15936	93.51	97.60
LGA04	40	5262	3613	5207	68.66	98.95
LGA05	20	12749	11968	12429	93.87	97.49
LGA06	24	32622	28450	30663	87.21	93.99
LGA07	130	39301	26768	38334	68.11	97.54
LGA08	18	26049	23467	25099	90.09	96.35
LGA09	11	41725	41185	41695	98.71	99.93
LGA10	15	31244	29147	30137	93.29	96.46
LGA11	65	88918	76049	87274	85.53	98.15
LGA12	17	63843	59431	61750	93.09	96.72

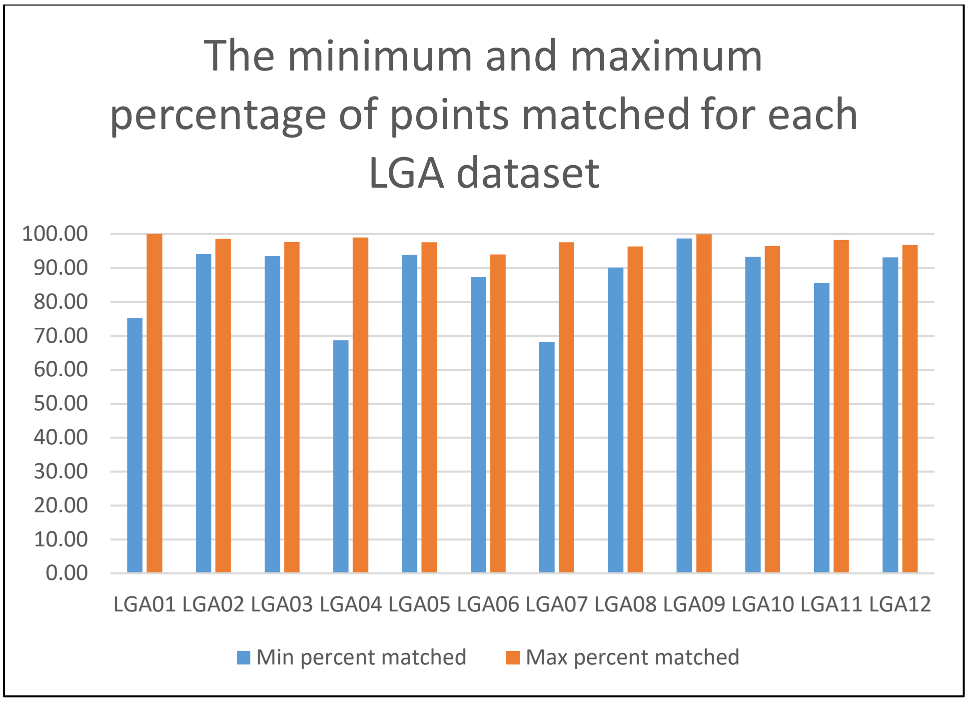

Figure 9.6 Minimum and maximum percentage of points matched LGA datasets

Figure 9.7 shows the test run results for each of 130 completed test runs executed on dataset LGA07 during this research. The graph shows the gradually improving success rate over time. LGA07 is a largely rural dataset with many especially difficult areas which have been extensively used to refine the matching algorithms. The increasing number of generated shift vectors occurred because of the refinement of the methods introduced to match long rural boundaries, i.e. PMA1 and PMA3. LGA07 has been chosen for illustration because it is the rural datasets with the largest number of completed test runs (see Table 9.9).

In contrast, the 280 test runs conducted on the urban dataset, LGA01, show no overall improvement in the number of points matched as the research proceeded. This is because points from urban parcels, which constituted about 98% of this dataset, are easily matched, especially in areas where there has been little apparent movement. LGA01 was small enough that it was not necessary to extract areas into smaller datasets for testing efficiency and its few areas of difficulty, for example, the turning circle illustrated in Figure 9.13, could easily be located and inspected in the GIS viewer. Because of its small size, this dataset has been used extensively in testing the complete solution after any major algorithm change. LGA01 has been chosen for illustration because it is the urban dataset with the largest number of completed test runs (Table 9.9).

In many of the full LGA datasets the number of points matched did not vary a great deal over time and, in any case, more matches do not always indicate improved results. The manual inspection of potential areas of error, described in Section 8.3 was of more value in determining whether an algorithm change had resulted in an improvement or whether the change had been disadvantageous.

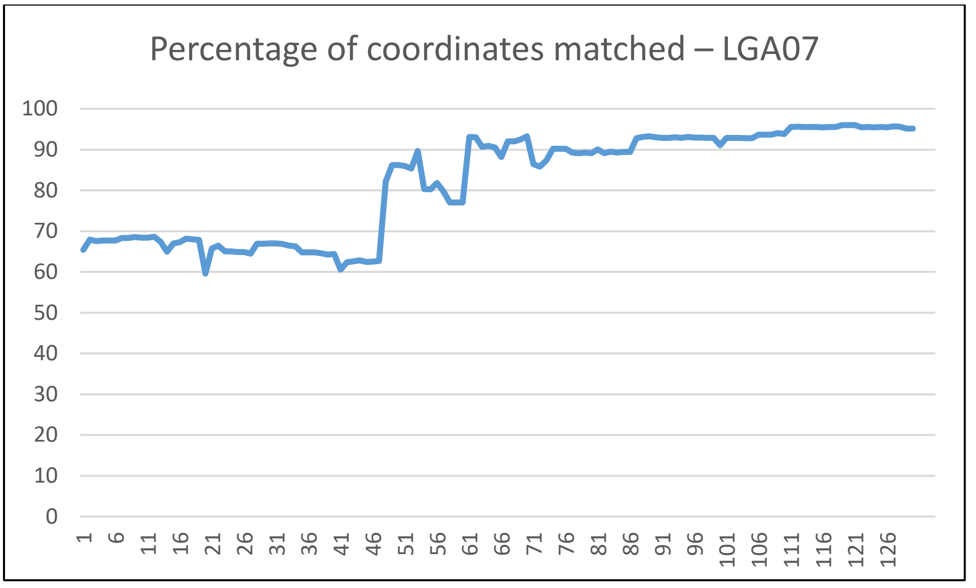

Figure 9.7 Test run results for LGA07

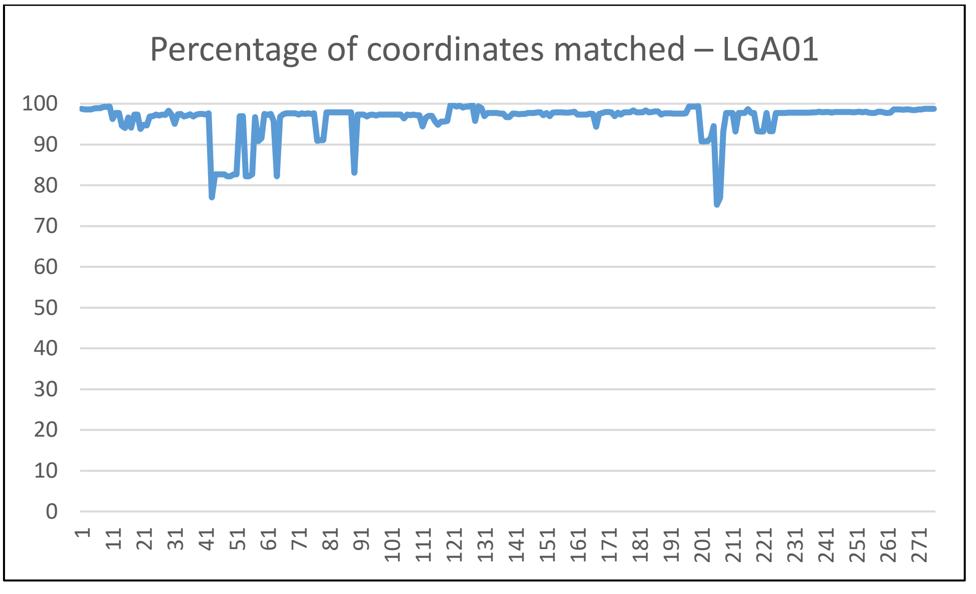

Figure 9.8 Test run results for LGA01

9.4 Example maps showing successful results

See Section 8.3 and 9.10 for examples of incorrect results.

Most of the illustrations in this section show the result of adjusting the old-cadastre dataset, using the shift vectors generated by the algorithms developed during this research, as input to the ArcGIS *RubbersheetFeatures* tool. It is assumed that where the old-cadastre adjustment is correct, inputting the same shift vectors to ArcGIS for the adjustment of the spatially dependent datasets would result in their correct adjustment also. This assumption has proven to be correct on the spatially dependent datasets processed during this research. Figure 1.2 showed an example, Figure 9.10 shows another.

When drawing the pairs of images in this section, the new-cadastre parcel boundaries in red were drawn first followed by the old-cadastre parcel boundaries in black; in each case, the map on the left shows the unadjusted old cadastre drawn over the new cadastre and the map on the right shows the adjusted old cadastre drawn over the new cadastre. Where there are any red boundaries to be seen in the right-hand image, the adjustment has been less than perfect.

In all cases, the LINEAR option for the ArcGIS *RubbersheetFeatures* tool was employed. According to the ArcGIS 10.4 help file for this tool, the LINEAR option is preferred when there are many rubber-sheet links (shift vectors) spread uniformly across the whole surface. The alternative option is NATURAL_NEIGHBOR; this option delivers more accurate results when there are fewer, more scattered points of correspondence between the datasets. Since the point matching algorithms developed for this research deliver large numbers of links across the entire area of the dataset, the LINEAR option has been chosen.

This section illustrates an adjustment result on a dependent dataset and typical results in the diverse types of areas likely to be present in any digital cadastre dataset.

9.4.1 Adjustment of a spatially dependent dataset

Figure 9.9 shows part of a town-planning scheme from the LGA07 dataset before adjustment, with the current cadastre (red) drawn on top. This image highlights the importance of carrying out the adjustment on this spatially dependent dataset before the publication of a town-planning scheme map with cadastral boundaries overlaid. If the scheme were to be published without adjusting the planning zones, the zone boundaries would appear to divide parcels rather than fall along parcel boundaries as can be seen on the yellow parcels near the centre of the map.

Figure 9.10 shows the same area after adjustment; the zone boundaries are now coincident with the parcel boundaries. The area has been selected to illustrate the desired outcome – a perfect adjustment – but, of course, wherever shift vectors are missing, for example, on truncated corners, or are incorrect for whatever reason, less than perfect results can be expected.

Figure 9.9 Part of a town-planning scheme before adjustment

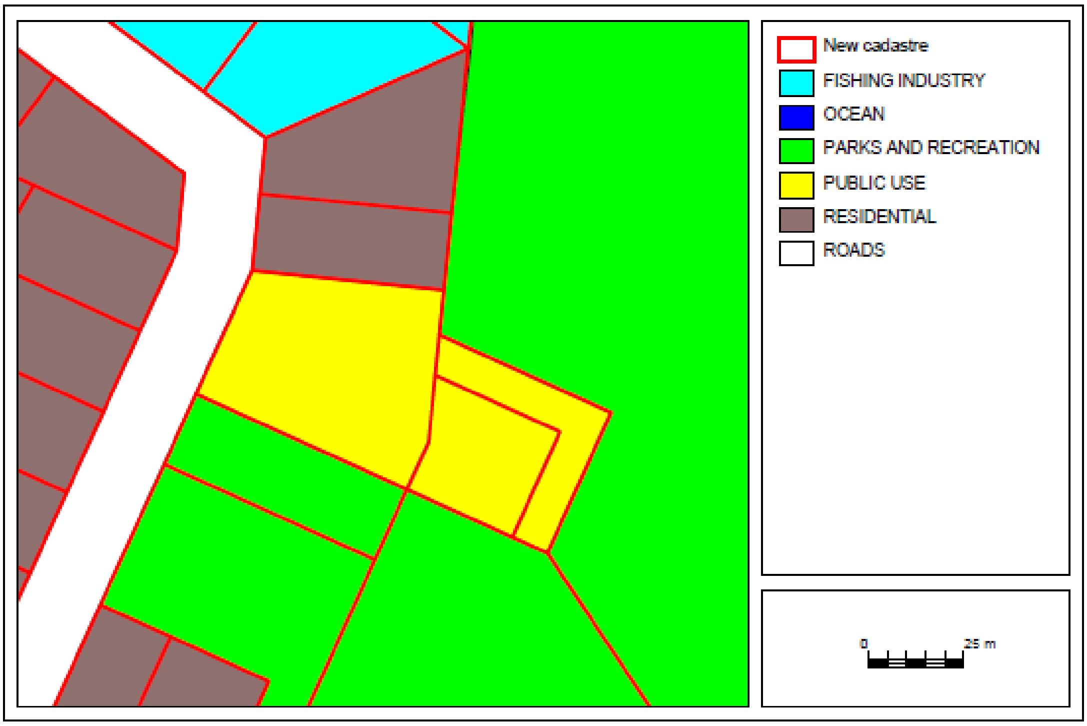

Figure 9.10 The same town-planning scheme after adjustment

9.4.2 Adjustment results in urban areas

Most of the purely urban datasets available for testing showed little apparent movement between cadastral versions and the adjustment results were excellent in those areas. Figure 9.11 shows eight small urban parcels (from LGA04) with an apparent shift of about one metre, before and after the shift vectors generated using the processes described in this thesis have been input to the ArcGIS *RubbersheetFeatures* tool. The new cadastre boundaries are invisible in the "after" image indicating perfect adjustment.

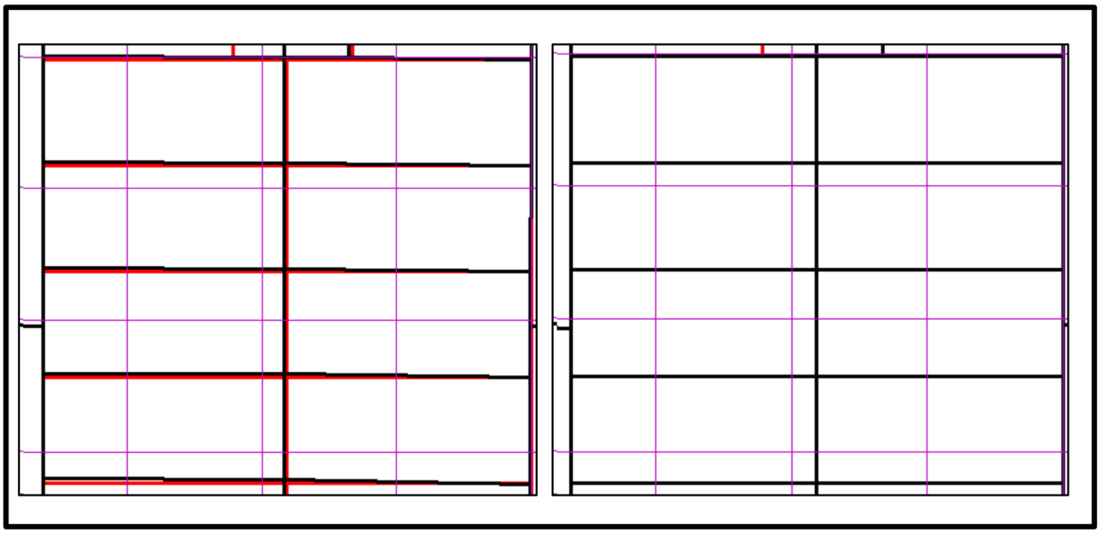

Figure 9.11 Eight small urban parcels before and after adjustment (25m graticule)

The urban areas present in the predominantly-rural datasets exhibited much greater apparent movements but here too the results were excellent in areas where there were no topological changes. Figure 9.12 shows an area from a town from the LGA11 dataset where the apparent movement of most parcel boundaries is more than 20 metres. Again, both layers were present when the images were drawn but the new-cadastre parcel boundaries are not visible underneath the perfectly adjusted old-cadastre boundaries in the right-hand image.

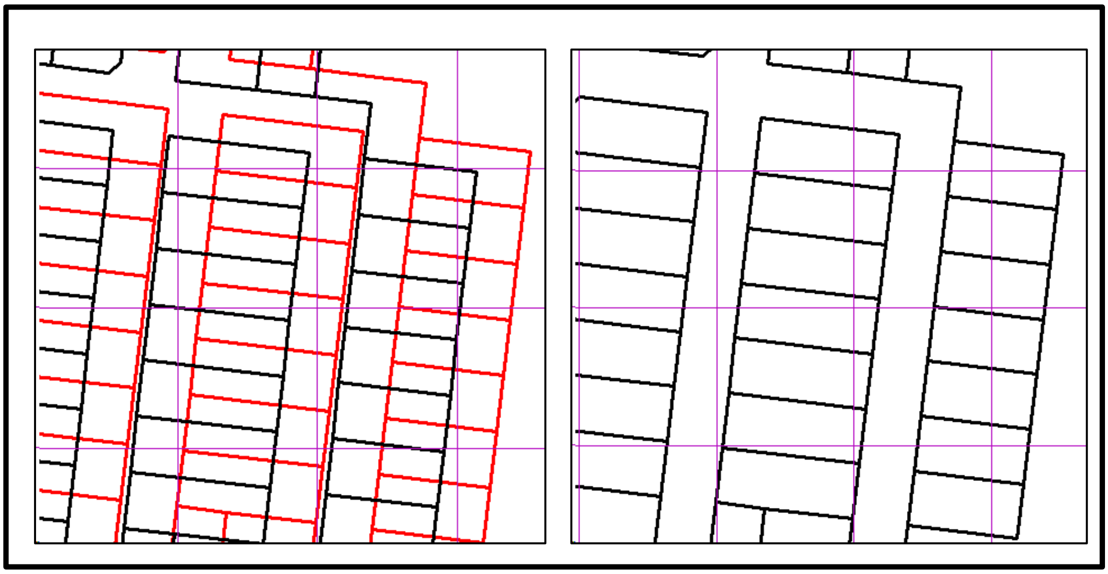

Figure 9.12 An area from a small rural town before and after adjustment (50m graticule)

Where results have been poorer in urban areas, this has typically been found, by inspecting maps of those areas, to be due to topological changes, or unidentified subdivisions and amalgamations, rather than by incorrect point matching by the PMA algorithms.

9.4.3 A turning circle adjustment

Figure 9.13 shows perfect adjustment on a turning circle (from LGA01) achieved using the shift vectors created by the matching algorithm PMA1 described Section 7.6. The old and new turning circle boundaries were matched to each other thus allowing PMA1 to be used.

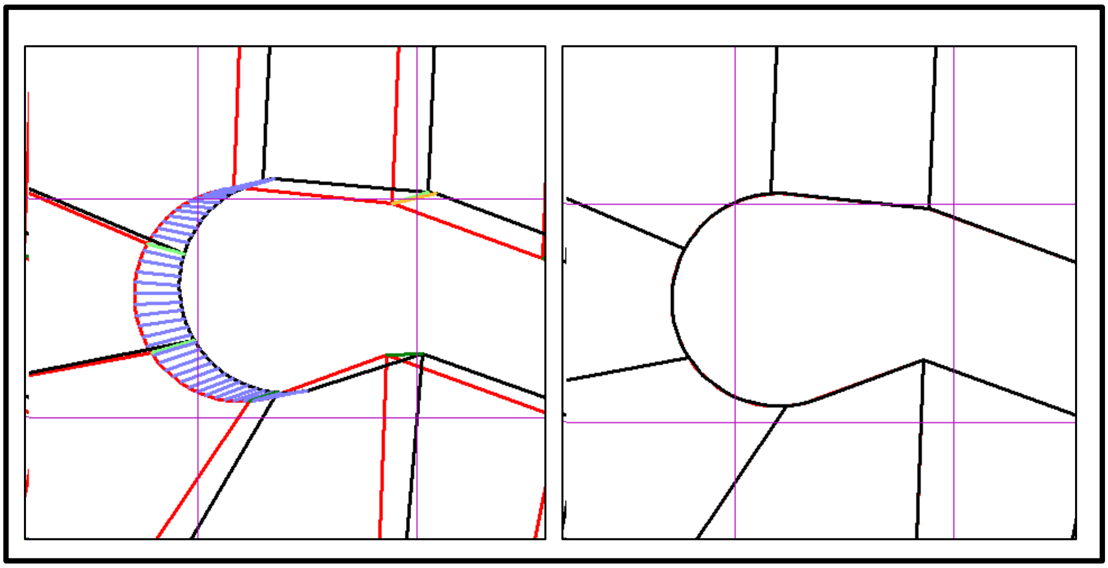

Figure 9.13 A perfect adjustment on a turning circle (25m graticule)

9.4.4 Adjustment on non-overlapping parcels

Figure 9.14 shows an area of non-overlapping parcels from the LGA11 dataset before and after adjustment. Once again, the adjustment appears to be perfect.

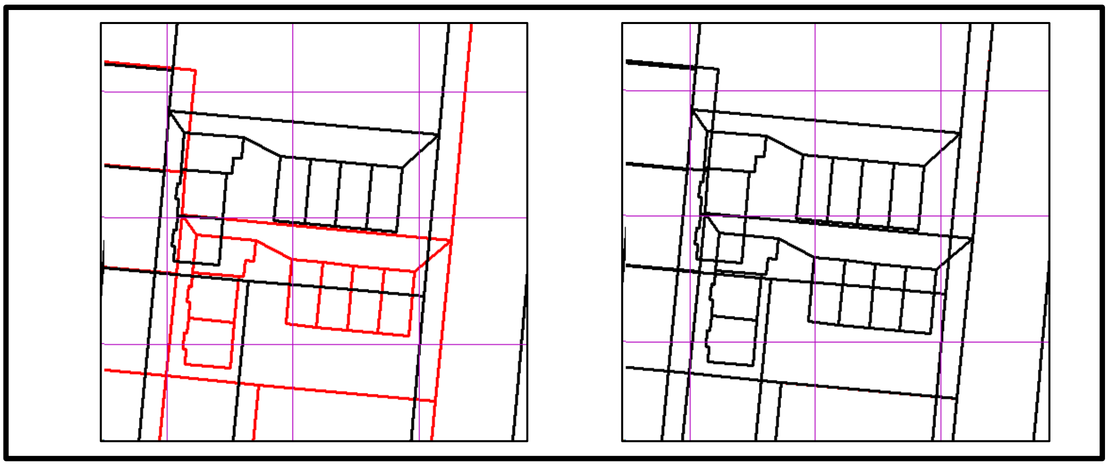

Figure 9.14 Non-overlapping parcels before and after adjustment (25m graticule)

It is unlikely that this correct result would have been achieved without the use of the block centroid shift vector described in Chapter 4 which indicated, in this case, that the matching parcel was expected to be found at about 20 metres distance from the original in the direction -80°. Because the parcels are correctly matched, the PMA2 algorithm generates an accurate target-point location and only considers points originating from the matched parcel.

9.4.5 Adjustment results in mixed urban and rural areas

Adjustment in rural areas has not always been as exact as that achieved in urban areas. The apparent movement of points in the rural datasets has usually been much greater than in the urban datasets and many more topological changes were also found in these areas, particularly in dataset LGA07. In addition, there is frequently no one-to-one correspondence between old and new cadastral points in rural areas. In general, however, results satisfactory to the author have been obtained over large parts of these datasets and, where the results have been poorer, this has typically occurred on riparian boundaries exhibiting large movements and in areas showing major topological changes such as those shown in several of the illustrations in Section 8.3.

Figure 9.15 shows an area from a small rural town in LGA07 before adjustment. It is clear from this map that most parcel boundaries have apparently moved between two versions of the cadastre issued eight years apart – by more than 50 metres in some cases.

Figure 9.16 shows the same area after the generated shift vectors have been input to the ArcGIS *RubbersheetFeatures* tool. Where the red new-cadastre boundaries are visible, the adjustment has not been perfect. Midway down the western edge of this map is an adjustment failure located on a truncated corner (the area is enlarged in the inset); problems arising from these are discussed further in Section 11.3.10. There are also some areas in the east where the visible red boundaries indicate a less than perfect adjustment.

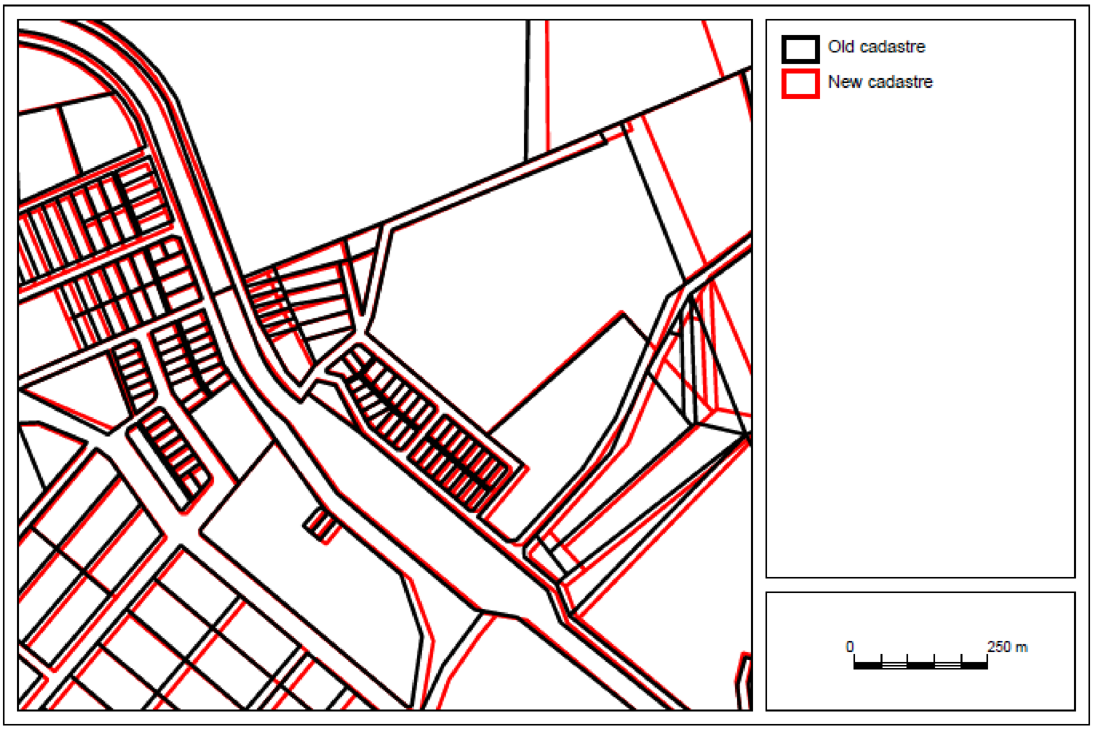

Figure 9.15 An area before adjustment

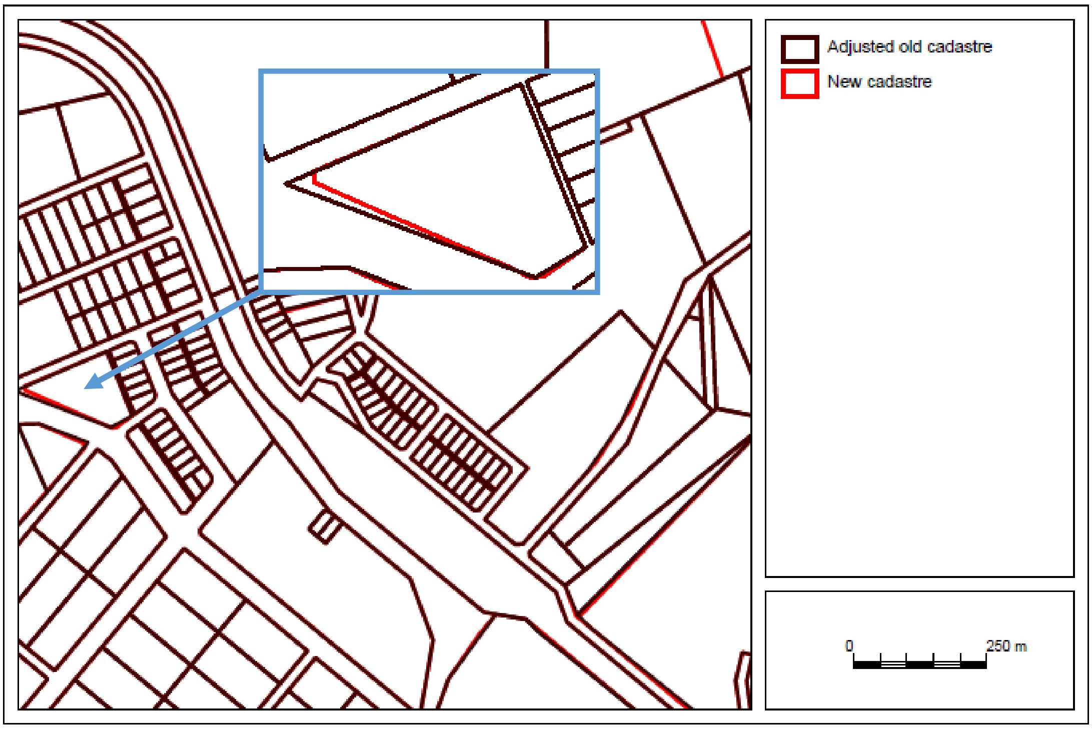

Figure 9.16 The same area as shown in Figure 9.15 after adjustment

The complete spatial adjustment process was also carried out using ArcGIS tools only, i.e. the *GenerateRubbersheetLinks* tool and the *RubbersheetFeatures* tool. A search tolerance of 25 metres was specified for the link generation tool. The results are shown in Figure 9.17.

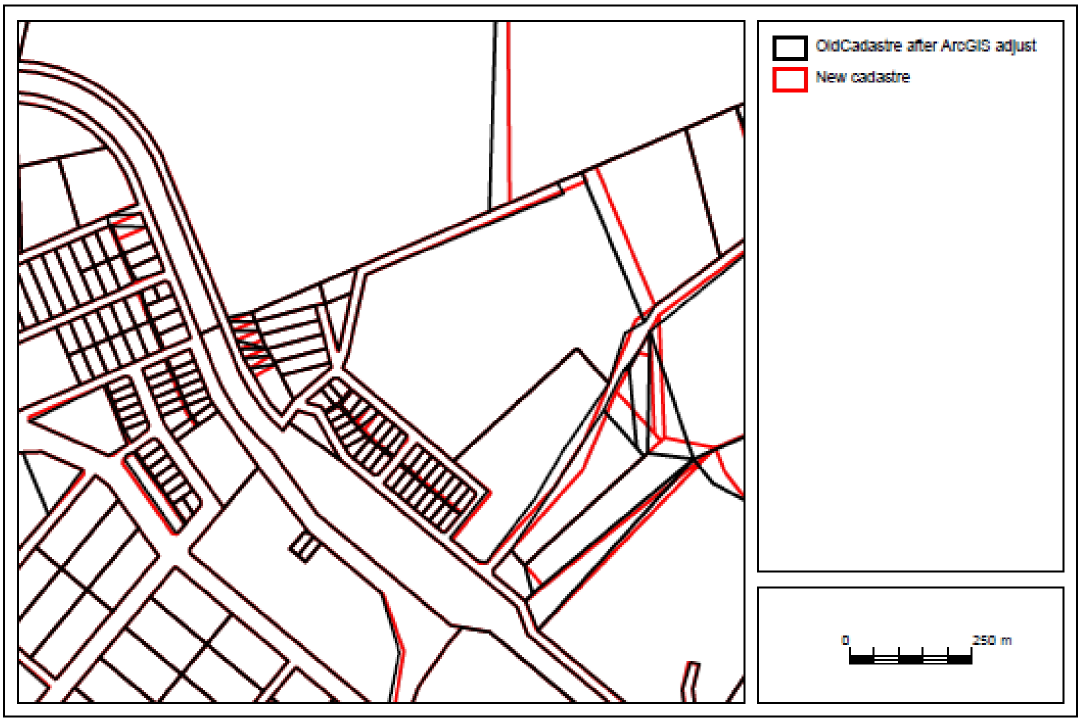

Figure 9.17 The same area after adjustment using ArcGIS tools only

As can be seen from the poor adjustment in the east of the area, the search tolerance here was too small. The results in that area were then slightly improved by increasing the tolerance to 60 metres (see Figure 9.18) but now the area towards the northwest has an even poorer adjustment because several over-long rubbersheet links have been generated; the inset shows an enlargement of that area.

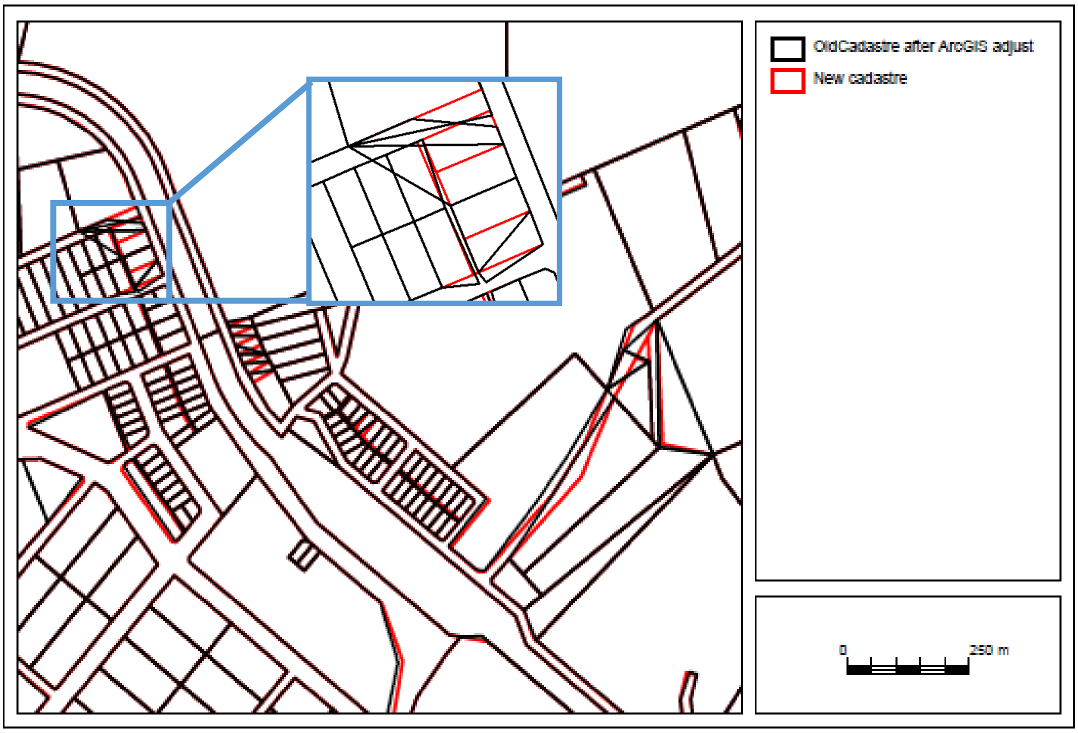

Figure 9.18 Adjustment using ArcGIS tool with increased search distance

The purpose of including these last two illustrations is to demonstrate the fact that, in a mixed urban and rural dataset such as this one, it is unlikely that a single maximum search-distance value, as required by commercial solutions, can deliver good results across the entire area thus illustrating the problem mentioned in Section 4.2 and supporting the conclusions arrived at during the Genetic Algorithm stage of the research (see also Appendix D).

9.4.6 Adjustment results in rural areas

Figure 9.19 illustrates satisfactory results achieved in a rural area of LGA07 where there have been no topological changes. The apparent movement of boundaries in this area is more than 20 metres. The maximum error after adjustment in the area shown in the insets is approximately four metres.

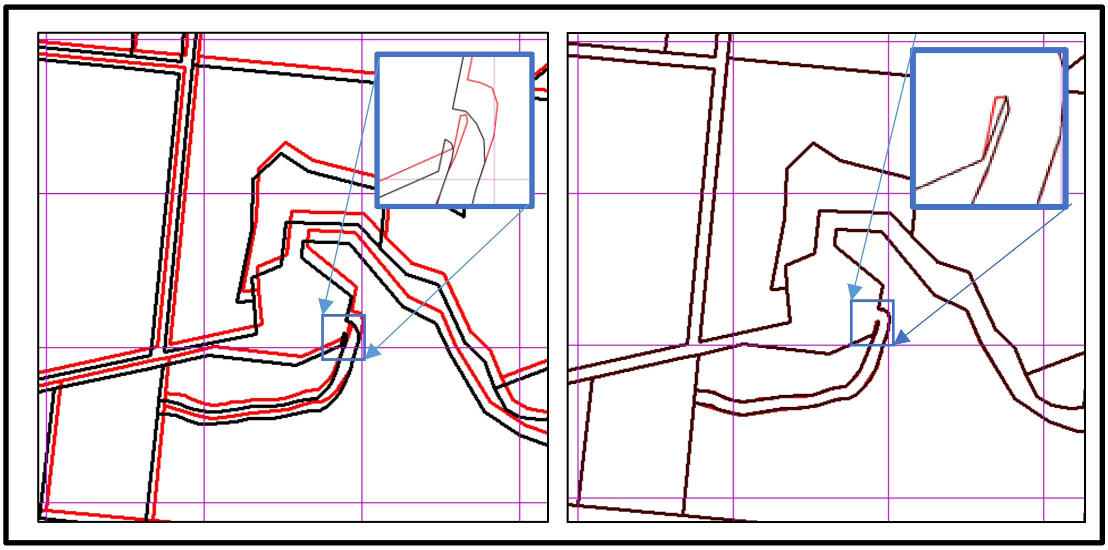

Figure 9.19 A rural area before and after adjustment (250m graticule)

Figure 9.20 shows a rural area from LGA11 with a creek before and after adjustment. In this case, the old and new riparian boundaries had been matched to each other thus allowing PMA1 to be used to generate the shift vectors. As before, the upgraded cadastre is drawn in red and the old cadastre before and after adjustment is drawn in black. The movement of the creek banks in this area is greater than 30 metres in some places. After adjustment, the maximum distance between the adjusted and new cadastre creek banks is less than two metres.

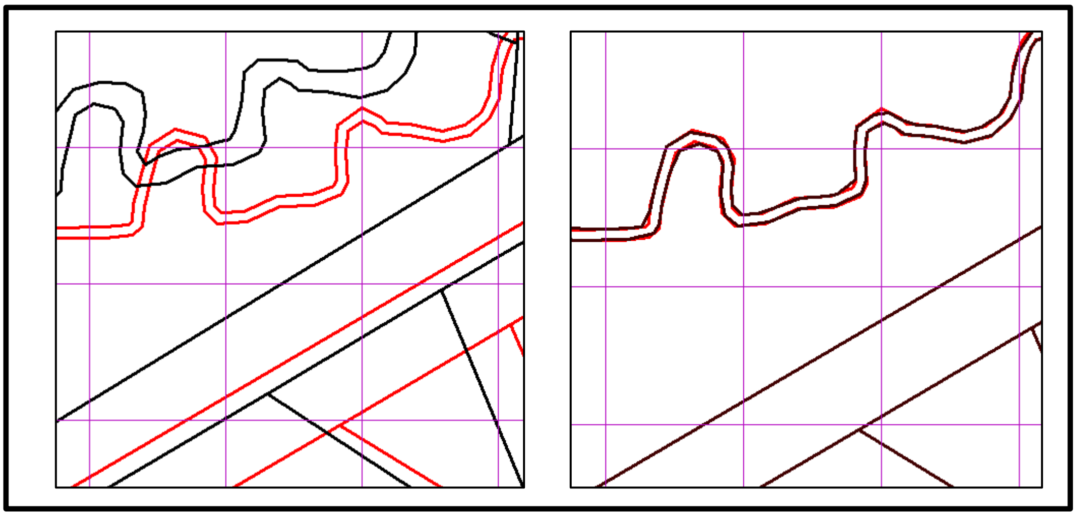

Figure 9.20 A riparian boundary before and after adjustment (50m graticule)

This section has provided several maps showing successful outcomes from the ShiftGen program. However, several control point matching problems have been identified during this research which have not been successfully resolved. They are documented in the section on future research in Chapter 10.

9.5 Comparison with ArcGIS

To measure the results of this research objectively it was necessary to compare the results with those from another solution. ArcGIS was selected for this purpose but it must be emphasised that ArcGIS is a generic toolkit that is not specifically tailored to cadastral datasets.

To obtain the results shown in this section, an ArcGIS Python script was developed and executed on all the complete LGA datasets and the difficult-area subsets. The results from the script were links (shift vectors) in both directions – old-to-new cadastre, and new-to-old cadastre. The same process, i.e. creating the shift vectors in both directions, was repeated for all the datasets using ShiftGen. All shift vectors and links that were found to connect identical pairs of control points were flagged and counted.

9.5.1 Identical reverse shifts where the match is wrong in both directions

It was initially assumed that identical reverse shift vectors would indicate correct control point matches as suggested by Siriba et al. (2013). However, inspection of results from ArcGIS in particularly difficult areas such as an area from LGA01 shown in Figure 9.21 have shown that this is not always the case. The eight almost vertical links in this area each have an identical reverse link, but all are incorrect – the remaining links are correct. The inset shows the correct vectors generated by the ShiftGen program.

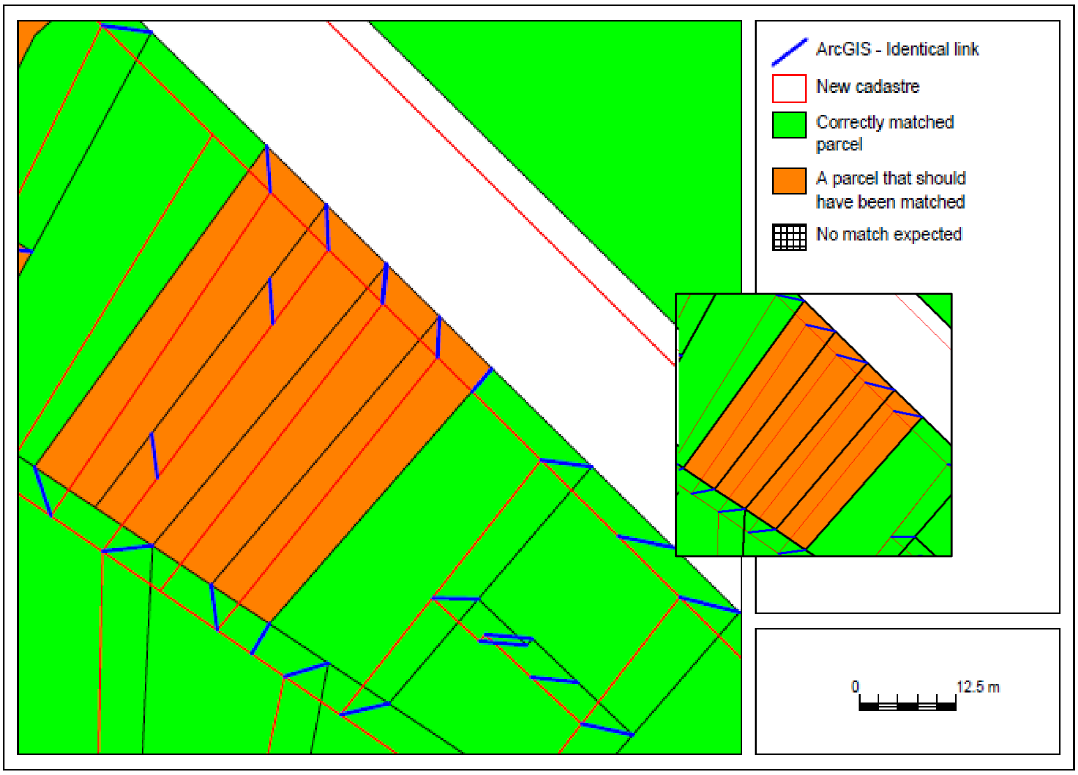

Figure 9.21 An area where eight identical reverse ArcGIS links are incorrect

The presence of identical reverse links on incorrect links in the above example demonstrates that the presence of an identical reverse link does not conclusively prove that a link is correct. It is interesting to note that this is a location where the ShiftGen parcel matching algorithm failed to correctly match four parcels (brown) in this area because their apparent direction of movement is inconsistent with other parcels in the same block. In this case, correct shift vectors have been generated by ShiftGen because the search distance and direction for matching points uses the average apparent movement of the surrounding matched parcels, not the apparent movement of the wrongly matched parcels, demonstrating the value of that portion of the solution (see Section 4.3.6.4).

Figure 9.22 shows a location from LGA11 where all the identical reverse links created by ArcGIS are incorrect. The identical ArcGIS links are shown in blue and the non-identical links are shown in yellow. The parcels in this area have apparently moved by more than half their street-frontage width. The map shows the ShiftGen results in magenta – these are all correct.

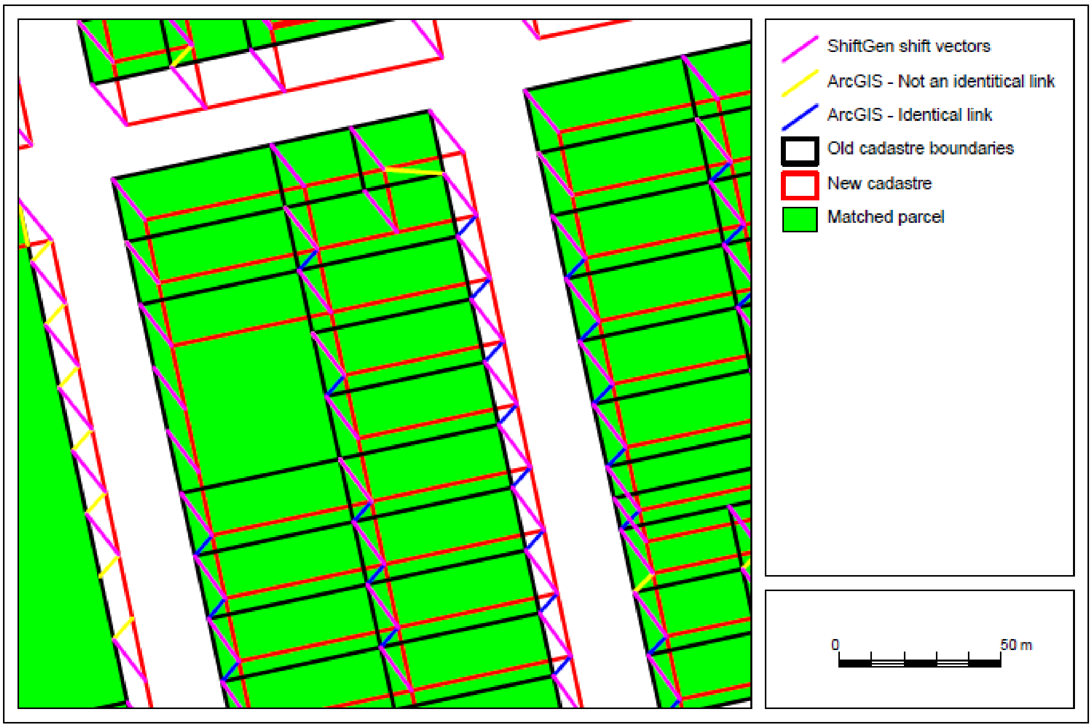

Figure 9.22 Identical links which are incorrect in both directions

Incorrect identical reverse shift vectors have not so far been discovered in the results generated by the ShiftGen solution although there is no objective way to ascertain that this has never happened in any of the test datasets because of the numbers of shift vectors that would need to be checked. Many incorrect identical reverse links have been observed in the ArcGIS results, particularly in urban areas with large apparent parcel movement.

9.5.2 Results from the difficult-area subsets.

The analyses in this section report the results from the difficult-area datasets. These datasets have been reported separately from the complete LGA datasets to highlight the fact that the ShiftGen solution matches a significantly larger number of control points than the ArcGIS solution in these areas. It is unsurprising that ArcGIS matches fewer points because the ShiftGen algorithms PMA1 and PMA3, can generate shift vectors where there may be no corresponding point in the new cadastre and many of the DA datasets contain complex riparian boundaries where the points would be processed by those algorithms. The ArcGIS algorithms are only attempting to match old-cadastre points to new-cadastre points and are, therefore, more likely to fail to find a match where there are fewer points on a corresponding new-cadastre boundary.

Table 9.10 shows the raw data from which the bar charts shown below were created. The blue shaded columns show the figures used for the blue bars and the pink columns show the figures used for the pink bars in Figure 9.23 and Figure 9.24. Rows with red text indicates datasets having predominantly urban sized parcels.

The **ShiftGen Point Count** column shows the number of points the **ShiftGen** algorithms are attempting to match. This count comprises all the unique location points but excludes those eliminated as being not important to match, i.e. non-salient points having segment angles of close to 180° (see Section 5.3).

The **ShiftGen Shift Count** column show the number of shift vectors created by the ShiftGen point matching algorithms.

The **% ShiftGen Points Matched** column records the number of shift vectors as a percentage of the number of unique-location points created by the processes described in Chapter 5.

The **% ShiftGen Identity** column records the number of identical reverse shifts as a percentage of all the shift vectors resulting from point-to-point matches rather than point-to-location matches; the latter cannot be expected to result in an identical reverse shift vector as there is no corresponding point in the upgraded cadastre to serve as the start point of a reverse direction vector.

The **ArcGIS Point Count** column shows the number of unique location points occurring in the line layers created by the ArcGIS *PolygonToLine* tool (Esri, 2017b) before the link generation step – *Generate Rubbersheet Links.* It has been assumed that this is the number of points that the *GenerateRubbersheetLinks* is attempting to match. The ArcGIS processes were run on the original copy of each cadastre, not the ShiftGen copy with subdivisions and amalgamations removed. For this reason, and because non-salient points have not been removed, in some cases ArcGIS is attempting to match more points than ShiftGen.

The **ArcGIS Link Count** shows the number of links (shift vectors) created by the *GenerateRubbersheetLinks* tool.

The **% ArcGIS Points Matched** column records the number of links as a percentage of the number of matches listed in the **ArcGIS Point Count** column.

The **% ArcGIS Identity Links** column records the number of identical reverse links as a percentage of the total number of links created by the *GenerateRubbersheetLinks* tool.

The **ArcGIS Search Distance** column records the search distance (SD) used for each dataset. It was computed from the mean parcel area (MPA) of the dataset using the equation:

$$\mathrm{SD} = \sqrt{\mathrm{MPA}} \,/\, 5$$

i.e. a search distance likely to be less than half the width of an average sized rectangular parcel. The ArcGIS *RubbersheetFeatures* tool requires a search distance parameter; in mixed urban-and-rural datasets the above equation gives rise to a figure that is a compromise between wrongly matching too many urban-area points and too few rural-area points. Note that no such parameter is required for the ShiftGen solution because the search distance in that solution uses a value computed from the properties of each individual point.

The last row in the table shows the means of the percentage of points matched and the percentage of identical reverse vectors for the ShiftGen and the ArcGIS results.

In almost all of these difficult-area datasets, the results from ShiftGen are significantly better than the ArcGIS results. The match rate is only equivalent in some of the more urban datasets, i.e. areas where the search distances were small because the mean parcel size in those datasets was small and the apparent parcel movement was also small. However, for ArcGIS, the lower number of identical reverse links in these difficult urban datasets suggests that many of the links are incorrect.

Table 9.10 ArcGIS comparison – difficult area datasets

Dataset Name	ShiftGen Point Count	ShiftGen Shift Count	% ShiftGen Points Matched	% ShiftGen Identity Shifts	ArcGIS Point Count	ArcGIS Link Count	% ArcGIS Points Matched	% ArcGIS Identity links	ArcGIS Search Distance
DA01	36	6	16.67	80.00	36	0	0.00		37.44
DA02	43	43	100.00	100.00	61	60	98.36	68.33	4.44
DA03	45	40	88.89	87.50	45	37	82.22	70.27	3.38
DA04	77	76	98.70	93.88	77	46	59.74	84.78	16.22
DA05	116	102	87.93	93.44	150	69	46.00	75.36	84.11
DA06	169	168	99.41	99.40	169	129	76.33	8.53	4.29
DA07	173	152	87.86	80.39	173	102	58.96	57.14	59.55
DA08	251	198	78.88	56.25	251	157	62.55	49.04	189.17
DA09	264	186	70.45	58.23	264	25	9.47	16.00	136.92
DA10	430	423	98.37	81.82	430	334	77.67	50.00	53.64
DA11	566	491	86.75	98.98	576	487	84.55	94.46	7.15
DA12	741	267	36.03	92.34	815	155	19.02	39.35	194.48
DA13	770	713	92.60	98.28	770	513	66.62	63.39	12.33
DA14	985	949	96.35	95.62	986	782	79.31	80.24	120.70
DA15	1103	915	82.96	81.30	1299	604	46.50	29.30	223.68
DA16	1146	1094	95.46	71.69	1157	388	33.54	36.08	166.42
DA17	1162	1149	98.88	99.19	1187	1172	98.74	77.73	7.15
DA18	1145	1128	98.52	98.49	1182	1145	96.87	89.15	38.64
DA19	1243	1231	99.03	97.99	1245	1195	95.98	82.82	42.84
DA20	1246	1166	93.58	76.93	1246	941	75.52	65.12	39.10
DA21	1421	1408	99.09	95.93	1426	1345	94.32	93.22	54.19
DA22	2214	2121	95.80	98.10	2219	2021	91.08	56.88	28.50
DA23	2341	1688	72.11	65.12	3782	2534	67.00	2.29	790.18
DA24	2377	2331	98.06	97.76	2426	2338	96.37	22.05	182.42
DA25	2703	2487	92.01	80.18	2710	2263	83.51	73.18	135.81
DA26	3140	2974	94.71	94.58	5042	4262	84.53	24.01	52.86
DA27	3562	3441	96.60	99.73	3566	3395	95.20	97.00	10.23
DA28	6088	5678	93.27	99.31	6244	4952	79.31	40.25	11.40
DA29	6802	6649	97.75	99.11	7053	6665	94.50	94.00	8.67
DA30	7219	6985	96.76	92.24	10172	7843	77.10	8.26	93.24
DA31	7514	6860	91.30	89.68	7587	5652	74.50	62.27	314.11
DA32	8114	7718	95.12	88.39	12453	9512	76.38	17.23	85.21
DA33	12157	11585	95.29	97.32	12268	11600	94.55	78.87	7.78
All DAs	77363	72422	**93.61**	**94.90**	89067	72723	**81.65**	**51.27**	

Figure 9.23 shows the percentage of points matched for ShiftGen and ArcGIS in a bar chart created from the figures in the Table 9.10. In most cases, significantly greater percentage of points have been matched by the ShiftGen algorithms than by the ArcGIS tools.

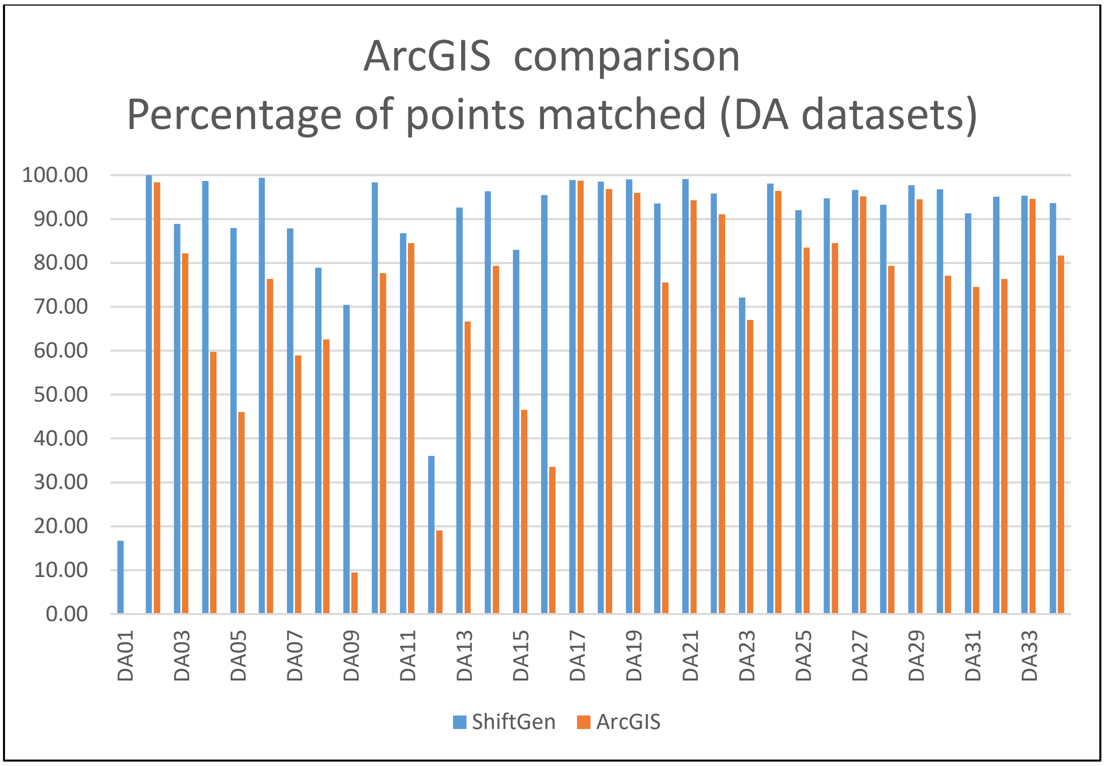

Figure 9.23 Percentages of points matched – DA datasets

Figure 9.24 shows the percentage of identical reverse vectors for ShiftGen and ArcGIS in a bar chart created from the figures in the Table 9.10. In every case, the ShiftGen algorithms have generated a higher proportion of identical reverse vectors.

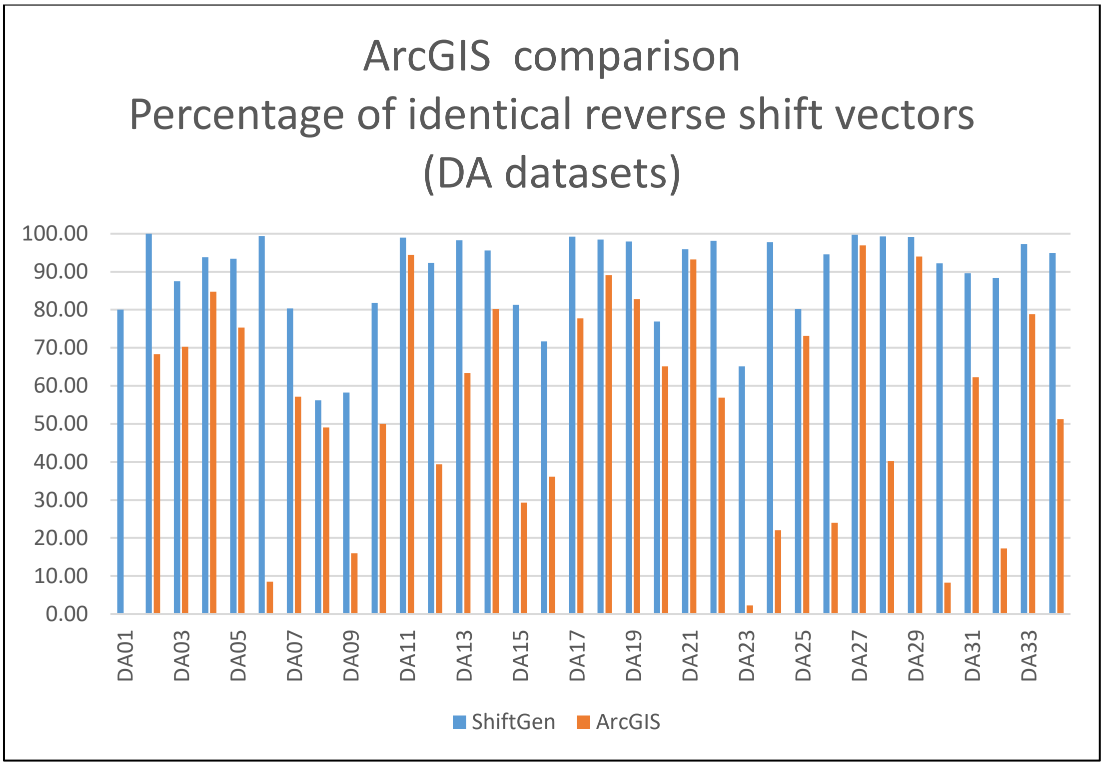

Figure 9.24 Percentages of identical reverse shift vectors – DA datasets

9.5.3 Results from the complete LGA datasets

Table 9.11 shows the results for the complete LGA datasets. The results have been computed and displayed as before (see Section 9.5.2). Although it may appear, from the raw data, that ArcGIS has created more shift vectors than ShiftGen, many of these vectors have been created from points that have been eliminated from the ShiftGen point matching process because they would have no significant effect on the final adjustment, for example, redundant vertices created where a straight parcel boundary is intersected by a map-tile boundary (see Section 5.3).

Figure 9.25 shows the bar chart of percentages of points matched. The bar chart was created from the results shown in Table 9.11. The percentages of matched points from these much larger datasets are much closer but, in absolute terms, the ShiftGen program has generated more than 12,000 additional control point matches with 97% identical reverse shift vectors, as opposed to the 65% identical reverse vectors created by ArcGIS.

Table 9.11 ArcGIS comparison – complete LGA datasets

Dataset Name	ShiftGen Point Count	ShiftGen Shift Count	% ShiftGen Points Matched	% ShiftGen Identity Shifts	ArcGIS Point Count	ArcGIS Link Count	% ArcGIS Points Matched	% ArcGIS Identity links	ArcGIS Search Distance
LGA01	2298	2270	98.78	98.24	2620	2303	87.90	63.70	7.31
LGA02	4260	4183	98.19	91.75	4280	4102	95.84	84.13	12.61
LGA03	16328	15934	97.59	97.90	20748	17516	84.42	37.36	179.08
LGA04	5262	5141	97.70	98.76	6998	6538	93.43	68.84	8.02
LGA05	12749	12401	97.27	98.37	15387	13463	87.50	58.65	288.54
LGA06	32622	30645	93.94	94.57	39021	32943	84.42	62.26	155.77
LGA07	39301	37472	95.35	95.08	55248	45018	81.48	41.70	75.39
LGA08	26049	24991	95.94	97.51	26301	24825	94.39	91.39	11.41
LGA09	41725	41356	99.12	96.79	65792	62138	94.45	60.79	6.90
LGA10	31244	29832	95.48	98.09	36603	34303	93.72	78.28	5.08
LGA11	88918	86083	96.81	97.13	90289	80594	89.26	66.77	70.62
LGA12	63843	61750	96.72	98.86	76203	72994	95.79	74.32	6.95
All LGAs	**364599**	**352058**	**96.56**	**97.20**	**439490**	**396737**	**90.27**	**65.16**	

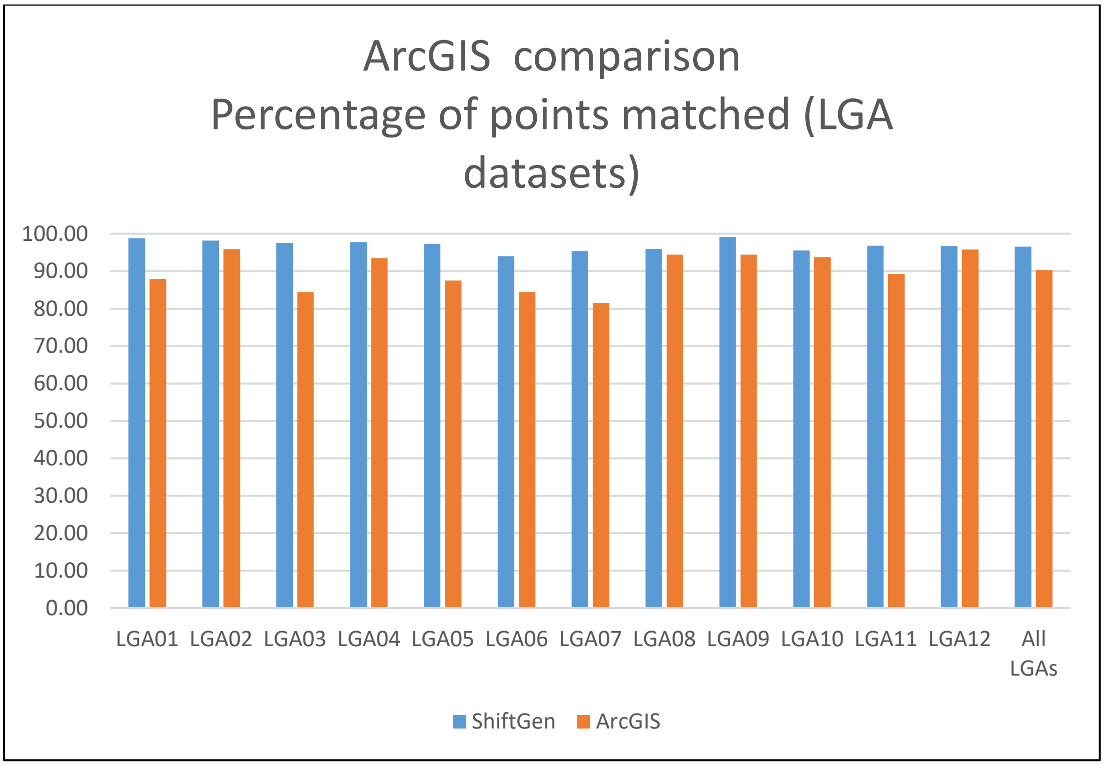

Figure 9.25 Percentages of points matched – LGA datasets

Of more importance than the number of shift vectors created is the correctness of those vectors, i.e. whether the points have been correctly matched. The count of identical shift vectors created by the two-way matching process can be an indication of correctness. Figure 9.26 show the percentages of identical reverse shift vectors as a bar chart using the figures from Table 9.11. This figure clearly shows that far more identical reverse shift vectors have been found among the ShiftGen created vectors than among the ArcGIS links.

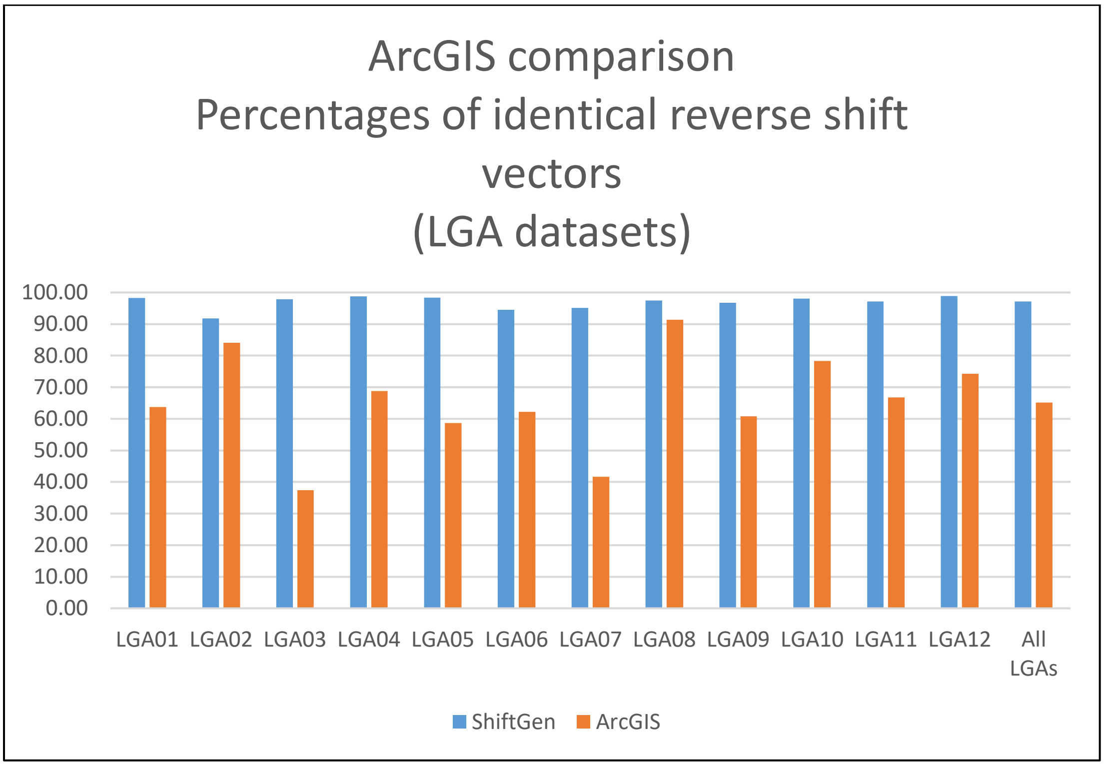

Figure 9.26 Percentages of identical reverse shift vectors – LGA datasets

The explanation for the inferior results from the ArcGIS tools is as follow. Although the datasets have previously been described as either Urban or Rural they all contain at least some parcels of each type, just in different proportions (see Figure 3.3 Figure 3.4). Because ArcGIS requires a single search distance parameter, the value generated automatically from the mean parcel area used for these tests has proven to be too large for some areas and too small for others, resulting in many incorrect matchings. Experimentation with the search distance values may have allowed the results to be improved. Also the ArcGIS documentation recommends the manual removal of intersecting links before executing the *RubbersheetFeatures* tool but, since the purpose of the comparisons in this section is to demonstrate that the algorithms developed for this research can produce superior results without the need for manual intervention, neither experimentation with search distances nor manual removal of intersecting links were performed for the ArcGIS tests.

The results summarised in this section demonstrate the value of a solution that requires no search distance parameters although, as has been noted before in Section 1.3, a manual check of the results arising from adjusting the old cadastre using the automatically generated shift vectors will always be advisable before proceeding to adjust the spatially dependent layers.

9.5.4 Comparison of geometry errors after adjustment

The ArcGIS *CheckGeometry* tool can be used to locate any areas where there are self-intersecting polygons and other geometrical errors. See (Esri, 2016c) for a complete list. Table 9.12 shows the number of geometry errors detected in the adjusted old cadastre resulting from the ShiftGen shift vectors and the number resulting from the adjusted old cadastre using the ArcGIS generated links. The counts are also express as a percentage of the number of parcels in the dataset.

Table 9.12 Geometry errors

LGA ID	Parcel count	Number of ShiftGen geometry errors	% ShiftGen geometry errors	Number of ArcGIS geometry errors	% ArcGIS geometry errors	Dataset type
LGA01	654	10	1.53	21	3.21	Urban
LGA02	1026	3	0.29	31	3.02	Urban
LGA03	2241	6	0.27	71	3.17	Rural
LGA04	2272	0	0.00	0	0.00	Urban
LGA05	2763	13	0.47	71	2.57	Rural
LGA06	5171	0	0.00	0	0.00	Rural
LGA07	5592	52	0.93	201	3.59	Rural
LGA08	7261	8	0.11	50	0.69	Urban
LGA09	8613	2	0.02	35	0.41	Urban
LGA10	11964	44	0.37	128	1.07	Urban
LGA11	12562	199	1.58	1179	9.39	Rural
LGA12	21345	0	0.00	0	0.00	Urban
Total	**81464**	**337**	**0.41**	**1787**	**2.19**	

Figure 9.27 shows the same data in a bar chart. Overall, ArcGIS has generated five times as many geometry errors from the ArcGIS links than from the ShiftGen shift vectors. In each case the *RubbersheetFeatures* tool has been used to effect the adjustment.

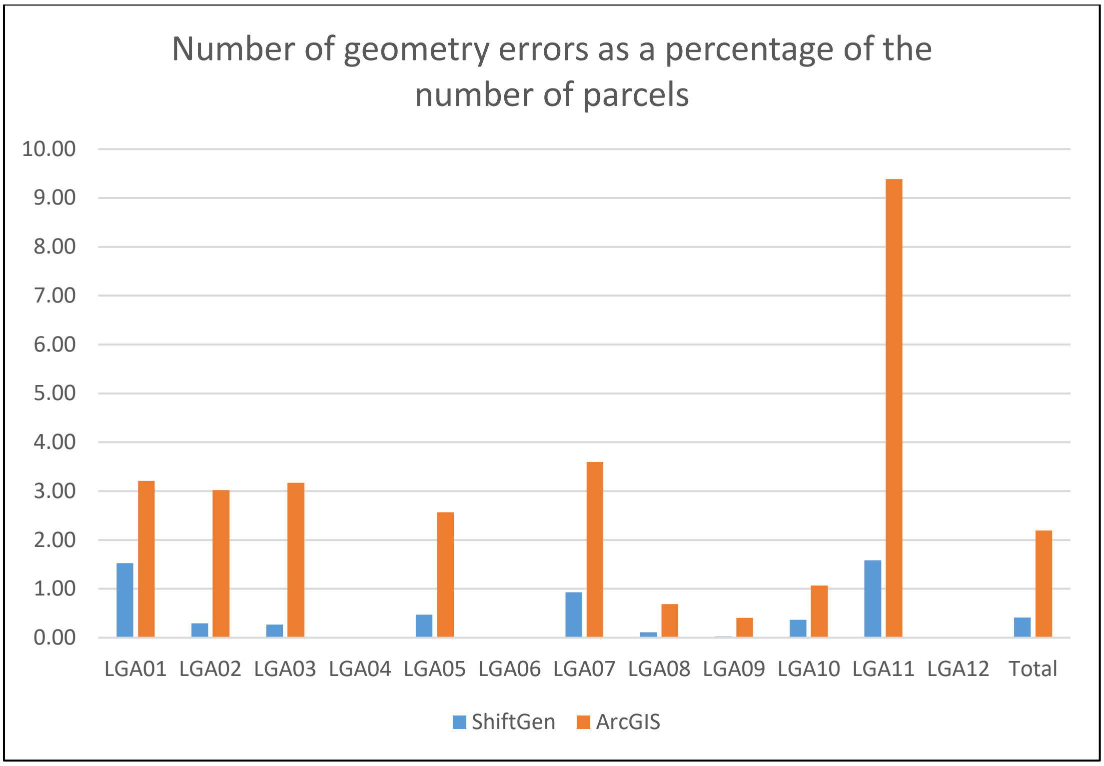

Figure 9.27 Geometry errors

9.5.5 Matched parcel area-test results

For the four LGA datasets where existing UIDs were available, the area of each adjusted parcel, from the layer adjusted using the ArcGIS tools only, was compared to the area of the corresponding new cadastre parcel. This is the same test as that conducted using the ShiftGen shift vectors, described in Section 9.3.4

Table 9.13 ArcGIS matched parcel area-test results

LGA ID	Number of parcels with greater than 5% area discrepancy	Total area of matched old cadastre parcels (ha)	Total area of matched new cadastre parcels (ha)	Area difference as a percentage of new cadastre area
LGA01	30	68.86	68.93	0.09
LGA02	59	406.65	319.85	27.14
LGA08	91	1,717.81	1,844.76	6.88
LGA11	2827	86,336.85	89,822.93	3.88
Totals	**3007**	**88,530.17**	**92,056.47**	**3.83**

The results shown here should be compared with those in Table 9.7 where only 365 parcels exhibited more than a 5% area discrepancy with a total area error of only .00333%. Note that the total areas in the two tables are not identical because the ShiftGen comparison test was executed on all the parcels matched by ShiftGen regardless of whether the parcels' original UIDs matched.

9.5.6 Maps comparing adjustment results from ShiftGen and ArcGIS

The figures in this section have been chosen to highlight the many advantages of a solution tailored to a cadastre over a generic toolkit such as ArcGIS. It must be emphasized that the ArcGIS tools deliver excellent results over urban areas where the apparent movement was small, but, in other areas, superior results were achieved by the ShiftGen program because of its use of embedded knowledge specific to cadastral datasets and because of the point matching algorithms such as PMA1 and PMA3 which allow otherwise unmatchable points to be used for generating shift vectors. In all the examples illustrated in the next sub-sections, an operator would need to manually correct the erroneous links if correct adjustment in those areas was important.

9.5.6.1 The benefits of utilising block matching information

Figure 9.28 shows shift vectors generated by the ShiftGen program (green) in a particularly difficult urban area where blocks appear to have moved in different directions by up to 40 metres. The map shows the unadjusted old cadastre in black and the new cadastre in red. The inset shows the resulting old-cadastre adjustment; the fact that the new cadastre drawn in red is not visible indicates a perfect adjustment.

Figure 9.29 shows the ArcGIS rubbersheet links generated for the same datasets in the same area. Four of the links cross the road void and link points falling in different blocks; this is prevented by the block match check implemented in the ShiftGen point matching algorithms (see Section 7.7.3). This dataset (LGA11) covered mostly rural areas with one large town; the ArcGIS vectors were created with a search distance of about 70 metres. The inset shows the results of the old-cadastre adjustment using the ArcGIS generated links.

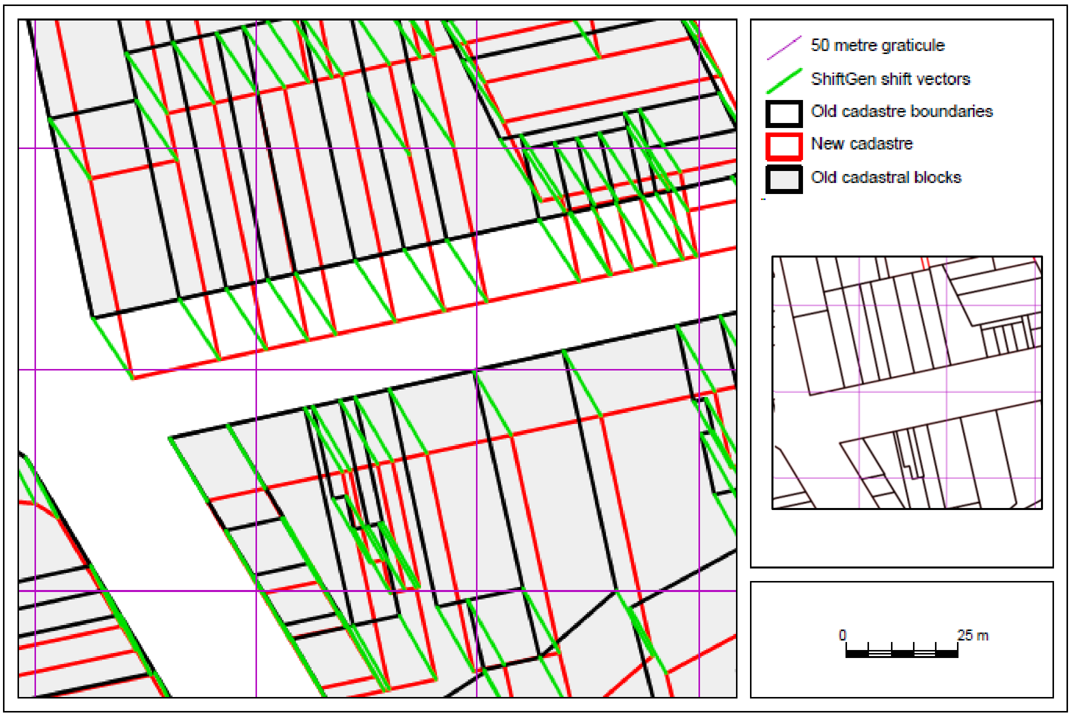

Figure 9.28 ShiftGen generated shift vectors and resulting adjustment

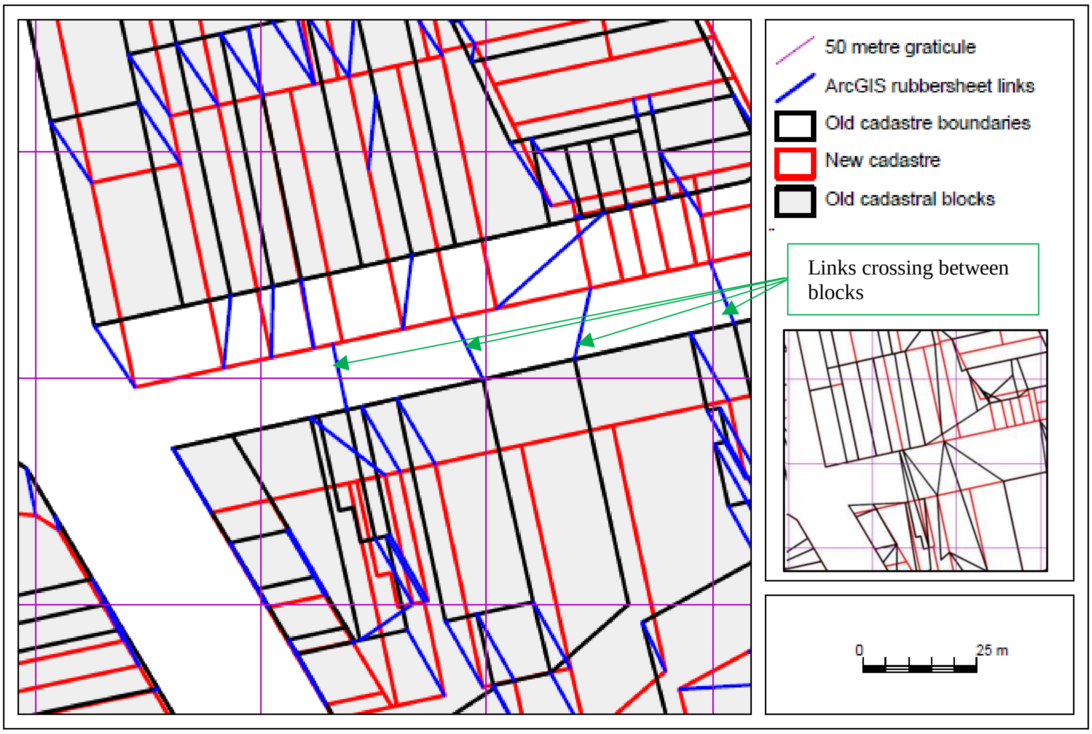

Figure 9.29 ArcGIS links and resulting adjustment

9.5.6.2 The benefits of utilising apparent parcel movement

Figure 9.30 shows maps covering an area from LGA11 where several small parcels (less than 100m^2) in the centre of the old-cadastre map layer have little or no overlap with the corresponding parcels in the new cadastre layer. ShiftGen was able to correctly match these parcels using the known apparent distance and direction of movement of the urban block. When the parcels are matched, their own apparent movement can also be computed making for tight constraints when seeking to match vertices from those parcels. The automatically generated shift vectors have all linked the correct pair of points by utilising the distance and direction of movement of their originating parcel centroids. The resulting old-cadastre adjustment is shown in the inset; it is correct in every area except in the centre south of the map where a node ambiguity has resulted in a poor result, i.e. there are two nodes in the old cadastre where there is only one in the new cadastre.

Figure 9.31 shows the same area and the ArcGIS-generated rubbersheet links. There are several that are incorrect so that the resulting adjustment shown in the inset is poor. The ArcGIS results in this area would have been improved a little by the manual removal of the intersecting shift vectors – a step that is not required in the ShiftGen solution as incorrect intersecting shifts are automatically removed by the process described in Section 8.2.3.

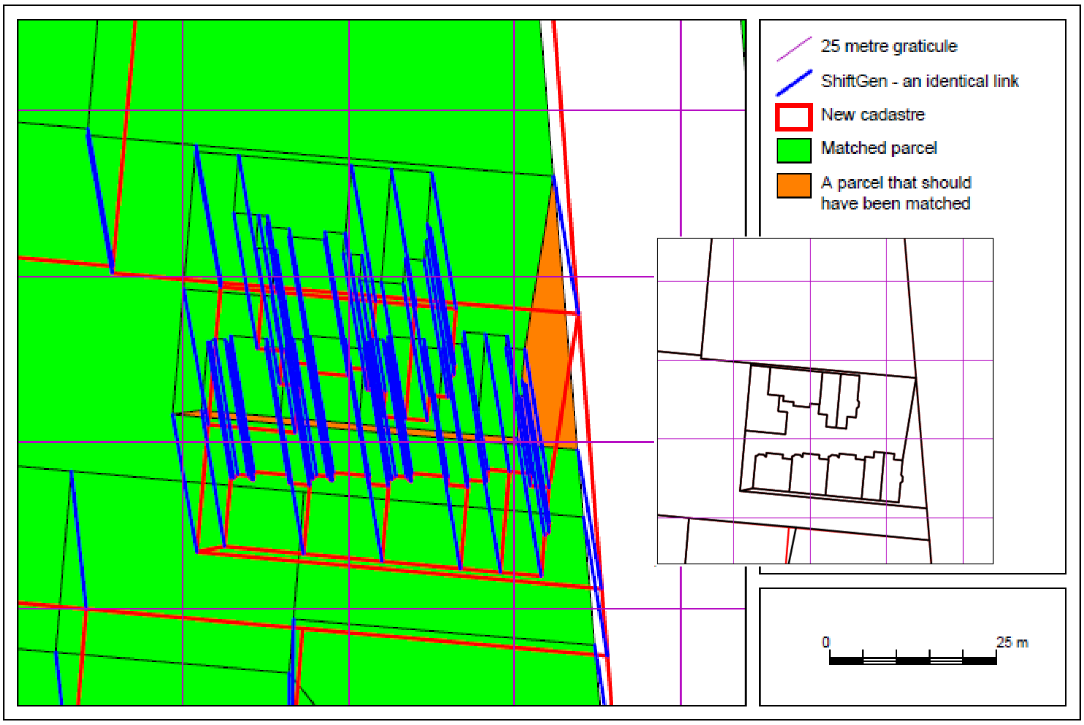

Figure 9.30 ShiftGen results on non-overlapping parcels

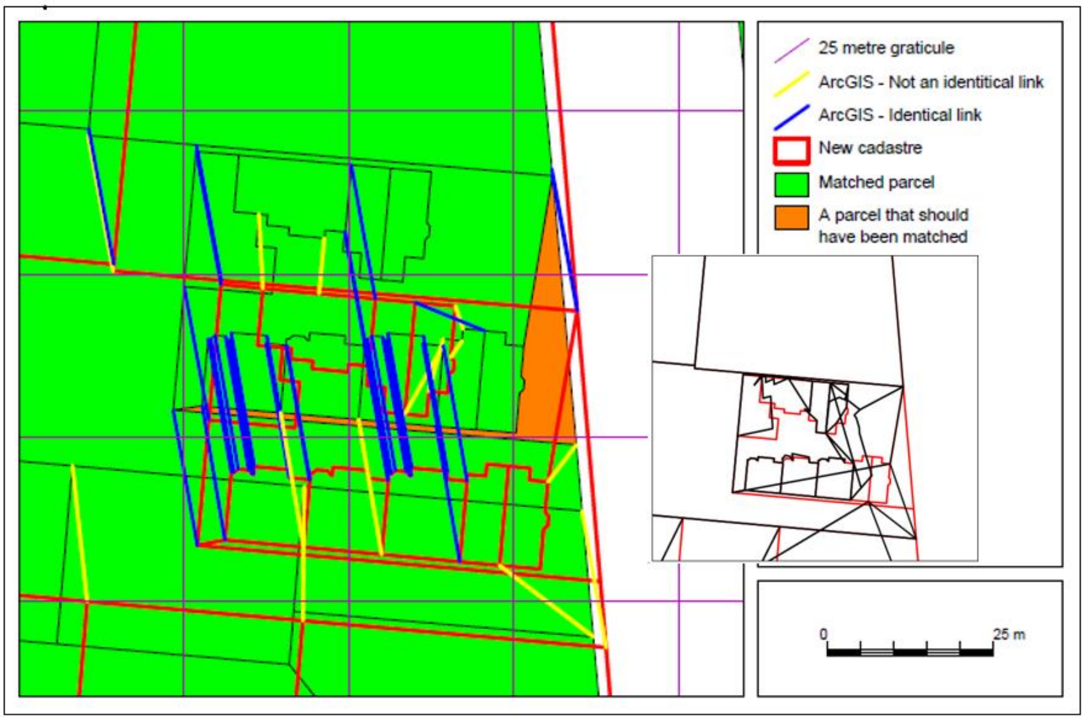

Figure 9.31 ArcGIS results on non-overlapping parcels

In these two maps, shift vectors and links have been symbolised using the result of discovering which vectors have an identical link in the reverse vector layer, i.e. the vectors resulting from matching the new cadastre points to the old cadastre points. In the case of the ShiftGen results, all the shift vectors in this area were found to have an identical reverse vector. This is not the case with the ArcGIS results, even for some of the correct links.

This illustration demonstrates the value of the information gained by first matching blocks and parcels when attempting to identify control points between two cadastres.

Even where matching parcels do overlap, results are superior to those generated by ArcGIS results because the matching process can make use of the apparent direction and distance of the parcel movement. Figure 9.32 shows an area where the parcels have apparently moved in a north-easterly direction by up to 19 metres. The inset shows the adjustment results from the ShiftGen program. Note that the two non-identical shift vectors are, in this case, both correct, Figure 9.33 shows the poor adjustment results obtained in the same area using the ArcGIS tools only. The ArcGIS links on almost all the rectangular parcels in the centre of this area have been wrongly matched to a point on an adjacent parcel in the new cadastre. Note, also that many of the identical ArcGIS links in this area are wrong in both directions.

Figure 9.32 ShiftGen results in an area with large apparent parcel movement

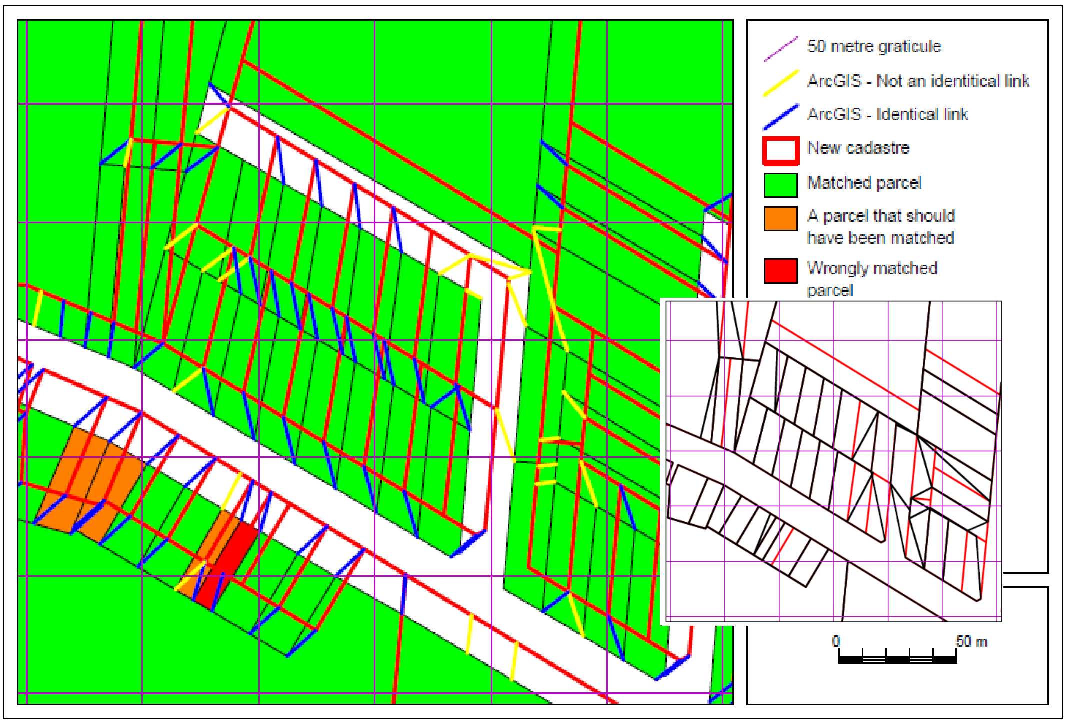

Figure 9.33 ArcGIS results in the same area as Figure 9.32

9.5.6.3 The benefit of identifying subdivisions and amalgamations

Figure 9.34 shows an area from LGA03, a rural dataset, where three old-cadastre parcels, shaded in grey, have been amalgamated in the interval between the release of the two cadastres. The green boundaries have been removed from the old cadastre before the parcel, boundary, and point matches proceeded. The insets show the adjusted original old cadastre, i.e. the dataset before removal of the green boundaries, drawn in black on top of the new cadastre in red; visible red boundaries indicate areas of poor adjustment. The upper inset shows the results of an adjustment using ShiftGen; in the area of the amalgamation, the adjustment results are perfect. The lower inset shows the results of an adjustment using ArcGIS tools only. ArcGIS generated incorrect links from the corners of the small square parcel outlined in green in the main map. The incorrect links have resulted in the incorrect adjustment of that parcel and the larger triangular parcel surrounding it.

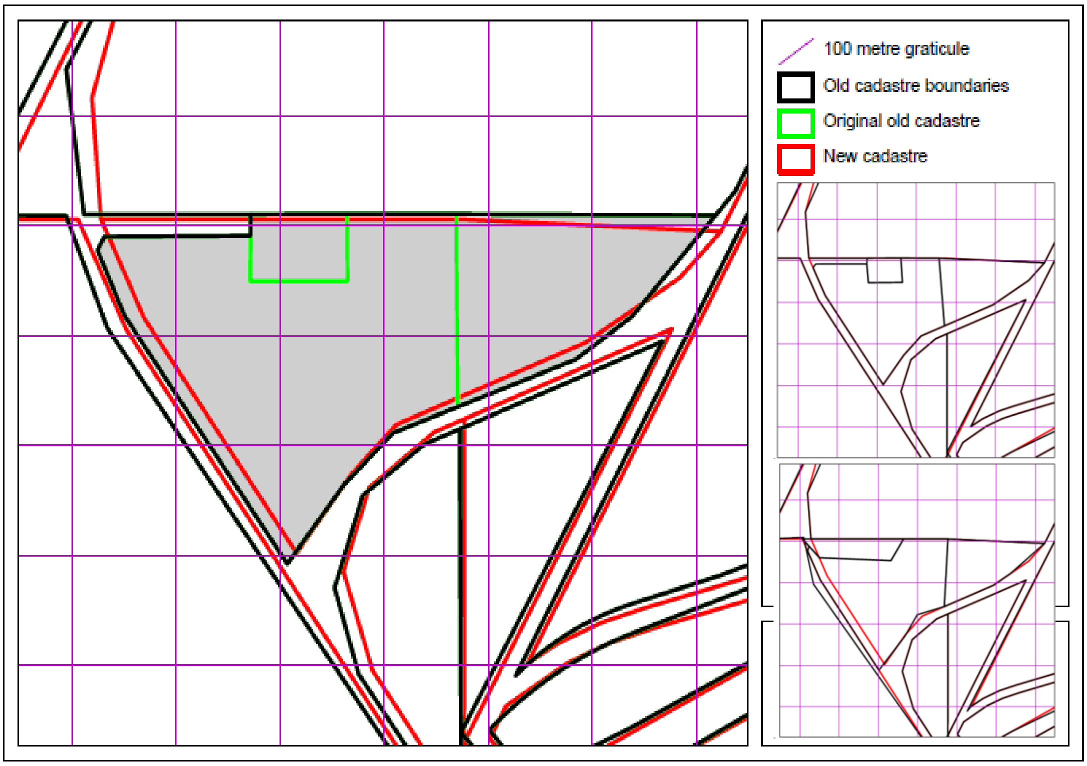

Figure 9.34 An area where parcels have been amalgamated in the new cadastre

9.5.6.4 Benefits of the PMA1 algorithm

Figure 9.35 shows another area from LGA11. The main map shows the old cadastre, new cadastre and the ShiftGen generated links. In this case, most of the riparian boundaries have been matched so that most of the shift vectors in this image have been created using the PMA1 algorithm. The inset shows the resulting old-cadastre adjustment. Although not perfect, the adjustment shows few errors, i.e. where the new cadastre boundaries drawn in red are showing; the maximum distance that the adjusted old-cadastre boundary deviates from the new-cadastre boundary is less than four metres.

Figure 9.36 shows the same area processed using ArcGIS. The adjustment result, in this case, shows many more errors. The maximum deviation here is more than 12 metres. In addition, there are some distorted boundaries in the adjusted old cadastre.

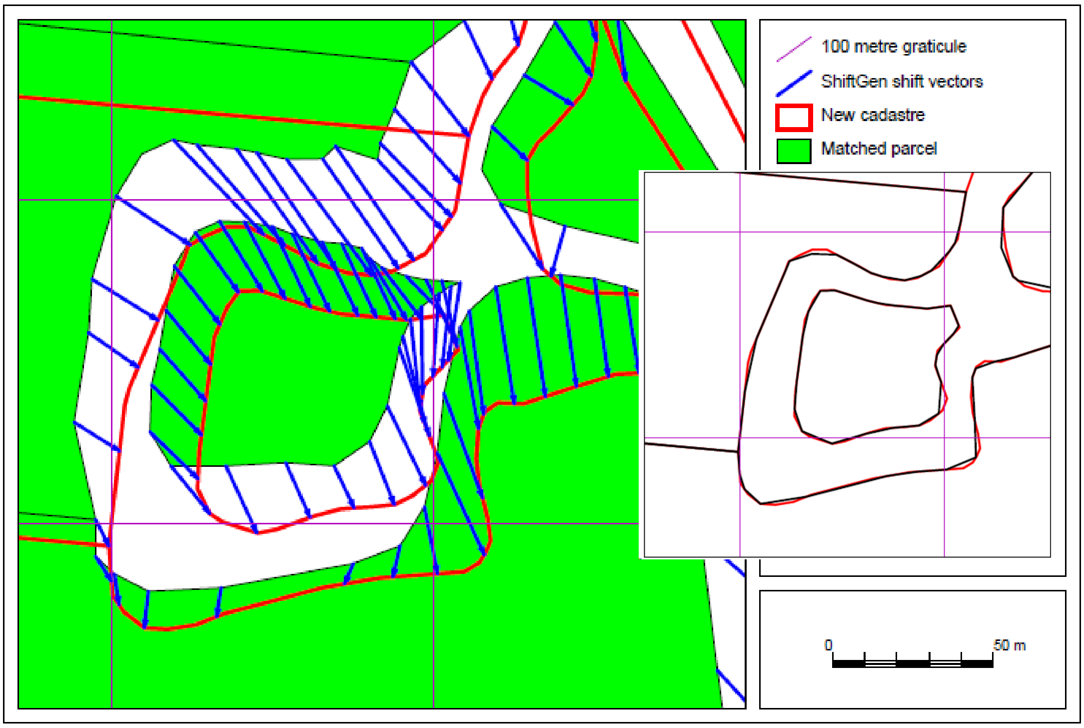

Figure 9.35 ShiftGen results on a riparian boundary

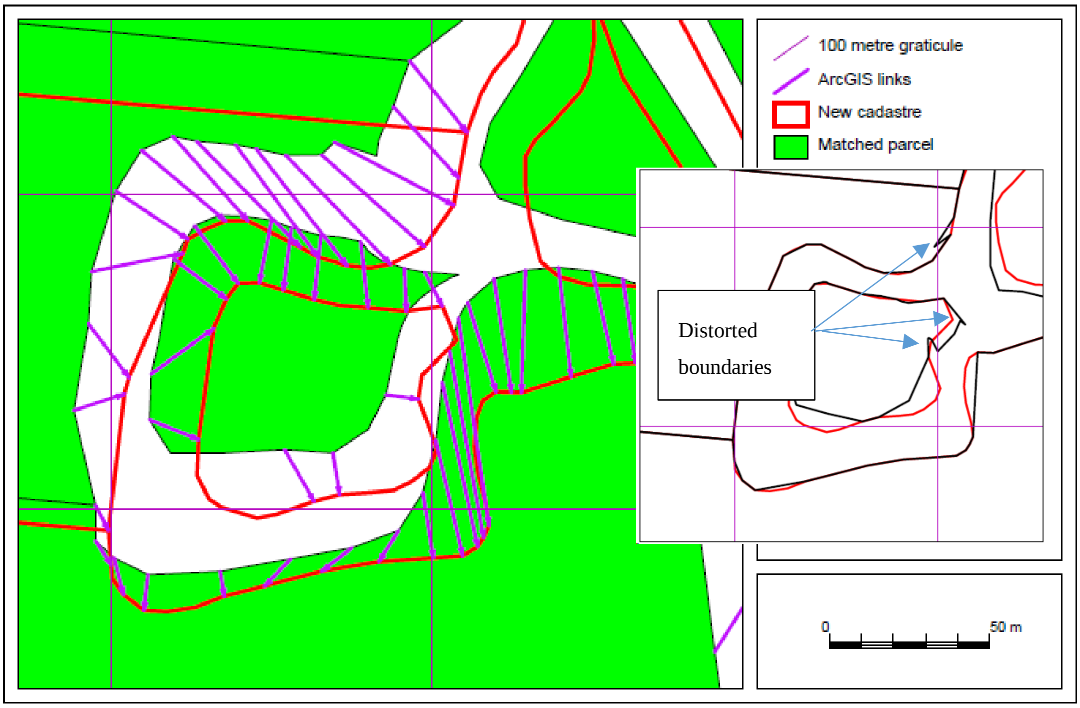

Figure 9.36 ArcGIS results from the same area as shown in Figure 9.35

9.5.6.5 Using boundary direction information

Figure 9.37 shows part of the rural dataset, LGA03. The map shows the shift vectors generated along some rural road boundaries. Most of the vectors have been created from the PMA3 algorithm which makes use of the parcel boundary direction to decide on the best point to which to link (see Section 7.8.2). The inset shows the same area after adjustment of the old cadastre using the shift vectors shown in the main map. Only two adjustment errors, indicated by the visible new cadastre red boundary lines, have occurred in this area, implying that an operator checking the correctness of the results before applying the shift vectors to the adjustment of dependent layers would need to do little work correct the vectors in those two areas.

Figure 9.38 shows a map of the ArcGIS links generated in the same area. The inset shows the result of adjusting the old cadastre using those links. From the number of new cadastre boundary lines showing after the adjustment, it is clear from the map that a number of links in this area would need to be manually replaced with correct links.

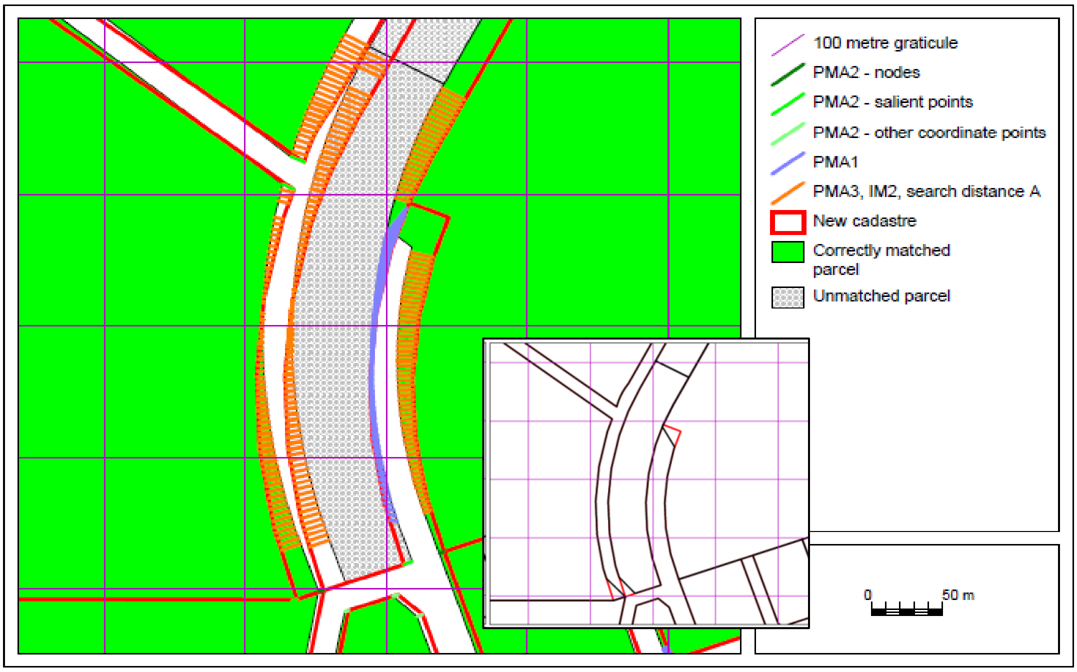

Figure 9.37 ShiftGen adjustment on road boundaries

Figure 9.38 The same area as in Figure 9.37 processed using ArcGIS

9.6 Achievement of research objectives (see Section 1.4)

The outcome of this research has been the set of algorithms detailed in Chapters 4 to 8. These were incorporated into a single executable test program, ShiftGen. Figure 9.39 illustrates the simplified workflow resulting from this research and should be compared with Figure 1.7 and Figure 1.8. The need for user supplied search-distance parameters has been removed so that, using this solution, there is no requirement for a trial and error process to determine the optimum parameter values. Nor is there any need for an intermediate manual process, i.e. assessing the results, when using Workbench, or manually removing intersecting links, when using ArcGIS. The aim of the research, to simplify the spatial adjustment workflow, has, therefore, been achieved.

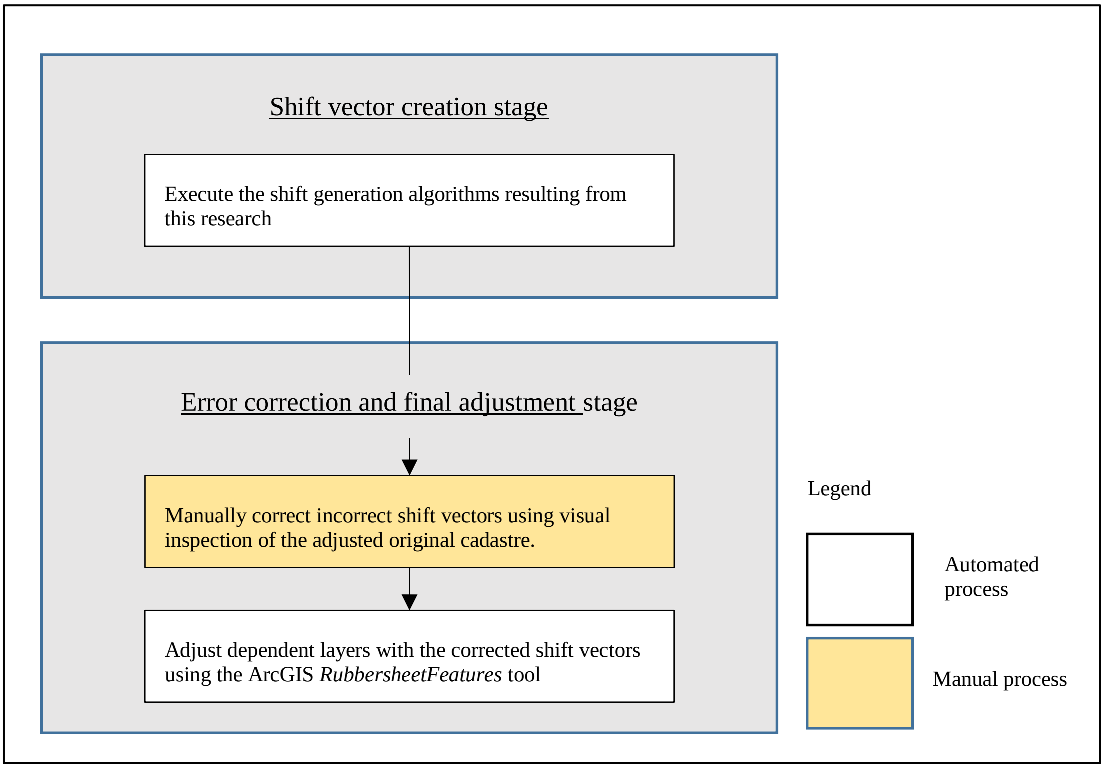

Figure 9.39 The final workflow

9.6.1 Remove the need for user supplied search-distance parameters.

This objective (Objective 1) has been achieved. The ShiftGen program that encapsulates all the algorithms developed for this research requires no user supplied parameters.

9.6.2 Attempt to improve on existing solutions in correctly matching control points

This objective (Objective 2) is believed by the author to have been achieved even though there is no definitive way to determine how many of the control point matches are correct. It was hoped that the reverse shift vector matching process described in Section 9.5.1 would provide an objective method but there are two problems here:

(a) Many of the shift vectors created by the algorithms PMA1 and PMA3 do not terminate on a point in the new cadastre, instead, they terminate somewhere along a line section between points. Therefore, it is not to be expected that a reverse shift vector will exists for these vectors.

(b) As experience with the ArcGIS results showed, in areas with many identical parcels and large parcel movement, it is possible for pairs of matching reverse shift vectors to both be incorrect, although incorrect identical reverse shift-vectors have not, so far, been detected in any of the ShiftGen results.

Table 9.11 shows that more than 97% of the PMA2 shift vectors generated by ShiftGen on all the points from the twelve LGA datasets resulted in identical reverse vectors; the ArcGIS results for the same datasets show a value of less than 66%.

It is not possible to prove that the maximum possible number of correct shift vectors has been generated for any given dataset; indeed, it is almost certain that this is not the case as all the constant and computed threshold values used by the algorithms have been a compromise between locating too few correct matches and too many incorrect ones. However, comparison of the numbers of shift vectors created by the algorithms developed for this research with the number of links created by ArcGIS for each dataset (see Table 9.10 and Table 9.11), and comparison with the old-cadastre adjustment results arising from the ArcGIS links (see Section 9.5.4) indicate that the algorithms developed here are delivering superior results, especially on the rural datasets.

The results reported in Section 9.5 demonstrate that the algorithms developed for this research deliver more correct control-point matches than the ArcGIS tools executed on the same datasets. The images in Section 9.5.4 also show examples of areas where the adjustment results arising from the shift vectors generated by the ShiftGen program are superior to the ArcGIS adjustment results.

Due to the complexity of its operation, it has not been possible to carry out the same comparison using the Workbench software except for LGA01 where example results supplied by Spatial Tapestry were available. Workbench created 1,733 shift vectors using the optimised parameter values provided with the example whereas ShiftGen created 2,272.

The comparison with ArcGIS conclusively demonstrates the improved results from ShiftGen.

9.6.3 Determine how to add automation to the current manual error correction processes.

This objective (Objective 3) has also been achieved (see Section 8.2). Owing to the size of the datasets it has not been possible to examine every crossing, touching and other eliminated vectors to determine whether these vectors were correctly deleted. Inspection of the smaller DA datasets and the resulting old-cadastre adjustment has indicated, to the satisfaction of the author, that the erroneous vector removal algorithms described in Section 8.2 are operating optimally.

The automatic removal of shift vectors that cross or touch leaving only correct vectors in the shift-vector layer has been described in detail in Section 8.2. Inspection of the results has shown the process to be highly effective in removing the majority of incorrect vectors although it is not possible to quantify the results in the absence of an objective knowledge as to which vectors are correct.

9.7 Solutions to the specific problems – summary

For the complete solution finally implemented, several specific problems that needed to be addressed were mentioned in Section 1.5, vis. cadastral parcel matching, parcel boundary matching, and parcel boundary classification. This section summarises the results obtained from the algorithms developed for each of these problems.

9.7.1 Parcel matching (see Chapter 4)

Table 9.1 shows that, overall, for the datasets with available UIDs, 96.33% of over 21,000 parcels were correctly matched using the algorithms developed for this research. This compares well with the 87.6% reported by Kim et al. (2018) on a much smaller number of polygons (222). The 1:M, M:1 and M:M problems, also mentioned by Kim et al. (2018) and (Ruiz-Lendínez et al., 2017), have been addressed and largely solved by the subdivision/amalgamation components of the solution (see Section 4.2.2.2); some exceptions to that solution have been mentioned in Section 8.3.2. It is interesting to note that, although the shape criteria arrived at in this research are different from the metrics proposed by Kim et al. (2018), both solutions involve a fusion of the shape criteria to arrive at the match/no match decision.

The use of block matching (see Section 4.2.4) prior to parcel matching on heterogenous datasets has facilitated correct parcel matching, even on small parcels where there is no area of overlap.

Similar match rates to those on parcels having UIDs were achieved for datasets without matching UIDs (see Table 9.2) although, in the absence of any objective method for evaluation, inspection of the thematically mapped results was the only method initially available for determining the correctness of the matches. Later, incorrect parcel matches could also be detected from the poor old-cadastre adjustment results and incorrect areas of matched parcels after adjustment (see Section 8.3.4).

9.7.2 Boundary matching (see Chapter 6)

In the absence of any objective methods for checking the correctness of the boundary matching, only subjective inspection of the thematically mapped results was available. However, the correctness of the boundary match affects the generation of the PMA1 shift vectors and, therefore, the final old-cadastre adjustment. The observed adjustment results have indicated that the boundary matching algorithms are achieving correct matching in most cases although there is no way to quantify this.

9.7.3 Boundary classification (see Section 6.5)

Once again, there is no objective measure for determining the correctness of the boundary classification but the correctness of the classification affects the final old-cadastre adjustment results because, depending on the classification, different point matching algorithms are executed. The adjustment results observed during this research have suggested that the boundary classification has been instrumental in improving shift-vector generation and rubber-sheeting results on long riparian boundaries.

9.8 Methods for locating potential errors.

The methods for locating potential errors, described in Section 8.3, have evolved during this research and have aided the efficient location of poor or incorrect shift vectors. The methods have been used extensively by the author to identify locations where the adjustment results were incorrect, enabling the iterative improvement of all the matching and classification algorithms. It is believed that these methods would be of value in any commercial solution developed in the future for the improvement of operator efficiency at the manual error checking stage.

9.9 Unexpected outcomes

Some unexpected outcomes have arisen during this research which may be of interest to cadastral custodians. For example, the parcel matching process has resulted in the identification of slivers, the location of duplicated parcels in one or other cadastral layer, the identification of wrongly polygonised roads, and the identification of potentially incorrect UIDs. Locations where adjustments have shown inaccuracies in urban areas have frequently occurred in areas where there is a node ambiguity, i.e. there is one node in one cadastre whereas there are two at the matching location in the other cadastre, a situation which may indicate that a correction to the new cadastre is necessary.

9.10 Unsolved problems

This section will discuss problems which have remained unresolved during this research. Several of these would be topics for future research and are discussed further in Section 11.3

9.10.1 Truncated corners

In some cases, especially in urban areas, a street corner on a parcel in one cadastral layer has been truncated where, in the other, it has not. Thus, there are no corresponding corners to be matched. This situation has been observed to be common in the urban areas of the available datasets.

Figure 9.40 shows an area where two parcel corners were truncated in the new cadastral layer (red) but not in the old (black). The figure shows the unadjusted cadastre on the left and the adjusted old cadastre on the right. During this research, an effort was made to identify these situations but no reliable solution was found. In the example shown, no shift vectors were created for the old-cadastre corners because the point matching algorithm was unable to find an appropriate single point in the new cadastre to which to match. Manual addition of shift vectors would be required to correct this situation before an adjustment on spatially dependent layers is attempted. Ways in which a human operator could be alerted to these locations have been described in Section 8.3.1; in the case illustrated there would be unmatched points on the two unmatched corners.

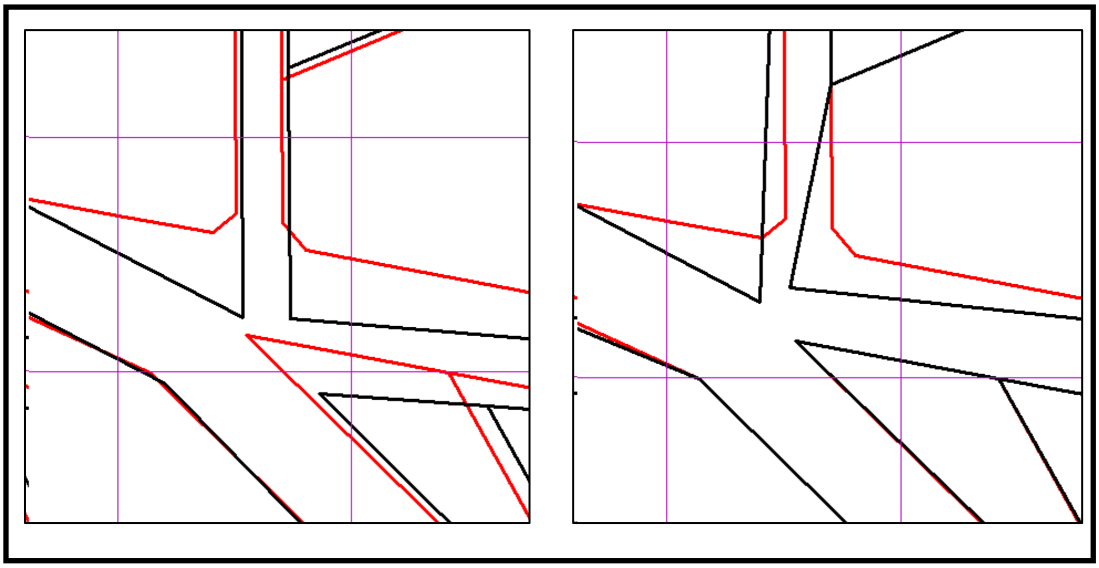

Figure 9.40 Truncated corners (100m graticule)

9.10.2 Missing vertices

The algorithms used by rubber-sheeting processes operate on the individual vertices of the dataset to be adjusted. In other words, only where a vertex is present can there be any change in the location of the vertex after adjustment. This can result in apparent adjustment failures. Figure 9.41 shows a location on a rural road where the original data capture process appears to have omitted some vertices. The ArcGIS *RubbersheetFeatures* tool failed to adjust the straight line in the centre for the reasons just described. There were no old-to-new shift vectors created at this location because, using the algorithms described in this thesis, all the shift vectors originate on old-cadastre points.

Consideration was given to using the reverse shift vectors from the new cadastre points to the old cadastre to handle this difficulty. Figure 9.42 shows the reverse vectors (blue) created by processing the points on the new cadastre and locating intersection points in the old cadastre. These would not improve the adjustment, however, as rubber-sheeting algorithms operate only vertices. It would be necessary to add additional points to the old cadastre at the intersection points of the reverse vectors. However, the additional points would also need to be added to any spatially dependent datasets that used that boundary. A fully developed solution could employ this technique. If adopted, the technique would allow the use of non-identical reverse shift vectors for identifying matching errors for all point types.

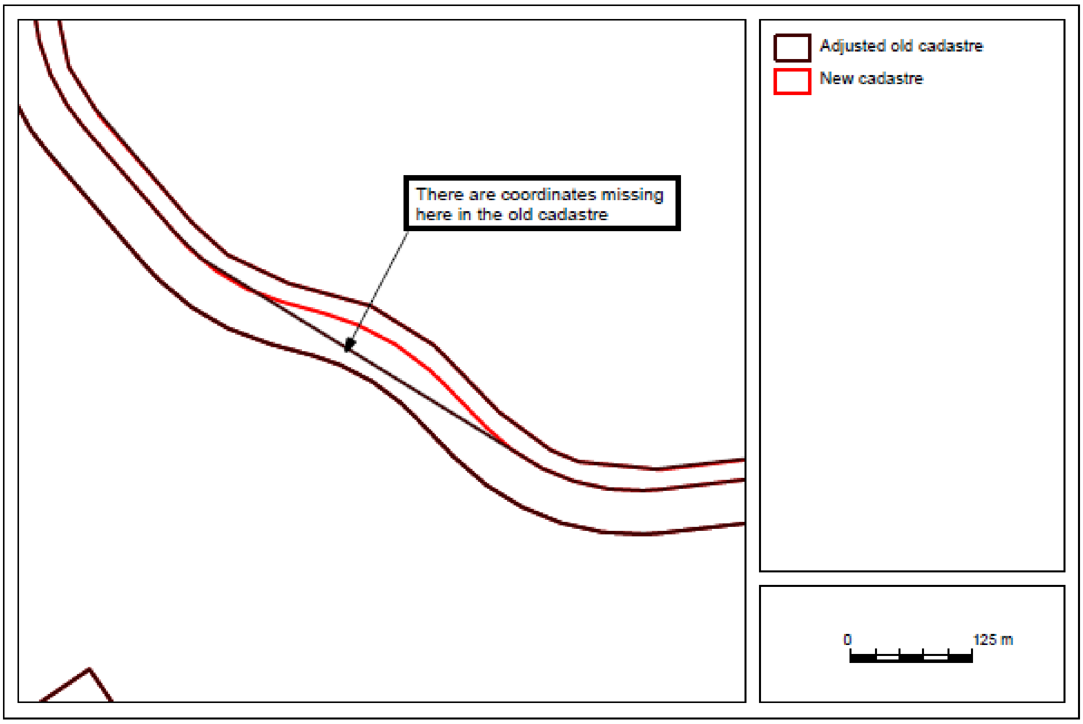

Figure 9.41 A failed adjustment where vertices are missing.

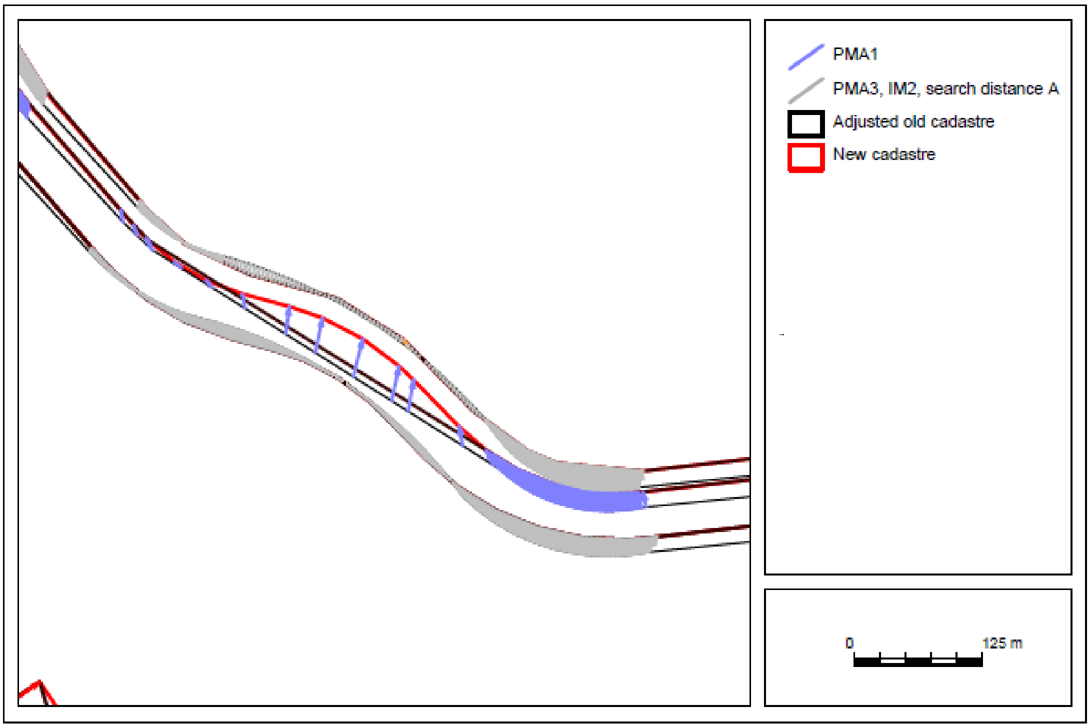

Figure 9.42 Reverse shift vectors

9.10.3 Problems peculiar to road and creek boundaries

Point matching problems can arise because a road or a creek may be represented as a single line in one layer and dual lines in the other. Such problems have been discovered in several locations in the large rural LGA datasets. In one location, a road was digitised as two lines in one layer and three in the other; perhaps the road had become a dual-carriageway. Under these circumstances, where there are more boundaries in the new cadastre than in the old, the algorithms described here cannot always determine the correct new boundary to which to link. Figure 9.43 shows an example where a road digitised as a single line in the old cadastre has been represented by two in the new cadastre.

Consideration was given to post-processing the vectors in this situation and taking a majority vote on preferred direction. However, only a human operator would be able to ascertain that the majority vote is correct and, if it were incorrect, the operator would need to manually correct an even larger number of incorrect vectors. It would be preferable, therefore, in any fully developed application, to provide a tool to reverse the direction of a selected group of vectors and automatically link them to the intersect location on the other boundary; such a tool would allow an operator to correct this situation efficiently.

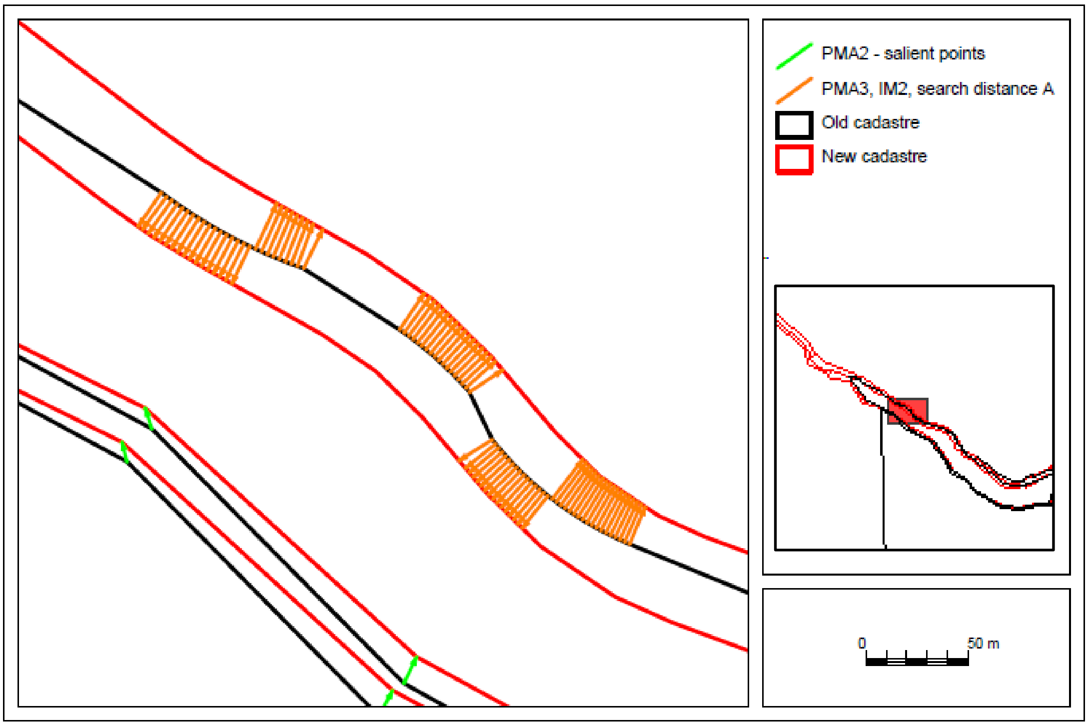

Figure 9.43 Differences in road representation in the old and new cadastre

Problems specific to road casements can arise even where both road boundaries are represented in both the old and the new cadastre. In these conditions, the PMA3 algorithm can sometimes create a link to the wrong boundary. This problem arises from the nature of the shapefile format where each boundary is present twice in the data if the road itself is polygonised and the algorithm cannot always determine the correct duplicated boundary to which to link. The PMA3 algorithm has been designed to slightly favour a link in the parcel centroid shift direction but examination of problem areas indicates that this weighting is not always reliable as shown in Figure 9.44. The presence of deleted shifts (in magenta) in this area would draw attention to the problem at the error checking stage described in Section 8.3.2. The inset shows the result of adjusting the old cadastre without first removing the incorrect shift vectors.

This problem does not arise on matched boundaries processed using PMA1.

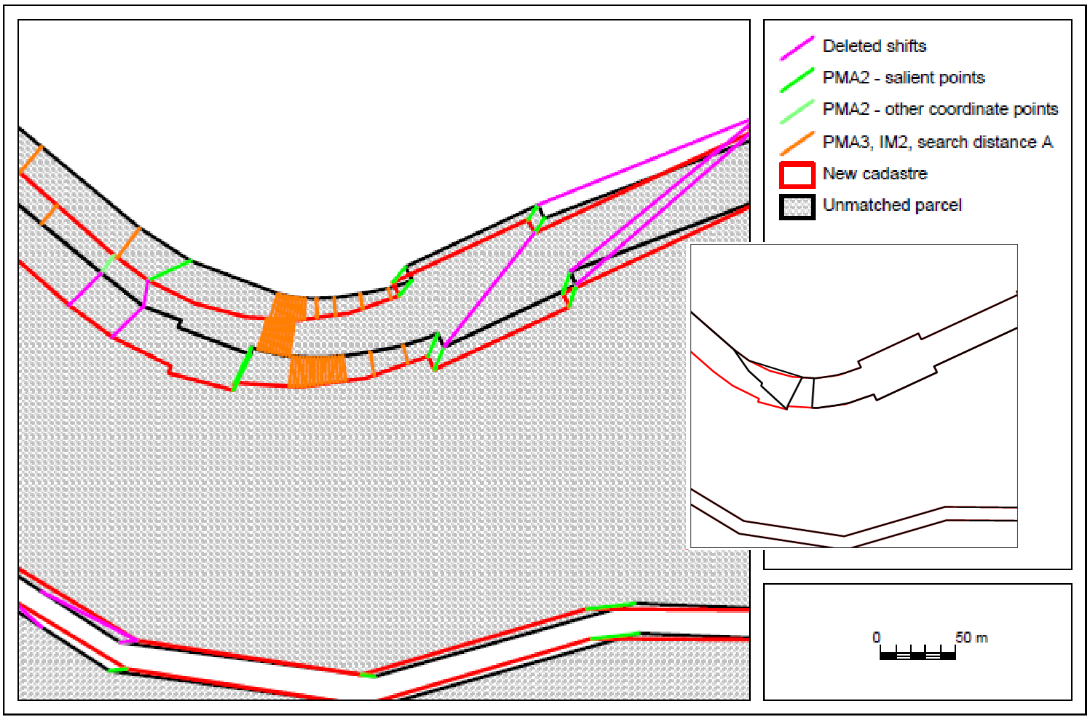

Figure 9.44 Incorrect PMA3 shift vectors

9.10.4 Topological ambiguities

Topological ambiguities can occur when a single node in one layer has become two nodes in the other or vice versa. Figure 9.45 illustrates a case where a single node at the junction of four parcels in the old cadastre has been replaced by two nodes, each at the junction of three parcels, in the new cadastre. Further, a drainage pipe from a spatially dependent dataset lies in the same area making it important to resolve the ambiguity. The unadjusted drainage pipe is drawn in yellow in the figure.

The error in this area would be visible to an operator because of the visibility of the red new-cadastre boundary line which has remained unadjusted (the inset shows the results of a trial adjustment). In this case, it is not clear from the vector map which cadastre represents the nodes correctly. However, a check with Google Maps (also inset and with the fence lines highlighted) shows that the newer cadastre is correct. This suggests that, if aerial photography were to be made available as a background layer in any complete solution, ambiguities of this type could be resolved.

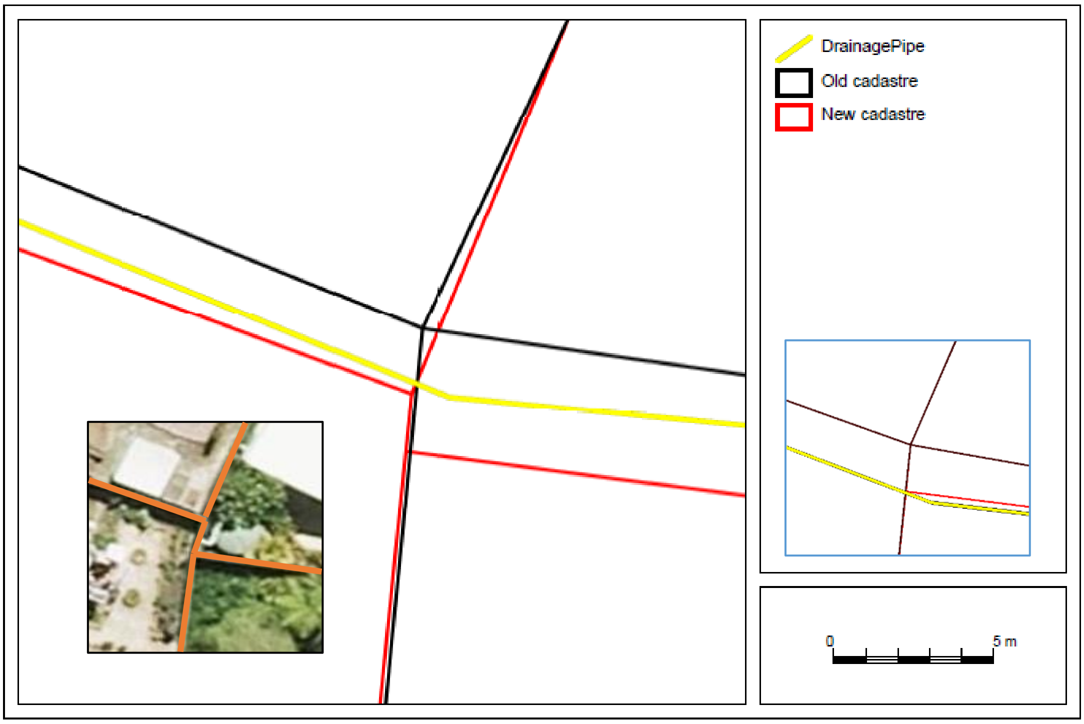

Figure 9.45 One node becomes two

Sometimes, topological differences can be so complex that even a human operator would find it impossible to create correct shift vectors. Figure 9.46 illustrates one such location; the inset at top right shows the shift vectors (green) that were created by the PMA3 point matching algorithms. It may be unlikely that any spatially dependent dataset would be affected by the matching failure at locations of this type, i.e. complex creek boundary changes, but this situation again highlights the necessity for the operator to have access to those datasets for display. In this case, a planning zone boundary runs vertically through the centre of this area and would be incorrectly adjusted (as the inset on the right shows) were the incorrect shift vectors not to be removed beforehand. This situation also suggests that redundant vertices with a turning angle of 180° should first be removed from the dependent datasets, or should be ignored by rubber-sheeting algorithms; the incorrect adjustment of the planning zone boundary would not have occurred in that case.

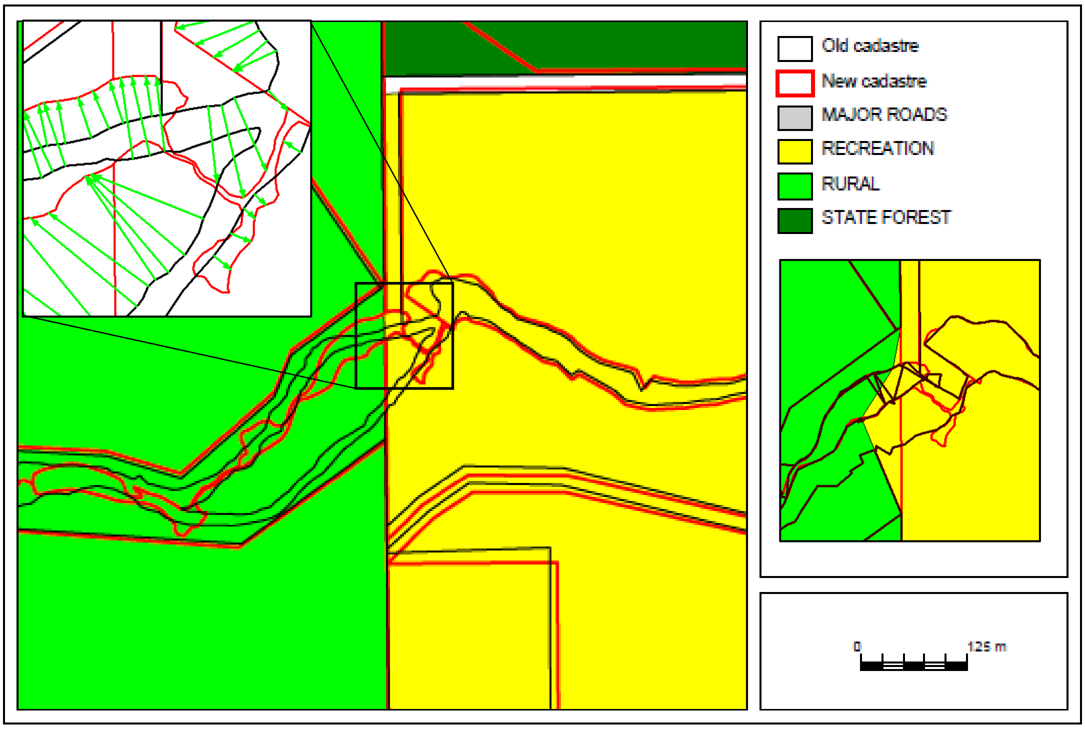

Figure 9.46 Even a human operator would have difficulties here

9.11 Summary

This chapter has illustrated the successful results achieved by this research both numerically and in the many maps. It has also provided examples of comparative results from ArcGIS and has described how the research objectives have been successfully achieved. Unsolved problems have also been discussed.

The next chapter discusses the many ideas that have arisen during the course of the research concerning practical aspects of cadastral management and spatial adjustment of dependent datasets.

10 DISCUSSION

During this research, insights have been gained concerning the effects of data quality, data structures adjustment of certain types of linear features, and specific issues regarding some cadastral points and spatially dependent-dataset types. They are discussed here.

10.1 Overview

Section 10.2 discusses ways in which pre-processing the data would result in addition correct control point matches. Section 10.3 discusses the pros and cons of using the automated parcel matching process when matching UIDs are present in the data. Section 10.4 presents some thoughts on the possible value of cadastral-custodian-provided exclusion zones. Section 10.5 discusses the ways in which a cadastral custodian can assist client organisations having cadastrally dependent datasets. Section 10.6 mentions the need for some special precautions during the adjustment process when applied to road boundaries and power lines. Section 10.7 discusses the authors regrets at the loss of topological data structures from today's spatial databases. Section 10.8 explains why the availability of background layers would be important to the manual checking process in any fully developed solution. Section 10.9 talks about the ways in which the spatial adjustment of riparian boundaries might, or might not, be necessary. Section 10.10 raises some issues that arising from the use of the PMA1 and PMA3 algorithms when adjusting point based dependent datasets and Section 10.11 discusses issues arising from the adoption of GDA2020 in Australia.

10.2 Pre-processing the data

The research for this thesis has been focussed on removing the need for user suppled search-distance parameters for the spatial adjustment process, whilst maximising, as far as possible, the number of correct shift vectors produced. However, it has become apparent that the cadastral custodian or the client GIS database administrator would be able to improve shift vector generation results by carrying out some operations on the cadastral datasets before attempting the spatial adjustment process. These issues are discussed in the following subsections.

10.2.1 "Cleaning" the datasets

Examination of the mapped results for LGA08, where initially the percentage of points matched was lower than for the other datasets revealed many unmatched nodes. The poorer results were unexpected as node points, especially on matched parcels, are the simplest points to match correctly. In this case, the reason was found to be because of the extremely unclean nature of the new-cadastre layer. In one location no less than 16 vertices were found all within less than one millimetre from each other. For most of these points it was computationally impossible to calculate a bisector azimuth because of the tiny separation distances with the result that these points were ignored by the PMA2 algorithm and the match failed.

This problem was solved by snapping nodes and edges using the ArcGIS snap tool with a small tolerance. After this the complete process was rerun; an inspection of the remaining unmatched nodes using a GIS viewer revealed that the majority were on nodes where there was no corresponding data in the new layer or there were topological differences at the location of the node, for example, where there were two nodes in the old cadastre but only one in the new or vice versa. After the snapping process, the number of points matched improved significantly.

The above example indicates that poor-quality data can be expected to adversely affect point matching and consequently the output from the adjustment tool. Pre-processing the dataset to ensure that all nodes are snapped and topological errors are removed before commencing point matching would reduce matching errors but it would need to be established that the time required to achieve this is more than recovered at the error checking stage.

At any site where ArcGIS is available, the *CheckGeometry* tool (Esri, 2016c) can generate a list of all polygon features that do not have correct geometry. This tool, or something similar from another package, could be used to discover errors in the topology of the old and new cadastral layers and eliminate them before attempting the spatial adjustment process. Alternatively, the *RepairGeometry* tool can be used to fix any geometry errors automatically.

If the work were to be carried out by the cadastral custodian before releasing an updated version of the cadastre, that would certainly reduce the time needed by client organisations to carry out their own dependent layer adjustments.

10.2.2 Densifying the vertices

Mention has been made of the effect of missing vertices on the adjustment process, i.e. rubber-sheeting algorithms only adjust the vertices, “dragging” the connected line segments with them (see Figure 9.41). This problem could be solved by densifying the vertices in the cadastral layers, at the cost of increased execution times. However, any layers derived from the cadastral boundaries, such as planning zones, would also need to be similarly densified to be successfully adjusted by ArcGIS or by any other adjustment software based on similar rubber-sheeting algorithms.

10.3 Parcel matching – use of unique identifiers

No attempt was made during this research to make use of existing matching UIDs except to check the correctness of the parcel-matching algorithms developed here. Any real-world application developed for the purpose would obviously benefit from using these UIDs when they exist although, as this research has shown, carrying out the matching process even when there are UIDs present in the data can expose UID errors. For example, Figure 10.1 shows a location where the values of the unique identifiers on two adjacent parcels have been exchanged resulting in them being flagged as wrongly matched.

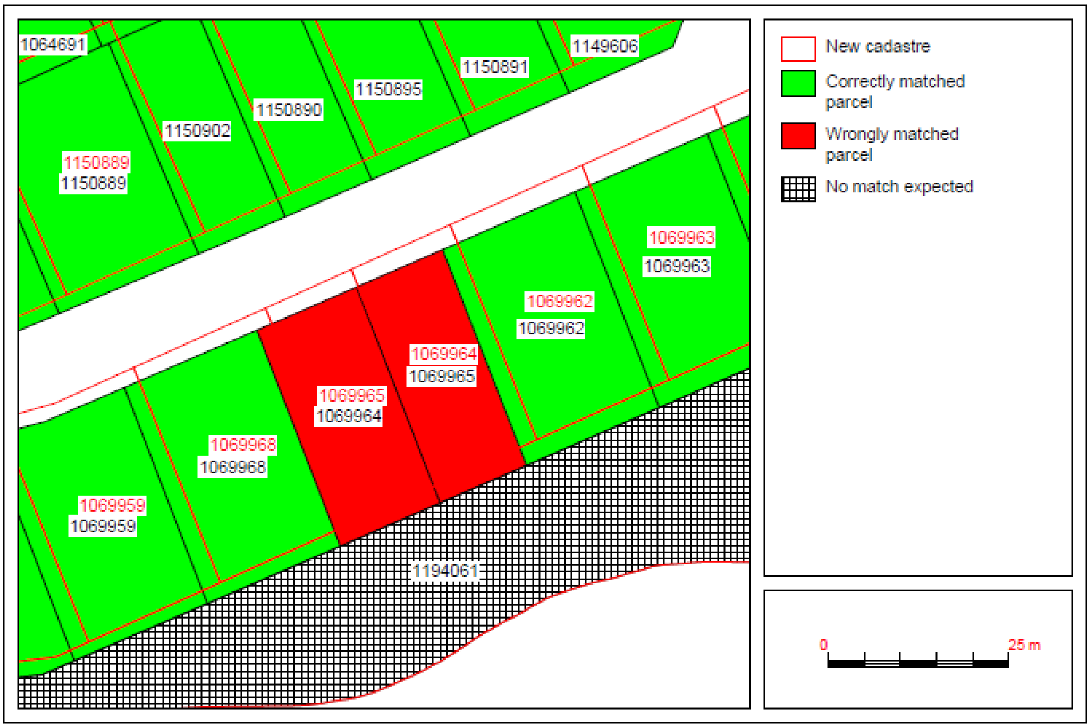

Figure 10.1 A pair of parcels where the UIDs have been exchanged between versions

Any application developed to undertake the spatially-dependent-datasets adjustment process could benefit from providing an initial stage of parcel matching to allow an operator to correct obvious errors such as the one illustrated above.

10.4 Exclusion zones for large subdivisions

Carrying out the subdivision identification process was found to improve the final adjustment by reducing the number of areas that required manual intervention because of incorrect point matches. However, many large new subdivisions were not identified because road voids caused the area check to fail (see Section 8.3.2). Many matching failures would be avoided if the shift generation algorithm allowed the input of fences to define areas to be excluded from the adjustment process.

It is possible that cadastral custodians would have such data readily available and could supply it to client organisations wishing to adjust their spatially dependent datasets.

10.5 Custodian responsibilities

In Australia, the cadastre is typically administered by a state government department but many of the spatially dependent datasets are administered by other departments or utility companies such as water and electricity. These other organisations would, therefore, have the need to carry out adjustments when a newer version of the cadastre is released. Most of the pre-processing operations described in this section could theoretically be carried out by the custodian before issuing an updated version to clients. Indeed, when considering the very time-consuming work involved in carrying out these adjustments, overall efficiency improvements amongst government departments would be considerable if the cadastral custodian were to provide clients with a correct set of shift vectors with each updated version of the cadastre (some already do so, (Merritt, 2009)). These could then be applied to each spatially dependent dataset and duplication of effort between different government departments would be eliminated. It would, of course, be essential that the adjustment process was executed on every dependent layer each time an updated version of the cadastre was released – a step that may not always be feasible given the potentially large amount of work involved.

Where an organisation cannot, for any reason, update every dependent dataset after every cadastral update, it would be important to retain the version of the cadastre from which the dependent dataset was derived, unless the custodian of the cadastre can provide an archival copy when required. The ShiftGen solution described here would only be suitable where a cadastre from which the dependent dataset was derived is available.

10.6 Spatial adjustment of linear features

Spatial adjustment of certain types of spatially dependent datasets may require a more sophisticated approach than that provided by many commercial solutions. For example, when adjusting power lines, it is important that the lines between power poles remain straight after adjustment, road widths also should remain unchanged, road centrelines should remain central in the road void, and, where the parcel boundaries along a stretch of road between corners are straight and parallel before adjustment, they should remain straight and parallel after adjustment, (Merritt, 2005). The removal from dependant datasets of all redundant vertices falling on a straight line, as implemented in the point creation stage of the ShiftGen program, would obviate some of these concerns; if that step is not implemented, care must be taken at the adjustment stage.

10.7 Data structures

It has become apparent during this research that the abandonment of topological data structures such as VPF by most GIS vendors has been a loss to the spatial industry. A large part of the software development undertaken for this project involved, effectively, the rebuilding of topology. This may well be true of many other spatial applications that have been or will be developed. The cleaning process discussed in Section 10.2.1 would be unnecessary if a topological data structure was employed to store the data. In summary, the following software development efficiencies could be achieved by the existence of a topological data structure:

(a) There would be no need to develop code to identify nodes. Execution times would also be reduced.

(b) Because each boundary is stored only once, the boundary-matching processes would be simplified. Some of the problems highlighted in Section 9.10.3 would not occur.

(c) Adjacency information is explicitly stored in the dataset structure; this could aid the future research suggested in 11.3.2.

(d) In a topological data structure, it is possible to store attributes against the parcel boundaries. Thus, it would be possible to assign boundary types at data capture time. This would obviate the need for complex computer algorithms that can only partially achieve correct classifications.

10.8 Background layers

It is suggested that, in any complete spatial adjustment application developed in the future, to minimise operator time during the process of correcting the shift vectors before final spatially dependent dataset adjustment, he or she should always have the option of displaying the unadjusted spatially dependent dataset(s). This can sometimes result in negating the need to correct every error. For example, in the map shown in Figure 10.2, there are small adjustment errors near the centre of the map; these can be seen because the new-cadastre boundaries (red) are visible. The unadjusted planning scheme map is also shown. The entire green area, in this case, is a single planning zone thus making shift vector correction at this location unimportant.

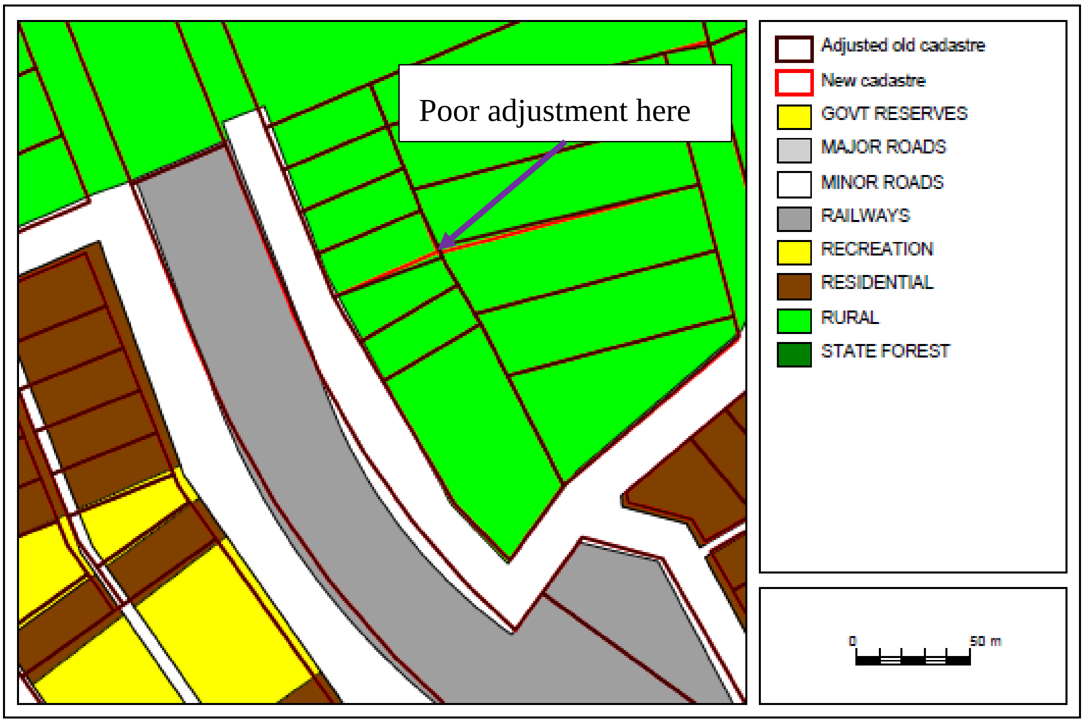

Figure 10.2 An adjustment error against a background of unadjusted planning zones

Displaying the unadjusted planning-zone dataset at the inspection stage, in this case, has revealed that correction at the location of the poor adjustment would be unnecessary if the planning-zone layer is the only spatially dependent dataset because there is no coincident zone boundary. Each spatially dependent dataset can have a different effect on the decision as to whether a correction is necessary. If the spatially dependent dataset were census mesh blocks, for example, it may well prove necessary to correct the error at this location.

10.9 Adjusting riparian boundaries

Depending on the spatially dependent dataset to be adjusted, there may or may not be a need to adjust riparian boundaries. In many cases, if the data to be adjusted is point data representing, say, highway authority assets, it is possible that there are no assets located near the stream or river boundaries so that such boundaries would not need to be matched. However, when adjusting polygon data such as planning zones, it is possible that some boundaries do follow riparian boundaries. In the latter case, the PMA1 and PMA3 algorithms used for points on these boundaries typically produce particularly good "cosmetic" results, i.e. the adjustment is likely to appear to be perfect and that may be all that is necessary.

It would be a simple matter, were any commercial application to be developed using the algorithms described in this thesis, to implement a user option to use or ignore riparian boundaries in the adjustment process. If such an option were to be offered it would be necessary to implement a preliminary process allowing the user to check the automated assignment of boundary type (see Section 6.5) to make sure that road casements are not wrongly flagged as riparian boundaries or vice versa.

If adjusting riparian boundaries is deemed to be essential, the operator, at the manual checking stage, could be guided by error-checking software to each such boundary in turn and permitted to add or delete shift vectors where appropriate when the adjustment is not perfect.

10.10 Use of the PMA1 and PMA3 matching algorithms for adjusting point datasets

The PMA1 and PMA3 algorithms do not in all cases match old cadastre points to new cadastre points. In most cases, these algorithms match the old cadastre point to a location on the corresponding boundary in the new cadastre and interpolate a point along the boundary prior to constructing a shift vector. It may or may not be advisable to use PMA1 and PMA3 on road or rail-track boundaries; if the spatially dependent dataset represents point assets such as lighting poles, rail signals or drainage maintenance holes, the interpolation algorithms cannot be absolutely guaranteed to produce accurate results when the shift vectors resulting from interpolation are applied to the spatially dependent data, especially along particularly convoluted boundaries. A user option to use PMA1 or PMA3 on road boundaries could easily be provided in any fully developed solution.

10.11 Points that fall on the world polygon

Typically, spatial adjustment processes are applied to subsets of a complete cadastral dataset. The datasets used here have all been areas delineated by local authority boundaries but often an adjustment only needs to be applied to a smaller area such as a new subdivision. In other cases, the data may be tiled by more arbitrary boundaries. Whichever is the case, it will always be important to attempt to match all points on the outer edges of the dataset to minimise edge-matching problems when the adjusted dataset is reintegrated into a larger one.

10.12 GDA2020

The Australian continental plate is moving in an approximately north-easterly direction by about 7 cm a year (Geoscience Australia, n.d.-a). The GDA94 datum, upon which Australian mapping coordinates are currently based, is now more than 20 years out of date. A new datum, GDA2020, has therefore been defined by the Intergovernmental Committee on Surveying and Mapping (ICSM) (Geoscience Australia, n.d.-a). This datum is based on the projected position of the Australian continent in the year 2020. This change will be of relevance to any organisations managing cadastrally dependent datasets which will need to be adjusted to the new datum.

The adoption of GDA2020 by cadastral custodians is expected to occur over the coming two years from 2018, at which point new versions of a cadastre are expected to exhibit movements of approximately 1.6 metres relative to the GDA94 datum. Fortunately, because the spatially dependent datasets which have been the topic of this research have been captured from or with reference to a cadastre, the movement should not present a problem. Providing that the original cadastre from which the data were derived is still available, the adjustment processes developed here or elsewhere will still facilitate the upgrade of the data to the new locations. Alternatively, Geoscience Australia is providing a range of tools to upgrade any GDA94 based dataset, including an online transformation service (Geoscience Australia, n.d.-b), although these would only be of value for datasets derived from the immediately preceding cadastral version.

10.13 Summary

This section has covered various topics arising from insights gained during the course of the research, in particular: the various steps that could be taken by the cadastral custodian to forestall some of the problems likely to be encountered by any organisation attempting to adjust spatially dependent datasets; the benefits that would be experienced if topologically structured datasets were available for input; the possible problems that could arise from the use of the PMA1 and PMA3 algorithms; and the importance of matching points falling on the outer edges of a dataset.

The next chapter summarises the major conclusions arising from the research conducted for this thesis and makes suggestions for future areas of research.

11 CONCLUSIONS AND FUTURE RESEARCH

This chapter outlines the conclusions drawn from the several stages of the research and makes suggestions as to future research directions.

11.1 Overview

Section 11.2 details the conclusions drawn from each of the major research stages. Section 11.3 covers several suggestions for potential areas for future research. Section 11.4 mentions the limitations of the research and Section 11.5 summarises the main conclusions.

11.2 Research conclusions

This section discusses the detailed conclusions drawn from each stage of the research including those stages arising from the specific problems listed in Section 1.5.

11.2.1 Removing the need for user supplied search distance parameters.

The conclusion drawn from this stage of the research is that it is possible to identify correct control points in pairs of cadastres exhibiting heterogenous spatial characteristics and large and inconsistent apparent parcel movements without the need for a user supplied search-distance parameter, thus fulfilling research aim number 1 listed in Section 1.4.

11.2.2 The Genetic Algorithm (see Section 4.2)

At the genetic algorithm stage of this research it became clear that a GA solution for automatically determining search-distance parameter values would be capable of solving the problem only on homogeneous datasets such as an entirely urban cadastre, and that a GA solution would not be appropriate for mixed urban and rural datasets because the search-distance maxima would need to be smaller for the urban areas and larger for the rural areas. A GA solution could not, therefore, be used without separating the parcels in the dataset according to size, which would only serve to complicate the solution.

However, the parcel matching research that resulted from attempting a GA solution gave rise to the important insight concerning the way in which, where parcels could be matched, the apparent parcel movement could be used to accurately predict the location of corresponding vertices in the new cadastre – at least for urban parcels. This insight became the foundation for the remainder of the research project. A further insight that arose at this stage was that parcel matching in some areas could be facilitated by first matching the cadastral superblocks.

The principal conclusion drawn from the GA stage of the research was that there could be no single values for maximum search distances that could achieve acceptable results when attempting to match two cadastres covering mixed urban and rural areas. Appendix D covers details of the GA research and details the reasons why the approach was discontinued.

11.2.3 Block and parcel matching (see Section 4.3)

Initially, as part of the GA solution and later as part of the complete solution, block and parcel matching algorithms were developed. The conclusions that were drawn from this stage of the research are as follows:

(a) Different parcel-matching algorithms were appropriate to different dataset types, i.e. entirely urban areas showing little apparent movement between cadastral versions and mixed urban and rural areas showing much greater movement.

(b) A preliminary simple parcel matching process followed by statistical analysis of apparent parcel movements can provide enough information to enable automatic selection of the most appropriate matching algorithm, i.e. whether the dataset should be handled as purely urban or mixed urban and rural.

(c) The parcel matching process can be used to identify some, but not all, subdivisions and amalgamations. Removing the subdivisions before attempting any matching processes can increase the number of polygons matched and improve the final point-matching results.

(d) Block matching can result in information regarding the direction and distance of the apparent movement of the block – information that can be used to guide a second-stage parcel matching process in datasets where there has been significant apparent movement between cadastral versions. Block matching was not found to be essential for purely urban datasets with a high degree of positional accuracy in both datasets and little apparent movement between them.

(e) Separation of road and creek polygons from other parcels before creating blocks can result in a greater number of correctly matched blocks.

(f) Parcel matching between two cadastral versions issued on different dates can be a straightforward matter in urban areas where there is little apparent movement between cadastral versions and corresponding parcels overlap in area by close to 100%.

(g) Where parcel UIDs exist, the parcel matching process can identify errors in the UIDs. Matching errors, in this case, can be used to guide an operator to areas that need special treatment or where the UIDs themselves are in error.

The overall result from this stage of the research was that more than 96% of all the parcels from the available datasets with UIDs were accurately matched between cadastral versions using the algorithms developed for this thesis, leading to the conclusion that the algorithms are highly effective.

11.2.4 Creating the point layers from the polygon vertices (see Chapter 5)

The process of assigning many spatial attributes to each cadastral point and then choosing only one point at each unique location allows the software to use just the most salient point at each location; this has increased the probability that a point can be matched correctly.

The conclusion drawn from this aspect of the research is that taking the spatial attributes of each point into account allows the selection of the most salient point at each unique location and further allows the most appropriate algorithm to be selected for matching the point.

11.2.5 Boundary matching (see Chapter 6)

Information regarding which pairs of parcel boundaries match between the old and the new cadastre can be used to ensure the accuracy of intermediate point matching along those boundaries. Where a boundary has been matched, an intermediate point on that boundary can be precisely matched to a point or location on the corresponding boundary in the other cadastre, thereby ensuring the correctness of the match because the proportioning method described in the PMA1 algorithm (Section 7.6) guarantees the creation of correct shift vectors along correctly matched boundaries. The PMA1 algorithm would not have been possible without the prior boundary-matching process.

The conclusion drawn from this stage of the research was that matching parcel boundaries, wherever possible, guarantees correct adjustment results on matched boundaries although problems can arise when one bank of a creek is matched and the other is not (see Section 7.11.3).

11.2.6 Point matching

The conclusions drawn from this stage of the research are that no single algorithm is appropriate for all points and that the methods used should depend on the locational details of the point, such as the distance from its neighbours, the turning angle at the point, the type of boundary it falls on, whether it is located at a node and many other attributes. Taking many of these attributes into account allowed the most suitable matching strategy to be selected.

This research has shown that it is possible to generate shift vectors from old cadastre points to locations on a new-cadastre boundary where there is no corresponding point to which to match. The inclusion of two algorithms, one for points on matched boundaries (PMA1) and one for all other points where no point-to-point match has been found (PMA3), have resulted in increased numbers of shift-vectors, enabling the rubber-sheeting process to deliver a more accurate adjustment, especially on complex rural boundaries.

The results from the algorithms developed, when applied to the twelve full LGA datasets, have delivered point matches for over 97% of the all the points from those datasets, and over 97% of the point-to-point matches resulted in identical reverse shift vectors implying that the control point identifications were correct (see Table 9.11). These results were obtained from datasets covering a variety of different area types, i.e. urban, rural, and mixed urban and rural. The comparison with both the numerical and mapped results from using just the ArcGIS generic tools indicates that the results from ShiftGen are superior (see Section 9.5).; the comparable figures from ArcGIS are 90.27% of points matched and 65.15% identical reverse links.

The conclusions drawn from the results achieved by executing the complete solution on all the available cadastral datasets is that it is possible to correctly match more than 90% of the points from one cadastre to the correct point or boundary location in another without the need to supply search-distance parameters and without any prior knowledge of the nature of the cadastral dataset. The successful completion of this stage of the research fulfils the research aim number 2 listed in Section 1.4

11.2.7 Removing erroneous shift vectors

The conclusion drawn from this stage of the research is that statistical methods can be employed to successfully determine which crossing or touching shift vectors should be removed and thus reduce the amount of manual processing needed in the final stage of the complete spatial adjustment process prior to adjusting dependant datasets. The successful development of this process has satisfied research objective number 3 listed in Section 1.4.

11.2.8 Checking the results

Human intervention will always be necessary to identify and correct any remaining errors. The research has not obviated the need for a human operator to check and correct the automatically generated and automatically corrected results. However, the research has identified many methods for guiding the operator to locations where errors are to be expected such as: missing identical shift vectors from a reverse shift vector generation process; unmatched parcels; unmatched points; vectors eliminated because they crossed or touched another; and locations where the adjusted block boundaries do not match exactly. These methods have proven valuable to the author in progressing the research. It is concluded that any complete solution developed in the future could assist the user by creating an output layer or layers showing potential error locations.

11.3 Further research

11.3.1 Algorithm components

It became clear during this research that many of the matching problems encountered could be addressed by first classifying the spatial features, for example, classifying the parcels into roads, slivers, regular and irregular parcels, classifying the parcel boundaries into road casements, riparian boundaries and regular boundaries, and classifying the individual points into classes such as nodes, corners, salient points, and isolated points. Even the datasets themselves required classification into urban datasets where apparent movements were small enough to use simple parcel conflation, and mixed urban and rural datasets benefitting from the use of block shift vectors. None of this was clear at the start of the research.

The classification process resulted in many different attributes being assigned to the cadastral parcels, boundaries and points and the values of these attributes were then used to calculate threshold values, for example, search distance and direction, for the expressions used in the matching algorithms. In the late stages of the point matching research, minor changes to the algorithms usually resulted in only minor changes to the results often delivering improvements for some point types and worse results for others. The final versions of the algorithms adopted have been documented in detail in the body of this thesis so that future researchers will be able to duplicate and improve on the work.

When considering all the individual elements of the solution, there can be no certainty that all of them are still required or that they could not be further improved. The need for some of the earlier procedural elements of the solution may have been obviated by improvements in later parts for the solution. For example, is it still necessary to identify corners after later improvements to point matching algorithms were implemented? Each of these elements could be removed one at a time and the solutions tested on all the available datasets but each element also affects the algorithms used for other elements so that different combinations would also need to be tested. During this research, many tests have been executed with and without different procedural elements of the solution. However, given the considerable number of combinations of elements, only some of the many possibilities have been tested. The final developed solution is, of course, very much a prototype. It is to be hoped that, should an improved version be developed, some of these questions could be answered by further research.

11.3.2 Adjacency

In theory, it should be possible to incorporate parcel adjacency checking to locate parcel matching errors. This would be especially useful where there are no UIDs available to test the correctness of the match. However, it is not obvious, where such a check indicates an error, which match should be considered correct and which should not. Nor is it obvious where such a check would best be spatially commenced in any given block. Whether such a check could be used to automatically correct the error or only used to alert the operator at the error checking stage would be a topic for additional research.

11.3.3 Bijective block and parcel matching

It is possible that a bijective matching process for blocks and parcels could be used to improve the parcel matching results in the absence of matching UIDs. Bijective matching results could be used to guide an operator to potential areas of error in the same way as the other methods discussed in Section 8.3. Whether they could be used to improve the automated matching process would be a topic for further research.

11.3.4 Bijective shift vectors

When reverse shift vectors are created it is possible that, where an identical shift is not created, the reverse vector will touch or cross another vector in the original set. Appending these new-to-old vectors to the original set before executing the automated error removal process described in Section 8.2 should result in some additional correct vectors.

11.3.5 Manual error checking

The process indicated in the yellow box in Figure 9.39 represents the process by which an operator must assess the results of the shift vector generation process and add or remove vectors as necessary to correct any errors. Although this research has suggested several output layers that could be created to aid this process, it never-the-less constitutes the major part of an operator's workload.

Obtaining a completely correct set of shift vectors is essential before proceeding to dependent layer adjustments. It has not been an aim of this research to determine the most efficient way in which this error checking and correction process should be carried out although it is to be hoped that the large number of correct shift vectors resulting from the ShiftGen solution (see Table 9.11) would help to minimise that workload. There are many ways in which the potential error locations described in Section 8.3 could be processed. Would it be more efficient to correct just some types of errors and then rerun the old cadastre adjustment using the amended shift vectors and then process the remaining errors, or would it be preferable to process the combined error layer described in Section 8.3.8 first? Only testing the alternatives would provide an answer.

Resulting from the development process described in this thesis, several by-products were created to aid the author in the identification of areas where the algorithms were not performing satisfactorily. These by-products would be valuable in any complete solution developed in the future; they can be used to aid the operator to locate errors more efficiently. Some errors located by these by-products would be valuable for identifying areas where the cadastre itself needs correction, in particular: identification of slivers, UID errors detected by the identification of wrongly matched parcels, and wrongly polygonised roads and creeks. All of these errors and others such as unmatched points, unmatched reverse-direction shift vectors, deleted shift vectors, rural road casements, and riparian boundaries can also be used as described in Section 8.3 to guide the operator to areas where the need for manual correction of the automatically generated shift vectors is to be expected. One by-product used during this research was a single dissolved polygon layer (see Section 8.3.8) containing all the unmatched points (buffered), all the deleted shift vectors (buffered), and various other potential areas of error. The exact composition of the potential-error information needed in this layer to achieve the greatest possible operator efficiency would also need further investigation.

11.3.6 Another potential error identification method

Many errors are discovered by drawing the old cadastre, after adjustment using the generated shift vectors, on top of the new cadastre. If the new cadastre is drawn in red, say, and the old in black, then any visible red pixels indicate an adjustment error. This method has been used extensively to identify errors during this research. By rasterising images created in a GIS viewer, it would be computationally possible to count the number of red pixels in the screen image of the map. The results of this count could be used to guide the user to areas that need inspection. It is suggested that the data would need to be tiled into areas covering some predetermined number of parcels before counting the pixels. Further research would be needed to establish the value, or lack thereof, of this approach.

11.3.7 Dependent point-based data on riparian and rural road boundaries

It is possible that the PMA1 and PMA3 algorithms should not be used for adjusting dependant datasets modelling point features (see also Section 10.10). No spatially-dependent point features falling on rural roads or riparian boundaries were available for this research. It is possible the adjustment of point features from transportation or waterway asset data would not be sufficiently accurate when using the vectors generated by the PMA1 or PMA3 algorithms. Further research on the use of these point matching algorithms for the adjustment of spatially dependent point layers is needed.

11.3.8 Identifying subdivisions and amalgamations

Generation of incorrect shift vectors in areas of subdivision has been shown to be reduced by identifying subdivisions and removing their internal boundaries before attempting point matching. However, the algorithms developed here failed to locate subdivisions containing road voids and some areas with complex boundary changes like those shown in Figure 4.36. Further research is needed to improve algorithms for detecting these situations.

11.3.9 Missing vertices

When adjusting the old cadastre, the algorithms used by rubber-sheeting processes operate on the individual vertices of the dataset to be adjusted. In other words, only where a vertex is present can there be any change in the location of that vertex after adjustment. This can result in apparent adjustment failures. Figure 9.41 shows a location on a rural road where the original data capture process appears to have omitted some vertices. There are two possible solutions to this problem:

(a) Densification of vertices. Especially on long rural boundaries this could potentially improve adjustment results and could remove the need for interpolation methods such as PMA1 and PMA3. Since rubber-sheeting algorithms work only on vertices, it would be preferable to densify the dependent datasets with the same vertices used to densify the old-cadastre parcel boundaries to assure the greatest possible adjustment accuracy.

(b) Addition of vertices at reverse shift vector intersection points. The additional vertices would need to be inserted into any linear or polygon dependent datasets whose boundaries are coincident with cadastral boundaries for accurate adjustment results.

Further research would be required to determine the value of either of these processes and what algorithms could be used to best identify the cadastral boundaries and dependant dataset boundaries needing the treatment.

11.3.10 Truncated corners

In some cases, especially in urban areas, a street corner on a parcel in one cadastral layer has been truncated where, in the other, it has not. Thus, there are no corresponding corners to be matched. This problem was discussed in Section 9.10.1.

It is possible that further research could arrive at an automated method for detecting this situation and creating appropriate shift vectors.

11.3.11 Problems with rural road and creek boundaries

Section 9.10.3 discussed the problem that arises when a road or creek is digitised as two lines in one cadastre and one in the other. Section 7.11.3 also highlighted a problem that can arise where one bank of a creek has been matched to the correct bank in the other cadastre but the other bank remains unmatched. The latter problem was avoided in this research by arbitrarily excluding riparian boundaries with more than 100 points from being processed by the PMA1 algorithm.

Further research is needed to develop an algorithm to detect situations where one riparian boundary in the old cadastre is represented by two in the new cadastre of vice-versa.

11.3.12 ArcGIS *CheckGeometry* tool (Esri, 2016c)

Where shift vectors are incorrect it is possible that some parcels may self-intersect after adjustment. The ArcGIS *CheckGeometry* tool can detect these and several other geometry errors. A potential research project would be to discover whether this information could be used to aid the automatic detection and correction of bad shift vectors.

11.3.13 Dissimilar sources

In the absence of any suitable test data, it is not known whether the algorithms will prove useful when attempting to match cadastres digitised from dissimilar sources, for example, one captured from a paper map and the other from orthophotography, or where the two sources were captured with different precision. Further research would reveal whether any of the components of the ShiftGen solution would be of value in this case.

11.3.14 Deep learning

In recent years several notable successes have been achieved by the application of the AI process known as “Deep Learning”, a technique that attempts to mimic the activity of a human brain using neural networks, rather than the evolutionary model used in a Genetic Algorithm. Amongst others, the technique has been used extensively in image recognition (LeCun, Bengio, & Hinton, 2015), (Zhang, He, & Liu, 2017). In view of the considerable number of parcel, boundary, and point characteristics that needed to be considered whilst developing the algorithms designed here, it seems probable that a deep learning technique could deliver improved solutions to the point matching problems addressed in this thesis. The algorithms described herein, followed by manual error checking and correction, could potentially be employed to generate large numbers of correct shift vectors on several datasets. Shift-vector layers created in this way, along with the spatial attributes of the source point, could then potentially be used as training data for a deep learning AI.

11.4 Research limitations

This research has been concentrated on pairs of digital cadastres from different dates. Each pair has been created by one organisation, with similar precision. Therefore, it was possible to use polygon matching processes for the blocks and parcels that relied on that precision similarity. As mentioned in Section 11.3.13, it is unknown whether the algorithms will prove useful when attempting to match cadastres digitised from dissimilar sources, for example, one captured from a paper map and the other from orthophotography, or where the two sources were captured with different precision.

Given that the research has relied on knowledge of the nature of a digital cadastre, i.e. the existence of blocks enclosed by spaces representing roads, rivers, and coastlines, it is not expected that the solution would successfully match other types of polygon data, although some of the component algorithms may prove of value.

11.5 Summary

The most significant conclusion to be drawn from this research is that it is possible to spatially match the points from two cadastres without the need to provide the search distance parameters required by existing solutions. This conclusion has been drawn from the application of the solution to twelve diverse cadastral datasets comprising more than 82,000 parcels with more than 800,000 vertices. More than 97% of all the cadastral points have been matched to either a point or a location in the other cadastre even though the rural areas in the datasets exhibit a wide variety of different spatial problems that complicate the point matching process.

The thesis has described several innovations arising from the research, in particular: the value of block matching, where there are no existing UIDs to facilitate parcel matching; the use of apparent parcel movement to generate target points to predict the location of matching points; the use of boundary matching to allow old-cadastre points to be matched, even where there is no corresponding point in the new cadastre; and the use of boundary classification into riparian and rural-road boundary classes to improve point matching on those boundaries.

The algorithms described in this thesis have been documented in enough detail to allow future researchers to reproduce them in any suitable development environment. The insights regarding the value of block, parcel, and boundary matching, in guiding the point matching process, would be of value in any solution endeavouring to solve the same problem, whether based on the algorithms herein or not.

APPENDIX A: WEKA OUTPUT

A.1 WEKA results for LGA01

```
=== Run information ===

Scheme:weka.classifiers.trees.J48 -C 0.25 -M 2
Relation:     LGA01
Instances:    636
Attributes:   8
              DiffAngle
              DiffLength
              AreaRatio
              PerimRatio
              OverlapRatio
              Similarity
              DirectionDiff
              UIDsMatch
Test mode:10-fold cross-validation

=== Classifier model (full training set) ===

J48 pruned tree
------------------

OverlapRatio <= 17.944992: FALSE (147.0)
OverlapRatio > 17.944992: TRUE (489.0/14.0)

Number of Leaves  :   2

Size of the tree :    3


Time taken to build model: 0.01 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances         621               97.6415 %
Incorrectly Classified Instances        15                2.3585 %
Kappa statistic                          0.9356
Mean absolute error                      0.0444
Root mean squared error                  0.1516
Relative absolute error                 11.7159 %
Root relative squared error             34.8585 %
Total Number of Instances              636

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 1         0.093      0.969     1         0.984      0.933    TRUE
                 0.907     0          1         0.907     0.951      0.933    FALSE
Weighted Avg.    0.976     0.07       0.977     0.976     0.976      0.933

=== Confusion Matrix ===

   a   b   <-- classified as
 475   0 |   a = TRUE
  15 146 |   b = FALSE
```

A.2 WEKA results for LGA11

```
=== Run information ===

Scheme:weka.classifiers.trees.J48 -C 0.25 -M 2
Relation:     LGA11
Instances:    11915
Attributes:   8
              DiffAngle
              DiffLength
              AreaRatio
              PerimRatio
              OverlapRatio
              Similarity
              DirectionDiff
              UIDsMatch
Test mode:10-fold cross-validation

=== Classifier model (full training set) ===

J48 pruned tree
------------------

PerimRatio <= 93.822662
|   OverlapRatio <= 67.52425: FALSE (5089.0/62.0)
```

```
|   OverlapRatio > 67.52425
|   |   DirectionDiff <= 4.042458
|   |   |   OverlapRatio <= 91.975097: TRUE (24.0/1.0)
|   |   |   OverlapRatio > 91.975097: FALSE (3.0/1.0)
|   |   DirectionDiff > 4.042458: FALSE (9.0/1.0)
PerimRatio > 93.822662
|   DiffAngle <= 24.026594
|   |   OverlapRatio <= 1.713431
|   |   |   DiffLength <= 42.687523
|   |   |   |   DiffAngle <= 0.00216
|   |   |   |   |   PerimRatio <= 99.984453: FALSE (153.0/15.0)
|   |   |   |   |   PerimRatio > 99.984453
|   |   |   |   |   |   DirectionDiff <= 0.225751: TRUE (18.0/2.0)
|   |   |   |   |   |   DirectionDiff > 0.225751: FALSE (2.0)
|   |   |   |   DiffAngle > 0.00216
|   |   |   |   |   DiffAngle <= 3.461067: TRUE (83.0)
|   |   |   |   |   DiffAngle > 3.461067: FALSE (6.0/1.0)
|   |   |   DiffLength > 42.687523
|   |   |   |   PerimRatio <= 96.016602
|   |   |   |   |   DirectionDiff <= 0.26509: TRUE (2.0)
|   |   |   |   |   DirectionDiff > 0.26509: FALSE (8.0)
|   |   |   |   PerimRatio > 96.016602: FALSE (105.0)
|   |   OverlapRatio > 1.713431
|   |   |   DirectionDiff <= 1.928199
|   |   |   |   Similarity <= 0.021281
|   |   |   |   |   DiffAngle <= 17.577573: TRUE (5219.62/152.0)
|   |   |   |   |   DiffAngle > 17.577573
|   |   |   |   |   |   OverlapRatio <= 58.644363: FALSE (6.0/1.0)
|   |   |   |   |   |   OverlapRatio > 58.644363: TRUE (35.07)
|   |   |   |   Similarity > 0.021281
|   |   |   |   |   OverlapRatio <= 51.836697
|   |   |   |   |   |   DiffAngle <= 0.000005
|   |   |   |   |   |   |   AreaRatio <= 88.177871: TRUE (4.0/1.0)
|   |   |   |   |   |   |   AreaRatio > 88.177871: FALSE (18.0/1.0)
|   |   |   |   |   |   DiffAngle > 0.000005: TRUE (8.0/1.0)
|   |   |   |   |   OverlapRatio > 51.836697: TRUE (119.0/14.0)
|   |   |   DirectionDiff > 1.928199
|   |   |   |   OverlapRatio <= 67.437639
|   |   |   |   |   DiffAngle <= 0.138677: FALSE (61.0/8.0)
|   |   |   |   |   DiffAngle > 0.138677
|   |   |   |   |   |   DiffAngle <= 10.685587: TRUE (6.0)
|   |   |   |   |   |   DiffAngle > 10.685587: FALSE (3.0)
|   |   |   |   OverlapRatio > 67.437639
|   |   |   |   |   DirectionDiff <= 5.830997: TRUE (38.0/4.0)
|   |   |   |   |   DirectionDiff > 5.830997: FALSE (2.0)
|   DiffAngle > 24.026594
|   |   OverlapRatio <= 84.651544: FALSE (821.0/7.0)
|   |   OverlapRatio > 84.651544
|   |   |   DiffLength <= 65.886515: TRUE (38.0)
|   |   |   DiffLength > 65.886515
|   |   |   |   DiffLength <= 94.926673
|   |   |   |   |   OverlapRatio <= 92.59809: FALSE (8.0/1.0)
|   |   |   |   |   OverlapRatio > 92.59809: TRUE (11.0/3.0)
|   |   |   |   DiffLength > 94.926673: TRUE (15.32)

Number of Leaves  :    29

Size of the tree :     57


Time taken to build model: 0.33 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances       11590               97.2723 %
Incorrectly Classified Instances       325                2.7277 %
Kappa statistic                          0.9452
Mean absolute error                      0.0453
Root mean squared error                  0.1592
Relative absolute error                  9.103  %
Root relative squared error             31.9222 %
Total Number of Instances            11915

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 0.977     0.031      0.965     0.977     0.971      0.979    TRUE
                 0.969     0.023      0.98      0.969     0.974      0.979    FALSE
Weighted Avg.    0.973     0.027      0.973     0.973     0.973      0.979

=== Confusion Matrix ===

    a    b   <-- classified as
 5412  129 |    a = TRUE
  196 6178 |    b = FALSE
```

APPENDIX B: ABBREVIATIONS AND PARAMETERS

AGDyy	Australian Geodetic Datum – year yy
BA	Bisector Azimuth
DA	Difficult Area
CI	Circularity Index. Produces the value 1 for a perfect circle and increasingly smaller values for more irregular shapes.
PMAn	Point Matching Algorithm n
GA	Genetic Algorithm
GDAyy	Geocentric Datum of Australia – year yy
FID	Feature Identity Number
ID	Intersection Distance
IMn	Intersect Method n
LGA	Local Government Authority
m	Metres
MAD	Median Absolute Deviation (robust statistics)
maxSD	The maximum Search Distance in metres for a dataset
MGA	Map Grid of Australia – a metre-based coordinate system
PSA	Parcel Shift Azimuth
PSL	Parcel Shift-vector Length
RTE	Run Time Efficiency
SA	Segment Angle. The acute angle between two line-segments either side of vertex.
UID	Unique Identification Number
TA	The clockwise Turning Angle between two line segments.
V1:	Vector Product Format

APPENDIX C: PYTHON SCRIPT FOR SPATIAL ADJUSTMENT

This script is a slightly modified version of one from the ArcGIS GenerateRubbersheetLinks documentation. It was used to carry out the spatial adjustment of all the old cadastre datasets, using ArcGIS tools only, in order to compare the adjustment results with those resulting from shift vectors output by the ShiftGen program. The comments are all from the ArcGIS documentation.

```
import sys
import arcpy
from arcpy import env
arcpy.env.overwriteOutput = True
script, workspace, search_distance = sys.argv
arcpy.env.workspace =workspace

# Set local variables
oldCadastre = "OrigOldCad.shp"
newCadastre = "OrigNewCad.shp
sourceFeatures = "AGOldCadLines.shp"
targetFeatures = "AGNewCadLines.shp"
sourceFeatures_Copy = "AGOldCad_Adj.shp"
grlOutput = "AGShifts.shp"
grlOutputPts = "AGShifts_Pnt.shp"

match_fields = ""

qaLocations = "qa_locations.shp"

##Make the line layers
arcpy.PolygonToLine_management(oldCadastre,sourceFeatures,"IDENTIFY_NEIGHBORS")
arcpy.PolygonToLine_management(newCadastre,targetFeatures,"IDENTIFY_NEIGHBORS")

# Generate rubbersheet links
arcpy.GenerateRubbersheetLinks_edit(sourceFeatures, targetFeatures, grlOutput,
search_distance, match_fields)

# ============================================================================
# Note 1:  The result of GenerateRubbersheetLinks may contain errors; see
#          tool reference. Inspection and editing may be necessary to ensure
#          correct links before using them for rubbersheeting.

#          One of the common errors are intersecting or touching links.
#          Their locations can be found by the process below.
# =============================================================================

# Find locations where links intersect or touch; the result contains
# coincident points
# arcpy.Intersect_analysis(grlOutput, qaLocations, "", "", "POINT")

# Delete coincident points
# arcpy.DeleteIdentical_management(qaLocations, "Shape")

# ============================================================================
# Note 2:  At this point you can manually inspect locations in qaLocations;
#          delete or modify links as needed.
# ============================================================================

# Make a copy of the sourceFeatures for rubbersheeting
arcpy.CopyFeatures_management(origOldCad, sourceFeatures_Copy)

# Use the links for rubbersheeting
arcpy.RubbersheetFeatures_edit(sourceFeatures_Copy, grlOutput, grlOutputPts,
"LINEAR")
```

APPENDIX D: THE GENETIC ALGORITHM RESEARCH

The initial approach adopted for this research was to use third-party software, the Spatial Tapestry Workbench (Spatial Tapestry, n.d.), to carry out the complete spatial adjustment process (see Section 1.1) using a Genetic Algorithm (GA) to automatically generate the multiple sets of five input parameters required by Workbench and thereby reduce operator intervention and required skill level.

D.1 Overview

This appendix describes the research undertaken to develop the genetic algorithm; the insights gained during this process are discussed, as are the reasons why the GA approach was eventually superseded.

D.2 Genetic algorithm background

Evolutionary programming techniques were first invented in the 1950s and 60s and the computational technique now known as a genetic algorithm (GA) was invented by J H Holland, also in the 1960s (Mitchell, 1996). The literature search conducted at the start of this research found no examples of the use of Genetic Algorithms for the specific purpose of spatial adjustment but more recently Ruiz-Lendínez et al. (2017) have published an article on the use of a genetic algorithm for polygon matching. Because a GA was used in the early stages of this research and later superseded for the reasons described in Chapter 4, some background information is provided here.

Genetic algorithms are just one type of method proposed to search for optimal solutions in which an error function is minimised. Other methods can be used when the error function is an equation that can be differentiated with respect to the parameters of interest and the minimum determined directly – the closed form solution. If this is not possible, methods such as the well-known Newton-Raphson method (Kelley, 2003) and its derivatives can be used to iteratively converge on a solution if the error function can be differentiated. However, these require an error function that has one minimum that can be easily reached by gradient descent. If an error function is not differentiable, iterative methods such as Powell's method (Powell, 1964) can be used. These methods sample the error space to determine the direction to descend. Powell's method also has methods to decide how far to jump at each iteration to get to the minimum quickly without overshooting. A problem with Powell's method is that it is computationally expensive. If the error space has many minima (local minima) then simulated annealing can be used (Khachaturyan, Semenovskaya, & Vainshtein, 1979). This method can jump over local minima by using large jumps across error space in early iterations and which get smaller for later iterations. The method simulates how annealing is used for materials such as metals and glass in which local stresses (like local minima) are removed by successive heating and cooling in a controlled manner. Simulated annealing has been used to aid the map generalisation process (Ai et al., 2013).

A genetic algorithm is one that attempts to emulate the natural process of evolution in a computer (Konfrst, 2004). The processes of breeding and mutation in nature have resulted in the extraordinary diversity of life on Earth today. Each life form is uniquely fitted to life in its own environment and computer scientists aimed to emulate this process to arrive at a good, though not necessarily optimal, solution to computationally complex optimisation problems. GAs have been used in many different fields including electronic circuit board design, image processing, financial market prediction, the travelling salesman problem and many others.

The principal concepts embodied in a simple GA are:

(a) Genes. Genes are bit strings representing the value of a variable whose value needs to be optimised.

(b) Chromosomes. Chromosomes are comprised of a set of one or more genes.

(c) Populations. A population is a set of chromosomes.

(d) An objective (fitness) function. The objective function is, typically, a piece of computer code to evaluate the fitness of each chromosome for its designated purpose although sometimes a human operator may be required to perform this function.

(e) Generations. Generations are the sets of populations produced by the algorithm after each round of fitness evaluation, breeding, and mutation. Each generation replaces the previous one after each round.

Genetic algorithms are particularly suitable for the solution of optimisation problems where several factors are involved. They have been used in a number of GIS-based applications, for example, dangerous goods routeing (Li, Leung, Huang, & Lin, 2013), sustainable land use optimisation (Cao, Huang, Wang, & Lin, 2012) and road matching for vehicle location (Velaga et al., 2012). Dangerous goods routeing involves many factors such as public safety, transportation costs, risks to the environment, and route suitability, all of which must be considered when determining an optimal route. The multiple factors make this an appropriate application of a genetic algorithm. Similarly, making decisions regarding optimisation of sustainable land use also involves multiple factors including geology, landform, accessibility, and human factors and several others again suggesting that the use of a GA is a suitable approach to the problem addressed by this research.

In the case of the use of a GA for road matching for vehicle location described by Velaga et al. (2012), the GA was used to improve the matching algorithm by optimising the values of several parameters required by the algorithm, i.e. a similar use to that proposed in the initial approach to this research described in this Appendix.

D.3 Rationale for the GA/Workbench approach

The methodology initially proposed to accomplish the goal of reducing the required skill level of the GIS operator was the exploration of a genetic algorithm to determine an optimised set of the five parameter values needed for input to a specific commercial software package, the Spatial Tapestry Workbench. Workbench itself was developed as a simplified user interface to a Spatial Tapestry package called Spatial Adjustment Manager (SAM) which is in regular use by several utility companies in the eastern states of Australia to bring their cadastrally dependant datasets into line with an upgraded cadastre. Workbench implements both stages of the complete spatial adjustment process (see Section 1.1), i.e. it carries out both the point matching component and the spatial adjustment component.

The Workbench software was chosen for this project as being one of the more sophisticated products on the market in its approach to the spatial adjustment process and for the fact that it is a tool specifically designed to address the spatial adjustment of cadastrally dependant layers. This package requires a great deal of skill and a trial-and-error process to arrive at a set of good parameter values for generating shift vectors (Merritt, 2009). The challenge was to develop an automated method for arriving at an optimal set of parameter values that would result in a reduction in the amount of skilled human input to the process.

To achieve the stated aim of this research, i.e. to reduce the human input required to carry out the complete spatial adjustment processes, a genetic algorithm was developed to drive the Workbench software. Workbench requires the user to input several parameters defining limits of search distances and angle tolerances (see Section D.4); these are used to control a point matching and shift generation process and ultimately the spatial adjustment of any spatially dependent datasets. The Workbench software uses an iterative process to match the cadastral points before carrying out spatial adjustment on the spatially dependent datasets. Each iteration requires a new set of values for the parameters. The problem is one of optimising the values of the parameters for each iteration and, ultimately, for the complete process. In effect, the operator, using a trial-and-error process, "evolves" a solution to the problem thereby suggesting that a GA approach to automating the process would be appropriate.

The GA approach to optimising the parameter values for Workbench was chosen because there is no simple equation available that can be used as an "error function" (see Section D.2) that could be used to solve the problem using a simulated annealing or the Newton-Raphson method. The number of shift-vectors created by each call to the Workbench was used as the value of the objective function for the GA. In summary, the aim of the GA program was to arrive at a set of parameter values that maximised the number of shift vectors created by the Workbench.

D.4 The Workbench software

The Workbench software requires the input of five parameter values to control the creation of shift vectors. The required parameters are:

(a) the maximum distance between parcel centroids,

(b) the maximum difference in length between parcel boundary lines,

(c) the maximum difference in angle between boundary bearings,

(d) the maximum distance between nodes,

(e) Another distance parameter used to cull intersections that are too close together for reliably matching.

Before shift vectors can be created, both parcels and the parcel boundaries need to be correctly matched by Workbench's internal algorithms. The values of the parameters are needed to constrain the process of searching for a correct match for the parcels and their boundaries, for example,, in the case of the parcel centroid distance parameter, a parcel in the new cadastre would only be considered as a possible match if its centroid lies within this distance from the centroid of a parcel in the old cadastre and a parcel boundary would only be considered as a possible match if the differences between the boundary bearings and boundary lengths fall within the parameter values. In effect, the parameter values reduce the search space and reduce the possibility of incorrectly matching similarly or identically shaped parcels.

Workbench attempts to iteratively improve the number of correct point matches. Each iteration requires a new set of the five parameter values.

The experience of running the Workbench software in standalone mode on one small dataset (LGA01) facilitated an understanding of the range of values that would be required as input, albeit only for that specific dataset, of course.

D.5 Outline of the GA research stages

Initially, an understanding of the Workbench program was obtained by operating it on an LGA dataset where appropriate parameter values (supplied by the vendor – Spatial Tapestry) were available. The Workbench software can be used manually through a GUI interface or can operate on a script file.

Next, a basic GA program was developed to generate script files from the evolving sets of parameter values; each generated script file was then input to the Workbench and the number of shift vectors generated was used as the value for the GA objective function. At this stage, fixed upper parameter value limits for the GA to explore were used. Fixed upper values for parameter values, however, are not desirable. If a call to the Workbench uses an overlarge parameter value for one of the search distances, this can result in numerous incorrect point matches because of the large area that can be explored for potential matches. If this happens the GA would be more likely to select the current chromosome for breeding because the number of shift vectors created (the objective function result) would be large. On the other hand, a maximum parameter value that is set too small may result in many correct matches being missed and an optimal result may not evolve. Further research was, therefore, undertaken to discover whether specific upper parameter value limits for any given dataset could be computed by analysis of the dataset. An analysis of the available LGA datasets, described in Chapter 3, was undertaken as part of this research stage.

Lastly, a polygon matching process was researched to supply matching UIDs to the Workbench software for those datasets lacking existing UIDs.

D.6 The genetic algorithm

The GA developed for this research implemented the concepts of genes (each gene represents a Workbench parameter value), chromosomes (a set of the five parameter values), generations (successive populations of chromosomes), mutation (a random change in a gene value) and selective breeding (selecting pairs of chromosomes to crossover and keep in the next generation). These concepts were described in Section D.2 .

The following is a brief outline of the processes developed by the author to operate the GA for this project:

(a) Generate a set of chromosomes with random values (within a fixed range of values) for each of the five genes. Each gene holds a value for one of the five parameters. Initially, the maximum value for each gene was arbitrarily assigned; the minimum value is always zero.

(b) Evaluate the fitness of each chromosome by inputting its set of gene values to the objective function, i.e. execute the Workbench software with the script containing the parameter values from the genes. Obtain the number of shift vectors that were generated using these inputs; the greater the number of shift vectors, the "fitter" the chromosome.

(c) Select a pair of chromosomes for breeding by roulette wheel selection (Ribeiro Filho, Treleaven, & Alippi, 1994). The roulette wheel algorithm is weighted to preferentially select from the fitter parents.

(d) Apply random mutations to the genes using a low probability of mutation and constraining the new value to one in the predefined parameter value range.

(e) Using a low probability of crossover, cross the selected pair of chromosomes at a randomly selected point between the genes. If it occurs, this crossover results in parameter values from one parent being exchanged with the corresponding parameter value from the other parent. The table below illustrates the result of the crossover process; in this case, the values of parameters (genes) four and five have been exchanged (the numbers in red).

Table D.1 Chromosome crossover

Gene number	Chromosome 1 gene values before crossover	Chromosome 2 gene values before crossover	Chromosome 1 gene values after crossover	Chromosome 2 gene values after crossover
1	8	3	8	3
2	4	4	4	4
3	2	5	2	5
4	9	7	7	9
5	8	2	2	8

(f) Breed each selected and possibly mutated and/or crossed pair to produce two new chromosomes.

(g) Repeat (c) to (f) above until a new generation of the same size as the parent generation has been created.

Repeat (b) to (f) above for a fixed number of generations (typically 50 or more).

Figure D.1 outlines these processes. The values of all the constants required by the GA could be changed between trial runs.

Typically, the GA was executed using: population size, 20 (P); number of parameter-set iterations, 3 (I); number of generations, 50 (G); crossover probability, 0.1 (pC); mutation probability, 0.02 (pM). In the early stage of the GA research, it was envisaged that further research would be conducted to experiment with the values of the mutation and crossover probabilities and other input constants, investigate different breeding selection algorithms, and investigate the possibility of terminating the breeding process once no significant improvement in the results occurred. Few of these intentions were ever actioned for the reasons described in Section 4.2.

"Fitness" was evaluated by comparing the number of shift vectors generated by each call to the objective function (Workbench) to the total number of points in the dataset. A perfect solution would result in a correct shift vector for each point from the old cadastre. (Shift vectors can be of zero length where no adjustment is needed, i.e. they collapse to identity points.) In effect, the GA is attempting to minimise the number of unmatched points.

Using the number of shift vectors generated by each Workbench run as the method for choosing the "best" or fittest chromosome was adopted for pragmatic reasons. More is not necessarily better. Ideally, the fittest chromosome would be determined by inspecting the result of spatially adjusting the old cadastre using the generated shift vectors as input to the adjustment process. In a genetic algorithm driven process, this is clearly impractical. The objective function may need to be executed many thousands of times before an optimal solution is converged upon and each adjustment run, and subsequent inspection would take, at the very least, many minutes and much longer for large datasets. In any case, this approach would result in a subjective evaluation of the objective function results whereas the shift vector count is purely objective.

The Figure D.1 below illustrates the steps involved in the GA solution.

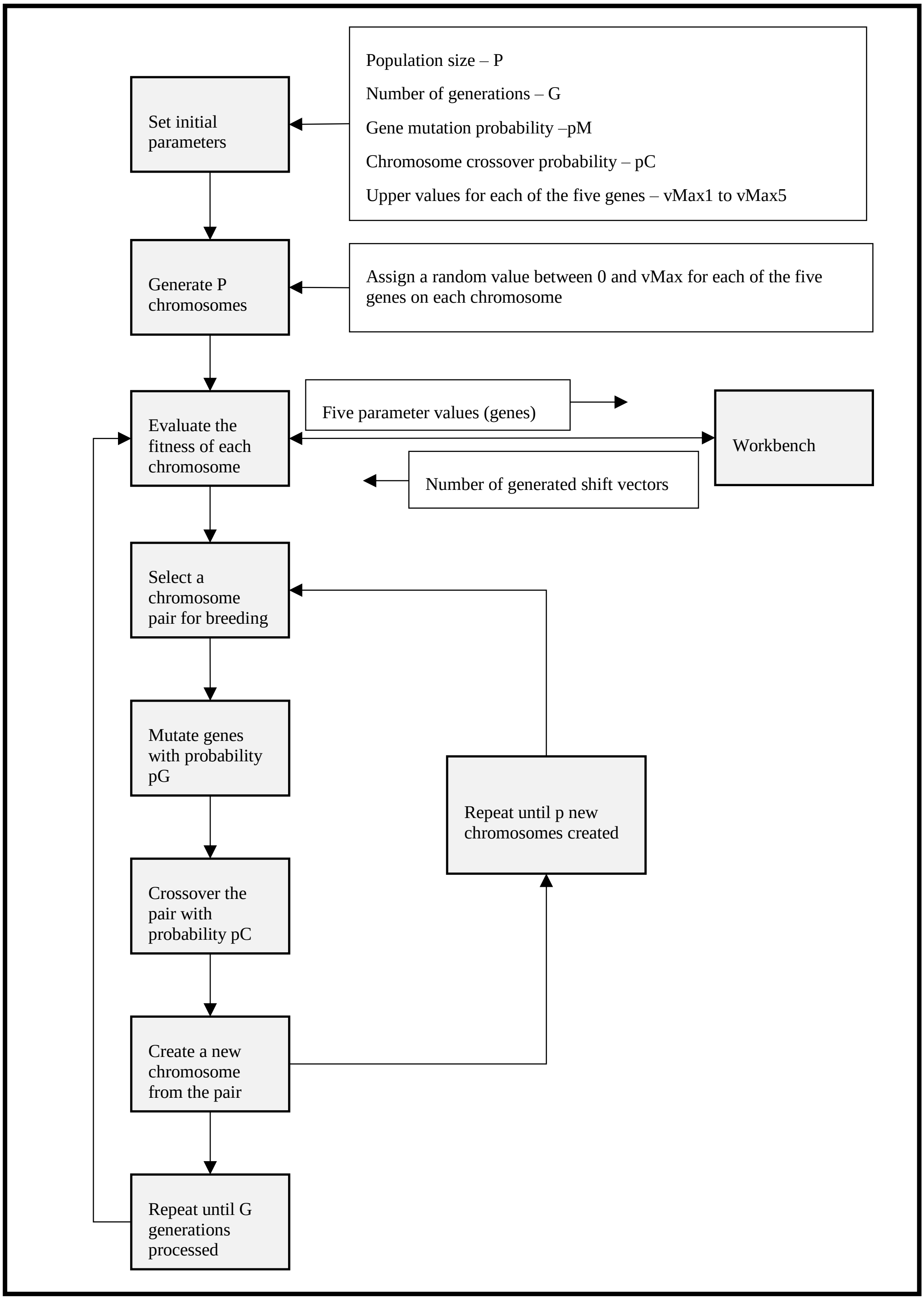

Figure D.1 The GA process

The genetic algorithm was implemented as outlined above in prototype form and tested initially on the single LGA dataset where sample results were available for comparison. At the end of each complete GA run, the old cadastre was adjusted using the Workbench generated shift vectors. The adjusted old cadastre was then compared with the new cadastre to evaluate the quality of the results.

Once the GA was delivering satisfactory results for the small (mainly urban) LGA dataset, it was then tested on several more datasets. As a result of these tests, it became apparent that fixed value ranges for the parameters would not be appropriate for all possible datasets. Inspection of the mapped cadastral datasets revealed that appropriate parcel centroid search distance limits for the GA to explore would need to be very much larger in rural areas than in urban areas.

Since the aim of the GA was to reduce user input and fixed value ranges for parameter values are not appropriate, it now became necessary to discover if there was any way to determine what the maximum value should be for any given parameter. Further analysis of the available datasets was therefore undertaken.

It had already become apparent that GA processing times were likely to be in the order of days rather than hours for each complete run of the GA on a large dataset, even with the use of parallel processing wherever possible. (The GA was implemented on an 8-core processor PC running Windows 7; eight instances of the Workbench objective function were launched at a time to achieve parallel processing.)

Even on a smallest LGA dataset with fewer than 700 parcels, each call to the Workbench was taking approximately two seconds so that a complete run of the GA for that dataset took more than 1.5 hours to run. For the larger datasets with thousands of parcels, a complete run would take many hours or even days. On one of the larger datasets with more than 12,000 parcels, calls to the Workbench were each taking about three minutes meaning that a complete run of three parameter sets, a population size of 20 and 50 generations could take more than six days to run on the available hardware.

To overcome these problems and enable the research to proceed at an acceptable rate, the prototype was modified to classify the parcels into size ranges and to extract a maximum of 500 parcels from each subset on which to execute the GA. The maximum of 500 was chosen because experiments had shown that each objective function value for this number would be obtained in just a few seconds. The final sets of parameter values derived from executing the GA on the subsets would then be used to carry out the adjustment on all the parcels in a size group and the separate area group layers finally being amalgamated into a single dataset again. This would obviously be a less than ideal solution as there could be no guarantee that the 500 extracted records would be a representative subset of the complete parcel size group so that the resulting parameter values may not be appropriate for the entire size group.

Figure D.2 illustrates the processes proposed for the application of the GA to rural and mixed urban and rural areas.

The GA was modified to automatically determine whether subdividing the dataset was necessary, i.e. that the dataset was mainly rural, and to automatically create the 500-parcel subgroup datasets if so. This implementation was then tested using fixed upper-value limits for the parameters. Once this implementation was operational, it then became necessary to discover whether parameter value maxima for each size group could be determined automatically.

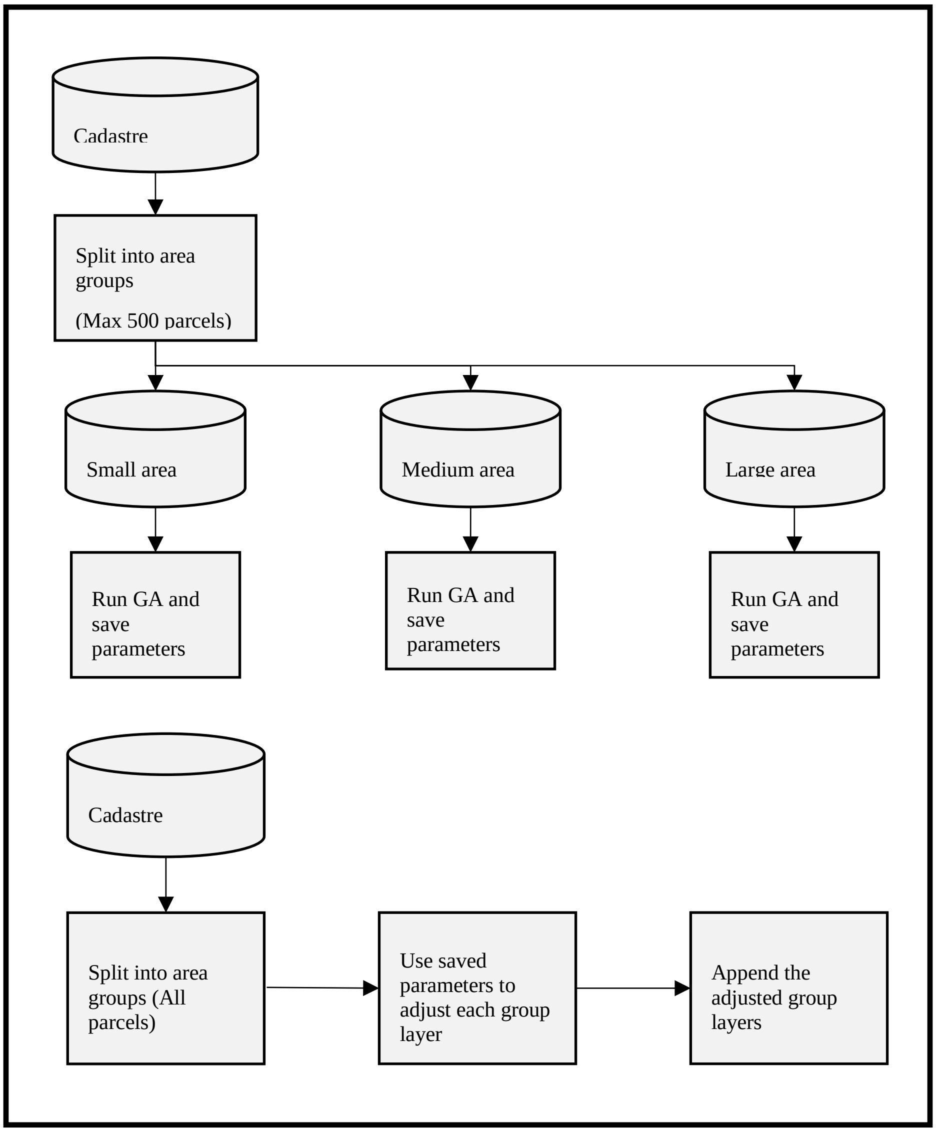

Figure D.2 Proposed GA process.

D.7 Gene (parameter) value ranges

The prototype version of the software initially developed used a fixed range of values to be explored by the random gene mutations. For reasons already mentioned, it was determined that it would not be appropriate to simply use large values for the upper limits of parameter values. When generating shift vectors with over-large parameter values, the result can be a spuriously large number of shift vectors but, as mentioned in Section D.5 , quantity does not always equal quality. The datasets were, therefore, examined to establish whether it would be possible to automatically generate upper values for at least the some of the required parameters. At this stage, no obvious approaches suggested themselves for automatic determination of maxima for all the parameter values but it seemed likely that a preliminary parcel matching process would allow a maximum search distance for parcel centroids to be determined. Since the

Workbench software also works more efficiently if there are matching UIDs, research, therefore, commenced on matching the parcels.

D.8 Assigning parcel UIDs

The Workbench software employed to generate the shift vectors was known to produce more satisfactory results if the corresponding parcels in the old and new cadastre have a matching unique identifier attribute (Merritt, 2009). Such an identifier can obviate the need for the Workbench to perform an initial parcel matching process and thus reduce the amount of time need to evaluate the objective function. Research attention was therefore refocused on developing a parcel-matching algorithm to generate matching UIDs where none were available and, at the same time, calculate a suitable maximum value for parcel centroid search distance. The parcel matching research is described in detail in Chapter 4 which also details the reasons why the GA research was discontinued.

Up to the point where research on the GA was terminated, the developed code was working satisfactorily using fixed parameter value ranges and was delivering as good, or better results than the those on the one small dataset where existing results were available for comparison, i.e. the number of shift vectors created using the final five-parameter sets was greater than the number generated using a supposedly optimal parameter value set that had been supplied with the dataset.

The research was discontinued for the following reasons:

(a) No one set of generated parameter values would be appropriate for all areas of a mixed urban and rural dataset (this problem is illustrated in Section 9.4.5).

(b) No one set of upper limits for parameter values would be appropriate for all data in a mixed urban and rural dataset.

(c) Establishing parameter value maxima using 500-parcel subsets would not necessarily result in good maxima for all the parcels in the same area-size group.

(d) Subdividing the dataset into separate datasets for urban and rural data would introduce edge-matching problems.

(e) Execution times would be unacceptably long for large datasets, even with the use of parallel processing.

Most important of all was the realisation that control points could be located by using the apparent movement of matched parcels meaning that spatial adjustment problem could be solved without the use of Workbench.

REFERENCES

Abduhl Rahman, A., Van Oosterom, P. J. M., Hua, T. C., Sharkawi, K. H., Duncan, E. E., Azri, N., & Hassan, M. I. (2012). *3D modelling for multipurpose cadastre*. Paper presented at the 3rd International FIG Workshop on 3D Cadastres: Developments and practices, Shenzhen, China, . https://repository.tudelft.nl/islandora/object/uuid:47f48db7-93e5-4528-9854-9b77aa6dc2fa?collection=research

Ai, T., Cheng, X., Liu, P., & Yang, M. (2013). A shape analysis and template matching of building features by the Fourier transform method. *Computers, Environment and Urban Systems, 41*(C), 219-233. doi:10.1016/j.compenvurbsys.2013.07.002

Akkaya Aslan, S. T., Gundogdu, K. S., & Arici, I. (2007). Some Metric Indices for the Assessment of Land Consolidation Projects. *Pakistan Journal of Biological Sciences, 10*(9), 1390-1397. doi:10.3923/pjbs.2007.1390.1397

ANZLIC Committee on Surveying and Mapping - ICSM. (2018). Cadastral maps and plans. Retrieved from http://www.icsm.gov.au/education/fundamentals-mapping/types-maps/cadastral-maps-and-plans

Arge, L., Hinrichs, K. H., Vahrenhold, J., & Vitter, J. S. (2002). Efficient Bulk Operations on Dynamic R-Trees. *An International Journal in Computer Science, 33*(1), 104-128. doi:10.1007/s00453-001-0107-6

Arkin, E. M., Chew, L. P., Huttenlocher, D. P., Kedem, K., & Mitchell, J. S. (1991). An Efficiently Computable Metric for Comparing Polygonal Shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 13*(3), 209-216.

Astrahan, M. M., Mehl, J. W., Putzolu, G. R., Traiger, I. L., Wade, B. W., Watson, V., . . . McJones, P. R. (1976). System R: relational approach to database management. *ACM Transactions on Database Systems, 1*(2), 97-137. doi:10.1145/320455.320457

Banič, I., & Taranenko, A. (2017). Measuring Closeness of Graphs—The Hausdorff Distance. *Bulletin of the Malaysian Mathematical Sciences Society, 40*(1), 75-95. doi:10.1007/s40840-015-0259-1

Blasby, D., Davis, M., Kim, D., & Ramsey, P. (2003). GIS Conflation using Open Source Tools. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.195.9292&rep=rep1&type=pdf

Borges, K. A. V., Davis, C. A., & Laender, A. H. F. (2001). OMT-G: An Object-Oriented Data Model for Geographic Applications. *GeoInformatica, 5*(3), 221-260. doi:10.1023/a:1011482030093

Cao, K., Huang, B., Wang, S., & Lin, H. (2012). Sustainable land use optimization using Boundary-based Fast Genetic Algorithm. *Computers, Environment and Urban Systems, 36*(3), 257-269. doi:10.1016/j.compenvurbsys.2011.08.001

Carstensen, W. (1989). Fractal self-similarity in cartographic generalization. *The American Cartographer, 16*(3), 181-189.

Chávez, E., Chávez-Cáliz, A., & López-López, J. (2013). Polygon Matching and Indexing Under Affine Transformations. *arXiv.org*. Retrieved from https://arxiv.org/abs/1304.4994.

Chen, W., Peng, J., Hong, H., Shahabi, H., Pradhan, B., Liu, J., . . . Duan, Z. (2018). Landslide susceptibility modelling using GIS-based machine learning techniques for Chongren County, Jiangxi Province, China. *Science of the Total Environment, 626*, 1121-1135. doi:10.1016/j.scitotenv.2018.01.124

Civica. (n.d.). GIS Tools. Retrieved from https://www.civica.co.uk/gis/utilities

Cobb, M. A., Chung, M. J., Foley, H. III, Petry, F. E., Shaw, K. B., & Miller, H. V. (1998). A Rule-based Approach for the Conflation of Attributed Vector Data. *GeoInformatica, 2*(1), 7-35. doi:10.1023/a:1009788905049

Deakin, R. E., Bird, S. C., & Grenfell, R. I. (2002). The Centroid? Where would you like it to be be? *Cartography, 31*(2), 153-167. doi:10.1080/00690805.2002.9714213

Deepak, P. (2013). *Distributed Bioenergy Systems For Expanding Rural Electricity Access In Tumkur District, India : A Feasibility Assessment Using GIS, Heuristics And Simulation Models.* Paper presented at the The International Society for Optical Engineering 8795.

Demetriou, D., Stillwell, J., & See, L. (2013). A GIS-based shape index for land parcels. Retrieved from http://proceedings.spiedigitallibrary.org

Department of Defense. (1996). Interface Standard for Vector Product Format. Retrieved from http://earth-info.nga.mil/publications/specs/printed/2407/2407_VPF.pdf

Devlin, M. (2009). [Personal communication].

Dueker, K. J. (1992). Dynamic segmentation revisited: A milepoint linear data model. Retrieved from https://www.researchgate.net/profile/Kenneth_Dueker/publication/268200279_Dynamic_segmentation_revisited_a_milepoint_linear_data_model/links/5509d9340cf20ed529e2276d/Dynamic-segmentation-revisited-a-milepoint-linear-data-model.pdf

Effenberg, W. (2001). *Spatial cadastral information systems : the maintenance of digital cadastral maps.* (Ph.D diss.), University of Melbourne, Retrieved from http://csdila.unimelb.edu.au/publication/theses/Wolfgang_PhD_thesis.pdf

Encyclopedia of Mathematics. (n.d.). Affine transformation. Retrieved from https://www.encyclopediaofmath.org/index.php/Affine_transformation

Envitia. (n.d.). MapRite for PAI. Retrieved from http://www.envitia.com/maprite/maprite-for-pai/

Esri. (1996). Arc/INFO Extensions. Retrieved from http://downloads.esri.com/support/whitepapers/ao_/aiexten.pdf

Esri. (1998). Esri Shapefile Technical Description. Retrieved from https://www.esri.com/library/whitepapers/pdfs/shapefile.pdf

Esri. (2008). What is a coverage? Retrieved from http://webhelp.esri.com/arcgisdesktop/9.3/index.cfm?topicname=what_is_a_coverage

Esri. (2009). About working with links. Retrieved from http://webhelp.esri.com/arcgisdesktop/9.3/index.cfm?TopicName=Working_with_links

Esri. (2016a). About control points. Retrieved from http://desktop.arcgis.com/en/arcmap/10.3/manage-data/editing-parcels/aboutcontrolpoints.htm

Esri. (2016b). About spatial adjustment rubbersheeting. Retrieved from http://desktop.arcgis.com/en/arcmap/10.3/manage-data/editing-existing-features/about-spatial-adjustment-rubbersheeting.htm

Esri. (2016c). Check Geometry. Retrieved from http://desktop.arcgis.com/en/arcmap/10.3/tools/data-management-toolbox/check-geometry.htm

Esri. (2016d). Dissolve. Retrieved from http://desktop.arcgis.com/en/arcmap/10.3/tools/coverage-toolbox/dissolve.htm

Esri. (2016e). How To: Perform rubber sheeting in Arcedit. Retrieved from https://support.esri.com/en/technical-article/000001316

Esri. (2016f). What is a coverage? Retrieved from http://desktop.arcgis.com/en/arcmap/10.3/manage-data/coverages/what-is-a-coverage.htm

Esri. (2016g). What is linear referencing? Retrieved from http://desktop.arcgis.com/en/arcmap/10.3/guide-books/linear-referencing/what-is-linear-referencing.htm

Esri. (2017a). Generate Rubbersheet Links. Retrieved from http://desktop.arcgis.com/en/arcmap/10.5/tools/editing-toolbox/generate-rubbersheet-links.htm

Esri. (2017b). Polygon To Line. Retrieved from http://desktop.arcgis.com/en/arcmap/10.5/tools/data-management-toolbox/polygon-to-line.htm

Esri. (2018). ArcGIS Desktop. Retrieved from http://desktop.arcgis.com/en/

Esri. (n.d.-a). GIS Dictionary - densify. Retrieved from https://support.esri.com/en/other-resources/gis-dictionary/term/9d1ec93d-7cba-4255-bf01-d767d11fc017

Esri. (n.d.-b). GIS Dictionary - edge. Retrieved from https://support.esri.com/en/other-resources/gis-dictionary/term/33017156-67a6-4024-acd5-ed7837eb7286

Esri. (n.d.-c). What is ArcSDE? Retrieved from http://edndoc.esri.com/arcobjects/9.2/NET_Server_Doc/manager/geodatabase/administering_a-557706548/what_is_arcsde_qst_.htm

Fanos, A., Pradhan, B., Mansor, S., Yusoff, Z., & Abdullah, A. (2018). A hybrid model using machine learning methods and GIS for potential rockfall source identification from airborne laser scanning data. *Journal of the International Consortium on Landslides*, *15*(9), 1833-1850. doi:10.1007/s10346-018-0990-4

Fletcher, D. (n.d.). Modelling GIS Transportation Networks. Retrieved from http://webapp1.dlib.indiana.edu/virtual_disk_library/index.cgi/1283872/FID631/MAIN/LIBRARY/CONF/FLETCHER.PDF

Gakstatter, E. (2009). TIGER Database Historical Perspective. Retrieved from http://geospatial-solutions.com/tiger-database-historical-perspective/

Gargantini, I. (1982). An effective way to represent quadtrees. *Communications of the ACM*, *25*(12), 905-910. doi:10.1145/358728.358741

Geoscience Australia. (n.d.-a). Datum Modernisation in Australia. Retrieved from http://www.ga.gov.au/scientific-topics/positioning-navigation/datum-modernisation

Geoscience Australia. (n.d.-b). GDA94 – GDA2020 Online Transformation Service. Retrieved from http://positioning.fsdf.org.au/

Geoscience Australia. (n.d.-c). The Map Grid of Australia 1994 Computational Manual. Retrieved from https://www.propertyandlandtitles.vic.gov.au/__data/assets/pdf_file/0023/55760/The_Map_Grid_of_Australia_1994_Computational_Manual.pdf

Gielsdorf, F., Gruendig, L., & Aschoff, B. (2004). Positional Accuracy Improvement - A Necessary Tool for Updating and Integration of GIS Data. Retrieved from http://www.fig.net/pub/athens/papers/ts02/ts02_2_gielsdorf_et_al.pdf

Gillman, D. W. (1985). *Triangulations for Rubber-Sheeting.* Paper presented at the Auto Carto 7 Conference, Washington.

GISGeography. (2018). Latitude, Longitude and Coordinate System Grids. Retrieved from https://gisgeography.com/latitude-longitude-coordinates/

GISGeography. (n.d.-a). Georeferencing: Rubbersheet with Control Points. Retrieved from https://gisgeography.com/georeferencing/

GISGeography. (n.d.-b). GIS Dictionary. Retrieved from https://gisgeography.com/gis-dictionary-definition-glossary/

GitHub, I. (2018). Least squares adjustment software. Retrieved from https://github.com/icsm-au/DynAdjust

González-Matesanz, F. J., & Malpica, J. A. (2006). Quasi-conformal mapping with genetic algorithms applied to coordinate transformations. *Computers & Geosciences*, *32*(9), 1432-1441. doi:10.1016/j.cageo.2006.01.002

Gonzalez, X. P., Alvarez, C. J., & Crecente, R. (2004). Evaluation of land distributions with joint regard to plot size and shape. *Agricultural Systems*, *82*(1), 31-43. doi:10.1016/j.agsy.2003.10.009

Goodchild, M. F. (2018). GIScience for a driverless age. *International Journal of Geographical Information Science*, *32*(5), 849-855. doi:10.1080/13658816.2018.1440397

Gray, K. J., & Gray, S. F. (2009). *Elements of Land Law*. London: Butterworths.

Guo, L., Lv, Z., Wang, H., Zhang, B., & Cui, T. (2010, June 2010). *Matching algorithms and tactics on vector area data based on spatial directional similarity*. Paper presented at the 18th International Conference on Geoinformatics, Beijing, China.

Guttman, A. (1984). R-trees. *ACM SIGMOD Record, 14*(2), 47. doi:10.1145/971697.602266

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., & Witten, I. H. (2009). The WEKA Data Mining Software: An Update. *SIGKDD Explorations, 11*(1).

Har-Peled, S., & Raichel, B. (2014). The fréchet distance revisited and extended. *ACM Transactions on Algorithms (TALG), 10*(1), 1-22. doi:10.1145/2532646

Hashim, N. M., Omar, A. H., Ramli, S. N. M., Omar, K. M., & Din, N. (2017). Cadastral Database Positional Accuracy Improvement. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, XLII-4/W5,* 91-96. doi:10.5194/isprs-archives-XLII-4-W5-91-2017

Hausdorff, F. (1919). Dimension und äußeres Maß. *Math. Annalen, 79*(S), 157-179.

Hoaglin, D. C., Mosteller, F., & Tukey, J. W. (1983). *Understanding Robust and Exploratory Data Analysis*. New York: John Wiley & Sons.

Huang, L., Wang, S., Ye, Y., Wang, B., & Wu, L. (2010). *Feature matching in cadastral map integration with a case study of Beijing*. Paper presented at the 18th International Conference on Geoinformatic, Beijing, China.

Hussain, S., Krishna Prasad, M. H. M., & Kumar, P. (2011). Performance Evaluation Of Spatial Indexing Techniques. *International Journal of Advanced Research in Computer Science, 2*(5).

Jasinski, M. J. (1990). The Comparison of Complexity Measures for Cartographic Lines (90-1). In.

Kanchanabhan, T. E., Abbas Mohaideen, J., Srinivasan, S., & Sundaram, V. L. (2011). Optimum municipal solid waste collection using geographical information system (GIS) and vehicle tracking for Pallavapuram municipality. *Waste Manag Res, 29*(3), 323-339. Retrieved from https://www.ncbi.nlm.nih.gov/pubmed/20392789. doi:10.1177/0734242X10366272

Kang, H. (2002). *Analytical conflation of spatial data from municipal and federal government agencies.* (Ph.D diss.), The Ohio State University, Retrieved from https://search-proquest-com.dbgw.lis.curtin.edu.au/docview/305539947/?pq-origsite=primo

Kelley, C. T. (2003). *Solving nonlinear equations with Newton's method*. Philadelphia, Pa.: Philadelphia, Pa. : Society for Industrial and Applied Mathematics (SIAM, 3600 Market Street, Floor 6, Philadelphia, PA 19104).

Kemp, K. K. E. (2008). *Encyclopedia of geographic information science* Los Angeles: SAGE Publications.

Khachaturyan, A., Semenovskaya, S., & Vainshtein, B. (1979). Statistical-Thermodynamic Approach to Determination of Structure Amplitude Phases. *Sov.Phys. Crystallography, 24*(5), 519-524.

Kim, J., Yu, K., & Bang, Y. (2018). A multi-criteria decision-making approach for geometric matching of areal objects. *Transactions in GIS, 22*(1), 269-287. doi:10.1111/tgis.12307

Kobler, A., & Adamic, M. (2000). Identifying brown bear habitat by a combined GIS and machine learning method. *Ecological Modelling, 135*(2), 291-300. doi:10.1016/S0304-3800(00)00384-7

Konfrst, Z. (2004). *Parallel genetic algorithms: advances, computing trends, applications and perspectives.* Paper presented at the Parallel and Distributed Processing Symposium, 2004. Proceedings. 18th International.

Kui Liu, Y., Qiang Wang, X., Zhe Bao, S., Gomboši, M., & Žalik, B. (2007). An algorithm for polygon clipping, and for determining polygon intersections and unions. *Computers & Geosciences, 33*(5), 589-598. doi:10.1016/j.cageo.2006.08.008

Landgate. (2015). Spatial Cadastral Database (SCDB). Retrieved from http://www0.landgate.wa.gov.au/__data/assets/pdf_file/0003/8877/20150521-Final-May-2015-SCDB-Website-Page-Update.pdf

Lateș, I., Luca, M., Chirica, Ș., & Dumitrașcu, N. I. (2017). The Use of the GIS Model on the Implementation of Urban Cadastre. *Present Environment and Sustainable Development, 11*(2), 163-172. doi:10.1515/pesd-2017-0034

Leahy, F. (n.d.). User profile. Retrieved from https://www.linkedin.com/in/frank-leahy-534b57a/?originalSubdomain=au

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature, 521*(7553), 436-444. Retrieved from https://www.ncbi.nlm.nih.gov/pubmed/26017442. doi:10.1038/nature14539

Lee, D., & Schachter, B. (1980). Two algorithms for constructing a Delaunay triangulation. *International Journal of Computer & Information Sciences, 9*(3), 219-242. doi:10.1007/BF00977785

Lembo, A. (1997). *A comparison of traditional and advanced transformation techniques used to improve the positional accuracy of digital tax maps.* (Ph.D diss.), State University of New York, Retrieved from https://search-proquest-com.dbgw.lis.curtin.edu.au/docview/304373475?pq-origsite=primo

Lembo, A., O'Rourke, T., & Moses, B. (2003). Quantifying coordinate improvement of vector GIS data using rubbersheeting methods. *Surveying and Land Information Science, 63*(1), 47-54.

Li, R., Leung, Y., Huang, B., & Lin, H. (2013). A genetic algorithm for multiobjective dangerous goods route planning. *International Journal of Geographical Information Science, 27*(6), 1073-1089. doi:10.1080/13658816.2012.737921

Longley, P. A., Goodchild, M., Maguire, D. J., & Rhind, D. W. (2011). *Geographic information systems & science* (3rd ed.. ed.). Hoboken, NJ: Hoboken, NJ : Wiley.

López-Vázquez, C. (2012). Positional Accuracy Improvement Using Empirical Analytical Functions. *Cartography and Geographic Information Science, 39*(3), 133-139. doi:10.1559/15230406393133

Lupien, A. E., & Moreland, W. H. (1987). A General Approach to Map Conflation. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.522.4247&rep=rep1&type=pdf

Macduff, V. A. G. (1987). *Modeling One-to-Many Relationships on Arc Data for a Highway Information System* Paper presented at the ESRI User Conference (1987), Palm Springs, U.S.A.

Macduff, V. A. G. (1988). *Handling Graphical One-to-Many relationships in a Geographical Information System*. M.Sc. Project Report. Birkbeck College, University of London

Maier, G. (1995). Spatial Search: Structure, Complexity, and Implications. *Geographical Analysis, 25*(3), 242-251.

Merritt, R. (2005). *An Assessment of using Least Squares Adjustment to Upgrade Spatial Data in GIS.* (Ph.D diss.), University of New South Wales, Retrieved from http://unsworks.unsw.edu.au/fapi/datastream/unsworks:808/SOURCE01

Merritt, R. (2009). [Personal communication].

Microsoft. (2017). Spatial Data (SQL Server). Retrieved from https://docs.microsoft.com/en-us/sql/relational-databases/spatial/spatial-data-sql-server?view=sql-server-2017

Mitchell, M. (1996). *An Introduction to Genetic Algorithms (Complex Adaptive Systems)*: MIT Press.

National Geodetic Survey. (2017). Positioning America for the Future. Retrieved from https://www.ngs.noaa.gov/PC_PROD/pc_prod.shtml

Peng, F., Wong, M. S., Nichol, J. E., & Chan, P. W. (2016). Historical GIS data and changes in urban morphological parameters for the analysis of urban heat islands in Hong Kong. *The International Archives of the Photogrammetry, XLI-B2*, 55-62. doi:10.5194/isprs-archives-XLI-B2-55-2016

PostGIS. (n.d.). Introduction to PostGIS. Retrieved from http://www.refractions.net/products/postgis/history/

Powell, M. J. D. (1964). An efficient method for finding the minimum of a function of several variables without calculating derivatives. *Computer Journal., 7*(2), 155-162.

Preparata, F. P. (1985). *Computational geometry : an introduction / Franco P. Preparata, Michael Ian Shamos*. New York: Springer-Verlag.

Quinlan, J. R. (1993). *C4.5: Programs for Machine Learning*. San Francisco, CA, USA: Morgan Kaufmann Publishers.

Ribeiro Filho, J. L., Treleaven, P. C., & Alippi, C. (1994). Genetic-algorithm programming environments. *Computer, 27*(6), 28-43.

Roumelis, G., Vassilakopoulos, M., Corral, A., & Manolopoulos, Y. (2018). An efficient algorithm for bulk-loading xBR[formula omitted]-trees. *Computer Standards & Interfaces, 57*(C), 83-100. doi:10.1016/j.csi.2017.05.003

Ruiz-Lendínez, J., Ureña-Cámara, M., & Ariza-López, F. (2017). A Polygon and Point-Based Approach to Matching Geospatial Features. *ISPRS International Journal of Geo-Information, 6*(12), 399-423. doi:10.3390/ijgi6120399

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

Saalfeld, A. (1985). A Fast Rubber-Sheeting Transformation Using Simplicial Coordinates. *The American Cartographer, 12*(2), 169-173. Retrieved from https://doi.org/10.1559/152304085783915072. doi:10.1559/152304085783915072

Saalfeld, A. (1988). Conflation Automated map compilation. *International Journal of Geographical Information Systems, 2*(3), 217-228. doi:10.1080/02693798808927897

Saalfeld, A. (1993). Conflation: Automated map compilation. *International Journal of Geographical Information Systems, 2*(3), 217-228.

Safra, E., Kanza, Y., Sagiv, Y., Beeri, C., & Doytsher, Y. (2010). Location-based algorithms for finding sets of corresponding objects over several geo-spatial data sets. *International Journal of Geographical Information Science, 24*(1), 69-106. doi:10.1080/13658810802275560

Schuurman, N., Grund, D., Hayes, M., & Dragicevic, S. (2006). Spatial/temporal mismatch: a conflation protocol for Canada Census spatial files. *The Canadian Geographer, 50*(1), 74-84.

Shimizu, E., & Fuse, T. (2003). Rubber-sheeting of historical maps in GIS and its application to landscape visualization of old-time cities: Focusing on Tokyo of the past. Retrieved from https://pdfs.semanticscholar.org/295a/b392f0b72102e1ed3f3e8e4ab729ed47a540.pdf

Shnaidman, A., Shoshani, U., & Doytsher, Y. (2013). Genetic Algorithms: a stochastic approach for improving the current cadastre accuracies. *Survey Review, 44*(325), 102-110. doi:10.1179/1752270611y.0000000012

Siriba, D. N., Dalyot, S., & Sester, M. (2013). Geometric quality enhancement of legacy graphical cadastral datasets through thin plate splines transformation. *Survey Review, 44*(325), 91-101. doi:10.1179/1752270611y.0000000011

Sisman, Y. (2014). Coordinate transformation of cadastral maps using different adjustment methods. *Journal of the Chinese Institute of Engineers, 37*(7), 869-882. doi:10.1080/02533839.2014.888800

Song, W. (2011). *Automated vector-vector and vector-imagery geospatial conflation.* (Ph.D diss.), University of Missouri, Retrieved from https://search-proquest-com.dbgw.lis.curtin.edu.au/docview/1237803894?pq-origsite=primo

Song, W., Keller, J. M., Haithcoat, T. L., & Davis, C. H. (2011). Relaxation-Based Point Feature Matching for Vector Map Conflation. *Transactions in GIS, 15*(1), 43-60. doi:10.1111/j.1467-9671.2010.01243.x

Spatial Tapestry. (n.d.). Spatial Tapestry. Retrieved from http://www.spatialtapestry.com/

Stoter, J., Ploeger, H., & van Oosterom, P. (2013). 3D cadastre in the Netherlands: Developments and international applicability. *Computers, Environment and Urban Systems, 40*(C), 56-67. doi:10.1016/j.compenvurbsys.2012.08.008

The GIS Encyclopedia. (2011a). Affine transformation. Retrieved from http://wiki.gis.com/wiki/index.php/Affine_transformation

The GIS Encyclopedia. (2011b). Line Feature Class. Retrieved from http://wiki.gis.com/wiki/index.php/Line_Feature_Class

The GIS Encyclopedia. (2011c). Point feature class. Retrieved from http://wiki.gis.com/wiki/index.php/Point_Feature_Class

The University of British Columbia. (n.d.). The Polygon Overlay Operation. Retrieved from http://ibis.geog.ubc.ca/courses/klink/gis.notes/ncgia/u34.html#OUT34.5.2

Tomlin, C. D. (1990). *Geographic information systems and cartographic modeling*. Englewood Cliffs, N.J.: Prentice Hall.

Tomlinson, R. F. (1974). *The Application of Electronic Computing Methods and Techniques to the Storage, Compilation, and Assessment of Mapped Data.* (Ph.D Diss.), University of London, Retrieved from http://discovery.ucl.ac.uk/1563584/1/Tomlinson_Thesis.pdf

Tong, X., Liang, D., Xu, G., & Zhang, S. (2011). Positional accuracy improvement: a comparative study in Shanghai, China. *International Journal of Geographical Information Science, 25*(7), 1147-1171. Retrieved from http://dx.doi.org/10.1080/13658816.2010.515948. doi:10.1080/13658816.2010.515948

Vandysheva, N., Tikhonov, V., Van Oosterom, P. J. M., Stoter, J. E., Ploeger, H. D., Wouters, R., & Penkov, V. (2011, May 2011). *3D Cadastre modelling in Russia.* Paper presented at the FIG Working Week 2011. "Bridging the Gap between Cultures" Marrakech, Morocco,.

Velaga, N. R., Quddus, M. A., & Bristow, A. L. (2012). Improving the Performance of a Topological Map-Matching Algorithm Through Error Detection and Correction. *Journal of Intelligent Transportation Systems, 16*(3), 147-158. doi:10.1080/15472450.2012.691852

Walter, V., & Fritsch, D. (1999). Matching spatial data sets: a statistical approach. *International Journal of Geographical Information Science, 13*(5), 445-473. doi:10.1080/136588199241157

Western Australian Planning Commission. (1996). Planning Bulletin. Western Australia. Retrieved from https://www.planning.wa.gov.au/dop_pub_pdf/pb15Jun1996r.pdf

White, M. S., & Griffin, P. (2013). Piecewise Linear Rubber-Sheet Map Transformation. *The American Cartographer, 12*(2), 123-131. doi:10.1559/152304085783915135

Wiener, J. M., Ehbauer, N. N., & Mallot, H. A. (2009). Planning paths to multiple targets: memory involvement and planning heuristics in spatial problem solving. *Psychol Res, 73*(5), 644-658. Retrieved from https://www.ncbi.nlm.nih.gov/pubmed/18998160. doi:10.1007/s00426-008-0181-3

Wikipedia. (2018a). ArcSDE. Retrieved from https://en.wikipedia.org/wiki/ArcSDE

Wikipedia. (2018b). Oracle Spatial and Graph. Retrieved from https://en.wikipedia.org/wiki/Oracle_Spatial_and_Graph

Witten, I. H., Frank, E., Hall, M. A., & Pal, C. J. (2017). *Data mining : practical machine learning tools and techniques* (4th ed.). Cambridge, Massachusets: Morgan Kaufmann.

Yan, L. e., & Igi Global, p. (2016). *Handbook of research on innovative database query processing techniques / Li Yan, editor*: Hershey, Pennsylvania (701 E. Chocolate Avenue, Hershey, PA 17033, USA) : IGI Global.

Yanling, H., Wenjing, T., & Yuxin, Z. (2008). Point feature adjusting algorithm in digital map based on multi-evaluation factors. In (pp. 5842-5846).

Yuan, S., & Tao, C. (1999). *Development of Conflation Components.* Paper presented at the Geoinformatics and Socioinformatics, Ann Arbor, Michigan.

Zandbergen, P. A. (2008). Positional Accuracy of Spatial Data: Non-Normal Distributions and a Critique of the National Standard for Spatial Data Accuracy. *Transactions in GIS, 12*(1), 103-130. doi:10.1111/j.1467-9671.2008.01088.x

Zhang, L., He, Z., & Liu, Y. (2017). Deep object recognition across domains based on adaptive extreme learning machine. *Neurocomputing, 239*, 194-203. doi:10.1016/j.neucom.2017.02.016

Every reasonable effort has been made to acknowledge the ownership of copyright material. I would be pleased to hear from any copyright owner who has been omitted or incorrectly acknowledged.