

UNIVERSITY OF CALGARY

3-D Cadastral Boundary Relationship Classification Algorithms using Conformal Geometric
Algebra

by

Dillon Pullano

A THESIS

SUBMITTED TO THE FACULTY OF GRADUATE STUDIES
IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE
DEGREE OF MASTER OF SCIENCE

GRADUATE PROGRAM IN GEOMATICS ENGINEERING

CALGARY, ALBERTA

APRIL, 2021

© Dillon Pullano 2021

Abstract

As urban centers continue to grow and develop, there is an increasing need for institutions to be able to digitally model and perform relationship analysis on 3-D cadastral boundary data. 3-D boundary analysis can be performed through visual inspection of survey plan drawings, but this often requires professional expertise such as a land surveyor or lawyer. Being able to digitally model 3-D cadastral boundaries and the relationships between them would allow for a more rigorous 3-D boundary analysis approach. The problem this study addresses is related to the relationship analysis between 3-D cadastral boundaries being stored in a cadastral database management system.

This study examined the development, testing, and application of methodological processes and algorithms that were designed to classify various geometrical and topological relationships between the boundary components of two 3-D cadastral units to solve 3-D cadastral boundary problems. It applied established mathematical theory using Conformal Geometric Algebra (CGA) objects and operational techniques, in combination with various 3-D point-point distance evaluations and geometric concepts to the classification of relationships between 3-D cadastral boundaries. It developed and tested a set of data modelling processes and algorithms that can be used to identify if 3-D cadastral boundary components are touching, intersecting, or are disjoint from one another. A literature search suggests that the theory and methodology as it was applied in this study have not been used to classify topological relationships between 3-D cadastral boundaries elsewhere.

A literature search related to 3-D cadastres and 3-D property management was used to identify cadastral objects that can be registered in various jurisdictions and to identify data structures that are commonly used to represent 3-D property boundaries. The scope of the boundary components considered here was limited to 3-D points with [x, y, z] coordinates, straight lines bound by two end points, and flat planes bound by four edge lines that may define the boundary of a volumetric legal space such as a cadastral unit registered using a strata or condominium survey plan. A literature search related to modelling 3-D boundary relationships identified topological frameworks that can be used to organize relationships between 3-D boundary components, as well as identified existing CGA mathematical theory that could be used to perform topological analysis on 3-D boundaries. The relationships modelled here were limited to 3-D boundary components touching, intersecting, or being disjoint from each other.

The approach to classifying relationships between two 3-D cadastral units A and B was to first evaluate the relationships between all boundary components in unit A with all boundary components in unit B. These relationships were then combined and interpreted together to describe the relationship between the two 3-D cadastral units A and B. Six sets of data flow processing algorithms were developed to determine the relationship classifications of the point-point, line-point, line-line, plane-point, plane-line, and plane-plane boundary component pair sets between the two cadastral units. These algorithms were coded in MATLAB and interacted with a CGA program GAViewer. CGA objects and operations were utilized to sort boundary component pairs into a projected relationship category such as components being parallel, collinear, coplanar, or intersecting at a point or line. A series of 3-D point-point distances and additional Conformal Geometric Algebra operations were then evaluated to classify the final relationships.

The classification processes were first validated using seven simulated experimental testing datasets, each consisting of two cube-like units. The classification processes were then applied to a cadastral dataset that was derived from a 3-D condominium survey plan registered in Alberta, Canada. Relationships for all datasets were known *a priori* to running the experiments and relationship classification results were validated manually through visual inspection to determine if the implementation program had produced the expected results. Results from the experimental datasets support the methods that were proposed to classify 53 distinct types of topological relationships between 3-D boundary component pair sets.

The methods developed here were not verified as being universally correct with all 3-D cadastral units, but rather have been validated through the testing of the simple cube-like cadastral experiments chosen. While this type of boundary relationship analysis can be done through visual inspection of survey plans, the methods developed here are more mathematically rigorous. These processes could be leveraged by land surveyors and land administration professionals when analyzing 3-D survey plan boundaries.

The theoretical contribution of this study is that the experimental results tested support how the methods that were developed using CGA along with 3-D point-point distance evaluations can be applied to classify topological relationships between cube-like 3-D cadastral boundaries. None of the experimental datasets tested show the methods developed here to be false. The practical contribution of this study is that it showed how the methods developed here can be applied to solving a practical 3-D cadastral boundary problem example in the land surveying field, specifically towards validating a shared boundary between two adjacent 3-D cadastral units as is intended on the plan before survey plan registration.

Acknowledgements

I would like to thank my supervisors, Michael Barry, and Xin Wang for their assistance in guiding me through the writing and editing of this thesis. Their feedback on my writing and their support of my ideas were very helpful and have been appreciated throughout the duration of completing my degree.

The main financial support for this research was provided by the Chair in Land Tenure and Cadastral Systems. For this I am grateful. I would also like to thank the Department of Geomatics Engineering and the Faculty of Graduate Studies at the University of Calgary for providing funding through scholarships, grants, and teaching assistant opportunities.

I would like to thank my family and friends for being supportive and understanding throughout the years of this degree. I would like to thank those who read some of my writing to provide valuable feedback including but not limited to Alex Cameron, Cheska Ronquillo, Dennis Muthama, Erik Ma, Jeremy Crowe, Kent Jones, Mary Pauls, Matt Sakatch, Nick Jarding, Nicolas Lambert, and Rodrigo Silva. For everyone who helped me, whether it was through providing feedback on my writing, helping me stay focused, or listening to and helping me develop my ideas before I could get them onto paper, I will forever be grateful.

Dedication

This thesis is dedicated to my supportive family, Dominic, Susan, Lindsay, and Dominique.

Table of Contents

Abstract.....	ii
Acknowledgements	iv
Dedication.....	v
Table of Contents	vi
List of Tables	x
List of Figures and Illustrations	xii
List of Symbols, Abbreviations, and Nomenclature.....	xv
CHAPTER ONE: INTRODUCTION	1
1.1 Introduction	1
1.2 Background	1
1.2.1 <i>Limitations of a 2-D Cadastral System</i>	3
1.2.2 <i>FIG Working Group on 3-D Cadastres Topics of Research</i>	4
1.3 Problem Statement.....	5
1.4 Research Objective	6
1.5 Questions to Guide the Research.....	6
1.5.1 <i>Questions Related to Research on 3-D Cadastres and the 3-D Cadastre in Alberta</i>	7
1.5.2 <i>Questions Related to 3-D Boundary Relationship Modelling & Topology</i>	7
1.6 Research Activities and Methods	8
1.7 Scope of the Research.....	10
1.8 Contribution to Knowledge and Significance of the Research.....	11
1.9 Thesis Organization	12
CHAPTER TWO: LITERATURE REVIEW RELATED TO 3-D CADASTRES AND 3-D PROPERTY MANAGEMENT	13
2.1 Potential Uses of a 3-D Cadastre	14
2.2 Components and Organization of a 3-D Cadastre	15
2.3 Legal Foundations for 3-D Property Registration	17
2.3.1 <i>3-D Rights, Restrictions, and Responsibilities</i>	18
2.3.2 <i>National Case Studies Regarding 3-D Property</i>	19
2.4 Surveying and Registering 3-D Property Boundaries in Alberta.....	20

<i>2.4.1 Condominium Title and Registration</i>	21
<i>2.4.2 Strata Title and Registration</i>	23
<i>2.5 Approaches to Digitally Representing 3-D Boundaries.....</i>	24
<i>2.5.1 Approaches to Modelling 2-D and 3-D Boundaries</i>	25
<i>2.5.2 Spatial Data Formats and Representation</i>	26
<i>2.5.3 Relationship Validations between 3-D Boundaries</i>	27
<i>2.6 Chapter Summary</i>	29
CHAPTER THREE: LITERATURE REVIEW RELATED TO 3-D BOUNDARY RELATIONSHIPS AND MATHEMATICAL THEORY.....	30
<i>3.1 Literature Review on Topology Frameworks and Relationship Modelling</i>	30
<i>3.1.1 Foundations for Describing Topological Relationships Between Spatial Objects</i>	31
<i>3.1.2 Organizing Topological Relationships between Geometric Objects in 3-D Space.....</i>	35
<i>3.1.3 Approaches to Implementing Topological Analysis between 3-D Cadastral Objects.....</i>	41
<i>3.2 Conformal Geometric Algebra Mathematical Theory and Application.....</i>	45
<i>3.2.1 CGA Spatial Framework</i>	45
<i>3.2.2 Modelling Cadastral Objects using CGA</i>	50
<i>3.3 Chapter Summary</i>	52
CHAPTER FOUR: METHODS AND MATHEMATICAL MODEL DESIGN	53
<i>4.1 Defining 3-D Cadastral Object Boundary Components.....</i>	53
<i>4.1.1 3-D Boundary Points (BP)</i>	54
<i>4.1.2 3-D Boundary Lines (BL)</i>	56
<i>4.1.3 3-D Boundary Planes (BPL).....</i>	57
<i>4.2 Cadastral Object Relationship Classification Approach and Projected CGA Relationships.....</i>	58
<i>4.2.1 Relationship Description using Boundary Component Pairs</i>	58
<i>4.2.2 CGA Intersections and Projected Relationships.....</i>	60
<i>4.3 Point-Point (BP-BP) Relationship Classification Methods</i>	62
<i>4.4 Line-Point (BL-BP) Relationship Classification Methods.....</i>	63
<i>4.4.1 Collinear BL-BP Classification Algorithm.....</i>	64
<i>4.5 Line-Line (BL-BL) Relationship Classification Methods.....</i>	66
<i>4.5.1 Common Collinear BL-BL Classification Algorithm</i>	68
<i>4.5.2 Reverse Collinear BL-BL Classification Algorithm</i>	70

<i>4.5.3 Coplanar BL-BL Classification Algorithm</i>	72
4.6 Plane-Point (BPL-BP) Relationship Classification Methods	74
<i>4.6.1 Coplanar BPL-BP Classification Algorithm</i>	75
4.7 Plane-Line (BPL-BL) Relationship Classification Methods	79
<i>4.7.1 Intersecting BPL-BL Classification Algorithm</i>	81
<i>4.7.2 Coplanar BPL-BL Classification Algorithm</i>	83
4.8 Plane-Plane (BPL-BPL) Relationship Classification Methods	89
<i>4.8.1 Intersecting BPL-BPL Classification Algorithm</i>	90
<i>4.8.2 Coplanar BPL-BPL Classification Algorithm</i>	95
4.9 Chapter Summary	97
 CHAPTER FIVE: IMPLEMENTATION OF METHODS	98
5.1 Creating Parameters for Testing Datasets (Building Objects from Coordinate Sets)	99
<i>5.1.1 Preparing Initial 3-D Cadastral Unit Boundary Storage Tables</i>	101
<i>5.1.2 Generating CGA Parameters for 3-D Boundary Components using GAViewer</i>	102
5.2 Generating and Storing CGA Geometric and Intersection Results.....	103
5.3 Interpreting CGA Analysis Results and Classifying Final Relationships	106
<i>5.3.1 Determining Projected Relationships from CGA Analysis Results</i>	106
<i>5.3.2 Classifying Final Boundary Component Pair Relationships</i>	109
5.4 Chapter Summary	112
 CHAPTER SIX: RESULTS AND APPLYING METHODS TO 3-D CADASTRAL BOUNDARIES..	114
6.1 Summary of Relationship Classifications	114
6.2 Simulated Experimental Testing Datasets and Classification Results	119
<i>6.2.1 Simulated Experimental Test 1</i>	120
<i>6.2.2 Simulated Experimental Test 2</i>	125
<i>6.2.3 Simulated Experimental Test 3</i>	129
<i>6.2.4 Simulated Experimental Test 4</i>	134
<i>6.2.5 Simulated Experimental Test 5</i>	139
<i>6.2.6 Simulated Experimental Test 6</i>	144
<i>6.2.7 Simulated Experimental Test 7</i>	149
6.3 Cadastral Condominium Experimental Test	154
6.4 Chapter Summary	161

CHAPTER SEVEN: DISCUSSION AND CONCLUSIONS	163
7.1 Research Summary	163
7.2 Conclusions and Contribution to Knowledge.....	164
7.3 Discussion on Methodological Approach	166
7.3.1 <i>Discussion on Data Format Chosen</i>	166
7.3.2 <i>Relationship Classifications Compared to the 9IM in 3-D Space</i>	167
7.3.3 <i>Geometrical Representations for Relationship Classifications</i>	169
7.4 Recommendations for Future Research	170
7.4.1 <i>Expanding Topological Descriptors</i>	170
7.4.2 <i>Expand Methods to Include Solid Bodies as Boundary Components.</i>	171
7.4.3 <i>Applying Methodology to Different Data Formats</i>	172
7.4.4 <i>Alternate Approach for Determining Projected Relationships</i>	172
7.4.5 <i>Optimizing Implementation Code</i>	173
7.5 Concluding Remarks.....	174
REFERENCES - GENERAL.....	175
REFERENCES – LEGISLATION AND GOVERNMENT DOCUMENTS	181

List of Tables

Table 3.1: Eight Topological Relationships and the 4IM (Derived from Egenhofer, 1989)	32
Table 3.2: The 9-Intersection Model Matrix (Derived from Egenhofer and Herring, 1990).....	33
Table 3.3: Comparison of 4IM, 9IM DEM, and DE+9IM (Derived from Clementini & Di Felice, 1995)	34
Table 3.4: Relationships in 3-D Space using the 9-Intersection Model (Derived from Zlatanova, 2000)..	36
Table 3.5: Relationships According to the Dimensional Model (Derived from Billen et al. 2002).....	39
Table 3.6: Disjoint and Touch Relationships in a Cadastral System (Derived from Fu et al., 2018).....	43
Table 3.7: 32 Subspace Blade Parameters of CGA R4,1.....	47
Table 4.1: Sample of Parameters for 3-D Boundary Points.....	55
Table 4.2: Sample of Parameters for 3-D Boundary Lines.....	56
Table 4.3: Sample of Parameters for 3-D Boundary Planes.....	57
Table 4.4: Boundary Component Pairs between Cadastral Units A and B.....	59
Table 4.5: Projected CGA Intersection Interpretations.....	61
Table 5.1: Initial CGA Operation Checks Implemented between Boundary Component Pairs	104
Table 6.1: Index for Testing Datasets and Overlap Relationship Classifications	116
Table 6.2: Index for Testing Datasets and Disjoint Relationship Classifications	118
Table 6.3: Testing Dataset 1 - 3-D Boundary Point Coordinates.....	121
Table 6.4: Testing Dataset 1 - 3-D Boundary Line Storage Parameters.....	122
Table 6.5: Testing Dataset 1 - 3-D Boundary Plane Storage Parameters	122
Table 6.6: Testing Dataset 1 - Program Output	124
Table 6.7: Testing Dataset 2 - 3-D Boundary Point Coordinates.....	126
Table 6.8: Testing Dataset 2 - 3-D Boundary Line Storage Parameters.....	127
Table 6.9: Testing Dataset 2 - 3-D Boundary Plane Storage Parameters	127
Table 6.10: Testing Dataset 2 - Program Output.....	128
Table 6.11: Testing Dataset 3 - 3-D Boundary Point Coordinates	130
Table 6.12: Testing Dataset 3 - 3-D Boundary Line Storage Parameters	131
Table 6.13: Testing Dataset 3 - 3-D Boundary Plane Storage Parameters	131
Table 6.14: Testing Dataset 3 - Program Output.....	133
Table 6.15: Testing Dataset 4 - 3-D Boundary Point Coordinates	136
Table 6.16: Testing Dataset 4 - 3-D Boundary Line Storage Parameters	136
Table 6.17: Testing Dataset 4 - 3-D Boundary Plane Storage Parameters	137

Table 6.18: Testing Dataset 4 - Program Output.....	138
Table 6.19: Testing Dataset 5 - 3-D Boundary Point Coordinates	141
Table 6.20: Testing Dataset 5 - 3-D Boundary Line Storage Parameters	141
Table 6.21: Testing Dataset 5 - 3-D Boundary Plane Storage Parameters	141
Table 6.22: Testing Dataset 5 - Program Output.....	143
Table 6.23: Testing Dataset 6 - 3-D Boundary Point Coordinates	146
Table 6.24: Testing Dataset 6 - 3-D Boundary Line Storage Parameters	146
Table 6.25: Testing Dataset 6 - 3-D Boundary Plane Storage Parameters	146
Table 6.26: Testing Dataset 6 - Program Output.....	148
Table 6.27: Testing Dataset 7 - 3-D Boundary Point Coordinates	151
Table 6.28: Testing Dataset 7 - 3-D Boundary Line Storage Parameters	151
Table 6.29: Testing Dataset 7 - 3-D Boundary Plane Storage Parameters	152
Table 6.30: Testing Dataset 7 – Sample of Program Output	153
Table 6.31: Condominium Testing Dataset - 3-D Boundary Point Coordinates.....	158
Table 6.32: Condominium Testing Dataset - 3-D Boundary Line Storage Parameters.....	158
Table 6.33: Condominium Testing Dataset - 3-D Boundary Plane Storage Parameters	159
Table 6.34: Condominium Testing Dataset – Sample of Program Output	160

List of Figures and Illustrations

Figure 1.1: Common 3-D Ownership Schema (Barry, M. – used with Permission)	2
Figure 1.2: Example of 3-D Condominium Boundaries using 2-D Drawings – Alberta, Canada	3
Figure 2.1: Basic structure of the LADM packages	16
Figure 2.2: Condominium Survey Cadastral Object Boundary Types – Alberta, Canada.....	22
Figure 2.3: Strata Survey Cadastral Object Boundary Types – Alberta, Canada.....	23
Figure 2.4: General Structure of Constructed 3-D Volumetric Objects (Derived from Ying et al., 2015).	26
Figure 3.1: Description of Ordered Points with Respect to the DM (Derived from Billen et al., 2002)	37
Figure 3.2: Cadastral Elements with Extensions and Boundaries with Respect to the DM (Derived from Billen et al., 2002).....	38
Figure 3.3: Meet Relationship Distinctions using the Dimensional Model (Derived from Billen et al. 2002)	40
Figure 3.4: Visual Representation of Topological Relationships (Derived from Fu et al., 2018).....	42
Figure 3.5: Summary of Relationships between Points, Lines, and Planes (Derived from Selig, 2000)....	43
Figure 3.6: CGA Objects of Interest (with Blade Parameters).....	48
Figure 4.1: Overview of 3-D Boundary Components Modelled in this Study.....	54
Figure 4.2: Projected CGA Relationship Categories Shown using Disjoint Examples.....	60
Figure 4.3: BP-BP Classification Algorithm	62
Figure 4.4: Final BP-BP Relationship Classifications.....	63
Figure 4.5: BL-BP Classification Algorithm	63
Figure 4.6: BL-BP Projected Relationship Categories	64
Figure 4.7: Collinear (COL) BL-BP Classification Algorithm.....	65
Figure 4.8: Final Collinear (COL) BL-BP Relationship Classifications	65
Figure 4.9: BL-BL Classification Algorithm	66
Figure 4.10: BL-BL Projected Relationship Categories	67
Figure 4.11: Common Collinear (C-COL) BL-BL Classification Algorithm.....	68
Figure 4.12: Final Common Collinear (C-COL) BL-BL Relationship Classifications	69
Figure 4.13: Reverse Collinear (R-COL) BL-BL Classification Algorithm.....	70
Figure 4.14: Final Reverse Collinear (R-COL) BL-BL Relationship Classifications	71
Figure 4.15: Coplanar (COP) BL-BL Classification Algorithm	72
Figure 4.16: Final Coplanar (COP) BL-BL Relationship Classifications.....	73

Figure 4.17: BPL-BP Classification Algorithm	74
Figure 4.18: BPL-BP Projected Relationship Categories	74
Figure 4.19: Coplanar (COP) BPL-BP Classification Algorithm	76
Figure 4.20: Final Coplanar (COP) BPL-BP Relationship Classifications.....	77
Figure 4.21: Coplanar (COP) BPL-BP Sightline Determination Example.....	78
Figure 4.22: BPL-BL Classification Algorithm	79
Figure 4.23: BPL-BL Projected Relationship Categories	80
Figure 4.24: Intersecting (INT) BPL-BL Classification Algorithm	81
Figure 4.25: Final Intersecting (INT) BPL-BL Relationship Classifications	82
Figure 4.26: Coplanar (COP) BPL-BL Classification Algorithm (Part 1 of 2).....	83
Figure 4.27: Coplanar (COP) BPL-BL Example (Part 1 of 2).....	85
Figure 4.28: Final Coplanar (COP) BPL-BL Relationship Classifications	86
Figure 4.29: Coplanar (COP) BPL-BL Classification Algorithm (Part 2 of 2).....	87
Figure 4.30: Coplanar (COP) BPL-BL Example (Part 2 of 2).....	88
Figure 4.31: BPL-BPL Classification Algorithm.....	89
Figure 4.32: BPL-BPL Projected Relationship Categories.....	89
Figure 4.33: Intersecting (INT) BPL-BPL Classification Algorithm	91
Figure 4.34: Final Intersecting (INT) BPL-BPL Relationship Classifications	92
Figure 4.35: Intersecting (INT) BPL-BPL Example	94
Figure 4.36: Coplanar (COP) BPL-BPL Classification Algorithm.....	95
Figure 4.37: Final Coplanar (COP) BPL-BPL Relationship Classifications	96
Figure 5.1: Boundary Component Limits, Extensions, and CGA Representations.....	100
Figure 5.2: Process Flow for Generating and Storing 3-D Boundary Component CGA Parameters.....	102
Figure 5.3: Summary of Projected Relationships between Boundary Component Pair Types.....	107
Figure 5.4: Organization of Final MATLAB Classification functions.....	109
Figure 6.1: Overlap Relationship Classifications between Component Pairs.....	115
Figure 6.2: Disjoint Relationship Classifications between Component Pairs.....	117
Figure 6.3: Overview of Simulated Testing Datasets and Relationship Classifications.....	119
Figure 6.4: Testing Dataset 1 Visualization.....	120
Figure 6.5: Testing Dataset 1 - Existing Overlap Relationship Classifications	123
Figure 6.6: Testing Dataset 2 Visualization.....	125
Figure 6.7: Testing Dataset 2 - Existing Overlap Relationship Classifications	127

Figure 6.8: Testing Dataset 3 Visualization.....	129
Figure 6.9: Testing Dataset 3 - Existing Overlap Relationship Classifications	132
Figure 6.10: Testing Dataset 4 Visualization.....	135
Figure 6.11: Testing Dataset 4 - Existing Overlap Relationship Classifications	137
Figure 6.12: Testing Dataset 5 Visualization.....	140
Figure 6.13: Testing Dataset 5 - Existing Overlap Relationship Classifications	142
Figure 6.14: Testing Dataset 6 Visualization.....	145
Figure 6.15: Testing Dataset 6 - Existing Overlap Relationship Classifications	147
Figure 6.16: Testing Dataset 7 Visualization.....	150
Figure 6.17: Testing Dataset 7 - Existing Disjoint Relationship Classifications	152
Figure 6.18: Condominium Experiment Survey Site Layout.....	154
Figure 6.19: Condominium Building Cross Section and Typical Floor Plan	155
Figure 6.20: Calculated Floor and Ceiling Boundary Point (x, y, z) Coordinates	156
Figure 6.21: Condominium Testing Dataset Visualization.....	157
Figure 6.22: Condominium Testing Dataset - Existing Overlap Relationship Classifications	159

List of Symbols, Abbreviations, and Nomenclature

0-D (0D)	Zero-dimensional
1-D (1D)	One-dimensional
2-D (2D)	Two-dimensional
3-D (3D)	Three-dimensional
3-D cadastre (operational definition)	A land information system in which the boundaries of both 3-D property and the land it resides on are clearly and indisputably recorded.
4IM	4-intersection topology model
9IM	9-intersection topology model
BIM	Building Information Modelling
BL	3-D boundary line
BL-BL	A line-line boundary component pair
BL-BP	A line-point boundary component pair
Boundary (∂A)	The boundary of a spatial object A, defined as the space that contains the object's interior
BP	3-D boundary point
BP-BP	A point-point boundary component pair
BPL	3-D boundary plane
BPL-BL	A plane-line boundary component pair
BPL-BP	A plane-point boundary component pair
BPL-BPL	A plane-plane boundary component pair
Cadastral parcel	A cadastral property defined using 2-D boundaries
Cadastral unit	A cadastral property defined using 2-D and 3-D boundaries
CGA	Conformal Geometric Algebra
DEM	Dimension extended topology model
DE+9IM	Dimension extended 9-intersection model
DBMS	Database Management System
DM	Dimensional Model spatial framework
Exterior (A^-)	The exterior of a spatial object A, defined as the space that is not the object's boundary or interior

FIG	International Federation of Surveyors
GAViewer	A program for performing geometric algebra computations
GIS	Geographic information system
ISO	International Organization for Standardization
Interior (A°)	The interior of a spatial object A, defined as the space that is contained within the object's boundary
K-blade	Parameters that represent subspace coordinates in CGA
LADM	Land Administration Domain Model
MATLAB	Matrix laboratory programming language
Multivector	Consists of multiple CGA k-blades that are used to represent various spatial objects
Order	The dimension (0-D, 1-D, 2-D, 3-D) of a point that is part of a spatial object
Projected line (L)	A straight continuous line that does not have any boundary points containing it
Projected plane (PL)	A flat continuous plane that does not have any boundary lines containing it
Projected relationships	Relationships between points, projected lines, and projected planes which can be parallel, skew, collinear, coplanar, or intersecting at a point or a line
\mathbb{R}^2	Euclidean 2-dimensional space
\mathbb{R}^3	Euclidean 3-dimensional space
$\mathbb{R}^{4,1}$	Conformal 5-dimensional space
RRRs	Rights, restrictions, and responsibilities associated with 2-D and 3-D cadastral property
Survey plan (condominium)	A survey plan that is used to register private and group interests in 2-D and 3-D property. Boundaries can be georeferenced to known coordinates or defined relative to physical structures
Survey plan (strata)	A survey plan that is used to register private interests in 3-D property. Boundaries need to be georeferenced to known coordinates
Topological relationships	Relationships that are invariant under topological transformations such as translations, scaling, and rotations

Chapter One: **Introduction**

1.1 Introduction

This study reports on the development and testing of methodological processes that were implemented to perform topological relationship analysis on 3-D legal boundary components. It applies the established mathematical theory of Conformal Geometric Algebra (CGA) objects and operations in combination with various point-point distance evaluations to develop a set of data flow processes and algorithms to achieve this. Specifically, it investigates topological and geometric relationships between sets of 3-D cadastral units and how processes could be applied to solve 3-D cadastral boundary problems.

This chapter introduces the reader to key concepts and questions related to the development of data modelling processes that can be used to classify geometric and topological relationships for 3-D cadastral purposes. The chapter is structured as follows. Section 1.2 introduces the general problem that this study focuses on and provides a background on its importance. Section 1.3 defines an operational definition for a 3-D cadastre as it is used here and presents the main problem statement. Section 1.4 presents the primary research objective. Section 1.5 and Section 1.6 present the research questions and activities that are relevant to achieving the primary objective. Section 1.7 discusses the scope of the study and Section 1.8 discusses the significance of this research and how it contributes to knowledge in the 3-D land management and professional surveying field. Section 1.9 outlines the organization and structure of the study.

1.2 Background

This section briefly describes why 3-D cadastre development is important. It provides background context on 3-D cadastre research, identifies the general gap in knowledge this study aims to contribute towards, and provides the rationale for performing this study. It highlights the functional limitations that can exist when modelling 3-D legal boundary data with existing 2-D topology models and frames the problem within the context of common research topics associated with 3-D cadastres. The topics discussed here are built upon in the literature review on 3-D cadastres presented in Chapter 2.

Traditionally, a cadastre is used to store all legal rights and many interests in land, along with a 2-D parcel description of the land (Stoter, 2004). As municipalities and other jurisdictions grow and develop, there is an increased need for mapping and managing complex legal structures that exist above and below the ground such as rights, restrictions, and responsibilities associated with buildings, underground facilities,

and utilities, among other right spaces such as is shown by El-Mekawy et al. (2015) and Kitsakis et al. (2016).

Currently, 3-D property can be registered to a title with a supplementary condominium or strata subdivision survey plan in many jurisdictions. In Alberta, a condominium plan can register group and personal interests in 2-D and 3-D property while a strata plan can register ownership rights overlapping in 3-D space. In both plan types, cadastral boundaries are recorded and registered as 2-D drawings, vertical profiles, cross-sections, and 3-D isometric drawings. An example layout describing an isometric view can be seen in Figure 1.1 below. Boundary analysis can be performed by visual inspection of the survey plan, but this often requires professional expertise such as a land surveyor or lawyer. In more complex layouts, investigating these plans becomes more difficult and can be time-consuming.

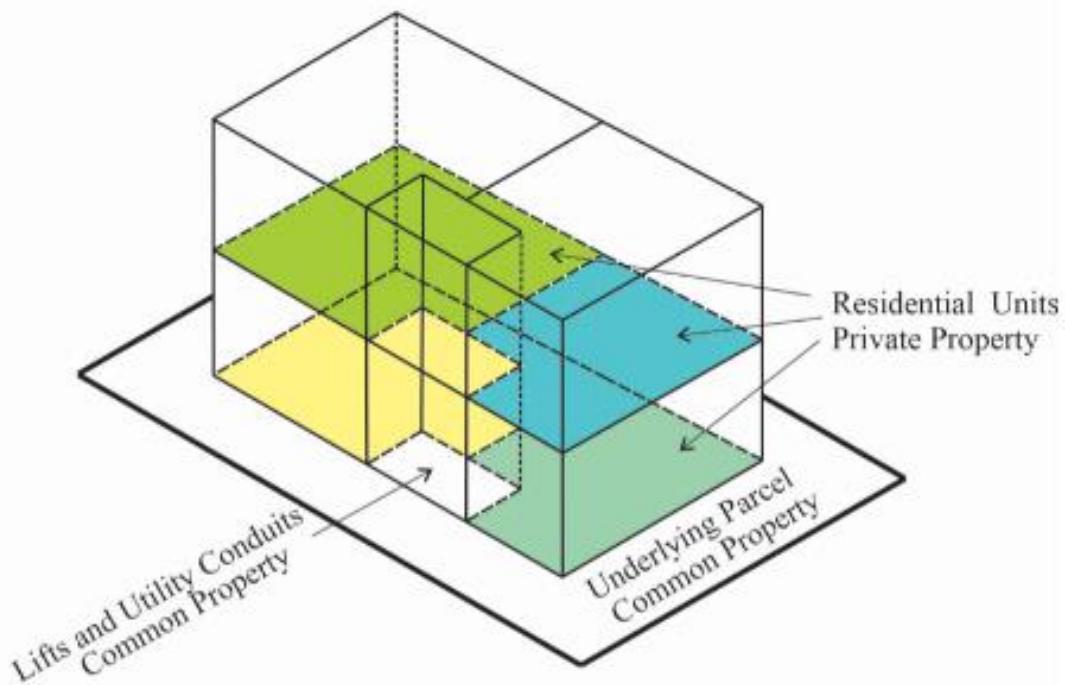


Figure 1.1: Common 3-D Ownership Schema (Barry, M. – used with Permission)

The introduction of 3-D legal structures into existing 2-D digital storage and mapping schemes results in restricted object management and boundary analysis capabilities. In this scenario, topological analysis is limited to the boundary of the underlying 2-D parcel and adjacent 2-D parcel boundaries. Relationships between the 3-D units contained within the underlying parcel need to be analyzed visually using plans.

1.2.1 Limitations of a 2-D Cadastral System

The reasons 3-D cadastral models and processes should be further developed can be highlighted by first discussing the limitations of registering 3-D objects in a 2-D cadastral system. This section highlights areas in which a 2-D cadastre is becoming increasingly limited or fails to provide unambiguous insight or analysis about a 3-D legal situation.

The extended use of land, water, and air has resulted in complex legal structures such as ownership and other rights linked to 3-D property (El-Mekawy, 2015). Zhang, J. (2016) states that 2-D cadastral systems are becoming increasingly limited for handling complex 3-D cadastral objects in urban areas. Drobetz et al. (2017) found several studies which showed that 2-D registration of real property no longer meets the demands of modern society. Stoter (2004) suggests that 2-D projected representations of rights, responsibilities, and restrictions (RRRs) and other interests cannot accommodate complex, overlapping real property scenarios when rights of cadastral objects extend above or below the Earth's surface in 3-D space. This complexity being managed in a lower-dimensional 2-D system is generally more likely to cause discrepancies than if it were managed in a 3-D system.

One of the main limitations of traditional 2-D cadastres is the way that 2-D and 3-D boundaries are digitally stored relative to each other. In many cadastral systems, 3-D property boundary data is registered and stored as a set of 2-D plan views, cross-sections (side elevations), and 3-D isometric drawings that are included in the survey plan that was used to register the 3-D property. These drawings can be used together to interpret the boundary locations of different 3-D cadastral units. An example of 2-D drawings being used to represent 3-D cadastral unit boundaries can be seen in Figure 1.2 below.

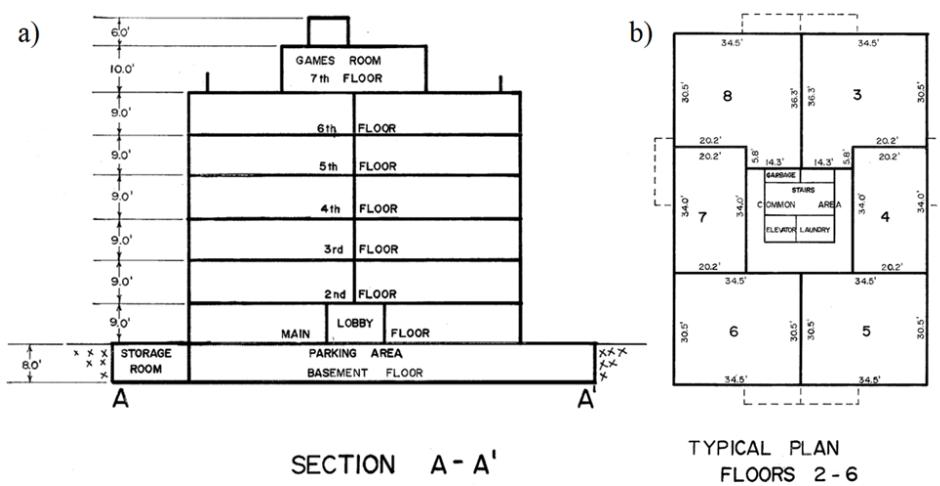


Figure 1.2: Example of 3-D Condominium Boundaries using 2-D Drawings – Alberta, Canada

Figure 1.2a shows a vertical cross-section of a condominium building. Measurements included in this drawing can be interpreted to derive the elevations of each floor and the points defining the boundary of each 3-D cadastral unit. Figure 1.2b shows a horizontal, plan view, cross-section for floors 2-6 of the same condominium building. Measurements included in this drawing can be interpreted to derive the 2-D horizontal point locations defining the boundary of each 3-D cadastral unit.

The digital 3-D boundary analysis capabilities using only these drawing representations are limited in the sense that the boundaries of objects registered in 3-D are not topologically related to one another directly in a cadastral database. In this scenario, determining relationships between adjacent 3-D cadastral units can be done indirectly through the inspection of 2-D drawings included in official survey plans.

Another shortfall of using a 2-D cadastral system to model 3-D cadastral units is that 3-D boundaries can only be visualized by looking at 2-D cross-section perspectives or isometric drawings generated from 3-D survey plans. It's not until you retain a copy of the survey plan which defines these 3-D cadastral units that you can start to get a glimpse of the situation. Even when the 3-D survey plan is acquired, the plan drawings might not be straightforward or easy to understand for all users of the cadastral system.

Pouliot & Girard (2016) performed a survey on the progress and future directions regarding the visualization component of a 3-D cadastre. One of the main conclusions of their study was that the “data to be visualized must be linked not only to physical objects but especially to legal boundaries” (Pouliot & Girard, 2016). These boundaries should relate to parcels, easements, and any restrictions or changes in rights. A proposed gap in their research involves the usability and interaction capabilities of visualizing geometrical and legal relationships in 3-D. There is currently a lack in understanding the varying needs of individuals that would be using a 3-D cadastre such as the public, city planners, land surveyors, and other professionals. Pouliot & Girard (2016) claim that understanding these specific needs and meeting them for all types of users is necessary for such a system to be accepted.

1.2.2 FIG Working Group on 3-D Cadastres Topics of Research

The International Federation of Surveyors (FIG) working group on 3-D cadastres is an international effort to develop various aspects of a 3-D cadastre through publishing research from multiple legal jurisdictions around the world. It has proposed different research topics that are related to 3-D cadastre development to establish best practices moving forward.

One main topic for 3-D cadastre research is related to 3-D cadastral information modelling. This study reviews the Land Administration Domain Model (LADM), which is a descriptive standard for 3-D cadastral

modelling published by the International Organization for Standardization (ISO). The LADM (ISO 19152:2012) organizes the spatial and non-spatial data related to 3-D property and links the legal foundations and initial registration of 3-D property for various jurisdictions. While the legal concepts are organized in detail and can be modified to any jurisdiction, the spatial data component of the model does not provide details on how to store and analyze relationships between the boundaries of 3-D objects.

Two other topics of 3-D cadastre research are associated with the legal foundations required for realizing 3-D property and research related to the initial registration of 3-D property. Legal foundations refer to the definition of different types of 3-D property and the variations of 3-D rights, restrictions, and responsibilities (RRRs) that can be associated with them. The initial registration of 3-D property refers to approaches for realizing 3-D property units as well as approaches for surveying and storing 3-D property boundaries into various types of legal documents that can be used to define and register rights in 3-D space.

The 3-D cadastre working group research topic that this study aims to contribute to relates to the development of 3-D spatial database management systems (DBMS) and processes that can be used for managing 3-D property boundaries. This topic focuses heavily on the development of 3-D data models used for storing and managing 3-D boundaries, as well as on approaches used for identifying topological relationships between 3-D boundaries and solving 3-D cadastral boundary problems.

1.3 Problem Statement

There are many definitions of a 3-D cadastre. I have adopted the following operational definition which draws extensively on Stoter (2004). For this study, a 3-D cadastre is defined as a land information system in which the boundaries of both 3-D property and the land it resides on are clearly and indisputably recorded. It is an implementation of a spatial database management system (DBMS) used to record, manage, and provide analytical insight on matters related to the legal status of land and 3-D property for a specified land jurisdiction. The 3-D objects modelled in this study are derived from legal condominium and strata property boundary concepts that can be registered in the jurisdiction of Alberta, Canada.

The specific 3-D cadastral problem that this study attempts to address is related to relationship analysis between 3-D cadastral boundaries being stored in a cadastral DBMS. The geometric data associated with a 3-D cadastral unit stored in such a database should be digitally modelled so that spatial relationship queries can be performed between the boundaries of different 3-D cadastral units. This study develops an implementation of methodological relationship classification processes that can be used to achieve this.

Such an implementation needs to have a standard for digitally representing, organizing, and storing 3-D boundary components. It needs to have defined computational data modelling processes that can be followed to properly classify and describe both geometric and topological relationships between 3-D cadastral units and their 3-D boundary components.

1.4 Research Objective

This study aimed to design and test methodological modelling processes that can be implemented to classify geometrical and topological relationships between the boundary components of two 3-D cadastral units. This was achieved by applying existing CGA mathematical algorithms and operational techniques along with 3-D point-point distance evaluations.

These computational processes were developed to solve 3-D cadastral boundary problems. Relationship classification processes should consider the data types that are commonly being used to record and register 3-D legal objects in various jurisdictions and have a digital format for organizing 3-D cadastral unit boundaries. The processes should be able to identify shared or intersecting boundaries between 3-D units. Two primary research questions were examined in achieving this objective:

1. How can topological and geometrical relationship classifications be determined between two 3-D cadastral units and their boundary components?
2. How can CGA objects and operations be applied to solving question 1?

1.5 Questions to Guide the Research

The following are specific research questions that guided the study. They frame the problem context, as well as how the investigation was structured. They are split into questions related to research on 3-D cadastres (Chapter 2) and questions related to 3-D boundary relationship modelling and topology (Chapter 3).

1.5.1 Questions Related to Research on 3-D Cadastres and the 3-D Cadastre in Alberta

1. What is the purpose of a 3-D cadastral system? (see Section 2.1)
2. Who would benefit from using a 3-D cadastral system? (see Section 2.1)
3. What is the Land Administration Domain Model? (see Section 2.2)
4. What are the main components of a 3-D cadastral data model? (see Section 2.2)
5. What rights, restrictions, and responsibilities can be defined and registered in 3-D? (see Section 2.3)
6. What types of cadastral spatial units can be registered using 3-D boundaries? (see Section 2.3)
7. What are the approaches to realizing 3-D property units? (see Section 2.3)
8. What are the types of survey plans that can be used to record and register 3-D property in Alberta? (see Section 2.4)
9. What types of 3-D cadastral property units can be registered in Alberta? (see Section 2.4)
10. How are 3-D cadastral property units structured within the underlying 2-D parcel in Alberta? (see Section 2.4)
11. What approaches exist for integrating 2-D and 3-D cadastral boundary data into a DBMS? (see Section 2.5)
12. What data formats can be used to represent 3-D boundaries digitally? (see Section 2.5)
13. What 3-D boundary components should be considered for modelling? (see Section 2.5)
14. What data checks performed on 3-D boundaries would be useful for survey plan validation? (see Section 2.5)

1.5.2 Questions Related to 3-D Boundary Relationship Modelling & Topology

15. What topological relationships can exist between spatial objects? (see Section 3.1.1)
16. What are the 4-intersection, 9-intersection, dimension extended, and dimension extended 9-intersection topological model frameworks, and how are they used to describe relationships? (see Section 3.1.1)
17. What frameworks can be used to describe geometric relationships between 3-D objects? (see Section 3.1.2)
18. How can the 9-Intersection Model be applied to distinguish relationships between multidimensional objects in 3-D space? (see Section 3.1.2)
19. What is the Dimensional Model framework as it is used for organizing relationships between spatial objects? (see Section 3.1.2)

20. How can geometric relationships be used to distinguish between different types of spatial relationships in 3-D? (see Section 3.1.3)
21. What projected relationships can exist between various 3-D boundary components? (see Section 3.1.3)
22. What approaches to processing spatial data have been used to implement relationship analysis between 3-D cadastral objects? (see Section 3.1.3)
23. What is the Conformal Geometric Algebra (CGA) spatial framework and how is it realized? (see Section 3.2.1)
24. How can 3-D boundary components be modelled using Conformal Geometric Algebra? (see Section 3.2.1)
25. What geometric and topological operators can be used to analyze relationships between spatial objects modelled using Conformal Geometric Algebra? (see Section 3.2.1)
26. What approaches have been used to determine topological relationships between 3-D cadastral objects using Conformal Geometric Algebra implementations? (see Section 3.2.2)

1.6 Research Activities and Methods

Addressing the following research objective activities was critical to achieving the primary objective. Objectives 1 through 5 below were achieved through interpreting answers to the guiding research questions presented in Section 1.5 above. Objectives 6 through 9 were achieved through the development and testing of processes that can be used to classify relationships between 3-D cadastral objects using CGA theory.

1. Identify the types of 3-D cadastral objects that can be realized in Alberta, Canada, and in various legal jurisdictions (Chapter 2)
2. Identify 3-D data structures that are commonly used to represent 3-D property boundaries (Chapter 2)
3. Decide on a data format that will be used for digitally representing 3-D boundaries (Chapter 4)
4. Identify topological relationships that can exist between 3-D boundary components (Chapter 3)
5. Identify existing mathematical theory that can be used to perform topological analysis on 3-D boundaries (Chapter 3)
6. Propose data flow processes that use existing mathematical algorithms and techniques to classify topological relationships that can exist between the 3-D boundary components of cadastral units (Chapter 4)

7. Implement the computational procedures proposed in the theoretical model that use CGA projected object representations and operations, as well as 3-D point-point distance evaluations using written MATLAB functions and other software (Chapter 5)
8. Test proposed computational procedures with simulated datasets to show the extent that they can work under (Chapter 6)
9. Test proposed computation procedures on two 3-D cadastral units derived from a real 3-D cadastral survey to show how the developed procedures can be applied to a practical 3-D boundary problem scenario (Chapter 6)

The methods to address the research activities above to achieve the primary objective of developing and testing the algorithms and data flow processes proposed in this study was as follows:

Research activities 1, 2, and 3 were performed to identify different variations of 3-D cadastral units that can be registered in various legal jurisdictions and identifying common formats that are used to register and store their 3-D boundaries. These were done to define a data format that could be used to digitally store the 3-D boundaries modelled for this study using 3-D geometry and point sets. Existing legislation for registering 3-D property in Alberta, Canada, and previous literature related to managing 3-D property boundaries in other jurisdictions were reviewed to achieve this.

Research activities 4 and 5 identified formal definitions for the possible relationships that can exist between different 3-D boundary components and identified mathematical techniques that could be applied to help distinguish between these relationships. Each of the 6 pairs of 3-D boundary components considered has a limited number of relationships that can be classified between them. Research related to defining and organizing relationships between spatial objects using topological frameworks, as well as research related to mathematical approaches used for analyzing 3-D spatial objects were reviewed to achieve this.

Research activity 6 used the results from research activities 4 and 5 to develop conceptual data flow processes that could be followed to classify relationships between individual 3-D boundary components that were defined through research activities 1, 2, and 3. Six sets of classification processes are proposed using flow chart diagrams that use Conformal Geometric Algebra mathematical operations, point-point distance checks, and condition-based decisions.

Research activity 7 consisted of generating simulated 3-D cadastral units and parameters for their 3-D boundary components, writing MATLAB functions that were used to interact with a Conformal Geometric Algebra program (GAViewer), and writing MATLAB functions that implemented the decision-based classification processes proposed in research activity 6. Parameters for 3-D boundary components were stored in EXCEL files and classifications between 3-D boundary components were output to tables.

Research activity 8 included running seven sets of data (each consisting of two simulated 3-D cadastral units) through the implementation functions developed in research activities 6 and 7 to classify relationships between their 3-D boundary components. Research activity 9 ran a dataset derived from a condominium survey plan registered in Alberta, Canada through the relationship classification functions to demonstrate how the processes developed here could be applied to a real 3-D cadastral scenario. Relationships between each set of 3-D cadastral units were known *a priori* to testing so that the proposed processes could be verified to work as designed.

1.7 Scope of the Research

This study was performed from the perspective of a professional surveying standpoint and not from a computer science world view. It is interdisciplinary and aimed primarily at land administration professionals and academics. It provides a methodological approach that can be used for 3-D cadastral plan validation and plan quality management. This study does not generate new theory related to topological relationships, but rather applies existing mathematical theory to classify topological and geometric relationships between modelled 3-D cadastral unit boundary components.

All proposed methods were implemented using a series of MATLAB functions written by the author. Conformal Geometric Algebra mathematical operations were executed in the open-source program (GAViewer) where results from these operations were imported and then analyzed in MATLAB.

The format for the datasets used to test the methods put forward in this study was defined to reflect strata and centerline boundary condominium properties in Alberta, Canada. The cadastral datasets ran through the written MATLAB programs were all simulated cube-like objects that have closed watertight volumetric boundaries. Each of these is composed of eight 3-D boundary points with [x, y, z] coordinates, 24 straight boundary lines with no curves defined by a start point and end point, and six flat piecewise planar boundary plane surfaces defined by four boundary points and four boundary lines. All planar surfaces are vertical, horizontal, or inclined at an angle. Digital parameters used to represent 3-D boundary components were stored using EXCEL file tables.

This model does not attempt to render visualizations of 3-D cadastral units or classified relationships between cadastral unit sets. Processes for implementing database queries were not developed here. The scope is limited to creating digital representations for sets of 3-D cadastral units and to classify various touch, intersection, and disjoint topological and geometric relationships that exist between their 3-D boundary components.

1.8 Contribution to Knowledge and Significance of the Research

This study has both methodological and substantive theoretical contributions, as well as practical contributions to knowledge. The methodological theoretical contribution of this study is that it developed, tested, and validated procedural data flow processes and algorithms that apply existing CGA theory to correctly classify relationships between the boundaries of volumetric 3-D cube-like objects using simulated datasets.

The substantive theoretical contribution of this study is that it applied the developed methodological processes and algorithms to the context of 3-D cadastral boundary problems by using 3-D boundary units derived from a condominium survey plan registered in Alberta, Canada. A literature search (Chapters 2 and 3) suggests that the methods developed in this study that use existing CGA and topology theory have previously not been applied to the classification of relationships between 3-D cadastral boundary components as they are applied here.

A practical contribution this study has in the 3-D land management and professional surveying field is that the methods developed here could be applied for validating boundaries in 3-D survey plans before they are registered as a legal document (i.e. survey plan checking). These validations would help to increase land tenure security. An example of this would be validating that two adjacent 3-D cadastral units share common boundary points or line and plane segments and do not intersect in 3-D space before a survey plan is submitted for registration.

The methods developed here could also increase the boundary analysis potential regarding 3-D cadastral objects being modelled in a 3-D spatial DBMS. Converting registered 3-D cadastral boundaries into digital objects and being able to classify and model the relationships between their 3-D boundary components would enhance the types of digital analysis available for land surveyors or others managing 3-D geospatial datasets. This could enable 3-D cadastral data to be used as the base dataset for other external applications related to building information modelling (BIM) or 3-D mapping. However, this is out of the scope of this study.

1.9 Thesis Organization

The remaining chapters of the study are structured as follows:

Chapter 2 reviews literature related to 3-D cadastre research and development. Various concepts related to the organization and management of a 3-D cadastral system are presented and discussed to highlight the developmental challenges associated with digitally modelling 3-D cadastral data. It compares a few of the previous modelling approaches that have been implemented while identifying the gap that this study aims to bridge.

Chapter 3 reviews literature related to 3-D boundary relationship modelling and topological relationships. It also provides an overview of mathematical concepts and theory related to the conformal model of geometric algebra. These concepts are applied in Chapters 4 - 6 to both represent cadastral boundary components and to perform topological operations between them.

Chapter 4 proposes the data flow processes that can be used to classify relationships between the boundary components of two 3-D cadastral units. It provides formal definitions for the boundary components as they are modelled in this study and discusses the approach followed to modelling relationships between their boundary components. Conceptual processes using CGA intersections and other distance checks are proposed that can be followed to classify the various topological boundary relationships that can exist between six sets of 3-D boundary component pairs.

Chapter 5 describes the methodology that was followed to implement the processes proposed in Chapter 4 in full detail. It describes how the simulated 3-D cadastral unit boundaries tested in Chapter 6 were generated and stored using a database table approach. It then describes how the conceptual data flow processes proposed in Chapter 4 were implemented using written MATLAB functions and GAViewer scripts while showing how the datasets were processed through the main MATLAB classification function.

Chapter 6 describes the seven simulated datasets that were used and presents the results for testing the topological relationship classification processes between individual boundary components. All relationships introduced in Chapter 4 were properly classified at least once between the seven datasets. Classification results from a real condominium example are presented here as well.

Chapter 7 provides a discussion and analysis on the results presented in Chapter 6 and how this model could be useful for the surveying profession. It discusses ways in which this model can be expanded to include other data types and concludes by reviewing the work that has been done and identifying how the methods presented in this study could be further applied to other areas of 3-D cadastral research.

Chapter Two: Literature Review related to 3-D Cadastres and 3-D Property Management

This chapter presents previous theory and methods related to 3-D cadastre research and development. It provides background context towards the need to design and implement the computational modelling processes presented in Chapter 4. It identifies the gap in existing research that this study aims to contribute towards that is related to modelling topological relationships between the recorded boundaries of 3-D cadastral properties. It is structured as follows.

Section 2.1 introduces the reader to the main concepts associated with a 3-D cadastre. It discusses the main purpose and other proposed uses of a 3-D cadastral system to highlight the importance of researching and developing processes related to 3-D cadastral object management.

Section 2.2 describes the different components of a 3-D cadastral system using the international data model ISO standard, the “Land Administration Domain Model” (LADM). This descriptive standard can be used to conceptually organize a 3-D cadastre into its main spatial and administrative components for a given land management jurisdiction as they would exist in a spatial database management system (DBMS).

Section 2.3 reviews the legal foundations that are required for realizing, recording, and registering 3-D property units and their boundaries in multiple jurisdictions globally. It identifies various types of 3-D property that can exist, as well as the variations of 3-D rights, restrictions, and responsibilities that can be associated with them.

Section 2.4 reviews different approaches to surveying and representing 3-D property units on official survey plans, as well as approaches to storing 3-D property boundaries into various types of legal documents in the jurisdiction of Alberta, Canada. It introduces two types of survey plans from which the datasets used in Chapter 6 can be derived.

Section 2.5 reviews literature related to the development of 3-D spatial DBMS processes that can be used for managing 3-D property boundaries. It compares three different approaches to digitally representing 3-D boundaries in a DBMS. It identifies validation checks that should be considered between 3-D property boundaries that are contained in the same survey plan. It summarizes different types of topological relationships that can exist between 3-D units registered to the same survey plan and highlights how classifying these relationships could be useful to both surveyors and end-users of a 3-D cadastral system.

2.1 Potential Uses of a 3-D Cadastre

This section introduces the reader to the main purposes and uses of a 3-D cadastre. It covers uses for the surveying profession overall, uses for those involved in urban planning, and applications that could apply 3-D cadastral datasets to supplement other external datasets for various purposes.

This section provides introductory context as to why 3-D cadastral dataset modelling processes should be developed. This section addresses research question 1: “What is the purpose of a 3-D cadastral system?” and research question 2: “Who would benefit from using a 3-D cadastral system?”.

Zhang, J. et al. (2016) state that the main purpose of a 3-D cadastral data model is to organize, store, and represent spatial representations for 3-D cadastral objects. The economy, science, and administration have an increasing demand for official three-dimensional spatial information (3-D-geodata) as a base for multiple applications linked to urban planning and property management. On a more general scale, the spatial data component of a 3-D Cadastre can be used as the base model for other applications.

Three of the biggest users of a 3-D cadastre are citizens, land developers, and land surveyors. Citizens and developers (property owners) can use the information from a cadastral system to guarantee the security of their registered interests in real property. The system can be used to provide information regarding interests (rights, restrictions, and responsibilities) in 3-D property, information about 3-D parcels, and information about the people that have interests in them (FIG, 1998). It can be used by developers to provide this information when acquiring a property for ownership or further development. Land surveyors can use the information system to access survey plans needed to record new surveys or to help with legal property boundary dispute resolution (FIG, 1998). A land surveyor could also use data analysis functionalities associated with a 3-D cadastral system to perform spatial property boundary validations before submitting a 3-D plan for registration. Data checks that could be useful to a land surveyor for validating 3-D survey plans are discussed further in Section 2.5.

Aside from its main purpose of securing land/property tenure and security, a 3-D cadastre can provide geoinformation that can be used as a potential base model for other applications. One of these applications would be to enrich the functionality of a Building Information Model (BIM) such as that described by Atazadeh et al. (2016). One of the current drawbacks of BIM is that it only incorporates detailed physical information about the building being modelled. Adding 3-D cadastral data into a BIM could incorporate a juridical (legal) aspect to an existing model by linking the detailed physical descriptions of a property with the RRRs and other interests of the people interacting with the building/property.

Aien et al. (2013) and Aien et al. (2015) developed a conceptual data model that can be used to integrate physical BIM models with legal cadastral models to increase the usability of both datasets for various applications. They showed how the model could be applied to represent physical object records of extended land use such as buildings, tunnels, and utility networks. Pouliot & Girard (2016) also showed how BIM can be used for storing the location of sub utility lines and networks while Lee et al. (2018) developed a BIM framework that could be used for managing the maintenance of utility tunnels. A 3-D cadastral system integrated with BIM could be used as a tool for city planners if it is structured and presented in a way so that spatial data can be easily analyzed and visualized.

On a larger scale, all real property in a jurisdiction could be used to create a city model and simulate various land use planning and management scenarios. Germany created a “multidimensional” land database to monitor photovoltaic technology, geothermal, and wind energy. The database includes a noise map of areas to help better plan for future developments (Seifert et al., 2016). Breunig & Zlatanova (2011) showed that 3-D geo databases could be applied to implement early warning systems against natural hazards and to coordinate emergency response scenarios. To achieve any of the examples presented above, methods to digitally represent and analyze relationships between the boundaries of 3-D cadastral units need to be developed and tested.

2.2 Components and Organization of a 3-D Cadastre

This section introduces the reader to the descriptive ISO standard data model that can be used to organize both 2-D and 3-D cadastral spatial and non-spatial (semantic) datasets. The three main data packages are discussed to organize the various components of data involved with a cadastral system that includes 3-D property. It also provides brief descriptions for some of the definitions linked to 3-D property data.

This section is used to organize the multiple components involved in 3-D cadastral datasets into a subset of less complicated concepts. While most components are related to non-spatial data, this study focuses on the spatial data component which is introduced last. This section addresses research question 3: “What is the Land Administration Domain Model?” and research question 4: “What are the main components of a 3-D cadastral data model?”.

The components of a 3-D cadastral system can be organized by reviewing the standardized model for organizing 3-D cadastral data. The LADM was created as a descriptive standard to provide “governments instruments like regulations and administrative procedures to support land tenure security, land markets,

land use planning and controls, land taxation, and the management of natural resources" (Lemmen et al., 2015). This standard can be used to describe the legal space covered by the RRRs without describing the physical space and was presented in ISO-19152 (ISO, 2012).

The LADM operates under the pretext that the data model behind any land administration system should be able to function as the core of any land administration system (Lemmen et al., 2015). Since each jurisdiction has its own legal system, the data model used needs to be flexible and widely applicable. For this reason, LADM should not be taken as a strict implementation method, but rather a descriptive standard for organizing spatial and non-spatial data related to 3-D cadastral property. It is important to note that the principles of the LADM will be used as a reference in this study, but they will not be held as a requirement. The conceptual data model for the LADM is organized into the party package 'LA_Party', the administrative package consisting of 'LA_RRR' and 'LA_BAUnit', and the spatial unit package 'LA_SpatialUnit'. These three main packages can be seen in Figure 2.1 below.

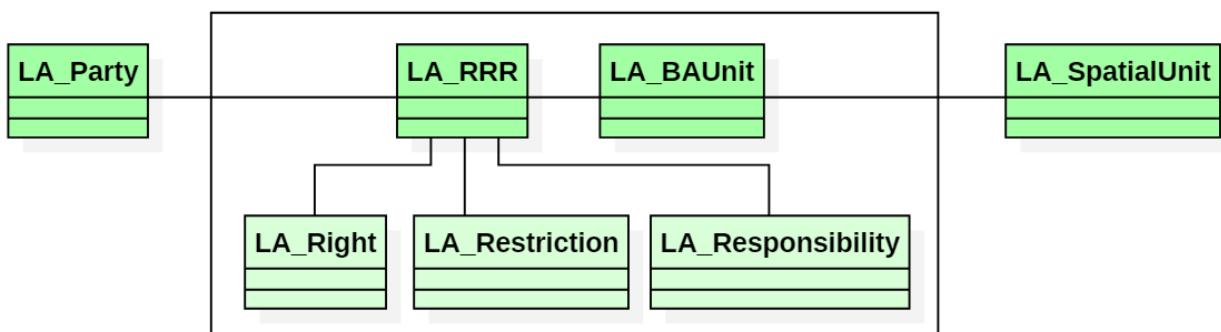


Figure 2.1: Basic structure of the LADM packages

Lemmen et al. (2015) say that the party package 'LA_Party' can be interpreted as the location in the data model for holding all information related to the people attached to the definitions of various rights, restrictions, and responsibilities related to property ownership. A party can be a person or organization that plays a role in an RRR transaction. Therefore, the party is directly linked to the RRR package. Organizations can include companies and municipalities, among others. For example, group party's can be used in cases of ownership like that of a condominium board in many jurisdictions that has multiple owners of common property (Lemmen et al., 2015).

The administrative package includes two sub-classes, 'LA_RRR' and 'LA_BAUnit'. 'LA_RRR' can be broken down further into sub-classes used for representing rights 'LA_Right', restrictions

‘LA_Restriction’, and responsibilities ‘LA_Responsibility’. Rights can include the right of ownership, tenancy, or possession, among others. Restrictions can include any example where a party is required to refrain from doing something related to the property. For example, a mortgage would be a restriction attached to the ownership of a property. Responsibilities are defined as formal or informal obligations to do something related to the property (Lemmen et al., 2015).

The spatial package ‘LA_SpatialUnit’ includes descriptions of geometric parcels that will be linked to the boundaries defining the extent of a basic administrative unit. Boundary descriptions can be topological, polygon, unstructured, point, or text based. The basic administrative unit package, ‘LA_BAUnit’ consists of zero or more spatial units that is linked to one or more of the rights, restrictions, or responsibilities defined in the ‘LA_RRR’ package. The spatial unit can be part of a spatial unit group like how a condominium unit would be part of a total condominium building (Lemmen et al., 2015).

Localized data models of the LADM have been implemented as a guide to help create a data structure used to organize legal documents associated with land administration in several jurisdictions. Some of these include models for Croatia (Vucic et al., 2013), Korea (Lee et al., 2015), Malaysia (Zulkifli et al., 2014), Poland (Gozdz & van Oosterom, 2016), The Netherlands (Kara, et al., 2019), and Victoria (Australia) (Kalantari & Kalogianni, 2018), among others. The LADM formalizes terminology that might exist in a cadastral system registering different types of 2-D and 3-D property. Some of these 3-D property types are discussed further in Section 2.3.

The literature above suggests that while administrative structures have been described in detail (first 2 packages), less research has been done regarding the analysis of boundaries represented in the ‘LA_SpatialUnit’ package that focuses on the spatial geometry types and topology enforcement components of the model. This study focuses on aspects of the spatial unit package with an emphasis on how 3-D boundaries defining spatial units can be represented and relationships between them can be analyzed digitally using computational processes.

2.3 Legal Foundations for 3-D Property Registration

This section reviews the legal foundations required for registering 3-D property. Firstly, it reviews the importance and types of rights, restrictions, and responsibilities that can be registered in 3-D. Secondly, it covers a review of possible solutions for defining and realizing 3-D property in a few different jurisdictions globally while comparing various degrees of 3-D property registration that are already in place.

This section gives the reader an idea of the various types of 3-D property that the methods proposed in Chapter 4 can be applied to and provides context as to why they would be useful when comparing their boundaries. This section addresses research question 5: “What rights, restrictions, and responsibilities can be defined and registered in 3-D?”, research question 6: “What types of cadastral spatial units can be registered using 3-D boundaries?”, and research question 7: “What are the approaches to realizing 3-D property units?”. This section contributes to research objective 1: “Identify the types of 3-D cadastral objects that can be realized in various jurisdictions”.

2.3.1 3-D Rights, Restrictions, and Responsibilities

One of the most important purposes of a cadastral system is to unambiguously record the varying RRRs associated with registered land and property. Any tenure system in which the legal status of property is ambiguously presented can have negative effects regarding the development or transfer of ownership opportunities related to that property. With 3-D property becoming more commonplace, a cadastral system must be able to manage RRRs in three dimensions to increase the robustness of land and property tenure security in its corresponding jurisdiction.

Stoter (2004) claims that a 3-D cadastre should record the geometries, ownership information, and other interests related to land and 3-D property. Among these other interests are rights, restrictions, and responsibilities (RRRs). El-Mekawy et al. (2015) say that these RRRs will be different depending on the legal system for land and property registration in a specific country or jurisdiction. Some examples of RRRs they give include the right to use a road on a property, the restriction not to erect buildings without permission, and the responsibility to keep your house in good condition. El-Mekawy et al. (2015) define four different types of RRRs that can be represented in 2-D or 3-D.

The first type of RRRs are those that are related to the underlying parcel itself. For these, a 3-D representation is not necessary as it can be linked to the 2-D property boundaries. The second type of RRRs are those that are affected by the type of property to be constructed on a parcel. For these, a 3-D representation is necessary as complex scenarios involving overlapping shared spaces are hard to represent using a 2-D representation. The third type of RRRs are those that are not clearly defined legally or are delimited. These can concern any objects or activities on the parcel in 3-D. An example of this would be a maximum building height or maximum digging depth and would usually be represented with a single elevation parameter. The fourth type of RRRs are those that are based on easements/agreements between real properties. This occurs whenever a property has the right to use another property for a specific use. In

a 3-D cadastre, this could include the overhanging of a 3-D legal object into a neighboring parcel or a 3-D object that crosses over many parcel boundaries such as an underground tunnel (El-Mekawy et al., 2015).

2.3.2 National Case Studies Regarding 3-D Property

Kitsakis et al. (2016) review how several jurisdictions are already implementing some form of a 3-D cadastral system. They highlight the legal issues that can arise when realizing 3-D property in various jurisdictions. They provide a review regarding the legal definition of 3-D objects and the possibilities for recording them, as well as reviewing what rights can be registered in 3-D in each jurisdiction. They found that there were varying degrees to which 3-D property was recorded and 3-D rights were able to be officially registered in the existing cadastral systems (Kitsakis et al., 2016).

Some jurisdictions such as Argentina, Bulgaria, Greece, and Jordan were able to record 3-D objects to some extent while not formally registering any rights in 3-D. Bulgaria had no 3-D cadastral legislation in place but recorded utilities under or above the ground and would provide 3-D visualization for certain cadastral objects. Argentina registered everything in 2-D but defined 3-D objects using various levels such as basement, floor, or roof. Greece registered 3-D objects using 2-D representations and tags representing different layers while Jordan registered 3-D objects in 2-D and had a separate register for buildings (Kitsakis et al., 2016).

Other jurisdictions such as Queensland (Australia), Victoria (Australia), Quebec (Canada), Croatia, and The Netherlands can register some form of 3-D rights in their cadastral systems. Victoria and Croatia can register various 3-D rights that are represented using 2-D diagrams. Queensland can register any 3-D right if the boundary of it can be mathematically defined and represents the boundaries using two variations of 3-D plans. In Quebec, overlapping property is shown on the 2-D cadastre using tags that refer to subdivision plans that are mainly used to register condominium units. Private and co-ownership rights can be registered in 3-D even though the concept of 3-D legal objects does not exist. The Netherlands recognizes 3-D volumetric parcels that can overlap with ground parcels and they use a 3-D PDF to represent the boundaries that are attached to the deed of the property. Sweden defines 3-D property as a unit that is delimited both horizontally and vertically and uses a 2-D representation on the cadastral map (Kitsakis et al., 2016).

Through all the case studies examined by Kitsakis et al. (2016), there are some overlaps regarding the types of 3-D objects that can be realized using 3-D boundaries. 2.5-D boundaries refer to boundaries that are delimited using a single elevation value to represent them. These include mostly boundaries related to depth and height such as digging/construction depth and airspace limitations. Full 3-D boundaries usually are

linked to volumetric parcels and can have more than a single elevation associated with them. Often, these can represent 3-D construction objects such as underground tunnels and physical buildings and can also represent volumetric legal units such as those in apartments or condominiums (Kitsakis et al., 2016). Full 3-D cadastral objects with closed boundaries are the objects considered for this study.

2.4 Surveying and Registering 3-D Property Boundaries in Alberta

This section reviews the different technical and legal concepts and standards related to surveying the different types of real property boundaries for condominium and strata units in the jurisdiction of Alberta, Canada. It covers two types of survey plans (i.e. Condominium surveys and Strata surveys) used as data sources for 3-D property boundaries and discusses how these boundaries can be defined and registered in each of them while reviewing the links between these 3-D unit boundaries and the underlying 2-D land parcel boundaries.

This section describes the foundations for the 3-D boundary definitions that the 3-D boundary datasets tested in Chapter 6 are derived from. These datasets were used to test the boundary relationship classification processes and methods that are proposed in Chapter 4. It also provides some context to the types of topological boundary validation checks that are discussed further in Section 2.5. This section addresses research question 8: “What are the types of survey plans that can be used to register 3-D property in Alberta?”, research question 9: “What types of 3-D cadastral property units can be registered in Alberta?” and research question 10: “How are 3-D cadastral property units structured within the underlying 2-D parcel in Alberta?”. This section contributes to research objective 2: “Identify 3-D data structures that are commonly used to represent 3-D property boundaries”.

In Alberta’s current cadastral system, the survey plan which includes geometric drawings with angles and distance measurements is what determines the extent of the real property’s boundary that is linked to each title of ownership. Land surveyors use registered survey plans to find existing lot pin monuments which can be used to establish new title boundaries or subdivide an existing lot even further (Surveys Act, R.S.A. 2000 Chapter S-26). For this reason, any data model used to store and represent cadastral objects in Alberta needs to be accurate and secure. Precautions need to be in place to guarantee that boundaries do not overlap when a new object is added into the system or an existing object is further subdivided.

In Alberta, a traditional 2-D land parcel follows the heaven to hell concept (also known as the infinite carrot). Land rights are split into surface rights and subsurface (mineral) rights. In most scenarios, the crown owns the mineral rights unless otherwise stated. The owner of a land parcel owns the space above the surface notwithstanding any air space or building restrictions that apply to that specific parcel (Service Alberta, 2011).

For this study, real property will be defined as including both traditional 2-D land parcels, as well as the different variations of 3-D property units that can be registered through condominium and strata survey plans. Both plan types are variations of a subdivision plan and 3-D property units are linked in some way to the underlying 2-D land parcel. Recording and registering 3-D property in Alberta is influenced by the Land Titles Act (Land Titles Act, R.S.A. 2000 Chapter L-4), the Condominium Property Act (Condominium Property Act, R.S.A. 2000 Chapter C-22), and the Law of Property Act (Law of Property Act, R.S.A. 2000 Chapter L-7). 3-D ownership rights can be registered through strata or condominium survey plans.

2.4.1 Condominium Title and Registration

A condominium plan is defined as a plan of subdivision. The Condominium Property Act (Condominium Property Act, R.S.A. 2000 Chapter C-22) governs how condominium spaces can be subdivided, what needs to be included in a condominium plan, how condominium boundaries are defined, along with other legal specifications which are related to establishing condominium titles of ownership in Alberta.

A condominium plan subdivides a parcel into at least two units and usually defines common property. The condominium area is managed by a condominium corporation in which the board members are the owners of one or more of the individual condominium units. The main purpose of this corporation is to manage the common spaces shared between all members and manage disputes between members.

When a condominium plan is registered in Alberta, the registrar cancels the certificate of title to the previously 2-D land parcel and issues a new certificate of title for each unit described in the plan. Any interests that appeared on the original 2-D parcel title (such as who owns the rights to mines and minerals) are carried over and shown on each new condominium unit title certificate. Only one unit can be included in one certificate of title along with the proportional share in common property owned by all unit holders. This share of the common property is defined by a unit factor ratio corresponding to each title. The land surveyor assigns these unit factors of common space co-ownership for each title based on the ratio of each title's owned square footage to the total square footage of the building.

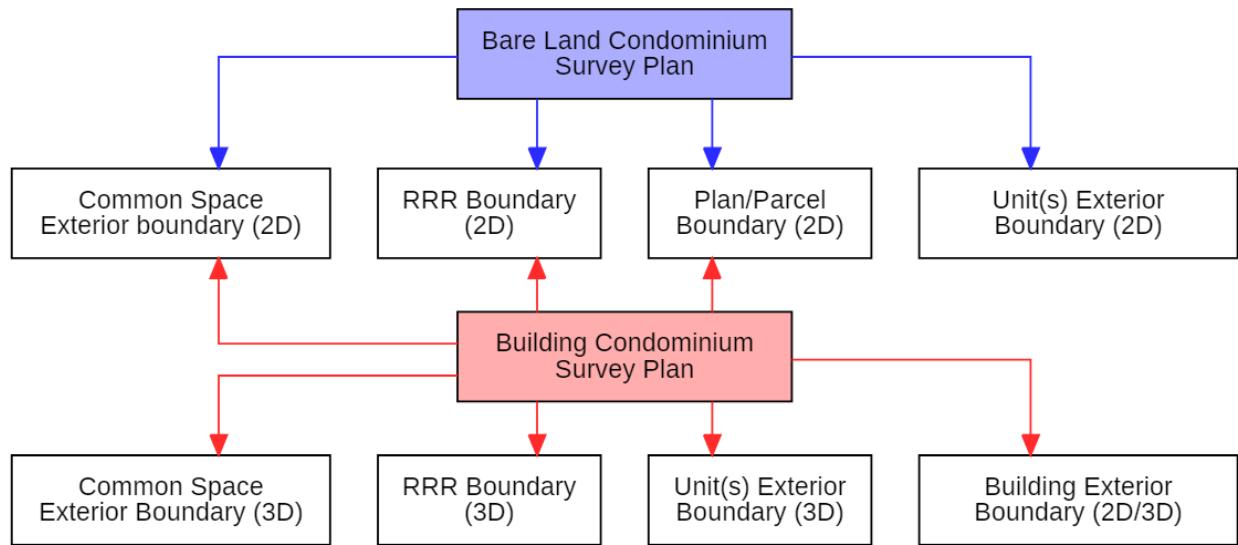


Figure 2.2: Condominium Survey Cadastral Object Boundary Types – Alberta, Canada

Figure 2.2 above shows the different 2-D and 3-D cadastral object boundaries that can be registered in Alberta, Canada using two variations of condominium survey plans. A bare-land condominium plan or a building condominium plan can be registered. Both plans require all structures, units, and common spaces to be contained within the 2-D parcel boundary (Condominium Property Act, R.S.A. 2000 Chapter C-22).

A bare-land plan would be like most other 2-D subdivision plans, aside from the definition of common space that is shared between condominium owners. The boundaries of units in bare-land condominium plans are georeferenced to the exterior of the building placed within the plan representing a unit. Common space is everything left after subtracting the individual units from the whole 2-D parcel and can be designated into various sections having varying RRRs associated with their co-ownership by all condominium co-owners (Condominium Property Act, R.S.A. 2000 Chapter C-22).

A building condominium plan requires that the exterior boundary of the condominium building is georeferenced to known ground control points. The 2-D building footprint needs to be contained within the underlying 2-D parcel. Building condominium plans can have 2-D RRR spaces such as parking stalls on the ground surface of the 2-D parcel. They can also have 3-D RRR spaces such as a locker or storage room that is designated to a specific unit. There can be 2-D common space defined on the surface of the 2-D parcel such as a common garden area or 3-D common space within the building such as utility corridors, hallways between unit entrances, or elevator shafts (Condominium Property Act, R.S.A. 2000 Chapter C-22).

The boundaries of each unit in building condominium plans are normally defined relative to the extent of the space within the unfinished interior floors, walls, and ceilings of each unit and are given a height and volume. In this scenario, coordinates cannot be georeferenced as they float somewhere within the structure of the physical building. Section 9(4) of the Condominium Property Act (Condominium Property Act, R.S.A. 2000 Chapter C-22) states that the 3-D boundary dimensions can also be defined to the centerline of the walls between adjacent units. In this scenario, 3-D boundary coordinates can be derived.

2.4.2 Strata Title and Registration

Strata space is defined in the Alberta Land Titles Act (Land Titles Act R.S.A. 2000 Chapter L-4:51) as volumetric space located above or below the earth's surface or occupied in whole or in part by any structure. It is needed whenever a single land parcel needs to be divided into more than one part defined by a change in rights associated with each part. A corporation does not need to be created to manage group interests and the strata lots can be owned by a single entity. The locations where a change in rights occurs are used to define the 3-D boundaries shown in the strata plan (Land Titles Act R.S.A. 2000 Chapter L-4).

An example of where strata registration is appropriate would be for a mixed-use development involving a parkade and commercial offices. A strata plan can also include a condo plan as part of the development depending on the situation and would require all condominium units and shared common property to be contained within the strata unit boundaries. Before construction can begin on the development, the physical extent of where each area will reside needs to be established. The boundaries of where the spaces meet can be defined by a plan and are surveyed for confirmation once the structures are built.

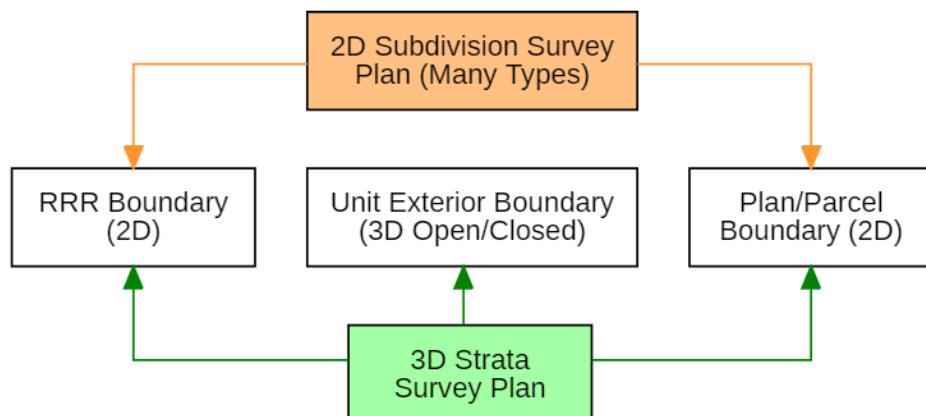


Figure 2.3: Strata Survey Cadastral Object Boundary Types – Alberta, Canada

Figure 2.3 above shows the different 2-D and 3-D cadastral object boundaries that can be registered in Alberta, Canada using a regular subdivision plan and a strata survey plan. A 2-D subdivision plan includes the 2-D parcel boundaries and can include additional 2-D RRR boundaries that must be contained within the parcel boundary. A 3-D strata plan subdivides a parcel into at least two 3-D units which can have open boundaries extending to infinity in the vertical or can be contained as a volumetric unit with a closed boundary. Each unit defined gets a separate title of ownership. Strata boundaries are defined by georeferenced coordinates that are referenced to a known boundary pin and network (Land Titles Act R.S.A. 2000 Chapter L-4).

This study identifies relationships between modelled 3-D objects that have closed boundaries. This refers to 3-D objects that do not extend upwards or downwards to infinity such as those that can exist in 3-D strata plans but rather are limited in height and are contained by a set of closed boundary planes. Some strata units have boundaries that extend upwards or downwards to infinity. The methods put forward in this study do not apply to them. The boundaries are assumed to be georeferenced meaning that condominium plans where unit boundaries are defined relative to interior building walls would not apply. The methods would still apply to centerline condominium boundary units where boundary coordinates can be derived.

2.5 Approaches to Digitally Representing 3-D Boundaries

This section reviews approaches to representing and analyzing 3-D boundaries in a digital system and is presented in three sub-sections. Firstly, three modelling approaches are introduced which highlight different degrees that 3-D boundaries can be referenced in a spatial database with existing 2-D land parcel boundaries. One of these is chosen as the premise for which the methods presented in Chapter 4 would work under. Secondly, spatial data formats that can be used to store and index 3-D boundaries in a digital system are reviewed. The approach used in this study to digitally construct 3-D boundaries is introduced. Lastly, 3-D boundary relationships that could be implemented between spatial objects in survey plans are reviewed. This includes validation checks between boundaries that could be completed before survey plan registration as well as during post-registration analysis.

This section introduces some approaches to modelling 3-D boundary data and provides context into some of the reasons being able to classify relationships between 3-D boundaries would be useful to those in the surveying profession. This section addresses research question 11: “What approaches exist for integrating 2-D and 3-D cadastral boundary data into a DBMS?”, research question 12: “What data formats can be used to represent 3-D boundaries digitally?”, research question 13: “What 3-D boundary components should be considered for modelling?”, and research question 14: “What data checks performed on 3-D boundaries

would be useful for survey plan validation?”. This section contributes to research objective 3: “Decide on a data format that will be used for digitally representing 3-D boundaries”.

2.5.1 Approaches to Modelling 2-D and 3-D Boundaries

Stoter, (2004) has identified three different ways in which 3-D property can be represented with existing 2-D property data in a cadastral system. Each method has drawbacks regarding varying amounts of analysis capabilities, storage costs, and implementation costs. They are presented in order from the least to most functional regarding the information a user can get from the implemented database. The main advantages and limitations of each are discussed.

The first approach applies 3-D tags representing various 3-D structures and rights to existing 2-D land parcels. This approach is the easiest to implement in an existing 2-D database schema and requires the least amount of additional storage. However, it produces the solution with the smallest amount of operability towards searching and analyzing boundaries within the system. Queries cannot be performed on 3-D objects crossing multiple 2-D parcels. 3-D boundaries contained within the same 2-D parcel cannot be analyzed together as it is only 3-D object ID tags that can be searched (Stoter, 2004).

The second approach uses a mixed combination of both 2-D and 3-D parcels. Most implementations of 3-D cadastral systems to date use some version of this. One of the main advantages of this approach is that most existing parcels in the system that are 2-D do not need to be altered in any way. Only property with 3-D boundaries are represented in 3-D and these boundaries can have various degrees of complexity. The main disadvantage to this approach like that of using 3-D tags is that it requires additional methods to analyze 2-D and 3-D boundaries together as they are in different storage formats. A homogeneous data format is required to relate and compare the two types of boundaries (Stoter, 2004).

The third approach requires all land and 3-D property in the cadastral system to be fully represented in 3-D as volumetric parcels. The advantage of this solution is that it will ensure that all boundaries are represented in the same homogeneous 3-D format. This will produce a solution that allows for the highest amount of functional operability regarding the analysis of boundaries. The disadvantages of this solution are that it requires converting existing 2-D parcels into fully 3-D units which can be costly and will require the most amount of additional storage and processing power while searching the system (Stoter, 2004).

This study runs under the premise that the second approach mentioned above will be used. This means that 3-D property registered in a cadastral system would have 3-D volumetric units that are contained within an existing 2-D parcel. If a 3-D construction object overlaps an adjacent 2-D parcel it will be registered to that

adjacent parcel through a right of way or easement. This approach follows the structure of how both condominium and strata survey plans can be used to register 3-D property in Alberta, Canada. Therefore, the processes proposed in Chapter 4 that are used for determining relationships between 3-D units will apply to 3-D cadastral objects that are registered within the same survey plan.

2.5.2 Spatial Data Formats and Representation

While there have been studies related to capturing 3-D physical objects using LiDAR and point clouds, they have challenges associated with them that make those formats undesirable for representing legal boundaries. Koeva & Oude Elberink (2016) review some of the challenges associated with updating 3-D cadastral objects using LiDAR and point clouds. Among these challenges are uncertainties that come from inaccuracies in the equipment used as well as environmental factors that can alter the data such as areas of shade or vegetation in the scene. Point clouds can often have missing data points depending on the viewing angle that the data was collected and there are often outlier points collected.

Ying et al. (2015) proposed an approach that can be used to digitally construct 3-D volumetric objects for a 3-D cadastral system. The approach describes how to directly generate volumetric objects using polyhedron plane surfaces with outwards and inwards orientations defined that when combined form the boundary of the legal object. Their method utilizes the existing surveyed boundaries from plans while representing volumetric objects with high quality, consistent topology, and produces no accidental intersections. A simplified description of the main boundary components that are used to form volumetric cadastral objects in this model is shown in Figure 2.4 below.

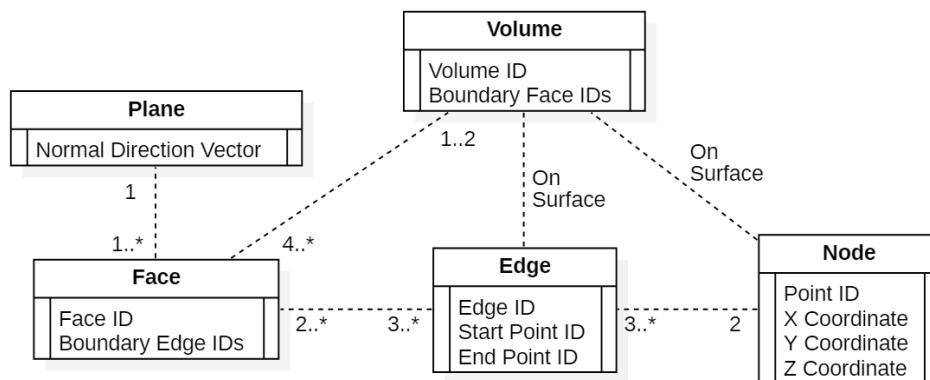


Figure 2.4: General Structure of Constructed 3-D Volumetric Objects (Derived from Ying et al., 2015)

The main geometric primitives (boundary components) that are combined to make volumetric cadastral objects in this model are node, edge, and face. A node has a 3-D [X, Y, Z] coordinate that can be generated by searching through survey plan measurements. An edge is defined by 2 nodes and edges only meet other edges at a common node. A face is a closed polygon defined by a list of 3 or more consecutive edges. The face has a normal direction vector defining a plane that is used to represent the inside or outside face of the volumetric boundary. A volumetric boundary is closed by at least 4 boundary faces with indicated normal plane direction vectors. These primitives are topologically related in a dimensional hierarchy where nodes form the boundary of edges, edges form the boundary of faces, and faces with normal directions form the boundary of volumes. This structure ensures topological consistency between the components of the volumetric object while not relying on a solid definition.

The 3-D objects considered in this study are volumetric objects with closed flat surfaces that do not have any holes through them. The surveyed objects that this study deals with are derived using vector-based datasets. The legal boundaries of these objects are surveyed and represented using 3-D point coordinate data, lines derived from angles and distances, and polylines that are combined to create flat planes. For this reason, and to reflect the accuracy of cadastral boundaries recorded in condominium and strata survey plans a vector-based storage method is used like that put forward by Ying et al. (2015). This implementation is discussed further in Section 4.1.

2.5.3 Relationship Validations between 3-D Boundaries

Shojaei et al. (2017) developed data validation rules to ensure the accuracy of new and updated data within a 3-D cadastre for the land registry in Victoria, Australia. This was done to expand their ‘eplan validation service’ from 2-D plans to 3-D plans. This research focused on developing geometrical validation rules, non-geometric (semantic) validation rules, and on implementing an online service that could validate 3-D cadastral boundary data. Four main spatial checks were implemented involving volume comparisons, intersections (clash) between objects, non-flat-face detection of an object, and watertight conditions for volumes.

Gulliver et al. (2016) reviewed theoretical concepts related to the development of New Zealand’s cadastral survey system to move towards a 3-D digital cadastre by 2021. Among these considerations were validations of spatial objects in the system. They claim that validations are needed to ensure that the data being inputted into the system complies with the rules local to the jurisdiction they are registered in. Out of

the four rules that were suggested, three directly included data related to 3-D parcels. The first was that the objects registered are correctly formed in shape and meet the definition of a spatial object. This could include validation checks such as 3-D units needing to be closed volumes having no holes. The second was that 3-D objects should be contained within the primary 2-D parcel it is registered to. The third was that the 3-D object should be checked for collisions with other spatial objects. This third recommendation is what this study aims to contribute towards.

Jaljolie et al. (2018) outlined recommendations for implementing a 3-D land management system in Israel related to processes that would be required to validate 3-D survey plans. Among the identified processes was the insertion of a new 3-D object into the system. Some of the functions required for this include validating if the 3-D object is a valid object similar to that of Gulliver et al. (2016), is it a safe distance from other objects (buffer), and does the object intersect any other existing 2-D or 3-D object in the database. These validations would require determining spatial intersection, spatial overlap, and spatial buffer checks between two objects.

Considerations from the literature presented above suggest that validations for the boundaries of 3-D cadastral units can be useful for managing 3-D property modelled in a digital system. The main contribution that this study aims to make is towards classifying relationships that exist between the boundary sub-components (i.e. point coordinates, line edges, and planar faces) of two 3-D cadastral units. Verifying these relationships could be used to help a surveyor ensure that there are no intersecting (slices) unit boundaries or gaps (gores) between units that should be adjacent after surveying and before registering a survey plan. By considering the distances between various components, a buffer function could be implemented as well. By comparing the relationship classifications between boundary components belonging to a single 3-D object, the object could be validated as having the correct format such as forming a closed volumetric solid with watertight surfaces that do not intersect each other. It could also be useful in determining if point or line sub-boundary facets are collinear or coplanar with boundary planes as they should be.

2.6 Chapter Summary

The relevance this chapter has to the primary research objective is that it reviewed literature related to 3-D cadastre research and development that was required to define 3-D cadastral objects and their boundary components as they are used here. This chapter addressed guiding research questions 1 – 14. This chapter was used to perform research activity #1: “Identify the types of 3-D cadastral objects that can be realized in Alberta, Canada and in various legal jurisdictions” and research activity #2: “Identify 3-D data structures that are commonly used to represent 3-D property boundaries”. The 3-D cadastral units modelled in this study were derived from closed boundary 3-D strata units or 3-D centerline condominium units in Alberta, Canada that could be registered through strata or condominium survey plans, respectively. The data storage structure used in this study to model 3-D boundaries associated with these 3-D cadastral units uses 3-D boundary points, boundary lines, and boundary planes.

Section 2.1 described the purposes of a 3-D cadastral system and briefly discussed the users that could benefit from using a 3-D cadastral system. Developing a 3-D cadastral system would mostly benefit those working in the 3-D land management and professional surveying field.

Section 2.2 introduced the different components of a 3-D cadastral system with respect to the LADM descriptive standard. This study focuses on problems associated with the spatial unit component, specifically on how 3-D boundaries can be modelled, and relationships between 3-D spatial unit boundaries can be classified.

Section 2.3 reviewed different RRRs that can be defined and registered to 3-D spatial units and discussed approaches to defining and realizing 3-D cadastral units in various legal jurisdictions. This was presented to highlight that there are many ways in which 3-D cadastral property can be realized in different jurisdictions, although many use similar definitions for how 3-D boundaries can be recorded.

Section 2.4 introduced two types of survey plans (condominium and strata) that can be used to register 3-D spatial units in Alberta, Canada. The definitions regarding 3-D spatial units and how they are structured with respect to the underlying 2-D parcel were discussed here.

Section 2.5 reviewed approaches for digitally representing 3-D boundaries in a DBMS. Possible boundary representation definitions and their associated data formats for modelling them, along with potential data checks that could be useful for performing survey plan validations were reviewed here.

Chapter Three: Literature Review related to 3-D Boundary Relationships and Mathematical Theory

This chapter presents existing theory and methods related to topological relationship modelling and mathematical concepts that were used to design and implement the computational modelling processes presented in Chapter 4. It is structured as follows:

Section 3.1 reviews literature related to organizing and classifying topological relationships between geometric objects being modelled in 3-D space. This section introduces the Dimensional Model (DM), which is the guiding framework used to develop the methodology in Chapter 4. Projected relationship concepts such as 3-D objects being parallel, collinear, or coplanar, as well as topological descriptors such as 3-D objects being ‘disjoint’, ‘touching’, and ‘intersecting’ at different dimensionalities are introduced here. Different approaches to modelling relationships between 3-D boundary data are reviewed here as well.

Section 3.2 introduces important mathematical concepts related to Conformal Geometric Algebra that are used in the methodology proposed in Chapter 4. The structure of Geometric Algebra and the Conformal model is presented here. Equations, objects, and topological processes are identified here and previous research that uses these concepts to model topological relationships between cadastral objects is discussed.

3.1 Literature Review on Topology Frameworks and Relationship Modelling

This section is included for the following purposes. Section 3.1.1 introduces the reader to different types of topological relationships that can exist between spatial objects. It introduces and compares the 4-intersection, 9-intersection, dimension extended, and DE+9-intersection topological frameworks that can be used to differentiate and organize a range of these relationships.

Section 3.1.2 links the relationships from Section 3.1.1 to the geometric objects (points, lines, planes) that are used to represent 3-D cadastral object boundaries in this study. It shows how the 9-intersection model has been applied to 3-D objects and introduces the Dimensional Model framework used to distinguish relationships between objects in 3-D space.

Section 3.1.3 reviews different approaches to implementing topological analysis between 3-D objects. Projected relationships between 3-D objects are introduced here and an approach to processing 3-D data to classify ‘Meet’ and ‘Overlap’ relationships between 3-D planar polygons is presented.

3.1.1 Foundations for Describing Topological Relationships Between Spatial Objects

This section introduces the reader to and compares the effectiveness of different frameworks for distinguishing topological relationships that can exist between spatial objects. Firstly, it introduces the 4-intersection model (4IM) framework that identifies 8 unique topological relationships that can exist between spatial objects. Secondly, it introduces the 9-intersection model (9IM) that was developed to expand the functionality of the 4IM between spatial objects rendered in 2-D space. Thirdly, it introduces the dimension extended method (DEM) framework which is an extension of the 4IM. Lastly, it introduces the dimensionally extended 9-intersection model (DE+9IM) and highlights the advantages it has compared to the 9IM.

This section addresses research question 15: “What topological relationships can exist between spatial objects?”, and research question 16: “What are the 4-intersection, 9-intersection, dimension extended, and dimension extended 9-intersection topological model frameworks, and how are they used to describe relationships?”.

Egenhofer (1989) created the 4-Intersection Model (4IM) as an approach for describing relationships between two spatial objects having the same dimension. According to this model, the relationships described are characterized as being invariant under topological transformations such as translations, scaling, and rotations. The model was originally derived from 8 topological relationships that can exist between closed interval sets in a 1-dimensional space. Closed interval sets are spatial objects such as lines and planes that have boundary limits. The closed interval sets considered in the 4IM are the spatial object’s boundary (represented by ∂) and interior (represented by $^\circ$). The boundary is defined as the part of the spatial object that contains the interior. The topological relationships of the 4IM are described in Table 3.1 below using the four intersection results between their boundary (∂) and interior ($^\circ$) sets. Visualizations for a subset of these can be seen in Figure 3.4 below.

Table 3.1: Eight Topological Relationships and the 4IM (Derived from Egenhofer, 1989)

Eight Topological Relationships and the 4-Intersection Model				
Topological Relationship R(A,B)	$\partial A \cap \partial B$	$A^\circ \cap B^\circ$	$\partial A \cap B^\circ$	$A^\circ \cap \partial B$
Disjoint	\emptyset	\emptyset	\emptyset	\emptyset
	$\neq \emptyset$	\emptyset	\emptyset	\emptyset
	\emptyset	$\neq \emptyset$	$\neq \emptyset$	$\neq \emptyset$
	\emptyset	$\neq \emptyset$	$\neq \emptyset$	\emptyset
	\emptyset	$\neq \emptyset$	\emptyset	$\neq \emptyset$
	$\neq \emptyset$	$\neq \emptyset$	\emptyset	$\neq \emptyset$
	$\neq \emptyset$	$\neq \emptyset$	$\neq \emptyset$	\emptyset
	$\neq \emptyset$	$\neq \emptyset$	\emptyset	\emptyset
where		$\partial = \text{Boundary}$	$\emptyset = \text{Empty Set}$	
		$^\circ = \text{Interior}$	$\neq \emptyset = \text{Non-Empty Set}$	
		$\cap = \text{Intersection}$		

Using the 4IM, the eight relationships can be differentiated by evaluating four intersections between the two boundaries ($\partial A \cap \partial B$), the two interiors ($A^\circ \cap B^\circ$), the boundary with the interior ($\partial A \cap B^\circ$), and the interior with the boundary ($A^\circ \cap \partial B$) of two spatial objects A and B. The four intersections can result in empty (\emptyset) and non-empty ($\neq \emptyset$) sets which are interpreted together to distinguish which of the eight relationships is occurring between the two objects being considered. The relationship description considers the order of the two objects being evaluated and can produce symmetric relationships such as disjoint, meet, overlap, and equal, as well as converse relationships such as inside/contains or covers/covered by. (Egenhofer, 1989).

Egenhofer & Herring (1990) created the 9-intersection model (9IM) by considering intersections between the exteriors of both spatial objects along with the intersections between interiors and boundaries that are included in the 4IM. This model extends the functionality of the 4IM to work between point, line, and region cell complexes in 2-D space. In the 9IM, a point is considered a 0-cell, a line is considered a 1-cell connecting two 0-cell points, and a region is considered a 2-cell area consisting of at least three non-intersecting 1-cell lines (Egenhofer & Herring, 1990:4). To describe the topological relationship between two cell objects (A and B), the nine intersection results between their interiors (A°, B°), boundaries ($\partial A, \partial B$) and exteriors (A^-, B^-) are evaluated and interpreted. These nine intersections can be seen in Table 3.2 below.

Table 3.2: The 9-Intersection Model Matrix (Derived from Egenhofer and Herring, 1990)

The 9-Intersection Model Matrix			
Topological Relationship R(A,B)	$A^\circ \cap B^\circ$	$A^\circ \cap \partial B$	$A^\circ \cap B^-$
	$\partial A \cap B^\circ$	$\partial A \cap \partial B$	$\partial A \cap B^-$
	$A^- \cap B^\circ$	$A^- \cap \partial B$	$A^- \cap B^-$
where	$\circ =$ Interior	$\cap =$ Intersection	
	$\partial =$ Boundary		
	$- =$ Exterior		

Like that of the 4IM, the results for each of the nine intersections result in empty (\emptyset) and non-empty ($\neq \emptyset$) sets that are interpreted to describe topological relationships between spatial objects. While both the 4IM and the 9IM can be used to describe the same eight topological relationships introduced in Table 3.1, the 9IM has the advantage that it can describe relationships between objects that are embedded into a higher dimensional space (Egenhofer and Herring, 1990:21).

Egenhofer et al. (1993) compared the 4IM and the 9IM to highlight the advantages that the 9IM has over the 4IM. It was found that when the two objects considered have the same dimension, the 4IM and the 9IM were both able to distinguish eight relationships for both line-line and region-region sets. However, when the objects were embedded into a higher-dimensional space, the 4IM was not able to determine the ‘equal’ relationship between two lines. When the objects being considered differed by a single dimension (i.e. relationships between 1-D lines and 2-D regions), the 4IM was able to distinguish 48 relationships while the 9IM was able to distinguish 74 relationships (Egenhofer et al., 1993).

Clementini et al. (1993) introduced the dimension extended (DE) method, which is a modification of the 4IM. When evaluating point, line, and region spatial objects, six groups of binary relationships can be evaluated being area/area, line/area, point/area, line/line, point/line, and point/point. One of the drawbacks of the 4IM is that some relationships that are considered different are classified as the same relationship. Clementini et al. (1993) give the example of two areas that have a point in common versus two areas that have a line in common that will both produce the same touch relationship (see discussion related to Figure 3.3 below). The dimension extended model fixes this problem by considering the dimension of the intersections instead of only if they are empty or non-empty. If the common geometry is a point with no lines or areas the intersection is a 0-D point. If the common geometry is a line with no areas the intersection is a 1-D line. If the common geometry is an area the intersection is a 2-D plane (Clementini et al., 1993).

Clementini & Di Felice (1995) applied the dimension extended method to the 9IM (DE+9IM) and compared it to the three previously mentioned approaches. The DE+9IM considers the dimension of the intersections between interiors, boundaries, and exteriors to increase the number of unique relationships that can be identified. Table 3.3 below summarizes the results from this comparative study between the 4IM, DEM, 9IM, and the DE+9IM (Clementini & Di Felice, 1995).

Table 3.3: Comparison of 4IM, 9IM DEM, and DE+9IM (Derived from Clementini & Di Felice, 1995)

A Summary of Topological Cases							
Method	Area-Area	Line-Area	Point-Area	Line-Line	Point-Line	Point-Point	Total
4IM	6	11	3	12	3	2	37
9IM	6	19	3	23	3	2	56
DEM	9	17	3	18	3	2	52
DE+9IM	9	31	3	33	3	2	81

Table 3.3 shows that the same number of relationships involving points (i.e. Point/Point, Point/Line, and Point/Area) can be identified using all four approaches. Both the DEM and the DE+9IM can identify three additional relationships between two areas than their original counterparts. The DEM modification results in an additional 15 unique relationships that can be identified when compared to the 4IM while the DE+9IM modification results in an additional 25 unique relationships that can be identified when compared to the 9IM (Clementini and Di Felice, 1995).

To summarize, this section identified topological relationships that this study aims to classify between 3-D cadastral boundary components using the methods put forward in Chapter 4. It discussed four approaches to realizing these relationships alternate to the processes implemented here and reviews their benefits and limitations.

These four approaches show that considering the dimension of both the spatial objects being considered, as well as the dimension of the intersections between their interiors, boundaries, and exteriors is beneficial to describing topological relationships between points, lines, and areas. These frameworks were initially created for identifying relationships in 2-D space and have issues when being applied to objects in 3-D space. The relationships modelled in this study are limited to 3-D boundary components touching, intersecting, or being disjoint from each other. This study considers the dimension of the intersection

occurring, like the DEM and the DE+9IM approaches presented here. Section 3.1.2 considers approaches to organizing and identifying these relationships in 3-D space.

3.1.2 Organizing Topological Relationships between Geometric Objects in 3-D Space

This section reviews frameworks that can be used to describe topological relationships between objects in 3-D space. Firstly, it identifies standard requirements for identifying topological relationships within 3-D Geographical Information System (GIS) applications. Secondly, it covers a study that derives possible relationships that can exist between multidimensional objects by extending the 9-intersection model into 3-D space. Thirdly, it covers the Dimensional Model framework that can be used as an alternative to the 9IM to organize spatial relationships for 3-D objects and their boundary components.

This section addresses research question 17: “What frameworks can be used to describe geometric relationships between 3-D objects?”, research question 18: “How can the 9-Intersection Model be applied to distinguish relationships between multidimensional objects in 3-D space?”, and research question 19: “What is the Dimensional Model framework as it is used for organizing relationships between spatial objects?”.

Ellul & Haklay (2006) reviewed several research articles to identify a general list of common requirements for topology in multiple 3-D GIS applications, cadastral modelling being one of them. Core standard analysis requirements for 3-D object relationships were grouped into terms of object adjacency, intersection, connectivity, containment, and disconnectedness analysis (Ellul & Haklay, 2006:165). Each topological element should also have some form of geometrical representation so that the analysis of results can be reconstructed and visualized in a front-end system (Ellul & Haklay, 2006). A graphical representation of the relationships modelled in this study is presented in Figure 6.1 and Figure 6.2 (see Section 6.1).

Ellul & Haklay (2006) note that while topological frameworks need to be able to examine relationships between 3-D objects, to be comprehensive they also need to examine topological relationships between lower-dimensional (0-D, 1-D, and 2-D) components that might be part of a 3-D object (i.e. points, lines, and planes). They claim that considering the dimension of the common part between two objects allows for greater differentiation between relationships (Ellul & Haklay, 2006:160). This consideration is like the approaches followed by the DEM and the DE+9IM frameworks introduced in Section 3.1.1 above and the Dimensional Model framework discussed below.

Zlatanova (2000) investigated relationships that could exist between objects in 1-D, 2-D, and 3-D space by applying the concepts of the 9-intersection model to 0-D points, 1-D lines, 2-D planes, and 3-D volumetric bodies. They composed a set of negative conditions which limited possibilities using concepts related to the (0-D, 1-D, or 2-D) dimension of the objects being considered and the difference in dimensions that exist between them. These negative conditions were used to eliminate relationships that are considered impossible. This was done to reduce the number of possible relationships that can exist between spatial objects. The remaining relationships in 3-D space can be seen in Table 3.4 below.

Table 3.4: Relationships in 3-D Space using the 9-Intersection Model (Derived from Zlatanova, 2000)

Possible Relationships in 3-D Space using 9-IM	
Point/Point	2
Point/Line	3
Point/Surface	3
Point/Body	3
Line/Line	33
Line/Surface	31
Line/Body	19
Surface/Surface	38
Surface/Body	19
Body/Body	8
Total Relationships	159

Table 3.4 shows that possible relationships between two points in 3-D space were reduced to two, being ‘disjoint’ and ‘equal’. There are three possible relationships each between point/line, point/surface, and point/body sets, being ‘Disjoint’, ‘Covers’ when the point is on the boundary of the other object, and ‘Contains’ when the point is within the boundary of the other object.

While there are many unique relationships between sets involving lines and surfaces, it is important to note that complex spatial objects were included in the study. Complex objects are defined as surfaces with multiple faces and lines with multiple vertexes (i.e. not simple straight lines or flat surfaces). It does not imply that complex numbers define these. The number of total relationships is reduced to 69 when only simple spatial objects are considered. There are only eight relationships that can be distinguished between

two 3-D bodies using this 3-D extension of the 9-intersection model. These are the same as the eight relationships introduced in Section 3.1.1 above.

The Dimensional Model (DM) is a conceptual topological framework introduced by Billen et al. (2002) to provide insight on relationships that can exist between spatial objects. Before the model is introduced, definitions for closed sets and ordered points are described. Billen et al. (2002) state that the 3-D Euclidean vector space R^3 is the topological space in which 3-D geometry can be formally studied. Within this space, there can exist an affine subspace A defined by a distinct x,y,z point belonging to A .

Points, straight lines, planes, and R^3 are the only affine subspace of R^3 having the dimensions 0, 1, 2, and 3, respectively (Billen et al., 2002:286). A closed set C in R^3 consists of points that have an order of one of these dimensions. A hyperplane in R^3 is an affine 2-D subspace (plane) that divides the whole space into two regions. The order of a point x that is part of a closed set C is defined as the dimension of the intersection of all supporting hyperplanes containing x . A summary of ordered points that can exist on or within the boundary of a 3-D closed volumetric object consisting of planar surfaces can be seen in Figure 3.1 below.

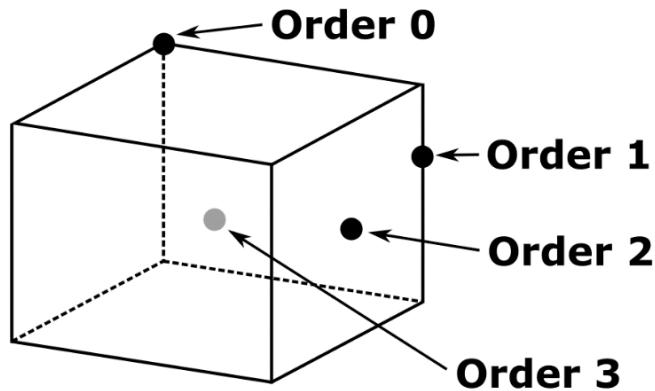


Figure 3.1: Description of Ordered Points with Respect to the DM (Derived from Billen et al., 2002)

Order 0 points refer to points that are on the edge of multiple boundary lines and exist at the 0-D intersection of three 2-D hyperplanes. Order 1 points refer to points that are within the end points of a boundary line and exist at the 1-D intersection of two 2-D hyperplanes. Order 2 points refer to points that exist on a planar surface within the closed boundary line edges and exist at the 2-D intersection of a single 2-D hyperplane. Order 3 points refer to points that are contained within the boundary of the closed 3-D set and exist at the 3-D intersection of no 2-D hyperplanes in R^3 .

The Dimensional Model is presented using the concepts of dimensional elements and dimensional relationships. Dimensional elements are organized as extensions and limits of dimensionally ordered boundary elements. Figure 3.2 below shows the dimensional spatial elements used in this study that can exist in 3-D Euclidean \mathbf{R}^3 as a composition of their extension and boundary limit elements. Extensions are bolded while boundaries (limits) are not (Billen et al., 2002).

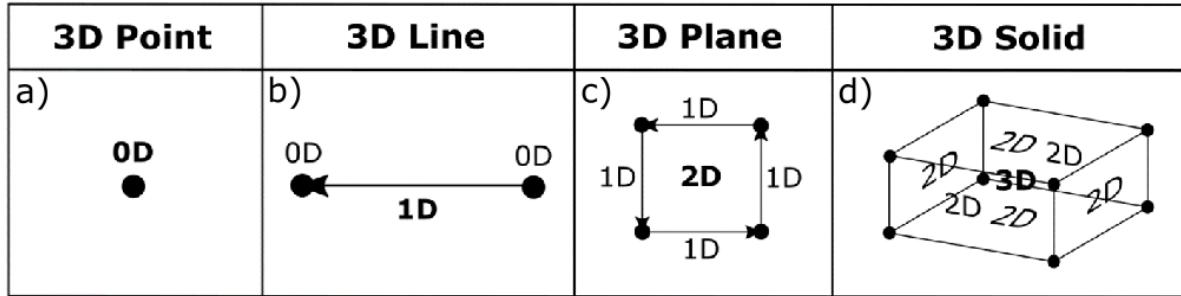


Figure 3.2: Cadastral Elements with Extensions and Boundaries with Respect to the DM (Derived from Billen et al., 2002)

Figure 3.2a shows a 3-D point dimensional element with a 0-D order extension. Figure 3.2b shows a 3-D line dimensional element with a 1-D order extension and two 0-D order boundary points. Figure 3.2c shows a 3-D plane dimensional element with a 2-D order extension, four 1-D order boundary lines, and four 0-D order boundary points. Figure 3.2d shows a 3-D solid dimensional element with a 3-D order extension, six 2-D order boundary planes, twelve 1-D order boundary lines, and eight 0-D order boundary points.

The DM classifies the spatial relationship between two 3-D objects by combining the relationships that exist between their (0D, 1D, 2D, 3D) dimensional elements. Relationships are organized into three possible categories being total relation, partial relation, and no relation. A total relation exists when the intersection between their extensions is not empty and is equal to the first element. A partial relation exists when the intersection between their extensions is not empty and is not equal to the first element. No relation exists when the intersection between their extensions is empty (Billen et al., 2002:291).

The approach to describing the existing relationship between two spatial objects is to determine the relationships between each of the dimensional elements of both objects. Each of these dimensional relationships is represented by R#D#, where R# refers to the dimension of the first element and D# refers to the dimension on the second element being considered. For the basic relationship format, each of these relationships can produce a value of 0, 1, or 2 representing no relation, a total relation, or a partial relation between dimensional elements, respectively. An extended format adds a possible value of 3 to the result if the intersection between two elements is inferior to the lowest dimension of both elements being considered.

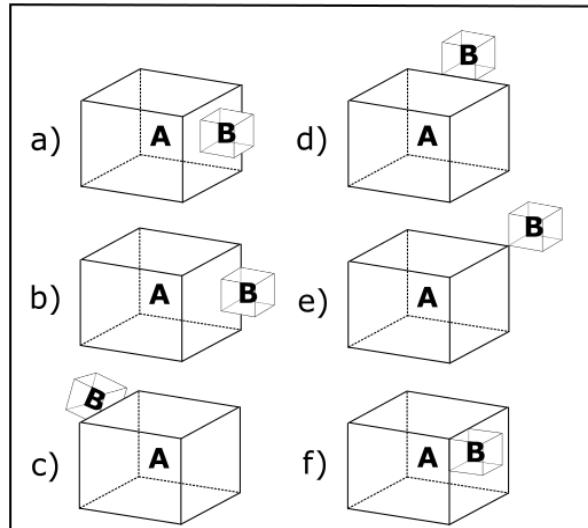
The example used by Billen et al. (2002) to explain this was if the intersection between a 2-D element and a 1-D element had an 0-D intersection, the resulting value would be a 3 instead of a 2 that would be used in the basic format. This distinction allows for a greater number of relationships to be uniquely identified.

Table 3.5: Relationships According to the Dimensional Model (Derived from Billen et al. 2002)

Possible Relationships According to the Dimensional Model							
Dimensional Element	Dimensional Relationship	Line/ Line	Line/ Surface	Line/ Body	Surface/ Surface	Surface/ Body	Body/ Body
(n)D	Basic	5	3	3	5	3	5
	Extended	7	5	3	11	3	5
(n)D & (n-1)D	Basic	33	31	19	43	19	8
	Extended	61	?	43	?	48	15
(n)D & (n-1)D & (n-2)D	Basic		?	?	?	?	?
	Extended		?	?	?	?	?
(n)D & (n-1)D & (n-2)D & (n-3)D	Basic			?		?	?
	Extended			?		?	?

A summary of basic and extended relationships that can be differentiated between 1-D lines, 2-D surfaces, and 3-D bodies using the Dimensional Model framework can be seen in Table 3.5 above. The number of relationships highlighted in grey is the same as the number of relationships that were derived from Zlatanova (2000) for the 9-intersection model above. It can be seen in row 4 of Table 3.5 that by using the extended formats, a larger number of relationships can be distinguished using the Dimensional Model framework than by using the 9-Intersection model. Results with a question mark in the last five rows of Table 3.5 were not determined in the study by Billen et al. (2002) as this study only considered the highest and second-highest dimensional elements. It can be interpreted that a larger number of relationships can be distinguished using the Dimensional Model framework by considering more dimensional elements.

Billen et al. (2002) compare the Dimensional Model to the 9-intersection model that was discussed in Section 3.1.1. While the 9-intersection model uses intersections between interior, boundary, and exterior topological primitives, the Dimensional Model uses intersections between multiple dimensional elements. This allows the Dimensional Model to represent spatial relationships at different levels of complexity based on the dimensional order of the elements considered.



	R3D3	R3D2	R3D1	R2D3	R2D2	R2D1	R1D3	R1D2	R1D1
a)	0	0	0	0	2	2	0	0	0
b)	0	0	0	0	2	2	0	0	2
c)	0	0	0	0	0	0	0	2	2
d)	0	0	0	0	0	0	0	0	2

Figure 3.3: Meet Relationship Distinctions using the Dimensional Model (Derived from Billen et al. 2002)

Billen et al. (2002) note that one area that the Dimensional Model is better at differentiating relationships is when considering ‘meet’ relationships between two 3-D (body) objects. Figure 3.3 above shows six variations of the ‘Meet’ relationship between two 3-D body objects. Under the 9-intersection model, they would all be recognized as the same relationship but under the Dimensional Model framework, they are organized as six different relationships as can be seen from the differences in results between their dimensional element relationships. Billen et al. (2002) suggest that elements and relationships could be computed using algorithms that apply collinearity and coplanarity concepts (Billen et al., 2002).

To summarize, this section linked the fundamental topological concepts from Section 3.1.1 for 2-D space to how they can be applied to 3-D boundaries as they could exist in a 3-D cadastral system. The literature shows that while the 9-Intersection model can be applied to 3-D objects, the Dimensional Model framework can be used to categorize more specialized relationships. This study follows an approach like the Dimensional Model in that relationships are first classified between point, line, and plane 3-D boundary components to describe the relationships that are occurring between two 3-D cadastral units.

3.1.3 Approaches to Implementing Topological Analysis between 3-D Cadastral Objects

This section reviews approaches to organizing relationships between 3-D cadastral objects focusing more on geometric concepts than topological concepts. These geometric concepts are linked back to the topological relationships discussed in previous sections.

Firstly, it presents a study that identifies many ‘Disjoint’ and ‘Touch’ relationships that can exist between two 3-D cadastral parcels that differ based on projected relationships between 3-D boundary components of both parcels. Secondly, it reviews a study that identifies the possible projected relationships between point, line, and plane objects in 3-D space, such as objects being collinear, coplanar, intersecting or being parallel to one another. Lastly, it presents an approach to classifying topological relationships using data process flows that evaluate projected relationships between sets of planar polygons to simplify the relationship classification problem.

This section addresses research question 20: “How can geometric relationships be used to distinguish between different types of spatial relationships in 3-D?”, research question 21: “What projected relationships can exist between various 3-D boundary components?”, and research question 22: “What approaches to processing spatial data have been used to perform relationship analysis between 3-D cadastral objects?”.

Fu et al. (2018) studied the characteristics of topological relationships that can exist between 3-D cadastral datasets and described an approach to classify them. They used projected relationships such as the boundary lines and boundary planes between two 3-D cadastral parcels being collinear or coplanar to organize various types of ‘Disjoint’ and ‘Touch’ relationships that can exist in a 3-D cadastral system. The 3-D cadastral parcel boundaries in this study were defined similarly to the node, edge, face organization by Ying et al. (2015) that was presented in Section 2.5.2.

Fu et al. (2018) initially categorized all relationships between 3-D spatial objects as being either ‘Disjoint’ from one another or having an ‘Intersect’ between them. The ‘Intersect’ relationship was broken into 4 refined classifications resulting in the possible relationships between 3-D spatial objects being classified as having either ‘Disjoint’, ‘Touch’, ‘Overlap’, ‘Contain’, or ‘Equal’ topology. A visual representation of these can be seen in Figure 3.4 below.

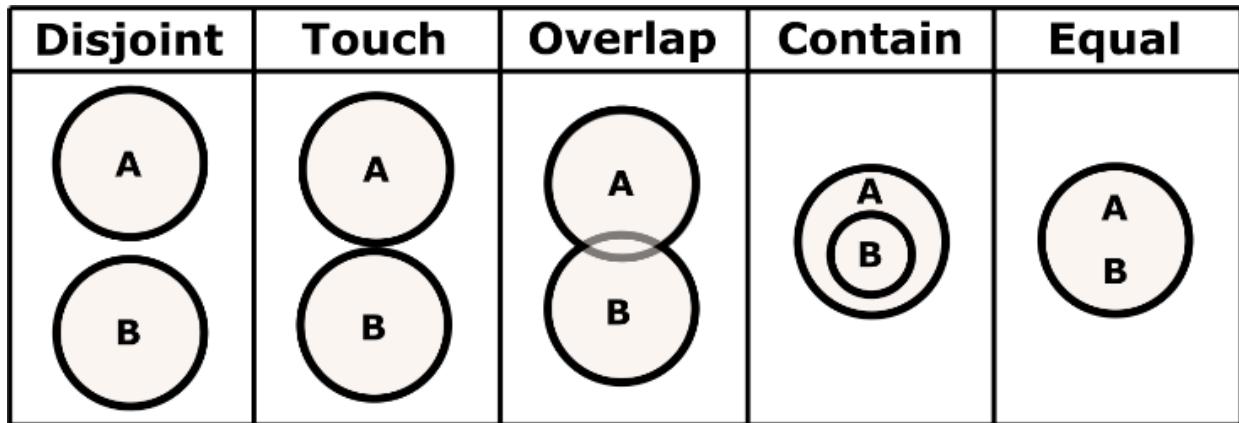


Figure 3.4: Visual Representation of Topological Relationships (Derived from Fu et al., 2018)

The ‘Overlap’ and ‘Contain’ intersect relationships were not included in the study as Fu et al. (2018) claim that they are not valid relationships between 3-D volumetric solid parcel boundaries in a cadastral system due to their assumption that 3-D cadastral space is fully partitioned. Therefore, 3-D parcel boundaries could either be ‘Disjoint’, ‘Touch’, or be ‘Equal’. Two 3-D parcels A and B can share overlapping boundary points, line segments, or faces which will all result in a ‘Touch’ relationship classification.

Fu et al. (2018) identified four different types of ‘Disjoint’ relationships that can exist when considering boundary components of 3-D cadastral objects being projected outwards. These are disjoint objects that share a line and face (LFD), disjoint objects that only share a line (LNFD), disjoint objects that only share a face (NLFD), and disjoint objects that have no shared projected elements (NLNFD) (Fu et al., 2018).

Furthermore, Fu et al. (2018) identified seven types of ‘Touch’ relationships that can exist when considering boundary components of 3-D cadastral objects being projected outwards. These are objects that touch at a point and share a line and face (PLFT), objects that touch at a point and only share a face (PNLFT), objects that touch at a point and only share a line (PLNFT), objects that touch at a point with no shared line or face (PNLNFT), objects that touch at a line and share a face (LFT), objects that touch at a line and don’t share a face (LNFT), and objects that touch at a face (FT) (Fu et al., 2018).

In total, Fu et al. (2018) identified 30 types of disjoint relationships and 30 types of touch relationships that can theoretically exist between 3-D cadastral object boundary points, line segments, faces, and solids using the concepts of these objects having combinations of collinear lines and coplanar planes. For example, in the physical world, a building may contain several strata units that share the same floor plane. In this scenario, these boundary planes between units should be coplanar even though the majority, however, will not share a boundary. These relationship possibilities are shown in Table 3.6 below.

Table 3.6: Disjoint and Touch Relationships in a Cadastral System (Derived from Fu et al., 2018)

Disjoint and Touch Relationships between 3-D Boundary Components in a Cadastral System				
Disjoint Relationships	Point	Line Segment	Face	Solid
Point	D	LFD / NLFD	LFD / NLFD / NLFND	LFD / NLFD / NLFND
Line segment	LFD / NLFD	LFD / NLFD / NLFND	LFD / NLFD / NLFND	LFD / NLFD / NLFND
Face	LFD / NLFD / NLFND	LFD / NLFD / NLFND	LFD / NLFD / LNF / NLFND	LFD / NLFD / LNF / NLFND
Solid	LFD / NLFD / NLFND	LFD / NLFD / NLFND	LFD / NLFD / LNF / NLFND	LFD / NLFD / LNF / NLFND
Touch Relationships	Point	Line Segment	Face	Solid
Point	--	PT	PT	PT
Line segment	PT	PLFT / PNLF	PLFT / PNLF / PNLFNFT	PLFT / PNLF / PNLFNFT
Face	PT	PLFT / PNLF / PNLFNFT	PLFT / PNLF / PLNF / PNLFNFT / LFT / LNFT	PLFT / PNLF / PLNF / PNLFNFT / LFT / LNFT
Solid	PT	PLFT / PNLF / PNLFNFT	PLFT / PNLF / PLNF / PNLFNFT / LFT / LNFT	PLFT / PNLF / PLNF / PNLFNFT / LFT / LNFT / FT

Fu et al. (2018) claim that the relationship distinctions from their study can be used to design a computational framework for classifying topological relationships in 3-D cadastral space. Their study suggests that computing topological relationships can be approached by first determining if boundary components between two 3-D cadastral parcels are collinear or coplanar. This study follows a similar approach in that relationship classifications between point, line, and face boundary components are first approached by implementing projected collinearity and coplanarity checks. Different relationship classification processes are followed depending on the results of these initial checks between boundary components.

Selig (2000) derived relationships that can exist between points, lines, and planes using Clifford Algebra. Relationships were described for Point-Point, Point-Line, Point-Plane, Line-Line, Line-Plane, and Plane-Plane sets using concepts of objects being collinear, coplanar, and parallel. A summarized visualization of these relationship possibilities is presented in Figure 3.5 below.

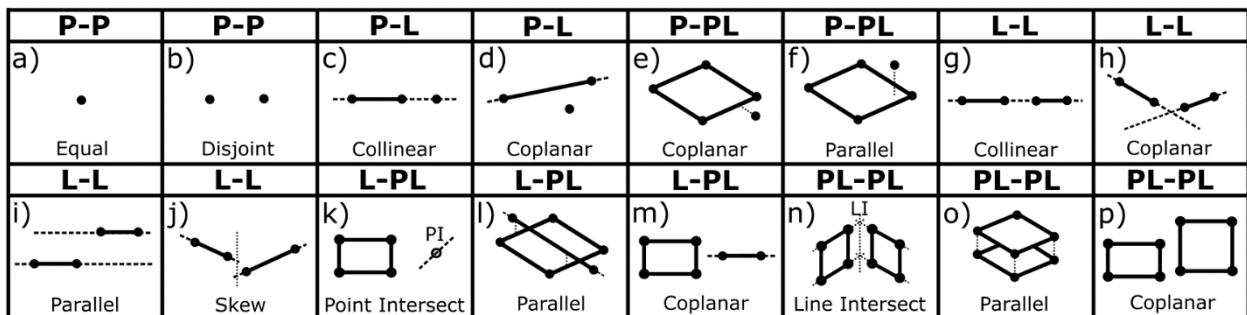


Figure 3.5: Summary of Relationships between Points, Lines, and Planes (Derived from Selig, 2000)

The relationship between two points (P-P) is that they can either have the same coordinate and are equal (Figure 3.5a) or they are disjoint and separated by a distance and direction between them (Figure 3.5b). The relationship between a point and a line (P-L) can either be collinear when the point lies within the equation of the line (Figure 3.5c) or coplanar and parallel when the point exists a defined distance away from the equation of the line (Figure 3.5d). The relationship between a point and a plane (P-PL) can either be coplanar when the point lies within the equation of the plane (Figure 3.5e) or not coplanar and parallel when the point is separated by a distance from the normal of the plane (Figure 3.5f) (Selig, 2000).

The relationship between two lines (L-L) can either be collinear (Figure 3.5g), intersect at a point with an angle between them and be coplanar (Figure 3.5h), be parallel and never touch (Figure 3.5i), or skew and never touch (Figure 3.5j). The relationship between a line and a plane (L-PL) can intersect producing an intersection point (Figure 3.5k), the line can be parallel to the plane with a distance between them (Figure 3.5l), or the line can lie within the plane being coplanar (Figure 3.5m). The relationship between two planes (PL-PL) can intersect producing an intersection line (Figure 3.5n), can be parallel with a distance between them (Figure 3.5o), or can be coplanar (Figure 3.5p) (Selig, 2000).

Xu & Zlatanova (2013) developed an approach to determining 9IM ‘Overlap’ and ‘Meet’ operators between two 3-D planar polygons by only using existing Oracle Spatial operators and applying rotations and projections on the polygons. The study was done under the assumption that the 3-D objects being modelled are composed of multiple planar polygons. Using results from Zlatanova (2000), they identified five different ‘Meet’ scenarios and eight different ‘Overlap’ scenarios to be determined. Like the categorization approach of Selig (2000), each of the polygons considered belongs to an infinite plane. These planes can only produce a non-disjoint relationship when they intersect at a line in 3-D or are coplanar in 2-D.

A processing flow was developed that first checked if the polygons had any interaction using the Oracle function ANYINTERACT. This function can determine if the polygonal planes are disjoint or not. If it was determined that there was some form of interaction, they checked to see if both were coplanar. If they were coplanar, the relationship between the two polygons was evaluated using 2-D projections to determine if they ‘Meet’ or ‘Overlap’. If they were not coplanar, the projected line of the intersection was constructed and the relationship between the line and shared part of each polygon was evaluated to determine if they ‘Meet’ or ‘Overlap’ (Xu & Zlatanova, 2013).

To summarize, this section showed how relationships between 3-D spatial objects can be organized using projected relationship concepts. This study follows an approach similar to Xu & Zlatanova (2013) where the relationship between 3-D boundary components is determined using data process flows that first identify

projected relationships between 3-D cadastral boundary components such as objects being coplanar or parallel. The projected relationship possibilities identified by Selig (2000) were used to organize approaches to solving topological relationship problems.

3.2 Conformal Geometric Algebra Mathematical Theory and Application

This section is included for the following purposes. Section 3.2.1 introduces the Conformal Geometric Algebra (CGA) spatial framework for modelling objects in 3-D. It provides a summary of how the algebra is set up, introduces relevant equations used in the implementation of this study, discusses how it can be used to model 3-D objects, and shows how it can perform relationship analysis on said objects. Section 3.2.2 reviews previous research approaches and implementations that use Conformal Geometric Algebra to model cadastral objects in 3-D and perform some level of topological analysis on them.

3.2.1 CGA Spatial Framework

This section provides a background introduction to Conformal Geometric Algebra and introduces the main equations that create the algebraic 5-D space. It covers multivector k-blades (coordinates) that are used to distinguish between various geometric objects and provides the equations that can be used to generate point, line, and plane spatial objects in CGA space. It reviews metric and topology operators that can be used to analyze modelled objects.

This section addresses research question 23: “What is the Conformal Geometric Algebra spatial framework and how is it realized?”, research question 24: “How can 3-D boundary components be modelled using Conformal Geometric Algebra?”, and research question 25: “What geometric and topological operators can be used to analyze relationships between spatial objects modelled using Conformal Geometric Algebra?”.

Conformal Geometric Algebra is a mathematical framework that can be used to model objects that exist in 3-D Euclidean space (\mathbf{R}^3) such as points, lines, and planes using mathematical concepts such as the inner product (see Equation 2), the outer product (see Equation 3), the geometric product (see Equation 1), and vector subspaces (Dorst et al., 2007:10). It has been proven to be useful for many applications in engineering and computer science (see Section 3.2.2). The algorithms used in this study apply CGA theory to assist in classifying topological relationships between sets of two 3-D boundary components.

The basic algebraic elements in an n-dimensional Vector Algebra are ($e_1, e_2, e_3, \dots, e_n$), or $e_1 = (x)$, $e_2 = (y)$, and $e_3 = (z)$ in 3-D Vector Algebra. In comparison, Hildenbrand & Oldenburg (2015) say that k-blades are the basic algebraic elements in an n-dimensional Geometric Algebra. Hildenbrand & Oldenburg (2015) describe these blades as having grades ($k = 0, 1, 2, 3, \dots, n$) that represent the dimensionality of each subspace. The maximum value for k is the dimension of the embedded space being worked in. A scalar value is considered a 0-blade (blade of grade $k = 0$) and the basis vectors ($e_1, e_2, e_3, \dots, e_n$) are considered 1-blades. 2-blades are spanned by two 1-blades (e.g. $e_1 \wedge e_2$, where \wedge is the outer product) and 3-blades are spanned by three 1-blades (e.g. $e_1 \wedge e_2 \wedge e_3$). This definition of spanning 1-blades continues until there is one element of the maximum grade n ($e_1 \wedge e_2 \wedge e_3 \dots \wedge e_n$ or I_n). This blade element is called the pseudoscalar (I_n) that represents the entire embedded space (Hildenbrand & Oldenburg, 2015). They describe a vector containing multiple k-blades of the same grade as a k-vector and a vector containing elements of different grades as a multivector.

The main product used in Geometric Algebra is called the geometric product (Hildenbrand & Oldenburg, 2015). The geometric product is defined as the summation of the inner (dot) product and the outer (cross) product, both of which are commonly used in 3-D Vector Algebra for describing geometric relationships between two vectors. The geometric product is mainly used in CGA to apply orthogonal transformations and rotations on spatial objects that are represented by multivectors. Applying the geometric product produces a new multivector that represents the symmetric (inner product) and antisymmetric (cross product) components between 2 multivectors, \mathbf{a} and \mathbf{b} (Dorst et al., 2007:143). The definition of the geometric product is presented in Equation 1 below.

$$\text{Geometric Product: } \mathbf{ab} = \mathbf{a} \cdot \mathbf{b} + \mathbf{a} \wedge \mathbf{b} \quad [1]$$

Since the geometric product is defined using the inner and outer products, they can be calculated with respect to the outer product representation. The definition of the inner product ($\mathbf{a} \cdot \mathbf{b}$) and outer product ($\mathbf{a} \wedge \mathbf{b}$) derived from the geometric product (\mathbf{ab}) presented in Equation 1 are presented below as Equations 2 and 3, respectively.

$$\text{Inner (dot) Product: } \mathbf{a} \cdot \mathbf{b} = \frac{1}{2}(\mathbf{ab} + \mathbf{ba}) \quad [2]$$

$$\text{Outer (cross) Product: } \mathbf{a} \wedge \mathbf{b} = \frac{1}{2}(\mathbf{ab} - \mathbf{ba}) \quad [3]$$

Equations 2 and 3 are presented to show that the inner product is the symmetric component of the geometric product, while the outer product is the antisymmetric component of the geometric product. The inner product can be applied to vectors to calculate metric information such as distances or angles (i.e. scalar values). The outer product can be used to span two vectors to create higher dimensional objects such as projected lines and projected planes (see Figure 3.6).

The conformal model of geometric algebra embeds 3-D Euclidean (x, y, z) space \mathbf{R}^3 into an orthogonal 5-D conformal vector space (n_o, e_1, e_2, e_3, n_i) using an $\mathbf{R}^{4,1}$ metric where e_1, e_2 , and e_3 are the x, y , and z vectors, respectively. The two added vectors are the point at the origin (n_o) and the point at infinity (n_i) (Dorst et al., 2007:9). The point at the origin is the origin of the coordinate system and the point at infinity is defined by Dorst et al. (2007:356,359) as a point that has an infinite distance to all finite points, is common to all lines and planes, and is invariant under Euclidean transformations. Dorst et al. (2007:151) say that multivectors of mixed grade are produced through applying the geometric product. The Conformal model of Geometric Algebra refers to the 1-blades as vectors, the 2-blades as bivectors, the 3-blades as trivectors, and the 4-blades as quadvectors. There are 32 k-blades associated with CGA that can be seen in Table 3.7 below.

Table 3.7: 32 Subspace Blade Parameters of CGA R4,1

Vector Subspace Blade Parameters of CGA (R4,1)					
Scalar (0-blades)	Vector (1-blades)	Bivector (2-blades)	Trivector (3-blades)	Quadvector (4-blades)	Pseudoscalar (5-blade)
1	no	$no \wedge e_1$	$no \wedge e_1 \wedge e_2$	$no \wedge e_1 \wedge e_2 \wedge e_3$	$no \wedge e_1 \wedge e_2 \wedge e_3 \wedge ni$
	e1	$no \wedge e_2$	$no \wedge e_1 \wedge e_3$	$no \wedge e_1 \wedge e_2 \wedge ni$	
	e2	$no \wedge e_3$	$no \wedge e_1 \wedge ni$	$no \wedge e_1 \wedge e_3 \wedge ni$	
	e3	$no \wedge ni$	$no \wedge e_2 \wedge e_3$	$no \wedge e_2 \wedge e_3 \wedge ni$	
	ni	$e_1 \wedge e_2$	$no \wedge e_2 \wedge ni$	$e_1 \wedge e_2 \wedge e_3 \wedge ni$	
		$e_1 \wedge e_3$	$no \wedge e_3 \wedge ni$		
		$e_1 \wedge ni$	$e_1 \wedge e_2 \wedge e_3$		
		$e_2 \wedge e_3$	$e_1 \wedge e_2 \wedge ni$		
		$e_2 \wedge ni$	$e_1 \wedge e_3 \wedge ni$		
		$e_3 \wedge ni$	$e_2 \wedge e_3 \wedge ni$		

A variety of geometric objects can be represented using the k-blade multivectors presented in Table 3.7. Yu et al. (2016) make a distinction between CGA objects as either being rounds, flats, Euclidean blades, free blades, and tangent blades. Rounds can have finite length, area, or volume while flats include a (n_i) component and can stretch outwards to infinity. Flat objects are the CGA objects that were used to represent line and plane extensions in this study and are referred to as ‘projected objects’ for the rest of this study. A summary of the spatial objects used in this study to represent 3-D boundary components can be seen in Figure 3.6 below.

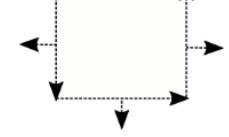
Null Vector (Point)	Projected Line	Projected Plane
$P_1 = no + P_1 + 0.5* P_1 * P_1 *ni$  1-Blades: $[e_1, e_2, e_3, no, ni]$	$L_1 = P_1 \wedge P_2 \wedge ni$  3-Blades: $[e_1 \wedge no \wedge ni, e_2 \wedge no \wedge ni, e_3 \wedge no \wedge ni, e_1 \wedge e_2 \wedge ni, e_1 \wedge e_3 \wedge ni, e_1 \wedge e_2 \wedge ni]$	$PL_1 = P_1 \wedge P_2 \wedge P_3 \wedge ni$  4-Blades: $[e_1 \wedge e_2 \wedge no \wedge ni, e_1 \wedge e_3 \wedge no \wedge ni, e_2 \wedge e_3 \wedge no \wedge ni, e_1 \wedge e_2 \wedge e_3 \wedge ni]$

Figure 3.6: CGA Objects of Interest (with Blade Parameters)

As can be seen in Figure 3.6, the CGA elements of interest for this study are null vectors (points), as well as projected lines and projected planes. While a null vector (point) is bound in space, projected lines, and projected planes extent to the point at infinity (n_i). Relating this to the Dimensional Model framework (see Section 3.1.2), projected lines and planes only have extension elements without boundary limits.

The equation used to generate the CGA parameters for a null vector (point) P is presented in Equation 4. A 3-D (x, y, z) point of \mathbf{R}^3 is embedded into the 5-D conformal space $\mathbf{R}^{4,1}$ by setting the n_o, e_1, e_2 , and e_3 parameters to 1, x, y, z , respectively, and the n_i parameter to $0.5(x^2 + y^2 + z^2)$ (Dorst et al., 2007:360).

$$\text{Null Vector (Point): } P = n_0 + (x) e_1 + (y) e_2 + (z) e_3 + \frac{1}{2}(x^2 + y^2 + z^2) n_i \quad [4]$$

The equations used to generate the CGA parameters for a projected line (L) and plane (PL) are presented in Equations 5 and 6, respectively, as is shown by Yuan et al. (2011) and Zhang, J. et al. (2016). Knowing the format of these equations and the parameters they produce allows CGA intersections to be interpreted into the correct geometry type in the methodology (Chapter 4 and Chapter 5) of this study.

$$\text{Projected Line: } L = P_1 \wedge P_2 \wedge n_i \quad [5]$$

Equation 5 can be used to generate a projected line L that connects 2 null vector points (P_1 and P_2). Using the outer product (\wedge) of these points with the point at infinity allows this line to continue through the space in both directions indefinitely. Six 3-blade parameters are used to represent a projected line L in $\mathbf{R}^{4,1}$ that

all include the ‘ n_i ’ component (see Figure 3.6). These parameters allow for intersections involving projected lines to be performed.

$$\text{Projected Plane:} \quad \mathbf{PL} = \mathbf{P}_1 \wedge \mathbf{P}_2 \wedge \mathbf{P}_3 \wedge n_i \quad [6]$$

Like Equation 5, Equation 6 uses the outer product (\wedge) between a set of three points (\mathbf{P}_1 , \mathbf{P}_2 , and \mathbf{P}_3) to generate a projected plane \mathbf{PL} that divides space into 2 half-spaces. This is like that of a hyperplane described by Billen et al. (2002) when defining point orders in the Dimensional Model and represents the projection of a flat surface outwards to infinity. Four 4-blade parameters can be used to represent a projected plane \mathbf{PL} in $\mathbf{R}^{4,1}$ that all include the ‘ n_i ’ component (see Figure 3.6). These parameters allow for intersections involving projected planes to be performed.

The last few equations in this section can be used to derive geometric and topological information between CGA objects.

$$\text{Line & Point Distance} \quad \mathbf{D}_{L\cdot P} = \text{norm}(\mathbf{L} \cdot \mathbf{P}) \quad [7]$$

Equation 7 is used to calculate the distance $\mathbf{D}_{L\cdot P}$ between a projected line \mathbf{L} and a null vector point \mathbf{P} (see Hildenbrand & Oldenburg, 2015). It is applied in Chapter 4 and Chapter 5 to determine if a boundary point and a boundary line are collinear. The norm of a multivector is defined as the magnitude of its 0-blade scalar components.

$$\text{Dualization:} \quad \mathbf{M}^* = \mathbf{M} \rfloor I_n^{-1} \quad (\text{Dorst et al., 2007:80}) \quad [8]$$

$$\begin{aligned} \text{Intersect/Meet: } \mathbf{Meet}(\mathbf{A}, \mathbf{B}) &= (\mathbf{B} \rfloor I_n^{-1}) \rfloor \mathbf{A} && (\text{Dorst et al., 2007:131}) && [9] \\ \mathbf{\rightarrow} &= \mathbf{B}^* \rfloor \mathbf{A} \end{aligned}$$

Equation 8 and Equation 9 use the left contraction (\rfloor) which is a generalization of the inner product for multivectors. Dorst et al. (2007:75-77) say that the left contraction between two multivectors \mathbf{A} and \mathbf{B} can

be described as the subspace of \mathbf{A} that can be taken out of \mathbf{B} . The dualization \mathbf{M}^* of a multivector \mathbf{M} (Equation 8) can be viewed as the perpendicular component of a multivector with respect to the space (I_n) it resides in (Dorst et al., 2007:80).

Equation 9 is used as a general intersection operation between two multivectors \mathbf{A} and \mathbf{B} . The meet operation is applied to sets of projected CGA boundary component representations in Chapter 4 to initially categorize the type of projected relationship that is occurring between two 3-D boundary components. This is done by interpreting the format of the return geometry type that the meet produces to distinguish what object type (point, line, plane, etc.) that it has. The possible relationship types are shown in Figure 3.5 above.

To summarize, this section introduced the mathematical framework and equations that are used to implement and test some of the methods that are proposed in Chapter 4. CGA Equations 4, 5, and 6 are used to generate multivector blade parameters for 3-D boundary points, lines, and planes. These parameters are used with Equations 7, 8, and 9 to determine if 3-D boundary components modelled in this study are parallel, collinear, coplanar, or intersecting at a point or line when their boundaries are projected outwards. This distinction between 3-D boundary components breaks the relationship determination process into small sets of easier to solve problems.

3.2.2 Modelling Cadastral Objects using CGA

This section reviews a few implementations that use Conformal Geometric Algebra to model 3-D cadastral objects and analyze 3-D boundaries to identify topological relationships that exist between various boundary components. It covers several studies that use the CAUSTA GIS environment to analyze 3-D boundaries, as well as a study that applies specific judgement rules to identify topological relationships using CGA. This section addresses research question 26: “What approaches have been used to determine topological relationships between 3-D cadastral objects using Conformal Geometric Algebra implementations?”.

There have been multiple studies in recent years that use CGA to model and perform analysis on cadastral objects using the CAUSTA geometric algebra environment introduced by Yuan et al. (2010). Yuan et al. (2011) created a 3-D GIS spatial data model in CAUSTA that was used to represent volumetric objects and

their boundaries in CGA space. They show how distances and angles between objects can be derived and how 3-D topological analysis is performed on individual volumetric objects by intersecting their planes and lines to analyze structure (Yuan et al., 2011).

Yuan et al. (2012) then showed how multidimensional objects can be organized in the same space using CGA. Yu et al. (2016) created a framework to compute geometry-oriented topological relations between 3-D objects being rendered using this model. They used the concepts of an object being inside, on, or outside another object to create a general topology operator for determining eight relationships between multidimensional round and flat objects. Results were presented between sets of disjoint 3-D objects in the CAUSTA environment (Yu et al., 2016).

Zhang, J. (2016) proposed a 3-D cadastral data model based on CGA. In this model, a 3-D parcel is stored as a multivector consisting of a summation of its points, lines, and planes. Boundary lines and planes are represented as a combination of the object's projected (flat) component and the set of lines/points that define its boundary (Zhang, J., 2016).

Zhang, F. (2016) used the intersection (meet) results of CGA projected lines and planes along with various other CGA operation checks to create specific judgement rules for testing if two 3-D boundary components intersect or not. The boundary objects considered were 3-D points, straight-line segments, and closed planar polygons.

Six sets of judgement rules were determined for the point-point, line-point, polygon-point, line-line, polygon-line, and polygon-polygon boundary object pairs. By first performing intersections between CGA projected objects, the complexity of the 3-D object intersection problem was reduced to several smaller problems. These problems were approached differently based on the return geometry type of the projected CGA intersections. Their model was able to distinguish between 22 different types of intersections between coplanar boundary components (Zhang, F. 2016).

To summarize, this section presented methods that use the CGA spatial framework to model 3-D cadastral boundaries and identify relationships between different sets of 3-D boundaries. It identified approaches that are promising and have been proven to work to some extent. This study uses methods proposed by Zhang, F. (2016) to categorize and sort boundary component pair relationships into several smaller problem groups which will all be approached differently. Figure 3.5 shows projected geometric relationships that can exist for the intersection result between point, projected line, and projected plane CGA object boundary component pairs in 3-D space.

3.3 Chapter Summary

The relevance this chapter has to the primary research objective is that it reviewed literature that focused on mathematical concepts associated to defining and identifying relationships between spatial objects, as well as approaches that can be used to realize them. This chapter addressed guiding research questions 15 – 26. This chapter was used to perform research activity #4: “Identify topological relationships that can exist between 3-D boundary components” and research activity #5: “Identify existing mathematical theory that can be used to perform topological analysis on 3-D boundaries”.

Section 3.1 addressed research questions 15 – 22 that were related to topological relationship modelling. Firstly, it explored foundations for topological relationships between spatial objects by reviewing the 4-intersection, 9-intersection, dimension extended, and dimension extended 9-intersection models that can be used to distinguish different spatial relationships. Secondly, it reviewed how the 9-intersection model has previously been applied to describe relationships between 3-D spatial objects and introduced the Dimensional Model framework for organizing relationships between spatial objects. Lastly, it reviewed approaches used for implementing topological relationship analysis between 3-D cadastral objects. Approaches that use geometric projected relationships to initially organize relationships, as well as approaches to processing spatial data to implement relationship analysis were discussed here.

Section 3.2 addressed research questions 23 – 26 that were related to Conformal Geometric Algebra and its application. Firstly, it reviewed theory related to the Conformal model of Geometric Algebra and how it can be used to represent and perform topological operations on points, lines, and planes in 3-D space. The equations applied in this study to initially sort relationships between boundary component pairs were presented here. Secondly, it reviewed previous approaches that apply Conformal Geometric Algebra to analyze relationships between 3-D cadastral objects.

The relationships between 3-D boundary components that this study classifies are limited to boundary components being disjoint from each other, touching at a point, line, or plane, and intersecting at a point or line. Relationships are evaluated between six types of boundary component pairs, being between point-point, line-point, line-line, plane-point, plane-line, and plane-plane sets.

The mathematical theory applied to achieve this is through applying and interpreting geometric and topological intersection operations between point, line, and plane spatial objects being modelled in a CGA space. Projected relationship concepts such as spatial objects being collinear, coplanar, parallel, or intersecting at a line or plane are utilized to initially sort relationships into a subset of projected relationship categories. Once these are categorized, most classifications can be achieved using a series of 3-D point-point distance evaluations.

Chapter Four: Methods and Mathematical Model Design

This chapter presents the proposed mathematical model design that can be used to classify geometric and topological relationships between the boundary components of two 3-D cadastral units. The general approach was to start by identifying relevant boundary object components that needed to be modelled, and then to define methodological data flow processes that would allow for relationships to be determined between them.

This chapter covers a definition of the data format used to represent 3-D cadastral unit boundaries, an overview of how these boundary components are evaluated together, and the definition of the specific algorithms that were developed and tested to classify relationships between 3-D boundary components. Section 4.1 describes the digital representations for each of the three 3-D boundary components that are considered in this study. Section 4.2 introduces the six boundary component pair types that are evaluated in the model. The approach to describing relationships between 3-D boundary component pairs is discussed in this section. Methods on how boundary component pairs are initially sorted into projected relationship categories by interpreting return results from CGA projected object intersections are introduced as well. Sections 4.3 - 4.8 present the various algorithms and methods that can be used to determine the final relationship classification between the six boundary component pair types. These last six sections include the bulk of the methodology that was developed in this study to address the primary research objective.

4.1 Defining 3-D Cadastral Object Boundary Components

This section defines the data structures for volumetric cadastral parcels and their 3-D boundary components as they are considered here. This organizational approach is included as a formal definition for how 3-D boundaries were represented in this study. This section presents the formal definitions for 3-D boundary point (BP), 3-D boundary line (BL), and 3-D boundary plane (BPL) objects that are used in this study to represent the boundary components of 3-D cadastral units that can be registered to a strata subdivision plan or centerline condominium plan in Alberta, Canada (review Section 2.4 for details on these plans). Figure 4.1 shows examples for 3-D points, lines, and planes with 3-D geometry for points, and point and line set topology for lines and planes as they relate to 3-D cadastral boundaries.

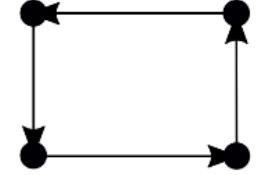
3D Boundary Point (BP)	3D Boundary Line (BL)	3D Boundary Plane (BPL)
 BP = [X, Y, Z]	 BL = [Ls, Le]	 BPL = [L1,.., Ln]

Figure 4.1: Overview of 3-D Boundary Components Modelled in this Study

3-D objects and their boundary components are first introduced by how they would be identified within a survey plan while discussing the topological characteristics that link them to other 3-D boundary component types through the object hierarchy. By enforcing topological hierarchy between (0-D, 1-D, and 2-D) boundary components, the data format ensures that there are no intersecting boundary components or open gaps in the boundary for each 3-D unit modelled.

The geometry for each boundary object is then described with respect to their extensions and limits as defined under the Dimensional Model (see Section 3.1.2). Digital table storage parameters consisting of 3-D point geometry, point-set topology combinations for line and plane orders, and additional CGA parameters for projected points, lines, and planes are defined here as they are used for the implementation (Chapter 5) and testing (Chapter 6) sections of this study. Each example presented in Chapter 6 consists of two 3-D cadastral units A and B.

3-D boundary points are presented in Section 4.1.1, 3-D boundary lines are defined in Section 4.1.2, and 3-D boundary planes are presented in Section 4.1.3. The processes that were followed to generate and store simulated digital representations for 3-D objects and their boundary components are presented in Section 5.1.

4.1.1 3-D Boundary Points (BP)

The geometry for each 3-D cadastral unit modelled in this study is defined by a set of georeferenced 3-D boundary points. Boundary lines and planes in a 3-D strata survey plan in Alberta are defined relative to at least two monument points of known elevation that are localized into a geographic positioning network (Surveys Act R.S.A. 2000 Chapter S-26, S.45-3b). Cadastral units and common property in a 3-D centerline condominium plan are defined relative to the centre of the floor, wall, or ceiling (Condominium Property

Act R.S.A. 2000 Chapter C-22, S.9-4). Therefore, when generating digital versions of 3-D boundaries in this study, 3-D point location estimates are first calculated using metric information that is included in the survey plan drawings. These 3-D boundary points are then used to define the boundary limits of 3-D boundary lines and 3-D boundary planes that define the 3-D boundary of each cadastral unit.

3-D points are boundary components that have a 0-D interior extension and no limit with respect to the Dimensional Model. A sample of the table structure including the parameters used to represent 3-D boundary points in this study can be seen in Table 4.1 below.

Table 4.1: Sample of Parameters for 3-D Boundary Points

3-D Cadastral Object A - Boundary Points					
Boundary Point ID	X (e1)	Y (e2)	Z (e3)	no	ni
A_1	2	2	3	1	8.5
A_2	3	2	3	1	11
...

Table 4.1 shows a sample of two boundary points (A_1 and A_2) as they are stored in a table structure. The ID system for boundary points in this study includes the name of the 3-D cadastral unit it belongs to, as well as the number of the point (A_#/B_#). Boundary point geometry is defined using 3-D point parameters consisting of X (easting), Y (Northing), and Z (Elevation) coordinates. Coordinates are calculated by applying direct (polar) calculations that use distance and angle measurements in a survey plan, which are common to those in the surveying profession. Five 1-blade multivector CGA parameters (e_1, e_2, e_3, n_o, n_i) are used to represent the boundary point extension as it would be loaded into GAViewer for performing topological operations on it. The e_1, e_2 , and e_3 CGA parameters are the same as the X, Y, and Z coordinates, respectively, while n_o and n_i are the point at the origin and the point at infinity, respectively. These CGA parameters were introduced in Section 3.2.1.

Each 3-D boundary point has an additional projected ‘CGA null vector’ point representation generated for it consisting of five 1-blade CGA parameters. The X, Y, and Z vector coordinates are the same as the e_1, e_2 , and e_3 CGA parameters, respectively, as they are shown by Yu et al. (2016: 8) (also see Section 3.2.1 and Equation 4). This CGA representation is used to implement the topological intersection operations introduced in Section 3.2.1, which are further used in this study to interpret the projected relationship between each 3-D boundary point and other 3-D boundary components that are described in Section 4.2.2.

4.1.2 3-D Boundary Lines (BL)

Boundary lines are defined on a strata survey plan by the line that connects two 3-D boundary points using the angles and distances between two points that are included in the plan (Surveys Act R.S.A. 2000 Chapter S-26, S.45-3b). In this study, the geometry of each boundary line is indirectly represented through the calculated start and end 3-D point coordinates that bound it. A closed set of non-intersecting sequential coplanar boundary lines is used to define a boundary plane (see Section 4.1.3), and all 3-D boundary lines modelled are straight and have no arc curves.

3-D lines are boundary components that have a 1-D interior extension (i.e. a line) and two 0-D (start and end) boundary limit points with respect to the Dimensional Model. A sample of the table structure including the parameters used to represent 3-D boundary lines in this model can be seen in Table 4.2 below.

Table 4.2: Sample of Parameters for 3-D Boundary Lines

3-D Cadastral Object A - Boundary Lines								
Boundary Line ID	Start Point ID	End Point ID	$e_1 \wedge n_o \wedge n_i$	$e_2 \wedge n_o \wedge n_i$	$e_3 \wedge n_o \wedge n_i$	$e_1 \wedge e_2 \wedge n_i$	$e_1 \wedge e_3 \wedge n_i$	$e_2 \wedge e_3 \wedge n_i$
AA_1	1	4	0	-1	0	2	0	-3
AA_2	4	3	-1	0	0	-3	-3	0
...

Table 4.2 shows a sample of two boundary lines (AA_1 and AA_2) as they are stored in a table structure. The ID system for boundary lines in this study includes the name of the 3-D cadastral unit it belongs to twice, as well as the number of the line (AA_# / BB_#). The topological direction of each boundary line along with the boundary line limits are defined by indicating a ‘Start Point ID’ and an ‘End Point ID’. Six 3-blade multivector CGA parameters ($e_1 \wedge n_o \wedge n_i$, $e_2 \wedge n_o \wedge n_i$, $e_3 \wedge n_o \wedge n_i$, $e_1 \wedge e_2 \wedge n_i$, $e_1 \wedge e_3 \wedge n_i$, $e_2 \wedge e_3 \wedge n_i$) are used to represent the boundary line extension as it would be loaded into GAViewer for performing topological operations on it. These CGA parameters were introduced in Section 3.2.1.

Each 3-D boundary line has an additional projected ‘CGA line’ representation generated for it consisting of six 3-blade CGA line and free bivector parameters as they are shown in (Yu et al., 2016:8) (also see Equation 5). Each CGA line has a 1-D extension through space with no limits. This CGA representation is used to implement topological intersection operations introduced in Section 3.2.1, which are further used to interpret the projected relationship between each 3-D boundary line and other 3-D boundary components that are described in Section 4.2.2.

4.1.3 3-D Boundary Planes (BPL)

Boundary planes are represented indirectly on a survey plan by identifying a closed set of coplanar points and straight-drawn lines that define it. In this study, there must be at least 3 sequential boundary lines defining a boundary plane that start and end at the same point. For this study, all 3-D boundary planes are flat (i.e. no curved surfaces), and are either horizontal, vertical, or inclined at an angle to the horizontal. While there are scenarios in which legal boundaries are defined using horizontal or vertical cylindrical columns, they are not considered here and are a subject for further investigation.

3-D planes are boundary components that have a 2-D interior extension with 1-D boundary line limits and 0-D boundary point limits with respect to the Dimensional Model. A sample of the table structure including the parameters used to represent 3-D boundary planes in this model can be seen in Table 4.3 below.

Table 4.3: Sample of Parameters for 3-D Boundary Planes

3-D Cadastral Object A - Boundary Planes					
Boundary Plane ID	Oriented Polyline IDs	$e_1 \wedge e_2 \wedge n_o \wedge n_i$	$e_1 \wedge e_3 \wedge n_o \wedge n_i$	$e_2 \wedge e_3 \wedge n_o \wedge n_i$	$e_1 \wedge e_2 \wedge e_3 \wedge n_i$
AAA_1	1_2_3_4	0	1	0	-2
AAA_2	5_6_7_8	0	0	-1	-2
...

Table 4.3 shows a sample of two boundary planes (AAA_1 and AAA_2) as they are stored in a table structure. The ID system for boundary planes in this study includes the name of the 3-D cadastral unit it belongs to three times, as well as the number of the plane (AAA_# / BBA_#). The boundary lines defining a boundary plane represent the boundary plane limits and follow each other in a clockwise oriented direction (oriented polylines) while viewing the plane from the inside of the 3-D cadastral unit. The normal vector associated with this direction gives the boundary plane an inward direction (towards the inside of the 3-D cadastral unit) and an outward direction (towards the outside of the 3-D cadastral unit). This format is like that of Ying et al. (2015) discussed in Section 2.5.2. Four 4-blade multivector CGA parameters ($e_1 \wedge e_2 \wedge n_o \wedge n_i$, $e_1 \wedge e_3 \wedge n_o \wedge n_i$, $e_2 \wedge e_3 \wedge n_o \wedge n_i$, $e_1 \wedge e_2 \wedge e_3 \wedge n_i$) are used to represent the boundary plane extension as it would be loaded into GAViewer for performing topological operations on it. These CGA parameters were introduced in Section 3.2.1.

Each 3-D boundary plane in this study has an additional ‘CGA plane’ representation generated for it consisting of four 4-blade CGA plane and free trivector parameters as they are shown by Yu et al. (2016:8)

(also see Section 3.2.1 and Equation 6). Each CGA plane has a 2-D extension through space with no limits. This CGA representation is used to implement topological intersection operations introduced in Section 3.2.1, which are further used to interpret the projected relationship between each 3-D boundary plane and other 3-D boundary components in Section 4.2.2.

4.2 Cadastral Object Relationship Classification Approach and Projected CGA Relationships

This section discusses the approach followed to model relationships between two 3-D cadastral units. Section 4.2.1 describes the approach to modelling relationships between two 3-D cadastral parcels using individual relationships between their sub-boundary components. The six types of 3-D boundary component pairs that this model evaluates and combines to describe the relationships that exist between two 3-D volumetric cadastral parcels are presented here.

Section 4.2.2 describes how relationships between each 3-D boundary component pair set are initially categorized into projected relationship categories using the concepts of two boundary components being parallel, collinear, coplanar, or intersecting at a point or line when their boundaries are projected outwards by considering their boundary extensions only without boundary limits (see Section 3.1.3 and Section 3.2.1). It presents the various return geometries from CGA projected object intersections that are interpreted to determine which of these different projected relationship categories is occurring between the different boundary component pairs. These interpretations are used in Sections 4.3-4.8 to initially determine which projected relationship scenario is occurring in the overall model algorithms used to classify relationships between boundary component pairs.

4.2.1 Relationship Description using Boundary Component Pairs

The legal objects modelled in this study are 3-D volumetric cadastral units. These are defined as having a 3-D extension, as well as 2-D, 1-D, and 0-D boundary limits with respect to the Dimensional Model. 3-D cadastral units have a boundary that is represented by a set of boundary planes that form a closed surface. All cadastral units tested in Chapter 6 have closed surfaces with no self intersecting boundaries.

When determining the relationship between two 3-D cadastral parcels, the relationships between their boundary sub-components are first classified individually. They are then combined to describe the overlap relationships that are occurring between both parcels if any exist. Relationships between the three boundary sub-components (points, lines, planes) of two 3-D cadastral units (A and B) are individually classified to

describe the relationships that are occurring between them. These classifications are grouped into several boundary component pairs that are presented in Table 4.4 below.

Table 4.4: Boundary Component Pairs between Cadastral Units A and B

Boundary Component Pairs between Units A and B			
	B Points	B Lines	B Planes
A Points	Point _A -Point _B	Point _A -Line _B	Point _A -Plane _B
A Lines	Line _A -Point _B	Line _A -Line _B	Line _A -Plane _B
A Planes	Plane _A -Point _B	Plane _A -Line _B	Plane _A -Plane _B

Table 4.4 shows the nine sets of boundary component pairs that are evaluated between cadastral units A and B. There are six different classification algorithms (point-point, line-point, line-line, plane-point, plane-line, plane-plane) that are applied to evaluate all nine sets of boundary component pairs. The five point-point, line-point, and plane-point boundary component pair sets can only have point geometry overlap. The three line-line and plane-line boundary component pair sets can only have point and line geometry overlap. The plane-plane boundary component pair set can have point, line, and plane geometry overlap.

A boundary point can only have an overlap relationship geometry description of (0D) dimensionality overlapping at its (0D) interior extension. A boundary line can have an overlap relationship geometry description of (0D) or (1D) dimensionality overlapping at one of its (0D) start or end boundary limits or at one of its (1D) interior extension points. A boundary plane can have an overlap relationship geometry description of (0D), (1D), or (2D) dimensionality overlapping at a (0D) boundary vertex point, a (1-D) boundary line, or a (2D) interior extension point.

Final overlap relationships between boundary component pairs are described using four characteristics that are additional to the projected relationship category they are placed in. The first and second characteristics are the dimensionalities of the overlap geometry on boundary components one and two, respectively. The third characteristic is the geometry type (point, line, or plane) of the overlap between the two boundary components being considered. The fourth characteristic is the topological relationship description of the overlap and can be ‘Disjoint’, ‘Touch’ or ‘Intersect’ as is described by Fu et al. (2018) and presented in Section 3.1.2. Disjoint relationships (see Figure 4.2 below) do not use the first three characteristics.

4.2.2 CGA Intersections and Projected Relationships

The topological relationships that this study identifies are if two 3-D boundary components ‘Touch’, ‘Overlap/Intersect’, or are ‘Disjoint’ from each other. Relationships between boundary components are initially categorized using the concepts of projected CGA objects being parallel, collinear, coplanar, or intersecting (see Figure 4.2). This is similar to the approach of Zhang, F. et al. (2016) (see Section 3.2.2).

For five of the six component pairs introduced above, there are possible projected relationships that can exist between them. The point_A-point_B relationship does not require initial CGA intersection interpretation and can be solved using only 3-D point-point distance calculations and interpretations (see Section 4.3). The various projected relationship categories that can exist between the five remaining boundary component pairs can be seen in Figure 4.2 below.

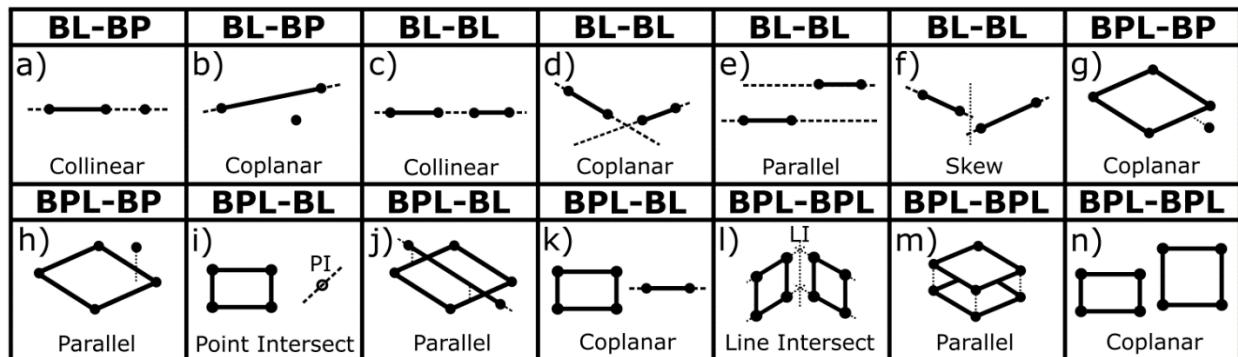


Figure 4.2: Projected CGA Relationship Categories Shown using Disjoint Examples

Each example in Figure 4.2 represents a projected relationship category between boundary component pairs using disjoint scenarios. These categories are evaluated in Sections 4.4 – 4.8 by calculating and interpreting geometric and topological CGA intersection operations between boundary component pairs. The methods that were used to evaluate these projected relationships were developed through running tests on various sets of boundary components in GAViewer. Point-Point component pairs do not require a projected relationship evaluation and are determined through interpreting a 3-D Euclidean point-point distance check between them (see Section 4.3).

Figures 4.2-a and 4.2-b represent disjoint examples of the projected relationships that can exist between the BL-BP (line_A-point_B and line_B-point_A) boundary component pairs. BL and BP are collinear in Figure 4.2-a

and are not collinear (coplanar) in Figure 4.2-b. The processes developed in this study to determine which of these two scenarios is occurring are presented in Section 4.4.

Figures 4.2-c - 4.2-f represent disjoint examples of the projected relationships that can exist between the BL-BL (line_A - line_B) boundary component pairs. Boundary lines are collinear in Figure 4.2-c, coplanar in Figure 4.2-d, parallel in Figure 4.2-e, and not collinear or coplanar (skew) in Figure 4.2-f. The processes developed in this study to determine which of these four scenarios is occurring are presented in Section 4.5.

Figures 4.2-g and 4.2-h represent disjoint examples of the projected relationships that can exist between the BPL-BP (plane_A - point_B and plane_B - point_A) boundary component pairs. BPL and BP are coplanar in Figure 4.2-g and are not coplanar in Figure 4.2-h. The processes developed in this study to determine which of these two scenarios is occurring are presented in Section 4.6.

Figures 4.2-i - 4.2k represent disjoint examples of the projected relationships that can exist between the BPL-BL (plane_A - line_B and plane_B - line_A) boundary component pairs. BPL and BL intersect at a projected point in Figure 4.2-i, are parallel in Figure 4.2-j, and are coplanar in Figure 4.2-k. The processes developed in this study to determine which of these three scenarios is occurring are presented in Section 4.7.

Figures 4.2-l - 4.2-n represent disjoint examples of the projected relationships that can exist between the BPL-BPL (plane_A - plane_B) boundary component pairs. Both planes intersect at a projected line in Figure 4.2-l, are parallel in Figure 4.2-m, and coplanar in Figure 4.2-n. The processes developed in this study to determine which of these three scenarios is occurring are presented in Section 4.8.

Table 4.5: Projected CGA Intersection Interpretations

CGA Object Intersections and Return Geometries							
Boundary Component Pair	CGA Intersection operator	Not Parallel	Parallel	Collinear	Coplanar	Intersection Point	Intersection Line
Line-Point	Equation 7	N/A	Scalar (+)	Scalar (0)	N/A	N/A	N/A
Plane-Point	Equation 9	N/A	Scalar (+/-)	N/A	Point	N/A	N/A
Line-Line	Equation 9	Free Scalar	Free Vector	Line	Flat Point	N/A	N/A
Plane-Line	Equation 9	N/A	Free Vector	N/A	Line	Flat Point	N/A
Plane-Plane	Equation 9	N/A	Free Bivector	N/A	Plane	N/A	Line

Table 4.5 above summarizes the CGA intersection equations (see Section 3.2.1) and the return geometry types that were interpreted to initially determine what projected relationship was occurring between individual boundary component pairs. Definitions for scalar, free scalar, free vector, free bivector, and flat point are described in detail in Section 5.3.1. Using these return geometry interpretations, each relationship

classification problem is simplified into a set of smaller problems. The processes followed to interpret these are discussed in greater detail in Section 5.3.1.

More refined relationships can be derived by determining the extent that the point set representations of each boundary component overlap the projected CGA object intersection geometry. Computational data processing methods for determining these refined final relationship classifications are presented in Sections 4.4-4.8. The algorithms presented in the following sections represent the bulk of the original methodology that was developed in this study to address the primary research objective.

4.3 Point-Point (BP-BP) Relationship Classification Methods

The data process flow algorithm that was proposed to model and classify the relationship **R(BP1,BP2)** between two (0D) boundary points **BP1** and **BP2** can be seen in Figure 4.3 below.

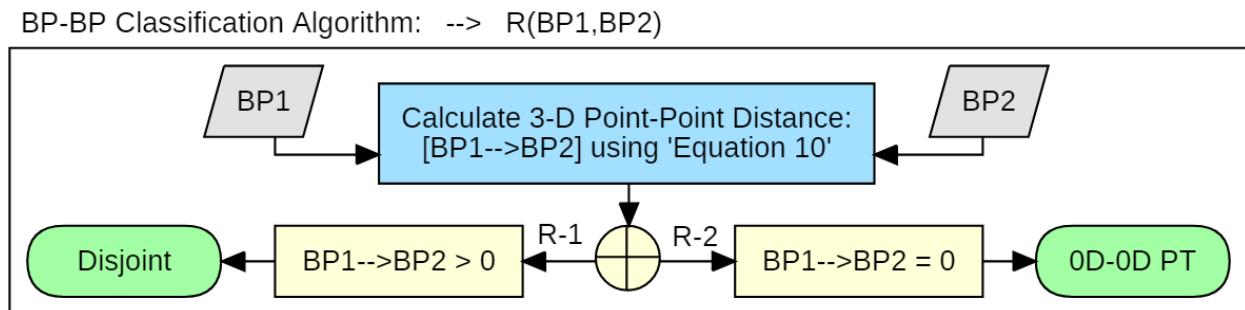


Figure 4.3: BP-BP Classification Algorithm

This classification algorithm (Figure 4.3) requires calculating the scalar ‘3-D Point-Point Distance’ (see Equation 10) between **BP1** and **BP2** using the following equation where **BP1** = (x₁, y₁, z₁) and **BP2** = (x₂, y₂, z₂).

$$3\text{-}D \text{ Point-Point Distance: } D_{p-p} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2} \quad [10]$$

The result of this distance is evaluated to determine if relationship type R-1 or R-2 shown in Figure 4.4 below is occurring between **BP1** and **BP2**.

R-1	R-2
a) BP1 BP2 Disjoint	b) BP1 BP2 0D-0D PT

Figure 4.4: Final BP-BP Relationship Classifications

Figure 4.4a shows an example of scenario R-1 where **BP1** and **BP2** are separated by a distance and are classified as having the ‘Disjoint’ relationship.

Figure 4.4b shows an example of scenario R-2 where **BP1** and **BP2** have no distance between them (same location/ coincident) and are classified as having the ‘0D-0D Point Touch’ relationship.

4.4 Line-Point (BL-BP) Relationship Classification Methods

The data process flow algorithm that was proposed to model and classify the relationship **R(BL,BP)** between a (1D) boundary line **BL** and a (0D) boundary point **BP** can be seen in Figure 4.5 below.

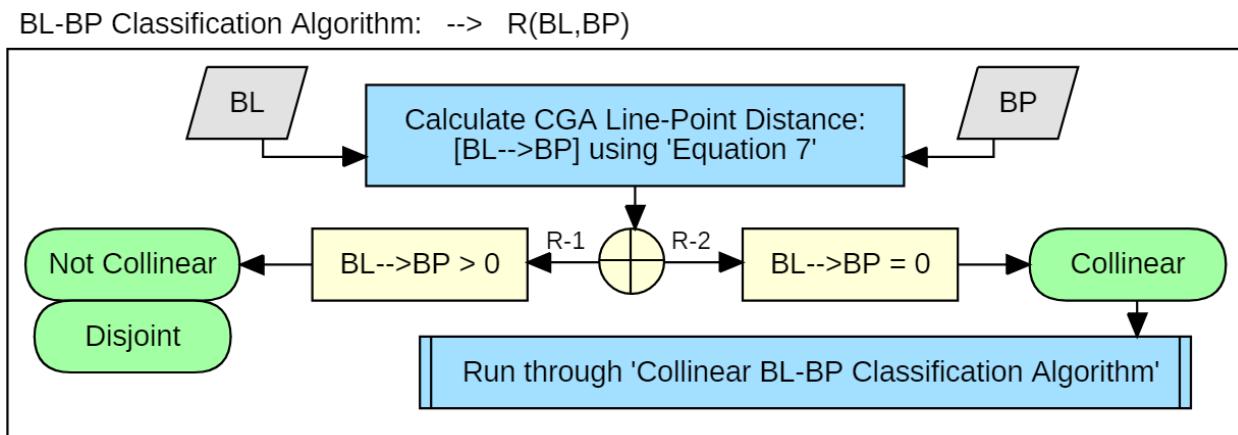


Figure 4.5: BL-BP Classification Algorithm

This BL-BP classification algorithm (Figure 4.5) requires calculating the scalar ‘Projected CGA Line-Point Distance’ between **BL** and **BP** using Equation 7. The result of this distance is evaluated to determine if R-1 or R-2 is occurring between **BL** and **BP**. Disjoint examples for both R-1 and R-2 categories can be seen in Figure 4.6 below.

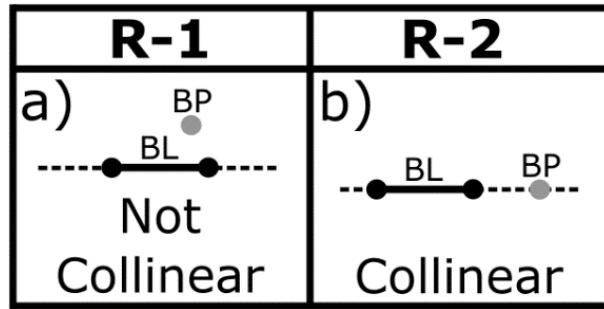


Figure 4.6: BL-BP Projected Relationship Categories

Figure 4.6a shows an example of the projected relationship category R-1 where **BL** and **BP** are not collinear. Any final classification for this scenario results in the ‘Disjoint’ relationship.

Figure 4.6b shows an example of the projected relationship category R-2 where **BL** and **BP** are collinear. If scenario R-2 is occurring, **BL** and **BP** have the collinear projected relationship and are processed through the ‘Collinear BL-BP Classification Algorithm’ (see Figure 4.7). The final relationship classification for this scenario depends on the overlap extent between **BP** and **BL** and can have the ‘Disjoint’, ‘0D-0D Point Touch’, or ‘1D-0D Point Touch’ relationships.

4.4.1 Collinear BL-BP Classification Algorithm

The data process flow algorithm that was developed to classify the relationship **R_COL(BL,BP)** between a (1D) boundary line **BL** and a (0D) boundary point **BP** that have been previously identified as being collinear (COL) using the algorithm shown in Figure 4.5 can be seen in Figure 4.7 below.

Collinear BL-BP Classification Algorithm: \rightarrow R_COL(BL,BP)

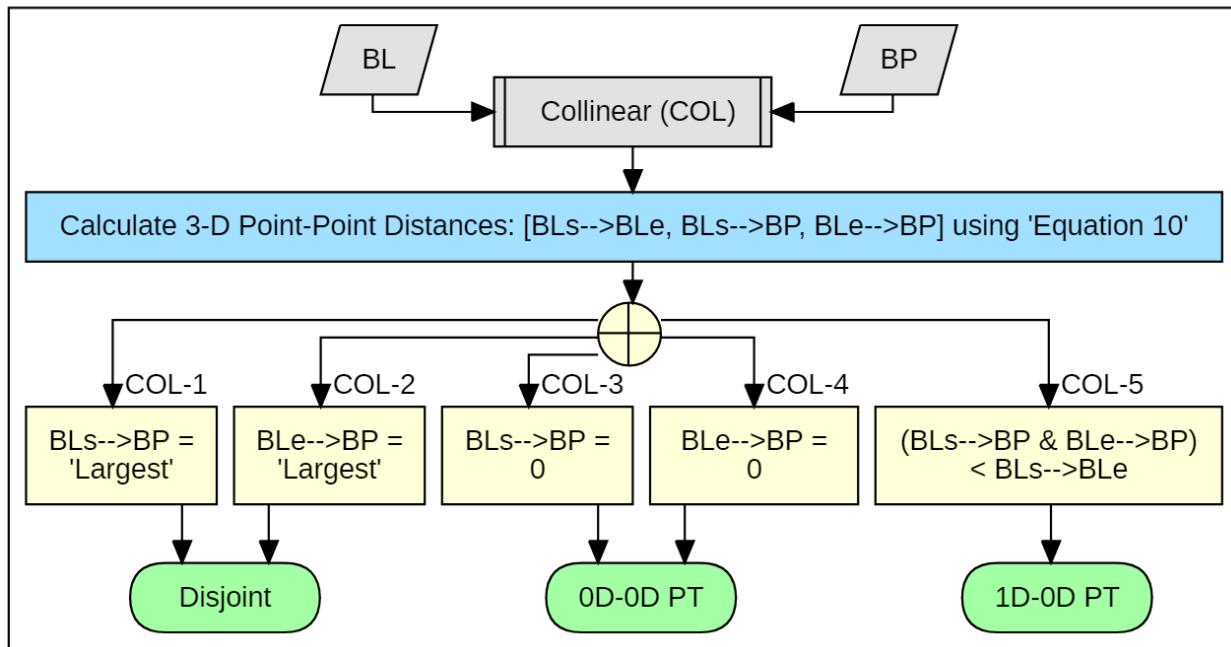


Figure 4.7: Collinear (COL) BL-BP Classification Algorithm

It has previously been determined that **BL** and **BP** are collinear using ‘BL-BP Classification Algorithm’ (see Figure 4.5). This collinear BL-BP classification algorithm (Figure 4.7) requires calculating three ‘3-D Point-Point Distances’ between the start and end points of **BL** (**Ls** and **Le**) and **BP** (**P**) using Equation 10. The results of these three scalar distances are interpreted to determine which of the three final relationship classifications (COL-1, COL-2, and COL-3) is occurring. The COL-3 scenario occurs only if the COL-1 and COL-2 conditions are not met. Visual examples for these three possible collinear BL-BP relationship classification scenarios can be seen in Figure 4.8 below.

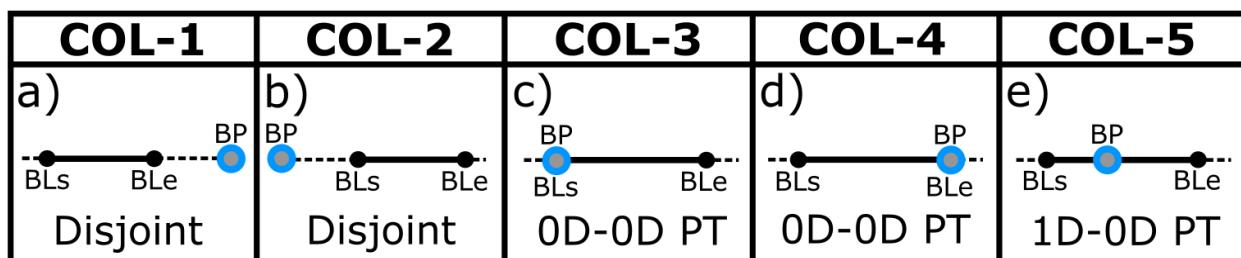


Figure 4.8: Final Collinear (COL) BL-BP Relationship Classifications

Figures 4.8a and 4.8b show examples of the COL-1 and COL-2 scenarios, respectively, where collinear **BL** and **BP** are separated by a distance and have the ‘Disjoint’ relationship classification. **BP** is closest to **BLe** in COL-1 and closest to BLs in COL-2.

Figure 4.8c shows the COL-3 scenario where collinear (0D) **BP** is touching the (0D) **BLs** start boundary point of **BL** and has the ‘0D-0D Point Touch’ relationship classification.

Figure 4.8d shows the COL-4 scenario where collinear (0D) **BP** is touching the (0D) **BLe** end boundary point of **BL** and has the ‘0D-0D Point Touch’ relationship classification.

Figure 4.8e shows an example of the COL-5 scenario where collinear (0D) **BP** is touching a (1D) point that is part of the interior line extension of **BL** and has the ‘1D-0D Point Touch’ relationship classification.

4.5 Line-Line (BL-BL) Relationship Classification Methods

The data process flow algorithm that was developed to classify the relationship $R(BL1, BL2)$ between two (1D) boundary lines **BL1** and **BL2** can be seen in Figure 4.9 below.

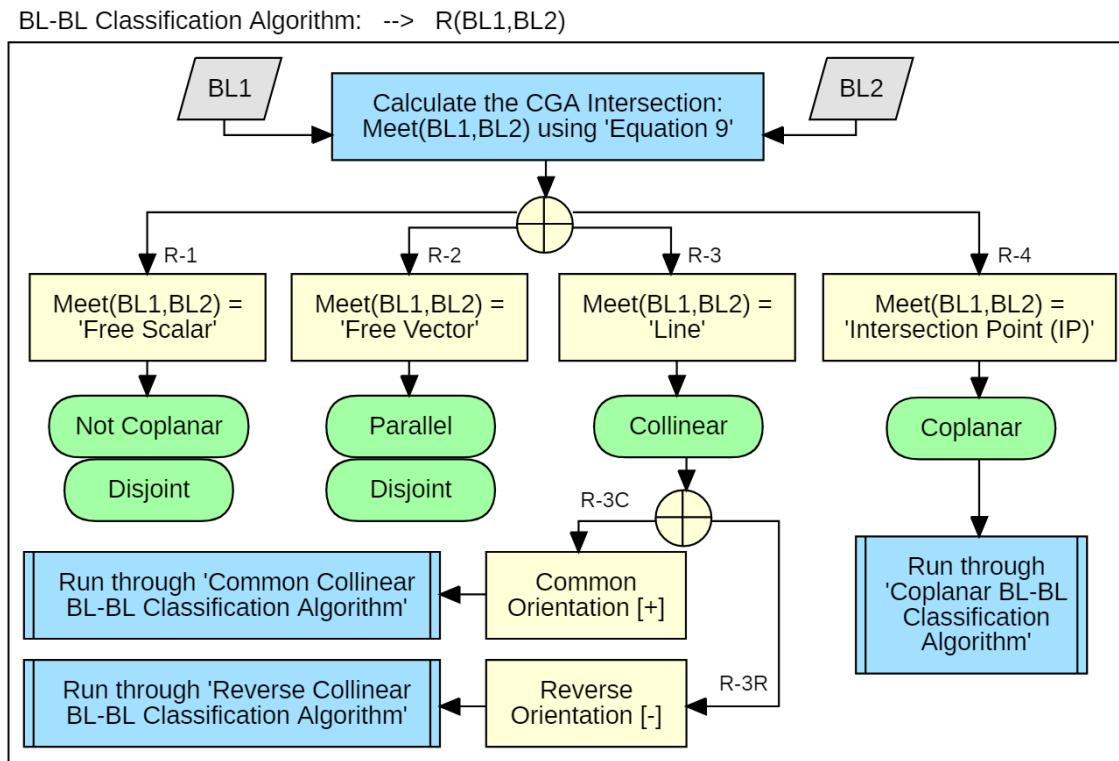


Figure 4.9: BL-BL Classification Algorithm

This BL-BL classification algorithm (Figure 4.9) requires calculating the ‘Projected CGA Meet/Intersection’ between **BL1** and **BL2** using Equation 9. The geometry of this intersection is interpreted to initially determine which of the four projected relationship categories (R-1, R-2, R-3, or R-4) is occurring between **BL1** and **BL2**. Disjoint examples for each of these BL-BL categories can be seen in Figure 4.10 below.

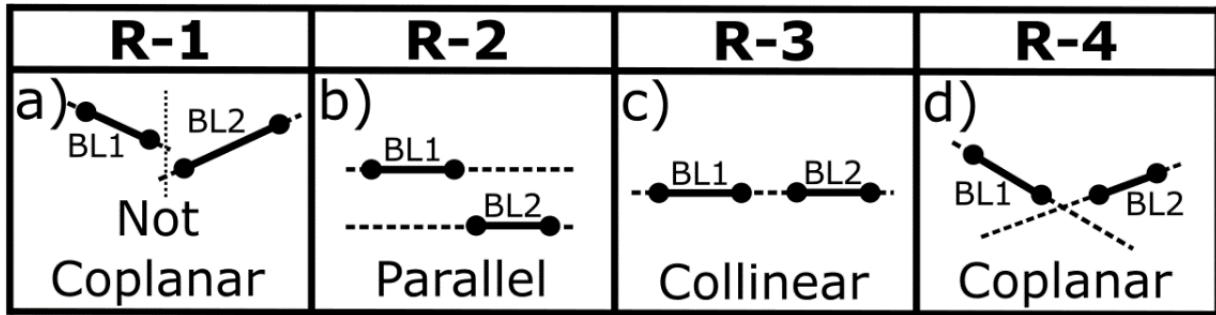


Figure 4.10: BL-BL Projected Relationship Categories

Figure 4.10a shows an example of the projected relationship category R-1 where **BL1** and **BL2** are not coplanar or collinear (skew). The CGA intersection in this scenario produces a ‘Free Scalar’ CGA geometry result (see Section 3.2.1). Any final classification for this scenario results in the ‘Disjoint’ relationship.

Figure 4.10b shows an example of the projected relationship category R-2 where **BL1** and **BL2** are parallel. The CGA intersection in this scenario produces a ‘Free Vector’ geometry result. This is because the only element common between both CGA lines is a direction and a scalar distance (see Section 3.2.1). Similarly, any final classification for this scenario results in the ‘Disjoint’ relationship.

Figure 4.10c shows an example of the projected relationship category R-3 where **BL1** and **BL2** are collinear. The CGA intersection in this scenario produces a CGA ‘Line’ geometry result. These lines can have a common or reverse orientation with respect to their start and end points (see Section 3.2.1). Collinear lines with common orientation are facing the same direction while collinear lines with reverse orientation are facing opposite directions. If scenario R-3 with common orientation is occurring, **BL1** and **BL2** are processed through the ‘Common Collinear BL-BL Classification Algorithm’ (see Figure 4.11). If scenario R-3 with reverse orientation is occurring, **BL1** and **BL2** are processed through the ‘Reverse Collinear BL-BL Classification Algorithm’ (see Figure 4.13). The final relationship classification for both scenarios depends on the extent of line overlap between **BL1** and **BL2** and can have the ‘Disjoint’, ‘0D-0D Point Touch’, or ‘1D-1D Line Touch’ relationships.

Figure 4.10d shows an example of the projected relationship category R-4 where **BL1** and **BL2** are coplanar. The CGA intersection in this scenario produces a ‘Flat Point/Intersection Point’ **IP** geometry result. If scenario R-4 is occurring, **BL** and **BP** have the coplanar projected relationship and are processed through the ‘Coplanar BL-BL Classification Algorithm’ (see Figure 4.15). The final relationship classification for this scenario depends on the extent of overlap **BL1** and **BL2** each have with **IP** and can have the ‘Disjoint’, ‘0D-0D Point Touch’, 1D-0D/0D-1D Point Touch’ or ‘1D-1D Point Intersection’ relationships.

4.5.1 Common Collinear BL-BL Classification Algorithm

The data process flow algorithm that was developed to classify the relationship **C_COL_R(BL1,BL2)** between two (1D) boundary lines **BL1** and **BL2** that have previously been identified as being collinear with common orientation (C-COL) using the algorithm shown in Figure 4.9 can be seen in Figure 4.11 below.

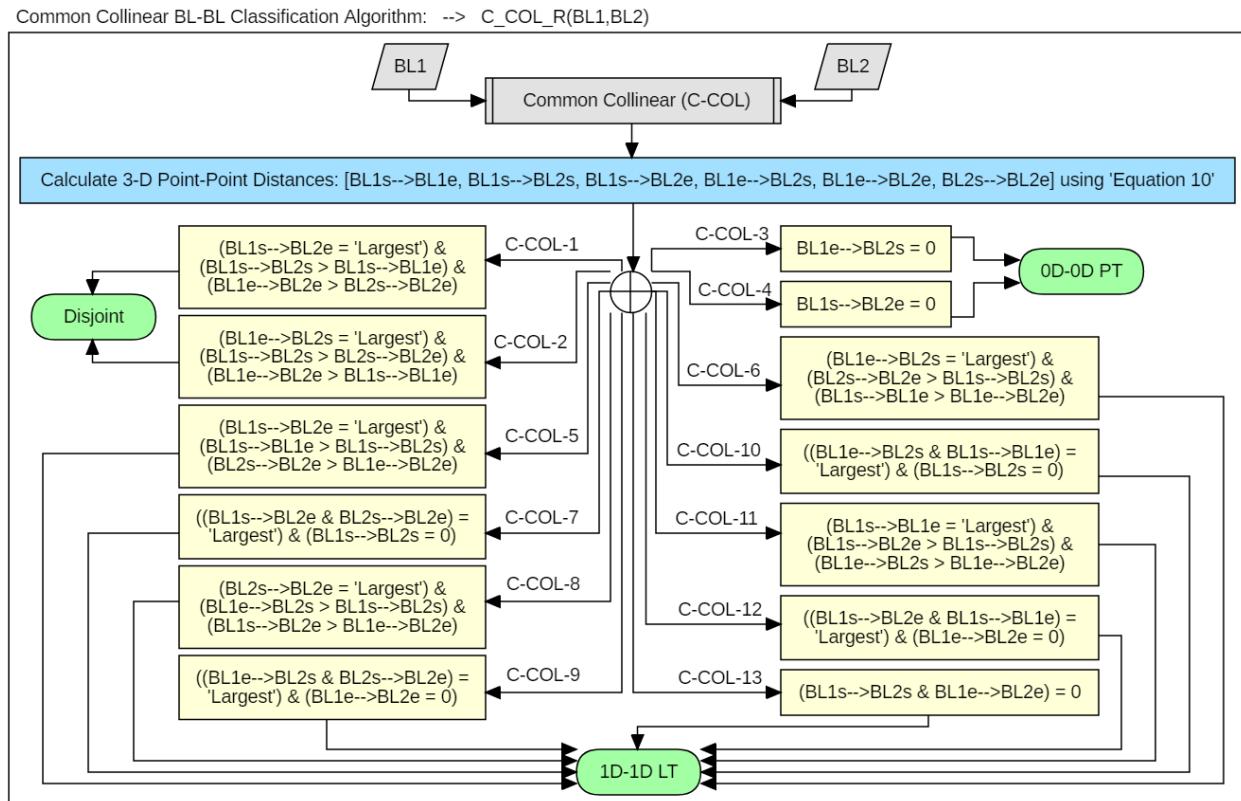


Figure 4.11: Common Collinear (C-COL) BL-BL Classification Algorithm

It has previously been determined that **BL1** and **BL2** are collinear with common orientation using ‘BL-BL Classification Algorithm’ (see Figure 4.9). This common collinear BL-BL classification algorithm (Figure 4.11) requires calculating six ‘3-D Point-Point Distances’ between the start and end boundary points of **BL1** (**L1s** and **L1e**) and **BL2** (**L2s** and **L2e**) using Equation 10. These six scalar distances are evaluated to determine which of the three final relationship classifications is occurring between common collinear lines **BL1** and **BL2**. Visual examples showing 13 variations of these three common collinear BL-BL relationship classifications can be seen in Figure 4.12 below.

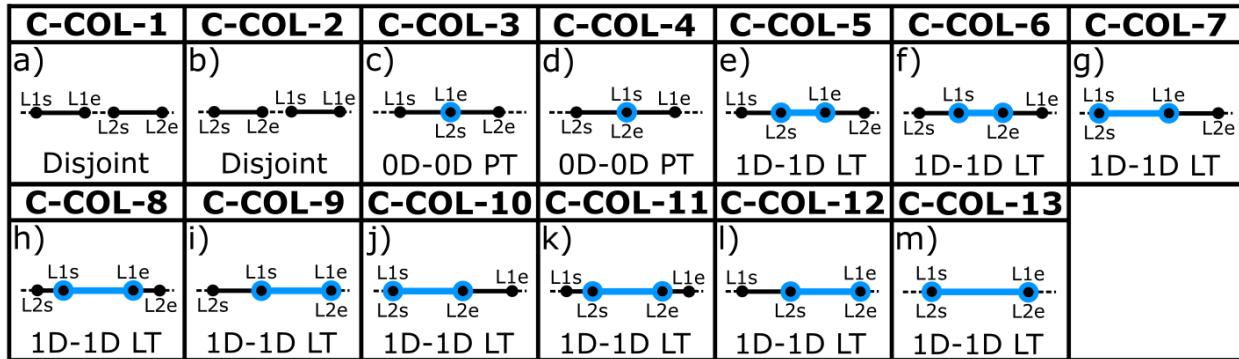


Figure 4.12: Final Common Collinear (C-COL) BL-BL Relationship Classifications

Figures 4.12a and 4.12b show examples of the C-COL-1 and C-COL-2 scenarios, respectively, where common collinear lines **BL1** and **BL2** are separated by a distance and have the ‘Disjoint’ relationship classification. **BL1** is ordered first in C-COL-1 while **BL2** is ordered first in C-COL-2.

Figures 4.12c and 4.12d show examples of the C-COL-3 and C-COL-4 scenarios, respectively, where common collinear lines **BL1** and **BL2** touch at a point and have the ‘0D-0D Point Touch’ relationship classification. **BL1** is ordered first in C-COL-3 while **BL2** is ordered first in C-COL-4.

Figures 4.12e and 4.12f show examples of the C-COL-5 and C-COL-6 scenarios, respectively, where common collinear lines **BL1** and **BL2** both have partial line overlap with the other and have the ‘1D-1D Line Touch’ relationship classification. **BL1** is ordered first in C-COL-5 while **BL2** is ordered first in C-COL-6.

Figures 4.12g, 4.12h, and 4.12i show examples of the C-COL-7, C-COL-8, and C-COL-9 scenarios, respectively, where common collinear line **BL2** overlaps **BL1** completely and has the ‘1D-1D Line Touch’

relationship classification. **BL1** and **BL2** have the same start point in C-COL-7 and the same end point in C-COL-9. **BL1** is fully contained in the interior of **BL2** in C-COL-8.

Figures 4.12j, 4.12k, and 4.12l show examples of the C-COL-10, C-COL-11, and C-COL-12 scenarios, respectively, where common collinear line **BL1** overlaps **BL2** completely and has the ‘1D-1D Line Touch’ relationship classification. **BL1** and **BL2** have the same start point in C-COL-10 and the same end point in C-COL-12. **BL2** is fully contained in the interior of **BL1** in C-COL-11.

Figure 4.12m shows an example of the C-COL-13 scenario where common collinear lines **BL1** and **BL2** have complete overlap with each other and have the ‘1D-1D Line Touch’ relationship classification. **BL1** and **BL2** have their start and end points in the same locations.

4.5.2 Reverse Collinear BL-BL Classification Algorithm

The data process flow algorithm that was developed to classify the relationship **R_COL_R(BL1,BL2)** between two (1D) boundary lines **BL1** and **BL2** that have previously been identified as being collinear with reverse orientation (R-COL) using the algorithm shown in Figure 4.9 can be seen in Figure 4.13 below.

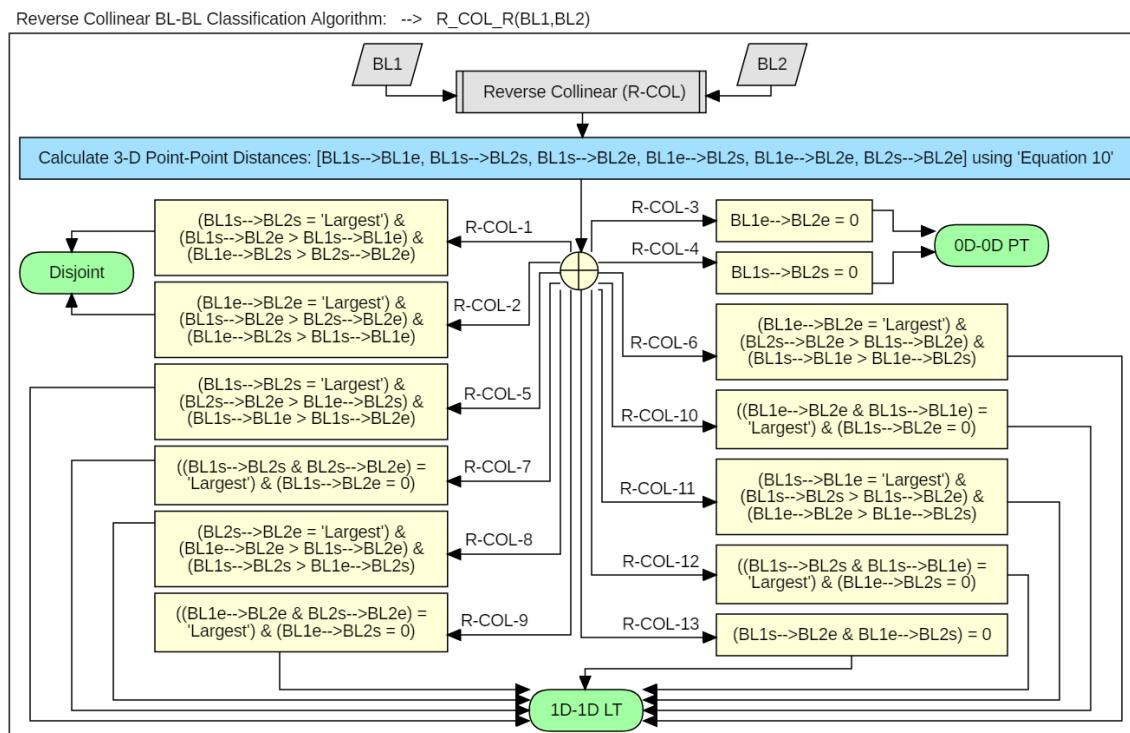


Figure 4.13: Reverse Collinear (R-COL) BL-BL Classification Algorithm

It has previously been determined that **BL1** and **BL2** are collinear with reverse orientation using ‘BL-BL Classification Algorithm’ (see Figure 4.9). This reverse collinear BL-BL classification algorithm (Figure 4.13) requires calculating six ‘3-D Point-Point Distances’ between the start and end boundary points of **BL1** (**L1s** and **L1e**) and **BL2** (**L2s** and **L2e**) using Equation 10. These are the same six distances that are initially calculated in the ‘Common Collinear BL-BL Classification Algorithm’ (see Figure 4.11). The results of these six scalar distances are evaluated to determine which of the three final (R-COL) relationship classifications are occurring between reverse collinear lines **BL1** and **BL2**. Visual examples showing 13 variations of these three reverse collinear BL-BL relationship classifications can be seen in Figure 4.14 below.

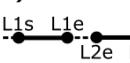
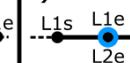
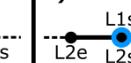
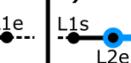
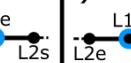
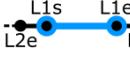
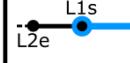
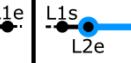
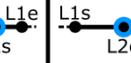
R-COL-1	R-COL-2	R-COL-3	R-COL-4	R-COL-5	R-COL-6	R-COL-7
a)  Disjoint	b)  Disjoint	c)  0D-0D PT	d)  0D-0D PT	e)  1D-1D LT	f)  1D-1D LT	g)  1D-1D LT
R-COL-8	R-COL-9	R-COL-10	R-COL-11	R-COL-12	R-COL-13	
h)  1D-1D LT	i)  1D-1D LT	j)  1D-1D LT	k)  1D-1D LT	l)  1D-1D LT	m)  1D-1D LT	

Figure 4.14: Final Reverse Collinear (R-COL) BL-BL Relationship Classifications

Figures 4.14a and 4.14b show examples of the R-COL-1 and R-COL-2 scenarios, respectively, where reverse collinear lines **BL1** and **BL2** are separated by a distance and have the ‘Disjoint’ relationship classification. The end points of **BL1** and **BL2** are closest in R-COL-1 while the start points of **BL1** and **BL2** are closest in R-COL-2.

Figures 4.14c and 4.14d show examples of the R-COL-3 and R-COL-4 scenarios, respectively, where reverse collinear lines **BL1** and **BL2** touch at a point and have the ‘0D-0D Point Touch’ relationship classification. The end points of **BL1** and **BL2** touch in R-COL-3 while the start points of **BL1** and **BL2** touch in R-COL-4.

Figures 4.14e and 4.14f show examples of the R-COL-5 and R-COL-6 scenarios, respectively, where reverse collinear lines **BL1** and **BL2** both have partial line overlap with the other and have the ‘1D-1D Line Touch’ relationship classification. End points of **BL1** and **BL2** have overlap in R-COL-5 while start points of **BL1** and **BL2** have overlap R-COL-6.

Figures 4.14g, 4.14h, and 4.14i show examples of the R-COL-7, R-COL-8, and R-COL-9 scenarios, respectively, where reverse collinear line **BL2** overlaps **BL1** completely and has the ‘1D-1D Line Touch’ relationship classification. The **BL1** start point has the same location as the **BL2** end point in R-COL-7 while the **BL1** end point has the same location as the **BL2** start point in R-COL-9. **BL1** is fully contained in the interior of **BL2** in R-COL-8.

Figures 4.14j, 4.14k, and 4.14l show examples of the R-COL-10, R-COL-11, and R-COL-12 scenarios, respectively, where reverse collinear line **BL1** overlaps **BL2** completely and has the ‘1D-1D Line Touch’ relationship classification. The **BL1** start point has the same location as the **BL2** end point in R-COL-10 while the **BL1** end point has the same location as the **BL2** start point in R-COL-12. **BL2** is fully contained in the interior of **BL1** in R-COL-11.

Figure 4.14m shows an example of the R-COL-13 scenario where reverse collinear lines **BL1** and **BL2** have complete overlap with each other and have the ‘1D-1D Line Touch’ relationship classification. **BL1** and **BL2** have their start and end points in the same locations.

4.5.3 Coplanar BL-BL Classification Algorithm

The data process flow algorithm that was developed to classify the relationship **R_COP(BL1,BL2)** between two (1D) boundary lines **BL1** and **BL2** that have previously been identified as being coplanar (COP) and intersecting at a projected intersection point **IP** using the algorithm shown in Figure 4.9 can be seen in Figure 4.15 below.

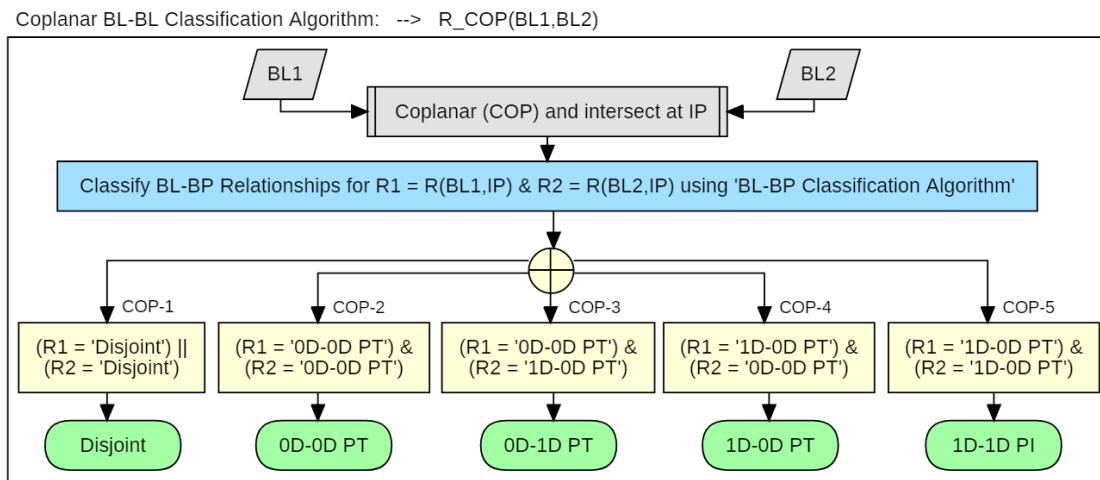


Figure 4.15: Coplanar (COP) BL-BL Classification Algorithm

It has previously been determined that **BL1** and **BL2** are coplanar and intersect at projected intersection point **IP** using ‘BL-BL Classification Algorithm’ (see Figure 4.9). This coplanar BL-BL classification algorithm (Figure 4.15) requires first classifying the BL-BP relationships between **BL1-IP** and **BL2-IP** using ‘BL-BP Classification Algorithm’ (see Figure 4.5). $R(BL1-IP)$ and $R(BL2-IP)$ are evaluated to determine which of the five final (COP) relationship classifications is occurring between coplanar lines **BL1** and **BL2**. Visual examples of these five coplanar BL-BL relationship classifications can be seen in Figure 4.16 below.

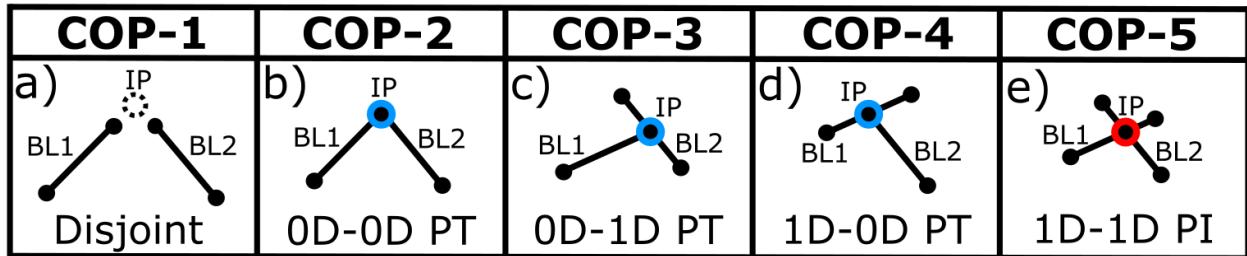


Figure 4.16: Final Coplanar (COP) BL-BL Relationship Classifications

Figure 4.16a shows an example of the COP-1 scenario where (0D) **IP** does not have any overlap with one or both (1D) lines **BL1** or **BL2**. In this scenario, coplanar (1D) lines **BL1** and **BL2** have the ‘Disjoint’ relationship classification.

Figure 4.16b shows an example of the COP-2 scenario where (0D) **IP** touches both (1D) lines **BL1** and **BL2** at one of their (0D) start or end boundary points. In this scenario, coplanar lines **BL1** and **BL2** have the ‘0D-0D Point Touch’ relationship classification.

Figure 4.16c shows an example of the COP-3 scenario where (0D) **IP** touches (1D) line **BL1** at one of its (0D) start or end boundary points and (0D) **IP** touches (1D) line **BL2** at a location within its (1D) line extension. In this scenario, coplanar lines **BL1** and **BL2** have the ‘0D-1D Point Touch’ relationship classification.

Figure 4.16d shows an example of the COP-4 scenario where (0D) **IP** touches (1D) line **BL1** at a location within its (1D) line extension and (0D) **IP** touches (1D) line **BL2** at one of its (0D) start or end boundary points. In this scenario, coplanar lines **BL1** and **BL2** have the ‘1D-0D Point Touch’ relationship classification.

Figure 4.16e shows an example of the COP-5 scenario where (0D) **IP** touches both (1D) lines **BL1** and **BL2** at a location within both of their (1D) line extensions. In this scenario, coplanar lines **BL1** and **BL2** have the ‘1D-1D Point Intersect’ relationship classification.

4.6 Plane-Point (BPL-BP) Relationship Classification Methods

The data process flow algorithm that was developed to classify the relationship **R(BPL,BP)** between a (2D) boundary plane **BPL** and a (0D) boundary point **BP** can be seen in Figure 4.17 below.

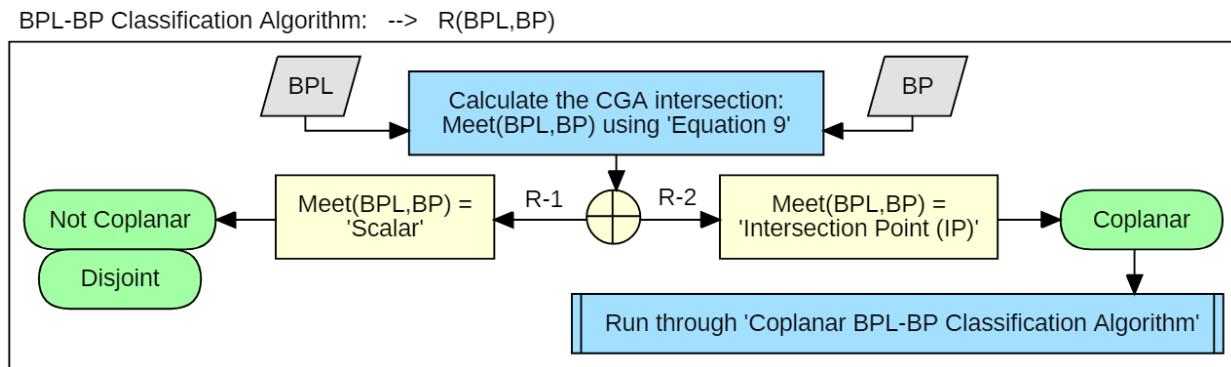


Figure 4.17: BPL-BP Classification Algorithm

This BPL-BP classification algorithm (Figure 4.17) requires calculating the ‘Projected CGA Intersection’ between **BPL** and **BP** using Equation 9. The geometry of this intersection is interpreted to initially determine which of the two projected relationship categories (**R-1** and **R-2**) is occurring between **BPL** and **BP**. If Disjoint examples for each of these BPL-BP categories can be seen in Figure 4.18 below.

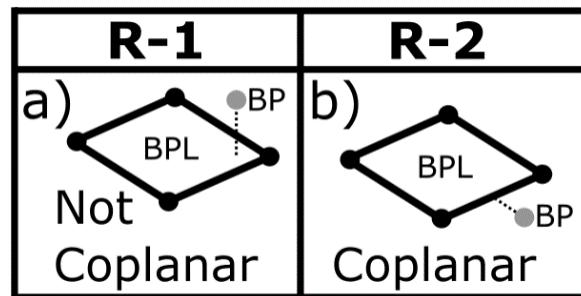


Figure 4.18: BPL-BP Projected Relationship Categories

Figure 4.18a shows an example of the projected relationship category R-1 where **BPL** and **BP** are not coplanar. The CGA intersection in this scenario produces a [+/-] ‘scalar’ geometry result representing the parallel distance between **BPL** and **BP**. Any final classification for this scenario results in the ‘Disjoint’ relationship.

Figure 4.18b shows an example of the projected relationship category R-2 where **BPL** and **BP** are coplanar. The CGA intersection in this scenario produces a ‘Point/Intersection Point’ **IP** geometry result. If scenario R-2 is occurring, **BPL** and **BP** are processed through the ‘Coplanar BPL-BP Classification Algorithm’ (see Figure 4.19). The final relationship classification for this scenario depends on the extent of geometry overlap between **BPL** and **BP** and can have the ‘Disjoint’, ‘0D-0D Point Touch’, 1D-0D Point Touch’, or ‘2D-0D Point Touch’ relationships. This final coplanar BPL-BP classification is determined using the methods described in Section 4.6.1.

4.6.1 Coplanar BPL-BP Classification Algorithm

The data process flow algorithm that was developed to classify the relationship **R_COP(BPL,BP)** between a (2D) **BPL** and a (0D) **BP** that have previously been identified as being coplanar (COP) using the algorithm shown in Figure 4.17 is presented using five steps and can be seen in Figure 4.19 below.

Coplanar BPL-BP Classification Algorithm: $\rightarrow R_{COP}(BPL, BP)$

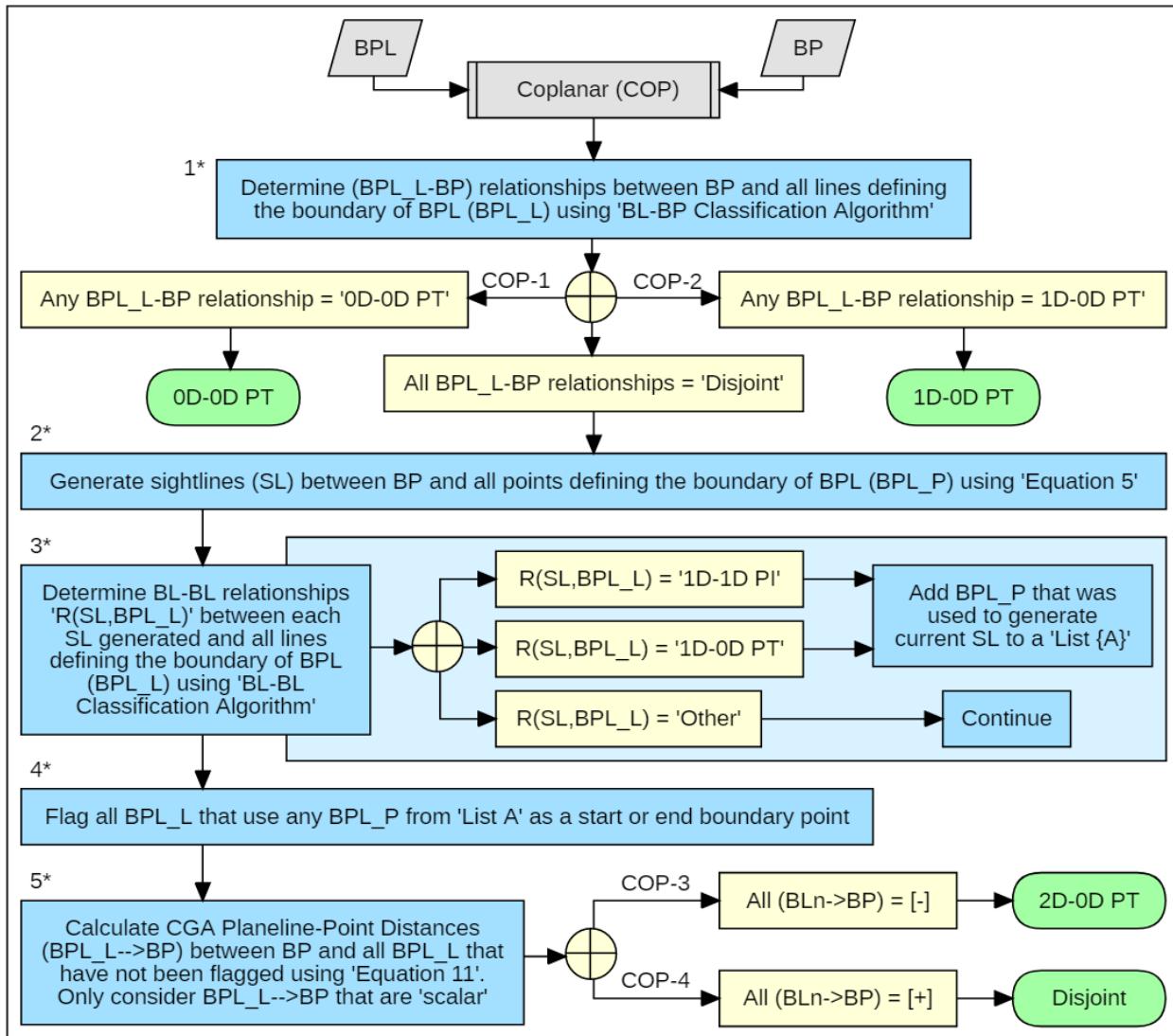


Figure 4.19: Coplanar (COP) BPL-BP Classification Algorithm

It has previously been determined that **BPL** and **BP** are coplanar using ‘BPL-BP Classification Algorithm’ (see Figure 4.17). The first step in this coplanar BPL-BP classification algorithm (Figure 4.19) requires classifying the BL-BP relationships between **BP** and all lines (**BPL_L**) that define the boundary of **BPL** using ‘BL-BP Classification Algorithm’ (see Figure 4.5). These relationships are evaluated to determine which of the three initial scenarios is occurring between coplanar **BPL** and **BP**. The COP-1 and COP-2 scenarios occur when **BP** is situated somewhere on the (0D) point boundary or (1D) line boundary of **BPL**. COP-1 occurs when two classifications between **BP** and all **BPL_L** have the ‘0D-0D PT’ relationship.

COP-2 occurs when one classification between **BP** and all **BPL_L** have the ‘1D-0D PT’ relationship. Visual examples for scenarios COP-1 and COP-2 can be seen in Figure 4.20a and 4.20b below.

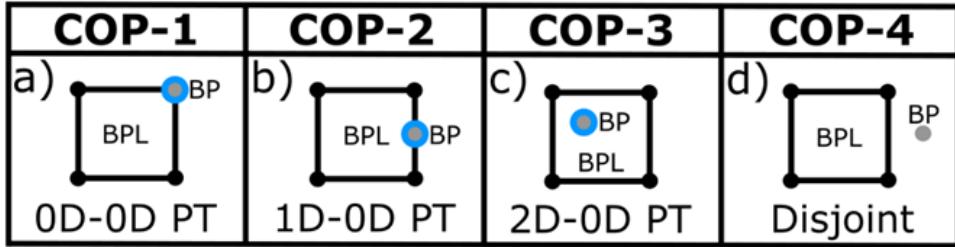


Figure 4.20: Final Coplanar (COP) BPL-BP Relationship Classifications

Steps 2-5 determine if the COP-3 or COP-4 scenario is occurring and are only followed if all classifications between **BP** and all **BPL_L** have the ‘Disjoint’ relationship. In this scenario, **BP** can be positioned inside (COP-3) or outside (COP-4) the boundary of **BPL**. Visual examples for scenarios COP-3 and COP-4 can be seen in Figures 4.20c and 4.20d, respectively.

Step 2 requires generating sightlines (**SL**) between **BP** and each **BPL** boundary point (**BPL_P**) using Equation 5. Step 3 requires classifying the BL-BL relationships between each **SL** generated and all **BPL_L** defining **BPL**. If any of these BL-BL relationships between all **BPL_L** and the current **SL** is ‘1D-1D PI’ or ‘1D-0D PT’, the **BPL_P** used to generate the current **SL** is added to a flagged point list {A}. Step 4 is to flag all **BPL_L** that use any flagged **BPL_P** in list {A} as a start or end boundary point. Step 5 requires calculating and evaluating oriented planeline-point distances (**BPL_L**→**BP**) between **BP** and the remaining **BPL_L** that have not been flagged using Equation 11 presented below. Equation 11 was derived through the testing of different scenarios in GAViewer.

$$\text{Planeline-Point Distance: } D_{BPL_L-BP} = \text{Meet}\left(-M^*(BPL_L) \cdot M^*(BPL), BP\right) \quad [11]$$

Equation 11 calculates the oriented planeline-point distance between **BPL_L** and **BP** by evaluating the Meet (Equation 9) between **BP** and an intermediate plane that is defined by being perpendicular to **BPL** and being coplanar with **BPL_L**. This intermediate plane is generated using the M^* dualization (Equation 8) of the inner product (Equation 1) between the M^* dualizations of **BPL_L** and **BPL**. Positive scalar results are interpreted as **BP** being in the ‘outside’ direction of **BPL** with respect to the position of the

current **BPL_L**. Negative scalar results are interpreted as **BP** being in the ‘‘inside’ direction of **BPL** with respect to the position of the current **BPL_L**. A CGA multivector with ‘Line’ geometry results mean that **BP** and the current **BPL_L** are collinear. These are interpreted as having 0 distance and are ignored for the final evaluation.

The final evaluation uses only projected distances from lines that produce a ‘scalar’ result and have not been flagged to determine if **BP** is inside (all remaining [-]) or outside (all remaining [+]). **BP** is outside the boundary of **BPL** if all remaining projected distances are positive resulting in the COP-4 ‘Disjoint’ relationship classification between (2D) **BPL** and (0D) **BP**. **BP** is inside the boundary of **BPL** if all remaining projected distances are negative resulting in the ‘2D-0D PT’ relationship classification between (2D) **BPL** and (0D) **BP**.

To help better explain the processes followed in steps 2-5, two conceptual examples representing scenarios COP-3 (inside) and COP-4 (outside) are presented in Figure 4.21 below.

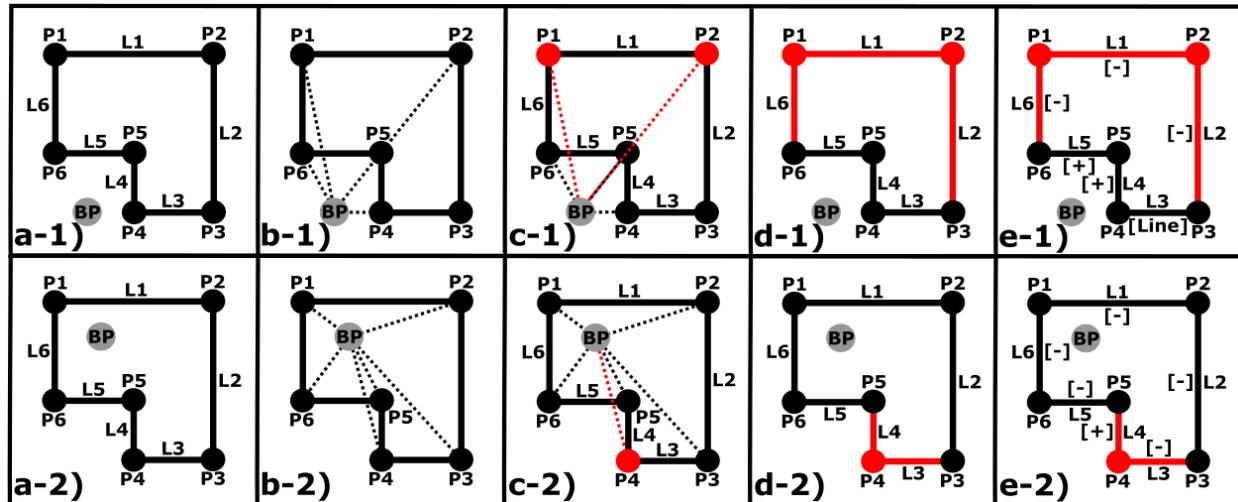


Figure 4.21: Coplanar (COP) BPL-BP Sightline Determination Example

Figure 4.21a-1 and 4.21a-2 show representations for **BP**, as well as for the lines (**BPL_L**) and points (**BPL_P**) defining the boundary of **BPL**.

Figure 4.21b-1 and 4.21b-2 (step 2) show the dotted sightlines (**SL**) generated from **BP** to each **BPL_P** defining **BPL**.

Figure 4.21c-1 and 4.21c-2 (step 3) highlight in red the flagged **SL** and **BPL_P** after evaluating their relationship classifications with all **BPL_L** defining **BPL**. In Figure 4.21c-1, **P1** and **P2** are flagged as $SL(BP \rightarrow P1)$ has a ‘1D-1D PI’ relationship classification with **L5** and $SL(BP \rightarrow P2)$ has ‘1D-0D PT’ relationship classifications with **L4** and **L5**. In Figure 4.21c-2, **P4** is flagged as $SL(BP \rightarrow P4)$ has a ‘1D-1D PI’ relationship classification with **L5**.

Figure 4.21d-1 and 4.21d-2 show the flagged **BPL_P** from step 3, as well as the flagged **BPL_L** lines (step 4) that use any of them as a start or end boundary point in red. In Figure 4.21d-1, **L1**, **L2**, and **L6** are flagged. In Figure 4.21d-2, **L3** and **L4** are flagged.

Figure 4.21e-1 and 4.21e-2 show representations for oriented planeline-point distance results (step 5) along with the flagged points and lines from previous steps. It can be seen in Figure 4.21e-1 that the oriented planeline-point distances from the lines that have not been flagged and are scalar (**L4** and **L5**) are both positive [+]. This results in (2D) **BPL** and (0D) **BP** having the ‘Disjoint’ relationship classification. It can be seen in Figure 4.21e-2 that the oriented planeline-point distances from the lines that have not been flagged and are scalar (**L1**, **L2**, **L5**, and **L6**) are all negative [-]. This results in (2D) **BPL** and (0D) **BP** having the ‘2D-0D PT’ relationship classification.

4.7 Plane-Line (BPL-BL) Relationship Classification Methods

The data process flow algorithm that was developed to classify the relationship **R(BPL,BL)** between a (2D) boundary plane **BPL** and a (1D) boundary line **BL** can be seen in Figure 4.22 below.

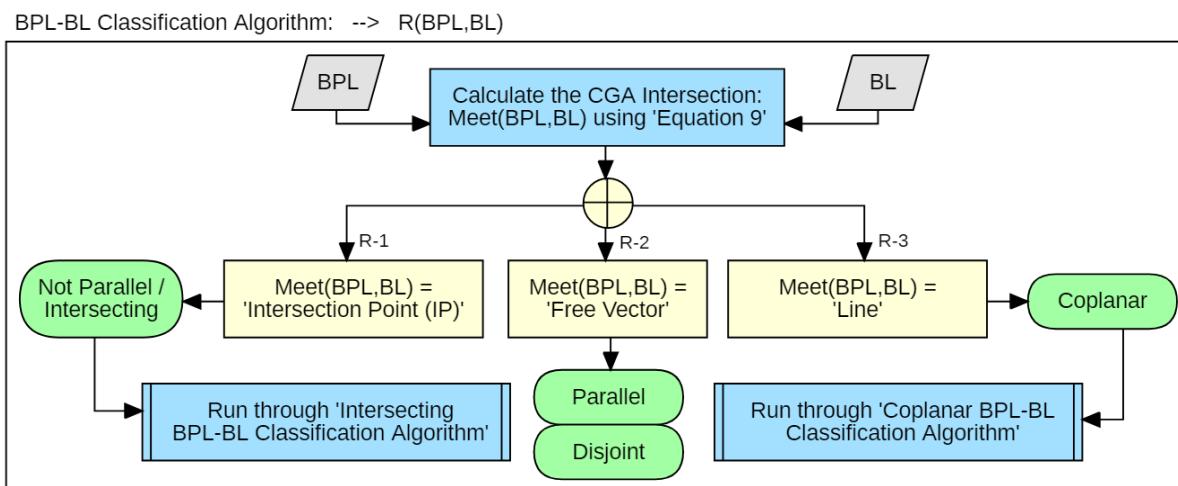


Figure 4.22: BPL-BL Classification Algorithm

This BPL-BL classification algorithm (Figure 4.22) requires calculating the ‘Projected CGA Intersection’ between **BPL** and **BL** using Equation 9. The geometry of this intersection is interpreted to initially determine which of the three projected relationship categories (R-1, R-2, and R-3) is occurring between **BPL** and **BL**. Disjoint examples for each of these BPL-BL projected categories can be seen in Figure 4.23 below.

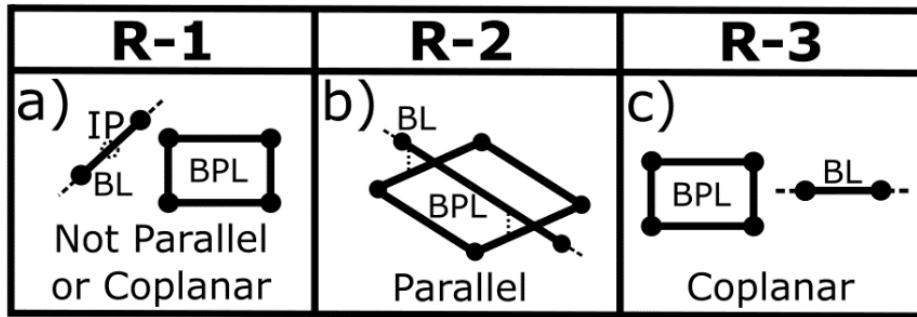


Figure 4.23: BPL-BL Projected Relationship Categories

Figure 4.23a shows an example of the projected relationship category R-1 where **BPL** and **BL** are not parallel or coplanar. The CGA intersection in this scenario produces a ‘Flat Point/Intersection Point’ **IP** geometry result. If scenario R-1 is occurring, **BPL** and **BL** are processed through the ‘Intersecting BPL-BL Classification Algorithm’ (see Figure 4.24). The final relationship classification for this scenario depends on the extent of overlap **BPL** and **BL** each have with **IP** and can result in the ‘Disjoint’, ‘0D-0D Point Touch’, ‘0D-1D Point Touch’, ‘1D-0D Point Touch’, ‘1D-1D Point Touch’, ‘2D-0D Point Touch’, or ‘2D-1D Point Intersection’ relationships.

Figure 4.23b shows an example of the projected relationship category R-2 where **BPL** and **BL** are parallel and are not coplanar. The CGA intersection in this scenario produces a ‘Free Vector’ geometry result. The common geometry between **BPL** and **BL** in this scenario is a ‘free vector’ pointing in their common direction (see Section 3.2.1). Any final relationship classification for this scenario will result in the ‘Disjoint’ relationship.

Figure 4.23c shows an example of the projected relationship category R-3 where **BPL** and **BL** are coplanar. The CGA intersection in this scenario produces a ‘Line’ geometry result. If scenario R-3 is occurring, **BPL** and **BL** are processed through the ‘Coplanar BPL-BL Classification Algorithm’ (see Figure 4.26 and Figure 4.29). The final relationship classification for this scenario depends on the extent of overlap that exists between (2D) **BPL** and the (0D) boundary and (1D) extension of **BL**. **BL** start (**BLs**) and end (**BLE**)

boundary point relationships with **BPL** can be classified as ‘Disjoint’, ‘0D-0D PT’, ‘1D-0D PT’, and ‘2D-0D PT’ while **BL** line extension relationships with **BPL** can be classified as ‘Disjoint’, ‘1D-1D Line Touch’, and ‘2D-1D Line Touch’. If it is determined that **BL** overlaps with the boundary of **BPL** at a point or line, **BL** is separated into ordered line segments using points that have overlap with the boundary of **BPL** as intermediate end points (**BL_i**) separating each line segment. In this scenario, classifications for **BL_i** points can result in the ‘Disjoint’, ‘0D-1D Point Touch’, and ‘1D-1D Point Touch’ relationships.

4.7.1 Intersecting BPL-BL Classification Algorithm

The data process flow algorithm that was developed to classify the relationship $R_{INT}(BPL, BL)$ between (2D) **BPL** and (1D) **BL** that have previously been identified as intersecting (INT) at a projected intersection point **IP** using the algorithm shown in Figure 22 can be seen in Figure 4.24 below.

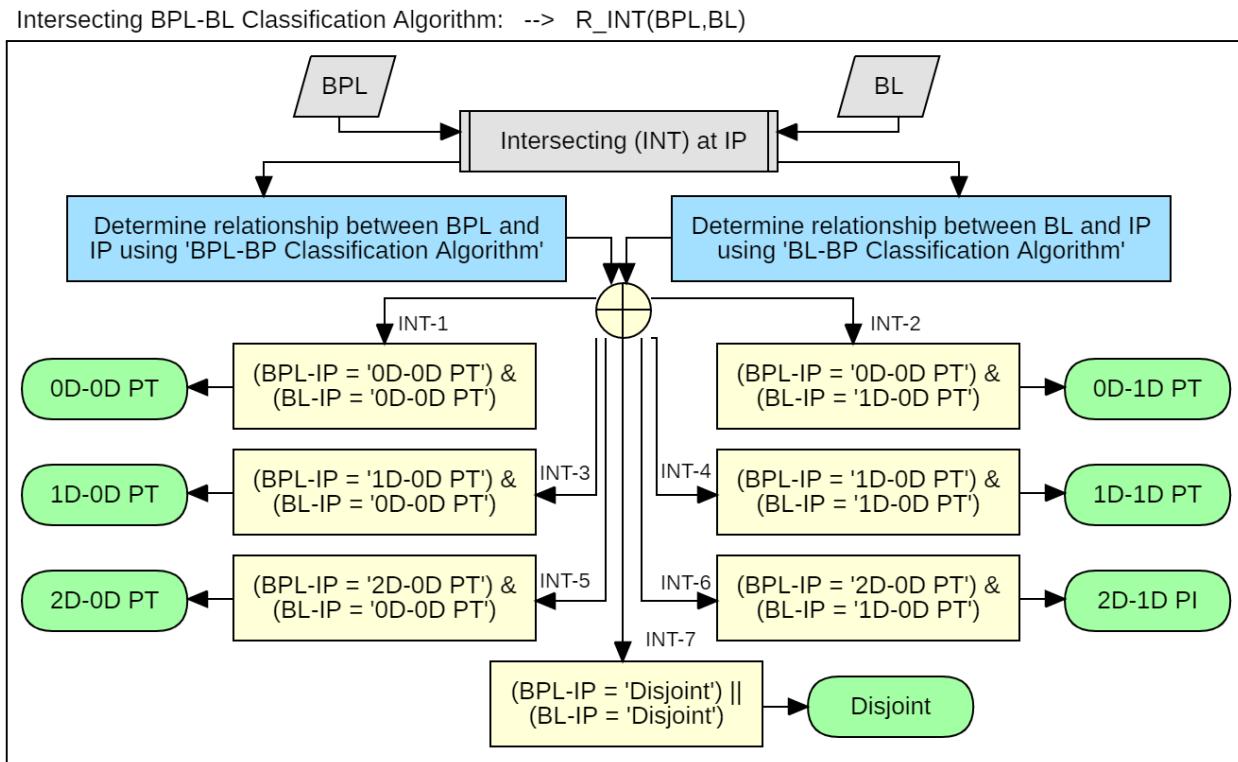


Figure 4.24: Intersecting (INT) BPL-BL Classification Algorithm

It has previously been determined that **BPL** and **BL** intersect at a projected point **IP** using ‘BPL-BL Classification Algorithm’ (see Figure 4.22). This intersecting BPL-BL classification algorithm (Figure 4.24) requires classifying the BPL-BP relationship between **BPL** and **IP** using ‘BPL-BP Classification Algorithm’ (see Figure 4.17) and classifying the BL-BP relationship between **BL** and **IP** using ‘BL-BP Classification Algorithm’ (see Figure 4.5). (BPL-IP) and (BL-IP) are evaluated to determine which of the final seven (INT) relationship classifications is occurring between **BPL** and **BL**. Visual examples showing these seven projected intersection BPL-BL relationship classifications can be seen in Figure 4.25 below.

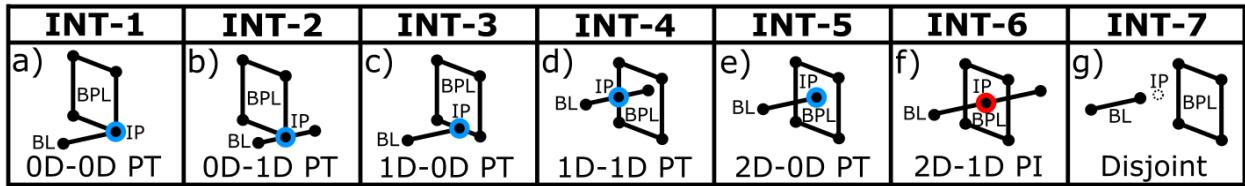


Figure 4.25: Final Intersecting (INT) BPL-BL Relationship Classifications

Figure 4.25a shows an example of the INT-1 scenario where **IP** overlaps **BPL** at an (0D) boundary point and overlaps **BL** at an (0D) boundary point. In this scenario, intersecting (2D) plane **BPL** and (1D) line **BL** have the ‘0D-0D Point Touch’ relationship classification.

Figure 4.25b shows an example of the INT-2 scenario where **IP** overlaps **BPL** at an (0D) boundary point and overlaps **BL** at a (1D) interior extension point. In this scenario, intersecting (2D) plane **BPL** and (1D) line **BL** have the ‘0D-1D Point Touch’ relationship classification.

Figure 4.25c shows an example of the INT-3 scenario where **IP** overlaps **BPL** at a (1D) boundary line and overlaps **BL** at an (0D) boundary point. In this scenario, intersecting (2D) plane **BPL** and (1D) line **BL** have the ‘1D-0D Point Touch’ relationship classification.

Figure 4.25d shows an example of the INT-4 scenario where **IP** overlaps **BPL** at a (1D) boundary line and overlaps **BL** at a (1D) interior extension point. In this scenario, intersecting (2D) plane **BPL** and (1D) line **BL** have the ‘1D-1D Point Touch’ relationship classification.

Figure 4.25e shows an example of the INT-5 scenario where **IP** overlaps **BPL** at a (2D) interior extension point and overlaps **BL** at an (0D) boundary point. In this scenario, intersecting (2D) plane **BPL** and (1D) line **BL** have the ‘2D-0D Point Touch’ relationship classification.

Figure 4.25f shows an example of the INT-6 scenario where **IP** overlaps **BPL** at a (2D) interior extension point and overlaps **BL** at a (1D) interior extension point. In this scenario, intersecting (2D) plane **BPL** and (1D) line **BL** have the ‘2D-1D Point Intersect’ relationship classification.

Figure 4.25g shows an example of the INT-7 scenario where (0D) **IP** does not have any overlap with **BPL** or **BL**. In this scenario, intersecting (2D) plane **BPL** and (1D) line **BL** have the ‘Disjoint’ relationship classification. This classification occurs if either **BPL** or **BL** produces the ‘Disjoint’ relationship with **IP**.

4.7.2 Coplanar BPL-BL Classification Algorithm

The data process flow algorithm that was developed to classify the relationship **R_COP(BPL,BL)** between (2D) **BPL** and (1D) **BL** that have previously been identified as being coplanar (COP) using the algorithm shown in Figure 22 is presented in two parts and can be seen in Figure 4.26 and Figure 4.29 below.

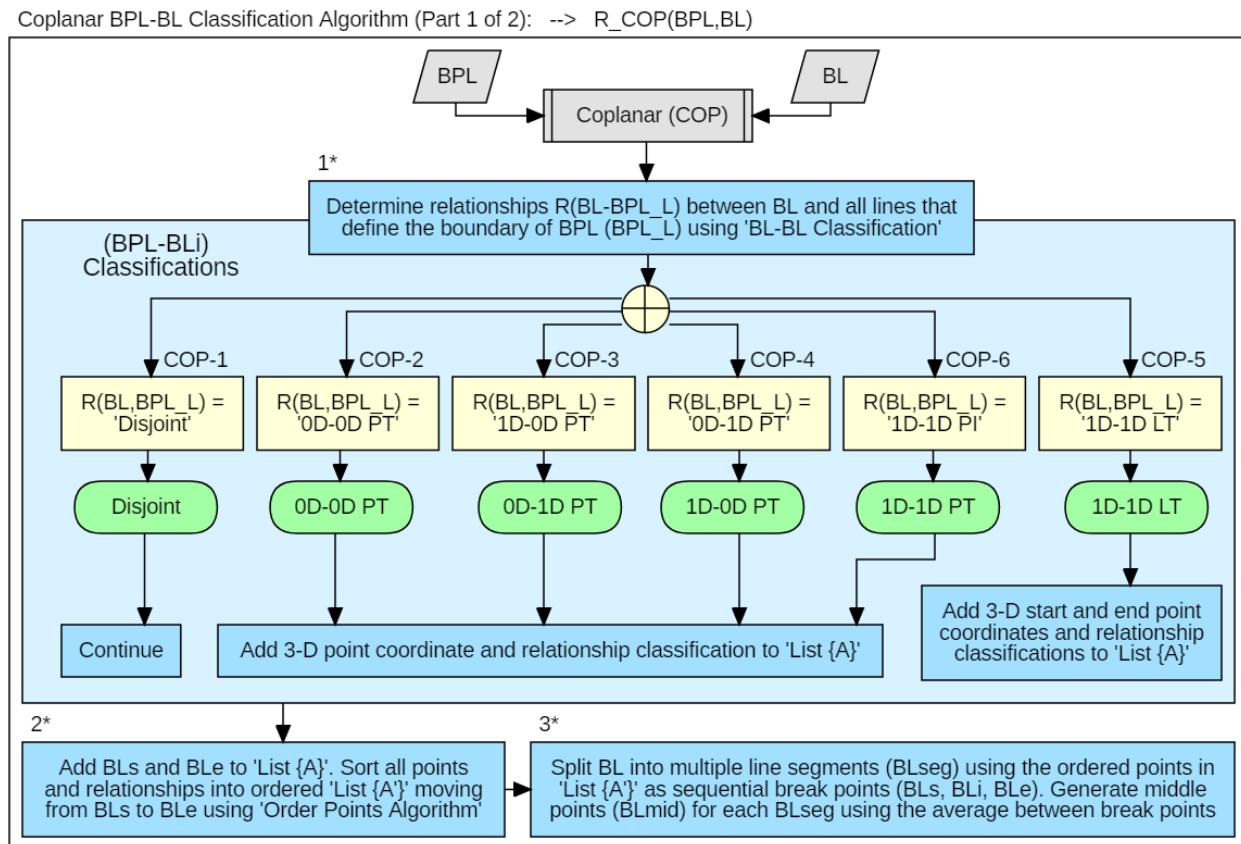


Figure 4.26: Coplanar (COP) BPL-BL Classification Algorithm (Part 1 of 2)

It has previously been determined that **BPL** and **BL** are coplanar using ‘BPL-BL Classification Algorithm’ (see Figure 4.22). The first part of this coplanar BPL-BL classification algorithm (Figure 4.26) determines if any interior (1D) extension points of **BL** (**BLi**) touch or cross the boundary of **BPL**. When this occurs, **BL** is broken down into line segments (**BLseg**) using these interior **BLi** as line break points. This is done so that multiple relationships between **BPL** and different parts of **BL** can be classified separately if they exist. A three-step process is followed to determine if the relationship between coplanar **BPL** and **BL** needs to be broken down into multiple (BPL-BLseg) relationships.

The first step requires classifying the BL-BL relationships between **BL** and all lines (**BPL_L**) that define the boundary of **BPL** using ‘BL-BL Classification Algorithm’ (see Figure 4.9). If any (BL-BPL_L) relationship results in point overlap geometry, the point geometry and relationship classification is added to ‘List A’. If any (BL-BPL_L) relationship results in line overlap geometry, the start and end point geometry of the line along with their relationship classifications are added to point ‘List A’. The cardinality of the relationship is flipped if scenario COP-3 or COP-4 is occurring so that the (BL-BPL_L) classification has the overlap geometry order (BPL-BL). The point intersection between **BL** and **BPL_L** in scenario COP-5 is interpreted as a point touch between **BPL** and **BL**.

The second step is to add the start and end boundary points for **BL** (**BLs** and **BLE**) to ‘List A’. Any point in ‘List A’ having the same geometry as another point is removed so that no duplicate points exist. ‘3-D point-point distances’ are calculated between **BLs** and all other points in ‘List A’ using Equation 10. These other points are sorted into interior line segment break points (**BLi**) by increasing distance away from **BLs** into the order they would exist moving from **BLs** to **BLE**.

The third step is to use the ordered points from **BLs** to **BLE** to create ordered line segments of **BL** (**BLseg**) that will be classified with **BPL** individually. There is only a single line segment **BL** if there are no **BLi** points produced between all **BPL_L** and **BL**. Middle points (**BLmid**) that are used to represent the 1D interior line extensions for each **BLseg** are generated by calculating the average [x, y, z] coordinates for each set of points between **BLs** and **BLE**.

A conceptual example to help explain this **BLseg** breakdown process is presented in Figure 4.27 below. Figure 4.27a shows representations for hypothetical **BL** and its boundary points (**BLs** and **BLE**), as well as for hypothetical **BPL** and its boundary lines (**BPL_L1**, **BPL_L2**, **BPL_L3**, **BPL_L4**) and boundary points (**BPL_P1**, **BPL_P2**, **BPL_P3**, **BPL_P4**).

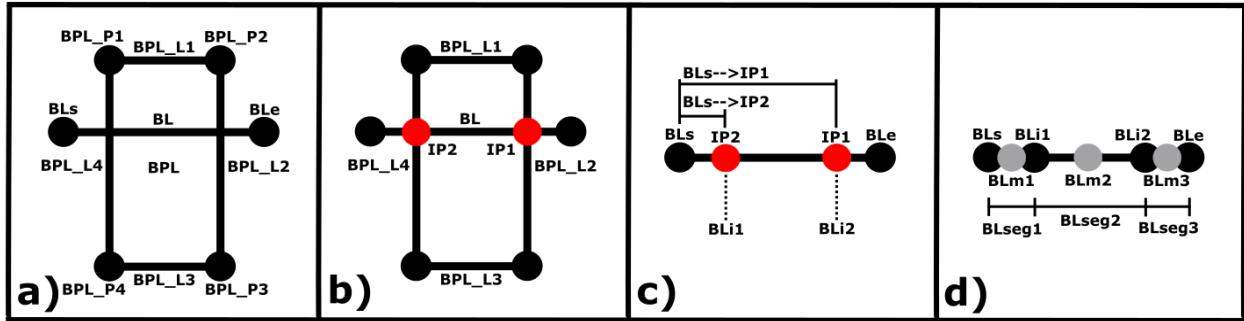


Figure 4.27: Coplanar (COP) BPL-BL Example (Part 1 of 2)

Figure 4.27b (step 1) highlights two intersection points (**IP1** and **IP2**) that would be added to ‘List A’ in red. **IP1** results from ‘1D-1D Point Intersect’ overlap between **BL** and **BPL_L2** while **IP2** results from ‘1D-1D Point Intersect’ overlap between **BL** and **BPL_L4**. These would both be passed through as having the ‘1D-1D Point Touch’ relationship classifications.

Figure 4.27c (step 2) shows representations for distances **BLs**→**IP1** and **BLs**→**IP2**. The distance from **BLs** to **IP2** is less than the distance from **BLs** to **IP1**. For this reason, **IP2** is the first interior break point (**BLi1**) and **IP1** is the second interior break point (**BLi2**). This process can be followed for any number of interior points to create multiple line segments (**BLseg**) of **BL** where a change in topology exists.

Figure 4.27d (step 3) shows how **BL** is divided into line segments (**BLseg1**, **BLseg2**, and **BLseg3**). **BLseg1** has **BLs** and **BLi1** as boundary points and **BLmid1** (**BLm1**) as an interior line extension point. **BLseg2** has **BLi1** and **BLi2** as boundary points and **BLmid2** (**BLm2**) as an interior line extension point. **BLseg3** has **BLi2** and **BLE** as boundary points and **BLmid3** (**BLm3**) as an interior line extension point.

The second part of this coplanar BPL-BL classification algorithm (Figure 4.29) is used to determine the classifications between **BPL** and the (0D) boundary points of **BL**, as well as between **BPL** and the (1D) line extensions for each **BLseg**. These are combined with the existing classifications between **BPL** and **BLi** points to finalize the relationships between **BPL** and **BL**. An overview of the relationships that can exist between **BPL** and each **BLseg** can be seen in Figure 4.28 below.

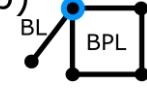
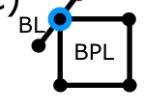
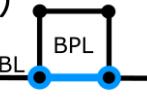
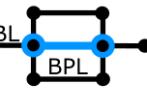
COP-1	COP-2	COP-3	COP-4	COP-5	COP-6
a)  Disjoint	b)  0D-0D PT	c)  0D-1D PT	d)  1D-0D PT	e)  1D-1D LT	f)  2D-1D LT

Figure 4.28: Final Coplanar (COP) BPL-BL Relationship Classifications

Figure 4.28a shows an example of the COP-1 scenario where (2D) plane **BPL** does not have any overlap with (1D) line **BL**. In this scenario, coplanar **BPL** and **BL** have the ‘Disjoint’ relationship classification.

Figure 4.28b shows an example of the COP-2 scenario where a (0D) plane point **BPL_P** touches an (0D) line boundary point **BL_P**. In this scenario, coplanar **BPL** and **BL** have the ‘0D-0D Point Touch’ relationship classification.

Figure 4.28c shows an example of the COP-3 scenario where a (0D) plane point **BPL_P** touches a (1D) line interior point **BLi**. In this scenario, coplanar **BPL** and **BL** have the ‘0D-1D Point Touch’ relationship classification.

Figure 4.28d shows an example of the COP-4 scenario where a (1D) plane line **BPL_L** touches an (0D) line boundary point **BL_P**. In this scenario, coplanar **BPL** and **BL** have the ‘1D-0D Point Touch’ relationship classification.

Figure 4.28e shows an example of the COP-5 scenario where a (1D) plane line **BPL_L** touches a (1D) line extension of **BL** at a line. In this scenario, coplanar **BPL** and **BL** have the ‘1D-1D Line Touch’ relationship classification.

Figure 4.28f shows an example of the COP-6 scenario where a (2D) plane extension of **BPL** touches a (1D) line extension of **BL** at a line. In this scenario, coplanar **BPL** and **BL** have the ‘2D-1D Line Touch’ relationship classification.

The data process flow algorithm that was developed to classify the relationship between (2D) **BPL** and the (0D) boundary points of **BL** (**BLs** and **BLE**) is presented in Figure 4.29a below. The algorithm developed to classify the relationship between (2D) **BPL** and the (1D) line extensions of **BL** (**BLmid**) is presented in Figure 4.29b below.

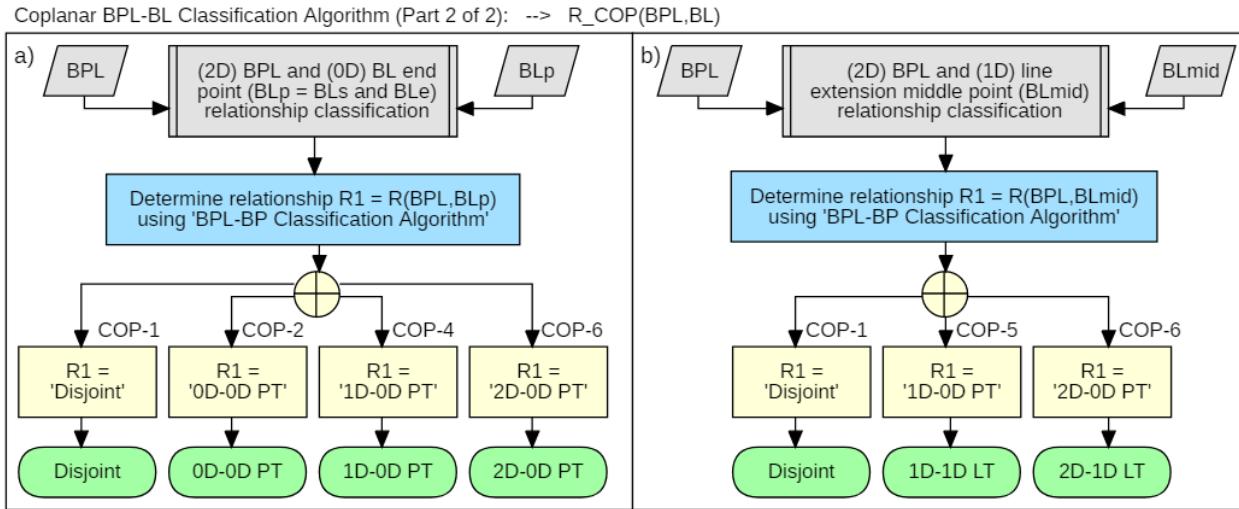


Figure 4.29: Coplanar (COP) BPL-BL Classification Algorithm (Part 2 of 2)

The first step (Figure 4.29a) is used to classify BPL-BP relationships between **BPL** and **BL** boundary points (**BLs**, **BLE**) using ‘Coplanar BPL-BP Classification Algorithm’ (see Figure 4.19). Since **BLs** and **BLE** are (0D) boundary points of **BL**, non-disjoint classifications involving **BPL** and **BL** boundary points (**BLs**, **BLE**) are passed through as they exist and can have the ‘Disjoint’, ‘0D-0D Point Touch’, ‘1D-0D Point Touch’, and ‘2D-0D Point Touch’ relationships.

The second step (Figure 4.29b) is used to classify BPL-BP relationships between **BPL** and all interior middle line extension points (**BLmid**) using ‘BPL-BP Classification Algorithm’ (see Figure 4.17). Since (0D) **BLmid** points are used to represent interior (1D) line extensions of each **BLseg**, non-disjoint classifications between **BPL** and any **BLmid** are modified to reflect this. In addition to modifying (0D) to (1D) dimensionality for the **BLmid** relationship description, all point geometry returns are modified into line geometry returns. Therefore, any initial BPL-BP relationship of ‘1D-0D Point Touch’ and ‘2D-0D Point Touch’ is modified to ‘1D-1D Line Touch’ and ‘2D-1D Line Touch’, respectively.

To help explain this two-step process, the conceptual example from part one is continued in Figure 4.30 below. Figure 4.30a shows **BL** broken down into three line segments (**BLseg1**, **BLseg2**, **BLseg3**) using **BLi1** and **BLi2** as intermediate break points. **BLi1** and **BLi2** have been previously classified as overlapping the boundary of **BPL** producing the ‘1D-1D PT’ relationship classification. A **BLmid** point is used to represent the 1D line extension of each **BLseg**.

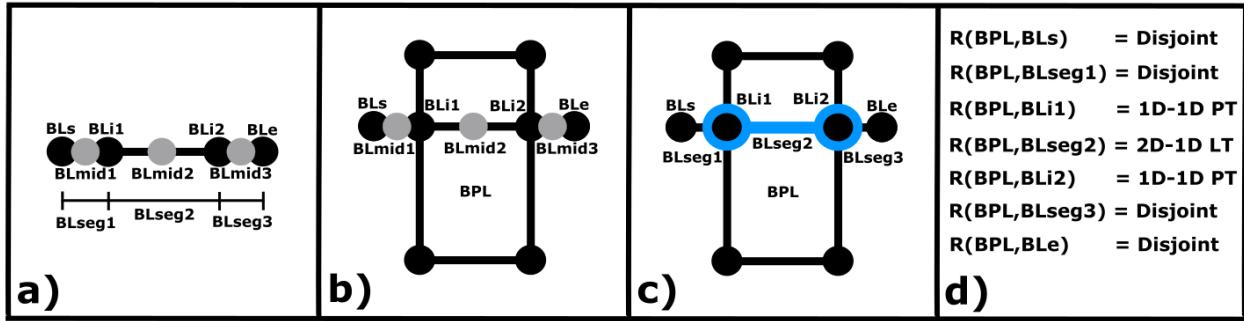


Figure 4.30: Coplanar (COP) BPL-BL Example (Part 2 of 2)

It can be seen in Figure 4.30b that both (BPL-BLs) and (BPL-BLe) BPL-BP classifications (step 1) will produce the ‘Disjoint’ relationship when ran through ‘BPL-BP Classification Algorithm’ (see Figure 4.17). It can also be seen that the (BPL-BLmid1) and (BPL-BLmid3) classifications (step 2) will produce the ‘Disjoint’ relationship when ran through ‘BPL-BP Classification Algorithm’ (see Figure 4.17) resulting in the final (BPL-BLseg1) and (BPL-BLseg3) relationships having the ‘Disjoint’ classification. The (BPL-BLmid2) classification (step 2) will produce the ‘2D-0D Point Touch’ relationship when ran through ‘BPL-BP Classification Algorithm’ (see Figure 4.17) resulting in the final (BPL-BLseg2) relationship having the ‘2D-1D Line Touch’ classification.

Figures 4.30c and 4.30d show the final classifications between **BPL** and each **BLseg** of **BL**. The classification between **BPL** and **BLseg2** has the ‘2D-1D Line Touch’ relationship with both end points (**BLi1** and **BLi2**) classified as overlapping the boundary of **BPL** with the ‘1D-1D Point Touch’ relationship shown highlighted in blue. Relationships between **BPL** and all other components of **BL** produce the ‘Disjoint’ relationship classification as can be seen in Figure 4.30d. This example is equivalent to the COP-6 scenario in Figure 4.28f.

4.8 Plane-Plane (BPL-BPL) Relationship Classification Methods

The data process flow algorithm that was developed to classify the relationship $\mathbf{R}(\mathbf{BPL1}, \mathbf{BPL2})$ between two (2D) boundary planes **BPL1** and **BPL2** can be seen in Figure 4.31 below.

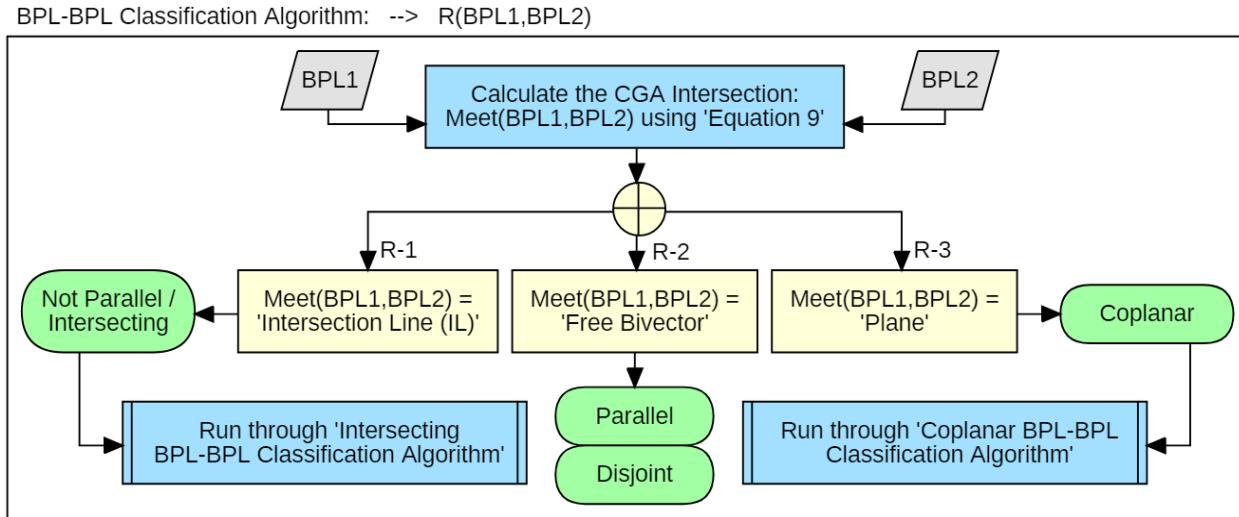


Figure 4.31: BPL-BPL Classification Algorithm

This BPL-BPL classification algorithm (Figure 4.31) requires calculating the ‘Projected CGA Intersection’ between **BPL1** and **BPL2** using Equation 9. The geometry of this intersection is interpreted to initially determine which of the three projected relationship categories (R-1, R-2, and R-3) is occurring between **BPL1** and **BPL2**. Disjoint examples for each of these BPL-BPL projected relationship categories can be seen in Figure 4.32 below.

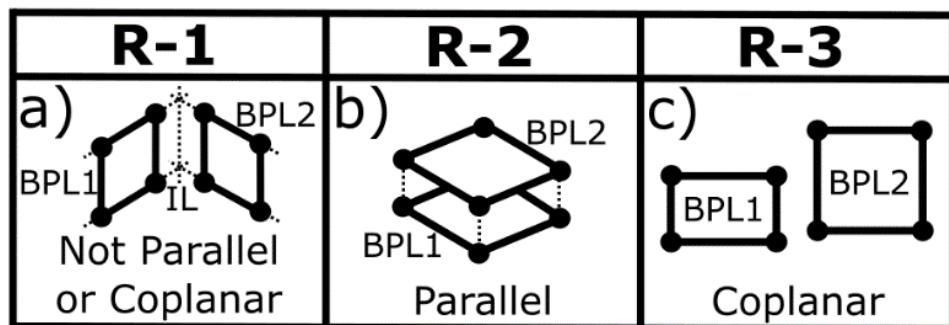


Figure 4.32: BPL-BPL Projected Relationship Categories

Figure 4.32a shows an example of the projected relationship category R-1 where **BPL1** and **BPL2** are not parallel or coplanar. The CGA intersection in this scenario produces an ‘Intersection Line’ (**IL**) geometry result. If scenario R-1 is occurring, **BPL1** and **BPL2** are processed through the ‘Intersecting BPL-BPL Classification Algorithm’ (see Figure 4.33). The final relationship for this scenario depends on the extent of geometry overlap between **BPL1** and **BPL2** and can have the ‘Disjoint’, ‘0D-0D Point Touch’, ‘1D-0D/0D-1D Point Touch’, ‘1D-1D Point Touch’, ‘2D-0D/0D-2D Point Touch’, ‘1D-1D Line Touch’, ‘2D-1D/1D-2D Line Touch’, or ‘2D-2D Line Intersect’ relationship classifications.

Figure 4.32b shows an example of the projected relationship category R-2 where **BPL1** and **BPL2** are parallel. The CGA intersection in this scenario produces a ‘Free Bivector’ geometry result. Any final classification for this scenario will result in the ‘Disjoint’ relationship.

Figure 4.32c shows an example of the projected relationship category R-3 where **BPL1** and **BPL2** are coplanar. The CGA intersection in this scenario produces a common ‘Plane’ geometry result. If scenario R-3 is occurring, **BPL1** and **BPL2** are processed through the ‘Coplanar BPL-BPL Classification Algorithm’ (see Figure 4.36). The final relationship classification for this scenario depends on the extent of geometry overlap between **BPL1** and **BPL2** and can have the ‘Disjoint’, ‘0D-0D Point Touch’, ‘0D-1D Point Touch’, ‘1D-0D Point Touch’, ‘1D-1D Line Touch’, or ‘2D-2D Plane Touch/Intersect’ relationship classifications.

4.8.1 Intersecting BPL-BPL Classification Algorithm

The data process flow algorithm that was developed to classify the relationship **R_INT(BPL1,BPL2)** between two (2D) boundary planes **BPL1** and **BPL2** that have previously been identified as intersecting (INT) at a projected intersection line **IL** using the algorithm shown in Figure 4.31 can be seen in Figure 4.33 below. Any overlap relationship between **BPL1** and **BPL2** will exist as a point or line segment somewhere on projected intersection line **IL**.

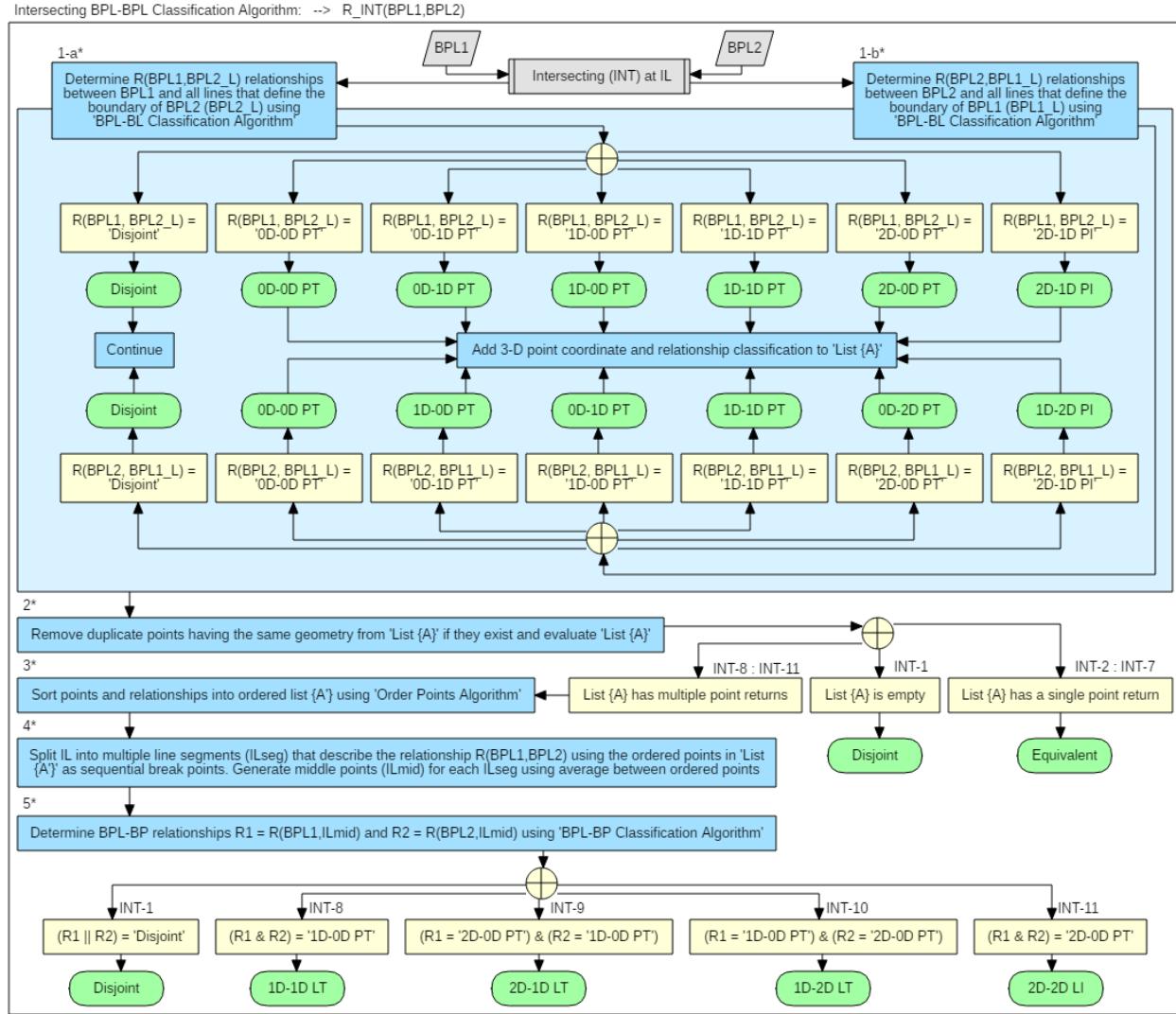


Figure 4.33: Intersecting (INT) BPL-BPL Classification Algorithm

It has previously been determined that **BPL1** and **BPL2** intersect at a projected line **IL** using ‘BPL-BPL Classification Algorithm’ (see Figure 4.31). Any relationship between **BPL1** and **BPL2** that is not disjoint will exist on **IL**. This intersecting BPL-BPL classification algorithm (Figure 4.33) first determines if any (0D or 1D) boundary points of **BPL1** overlap **BPL2** and if and (0D or 1D) boundary points of **BPL2** overlap **BPL1** at break points (**ILb**) on **IL**. If multiple **ILb** break points exist between **BPL1** and **BPL2**, the final relationship between both planes is described using closed line segments on **IL** (**ILseg**) that use these break points as boundary points. This is done so that multiple relationships between **BPL1** and **BPL2** can be classified separately if they exist. An overview of the relationships that can exist between **BPL1** and **BPL2** can be seen in Figure 4.34 below.

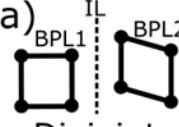
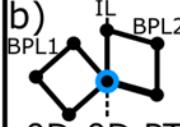
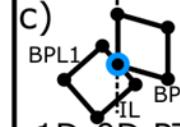
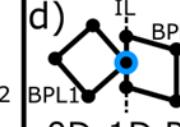
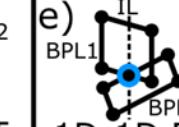
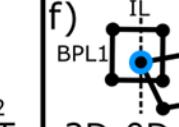
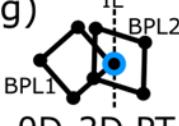
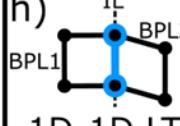
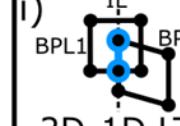
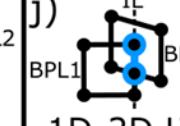
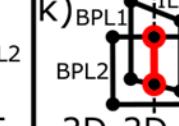
INT-1	INT-2	INT-3	INT-4	INT-5	INT-6
a)  Disjoint	b)  0D-0D PT	c)  1D-0D PT	d)  0D-1D PT	e)  1D-1D PT	f)  2D-0D PT
INT-7	INT-8	INT-9	INT-10	INT-11	
g)  0D-2D PT	h)  1D-1D LT	i)  2D-1D LT	j)  1D-2D LT	k)  2D-2D LI	

Figure 4.34: Final Intersecting (INT) BPL-BPL Relationship Classifications

Figure 4.34a shows an example of the INT-1 scenario where **BPL1** and **BPL2** have no overlap producing the ‘Disjoint’ relationship classification.

Figure 4.34b shows an example of the INT-2 scenario where an (0D) boundary point of **BPL1** touches an (0D) boundary point of **BPL2** producing the ‘0D-0D Point Touch’ relationship classification.

Figure 4.34c shows an example of the INT-3 scenario where a (1D) boundary point of **BPL1** touches an (0D) boundary point of **BPL2** producing the ‘1D-0D Point Touch’ relationship classification.

Figure 4.34d shows an example of the INT-4 scenario where an (0D) boundary point of **BPL1** touches a (1D) boundary point of **BPL2** producing the ‘0D-1D Point Touch’ relationship classification.

Figure 4.34e shows an example of the INT-5 scenario where a (1D) boundary point of **BPL1** touches a (1D) boundary point of **BPL2** producing the ‘1D-1D Point Touch’ relationship classification.

Figure 4.34f shows an example of the INT-6 scenario where a (2D) interior extension point of **BPL1** touches an (0D) boundary point of **BPL2** producing the ‘2D-0D Point Touch’ relationship classification.

Figure 4.34g shows an example of the INT-7 scenario where an (0D) boundary point of **BPL1** touches a (2D) interior extension point of **BPL2** producing the ‘0D-2D Point Touch’ relationship classification.

Figure 4.34h shows an example of the INT-8 scenario where a (1D) boundary line of **BPL1** touches a (1D) boundary line of **BPL2** producing the ‘1D-1D Line Touch’ relationship classification.

Figure 4.34i shows an example of the INT-9 scenario where a (2D) interior extension line of **BPL1** touches a (1D) boundary line of **BPL2** producing the ‘2D-1D Line Touch’ relationship classification.

Figure 4.34j shows an example of the INT-10 scenario where a (1D) boundary line of **BPL1** touches a (2D) interior extension line of **BPL2** producing the ‘1D-2D Line Touch’ relationship classification.

Figure 4.34k shows an example of the INT-11 scenario where a (2D) interior extension line of **BPL1** touches a (2D) interior extension line of **BPL2** producing the ‘2D-2D Line Intersection’ relationship classification.

A five-step process is followed to determine which of the 11 INT scenarios is occurring between **BPL1** and **BPL2**. The first step requires classifying and evaluating the BPL-BL relationships between **BPL1** and all lines (**BPL2_L**) that define the boundary of **BPL2**, as well as the relationships between **BPL2** and all lines (**BPL1_L**) that define the boundary of **BPL1** using ‘BPL-BL Classification Algorithm’ (see Figure 4.22). The dimensional cardinality for all BPL2-BPL1_L relationships are reversed so that the relationship is consistent with the BPL1-BPL2_L relationships and is described with respect to **BPL1** geometry overlap first. If any of these BPL-BL relationships result in point overlap, the point geometry and relationship classification is added to ‘List A’.

The second step is to remove duplicate points that have the same location geometry and classification from ‘List {A}’ if any exist and to evaluate ‘List {A}’. The INT-1 scenario that produces the ‘Disjoint’ relationship classification is occurring if ‘List A’ remains empty. If ‘List A’ only has a single point return in it, the relationship classification associated with this point with respect to **BPL1** is the final relationship classification between **BPL1** and **BPL2**. This can result in the (INT-2, INT-3, INT-4, INT-5, INT-6, INT-7) relationship classifications.

Steps 3 -5 are only followed if ‘List {A}’ has multiple point returns in it. The third step is to sort the points in ‘List {A}’ into an ordered ‘List {A}’. To do this, the ‘3-D point-point distances’ between all points in ‘List A’ are calculated using Equation 10. The point pair with the largest distance between them will be used as the start and end points (**ILs** and **ILe**) on **IL** that constrain the relationship between **BPL1** and **BPL2**. All other points from ‘List A’ are sorted into intermediate line segment break points (**ILI**) by increasing distances away from **ILs** into the order they appear moving from **ILs** to **ILe**. This is like the methods described in Section 4.7.2 above.

The fourth step is to use the ordered points from **ILs** to **ILe** to create ordered line segments (**ILseg**) existing on **IL**. Middle points (**ILmid**) representing the 1D interior line extensions for each **ILseg** are generated by calculating the average [x, y, z] coordinates for each set of points between **ILs** and **ILe**.

The fifth step requires classifying and evaluating the relationship between **ILmid** point and **BPL1**, as well as the relationship between **ILmid** and **BPL2** for each **ILmid** point using ‘BPL-BP Classification Algorithm’ (see Figure 4.17). These two results for each **ILmid** point are interpreted to produce the ‘Disjoint’, 1D-1D Line Touch’, ‘2D-1D Line Touch’, ‘1D-2D Line Touch, or ‘2D-2D Line Intersection’ relationship classification.

A conceptual example to help explain this process is presented in Figure 4.35 below. Figure 4.35a shows representations for hypothetical **BPL1** (**A**) and **BPL2** (**B**) that have overlap somewhere on their projected intersection line **IL**.

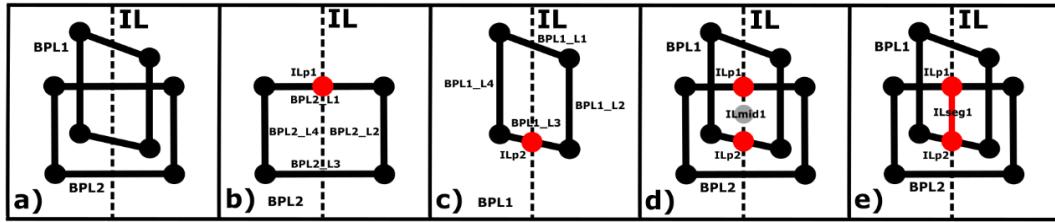


Figure 4.35: Intersecting (INT) BPL-BPL Example

Figure 4.35b shows **BPL2** with its set of boundary lines (**BPL2_L**) and set of boundary points (**BPL2_P**). After evaluating relationships between **BPL1** and all **BPL2_L** (step 1-a in Figure 4.33), the relationship between **BPL1** and **BPL2_L1** produces the ‘2D-1D Point Intersection’ classification at point **ILp1**. The cardinality of this relationship is preserved. Figure 4.35c shows **BPL1** with its boundary lines (**BPL1_L**) and boundary points (**BPL1_P**). After evaluating relationships between **BPL2** and all **BPL1_L** (step 1-b in Figure 4.33), the relationship between **BPL2** and **BPL1_L3** produces the ‘2D-1D Point Intersection’ classification. The cardinality of this relationship is reversed so that the final relationship added to ‘List A’ is the ‘1D-2D Point Intersect’ classification at point **ILp2**. Steps 2 and 3 can be skipped in this example since there are only two unique overlap points (**ILp1** and **ILp2**). Figure 4.35d shows these as the end points of a single line segment on **IL** where overlap is occurring. A middle point (**ILmid1**) is generated (step 4) to represent the interior extension of the single line segment that has **ILp1** and **ILp2** as end points. Both the **R(BPL1,ILmid1)** and **R(BPL2,ILmid1)** BPL-BP relationships produce the ‘2D-0D Point Touch’ relationship classification. This results in the final classification between **BPL1** and **ILseg1** having the ‘2D-2D Line Intersection’ relationship classification. This relationship can be seen highlighted in red in Figure 4.35e.

4.8.2 Coplanar BPL-BPL Classification Algorithm

The data process flow algorithm that was developed to classify the relationship **R_COP(BPL1,BPL2)** between two (2D) boundary planes **BPL1** and **BPL2** that have previously been identified as being coplanar (COP) using the algorithm shown in Figure 4.31 can be seen in Figure 4.36 below.

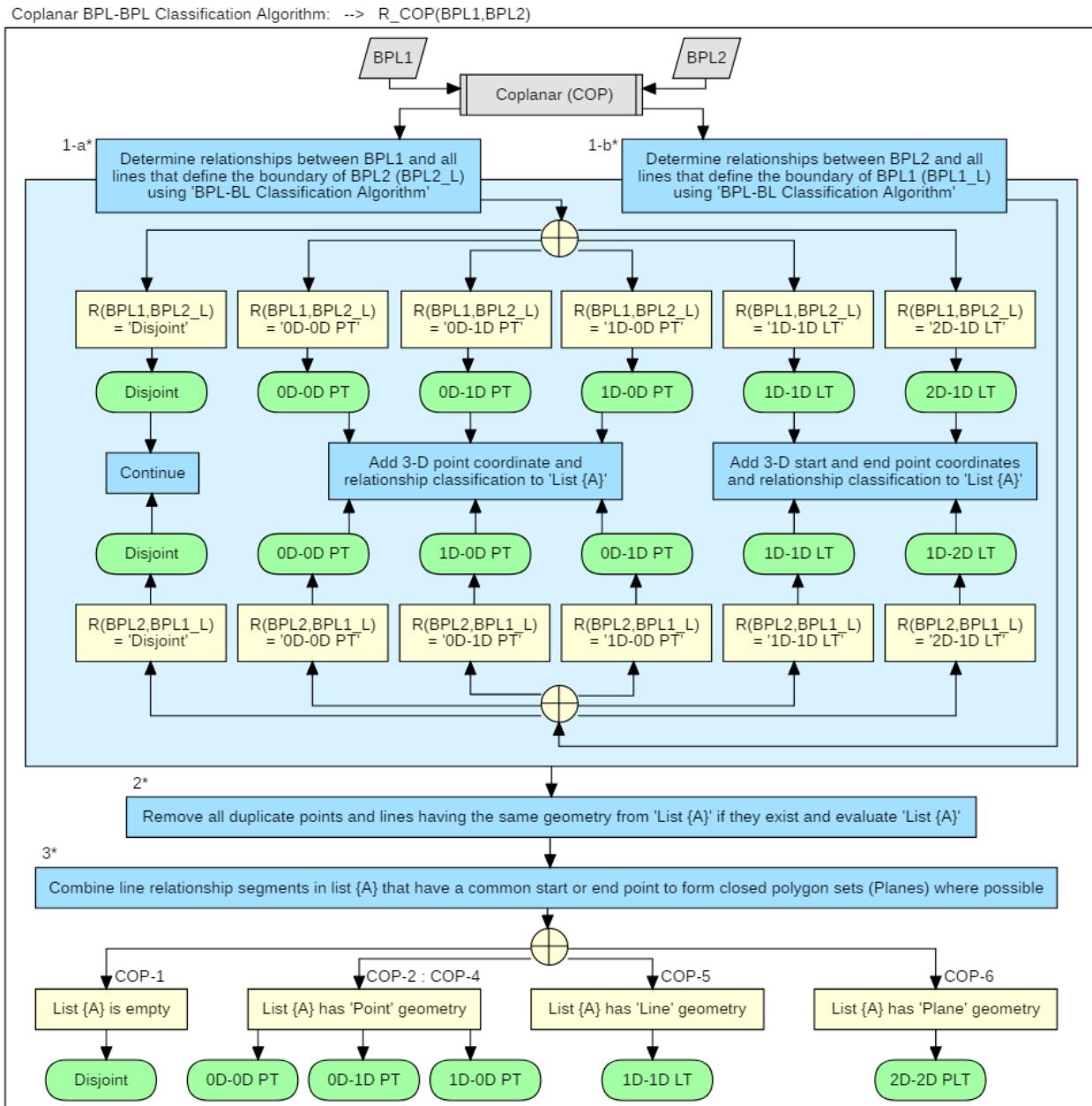


Figure 4.36: Coplanar (COP) BPL-BPL Classification Algorithm

It has previously been determined that **BPL1** and **BPL2** are coplanar using ‘BPL-BPL Classification Algorithm’ (see Figure 4.31). This coplanar BPL-BPL classification algorithm (Figure 4.36) requires first classifying the BPL-BL relationships between **BPL1** and all lines that define the boundary of **BPL2**, as well as classifying the relationships between **BPL2** and all lines that define the boundary of **BPL1** using ‘BPL-BL Classification Algorithm’ (see Figure 4.22). All non-disjoint relationships having point or line geometry are added to a ‘List {A}’.

Points or lines that have the same geometry and classification as another are removed from ‘List {A}’ so that there are no duplicates. Lines are combined by identifying common end points between lines to form closed polygons where possible. The final classification between **BPL1** and **BPL2** is grouped into relationships having point, line, or plane geometry types. Visual examples showing six coplanar BPL-BPL relationship classifications can be seen in Figure 4.37 below.

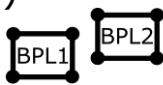
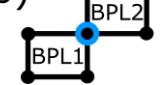
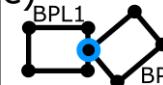
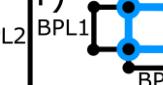
COP-1	COP-2	COP-3	COP-4	COP-5	COP-6
a)  Disjoint	b)  0D-0D PT	c)  1D-0D PT	d)  0D-1D PT	e)  1D-1D LT	f)  2D-2D PLT

Figure 4.37: Final Coplanar (COP) BPL-BPL Relationship Classifications

Figure 4.37a shows an example of the COP-1 scenario where **BPL1** and **BPL2** have no overlap producing the ‘Disjoint’ relationship classification.

Figure 4.37b shows an example of the COP-2 scenario where an (0D) boundary point of **BPL1** touches an (0D) boundary point of **BPL2** producing the ‘0D-0D Point Touch’ relationship classification.

Figure 4.37c shows an example of the COP-3 scenario where a (1D) boundary point of **BPL1** touches an (0D) boundary point of **BPL2** producing the ‘1D-0D Point Touch’ relationship classification.

Figure 4.37d shows an example of the COP-4 scenario where an (0D) boundary point of **BPL1** touches a (1D) boundary point of **BPL2** producing the ‘0D-1D Point Touch’ relationship classification.

Figure 4.37e shows an example of the COP-5 scenario where a (1D) boundary line of **BPL1** touches a (1D) boundary line of **BPL2** producing the ‘1D-1D Line Touch’ relationship classification.

Figure 4.37f shows an example of the COP-6 scenario where a (2D) interior plane extension of **BPL1** touches a (2D) interior plane extension of **BPL2** producing the ‘2D-2D Plane Touch’ relationship classification.

4.9 Chapter Summary

The relevance this chapter has to the primary research objective is that it presents the processes and algorithms that were designed to classify geometrical and topological relationships between two 3-D cadastral units and their boundary components. This chapter was used to perform research activity #3: “Decide on a data format that will be used for digitally representing 3-D boundaries” and research activity #6: “Propose data flow processes that use existing mathematical algorithms and techniques to classify topological relationships that can exist between the 3-D boundary components of cadastral units”.

Section 4.1 defined the table storage format for digitally representing 3-D cadastral unit boundary components as they are modelled in this study. Boundary points, boundary lines, and boundary planes were defined using 3-D point geometry, internal point and line set topology combinations, and CGA multivector blade parameters. Datasets can be created by interpreting measurements in survey plan drawings. A more detailed explanation of how to generate these parameters is presented in Chapter 5.

Section 4.2 presented the mathematical approach followed to model relationships between 3-D cadastral units. Relationships between six types of boundary component pairs are interpreted together to describe the relationship occurring between the two 3-D cadastral units. Relationships are described using the returned geometry associated with the relationship along with four descriptive characteristics.

Sections 4.3 – 4.8 presented the developed data flow processes and algorithms that were proposed to classify relationships between 3-D cadastral unit boundary component pairs. Relationship classification algorithms were put forward for each of the six types of boundary component pairs considered here that include main and secondary classification functions. The main functions were developed to identify which projected relationship was occurring between boundary component pair sets (i.e. collinear, coplanar, parallel, intersecting, etc.). The secondary functions were developed for each projected relationship scenario to evaluate the final relationship classifications that can exist between boundary component pairs.

The approach followed to implement these algorithms using written MATLAB functions in combination with GAViewer files is presented next in Chapter 5 of this study. The algorithms presented here are later validated in Chapter 6 using simulated datasets and applied to a real cadastral dataset derived from a 3-D condominium survey plan in Alberta, Canada to apply them to a practical 3-D boundary problem.

Chapter Five: Implementation of Methods

This chapter presents the methods and functions that were used to implement and test the relationship classification modelling theory and algorithms that were developed and presented in Chapter 4. These functions are applied in Chapter 6 to evaluate and present relationships that can be classified between the boundaries of sets of two volumetric 3-D cadastral units by evaluating relationships between their lower-dimensional (0-D) boundary points, (1-D) boundary lines, and (2-D) boundary planes.

All functions were written in MATLAB. Specific MATLAB functions were written to generate and read files that were executed in GAViewer. GAViewer is an open-source geometric algebra program that was used to implement all CGA computational operations on 3-D cadastral boundary components. More information on GAViewer can be found in the program documentation at (Fontijne, 2010). GAViewer (.geo) object files are used to store CGA objects that can be loaded into GAViewer and (.g) operations files are used to store mathematical operations and commands that can be implemented on CGA objects in GAViewer. MATLAB functions were written to generate and read GAViewer (.geo) object and (.g) operation files so that data could be indirectly shared between the two programs. This was utilized to generate CGA parameters for boundary components and to perform spatial analysis between them. The chapter is structured as follows:

Section 5.1 describes the processes followed and the functions that were written to generate digital representations for 3-D cadastral object boundary components. In brief, digital parameters for boundary points, lines, and planes of simulated 3-D cadastral units were stored in tables in EXCEL files. These tables include 3-D geometric coordinates for boundary points, start and end point numbers for boundary lines, and lists of ordered line numbers for boundary planes. Additional CGA parameters were generated using GAViewer that were then used to perform geometric and topological operations between boundary points, lines, and planes.

Section 5.2 describes the processes followed and the functions that were written to generate and store CGA geometric and topological intersection results between 3-D boundary component pairs. The CGA parameters for the boundary points, lines, and planes of two individual 3-D cadastral units were written to a GAViewer (.geo) object file. A list of GAViewer function commands between all boundary component pairs was then written to a GAViewer (.g) operation file. The (.geo) object file containing representations for the boundary components and the (.g) operation file containing function commands were then loaded into GAViewer and executed. The geometric and topological intersection results (see Table 5.1) were exported as a new (.geo) object file containing the parameters of the boundary components for both 3-D cadastral units.

Section 5.3 describes the processes followed and the functions that were written to interpret the geometric and topological intersection checks between 3-D cadastral objects boundary components and to finalize the relationship classifications between boundary component pairs. Intersection results were interpreted between point-point, line-point, line-line, plane-point, plane-line, and plane-plane component pairs in MATLAB to determine which projected relationship category (i.e. parallel, collinear, coplanar, point intersection, line intersection, etc.) was occurring between boundary component pairs. A combination of point-point distance and additional CGA topological intersection checks are evaluated and interpreted using MATLAB functions and GAViewer files to reach the final relationship classifications for each of the six types of boundary component pair sets (point-point, line-point, line-line, plane-point, plane-line, and plane-plane) that exist between the boundaries of two 3-D cadastral units.

5.1 Creating Parameters for Testing Datasets (Building Objects from Coordinate Sets)

Each testing dataset used in Chapter 6 consists of two volumetric 3-D cadastral units. Each 3-D cadastral unit is composed of lower-dimensional (point, line, and plane) boundary components as they are defined in Section 4.1. Before relationships can be classified between the two units, digital parameters for their lower-dimensional boundary components were first generated and stored in EXCEL file tables for boundary points, boundary lines, and boundary planes.

For each 3-D cadastral unit boundary tested in Chapter 6, a two-step process was followed to input initial 3-D boundary point [x, y, z] geometry coordinates, to establish point set topology for boundary lines and boundary planes, and to generate additional CGA parameters for the boundary points, boundary lines, and boundary planes. The process uses a combination of written MATLAB functions and generated GAViewer CGA files to generate digital 3-D boundary components and their additional projected CGA parameters, as well as to add them to the final EXCEL file table storage structure for each 3-D cadastral unit. A general overview of the parameters used to define each 3-D boundary component and the processes used to generate them can be seen in Figure 5.1 below.

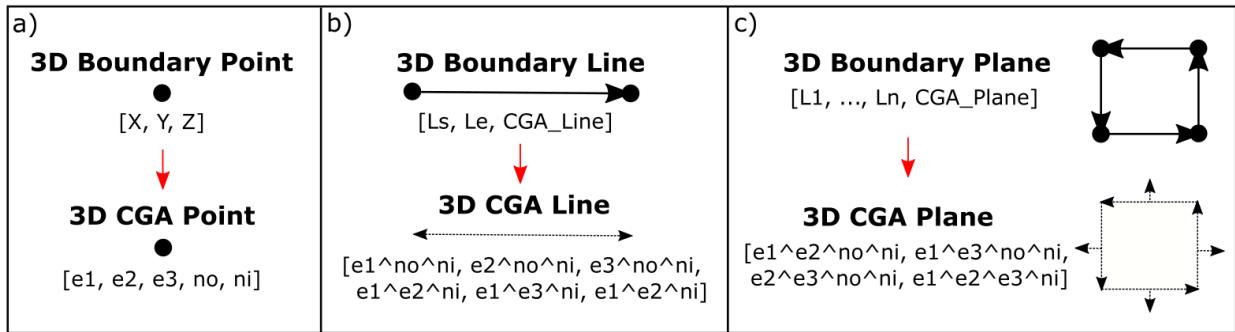


Figure 5.1: Boundary Component Limits, Extensions, and CGA Representations

Figure 5.1 shows how each boundary component type is defined with respect to their point set topology having boundary extensions and boundary limits as they are described with respect to the Dimensional Model presented in Section 3.1.2. Additional CGA representations for how these objects are modelled in GAViewer are shown here as well. The relationship between the extensions of any two 3-D boundary components was initially used to sort the relationship between a boundary component pair into one of the 16 projected categories introduced in Figure 3.5.

Figure 5.1a shows that 3-D boundary points have (x, y, z) parameters that represent their 0-D extensions with no limits. Five CGA 1-blade multivector parameters (e_1, e_2, e_3, n_o, n_i) are used to model 3-D CGA point extensions in GAViewer. Figure 5.1b shows that 3-D boundary lines have start and end point ID parameters (L_s, L_e) that represent their 0-D boundary limits. An additional 3-D CGA line is used to represent their 1-D boundary extensions. Six CGA 3-blade multivector parameters ($e_1 \wedge n_o \wedge n_i, e_2 \wedge n_o \wedge n_i, e_3 \wedge n_o \wedge n_i, e_1 \wedge e_2 \wedge n_i, e_1 \wedge e_3 \wedge n_i, e_2 \wedge e_3 \wedge n_i$) are used to model 3-D CGA line extensions in GAViewer. Figure 5.1c shows that 3-D boundary planes have ordered multi-line ID parameters (L_1, \dots, L_n) that represent their 0-D and 1-D boundary limits. An additional 3-D CGA plane is used to represent their 2-D boundary extensions. Four CGA 4-blade multivector parameters ($e_1 \wedge e_2 \wedge n_o \wedge n_i, e_1 \wedge e_3 \wedge n_o \wedge n_i, e_2 \wedge e_3 \wedge n_o \wedge n_i, e_1 \wedge e_2 \wedge e_3 \wedge n_i$) are used to model 3-D CGA plane extensions in GAViewer.

The first step in the 3-D cadastral unit generation process was to define the geometry related to 3-D boundary point locations and to define the point and line topology sets associated with the limits of boundary lines and planes, respectively. This process is discussed in Section 5.1.1. The second step in the 3-D cadastral unit generation process was to generate the CGA parameters for all null vector CGA points, projected CGA lines, and projected CGA planes in GAViewer using generated (.geo) object files and (.g) operation files. Results were exported from GAViewer as an updated (.geo) object file and loaded into MATLAB where the parameters were parsed, sorted, and stored back into the EXCEL file table storage structure for each 3-D cadastral unit. This process is discussed in Section 5.1.2.

5.1.1 Preparing Initial 3-D Cadastral Unit Boundary Storage Tables

All 3-D cadastral units tested in experiments were watertight closed-boundary cube-like objects, each consisting of 6 boundary planes. Each boundary plane is contained by 4 boundary lines. Each 3-D cadastral unit has 24 boundary lines, each of which is contained by 2 boundary points. Each 3-D cadastral unit has 8 boundary points, each of which is defined by 3-D (x, y, z) coordinates. An overview of the 3-D cadastral units matching this description that were tested in experiments can be seen in Figure 6.3 and Figure 6.21.

A template EXCEL file was created that contained three sheets for boundary points, lines, and planes, respectively. All 3-D cadastral unit datasets were initially created by inserting the 3-D boundary point coordinates, the point sets representing boundary line topology, and the line sets representing boundary plane topology into the tables of the EXCEL file template. Preparing these EXCEL tables was executed in three steps.

The first step was to identify and input the [x, y, z] coordinates for each of the 8 boundary points associated with the boundary of each 3-D cadastral unit being modelled. In a survey plan, these boundary point locations can be calculated using direct (polar) calculations involving bearings (angles) and distances in the plan from a previously known location. If tracing a survey plan, these direct calculations would start from a known control or reference point. For the 7 simulated datasets used in Chapter 6, boundary points and their coordinates for each 3-D cadastral unit were established *a priori* to processing. For the real cadastral dataset used in Chapter 6 derived from a condominium survey plan registered in Alberta, Canada, the point coordinates were calculated using measurements included in the survey plan. A table showing a sample of boundary points with their respective 3-D coordinates was introduced in Table 4.1.

The second step was to identify and input the start and end boundary point sets that describe the topology for each of the 24 boundary lines associated with the boundary of each 3-D cadastral unit being modelled. Because the 3-D cadastral units tested here have closed boundaries, each boundary line defined has a symmetric counterpart. What is meant by this is if a boundary line is defined using boundary point 1 as a start point and boundary point 2 as an end point, another boundary line will be defined using boundary point 2 as a start point and boundary point 1 as an end point. Each of the boundary lines in the above example is used to define two different boundary planes that meet at those lines. A table showing a sample of boundary lines with their respective start and end point IDs was introduced in Table 4.2.

The third step was to identify the ordered boundary polyline sets, each containing 4 lines that describe the topology for each of the 6 boundary planes associated with the boundary of each 3-D cadastral unit being modelled. The order that the boundary lines are stored for each boundary plane describes and enforces

clockwise rotation when viewing the boundary plane from inside the 3-D cadastral unit. A table showing a sample of boundary planes with their respective oriented polyline IDs was introduced in Table 4.3.

5.1.2 Generating CGA Parameters for 3-D Boundary Components using GAViewer

A MATLAB function was written that loads in data from the EXCEL file template and writes GAViewer (.geo) object and (.g) operation files. These files were then opened and executed in GAViewer to generate the CGA parameters associated with all boundary components for each 3-D cadastral unit and the parameters were updated back into the EXCEL file template. Figure 5.2 below shows the five-step process that was followed to achieve this.

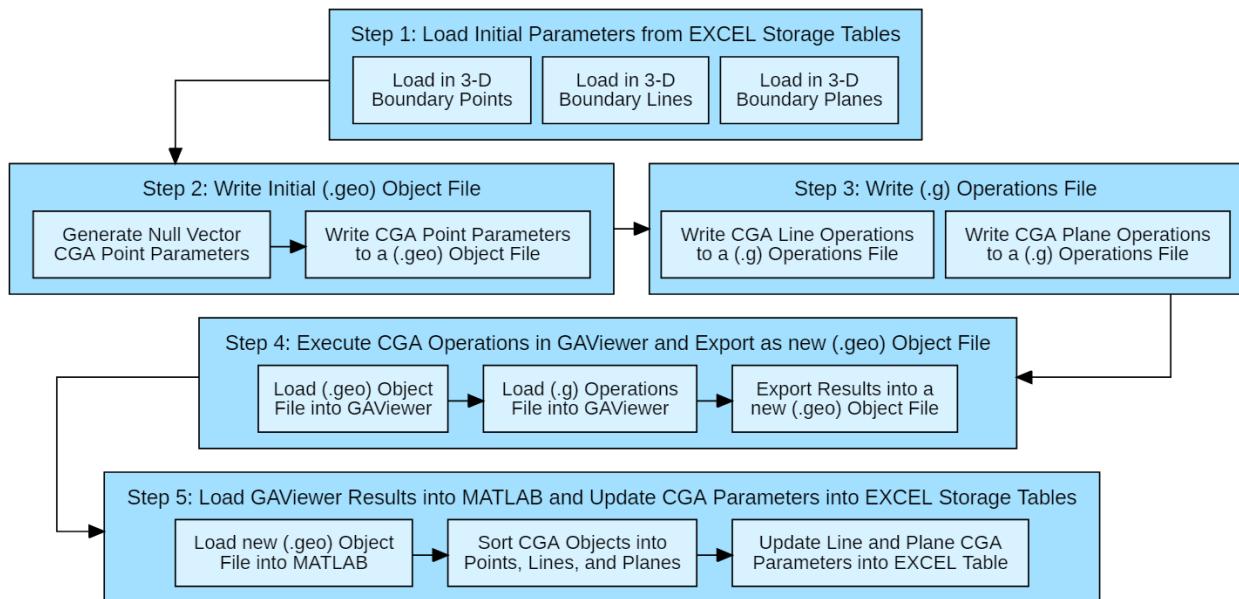


Figure 5.2: Process Flow for Generating and Storing 3-D Boundary Component CGA Parameters

The first step was for the MATLAB function to load in the point IDs and [x, y, z] coordinates of the eight boundary points, the line IDs and the start point and end point IDs associated with each boundary line, and the plane IDs and the oriented polyline IDs associated with each boundary plane from the EXCEL file template that was filled in using the methods described in Section 5.1.1 above.

The second step was for the MATLAB function to generate and store the null vector point extension CGA 1-blade multivector parameters (e_1, e_2, e_3, n_o, n_i) for each of the 8 boundary points, where $n_o = 1$ and n_i was calculated using Equation 4 (see Section 3.2.1). These boundary point extension parameters were then written to a GAViewer (.geo) object file and the n_o and n_i parameters for each point were updated back into the boundary point table in the EXCEL file template.

The third step was for the MATLAB function to write the GAViewer operations that are used to generate the projected line extension CGA 3-blade multivector parameters ($e_1 \wedge n_o \wedge n_i, e_2 \wedge n_o \wedge n_i, e_3 \wedge n_o \wedge n_i, e_1 \wedge e_2 \wedge n_i, e_1 \wedge e_3 \wedge n_i, e_2 \wedge e_3 \wedge n_i$) for each of the 24 boundary lines to a (.g) operation file. This was done using the start point and end point IDs associated with each boundary line and Equation 5 (see Section 3.2.1). The GAViewer operations that are used to generate the projected plane extension CGA 4-blade multivector parameters ($e_1 \wedge e_2 \wedge n_o \wedge n_i, e_1 \wedge e_3 \wedge n_o \wedge n_i, e_2 \wedge e_3 \wedge n_o \wedge n_i, e_1 \wedge e_2 \wedge e_3 \wedge n_i$) for each of the 6 boundary planes were then written to the same (.g) operation file using the first three ordered point IDs associated with each boundary plane and Equation 6 (see Section 3.2.1).

The fourth step was to load the (.geo) object file containing null vector (point) parameters into GAViewer. Once this was opened, the (.g) operation file containing the operations that are used to generate the projected line extension and plane extension CGA multivector parameters was opened and the operations were executed. The results from these operations were then exported and saved as a new (.geo) object file. This new (.geo) object file contains the CGA parameters for the eight null vector (points), the 24 projected line extensions, and the six projected plane extensions associated with the current 3-D cadastral unit boundary components.

The fifth step was to read the new (.geo) object file using a MATLAB function to sort through the CGA objects and to update the six projected line extension CGA multivector parameters and the four projected plane extension CGA multivector parameters into the boundary line and boundary plane EXCEL file tables, respectively. After this was completed, the current cadastral unit boundary is now represented in an EXCEL file containing boundary point, boundary line, and boundary plane tables, as well as represented in a (.geo) object file that contains representations for the null vector CGA point extensions, projected CGA line extensions, and projected CGA plane extensions.

5.2 Generating and Storing CGA Geometric and Intersection Results

To classify the final relationships between boundary component pairs, each component pair first needs to be sorted into one of the 16 projected categories introduced in Figure 3.5. These relationship categories use

the concepts of projected CGA boundary extension components being skew, parallel, collinear, coplanar, or intersecting at a point or line. To determine which category is occurring for each boundary component pair, CGA geometric distance and intersection operators were first generated in GAViewer.

To start the projected relationship determination process, the EXCEL files containing the boundary components of 3-D cadastral units **A** and **B** that were generated using the processes described in Section 5.1 above were added to the same folder. A written MATLAB function loaded in the parameters for all boundary components of both units from these EXCEL files and wrote them to a single (.geo) object file so that they could be loaded into GAViewer simultaneously.

The MATLAB function then wrote a series of GAViewer commands to a single (.g) operation file so that they could be opened in GAViewer to be executed. Sets of operations were written for point-point, line-point, line-line, plane-point, plane-line, and plane-plane boundary component pairs between 3-D cadastral units **A** and **B**. A summary of the operations written for each component pair can be seen in Table 5.1 below.

Table 5.1: Initial CGA Operation Checks Implemented between Boundary Component Pairs

Initial CGA Operation Checks between Boundary Component Pairs		
Point-Point (A-B)	Line-Point (AA-B / BB-A)	Plane-Point (AAA-B / BBB-A)
Distance(A, B)	Distance(A, BB)	Meet(A, BBB)
Sightline(A, B) --> (A_B_SL)	Distance(B, AA)	Meet(B, AAA)
Meet(A_B_SL, BB)		Planeline_Distance(A, BBB_L)
Meet(A_B_SL, AA)	Line-Line (AA-BB)	Planeline_Distance(B, AAA_L)
Meet(A_B_SL, BBB)	Meet(AA, BB)	
Meet(A_B_SL, AAA)		Plane-Line (AAA-BB / BBB-AA)
Sightline(B, A) --> (B_A_SL)		Meet(AA, BBB)
Meet(B_A_SL, A Lines)		Meet(BB, AAA)
Meet(B_A_SL, B Lines)		
Meet(B_A_SL, A Planes)		Plane-Plane (AAA-BBB)
Meet(B_A_SL, B Planes)		Meet(AAA, BBB)

For point-point boundary component pair sets, operations for calculating the geometric distances between each boundary point defining the boundary of **A** and each boundary point defining the boundary of **B** were first calculated in MATLAB. Secondly, operations for generating projected sightlines from each boundary point of **A** to each boundary point of **B**, as well as the reciprocal projected sightlines from each boundary

point of **B** to each boundary point of **A** were written using boundary point IDs and Equation 5 (see Section 3.2.1). Lastly, operations for generating the meet intersections between the projected sightlines of **A** to **B** and the projected sightlines of **B** to **A** with the boundary lines and boundary planes of **A** and **B** were written using sightline IDs, boundary line IDs, boundary plane IDs, and Equation 9 (see Section 3.2.1).

For line-point boundary component pair sets, operations for calculating the geometric distances between each boundary point defining the boundary of **A** and each boundary line defining the boundary of **B**, as well as the distances between each boundary point defining the boundary of **B** and each boundary line defining the boundary of **A** were written using boundary point IDs, boundary line IDs, and Equation 7 (see Section 3.2.1).

For line-line boundary component pair sets, operations for calculating the meet intersections between each boundary line defining the boundary of **A** and each boundary line defining the boundary of **B** were written using boundary line IDs and Equation 9 (see Section 3.2.1).

For plane-point boundary component pair sets, operations for calculating the meet intersections between each boundary point defining the boundary of **A** and each boundary plane defining the boundary of **B**, as well as the meet intersections between each boundary point defining the boundary of **B** and each boundary plane defining the boundary of **A** were written first using boundary point IDs, boundary plane IDs, and Equation 9 (see Section 3.2.1). Secondly, operations for calculating the oriented perpendicular distance between each boundary point defining the boundary of **A** and each boundary line defining the boundary of **B** with respect to each boundary plane defining **B**, as well as operations for calculating the oriented perpendicular distance between each boundary point defining the boundary of **B** and each boundary line defining the boundary of **A** with respect to each boundary plane defining **A** were written using boundary point IDs, boundary line IDs, boundary plane IDs, and Equation 11 (see Section 4.6.1).

For plane-line boundary component pair sets, operations for calculating the meet intersections between each boundary line defining the boundary of **A** and each boundary plane defining the boundary of **B**, as well as the meet intersections between each boundary line defining the boundary of **B** and each boundary plane defining the boundary of **A** were written using boundary line IDs, boundary plane IDs, and Equation 9 (see Section 3.2.1).

For plane-plane boundary component pair sets, operations for calculating the meet intersections between each boundary plane defining the boundary of **A** and each boundary plane defining the boundary of **B** were written using boundary plane IDs and Equation 9 (see Section 3.2.1).

The (.geo) object file containing the boundary components of cadastral units **A** and **B** was loaded into GAViewer. Once this was opened, the (.g) operation file containing the geometric and topological intersection operations between boundary component pairs was opened and the operations were executed. The results from these operations were then exported and saved to a new (.geo) object file. This new (.geo) object file contains the boundary points, boundary lines, and boundary planes for both 3-D cadastral units **A** and **B**, as well as the results for the geometric and topological intersection checks.

After this was completed, the EXCEL files storing the boundaries of cadastral units **A** and **B**, as well as the new (.geo) object file containing geometric and topological intersection results could be analyzed together to classify relationships between their boundary points, boundary lines, and boundary planes.

5.3 Interpreting CGA Analysis Results and Classifying Final Relationships

A series of MATLAB functions were written to read in the previously generated EXCEL and GAViewer files, to interpret which of the projected relationship categories was occurring for each boundary component pair, and to finalize the relationship classifications of each boundary component pair that exists between 3-D cadastral units **A** and **B**.

This process is explained in two parts. Section 5.3.1 describes how the initial CGA intersection checks generated in Section 5.2 were interpreted to determine the initial projected relationship categories for each boundary component pair. Section 5.3.2 describes how the final relationship classification processes were implemented using MATLAB functions and additional CGA intersection checks.

5.3.1 Determining Projected Relationships from CGA Analysis Results

A written MATLAB function loads in the (.geo) object file that was generated in the previous Section 5.2 and sorts CGA objects into the boundary points, boundary lines, and boundary planes for 3-D cadastral units **A** and **B**, as well as the geometric and topological intersection results that were generated. Six types of CGA distance and intersection evaluations were performed to categorize each boundary component pair into the correct projected relationship category. Evaluation conditions for all possible projected relationships were introduced in Table 4.5 and a summary of the possible outcomes for each component pair type is shown in Figure 5.3 below.

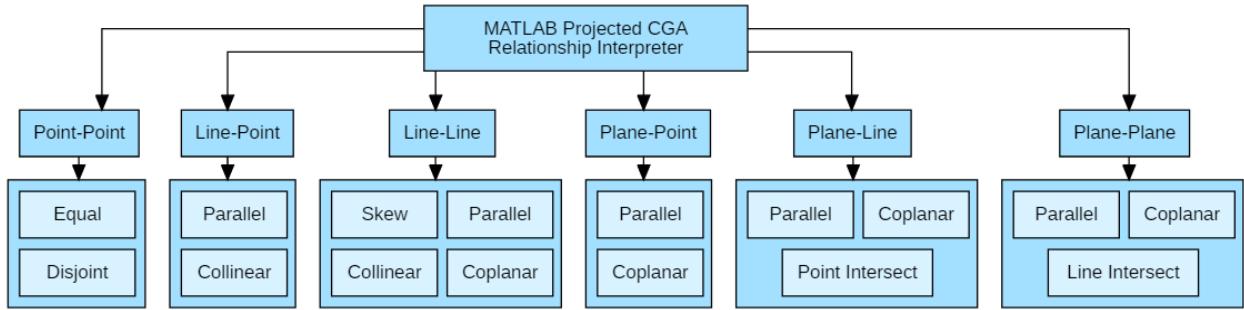


Figure 5.3: Summary of Projected Relationships between Boundary Component Pair Types

Point-point distances were evaluated to determine the final relationship classification between point-point boundary component pairs. Any two points are equal if the distance between them is zero. Any two points are disjoint if there was a non-zero distance between them. This represents the process defined in Figure 4.3.

Line-point distances for each line-point boundary component pair were evaluated to determine if the projected relationship between them is collinear or not. The current boundary point is collinear with the current boundary line if the line-point distance between them was equal to zero. The current boundary point is not collinear with the current boundary line if the line-point distance between them was not equal to zero. This represents the first step in the process defined in Figure 4.5.

The meet intersection result for each line-line boundary component pair was evaluated to determine if the projected relationship between them is not parallel (skew), parallel, collinear, or coplanar. The current boundary lines are not parallel and are categorized into the skew projected relationship category if the line-line meet intersection between them produces a free scalar (distance without a location) containing the n_i CGA multivector parameter only. The current boundary lines are categorized into the coplanar projected relationship category if the line-line meet intersection between them produces a flat point containing the $n_o \wedge e_1, n_o \wedge e_2, n_o \wedge e_3$, and $n_o \wedge n_i$ CGA multivector parameters only. The current boundary lines are categorized into the collinear projected relationship category if the line-line meet intersection between them produces a line containing the $e_1 \wedge n_o \wedge n_i, e_2 \wedge n_o \wedge n_i, e_3 \wedge n_o \wedge n_i, e_1 \wedge e_2 \wedge n_i, e_1 \wedge e_3 \wedge n_i$, and $e_2 \wedge e_3 \wedge n_i$ CGA multivector parameters only. The current boundary lines are categorized into the parallel projected relationship category if the line-line meet intersection between them produces a free vector (line direction without a location) containing the $e_1 \wedge n_i, e_2 \wedge n_i$, and $e_3 \wedge n_i$ CGA multivector parameters only. This represents the first step in the process defined in Figure 4.9.

The meet intersection result for each plane-point boundary component pair was evaluated to determine if the projected relationship between them is coplanar or not. The current boundary point is not coplanar with the current boundary plane if the plane-point meet intersection between them produces a scalar value. The current boundary point is coplanar with the current boundary plane if the plane-point meet intersection between them produces a null vector point containing the n_o , e_1 , e_2 , e_3 , and n_i CGA multivector parameters only. This represents the first step in the process defined in Figure 4.17.

The meet intersection result for each plane-line boundary component pair was evaluated to determine if the projected relationship between them is parallel, coplanar, or if they intersect at a point. The current plane-line boundary component pair is categorized into the coplanar projected relationship category if the plane-line meet intersection between them produces a line containing the $e_1 \wedge n_o \wedge n_i$, $e_2 \wedge n_o \wedge n_i$, $e_3 \wedge n_o \wedge n_i$, $e_1 \wedge e_2 \wedge n_i$, $e_1 \wedge e_3 \wedge n_i$, and $e_2 \wedge e_3 \wedge n_i$ CGA multivector parameters only. The current plane-line boundary component pair is categorized into the intersection point projected relationship category if the plane-line meet intersection between them produces a flat point containing the $n_o \wedge e_1$, $n_o \wedge e_2$, $n_o \wedge e_3$, and $n_o \wedge n_i$ CGA multivector parameters only. The current boundary plane-line component pair is categorized into the parallel projected relationship category if the plane-line meet intersection between them produces a free vector containing the $e_1 \wedge n_i$, $e_2 \wedge n_i$, and $e_3 \wedge n_i$ CGA multivector parameters only. This represents the first step in the process defined in Figure 4.22.

The meet intersection result for each plane-plane boundary component pair was evaluated to determine if the projected relationship between them is parallel, coplanar, or if they intersect at a line. The current plane-plane boundary component pair is categorized into the coplanar projected relationship category if the plane-plane meet intersection between them produces a plane containing the $e_1 \wedge e_2 \wedge n_o \wedge n_i$, $e_1 \wedge e_3 \wedge n_o \wedge n_i$, $e_2 \wedge e_3 \wedge n_o \wedge n_i$, and $e_1 \wedge e_2 \wedge e_3 \wedge n_i$ CGA multivector parameters only. The current plane-plane boundary component pair is categorized into the line intersection category if the plane-plane meet intersection produces a line containing the $e_1 \wedge n_o \wedge n_i$, $e_2 \wedge n_o \wedge n_i$, $e_3 \wedge n_o \wedge n_i$, $e_1 \wedge e_2 \wedge n_i$, $e_1 \wedge e_3 \wedge n_i$, and $e_2 \wedge e_3 \wedge n_i$ CGA multivector parameters only. The current plane-plane boundary component pair is categorized into the parallel projected relationship category if the plane-plane meet produces a free bivector (plane direction without a location) containing the $e_1 \wedge e_2 \wedge n_i$, $e_1 \wedge e_3 \wedge n_i$, and $e_2 \wedge e_3 \wedge n_i$ CGA multivector parameters only. This represents the first step in the process defined in Figure 4.31.

The meet intersection result for each sightline-line pair was evaluated to determine if the projected relationship between them is not parallel (skew), parallel, collinear, or coplanar. The same evaluation process was used here as it is described between line-line component pairs above. The meet intersection result for each sightline-plane pair was evaluated to determine if the projected relationship between them

is parallel, coplanar, or if they intersect at a point. The same evaluation process was used here as it is described between plane-line component pairs above.

5.3.2 Classifying Final Boundary Component Pair Relationships

Six main MATLAB functions were written to determine the final relationship classifications between point-point, line-point, line-line, plane-point, plane-line, and plane-plane boundary component pair sets after they had been categorized into their correct projected relationship categories. An overview of how these six functions are called from the main MATLAB classification function is shown in Figure 5.4 below.

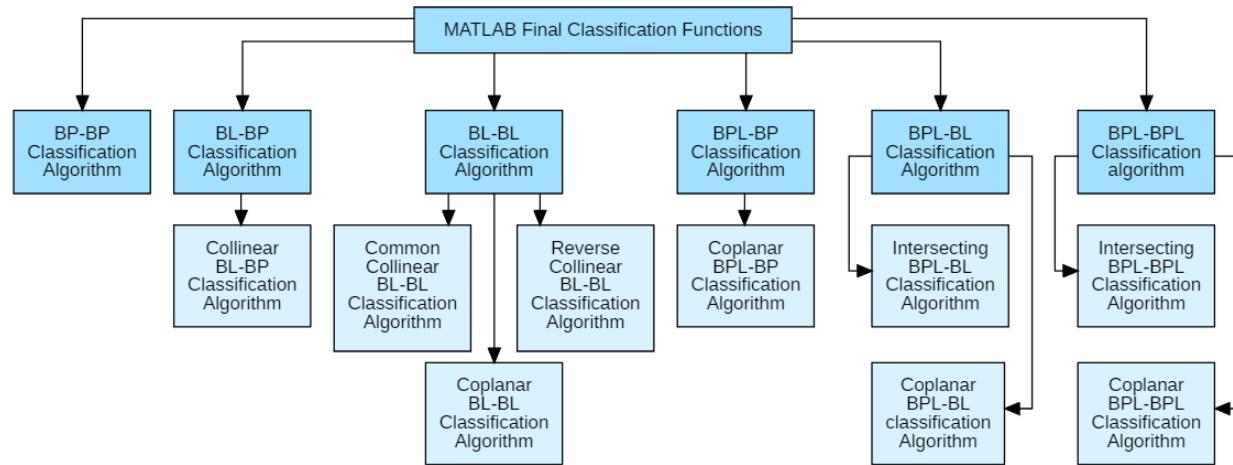


Figure 5.4: Organization of Final MATLAB Classification functions

The point-point determination MATLAB function ‘BP-BP Classification Algorithm’ (see Figure 4.3) loaded in the equal or disjoint relationship categorization result from the previous function for each point-point boundary component pair. If the relationship was equal, the final relationship was classified as ‘0D-0D PT’ and the 3-D boundary point coordinate was stored. If the relationship is disjoint, the final relationship was classified as ‘Disjoint’ and a distance between the boundary points was stored.

The line-point determination MATLAB function ‘BL-BP Classification Algorithm’ (see Figure 4.5) loaded in the collinear or not collinear projected relationship categorization result from the previous function for each line-point boundary component pair. If the projected relationship was not collinear, the final relationship was classified as ‘Disjoint’ and the parallel distance between the current boundary line and boundary point was stored. If the projected relationship was collinear, the start and end boundary points of

the boundary line, as well as the boundary point being considered were passed into the ‘Collinear BL-BP Classification Algorithm’ (see Figure 4.7). This function calculated distances between the boundary line start point, the boundary line end point, and the boundary point, and determined which of the 5 relationship classification outcomes were occurring that are presented in Figure 4.8.

The line-line determination MATLAB function ‘BL-BL Classification Algorithm’ (see Figure 4.9) loaded in the not parallel (skew), parallel, collinear, or coplanar projected relationship categorization result from the previous function for each line-line boundary component pair. If the projected relationship was not parallel (skew) or parallel, the final relationship was classified as ‘Disjoint’ and no geometry was stored. If the projected relationship was collinear, the signs in front of the boundary line extension parameters were compared to determine if the boundary lines had common or reverse orientation. Collinear boundary lines having common orientation were passed into the ‘Common Collinear BL-BL Classification Algorithm’ (see Figure 4.11) and collinear boundary lines having reverse orientation were passed into the ‘Reverse Collinear BL-BL Classification Algorithm’ (see Figure 4.13). Coplanar boundary lines and their projected intersection point were passed into the ‘Coplanar BL-BL Classification Algorithm’ (see Figure 4.15). The common and reverse collinear line-line determination functions each calculated the six possible 3-D point-point distances between the start and end boundary points of the two boundary lines and evaluated which of the 13 relationship classification outcomes were occurring that are presented in Figure 4.12 and Figure 4.14, respectively. The coplanar line-line determination function determined the relationship between the intersection point and both boundary lines using the ‘BL-BP Classification Algorithm’ (see Figure 4.5) and determined which of the 5 relationship classifications were occurring that are presented in Figure 4.16.

The plane-point determination MATLAB function ‘BPL-BP Classification Algorithm’ (see Figure 4.17) loaded in the parallel or coplanar projected relationship categorization result from the previous function for each plane-point boundary component pair. If the projected relationship was parallel, the final relationship was classified as ‘Disjoint’ and no geometry was stored. If the projected relationship was coplanar, the boundary plane and boundary point geometry, the point-line and sightline-line meets associated with the boundary point and boundary plane, and the perpendicular plane-line distances between the boundary point and all boundary lines associated with the boundary plane were collected and passed into the ‘Coplanar BPL-BP Classification Algorithm’ (see Figure 4.19). This function determined which of the 4 relationship classifications were occurring that are presented in Figure 4.20.

The plane-line determination MATLAB function ‘BPL-BL Classification Algorithm’ (see Figure 4.22) loaded in the parallel, coplanar, or point intersection projected relationship categorization result from the previous function for each plane-line boundary component pair. If the projected relationship was parallel, the final relationship was classified as ‘Disjoint’ and no geometry was stored. If the projected relationship

was coplanar, the boundary plane and boundary line data were passed into the ‘Coplanar BPL-BL Classification Algorithm’ (see Figure 4.26 and Figure 4.29). For any point that crosses the boundary of the plane, a new line segment was created that has a start point, an end point, and a generated middle point. Additional CGA projected intersection parameters used for plane-point determination were written to a (.g) operation file for each of these newly generated points and were executed in GAViewer. This coplanar function determined which of the 6 relationship classifications were occurring that are presented in Figure 4.28.

If the projected relationship was point intersection, the boundary plane and boundary line data was passed into the ‘Intersecting BPL-BL Classification Algorithm’ (see Figure 4.24) and additional CGA projected intersection parameters used for plane-point determination between the single intersection point and the boundary plane were written to a (.g) operation file and were evaluated in GAViewer. The final relationship between the boundary line and boundary plane was then determined by classifying the relationship between the intersection point and the boundary line, as well as the relationship between the intersection point and the boundary plane. This intersecting function determined which of the 7 relationship classifications were occurring that are presented in Figure 4.25.

The plane-plane determination MATLAB function ‘BPL-BPL Classification Algorithm’ (see Figure 4.31) loaded in the parallel, coplanar, or line intersection projected relationship categorization result from the previous function for each plane-plane boundary component pair. If the projected relationship category was parallel, the final relationship was classified as ‘Disjoint’ and no geometry was stored. If the projected relationship was coplanar or line intersection, the relationships between each boundary line defining the boundary of the first boundary plane with each boundary line defining the boundary of the second boundary plane were collected. If the projected relationship was coplanar, the boundary planes and collected Plane-line relationships were passed into the ‘Coplanar BPL-BPL Classification Algorithm’ (see Figure 4.36). This coplanar function determined which of the 6 relationship classifications were occurring that are presented in Figure 4.37.

If the projected relationship was line intersection, the boundary planes and the collected plane-line relationships were passed into the ‘Intersecting BPL-BPL Classification Algorithm’ (see Figure 4.33) where the intersection line was split into multiple line segments and middle points were generated for each line segment. Relationship classifications between each boundary plane and each mid point were determined using ‘BPL-BP Classification Algorithm’ (see Figure 4.17) and were further interpreted to finalize the relationship between both boundary planes. This intersecting function determined which of the 11 relationship classifications were occurring that are presented in Figure 4.34.

5.4 Chapter Summary

The relevance this chapter has to the primary research objective is that it presents the approach followed to implement the relationship classification processes that were developed in this study using written MATLAB functions and GAViewer. This chapter was used to report on research activity #7: “Implement the computational procedures proposed in the theoretical model that use CGA projected object representations and operations, as well as 3-D point-point distance evaluations using written MATLAB functions and other software”. The approach and methods described here were followed to run the experimental tests presented in Chapter 6.

Section 5.1 provided an overview of the processes that were followed and the MATLAB functions that were written to generate parameters to digitally represent and store boundary components for each 3-D cadastral unit modelled in this study. The 3-D geometry for boundary points, point-set topology for boundary lines, line set topology for boundary planes, and additional CGA multivector blade parameters for boundary points, boundary lines, and boundary planes were generated using written MATLAB functions and GAViewer files. These parameters were stored in EXCEL tables before relationship classifications were implemented.

The EXCEL tables were initially prepared by filling in the 3-D point geometry for each boundary point, the boundary point IDs defining each boundary line, and the boundary line IDs defining each boundary plane. A written MATLAB function loaded in these tables, generated n_0 and n_i CGA parameters for each point, wrote the point parameters to a GAViewer object file, and wrote operations to a GAViewer operations file to generate CGA boundary line and boundary plane extension parameters. Both files were opened and executed in GAViewer, exported to a new GAViewer object file, imported into a written MATLAB function, and all parameters were updated back into the EXCEL storage tables so that the 3-D cadastral units were ready for classification analysis.

Section 5.2 presented the methods that were followed to initially process boundary component pairs between two 3-D cadastral units and determine which projected relationships were occurring between each of them. The parameters for the boundary components of both cadastral units were written to a single GAViewer (.geo) object file. All CGA object geometric and topological intersection operations were then written to a GAViewer (.g) operations file. Both files were consecutively opened in GAViewer and the geometric and topological intersection operations associated with all boundary component pairs were executed in batch. These were then exported as a new GAViewer (.geo) object file.

Section 5.3 presented the methods that were followed to evaluate projected relationships and to reach final relationship classification outcomes between boundary component pairs. The structure of the various main and secondary classification algorithms for each of the six boundary component pair types was described here. Geometric and topological CGA operations that were generated using processes in Section 5.2 were imported into a written MATLAB function where they were evaluated to determine which projected relationship was occurring between each boundary component pair. This process was repeated when additional CGA operations were needed, specifically for the plane-line relationship classification functions. All 3-D point-point distance checks and interpretations of GAViewer geometric and topological intersection results were evaluated using the various main and secondary written MATLAB functions to determine the final classifications for each boundary component pair.

Chapter Six: Results and Applying Methods to 3-D Cadastral Boundaries

This chapter presents the results of the experimental work that was completed. It summarizes the relationships that can be classified using the processes defined in Chapter 4. It presents the results from seven datasets that were simulated to show that the algorithms and processes described in Chapter 4 and the methods used to implement them described in Chapter 5 can classify 15 different types of disjoint relationships and 38 types of touch/overlap relationships. Lastly, it presents an example of how the processes developed here were tested in a real 3-D cadastral scenario. This chapter is structured as follows:

Section 6.1 provides an overview of the disjoint and overlap relationships that can be classified between 3-D boundary components that make up the boundary of a 3-D cadastral unit using the methods described in Chapters 4 and 5. It also provides an index for the relationships that are classified for each simulated experimental test that was run through the implementation code.

Section 6.2 presents experimental results from seven testing datasets, each consisting of two simulated 3-D cadastral units A and B. The datasets were generated using the processes described in Sections 4.1 and 5.1 and the relationships between the boundary components of units A and B were known *a priori* to testing each dataset.

Section 6.3 presents experimental results from an example derived from a 3-D centerline boundary condominium plan registered in Alberta, Canada where the boundaries of each unit are defined relative to the centerlines of the walls, floors, and ceilings. An overview of condominium boundaries as they are used here can be found in Section 2.4.

6.1 Summary of Relationship Classifications

There are 38 relationship classifications that describe some form of overlap geometry relationship existing between the six different types of boundary component pairs being considered. These were presented in Sections 4.3-4.8 and are summarized in Figure 6.1 below. These overlap classifications can result in boundary touch or boundary intersect geometries.

BP-BP	BL-BP	BL-BP	BPL-BP	BPL-BP	BPL-BP	BL-BL	BL-BL
1 0D-0D PT	2 0D-0D PT	3 1D-0D PT	4 0D-0D PT	5 1D-0D PT	6 2D-0D PT	7 0D-0D PT	8 1D-1D LT
BL-BL	BL-BL	BL-BL	BL-BL	BPL-BL	BPL-BL	BPL-BL	BPL-BL
9 0D-0D PT	10 0D-1D PT	11 1D-0D PT	12 1D-1D PI	13 0D-0D PT	14 0D-1D PT	15 1D-0D PT	16 1D-1D PT
BPL-BL	BPL-BL	BPL-BL	BPL-BL	BPL-BL	BPL-BL	BPL-BL	BPL-BL
17 2D-0D PT	18 2D-1D PI	19 0D-0D PT	20 0D-1D PT	21 1D-0D PT	22 1D-1D LT	23 2D-1D LT	24 0D-0D PT
BPL-BPL	BPL-BPL	BPL-BPL	BPL-BPL	BPL-BPL	BPL-BPL	BPL-BPL	BPL-BPL
25 1D-0D PT	26 0D-1D PT	27 1D-1D LT	28 2D-2D PLT	29 0D-0D PT	30 1D-0D PT	31 0D-1D PT	32 1D-1D PT
BPL-BPL	BPL-BPL	BPL-BPL	BPL-BPL	BPL-BPL	BPL-BPL		
33 2D-0D PT	34 0D-2D PT	35 1D-1D LT	36 2D-1D LT	37 1D-2D LT	38 2D-2D LI		

Point Touch (PT)
 Point Intersect (PI)
 Line Touch (LT)
 Line Intersect (LI)
 Plane Touch (PLT)

Figure 6.1: Overlap Relationship Classifications between Component Pairs

The single overlap relationship that can be classified between two 3-D boundary points (BP-BP) that were presented in Section 4.3 is shown in Figure 6.1 scenario 1. There is no projected relationship in overlap classification 1.

The two overlap relationships that can be classified between a 3-D boundary line and a 3-D boundary point (BL-BP) that were presented in Section 4.4 are shown in Figure 6.1 scenarios 2 and 3. BL and BP have the collinear projected relationship for overlap classifications 2 and 3.

The three overlap relationships that can be classified between a 3-D boundary plane and a 3-D boundary point (BPL-BP) that were presented in Section 4.6 are shown in Figure 6.1 scenarios 4 – 6. BPL and BP have the coplanar projected relationship for overlap classifications 4, 5, and 6.

The six overlap relationships that can be classified between two 3-D boundary lines (BL-BL) that were presented in Section 4.5 are shown in Figure 6.1 scenarios 7 – 12. Both lines have the collinear projected relationship in overlap classifications 7 and 8, and the coplanar projected relationship in overlap classifications 9, 10, 11, and 12.

The 11 overlap relationships that can be classified between a 3-D boundary plane and a 3-D boundary line (BPL-BL) that were presented in Section 4.7 are shown in Figure 6.1 scenarios 13 – 23. BPL and BL have the point intersection projected relationship in overlap classifications 13 – 18, and the coplanar projected relationship in overlap classifications 19 – 23.

The 15 overlap relationships that can be classified between two 3-D boundary planes (BPL-BPL) that were presented in Section 4.8 are shown in Figure 6.1 scenarios 24 – 38. Both planes have the coplanar projected relationship in overlap classifications 24 – 28, and the line intersection projected relationship in overlap classifications 29 – 38. For these line intersection scenarios, the projected line at which the overlap classification occurs is represented as a vertical dotted line.

Table 6.1: Index for Testing Datasets and Overlap Relationship Classifications

Overlap Relationship Classifications and Experimental Data Sets													
Overlap Classification	Test 1	Test 2	Test 3	Test 4	Test 5	Test 6	Overlap Classification	Test 1	Test 2	Test 3	Test 4	Test 5	Test 6
1	✓						20				✓		
2	✓						21			✓	✓		
3			✓	✓			22		✓				
4	✓						23			✓	✓	✓	
5			✓	✓			24	✓					
6		✓			✓	✓	25			✓			
7	✓						26			✓			
8			✓				27		✓				
9	✓						28				✓	✓	
10			✓	✓			29	✓					
11			✓	✓			30			✓	✓		
12					✓	✓	31		✓	✓			
13	✓						32					✓	
14			✓	✓			33		✓				
15			✓	✓			34		✓				
16					✓	✓	35			✓			
17		✓			✓	✓	36			✓	✓	✓	
18						✓	37			✓	✓	✓	
19	✓						38						✓

Table 6.1 provides an overview of which overlap relationships are classified in each of the simulated testing datasets presented in Section 6.2. The first six testing datasets are primarily used to highlight non-disjoint relationships that can be classified between 3-D boundary components. The overlap classification column in Table 6.1 refers to those relationships that were presented in Sections 4.3-4.8 and summarized in Figure 6.1 above.

There are 15 relationship classifications (different from Figure 6.1) that describe some form of disjoint relationship existing between the six different types of boundary component pairs being considered. These were presented in Sections 4.3-4.8 and are summarized in Figure 6.2 below.

BP-BP	BL-BP	BL-BP	BL-BL	BL-BL
1 • • (Disjoint)	2 Not Collinear (Disjoint)	3 Collinear (Disjoint)	4 Not Parallel (Disjoint)	5 Parallel (Disjoint)
BL-BL	BL-BL	BPL-BP	BPL-BP	BPL-BL
6 Collinear (Disjoint)	7 Coplanar (Disjoint)	8 Parallel (Disjoint)	9 Coplanar (Disjoint)	10 Parallel (Disjoint)
BPL-BL	BPL-BL	BPL-BPL	BPL-BPL	BPL-BPL
11 Point Intersect (Disjoint)	12 Coplanar (Disjoint)	13 Parallel (Disjoint)	14 Coplanar (Disjoint)	15 Line Intersect (Disjoint)

Figure 6.2: Disjoint Relationship Classifications between Component Pairs

The single disjoint relationship that can be classified between two 3-D boundary points (BP-BP) that were presented in Section 4.3 is shown in Figure 6.2 scenario 1. There is no projected relationship in disjoint classification 1.

The two disjoint relationships that can be classified between a 3-D boundary line and a 3-D boundary point (BL-BP) that were presented in Section 4.4 are shown in Figure 6.2 scenarios 2 and 3. BL and BP have the coplanar (not collinear) projected relationship in disjoint classification 2 and the collinear projected relationship in disjoint classification 3.

The four disjoint relationships that can be classified between two 3-D boundary lines (BL-BL) that were presented in Section 4.5 are shown in Figure 6.2 scenarios 4 – 7. Both lines have the skew (not parallel) projected relationship in disjoint classification 4, the parallel projected relationship in disjoint classification 5, the collinear projected relationship in disjoint classification 6, and the coplanar projected relationship in disjoint classification 7.

The two disjoint relationships that can be classified between a 3-D boundary plane and a 3-D boundary point (BPL-BP) that were presented in Section 4.6 are shown in Figure 6.2 scenarios 8 – 9. BPL and BP have the parallel projected relationship in disjoint classification 8 and the coplanar projected relationship in disjoint classification 9.

The three disjoint relationships that can be classified between a 3-D boundary plane and a 3-D boundary line (BPL-BL) that were presented in Section 4.7 is shown in Figure 6.2 scenarios 10 – 12. BPL and BL have the parallel projected relationship in disjoint classification 10, the point intersect projected relationship in disjoint classification 11, and the coplanar projected relationship in disjoint classification 12.

The three disjoint relationships that can be classified between two 3-D boundary planes (BPL-BPL) that were presented in Section 4.8 are shown in Figure 6.2 scenarios 13 – 15. Both planes have the parallel projected relationship in disjoint classification 13, the coplanar projected relationship in disjoint classification 14, and the line intersect projected relationship in disjoint classification 15.

Table 6.2: Index for Testing Datasets and Disjoint Relationship Classifications

Disjoint Relationship Classifications and Experimental Data Sets							
Disjoint Classification	Test 1	Test 2	Test 3	Test 4	Test 5	Test 6	Test 7
1	✓	✓	✓	✓	✓	✓	✓
2	✓	✓	✓	✓	✓	✓	✓
3	✓		✓				✓
4	✓	✓	✓	✓	✓	✓	✓
5	✓		✓	✓	✓	✓	✓
6							✓
7	✓		✓	✓	✓		✓
8	✓	✓	✓	✓	✓	✓	✓
9	✓		✓	✓	✓		✓
10	✓		✓	✓	✓	✓	✓
11	✓	✓	✓	✓	✓	✓	✓
12	✓		✓	✓	✓		✓
13	✓		✓	✓	✓	✓	✓
14							✓
15	✓	✓	✓	✓	✓	✓	✓

Table 6.2 provides an overview of which disjoint relationships can be found in each of the simulated testing datasets presented in Section 6.2. While the first six testing datasets have some 3-D boundary component pairs that represent most of the disjoint relationships, testing dataset seven is used to show all the 15 distinct types of disjoint relationships that can be classified between boundary component pair sets. The disjoint classification column in Table 6.2 refers to those relationships that were presented in Sections 4.3-4.8 and are summarized in Figure 6.2 above.

6.2 Simulated Experimental Testing Datasets and Classification Results

Seven experimental datasets, each consisting of two 3-D cadastral units A and B were simulated to initially test and validate the relationship classification processes developed in Chapter 4. The relationships between 3-D boundary components were known *a priori* to running each dataset through the implementation code described throughout Chapter 5. An overview of these simulated datasets, along with a list of the *a priori* relationships that exist between the two 3-D units can be seen in Figure 6.3 below.

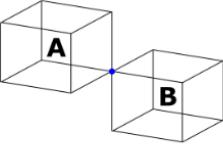
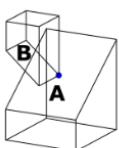
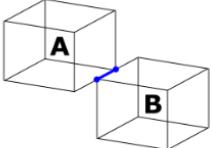
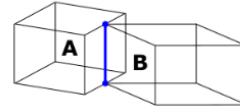
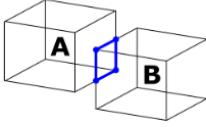
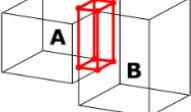
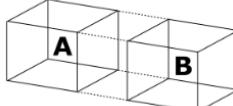
Test 1	Test 2	Test 3	Test 4
			
Overlap Relationships: 1,2,4,7,9,13, 19,24,29	Overlap Relationships: 6,17,33,34	Overlap Relationships: 3,5,8,10,11,14,15, 21,22,27,30,31,35	Overlap Relationships: 3,5,10,11,14,15,20,21, 23,25,26,30,31,36,37
Test 5	Test 6	Test 7	
			
Overlap Relationships: 6,12,16,17,23, 28,32,36,37	Overlap Relationships: 6,12,16,17,18,23, 28,36,37,38	Disjoint Relationships: 1,2,3,4,5,6,7,8,9,10, 11,12,13,14,15	

Figure 6.3: Overview of Simulated Testing Datasets and Relationship Classifications

Simulated experimental tests 1 to 6 were used to highlight the different overlap relationship classifications that are summarized in Figure 6.1 and Table 6.1. Simulated experimental test 7 was used to highlight the different disjoint relationship classifications that are summarized in Figure 6.2 and Table 6.2. Experiment 7 contains all possible disjoint relationship classifications in a single example, as it can be seen in Table 6.2 that tests 1 – 6 only classify a subset of these relationships. The datasets were designed to highlight the various relationship that can be classified using the processes described in Sections 4.3-4.8 and each 3-D cadastral unit boundary was simulated using the processes described in Section 5.1.

Each experimental testing dataset is first introduced using figures and tables representing the boundary point coordinates (geometry), line topology, and plane topology point sets for each of the two 3-D cadastral

units A and B. Boundary points are represented using the (A#, B#) writing convention, boundary lines are represented using the (AA#, BB#) writing convention, and boundary planes are represented using the (AAA#, BBB#) writing convention. Secondly, the *a priori* relationships that exist between 3-D cadastral units A and B are then described in a figure (e.g. see Figure 6.5). Thirdly, the relationship classification output results from the implementation code are presented using a final table (e.g. see Table 6.6).

6.2.1 Simulated Experimental Test 1

Simulated experimental test 1 consists of 3-D cadastral units A and B that are both (1m*1m*1m) volumetric cubes in 3-D space. This experiment is used to highlight overlap relationship classifications 1, 2, 4, 7, 9, 13, 19, 24, and 29 (see Figure 6.5). Visualizations of the boundary points, boundary lines, and boundary planes associated with each 3-D cadastral unit in this experiment can be seen in Figure 6.4 below.

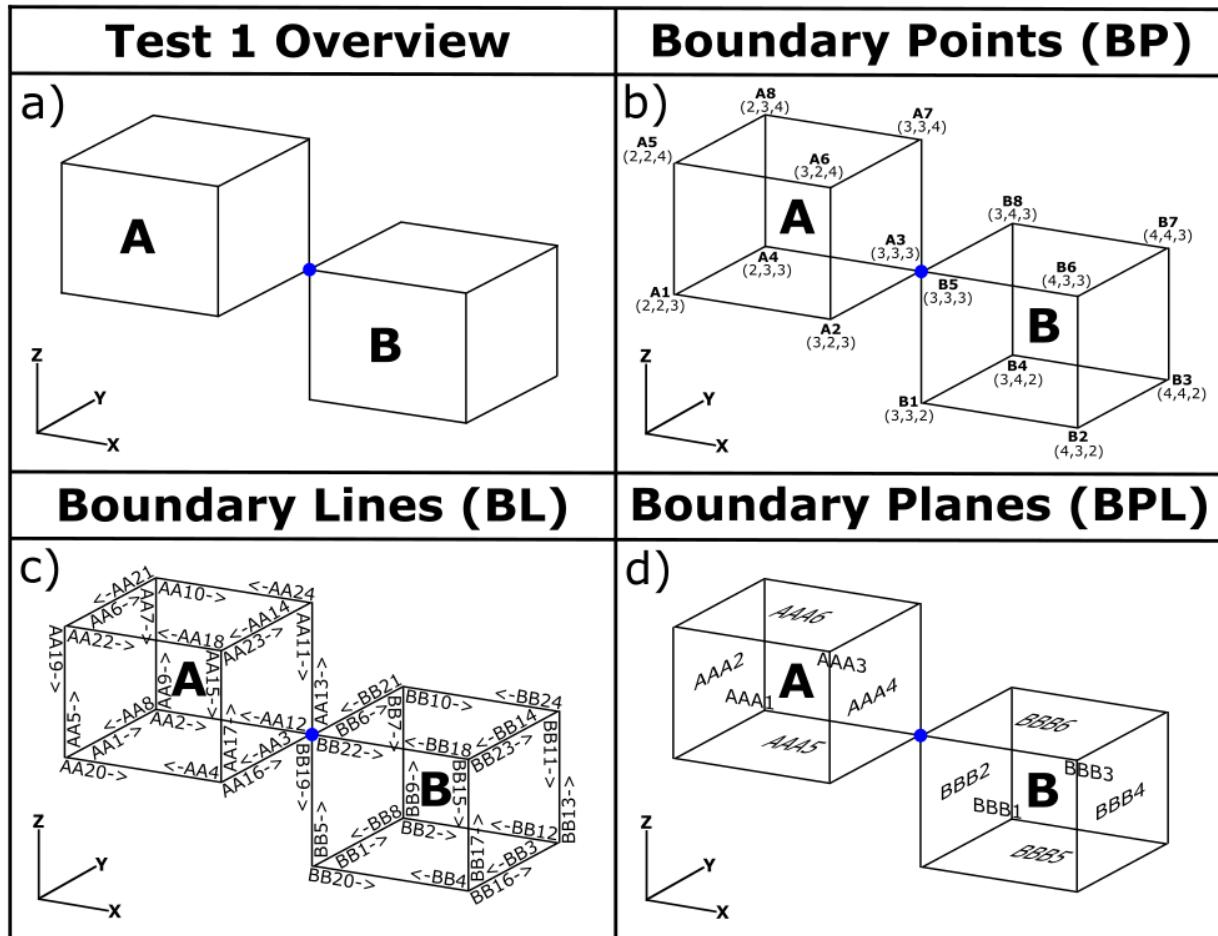


Figure 6.4: Testing Dataset 1 Visualization

Figure 6.4a shows an overview of simulated 3-D cadastral units A and B used in test 1. The main overlap between the two 3-D cadastral units in this experiment can be described by the boundary point A3 touching the boundary point B5. Figure 6.4b shows the names and 3-D coordinates of all boundary points, Figure 6.4c shows the names and topological directions of all boundary lines, and Figure 6.4d shows the names of all boundary planes. Unit A's floor and unit B's ceiling are coplanar, and they have two sets of coplanar walls between them (AAA3-BBB1 & AAA4-BBB2). These figures show that all output overlap classifications will be related to the relationship between boundary points A3 and B5 which should produce the 3-D point return coordinate [3,3,3].

Tables 6.3 – 6.5 below show the object input storage parameters for the 3-D cadastral units A and B used to create experimental testing dataset 1. Table 6.3 shows the boundary point storage parameters, Table 6.4 shows the boundary line storage parameters, and Table 6.5 shows the boundary plane storage parameters used to define the boundaries of 3-D cadastral units A and B. These parameters were generated using the processes described in Section 5.1.

Table 6.3: Testing Dataset 1 - 3-D Boundary Point Coordinates

		A Boundary Points					B Boundary Points						
Point ID	Point Name	X (e1)	Y (e2)	Z (e3)	no	ni	Point ID	Point Name	X (e1)	Y (e2)	Z (e3)	no	ni
1	A1	2	2	3	1	8.5	1	B1	3	3	2	1	11
2	A2	3	2	3	1	11	2	B2	4	3	2	1	14.5
3	A3	3	3	3	1	13.5	3	B3	4	4	2	1	18
4	A4	2	3	3	1	11	4	B4	3	4	2	1	14.5
5	A5	2	2	4	1	12	5	B5	3	3	3	1	13.5
6	A6	3	2	4	1	14.5	6	B6	4	3	3	1	17
7	A7	3	3	4	1	17	7	B7	4	4	3	1	20.5
8	A8	2	3	4	1	14.5	8	B8	3	4	3	1	17

Table 6.4: Testing Dataset 1 - 3-D Boundary Line Storage Parameters

A Boundary Lines									B Boundary Lines										
Line ID	Line Name	Start Point ID	End Point ID	CGA 3-Blade Parameters					Line ID	Line Name	Start Point ID	End Point ID	CGA 3-Blade Parameters						
				e1^no^ni	e2^no^ni	e3^no^ni	e1^e2^ni	e1^e3^ni					e1^no^ni	e2^no^ni	e3^no^ni	e1^e2^ni	e1^e3^ni	e2^e3^ni	
1	AA1	1	4	0	-1	0	2	0	-3	1	BB1	1	4	0	-1	0	3	0	-2
2	AA2	4	3	-1	0	0	-3	-3	0	2	BB2	4	3	-1	0	0	-4	-2	0
3	AA3	3	2	0	1	0	-3	0	3	3	BB3	3	2	0	1	0	-4	0	2
4	AA4	2	1	1	0	0	2	3	0	4	BB4	2	1	1	0	0	3	2	0
5	AA5	1	5	0	0	-1	0	2	2	5	BB5	1	5	0	0	-1	0	3	3
6	AA6	5	8	0	-1	0	2	0	-4	6	BB6	5	8	0	-1	0	3	0	-3
7	AA6	8	4	0	0	1	0	-2	-3	7	BB7	8	4	0	0	1	0	-3	-4
8	AA7	4	1	0	1	0	-2	0	3	8	BB8	4	1	0	1	0	-3	0	2
9	AA9	4	8	0	0	-1	0	2	3	9	BB9	4	8	0	0	-1	0	3	4
10	AA10	8	7	-1	0	0	-3	-4	0	10	BB10	8	7	-1	0	0	-4	-3	0
11	AA11	7	3	0	0	1	0	-3	-3	11	BB11	7	3	0	0	1	0	-4	-4
12	AA12	3	4	1	0	0	3	3	0	12	BB12	3	4	1	0	0	4	2	0
13	AA13	3	7	0	0	-1	0	3	3	13	BB13	3	7	0	0	-1	0	4	4
14	AA14	7	6	0	1	0	-3	0	4	14	BB14	7	6	0	1	0	-4	0	3
15	AA15	6	2	0	0	1	0	-3	-2	15	BB15	6	2	0	0	1	0	-4	-3
16	AA16	2	3	0	-1	0	3	0	-3	16	BB16	2	3	0	-1	0	4	0	-2
17	AA17	2	6	0	0	-1	0	3	2	17	BB17	2	6	0	0	-1	0	4	3
18	AA18	6	5	1	0	0	2	4	0	18	BB18	6	5	1	0	0	3	3	0
19	AA19	5	1	0	0	1	0	-2	-2	19	BB19	5	1	0	0	1	0	-3	-3
20	AA20	1	2	-1	0	0	-2	-3	0	20	BB20	1	2	-1	0	0	-3	-2	0
21	AA21	8	5	0	1	0	-2	0	4	21	BB21	8	5	0	1	0	-3	0	3
22	AA22	5	6	-1	0	0	-2	-4	0	22	BB22	5	6	-1	0	0	-3	-3	0
23	AA23	6	7	0	-1	0	3	0	-4	23	BB23	6	7	0	-1	0	4	0	-3
24	AA24	7	8	1	0	0	3	4	0	24	BB24	7	8	1	0	0	4	3	0

Table 6.5: Testing Dataset 1 - 3-D Boundary Plane Storage Parameters

A Boundary Planes					B Boundary Planes								
Plane ID	Plane Name	Polyline Numbers	CGA 4-Blade Parameters				Plane ID	Plane Name	Polyline Numbers	CGA 4-Blade Parameters			
			e1^e2^no^ni	e1^e3^no^ni	e2^e3^no^ni	e1^e2^e3^ni				e1^e2^no^ni	e1^e3^no^ni	e2^e3^no^ni	e1^e2^e3^ni
1	AAA1	17_18_19_20	0	1	0	-2	1	BBB1	17_18_19_20	0	1	0	-3
2	AAA2	5_6_7_8	0	0	-1	-2	2	BBB2	5_6_7_8	0	0	-1	-3
3	AAA3	9_10_11_12	0	-1	0	3	3	BBB3	9_10_11_12	0	-1	0	4
4	AAA4	13_14_15_16	0	0	1	3	4	BBB4	13_14_15_16	0	0	1	4
5	AAA5	1_2_3_4	-1	0	0	-3	5	BBB5	1_2_3_4	-1	0	0	-2
6	AAA6	21_22_23_24	1	0	0	4	6	BBB6	21_22_23_24	1	0	0	3

There are nine different types of overlap relationship classifications between 3-D cadastral units A and B in test 1 that are known to exist *a priori* to testing. These can be seen in Figure 6.5 below.

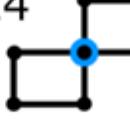
BP-BP	BL-BP	BPL-BP	BL-BL	BL-BL
1  0D-0D PT	2  0D-0D PT	4  0D-0D PT	7  0D-0D PT	9  0D-0D PT
BPL-BL	BPL-BL	BPL-BPL	BPL-BPL	
13  0D-0D PT	19  0D-0D PT	24  0D-0D PT	29  0D-0D PT	

Figure 6.5: Testing Dataset 1 - Existing Overlap Relationship Classifications

Table 6.6 below shows a compiled summary of the program output for experimental testing dataset 1. It is organized into relationships between the six different component pair sets. Each component pair producing an overlap relationship classification was manually verified through visual inspection to be one of the *a priori* relationships shown in Figure 6.5 above. This was done by comparing the main classification algorithm program output from Table 6.6 with the experimental test 1 scenario visualizations shown in Figure 6.4.

Table 6.6: Testing Dataset 1 - Program Output

Test 1 - Component Pair Classification Results														
BP-BP					BL-BL		BPL-BL							
Component Pair	Projected Relationship	Relationship Description	Overlap Classification	Geometry	Component Pair	Projected Relationship	Relationship Description	Overlap Classification	Geometry	Component Pair	Projected Relationship	Relationship Description	Overlap Classification	Geometry
A3-B5	N/A	OD-OD PT	1	[3,3,3]	AA11-BB18	Coplanar	OD-OD PT	9	[3,3,3]	BBB1-AA11	Coplanar	OD-OD PT	19	[3,3,3]
BS-A3	N/A	OD-OD PT	1	[3,3,3]	AA11-BB19	Collinear (C)	OD-OD PT	7	[3,3,3]	BBB2-AA11	Coplanar	OD-OD PT	19	[3,3,3]
BL-BP					AA11-BB21	Coplanar	OD-OD PT	9	[3,3,3]	BBB6-AA11	P Intersect	OD-OD PT	13	[3,3,3]
					AA11-BB22	Coplanar	OD-OD PT	9	[3,3,3]	BBB1-AA12	Coplanar	OD-OD PT	19	[3,3,3]
					AA11-BB5	Collinear (R)	OD-OD PT	7	[3,3,3]	BBB2-AA12	P Intersect	OD-OD PT	13	[3,3,3]
					AA11-BB6	Coplanar	OD-OD PT	9	[3,3,3]	BBB6-AA12	Coplanar	OD-OD PT	19	[3,3,3]
					AA12-BB18	Collinear (C)	OD-OD PT	7	[3,3,3]	BBB1-AA13	Coplanar	OD-OD PT	19	[3,3,3]
					AA12-BB19	Coplanar	OD-OD PT	9	[3,3,3]	BBB2-AA13	Coplanar	OD-OD PT	19	[3,3,3]
					AA12-BB21	Coplanar	OD-OD PT	9	[3,3,3]	BBB6-AA13	P Intersect	OD-OD PT	13	[3,3,3]
					AA12-BB22	Collinear (R)	OD-OD PT	7	[3,3,3]	BBB1-AA16	P Intersect	OD-OD PT	13	[3,3,3]
					AA12-BB5	Coplanar	OD-OD PT	9	[3,3,3]	BBB2-AA16	Coplanar	OD-OD PT	19	[3,3,3]
					AA12-BB6	Coplanar	OD-OD PT	9	[3,3,3]	BBB6-AA16	Coplanar	OD-OD PT	19	[3,3,3]
					AA13-BB18	Coplanar	OD-OD PT	9	[3,3,3]	BBB1-AA2	Coplanar	OD-OD PT	19	[3,3,3]
					AA13-BB19	Collinear (R)	OD-OD PT	7	[3,3,3]	BBB2-AA2	P Intersect	OD-OD PT	13	[3,3,3]
					AA13-BB21	Coplanar	OD-OD PT	9	[3,3,3]	BBB6-AA2	Coplanar	OD-OD PT	19	[3,3,3]
					AA13-BB22	Coplanar	OD-OD PT	9	[3,3,3]	BBB1-AA3	P Intersect	OD-OD PT	13	[3,3,3]
					AA13-BB5	Collinear (C)	OD-OD PT	7	[3,3,3]	BBB2-AA3	Coplanar	OD-OD PT	19	[3,3,3]
					AA13-BB6	Coplanar	OD-OD PT	9	[3,3,3]	BBB6-AA3	Coplanar	OD-OD PT	19	[3,3,3]
					AA16-BB18	Coplanar	OD-OD PT	9	[3,3,3]	AAA1-BB18	Coplanar	OD-OD PT	19	[3,3,3]
					AA16-BB19	Coplanar	OD-OD PT	9	[3,3,3]	AAA4-BB18	P Intersect	OD-OD PT	13	[3,3,3]
					AA16-BB21	Collinear (R)	OD-OD PT	7	[3,3,3]	AAA5-BB18	Coplanar	OD-OD PT	19	[3,3,3]
					AA16-BB22	Coplanar	OD-OD PT	9	[3,3,3]	AAA3-BB19	Coplanar	OD-OD PT	19	[3,3,3]
					AA16-BB5	Coplanar	OD-OD PT	9	[3,3,3]	AAA4-BB19	Coplanar	OD-OD PT	19	[3,3,3]
					AA16-BB6	Collinear (C)	OD-OD PT	7	[3,3,3]	AAA5-BB19	P Intersect	OD-OD PT	13	[3,3,3]
					AA2-BB18	Collinear (R)	OD-OD PT	7	[3,3,3]	AAA3-BB21	P Intersect	OD-OD PT	13	[3,3,3]
					AA2-BB19	Coplanar	OD-OD PT	9	[3,3,3]	AAA4-BB21	Coplanar	OD-OD PT	19	[3,3,3]
					AA2-BB21	Coplanar	OD-OD PT	9	[3,3,3]	AAA5-BB21	Coplanar	OD-OD PT	19	[3,3,3]
					AA2-BB22	Collinear (C)	OD-OD PT	7	[3,3,3]	AAA3-BB22	Coplanar	OD-OD PT	19	[3,3,3]
					AA2-BB5	Coplanar	OD-OD PT	9	[3,3,3]	AAA4-BB22	P Intersect	OD-OD PT	13	[3,3,3]
					AA2-BB6	Coplanar	OD-OD PT	9	[3,3,3]	AAA5-BB22	Coplanar	OD-OD PT	19	[3,3,3]
					AA3-BB18	Coplanar	OD-OD PT	9	[3,3,3]	AAA3-BB22	Coplanar	OD-OD PT	19	[3,3,3]
					AA3-BB19	Coplanar	OD-OD PT	9	[3,3,3]	AAA4-BB19	Coplanar	OD-OD PT	19	[3,3,3]
					AA3-BB21	Collinear (C)	OD-OD PT	7	[3,3,3]	AAA5-BB19	P Intersect	OD-OD PT	13	[3,3,3]
					AA3-BB22	Coplanar	OD-OD PT	9	[3,3,3]	AAA3-BB21	P Intersect	OD-OD PT	13	[3,3,3]
					AA3-BB5	Coplanar	OD-OD PT	9	[3,3,3]	AAA4-BB21	Coplanar	OD-OD PT	19	[3,3,3]
					AA3-BB6	Collinear (R)	OD-OD PT	7	[3,3,3]	AAA5-BB21	P Intersect	OD-OD PT	13	[3,3,3]
										AAA4-BB22	Coplanar	OD-OD PT	19	[3,3,3]
										AAA5-BB22	Coplanar	OD-OD PT	19	[3,3,3]
										AAA3-BB5	Coplanar	OD-OD PT	19	[3,3,3]
										AAA4-BB5	Coplanar	OD-OD PT	19	[3,3,3]
										AAA5-BB5	P Intersect	OD-OD PT	13	[3,3,3]
										AAA3-BB6	P Intersect	OD-OD PT	13	[3,3,3]
										AAA4-BB6	Coplanar	OD-OD PT	19	[3,3,3]
										AAA5-BB6	Coplanar	OD-OD PT	19	[3,3,3]

Table 6.6 shows that there are two BP-BP pairs involving the A3 and B5 points that produce the '0D-0D Point Touch' relationship (overlap classification 1). 12 BL-BP pairs produce the 'Collinear 0D-0D Point Touch' relationship (overlap classification 2). Six BPL-BP pairs produce the 'Coplanar 0D-0D Point Touch' relationship (overlap classification 4).

12 BL-BL pairs produce the 'Collinear (C/R) 0D-0D Point Touch' relationship (overlap classification 7) and 24 BL-BL pairs produce the 'Coplanar 0D-0D Point Touch' relationship (overlap classification 9).

12 BPL-BL pairs produce the 'Point Intersect 0D-0D Point Touch' relationship (overlap classification 13) and 24 BPL-BL pairs produce the 'Coplanar 0D-0D Point Touch' relationship (overlap classification 19).

Three BPL-BPL pairs produce the 'Coplanar 0D-0D Point Touch' relationship (overlap classification 24) and six BPL-BPL pairs produce the 'Line Intersect 0D-0D Point Touch' relationship (overlap classification 29).

6.2.2 Simulated Experimental Test 2

Simulated experimental test 2 consists of 3-D cadastral units A and B that are both volumetric cube-like objects. Each is defined by two trapezoidal and four rectangular boundary planes in 3-D space. This experiment is used to highlight overlap relationship classifications 6, 17, 33, and 34 (see Figure 6.7). Visualizations of the boundary points, boundary lines, and boundary planes associated with each 3-D cadastral unit in this experiment can be seen in Figure 6.6 below.

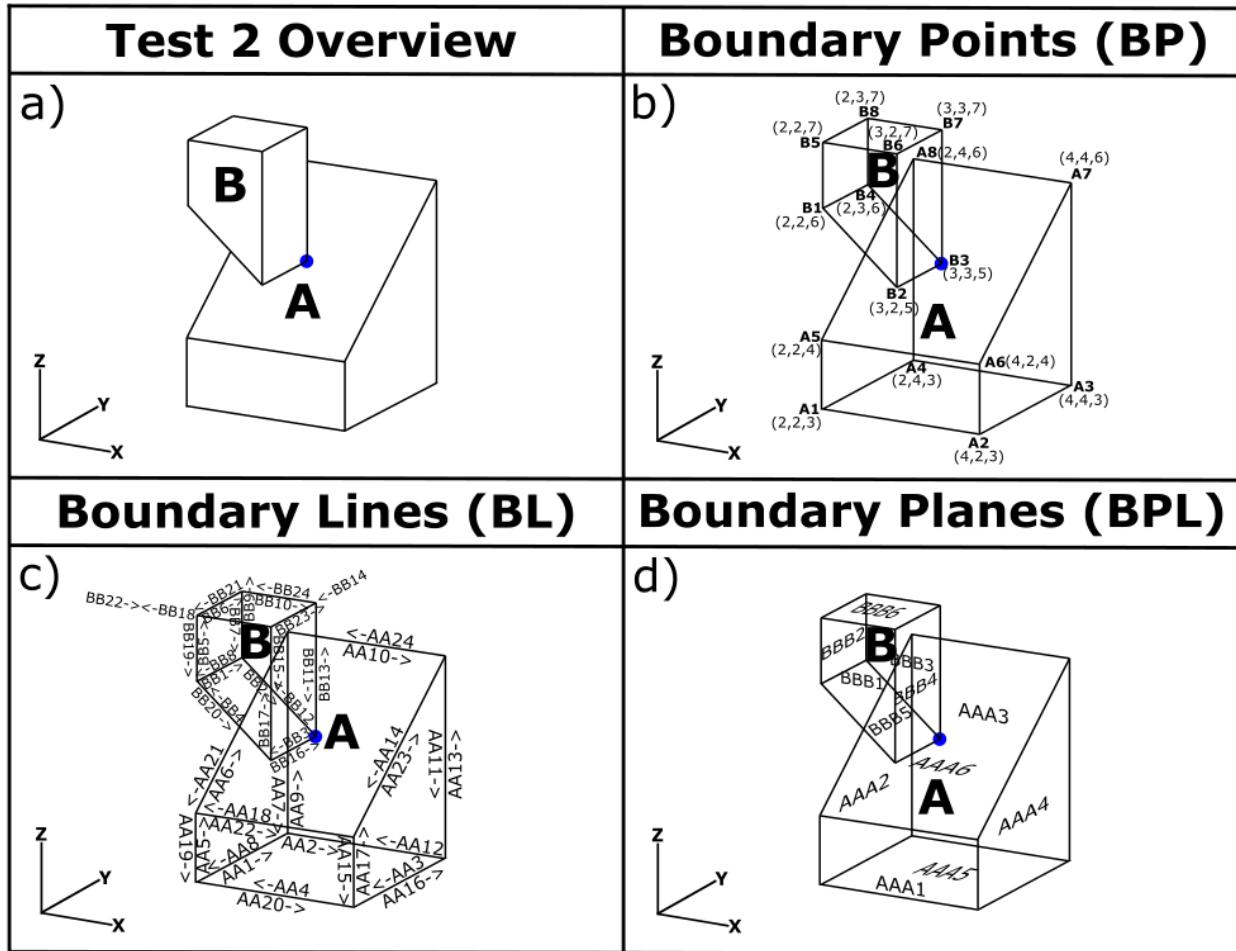


Figure 6.6: Testing Dataset 2 Visualization

Figure 6.6a shows an overview of simulated 3-D cadastral units A and B used in test 2. The main overlap between the two 3-D cadastral units in this experiment can be described by the boundary point B3 touching the boundary plane AAA6. Figure 6.6b shows the names and 3-D coordinates of all boundary points, Figure 6.6c shows the names and topological directions of all boundary lines, and Figure 6.6d shows the names of

all boundary planes. These figures show that all output overlap relationships will be related to the relationship between boundary point B3 and boundary plane AAA6 which should produce the 3-D point return coordinate [3,3,5].

Tables 6.7 – 6.9 below show the object input storage parameters for the 3-D cadastral units A and B used to create experimental testing dataset 2. Table 6.7 shows the boundary point storage parameters, Table 6.8 shows the boundary line storage parameters, and Table 6.9 shows the boundary plane storage parameters used to define the boundaries of 3-D cadastral units A and B. These parameters were generated using the processes described in Section 5.1.

Table 6.7: Testing Dataset 2 - 3-D Boundary Point Coordinates

A Boundary Points						B Boundary Points							
Point ID	Point Name	X (e1)	Y (e2)	Z (e3)	no	Point ID	Point Name	X (e1)	Y (e2)	Z (e3)	no	ni	
1	A1	2	2	3	1	8.5	1	B1	2	2	6	1	22
2	A2	4	2	3	1	14.5	2	B2	3	2	5	1	19
3	A3	4	4	3	1	20.5	3	B3	3	3	5	1	21.5
4	A4	2	4	3	1	14.5	4	B4	2	3	6	1	24.5
5	A5	2	2	4	1	12	5	B5	2	2	7	1	28.5
6	A6	4	2	4	1	18	6	B6	3	2	7	1	31
7	A7	4	4	6	1	34	7	B7	3	3	7	1	33.5
8	A8	2	4	6	1	28	8	B8	2	3	7	1	31

Table 6.8: Testing Dataset 2 - 3-D Boundary Line Storage Parameters

A Boundary Lines									B Boundary Lines										
Line ID	Line Name	Start Point ID	End Point ID	CGA 3-Blade Parameters					Line ID	Line Name	Start Point ID	End Point ID	CGA 3-Blade Parameters						
				e1^no^ni	e2^no^ni	e3^no^ni	e1^e2^ni	e1^e3^ni					e1^no^ni	e2^no^ni	e3^no^ni	e1^e2^ni	e1^e3^ni	e2^e3^ni	
1	AA1	1	4	0	-2	0	4	0	-6	1	BB1	1	4	0	-1	0	2	0	-6
2	AA2	4	3	-2	0	0	-8	-6	0	2	BB2	4	3	-1	0	1	-3	-8	-3
3	AA3	3	2	0	2	0	-8	0	6	3	BB3	3	2	0	1	0	-3	0	5
4	AA4	2	1	2	0	0	4	6	0	4	BB4	2	1	1	0	-1	2	8	2
5	AA5	1	5	0	0	-1	0	2	2	5	BB5	1	5	0	0	-1	0	2	2
6	AA6	5	8	0	-2	-2	4	4	-4	6	BB6	5	8	0	-1	0	2	0	-7
7	AA6	8	4	0	0	3	0	-6	-12	7	BB7	8	4	0	0	1	0	-2	-3
8	AA7	4	1	0	2	0	-4	0	6	8	BB8	4	1	0	1	0	-2	0	6
9	AA9	4	8	0	0	-3	0	6	12	9	BB9	4	8	0	0	-1	0	2	3
10	AA10	8	7	-2	0	0	-8	-12	0	10	BB10	8	7	-1	0	0	-3	-7	0
11	AA11	7	3	0	0	3	0	-12	-12	11	BB11	7	3	0	0	2	0	-6	-6
12	AA12	3	4	2	0	0	8	6	0	12	BB12	3	4	1	0	-1	3	8	3
13	AA13	3	7	0	0	-3	0	12	12	13	BB13	3	7	0	0	-2	0	6	6
14	AA14	7	6	0	2	2	-8	-8	4	14	BB14	7	6	0	1	0	-3	0	7
15	AA15	6	2	0	0	1	0	-4	-2	15	BB15	6	2	0	0	2	0	-6	-4
16	AA16	2	3	0	-2	0	8	0	-6	16	BB16	2	3	0	-1	0	3	0	-5
17	AA17	2	6	0	0	-1	0	4	2	17	BB17	2	6	0	0	-2	0	6	4
18	AA18	6	5	2	0	0	4	8	0	18	BB18	6	5	1	0	0	2	7	0
19	AA19	5	1	0	0	1	0	-2	-2	19	BB19	5	1	0	0	1	0	-2	-2
20	AA20	1	2	-2	0	0	-4	-6	0	20	BB20	1	2	-1	0	1	-2	-8	-2
21	AA21	8	5	0	2	2	-4	-4	4	21	BB21	8	5	0	1	0	-2	0	7
22	AA22	5	6	-2	0	0	-4	-8	0	22	BB22	5	6	-1	0	0	-2	-7	0
23	AA23	6	7	0	-2	-2	8	8	-4	23	BB23	6	7	0	-1	0	3	0	-7
24	AA24	7	8	2	0	0	8	12	0	24	BB24	7	8	1	0	0	3	7	0

Table 6.9: Testing Dataset 2 - 3-D Boundary Plane Storage Parameters

A Boundary Planes									B Boundary Planes											
Plane ID	Plane Name	Polyline ID Numbers	CGA 4-Blade Parameters				Plane ID	Plane Name	Polyline ID Numbers	CGA 4-Blade Parameters				Plane ID	Plane Name	Polyline ID Numbers	CGA 4-Blade Parameters			
			e1^e2^no^ni	e1^e3^no^ni	e2^e3^no^ni	e1^e2^e3^ni				e1^e2^no^ni	e1^e3^no^ni	e2^e3^no^ni	e1^e2^e3^ni							
1	AAA1	17_18_19_20	0	2	0	-4	1	BBB1	17_18_19_20	0	2	0	-4	2	BBB2	5_6_7_8	0	0	-1	-2
2	AAA2	5_6_7_8	0	0	-2	-4	3	BBB3	9_10_11_12	0	-1	0	3	4	BBB4	13_14_15_16	0	0	2	6
3	AAA3	9_10_11_12	0	-6	0	24	5	BBB5	1_2_3_4	-1	0	-1	-8	6	BBB6	21_22_23_24	1	0	0	7
4	AAA4	13_14_15_16	0	0	6	24														
5	AAA5	1_2_3_4	-4	0	0	-12														
6	AAA6	21_22_23_24	4	4	0	8														

There are four different types of overlap relationship classifications between cadastral units A and B in test 2 that are known to exist *a priori* to testing. These can be seen in Figure 6.7 below.

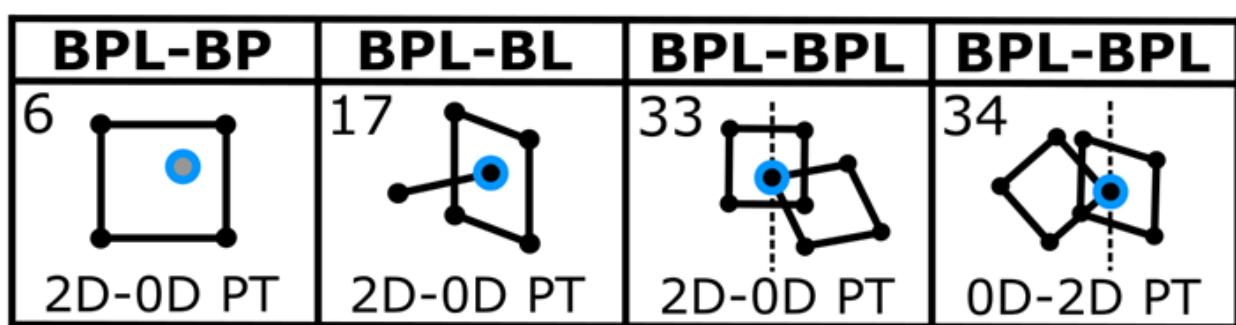


Figure 6.7: Testing Dataset 2 - Existing Overlap Relationship Classifications

Table 6.10 below shows a compiled summary of the program output for experimental testing dataset 2. It is organized into relationships between the six different component pair sets. Each component pair producing an overlap relationship classification was manually verified through visual inspection to be one of the *a priori* relationships shown in Figure 6.7 above. This was done by comparing the main classification algorithm program output from Table 6.10 with the experimental test 2 scenario visualizations shown in Figure 6.6.

Table 6.10: Testing Dataset 2 - Program Output

Test 2 - Component Pair Classification Results					
BP-BP			BPL-BL		
All Disjoint Relationship Classifications					
BL-BP					
All Disjoint Relationship Classifications					
BL-BL					
All Disjoint Relationship Classifications					
BPL-BP			BPL-BPL		
Component Pair	Projected Relationship	Relationship Description	Overlap Classification	Geometry	
AAA6-BB11	P_Intersect	2D-0D PT	17	[3,3,5]	
AAA6-BB12	P_Intersect	2D-0D PT	17	[3,3,5]	
AAA6-BB13	P_Intersect	2D-0D PT	17	[3,3,5]	
AAA6-BB16	P_Intersect	2D-0D PT	17	[3,3,5]	
AAA6-BB2	P_Intersect	2D-0D PT	17	[3,3,5]	
AAA6-BB3	P_Intersect	2D-0D PT	17	[3,3,5]	
Component Pair	Projected Relationship	Relationship Description	Overlap Classification	Geometry	
AAA6-BBB3	L_Intersect	2D-0D PT	33, 34	[3,3,5]	
AAA6-BBB4	L_Intersect	2D-0D PT	33, 34	[3,3,5]	
AAA6-BBB5	L_Intersect	2D-0D PT	33, 34	[3,3,5]	

Table 6.10 shows that all BP-BP, BL-BP, and BL-BL boundary component pairs in test 2 produce disjoint relationship classifications. There is a single BPL-BP pair that produces the 'Coplanar 2D-0D Point Touch' relationship (overlap classification 6). Six BPL-BL pairs produce the 'Point Intersect 2D-0D Point Touch' relationship (overlap classification 17). Three BPL-BPL pairs produce the 'Line Intersect 2D-0D/0D-2D Point Touch' relationships (symmetric overlap classifications 33 and 34).

6.2.3 Simulated Experimental Test 3

Simulated experimental test 3 consists of 3-D cadastral units A and B that are both (1m*1m*1m) volumetric cubes in 3-D space. This experiment is used to highlight overlap relationship classifications 3, 5, 8, 10, 11, 14, 15, 21, 22, 27, 30, 31, and 35 (see Figure 6.9). Visualizations of the boundary points, boundary lines, and boundary planes associated with each 3-D cadastral unit in this experiment can be seen in Figure 6.8 below.

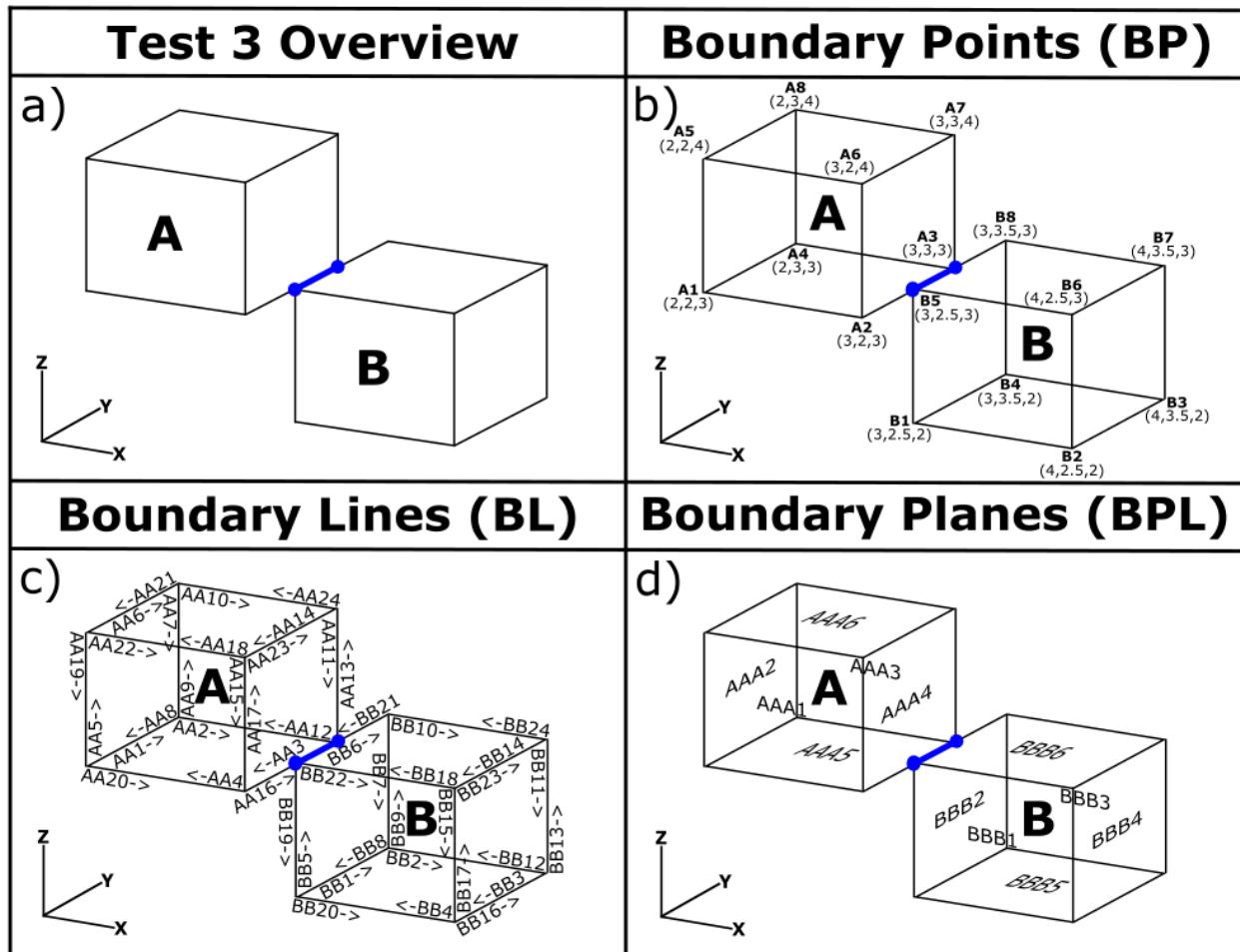


Figure 6.8: Testing Dataset 3 Visualization

Figure 6.8a shows an overview of simulated 3-D cadastral units A and B used in test 3. The main overlap between the two 3-D cadastral units in this experiment can be described by the partial line overlap existing between them. Unit A's floor (AAA5) is coplanar with unit B's ceiling (BBB6) and a wall of unit A (AAA4) is coplanar with a wall of unit B (BBB2). Figure 6.8b shows the names and 3-D coordinates of all boundary

points, Figure 6.8c shows the names and topological directions of all boundary lines, and Figure 6.8d shows the names of all boundary planes. These figures show that all output overlap classifications will be related to the relationships between the partial line overlap between boundary points A3 and B5 which should produce the 3-D point return coordinates [3,3,3], [3,2.5,3], or the 3-D line segment connecting them.

Tables 6.11 – 6.13 below show the object input storage parameters for the 3-D cadastral units A and B used to create experimental testing dataset 3. Table 6.11 shows the boundary point storage parameters, Table 6.12 shows the boundary line storage parameters, and Table 6.13 shows the boundary plane storage parameters used to define the boundaries of 3-D cadastral units A and B. These parameters were generated using the processes described in Section 5.1.

Table 6.11: Testing Dataset 3 - 3-D Boundary Point Coordinates

A Boundary Points		B Boundary Points											
Point ID	Point Name	X (e1)	Y (e2)	Z (e3)	no	ni	Point ID	Point Name	X (e1)	Y (e2)	Z (e3)	no	ni
1	A1	2	2	3	1	8.5	1	B1	3	2.5	2	1	9.625
2	A2	3	2	3	1	11	2	B2	4	2.5	2	1	13.125
3	A3	3	3	3	1	13.5	3	B3	4	3.5	2	1	16.125
4	A4	2	3	3	1	11	4	B4	3	3.5	2	1	12.625
5	A5	2	2	4	1	12	5	B5	3	2.5	3	1	12.125
6	A6	3	2	4	1	14.5	6	B6	4	2.5	3	1	15.625
7	A7	3	3	4	1	17	7	B7	4	3.5	3	1	18.625
8	A8	2	3	4	1	14.5	8	B8	3	3.5	3	1	15.125

Table 6.12: Testing Dataset 3 - 3-D Boundary Line Storage Parameters

A Boundary Lines									B Boundary Lines										
Line ID	Line Name	Start Point ID	End Point ID	CGA 3-Blade Parameters					Line ID	Line Name	Start Point ID	End Point ID	CGA 3-Blade Parameters						
				e1^no^ni	e2^no^ni	e3^no^ni	e1^e2^ni	e1^e3^ni					e1^no^ni	e2^no^ni	e3^no^ni	e1^e2^ni	e1^e3^ni	e2^e3^ni	
1	AA1	1	4	0	-1	0	2	0	-3	1	BB1	1	4	0	-1	0	3	0	-2
2	AA2	4	3	-1	0	0	-3	-3	0	2	BB2	4	3	-1	0	0	-3.5	-2	0
3	AA3	3	2	0	1	0	-3	0	3	3	BB3	3	2	0	1	0	-4	0	2
4	AA4	2	1	1	0	0	2	3	0	4	BB4	2	1	1	0	0	2.5	2	0
5	AA5	1	5	0	0	-1	0	2	2	5	BB5	1	5	0	0	-1	0	3	2.5
6	AA6	5	8	0	-1	0	2	0	-4	6	BB6	5	8	0	-1	0	3	0	-3
7	AA6	8	4	0	0	1	0	-2	-3	7	BB7	8	4	0	0	1	0	-3	-3.5
8	AA7	4	1	0	1	0	-2	0	3	8	BB8	4	1	0	1	0	-3	0	2
9	AA9	4	8	0	0	-1	0	2	3	9	BB9	4	8	0	0	-1	0	3	3.5
10	AA10	8	7	-1	0	0	-3	-4	0	10	BB10	8	7	-1	0	0	-3.5	-3	0
11	AA11	7	3	0	0	1	0	-3	-3	11	BB11	7	3	0	0	1	0	-4	-3.5
12	AA12	3	4	1	0	0	3	3	0	12	BB12	3	4	1	0	0	3.5	2	0
13	AA13	3	7	0	0	-1	0	3	3	13	BB13	3	7	0	0	-1	0	4	3.5
14	AA14	7	6	0	1	0	-3	0	4	14	BB14	7	6	0	1	0	-4	0	3
15	AA15	6	2	0	0	1	0	-3	-2	15	BB15	6	2	0	0	1	0	-4	-2.5
16	AA16	2	3	0	-1	0	3	0	-3	16	BB16	2	3	0	-1	0	4	0	-2
17	AA17	2	6	0	0	-1	0	3	2	17	BB17	2	6	0	0	-1	0	4	2.5
18	AA18	6	5	1	0	0	2	4	0	18	BB18	6	5	1	0	0	2.5	3	0
19	AA19	5	1	0	0	1	0	-2	-2	19	BB19	5	1	0	0	1	0	-3	-2.5
20	AA20	1	2	-1	0	0	-2	-3	0	20	BB20	1	2	-1	0	0	-2.5	-2	0
21	AA21	8	5	0	1	0	-2	0	4	21	BB21	8	5	0	1	0	-3	0	3
22	AA22	5	6	-1	0	0	-2	-4	0	22	BB22	5	6	-1	0	0	-2.5	-3	0
23	AA23	6	7	0	-1	0	3	0	-4	23	BB23	6	7	0	-1	0	4	0	-3
24	AA24	7	8	1	0	0	3	4	0	24	BB24	7	8	1	0	0	3.5	3	0

Table 6.13: Testing Dataset 3 - 3-D Boundary Plane Storage Parameters

A Boundary Planes					B Boundary Planes									
Plane ID	Plane Name	Polyline ID Numbers	CGA 4-Blade Parameters				Plane ID	Plane Name	Polyline ID Numbers	CGA 4-Blade Parameters				
			e1^e2^no^ni	e1^e3^no^ni	e2^e3^no^ni	e1^e2^e3^ni				e1^no^ni	e1^e3^no^ni	e2^e3^no^ni	e1^e2^e3^ni	
1	AAA1	17_18_19_20	0	1	0	-2	1	BBB1	17_18_19_20	0	1	0	-2.5	
2	AAA2	5_6_7_8	0	0	-1	-2	2	BBB2	5_6_7_8	0	0	-1	-3	
3	AAA3	9_10_11_12	0	-1	0	3	3	BBB3	9_10_11_12	0	-1	0	3.5	
4	AAA4	13_14_15_16	0	0	1	3	4	BBB4	13_14_15_16	0	0	1	4	
5	AAA5	1_2_3_4	-1	0	0	-3	5	BBB5	1_2_3_4	-1	0	0	-2	
6	AAA6	21_22_23_24	1	0	0	4	6	BBB6	21_22_23_24	1	0	0	3	

There are 14 different types of overlap relationship classifications between cadastral units A and B in test 3 that are known to exist *a priori* to testing. These can be seen in Figure 6.9 below.

BL-BP	BPL-BP	BL-BL	BL-BL	BL-BL	BPL-BL	BPL-BL
3 1D-0D PT	5 1D-0D PT	8 1D-1D LT	10 0D-1D PT	11 1D-0D PT	14 0D-1D PT	15 1D-0D PT
BPL-BL	BPL-BL	BPL-BPL	BPL-BPL	BPL-BPL	BPL-BPL	
21 1D-0D PT	22 1D-1D LT	27 1D-1D LT	30 1D-0D PT	31 0D-1D PT	35 1D-1D LT	

Figure 6.9: Testing Dataset 3 - Existing Overlap Relationship Classifications

Table 6.14 below shows a compiled summary of the program output for experimental testing dataset 3. It is organized into relationships between the six different component pair sets. Each component pair producing an overlap relationship classification was manually verified through visual inspection to be one of the *a priori* relationships shown in Figure 6.9 above. This was done by comparing the main classification algorithm program output from Table 6.14 with the experimental test 3 scenario visualizations shown in Figure 6.8.

Table 6.14: Testing Dataset 3 - Program Output

Test 3 - Component Pair Classification Results									
BP-BP					BPL-BP				
All Disjoint Relationship Classifications					Component Pair				
BL-BP									
Component Pair	Projected Relationship	Relationship Description	Overlap Classification	Geometry	Component Pair				
BB21-A3	Collinear	1D-0D PT	3	[3,3,3]	BBB2-A3	Coplanar	1D-0D PT	5	[3,3,3]
BB6-A3	Collinear	1D-0D PT	3	[3,3,3]	BBB6-A3	Coplanar	1D-0D PT	5	[3,3,3]
AA16-B5	Collinear	1D-0D PT	3	[3,2,5,3]	AAA4-B5	Coplanar	1D-0D PT	5	[3,2,5,3]
AA3-B5	Collinear	1D-0D PT	3	[3,2,5,3]	AAA5-B5	Coplanar	1D-0D PT	5	[3,2,5,3]
BL-BL									
Component Pair	Projected Relationship	Relationship Description	Overlap Classification	Geometry	Component Pair				
AA11-BB21	Coplanar	0D-1D PT	10, 11	[3,3,3]	BBB2-AA11	Coplanar	1D-0D PT	21	[3,3,3]
AA11-BB6	Coplanar	0D-1D PT	10, 11	[3,3,3]	BBB6-AA11	P Intersect	1D-0D PT	15	[3,3,3]
AA12-BB21	Coplanar	0D-1D PT	10, 11	[3,3,3]	BBB2-AA12	P Intersect	1D-0D PT	15	[3,3,3]
AA12-BB6	Coplanar	0D-1D PT	10, 11	[3,3,3]	BBB6-AA12	Coplanar	1D-0D PT	21	[3,3,3]
AA13-BB21	Coplanar	0D-1D PT	10, 11	[3,3,3]	BBB2-AA13	Coplanar	1D-0D PT	21	[3,3,3]
AA13-BB6	Coplanar	0D-1D PT	10, 11	[3,3,3]	BBB6-AA13	P Intersect	1D-0D PT	15	[3,3,3]
AA16-BB18	Coplanar	1D-0D PT	11, 10	[3,2,5,3]	BBB1-AA16	P Intersect	0D-1D PT	14	[3,2,5,3]
AA16-BB19	Coplanar	1D-0D PT	11, 10	[3,2,5,3]			0D-1D PT	--	[3,2,5,3]
AA16-BB21	Collinear (R)	1D-1D LT	8	[3,3,3] [3,2,5,3]			1D-1D LT	22	[3,2,5,3] [3,3,3]
AA16-BB22	Coplanar	1D-0D PT	11, 10	[3,2,5,3]			1D-0D PT	--	[3,3,3]
AA16-BB5	Coplanar	1D-0D PT	11, 10	[3,2,5,3]			0D-1D PT	--	[3,2,5,3]
AA16-BB6	Collinear (C)	1D-1D LT	8	[3,3,3] [3,2,5,3]			1D-1D LT	22	[3,2,5,3] [3,3,3]
AA2-BB21	Coplanar	0D-1D PT	10, 11	[3,3,3]			1D-0D PT	--	[3,3,3]
AA2-BB6	Coplanar	0D-1D PT	10, 11	[3,3,3]			1D-1D LT	22	[3,3,3] [3,2,5,3]
AA3-BB18	Coplanar	1D-0D PT	11, 10	[3,2,5,3]			0D-1D PT	--	[3,2,5,3]
AA3-BB19	Coplanar	1D-0D PT	11, 10	[3,2,5,3]			1D-1D LT	22	[3,3,3] [3,2,5,3]
AA3-BB21	Collinear (C)	1D-1D LT	8	[3,3,3] [3,2,5,3]			0D-1D PT	--	[3,3,3]
AA3-BB22	Coplanar	1D-0D PT	11, 10	[3,2,5,3]			1D-0D PT	--	[3,2,5,3]
AA3-BB5	Coplanar	1D-0D PT	11, 10	[3,2,5,3]			1D-0D PT	--	[3,2,5,3]
AA3-BB6	Collinear (R)	1D-1D LT	8	[3,3,3] [3,2,5,3]			1D-0D PT	--	[3,2,5,3]
BPL-BPL									
Component Pair	Projected Relationship	Relationship Description	Overlap Classification	Geometry	Component Pair				
AAA3-BBB2	L Intersect	0D-1D PT	31, 30	[3,3,3]	AAA4-BB21	Coplanar	1D-1D LT	22	[3,3,3] [3,2,5,3]
AAA3-BBB6	L Intersect	0D-1D PT	31, 30	[3,3,3]			1D-0D PT	--	[3,2,5,3]
AAA4-BBB1	L Intersect	1D-0D PT	30, 31	[3,2,5,3]			0D-1D PT	--	[3,3,3]
AAA4-BBB2	Coplanar	1D-0D PT	--	[3,2,5,3]			1D-1D LT	22	[3,3,3] [3,2,5,3]
		1D-1D LT	27	[3,2,5,3] [3,3,3]			0D-1D PT	--	[3,2,5,3]
		0D-1D PT	--	[3,3,3]			1D-0D PT	--	[3,3,3]
AAA4-BBB6	L Intersect	0D-1D PT	--	[3,3,3]			1D-1D LT	22	[3,3,3] [3,2,5,3]
		1D-1D LT	35	[3,3,3] [3,2,5,3]			0D-1D PT	--	[3,2,5,3]
		1D-0D PT	--	[3,2,5,3]			1D-0D PT	--	[3,2,5,3]
AAA5-BBB1	L Intersect	1D-0D PT	30, 31	[3,2,5,3]			1D-0D PT	--	[3,3,3]
AAA5-BBB2	L Intersect	1D-0D PT	--	[3,2,5,3]			1D-0D PT	--	[3,2,5,3]
		1D-1D LT	35	[3,2,5,3] [3,3,3]			1D-0D PT	--	[3,2,5,3]
		0D-1D PT	--	[3,3,3]			1D-1D LT	22	[3,2,5,3] [3,3,3]
AAA5-BBB6	Coplanar	1D-0D PT	--	[3,2,5,3]			0D-1D PT	--	[3,2,5,3]
AAA5-BBB6	Coplanar	1D-1D LT	27	[3,3,3] [3,2,5,3]			1D-0D PT	--	[3,2,5,3]
		1D-0D PT	--	[3,2,5,3]			1D-1D LT	22	[3,2,5,3] [3,3,3]
		1D-0D PT	--	[3,3,3]			0D-1D PT	--	[3,3,3]

Table 6.14 shows that all BP-BP boundary component pairs in test 3 produce a disjoint relationship classification. Four BL-BP pairs produce the 'Collinear 1D-0D Point Touch' relationship (overlap classification 3).

Four BL-BL pairs produce the 'Collinear (C/R) 1D-1D Line Touch' relationship (overlap classification 8), eight BL-BL pairs produce the 'Coplanar 0D-1D Point Touch' relationship (overlap classification 10), eight BL-BL pairs produce the 'Coplanar 1D-0D Point Touch' relationship (overlap classification 11), and four BPL-BP pairs produce the 'Coplanar 1D-0D Point Touch' relationship (overlap classification 5).

Four BPL-BL pairs produce the 'Point Intersect 0D-1D Point Touch' relationship (overlap classification 14), eight BPL-BL pairs produce the 'Point Intersect 1D-0D Point Touch' relationship (overlap classification 15), eight BPL-BL pairs produce the 'Coplanar 1D-0D Point Touch' relationship (overlap classification 21), and eight BPL-BL pairs produce the 'Coplanar 1D-1D Line Touch' relationship (overlap classification 22).

Two BPL-BPL pairs produce the 'Line Intersect 1D-0D Point Touch' relationship (overlap classification 30), two BPL-BPL pairs produce the 'Line Intersect 0D-1D Point Touch' relationship (overlap classification 30), two BPL-BPL pairs produce the 'Coplanar 1D-1D Line Touch' relationship (overlap classification 27), and two BPL-BPL pairs produce the 'Line Intersect 1D-1D Line Touch' relationship (overlap classification 35).

6.2.4 Simulated Experimental Test 4

Simulated experimental test 4 consists of 3-D cadastral units A and B in 3-D space. A is a (1m*1m*1m) volumetric cube while B has two diamond-shaped boundary planes and four rectangular boundary planes. This experiment is used to highlight overlap relationship classifications 3, 5, 10, 11, 14, 15, 20, 21, 23, 25, 26, 30, 31, 36, and 37 (see Figure 6.11). Visualizations of the boundary points, boundary lines, and boundary planes associated with each 3-D cadastral unit in this experiment can be seen in Figure 6.10 below.

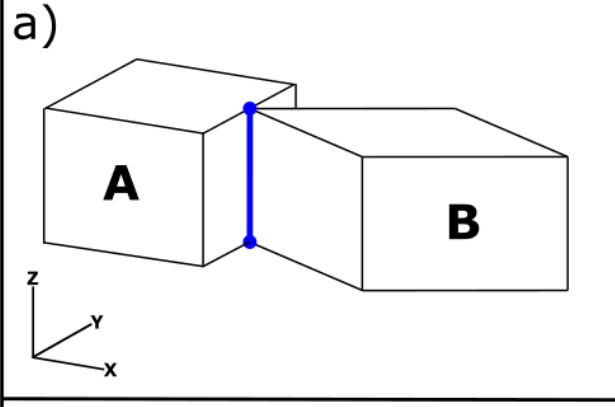
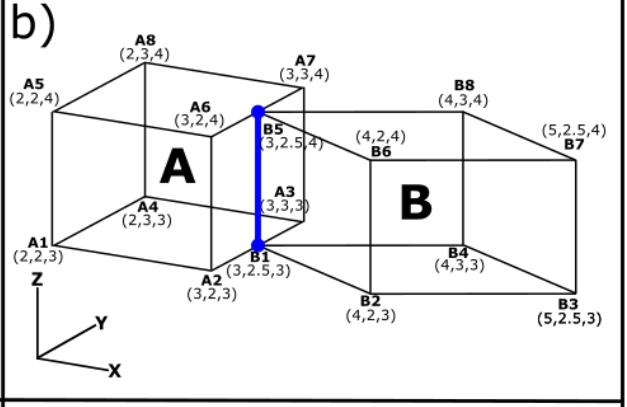
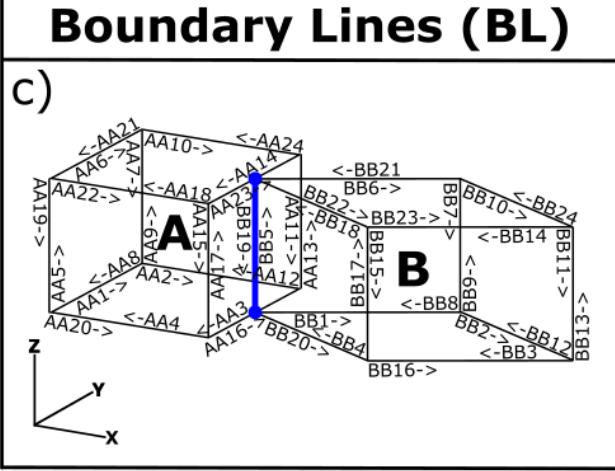
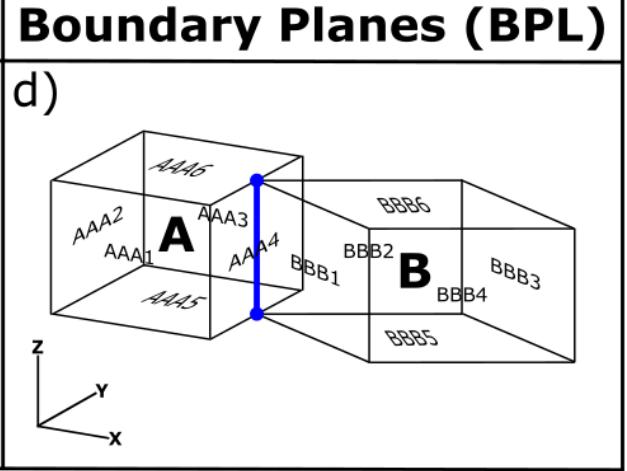
Test 4 Overview	Boundary Points (BP)
	
Boundary Lines (BL)	Boundary Planes (BPL)
	

Figure 6.10: Testing Dataset 4 Visualization

Figure 6.10a shows an overview of simulated 3-D cadastral units A and B used in test 4. The main overlap between the two 3-D cadastral units in this experiment can be described by the line touch existing between them. A boundary plane of Unit A (AAA4) touches two boundary planes of unit B (BBB1 and BBB2) at a line segment. Figure 6.10b shows the names and 3-D coordinates of all boundary points, Figure 6.10c shows the names and topological directions of all boundary lines, and Figure 6.10d shows the names of all boundary planes. Unit A's floor and ceiling are coplanar with that of unit B. These figures show that all output overlap classifications will be related to the relationship between boundary lines BB5 and BB19 and boundary planes AAA4, AAA5, and AAA6 which should produce the 3-D point return coordinates B1 [3,2,5,3] or B5 [3,2,5,4], or the 3-D line segment connecting them.

Tables 6.15 – 6.17 below show the object input storage parameters for the 3-D cadastral units A and B used to create experimental testing dataset 4. Table 6.15 shows the boundary point storage parameters, Table 6.16 shows the boundary line storage parameters, and Table 6.17 shows the boundary plane storage parameters used to define the boundaries of 3-D cadastral units A and B. These parameters were generated using the processes described in Section 5.1.

Table 6.15: Testing Dataset 4 - 3-D Boundary Point Coordinates

A Boundary Points						B Boundary Points							
Point ID	Point Name	X (e1)	Y (e2)	Z (e3)	no	ni	Point ID	Point Name	X (e1)	Y (e2)	Z (e3)	no	ni
1	A1	2	2	3	1	8.5	1	B1	3	2.5	3	1	12.125
2	A2	3	2	3	1	11	2	B2	4	2	3	1	14.5
3	A3	3	3	3	1	13.5	3	B3	5	2.5	3	1	20.125
4	A4	2	3	3	1	11	4	B4	4	3	3	1	17
5	A5	2	2	4	1	12	5	B5	3	2.5	4	1	15.625
6	A6	3	2	4	1	14.5	6	B6	4	2	4	1	18
7	A7	3	3	4	1	17	7	B7	5	2.5	4	1	23.625
8	A8	2	3	4	1	14.5	8	B8	4	3	4	1	20.5

Table 6.16: Testing Dataset 4 - 3-D Boundary Line Storage Parameters

A Boundary Lines								B Boundary Lines											
Line ID	Line Name	Start Point ID	End Point ID	CGA 3-Blade Parameters						Line ID	Line Name	Start Point ID	End Point ID	CGA 3-Blade Parameters					
				e1^no^ni	e2^no^ni	e3^no^ni	e1^e2^ni	e1^e3^ni	e2^e3^ni					e1^no^ni	e2^no^ni	e3^no^ni	e1^e2^ni	e1^e3^ni	e2^e3^ni
1	AA1	1	4	0	-1	0	2	0	-3	1	BB1	1	4	-1	-0.5	0	-1	-3	-1.5
2	AA2	4	3	-1	0	0	-3	-3	0	2	BB2	4	3	-1	0.5	0	-5	-3	1.5
3	AA3	3	2	0	1	0	-3	0	3	3	BB3	3	2	1	0.5	0	0	3	1.5
4	AA4	2	1	1	0	0	2	3	0	4	BB4	2	1	1	-0.5	0	4	3	-1.5
5	AA5	1	5	0	0	-1	0	2	2	5	BB5	1	5	0	0	-1	0	3	2.5
6	AA6	5	8	0	-1	0	2	0	-4	6	BB6	5	8	-1	-0.5	0	-1	-4	-2
7	AA6	8	4	0	0	1	0	-2	-3	7	BB7	8	4	0	0	1	0	-4	-3
8	AA7	4	1	0	1	0	-2	0	3	8	BB8	4	1	1	0.5	0	1	3	1.5
9	AA9	4	8	0	0	-1	0	2	3	9	BB9	4	8	0	0	-1	0	4	3
10	AA10	8	7	-1	0	0	-3	-4	0	10	BB10	8	7	-1	0.5	0	-5	-4	2
11	AA11	7	3	0	0	1	0	-3	-3	11	BB11	7	3	0	0	1	0	-5	-2.5
12	AA12	3	4	1	0	0	3	3	0	12	BB12	3	4	1	-0.5	0	5	3	-1.5
13	AA13	3	7	0	0	-1	0	3	3	13	BB13	3	7	0	0	-1	0	5	2.5
14	AA14	7	6	0	1	0	-3	0	4	14	BB14	7	6	1	0.5	0	0	4	2
15	AA15	6	2	0	0	1	0	-3	-2	15	BB15	6	2	0	0	1	0	-4	-2
16	AA16	2	3	0	-1	0	3	0	-3	16	BB16	2	3	-1	-0.5	0	0	-3	-1.5
17	AA17	2	6	0	0	-1	0	3	2	17	BB17	2	6	0	0	-1	0	4	2
18	AA18	6	5	1	0	0	2	4	0	18	BB18	6	5	1	-0.5	0	4	4	-2
19	AA19	5	1	0	0	1	0	-2	-2	19	BB19	5	1	0	0	1	0	-3	-2.5
20	AA20	1	2	-1	0	0	-2	-3	0	20	BB20	1	2	-1	0.5	0	-4	-3	1.5
21	AA21	8	5	0	1	0	-2	0	4	21	BB21	8	5	1	0.5	0	1	4	2
22	AA22	5	6	-1	0	0	-2	-4	0	22	BB22	5	6	-1	0.5	0	-4	-4	2
23	AA23	6	7	0	-1	0	3	0	-4	23	BB23	6	7	-1	-0.5	0	0	-4	-2
24	AA24	7	8	1	0	0	3	4	0	24	BB24	7	8	1	-0.5	0	5	4	-2

Table 6.17: Testing Dataset 4 - 3-D Boundary Plane Storage Parameters

A Boundary Planes						B Boundary Planes							
Plane ID	Plane Name	Polyline ID Numbers	CGA 4-Blade Parameters				Plane ID	Plane Name	Polyline ID Numbers	CGA 4-Blade Parameters			
			e1^e2^no^ni	e1^e3^no^ni	e2^e3^no^ni	e1^e2^e3^ni				e1^e2^no^ni	e1^e3^no^ni	e2^e3^no^ni	e1^e2^e3^ni
1	AAA1	17_18_19_20	0	1	0	-2	1	BBB1	17_18_19_20	0	1	-0.5	-4
2	AAA2	5_6_7_8	0	0	-1	-2	2	BBB2	5_6_7_8	0	-1	-0.5	1
3	AAA3	9_10_11_12	0	-1	0	3	3	BBB3	9_10_11_12	0	-1	0.5	5
4	AAA4	13_14_15_16	0	0	1	3	4	BBB4	13_14_15_16	0	1	0.5	0
5	AAA5	1_2_3_4	-1	0	0	-3	5	BBB5	1_2_3_4	-1	0	0	-3
6	AAA6	21_22_23_24	1	0	0	4	6	BBB6	21_22_23_24	1	0	0	4

There are 15 different types of overlap relationship classifications between cadastral units A and B in test 4 that are known to exist *a priori* to testing. These can be seen in Figure 6.11 below.

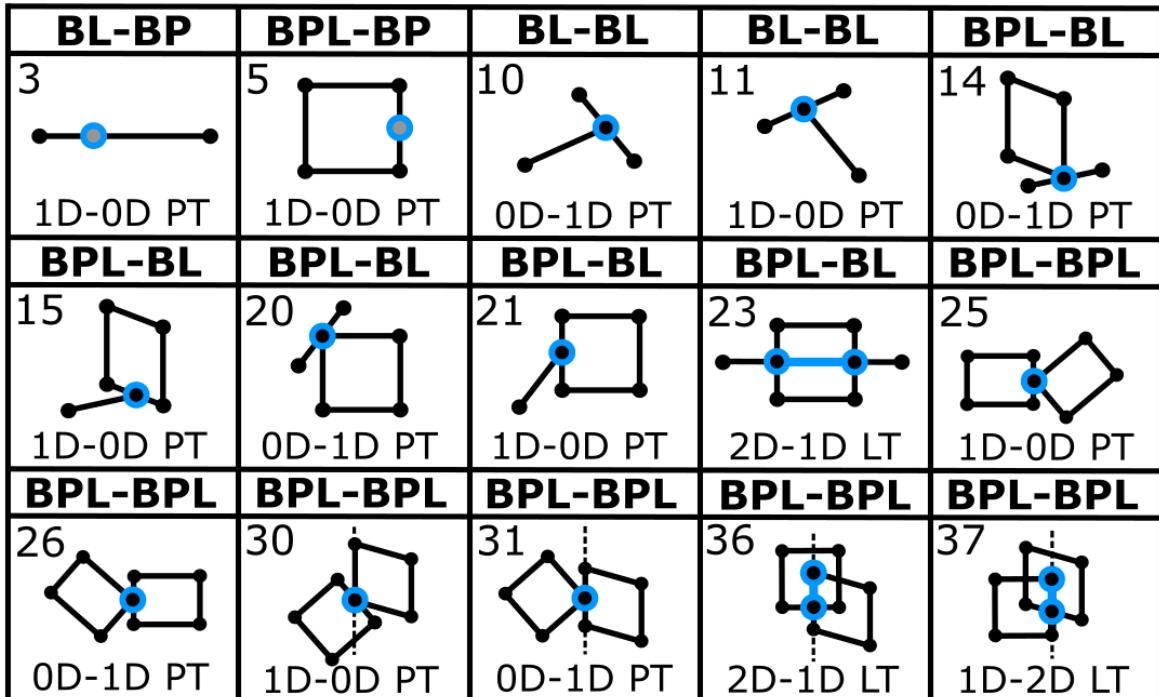


Figure 6.11: Testing Dataset 4 - Existing Overlap Relationship Classifications

Table 6.18 below shows a compiled summary of the program output for experimental testing dataset 4. It is organized into relationships between the six different component pair sets. Each component pair producing an overlap relationship classification was manually verified through visual inspection to be one of the *a priori* relationships shown in Figure 6.11 above. This was done by comparing the main classification algorithm program output from Table 6.18 with the experimental test 4 scenario visualizations shown in Figure 6.10.

Table 6.18: Testing Dataset 4 - Program Output

Test 4 - Component Pair Classification Results									
BP-BP					BPL-BL				
All Disjoint Relationship Classifications					Component Pair	Projected Relationship	Relationship Description	Overlap Classification	Geometry
BL-BP									
Component Pair	Projected Relationship	Relationship Description	Overlap Classification	Geometry	BBB1-AA14	P Intersect	0D-1D PT	14	[3,2,5,4]
AA16-B1	Collinear	1D-0D PT	3	[3,2,5,3]	BBB2-AA14	P Intersect	0D-1D PT	14	[3,2,5,4]
AA3-B1	Collinear	1D-0D PT	3	[3,2,5,3]	BBB6-AA14	Coplanar	0D-1D PT	20	[3,2,5,4]
AA14-B5	Collinear	1D-0D PT	3	[3,2,5,4]	BBB1-AA16	P Intersect	0D-1D PT	14	[3,2,5,3]
AA23-B5	Collinear	1D-0D PT	3	[3,2,5,4]	BBB2-AA16	P Intersect	0D-1D PT	14	[3,2,5,3]
BL-BL									
Component Pair	Projected Relationship	Relationship Description	Overlap Classification	Geometry	BBB6-AA23	P Intersect	0D-1D PT	14	[3,2,5,4]
AA14-BB18	Coplanar	1D-0D PT	10, 11	[3,2,5,4]	BBB1-AA23	P Intersect	0D-1D PT	14	[3,2,5,4]
AA14-BB19	Coplanar	1D-0D PT	10, 11	[3,2,5,4]	BBB6-AA23	Coplanar	0D-1D PT	20	[3,2,5,4]
AA14-BB21	Coplanar	1D-0D PT	10, 11	[3,2,5,4]	BBB1-AA3	P Intersect	0D-1D PT	14	[3,2,5,3]
AA14-BB22	Coplanar	1D-0D PT	10, 11	[3,2,5,4]	BBB2-AA3	P Intersect	0D-1D PT	14	[3,2,5,3]
AA14-BB5	Coplanar	1D-0D PT	10, 11	[3,2,5,4]	BBB5-AA3	Coplanar	0D-1D PT	20	[3,2,5,3]
AA14-BB6	Coplanar	1D-0D PT	10, 11	[3,2,5,4]	AAA4-BB18	P Intersect	1D-0D PT	15	[3,2,5,4]
AA16-BB1	Coplanar	1D-0D PT	10, 11	[3,2,5,3]	AAA6-BB18	Coplanar	1D-0D PT	21	[3,2,5,4]
AA16-BB19	Coplanar	1D-0D PT	10, 11	[3,2,5,3]			1D-0D PT	--	[3,2,5,4]
AA16-BB20	Coplanar	1D-0D PT	10, 11	[3,2,5,3]	AAA4-BB19	Coplanar	2D-1D LT	23	[3,2,5,4] [3,2,5,3]
AA16-BB4	Coplanar	1D-0D PT	10, 11	[3,2,5,3]			1D-0D PT	--	[3,2,5,3]
AA16-BB5	Coplanar	1D-0D PT	10, 11	[3,2,5,3]	AAA5-BB19	P Intersect	1D-0D PT	15	[3,2,5,3]
AA16-BB8	Coplanar	1D-0D PT	10, 11	[3,2,5,3]	AAA6-BB19	P Intersect	1D-0D PT	15	[3,2,5,4]
AA23-BB18	Coplanar	1D-0D PT	10, 11	[3,2,5,4]	AAA4-BB1	P Intersect	1D-0D PT	15	[3,2,5,3]
AA23-BB19	Coplanar	1D-0D PT	10, 11	[3,2,5,4]	AAA5-BB1	Coplanar	1D-0D PT	21	[3,2,5,3]
AA23-BB21	Coplanar	1D-0D PT	10, 11	[3,2,5,4]	AAA4-BB20	P Intersect	1D-0D PT	15	[3,2,5,3]
AA23-BB22	Coplanar	1D-0D PT	10, 11	[3,2,5,4]	AAA5-BB20	Coplanar	1D-0D PT	21	[3,2,5,3]
AA23-BB5	Coplanar	1D-0D PT	10, 11	[3,2,5,4]	AAA4-BB21	P Intersect	1D-0D PT	15	[3,2,5,4]
AA23-BB6	Coplanar	1D-0D PT	10, 11	[3,2,5,4]	AAA6-BB21	Coplanar	1D-0D PT	21	[3,2,5,4]
AA3-BB1	Coplanar	1D-0D PT	10, 11	[3,2,5,3]	AAA4-BB22	P Intersect	1D-0D PT	15	[3,2,5,4]
AA3-BB19	Coplanar	1D-0D PT	10, 11	[3,2,5,3]	AAA6-BB22	Coplanar	1D-0D PT	21	[3,2,5,4]
AA3-BB20	Coplanar	1D-0D PT	10, 11	[3,2,5,3]	AAA4-BB4	P Intersect	1D-0D PT	15	[3,2,5,3]
AA3-BB4	Coplanar	1D-0D PT	10, 11	[3,2,5,3]	AAA5-BB4	Coplanar	1D-0D PT	21	[3,2,5,3]
AA3-BB5	Coplanar	1D-0D PT	10, 11	[3,2,5,3]			1D-0D PT	--	[3,2,5,3]
AA3-BB8	Coplanar	1D-0D PT	10, 11	[3,2,5,3]	AAA4-BB5	Coplanar	2D-1D LT	23	[3,2,5,3] [3,2,5,4]
BPL-BP									
Component Pair	Projected Relationship	Relationship Description	Overlap Classification	Geometry					
AAA4-B1	Coplanar	1D-0D PT	5	[3,2,5,3]					
AAA5-B1	Coplanar	1D-0D PT	5	[3,2,5,3]					
AAA4-B5	Coplanar	1D-0D PT	5	[3,2,5,4]					
AAA6-B5	Coplanar	1D-0D PT	5	[3,2,5,4]					
BPL-BL									
Component Pair	Projected Relationship	Relationship Description	Overlap Classification	Geometry	AAA4-BBB1	L Intersect	1D-0D PT	--	[3,2,5,4]
							2D-1D LT	36, 37	[3,2,5,4] [3,2,5,3]
							1D-0D PT	--	[3,2,5,3]
					AAA4-BBB2	L Intersect	1D-0D PT	--	[3,2,5,3]
							2D-1D LT	36, 37	[3,2,5,3] [3,2,5,4]
							1D-0D PT	--	[3,2,5,4]
					AAA4-BBB5	L Intersect	1D-0D PT	30, 31	[3,2,5,3]
							1D-0D PT	30, 31	[3,2,5,4]
					AAA4-BBB6	L Intersect	1D-0D PT	30, 31	[3,2,5,3]
							1D-0D PT	30, 31	[3,2,5,3]
					AAA5-BBB1	L Intersect	1D-0D PT	30, 31	[3,2,5,4]
					AAA5-BBB2	L Intersect	1D-0D PT	30, 31	[3,2,5,3]
					AAA5-BBB5	Coplanar	1D-0D PT	25, 26	[3,2,5,3]
					AAA6-BBB1	L Intersect	1D-0D PT	30, 31	[3,2,5,4]
					AAA6-BBB2	L Intersect	1D-0D PT	30, 31	[3,2,5,4]
					AAA6-BBB6	Coplanar	1D-0D PT	25, 26	[3,2,5,4]

Table 6.18 shows that all BP-BP boundary component pairs in test 4 produce a disjoint relationship classification. Four BL-BP pairs produce the 'Collinear 1D-0D Point Touch' relationship (overlap classification 3).

24 BL-BL pairs produce the 'Coplanar 1D-0D/0D-1D Point Touch' relationship (symmetric overlap classifications 10 and 11). Four BPL-BP pairs produce the 'Coplanar 1D-0D Point Touch' relationship (overlap classification 5).

Eight BPL-BL pairs produce the 'Point Intersect 0D-1D Point Touch' relationship (overlap classification 14), 12 BPL-BL pairs produce the 'Point Intersect 1D-0D Point Touch' relationship (overlap classification 15), four BPL-BL pairs produce the 'Coplanar 0D-1D Point Touch' relationship (overlap classification 20), eight BPL-BL pairs produce the 'Coplanar 1D-0D Point Touch' relationship (overlap classification 21), and two BPL-BL pairs produce the 'Coplanar 2D-1D Line Touch' relationship (overlap classification 23).

Two BPL-BPL pairs produce the 'Coplanar 1D-0D/0D-1D Point Touch' relationship (symmetric overlap classifications 25 and 26), six BPL-BPL pairs produce the 'Line Intersect 1D-0D/0D-1D Point Touch' relationship (symmetric overlap classifications 30 and 31), and two BPL-BPL pairs produce the 'Line Intersect 2D-1D Line Touch' relationship (symmetric overlap classifications 36 and 37).

6.2.5 Simulated Experimental Test 5

Simulated experimental test 5 consists of 3-D cadastral units A and B that are both (2m*2m*2m) volumetric cubes in 3-D space. This experiment is used to highlight overlap relationship classifications 6, 12, 16, 17, 23, 28, 32, 36, and 37 (see Figure 6.13). Visualizations of the boundary points, boundary lines, and boundary planes associated with each 3-D cadastral unit in this experiment can be seen in Figure 6.12 below.

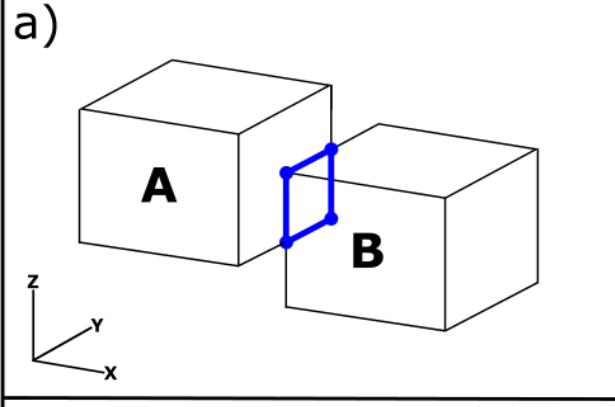
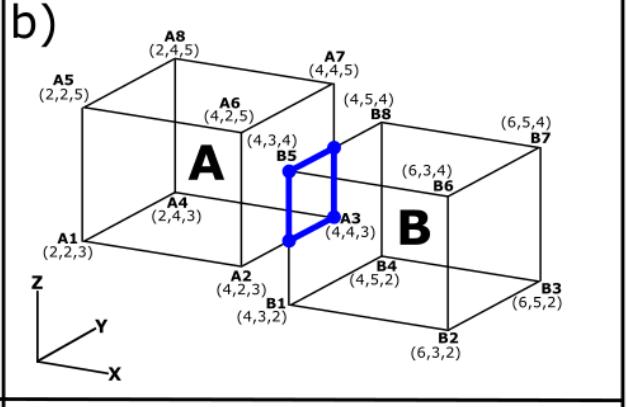
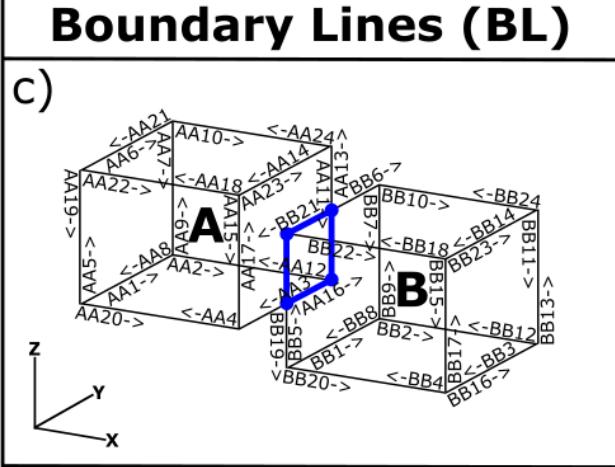
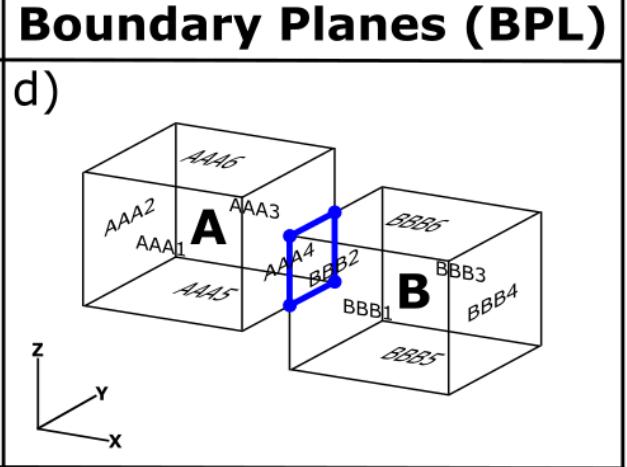
Test 5 Overview	Boundary Points (BP)
	
Boundary Lines (BL)	Boundary Planes (BPL)
	

Figure 6.12: Testing Dataset 5 Visualization

Figure 6.12a shows an overview of simulated 3-D cadastral units A and B used in test 5. The main overlap between the two 3-D cadastral units in this experiment can be described by the partial plane touch existing between them. A boundary plane of Unit A (AAA4) partially touches a boundary plane of unit B (BBB2) at a plane segment. Figure 6.12b shows the names and 3-D coordinates of all boundary points, Figure 6.12c shows the names and topological directions of all boundary lines, and Figure 6.12d shows the names of all boundary planes.

Tables 6.19 – 6.21 below show the object input storage parameters for the 3-D cadastral units A and B used to create experimental testing dataset 5. Table 6.19 shows the boundary point storage parameters, Table 6.20 shows the boundary line storage parameters, and Table 6.21 shows the boundary plane storage parameters used to define the boundaries of 3-D cadastral units A and B. These parameters were generated using the processes described in Section 5.1.

Table 6.19: Testing Dataset 5 - 3-D Boundary Point Coordinates

A Boundary Points						B Boundary Points							
Point ID	Point Name	X (e1)	Y (e2)	Z (e3)	no	ni	Point ID	Point Name	X (e1)	Y (e2)	Z (e3)	no	ni
1	A1	2	2	3	1	8.5	1	B1	4	3	2	1	14.5
2	A2	4	2	3	1	14.5	2	B2	6	3	2	1	24.5
3	A3	4	4	3	1	20.5	3	B3	6	5	2	1	32.5
4	A4	2	4	3	1	14.5	4	B4	4	5	2	1	22.5
5	A5	2	2	5	1	16.5	5	B5	4	3	4	1	20.5
6	A6	4	2	5	1	22.5	6	B6	6	3	4	1	30.5
7	A7	4	4	5	1	28.5	7	B7	6	5	4	1	38.5
8	A8	2	4	5	1	22.5	8	B8	4	5	4	1	28.5

Table 6.20: Testing Dataset 5 - 3-D Boundary Line Storage Parameters

A Boundary Lines								B Boundary Lines											
Line ID	Line Name	Start Point ID	End Point ID	CGA 3-Blade Parameters						Line ID	Line Name	Start Point ID	End Point ID	CGA 3-Blade Parameters					
				e1^no^ni	e2^no^ni	e3^no^ni	e1^e2^ni	e1^e3^ni	e2^e3^ni					e1^no^ni	e2^no^ni	e3^no^ni	e1^e2^ni	e1^e3^ni	e2^e3^ni
1	AA1	1	4	0	-2	0	4	0	-6	1	BB1	1	4	0	-2	0	8	0	-4
2	AA2	4	3	-2	0	0	-8	-6	0	2	BB2	4	3	-2	0	0	-10	-4	0
3	AA3	3	2	0	2	0	-8	0	6	3	BB3	3	2	0	2	0	-12	0	4
4	AA4	2	1	2	0	0	4	6	0	4	BB4	2	1	2	0	0	6	4	0
5	AA5	1	5	0	0	-2	0	4	4	5	BB5	1	5	0	0	-2	0	8	6
6	AA6	5	8	0	-2	0	4	0	-10	6	BB6	5	8	0	-2	0	8	0	-8
7	AA6	8	4	0	0	2	0	-4	-8	7	BB7	8	4	0	0	2	0	-8	-10
8	AA7	4	1	0	2	0	-4	0	6	8	BB8	4	1	0	2	0	-8	0	4
9	AA9	4	8	0	0	-2	0	4	8	9	BB9	4	8	0	0	-2	0	8	10
10	AA10	8	7	-2	0	0	-8	-10	0	10	BB10	8	7	-2	0	0	-10	-8	0
11	AA11	7	3	0	0	2	0	-8	-8	11	BB11	7	3	0	0	2	0	-12	-10
12	AA12	3	4	2	0	0	8	6	0	12	BB12	3	4	2	0	0	10	4	0
13	AA13	3	7	0	0	-2	0	8	8	13	BB13	3	7	0	0	-2	0	12	10
14	AA14	7	6	0	2	0	-8	0	10	14	BB14	7	6	0	2	0	-12	0	8
15	AA15	6	2	0	0	2	0	-8	-4	15	BB15	6	2	0	0	2	0	-12	-6
16	AA16	2	3	0	-2	0	8	0	-6	16	BB16	2	3	0	-2	0	12	0	-4
17	AA17	2	6	0	0	-2	0	8	4	17	BB17	2	6	0	0	-2	0	12	6
18	AA18	6	5	2	0	0	4	10	0	18	BB18	6	5	2	0	0	6	8	0
19	AA19	5	1	0	0	2	0	-4	-4	19	BB19	5	1	0	0	2	0	-8	-6
20	AA20	1	2	-2	0	0	-4	-6	0	20	BB20	1	2	-2	0	0	-6	-4	0
21	AA21	8	5	0	2	0	-4	0	10	21	BB21	8	5	0	2	0	-8	0	8
22	AA22	5	6	-2	0	0	-4	-10	0	22	BB22	5	6	-2	0	0	-6	-8	0
23	AA23	6	7	0	-2	0	8	0	-10	23	BB23	6	7	0	-2	0	12	0	-8
24	AA24	7	8	2	0	0	8	10	0	24	BB24	7	8	2	0	0	10	8	0

Table 6.21: Testing Dataset 5 - 3-D Boundary Plane Storage Parameters

A Boundary Planes						B Boundary Planes							
Plane ID	Plane Name	Polyline ID Numbers	CGA 4-Blade Parameters				Plane ID	Plane Name	Polyline ID Numbers	CGA 4-Blade Parameters			
			e1^e2^no^ni	e1^e3^no^ni	e2^e3^no^ni	e1^e2^e3^ni				e1^e2^no^ni	e1^e3^no^ni	e2^e3^no^ni	e1^e2^e3^ni
1	AAA1	17_18_19_20	0	4	0	-8	1	BBB1	17_18_19_20	0	4	0	-12
2	AAA2	5_6_7_8	0	0	-4	-8	2	BBB2	5_6_7_8	0	0	-4	-16
3	AAA3	9_10_11_12	0	-4	0	16	3	BBB3	9_10_11_12	0	-4	0	20
4	AAA4	13_14_15_16	0	0	4	16	4	BBB4	13_14_15_16	0	0	4	24
5	AAA5	1_2_3_4	-4	0	0	-12	5	BBB5	1_2_3_4	-4	0	0	-8
6	AAA6	21_22_23_24	4	0	0	20	6	BBB6	21_22_23_24	4	0	0	16

There are nine different types of overlap relationship classifications between cadastral units A and B in test 5 that are known to exist *a priori* to testing. These can be seen in Figure 6.13 below.

BPL-BP	BL-BL	BPL-BL	BPL-BL	BPL-BL
6 2D-0D PT	12 1D-1D PI	16 1D-1D PT	17 2D-0D PT	23 2D-1D LT
BPL-BPL	BPL-BPL	BPL-BPL	BPL-BPL	
28 2D-2D PLT	32 1D-1D PT	36 2D-1D LT	37 1D-2D LT	

Figure 6.13: Testing Dataset 5 - Existing Overlap Relationship Classifications

Table 6.22 below shows a compiled summary of the program output for experimental testing dataset 5. It is organized into relationships between the six different component pair sets. Each component pair producing an overlap relationship classification was manually verified through visual inspection to be one of the *a priori* relationships shown in Figure 6.13 above. This was done by comparing the main classification algorithm program output from Table 6.22 with the experimental test 5 scenario visualizations shown in Figure 6.12.

Table 6.22: Testing Dataset 5 - Program Output

Test 5 - Component Pair Classification Results					
BP-BP			BL-BP		
All Disjoint Relationship Classifications			All Disjoint Relationship Classifications		
BL-BL					
Component Pair	Projected Relationship	Relationship Description	Overlap Classification	Geometry	
AA11-BB21	Coplanar	1D-1D PI	12	[4,4,4]	
AA11-BB6	Coplanar	1D-1D PI	12	[4,4,4]	
AA13-BB21	Coplanar	1D-1D PI	12	[4,4,4]	
AA13-BB6	Coplanar	1D-1D PI	12	[4,4,4]	
AA16-BB19	Coplanar	1D-1D PI	12	[4,3,3]	
AA16-BB5	Coplanar	1D-1D PI	12	[4,3,3]	
AA3-BB19	Coplanar	1D-1D PI	12	[4,3,3]	
AA3-BB5	Coplanar	1D-1D PI	12	[4,3,3]	
BPL-BP					
Component Pair	Projected Relationship	Relationship Description	Overlap Classification	Geometry	
BBB2-A3	Coplanar	2D-0D PT	6	[4,4,3]	
AAA4-B5	Coplanar	2D-0D PT	6	[4,3,4]	
BPL-BPL					
Component Pair	Projected Relationship	Relationship Description	Overlap Classification	Geometry	
AAA3-BBB2	L Intersect	1D-1D PT	--	[4,4,4]	
		2D-1D LT	36, 37	[4,4,4] [4,4,3]	
		1D-0D PT	--	[4,4,3]	
AAA3-BBB6	L Intersect	1D-1D PT	32	[4,4,4]	
		2D-0D PT	--	[4,3,4]	
AAA4-BBB1	L Intersect	2D-1D LT	36, 37	[4,3,4] [4,3,3]	
		1D-1D PT	--	[4,3,3]	
		1D-1D PT	--	[4,3,3]	
		2D-1D LT	23 (28)	[4,3,3] [4,3,4]	
		2D-0D PT	--	[4,3,4]	
AAA4-BBB2	Coplanar (2D-2D PLT)	2D-1D LT	23 (28)	[4,3,4] [4,4,4]	
		1D-1D PT	--	[4,4,4]	
		2D-1D LT	23 (28)	[4,4,4] [4,4,3]	
		2D-0D PT	--	[4,4,3]	
		2D-1D LT	23 (28)	[4,4,3] [4,3,3]	
AAA4-BBB6	L Intersect	1D-1D PT	--	[4,4,4]	
	2D-1D LT	36, 37	[4,4,4] [4,3,4]		
	2D-0D PT	--	[4,3,4]		
AAA5-BBB1	L Intersect	1D-1D PT	32	[4,3,3]	
		2D-0D PT	--	[4,4,3]	
AAA5-BBB2	L Intersect	2D-1D LT	36, 37	[4,4,3] [4,3,3]	
		1D-1D PT	--	[4,3,3]	
BL-BL					
Component Pair	Projected Relationship	Relationship Description	Overlap Classification	Geometry	
BBB2-AA11	Coplanar	1D-1D PT	--	[4,4,4]	
		2D-1D LT	23	[4,4,4] [4,4,3]	
		2D-0D PT	--	[4,4,3]	
BBB6-AA11	P Intersect	1D-1D PT	16	[4,4,4]	
BBB2-AA12	P Intersect	2D-0D PT	17	[4,4,3]	
		2D-0D PT	--	[4,4,3]	
BBB2-AA13	Coplanar	2D-1D LT	23	[4,4,3] [4,4,4]	
		1D-1D PT	--	[4,4,4]	
BBB6-AA13	P Intersect	1D-1D PT	16	[4,4,4]	
BBB1-AA16	P Intersect	1D-1D PT	16	[4,3,3]	
		1D-1D PT	--	[4,3,3]	
BBB2-AA16	Coplanar	2D-1D LT	23	[4,3,3] [4,4,3]	
		2D-0D PT	--	[4,4,3]	
BBB2-AA2	P Intersect	2D-0D PT	17	[4,4,3]	
BBB1-AA3	P Intersect	1D-1D PT	16	[4,3,3]	
		2D-0D PT	--	[4,4,3]	
BBB2-AA3	Coplanar	2D-1D LT	23	[4,4,3] [4,3,3]	
		1D-1D PT	--	[4,3,3]	
AAA4-BB18	P Intersect	2D-0D PT	17	[4,3,4]	
		2D-0D PT	--	[4,3,4]	
AAA4-BB19	Coplanar	2D-1D LT	23	[4,3,4] [4,3,3]	
		1D-1D PT	--	[4,3,3]	
AAA5-BB19	P Intersect	1D-1D PT	16	[4,3,3]	
AAA3-BB21	P Intersect	1D-1D PT	16	[4,4,4]	
		1D-1D PT	--	[4,4,4]	
AAA4-BB21	Coplanar	2D-1D LT	23	[4,4,4] [4,3,4]	
		2D-0D PT	--	[4,3,4]	
AAA4-BB22	P Intersect	2D-0D PT	17	[4,3,4]	
		1D-1D PT	--	[4,3,3]	
AAA4-BB5	Coplanar	2D-1D LT	23	[4,3,3] [4,3,4]	
		2D-0D PT	--	[4,3,4]	
AAA5-BB5	P Intersect	1D-1D PT	16	[4,3,3]	
AAA3-BB6	P Intersect	1D-1D PT	16	[4,4,4]	
		2D-0D PT	--	[4,3,4]	
AAA4-BB6	Coplanar	2D-1D LT	23	[4,3,4] [4,4,4]	
		1D-1D PT	--	[4,4,4]	

Table 6.22 shows that all BP-BP and BL-BP boundary component pairs in test 5 produce a disjoint relationship classification. Eight BL-BL pairs produce the 'Coplanar 1D-1D Point Intersection' relationship (overlap classification 12) and two BPL-BP pairs produce the 'Coplanar 2D-0D Point Touch' relationship (overlap classification 6).

Eight BPL-BL pairs produce the 'Point Intersect 1D-1D Point Touch' relationship (overlap classification 16), four BPL-BL pairs produce the 'Point Intersect 2D-0D Point Touch' relationship (overlap classification 17), and eight BPL-BL pairs produce the 'Coplanar 2D-1D Line Touch' relationship (overlap classification 23).

One BPL-BPL pair (AAA4-BBB2) produces the 'Coplanar 2D-2D Plane Touch' relationship (overlap classification 28) consisting of four 'Coplanar 2D-1D Line Touch' segments. Two BPL-BPL pairs produce the 'Line Intersect 1D-1D Point Touch' relationship (overlap classification 32), and four BPL-BPL pairs produce the 'Line Intersect 2D-1D/1D-2D Line Touch' relationship (symmetric overlap classifications 36 and 37).

6.2.6 Simulated Experimental Test 6

Simulated experimental test 6 consists of 3-D cadastral units A and B in 3-D space. Unit A is a (2m*2m*2m) volumetric cube and unit B is a (2m*2m*3m) volumetric rectangular prism. This experiment is used to highlight overlap relationship classifications 6, 12, 16, 17, 18, 23, 28, 36, 37, and 38 (see Figure 6.15). Visualizations of the boundary points, boundary lines, and boundary planes associated with each 3-D cadastral unit in this experiment can be seen in Figure 6.14 below.

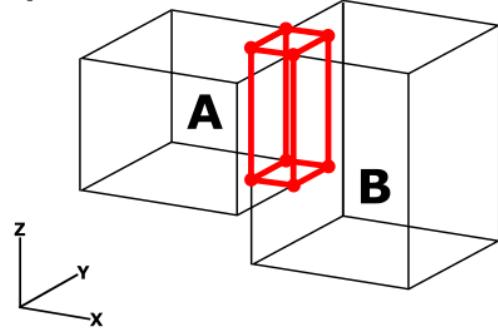
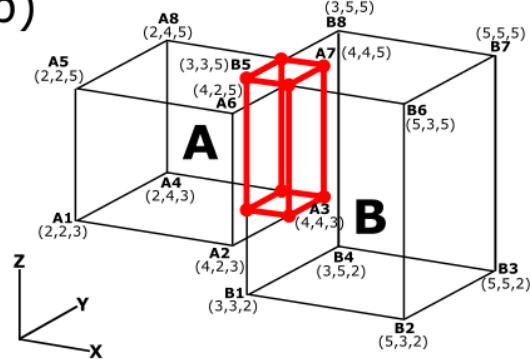
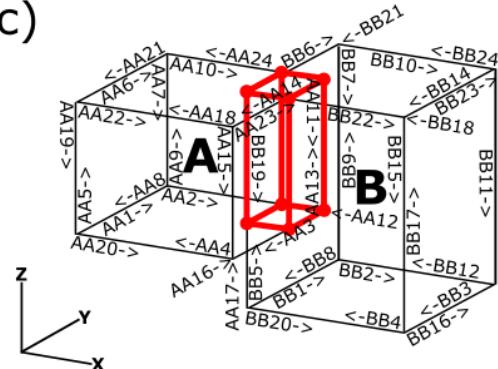
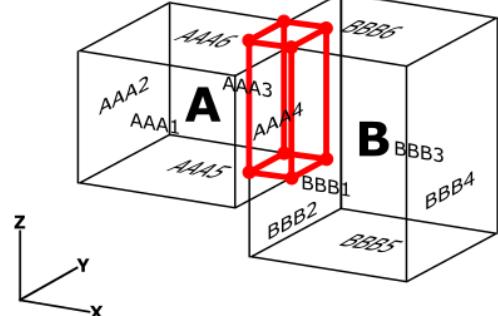
Test 6 Overview	Boundary Points (BP)
<p>a)</p> 	
Boundary Lines (BL)	Boundary Planes (BPL)
<p>c)</p> 	

Figure 6.14: Testing Dataset 6 Visualization

Figure 6.14a shows an overview of simulated 3-D cadastral units A and B used in test 6. The main overlap between the two 3-D cadastral units in this experiment can be described by the partial intersection of multiple boundary planes between them. BBB1 intersects with AAA4 and AAA5 while BBB2 intersects with AAA3 and AAA5. Figure 6.14b shows the names and 3-D coordinates of all boundary points, Figure 6.14c shows the names and topological directions of all boundary lines, and Figure 6.14d shows the names of all boundary planes.

Tables 6.23 – 6.25 below show the object input storage parameters for the 3-D cadastral units A and B used to create experimental testing dataset 6. Table 6.23 shows the boundary point storage parameters, Table 6.24 shows the boundary line storage parameters, and Table 6.25 shows the boundary plane storage parameters used to define the boundaries of 3-D cadastral units A and B. These parameters were generated using the processes described in Section 5.1.

Table 6.23: Testing Dataset 6 - 3-D Boundary Point Coordinates

A Boundary Points						B Boundary Points							
Point ID	Point Name	X (e1)	Y (e2)	Z (e3)	no	ni	Point ID	Point Name	X (e1)	Y (e2)	Z (e3)	no	ni
1	A1	2	2	3	1	8.5	1	B1	3	3	2	1	11
2	A2	4	2	3	1	14.5	2	B2	5	3	2	1	19
3	A3	4	4	3	1	20.5	3	B3	5	5	2	1	27
4	A4	2	4	3	1	14.5	4	B4	3	5	2	1	19
5	A5	2	2	5	1	16.5	5	B5	3	3	5	1	21.5
6	A6	4	2	5	1	22.5	6	B6	5	3	5	1	29.5
7	A7	4	4	5	1	28.5	7	B7	5	5	5	1	37.5
8	A8	2	4	5	1	22.5	8	B8	3	5	5	1	29.5

Table 6.24: Testing Dataset 6 - 3-D Boundary Line Storage Parameters

A Boundary Lines								B Boundary Lines											
Line ID	Line Name	Start Point ID	End Point ID	CGA 3-Blade Parameters						Line ID	Line Name	Start Point ID	End Point ID	CGA 3-Blade Parameters					
				e1^no^ni	e2^no^ni	e3^no^ni	e1^e2^ni	e1^e3^ni	e2^e3^ni					e1^no^ni	e2^no^ni	e3^no^ni	e1^e2^ni	e1^e3^ni	e2^e3^ni
1	AA1	1	4	0	-2	0	4	0	-6	1	BB1	1	4	0	-2	0	6	0	-4
2	AA2	4	3	-2	0	0	-8	-6	0	2	BB2	4	3	-2	0	0	-10	-4	0
3	AA3	3	2	0	2	0	-8	0	6	3	BB3	3	2	0	2	0	-10	0	4
4	AA4	2	1	2	0	0	4	6	0	4	BB4	2	1	2	0	0	6	4	0
5	AA5	1	5	0	0	-2	0	4	4	5	BB5	1	5	0	0	-3	0	9	9
6	AA6	5	8	0	-2	0	4	0	-10	6	BB6	5	8	0	-2	0	6	0	-10
7	AA6	8	4	0	0	2	0	-4	-8	7	BB7	8	4	0	0	3	0	-9	-15
8	AA7	4	1	0	2	0	-4	0	6	8	BB8	4	1	0	2	0	-6	0	4
9	AA9	4	8	0	0	-2	0	4	8	9	BB9	4	8	0	0	-3	0	9	15
10	AA10	8	7	-2	0	0	-8	-10	0	10	BB10	8	7	-2	0	0	-10	-10	0
11	AA11	7	3	0	0	2	0	-8	-8	11	BB11	7	3	0	0	3	0	-15	-15
12	AA12	3	4	2	0	0	8	6	0	12	BB12	3	4	2	0	0	10	4	0
13	AA13	3	7	0	0	-2	0	8	8	13	BB13	3	7	0	0	-3	0	15	15
14	AA14	7	6	0	2	0	-8	0	10	14	BB14	7	6	0	2	0	-10	0	10
15	AA15	6	2	0	0	2	0	-8	-4	15	BB15	6	2	0	0	3	0	-15	-9
16	AA16	2	3	0	-2	0	8	0	-6	16	BB16	2	3	0	-2	0	10	0	-4
17	AA17	2	6	0	0	-2	0	8	4	17	BB17	2	6	0	0	-3	0	15	9
18	AA18	6	5	2	0	0	4	10	0	18	BB18	6	5	2	0	0	6	10	0
19	AA19	5	1	0	0	2	0	-4	-4	19	BB19	5	1	0	0	3	0	-9	-9
20	AA20	1	2	-2	0	0	-4	-6	0	20	BB20	1	2	-2	0	0	-6	-4	0
21	AA21	8	5	0	2	0	-4	0	10	21	BB21	8	5	0	2	0	-6	0	10
22	AA22	5	6	-2	0	0	-4	-10	0	22	BB22	5	6	-2	0	0	-6	-10	0
23	AA23	6	7	0	-2	0	8	0	-10	23	BB23	6	7	0	-2	0	10	0	-10
24	AA24	7	8	2	0	0	8	10	0	24	BB24	7	8	2	0	0	10	10	0

Table 6.25: Testing Dataset 6 - 3-D Boundary Plane Storage Parameters

A Boundary Planes						B Boundary Planes							
Plane ID	Plane Name	Polyline ID Numbers	CGA 4-Blade Parameters				Plane ID	Plane Name	Polyline ID Numbers	CGA 4-Blade Parameters			
			e1^e2^no^ni	e1^e3^no^ni	e2^e3^no^ni	e1^e2^e3^ni				e1^e2^no^ni	e1^e3^no^ni	e2^e3^no^ni	e1^e2^e3^ni
1	AAA1	17_18_19_20	0	4	0	-8	1	BBB1	17_18_19_20	0	6	0	-18
2	AAA2	5_6_7_8	0	0	-4	-8	2	BBB2	5_6_7_8	0	0	-6	-18
3	AAA3	9_10_11_12	0	-4	0	16	3	BBB3	9_10_11_12	0	-6	0	30
4	AAA4	13_14_15_16	0	0	4	16	4	BBB4	13_14_15_16	0	0	6	30
5	AAA5	1_2_3_4	-4	0	0	-12	5	BBB5	1_2_3_4	-4	0	0	-8
6	AAA6	21_22_23_24	4	0	0	20	6	BBB6	21_22_23_24	4	0	0	20

There are 10 different types of overlap relationship classifications between cadastral units A and B in test 6 that are known to exist *a priori* to testing. These can be seen in Figure 6.15 below.

BPL-BP	BL-BL	BPL-BL	BPL-BL	BPL-BL
6 2D-0D PT	12 1D-1D PI	16 1D-1D PT	17 2D-0D PT	18 2D-1D PI
BPL-BL	BPL-BPL	BPL-BPL	BPL-BPL	BPL-BPL
23 2D-1D LT	28 2D-2D PLT	36 2D-1D LT	37 1D-2D LT	38 2D-2D LI

Figure 6.15: Testing Dataset 6 - Existing Overlap Relationship Classifications

Table 6.26 below shows a compiled summary of the program output for experimental testing dataset 6. It is organized into relationships between the six different component pair sets. Each component pair producing an overlap relationship classification was manually verified through visual inspection to be one of the *a priori* relationships shown in Figure 6.15 above. This was done by comparing the main classification algorithm program output from Table 6.26 with the experimental test 6 scenario visualizations shown in Figure 6.14.

Table 6.26: Testing Dataset 6 - Program Output

Test 6 - Component Pair Classification Results										
BP-BP			BL-BP							
All Disjoint Relationship Classifications					All Disjoint Relationship Classifications					
BPL-BP					BPL-BL					
Component Pair						Component Pair	Projected Relationship	Relationship Description	Overlap Classification	Geometry
BBB6-A7	Coplanar	2D-0D PT	6	[4,4,5]	BBB2-AA10	P Intersect	1D-1D PT	16	[3,4,5]	
AAA6-B5	Coplanar	2D-0D PT	6	[3,3,5]	BBB6-AA10	Coplanar	1D-1D PT	--	[3,4,5]	
BL-BL					BBB6-AA11		2D-1D LT	23	[3,4,5] [4,4,5]	
Component Pair	Projected Relationship	Relationship Description	Overlap Classification	Geometry	BBB6-AA12		2D-0D PT	--	[4,4,5]	
AA10-BB21	Coplanar	1D-1D PI	12	[3,4,5]	BBB6-AA13	P Intersect	2D-0D PT	17	[4,4,5]	
AA10-BB6	Coplanar	1D-1D PI	12	[3,4,5]	BBB1-AA14		1D-1D PT	16	[4,3,5]	
AA14-BB18	Coplanar	1D-1D PI	12	[4,3,5]	BBB6-AA14	Coplanar	2D-0D PT	--	[4,4,5]	
AA14-BB22	Coplanar	1D-1D PI	12	[4,3,5]	BBB6-AA14		2D-1D LT	23	[4,4,5] [4,3,5]	
AA23-BB18	Coplanar	1D-1D PI	12	[4,3,5]	BBB1-AA16	P Intersect	1D-1D PI	18	[4,3,3]	
AA23-BB22	Coplanar	1D-1D PI	12	[4,3,5]	BBB1-AA23		1D-1D PT	16	[4,3,5]	
AA24-BB21	Coplanar	1D-1D PI	12	[3,4,5]	BBB6-AA23	Coplanar	1D-1D PT	--	[4,3,5]	
AA24-BB6	Coplanar	1D-1D PI	12	[3,4,5]	BBB6-AA23		2D-1D LT	23	[4,3,5] [4,4,5]	
BPL-BPL					BBB6-AA23		2D-0D PT	--	[4,4,5]	
Component Pair	Projected Relationship	Relationship Description	Overlap Classification	Geometry	BBB2-AA24	P Intersect	1D-1D PT	16	[3,4,5]	
AAA3-BBB2	L Intersect	1D-1D PT	16 (BPL-BL)	[3,4,5]	BBB2-AA24		2D-0D PT	--	[4,4,5]	
		2D-2D LI	38	[3,4,5] [3,4,3]	BBB6-AA24	Coplanar	2D-1D LT	23	[4,4,5] [3,4,5]	
	L Intersect	2D-1D PI	18 (BPL-BL)	[3,4,3]	BBB2-AA24		1D-1D PT	--	[3,4,5]	
		1D-1D PT	--	[3,4,5]	BBB2-AA25	P Intersect	2D-1D PI	18	[3,4,3]	
	L Intersect	2D-1D LT	36, 37	[3,4,5] [4,4,5]	BBB1-AA25		2D-1D PI	18	[4,3,3]	
		2D-0D PT	--	[4,4,5]	AAA4-BB18	P Intersect	1D-1D PT	16	[4,3,5]	
	L Intersect	1D-1D PT	16 (BPL-BL)	[4,3,5]	AAA4-BB18		1D-1D PT	--	[4,3,5]	
		2D-2D LI	38	[4,3,5] [4,3,3]	AAA6-BB18	Coplanar	2D-1D LT	23	[4,3,5] [3,3,5]	
	L Intersect	2D-1D PI	18 (BPL-BL)	[4,3,3]	AAA6-BB18		2D-0D PT	--	[3,3,5]	
		2D-0D PT	--	[4,4,5]	AAA5-BB19	P Intersect	2D-1D PI	18	[3,3,3]	
	L Intersect	2D-1D LT	36, 37	[4,4,5] [4,3,5]	AAA6-BB19		2D-0D PT	17	[3,3,5]	
		1D-1D PT	--	[4,3,5]	AAA3-BB21	P Intersect	1D-1D PT	16	[3,4,5]	
	L Intersect	2D-1D PI	18 (BPL-BL)	[3,3,3]	AAA3-BB21		1D-1D PT	--	[3,4,5]	
		2D-2D LI	38	[3,3,3] [3,4,3]	AAA6-BB21	Coplanar	2D-1D LT	23	[3,4,5] [3,3,5]	
	L Intersect	2D-1D PI	18 (BPL-BL)	[3,4,3]	AAA6-BB21		2D-0D PT	--	[3,3,5]	
		2D-0D PT	--	[4,4,5]	AAA4-BB22	P Intersect	1D-1D PT	16	[4,3,5]	
	L Intersect	2D-1D LT	36, 37	[3,3,3] [4,3,5]	AAA4-BB22		2D-0D PT	--	[3,3,5]	
		1D-1D PT	--	[4,3,5]	AAA6-BB22	Coplanar	2D-1D LT	23	[3,3,5] [4,3,5]	
	L Intersect	1D-1D PT	--	[4,3,5]	AAA6-BB22		1D-1D PT	--	[4,3,5]	
		2D-1D LT	36, 37	[4,3,5] [3,3,5]	AAA5-BB5	P Intersect	2D-1D PI	18	[3,3,3]	
	L Intersect	2D-0D PT	--	[3,3,5]	AAA6-BB5		2D-0D PT	17	[3,3,5]	
		2D-0D PT	--	[3,3,5]	AAA3-BB6	P Intersect	1D-1D PT	16	[3,4,5]	
	L Intersect	2D-1D LT	36, 37	[3,3,5] [3,4,5]	AAA3-BB6		2D-0D PT	--	[3,3,5]	
		1D-1D PT	--	[3,4,5]	AAA6-BB6	Coplanar	2D-1D LT	23	[3,3,5] [3,4,5]	
	Coplanar (2D-2D PLT)	1D-1D PT	--	[3,4,5]	AAA6-BB6		1D-1D PT	--	[3,4,5]	
		2D-1D LT	23 (28)	[3,4,5] [3,3,5]						
		2D-0D PT	--	[3,3,5]						
		2D-1D LT	23 (28)	[3,3,5] [4,3,5]						
		1D-1D PT	--	[4,3,5]						
		2D-1D LT	23 (28)	[4,3,5] [4,4,5]						
		2D-0D PT	--	[4,4,5]						
		2D-1D LT	23 (28)	[4,4,5] [3,4,5]						

Table 6.26 shows that all BP-BP and BL-BP boundary component pairs in test 6 produce a disjoint relationship classification. Eight BL-BL pairs produce the 'Coplanar 1D-1D Point Intersection' relationship (overlap classification 12). Two BPL-BP pairs produce the 'Coplanar 2D-0D Point Touch' relationship (overlap classification 6).

Eight BPL-BL pairs produce the 'Point Intersect 1D-1D Point Touch' relationship (overlap classification 16), four BPL-BL pairs produce the 'Point Intersect 2D-0D Point Touch' relationship (overlap classification 17), six BPL-BL pairs produce the 'Point Intersect 2D-1D Point Intersection' relationship (overlap classification 18), and eight BPL-BL pairs produce the 'Coplanar 2D-1D Line Touch' relationship (overlap classification 23).

One BPL-BPL pair (AAA6-BBB6) produces the 'Coplanar 2D-2D Plane Touch' relationship (overlap classification 28) consisting of four 'Coplanar 2D-1D Line Touch' segments. Four BPL-BPL pairs produce the 'Line Intersect 2D-1D/1D-2D Line Touch' relationship (symmetric overlap classifications 36 and 37) and four BPL-BPL pairs produce the 'Line Intersect 2D-2D Line Intersection' relationship (overlap classification 38). Overlap classifications 38 (in bold) were manually derived from the implementation results whenever one of the BPL-BL point classifications (not in bold) associated with each BPL-BPL pair resulted in the '2D-1D Point Intersection' relationship (overlap classification 18).

6.2.7 Simulated Experimental Test 7

Simulated experimental test 7 consists of 3-D cadastral units A and B that are both (1m*1m*1m) volumetric cubes in 3-D space. This experiment is used to highlight all 15 disjoint relationship classifications (see Figure 6.17). Visualizations of the boundary points, boundary lines, and boundary planes associated with each 3-D cadastral unit in this experiment can be seen in Figure 6.16 below.

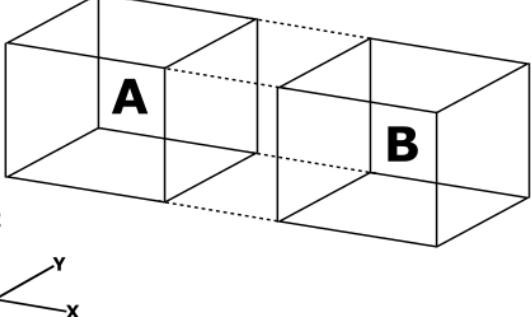
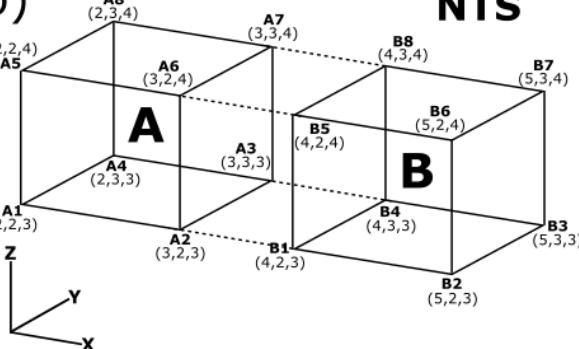
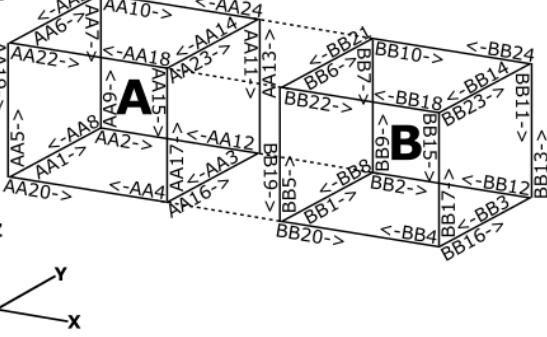
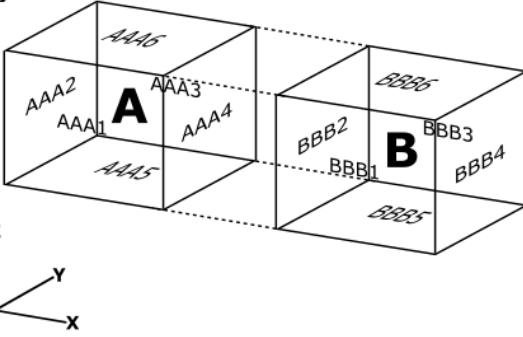
Test 7 Overview	Boundary Points (BP)
<p>a)</p> 	<p>b)</p>  <p style="text-align: center;">NTS</p>
<p>Boundary Lines (BL)</p> <p>c)</p> 	<p>Boundary Planes (BPL)</p> <p>d)</p> 

Figure 6.16: Testing Dataset 7 Visualization

Figure 6.16a shows an overview of simulated 3-D cadastral units A and B used in test 7. While both 3-D cadastral units in this experiment have coplanar boundary planes between them, they run parallel to each other and do not have any overlap relationships between them. Figure 6.16b shows the names and 3-D coordinates of all boundary points, Figure 6.16c shows the names and topological directions of all boundary lines, and Figure 6.16d shows the names of all boundary planes. These figures show that all output relationships will produce one of the 15 disjoint classifications. There is no common geometry between them, and only projected relationships exist.

Tables 6.27 – 6.29 below show the object input storage parameters for the 3-D cadastral units A and B used to create experimental testing dataset 7. Table 6.27 shows the boundary point storage parameters, Table 6.28 shows the boundary line storage parameters, and Table 6.29 shows the boundary plane storage

parameters used to define the boundaries of 3-D cadastral units A and B. These parameters were generated using the processes described in Section 5.1.

Table 6.27: Testing Dataset 7 - 3-D Boundary Point Coordinates

A Boundary Points						B Boundary Points							
Point ID	Point Name	X (e1)	Y (e2)	Z (e3)	no	ni	Point ID	Point Name	X (e1)	Y (e2)	Z (e3)	no	ni
1	A1	2	2	3	1	8.5	1	B1	4	2	3	1	14.5
2	A2	3	2	3	1	11	2	B2	5	2	3	1	19
3	A3	3	3	3	1	13.5	3	B3	5	3	3	1	21.5
4	A4	2	3	3	1	11	4	B4	4	3	3	1	17
5	A5	2	2	4	1	12	5	B5	4	2	4	1	18
6	A6	3	2	4	1	14.5	6	B6	5	2	4	1	22.5
7	A7	3	3	4	1	17	7	B7	5	3	4	1	25
8	A8	2	3	4	1	14.5	8	B8	4	3	4	1	20.5

Table 6.28: Testing Dataset 7 - 3-D Boundary Line Storage Parameters

A Boundary Lines								B Boundary Lines											
Line ID	Line Name	Start Point ID	End Point ID	CGA 3-Blade Parameters						Line ID	Line Name	Start Point ID	End Point ID	CGA 3-Blade Parameters					
				e1^no^ni	e2^no^ni	e3^no^ni	e1^e2^ni	e1^e3^ni	e2^e3^ni					e1^no^ni	e2^no^ni	e3^no^ni	e1^e2^ni	e1^e3^ni	e2^e3^ni
1	AA1	1	4	0	-1	0	2	0	-3	1	BB1	1	4	0	-1	0	4	0	-3
2	AA2	4	3	-1	0	0	-3	-3	0	2	BB2	4	3	-1	0	0	-3	-3	0
3	AA3	3	2	0	1	0	-3	0	3	3	BB3	3	2	0	1	0	-5	0	3
4	AA4	2	1	1	0	0	2	3	0	4	BB4	2	1	1	0	0	2	3	0
5	AA5	1	5	0	0	-1	0	2	2	5	BB5	1	5	0	0	-1	0	4	2
6	AA6	5	8	0	-1	0	2	0	-4	6	BB6	5	8	0	-1	0	4	0	-4
7	AA6	8	4	0	0	1	0	-2	-3	7	BB7	8	4	0	0	1	0	-4	-3
8	AA7	4	1	0	1	0	-2	0	3	8	BB8	4	1	0	1	0	-4	0	3
9	AA9	4	8	0	0	-1	0	2	3	9	BB9	4	8	0	0	-1	0	4	3
10	AA10	8	7	-1	0	0	-3	-4	0	10	BB10	8	7	-1	0	0	-3	-4	0
11	AA11	7	3	0	0	1	0	-3	-3	11	BB11	7	3	0	0	1	0	-5	-3
12	AA12	3	4	1	0	0	3	3	0	12	BB12	3	4	1	0	0	3	3	0
13	AA13	3	7	0	0	-1	0	3	3	13	BB13	3	7	0	0	-1	0	5	3
14	AA14	7	6	0	1	0	-3	0	4	14	BB14	7	6	0	1	0	-5	0	4
15	AA15	6	2	0	0	1	0	-3	-2	15	BB15	6	2	0	0	1	0	-5	-2
16	AA16	2	3	0	-1	0	3	0	-3	16	BB16	2	3	0	-1	0	5	0	-3
17	AA17	2	6	0	0	-1	0	3	2	17	BB17	2	6	0	0	-1	0	5	2
18	AA18	6	5	1	0	0	2	4	0	18	BB18	6	5	1	0	0	2	4	0
19	AA19	5	1	0	0	1	0	-2	-2	19	BB19	5	1	0	0	1	0	-4	-2
20	AA20	1	2	-1	0	0	-2	-3	0	20	BB20	1	2	-1	0	0	-2	-3	0
21	AA21	8	5	0	1	0	-2	0	4	21	BB21	8	5	0	1	0	-4	0	4
22	AA22	5	6	-1	0	0	-2	-4	0	22	BB22	5	6	-1	0	0	-2	-4	0
23	AA23	6	7	0	-1	0	3	0	-4	23	BB23	6	7	0	-1	0	5	0	-4
24	AA24	7	8	1	0	0	3	4	0	24	BB24	7	8	1	0	0	3	4	0

Table 6.29: Testing Dataset 7 - 3-D Boundary Plane Storage Parameters

A Boundary Planes					B Boundary Planes								
Plane ID	Plane Name	Polyline ID Numbers	CGA 4-Blade Parameters				Plane ID	Plane Name	Polyline ID Numbers	CGA 4-Blade Parameters			
			e1^e2^no^ni	e1^e3^no^ni	e2^e3^no^ni	e1^e2^e3^ni				e1^e2^no^ni	e1^e3^no^ni	e2^e3^no^ni	e1^e2^e3^ni
1	AAA1	17_18_19_20	0	1	0	-2	1	BBB1	17_18_19_20	0	1	0	-2
2	AAA2	5_6_7_8	0	0	-1	-2	2	BBB2	5_6_7_8	0	0	-1	-4
3	AAA3	9_10_11_12	0	-1	0	3	3	BBB3	9_10_11_12	0	-1	0	3
4	AAA4	13_14_15_16	0	0	1	3	4	BBB4	13_14_15_16	0	0	1	5
5	AAA5	1_2_3_4	-1	0	0	-3	5	BBB5	1_2_3_4	-1	0	0	-3
6	AAA6	21_22_23_24	1	0	0	4	6	BBB6	21_22_23_24	1	0	0	4

There are 15 different types of disjoint relationship classifications between cadastral units A and B in test 7 that are known to exist *a priori* to testing. These were presented in this chapter in Figure 6.2 and are shown again in Figure 6.17 below.

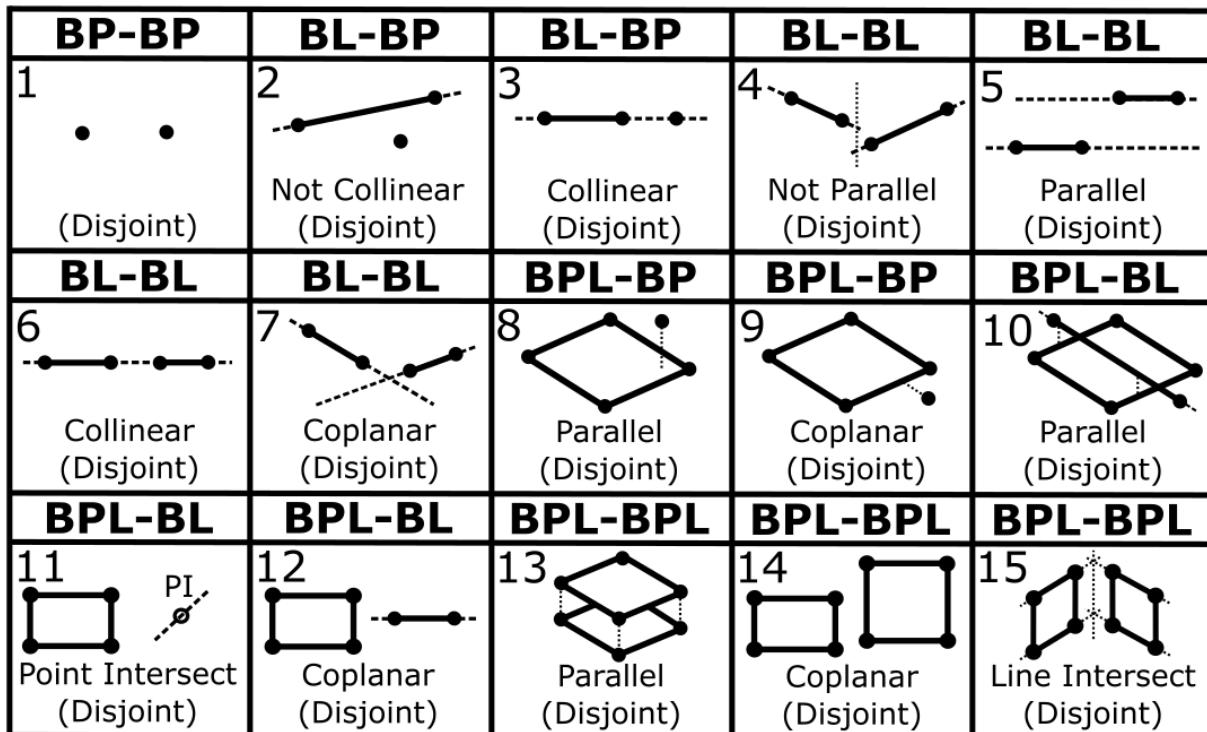


Figure 6.17: Testing Dataset 7 - Existing Disjoint Relationship Classifications

Since there are numerous occurrences of each disjoint relationship type existing in testing dataset 7, only four of each relationship type are included in the program output table to link the relationship classifications to experimental testing dataset 7. Table 6.30 below shows a compiled sub-sample of the main classification algorithm program output for experimental testing dataset 7.

Table 6.30: Testing Dataset 7 – Sample of Program Output

Test 7 - Component Pair Classification Results									
BP-BP					BL-BL				
All Disjoint Relationship Classifications					Component Pair				
BL-BP					Projected Relationship				
					Relationship Description				
					Disjoint Classification				
					Geometry				
BB10-A1	Not Collinear	Disjoint	2	N/A	AA10-BB1	Not Parallel	Disjoint	4	N/A
BB14-A8	Not Collinear	Disjoint	2	N/A	AA18-BB13	Not Parallel	Disjoint	4	N/A
AA15-B1	Not Collinear	Disjoint	2	N/A	AA1-BB24	Not Parallel	Disjoint	4	N/A
AA24-B5	Not Collinear	Disjoint	2	N/A	AA8-BB19	Not Parallel	Disjoint	4	N/A
BB20-A1	Collinear	Disjoint	3	N/A	AA10-BB12	Parallel	Disjoint	5	N/A
BB10-A7	Collinear	Disjoint	3	N/A	AA17-BB19	Parallel	Disjoint	5	N/A
AA4-B1	Collinear	Disjoint	3	N/A	AA24-BB12	Parallel	Disjoint	5	N/A
AA12-B4	Collinear	Disjoint	3	N/A	AA9-BB9	Parallel	Disjoint	5	N/A
BPL-BP					AA10-BB10	Collinear (C)	Disjoint	6	N/A
					AA18-BB18	Collinear (C)	Disjoint	6	N/A
					AA24-BB10	Collinear (R)	Disjoint	6	N/A
					AA4-BB20	Collinear (R)	Disjoint	6	N/A
					AA10-BB11	Coplanar	Disjoint	7	N/A
					AA19-BB18	Coplanar	Disjoint	7	N/A
					AA24-BB9	Coplanar	Disjoint	7	N/A
					AA9-BB2	Coplanar	Disjoint	7	N/A
BPL-BL					BPL-BL				
					Component Pair	Projected Relationship	Relationship Description	Disjoint Classification	Geometry
BBB2-A1	Parallel	Disjoint	8	N/A	BBB1-AA10	Parallel	Disjoint	10	N/A
BBB5-A8	Parallel	Disjoint	8	N/A	BBB6-AA2	Parallel	Disjoint	10	N/A
AAA3-B2	Parallel	Disjoint	8	N/A	AAA2-BB16	Parallel	Disjoint	10	N/A
AAA4-B7	Parallel	Disjoint	8	N/A	AAA4-BB9	Parallel	Disjoint	10	N/A
BBB1-A1	Coplanar	Disjoint	9	N/A	BBB2-AA10	P Intersect	Disjoint	11	N/A
BBB6-A5	Coplanar	Disjoint	9	N/A	BBB4-AA2	P Intersect	Disjoint	11	N/A
AAA5-B2	Coplanar	Disjoint	9	N/A	AAA3-BB16	P Intersect	Disjoint	11	N/A
AAA6-B8	Coplanar	Disjoint	9	N/A	AAA5-BB9	P Intersect	Disjoint	11	N/A
BPL-BPL					BBB3-AA10	Coplanar	Disjoint	12	N/A
					BBB5-AA2	Coplanar	Disjoint	12	N/A
					AAA1-BB17	Coplanar	Disjoint	12	N/A
					AAA3-BB9	Coplanar	Disjoint	12	N/A

Table 6.30 is organized into disjoint relationships between the six different component pair sets. Each component pair producing a disjoint relationship classification was manually verified through visual inspection to be one of the *a priori* relationships shown in Figure 6.17 above by comparing the main classification algorithm program output from Table 6.30 with the experimental test 7 scenario visualizations shown in Figure 6.16.

6.3 Cadastral Condominium Experimental Test

A final experiment using two 3-D cadastral units derived from a registered condominium survey plan in Alberta, Canada is presented in this section to show how the methods developed and tested here can be applied to a practical 3-D cadastral scenario. Figure 6.18 below shows the survey site plan outlining the footprint of the building used in relation to the 2-D parcel boundary. This drawing is mainly used to confirm that the building is situated properly within the 2-D parcel that it is registered to. Figure 6.18 shows that the footprint of the condominium building fits within lots 39 – 42.

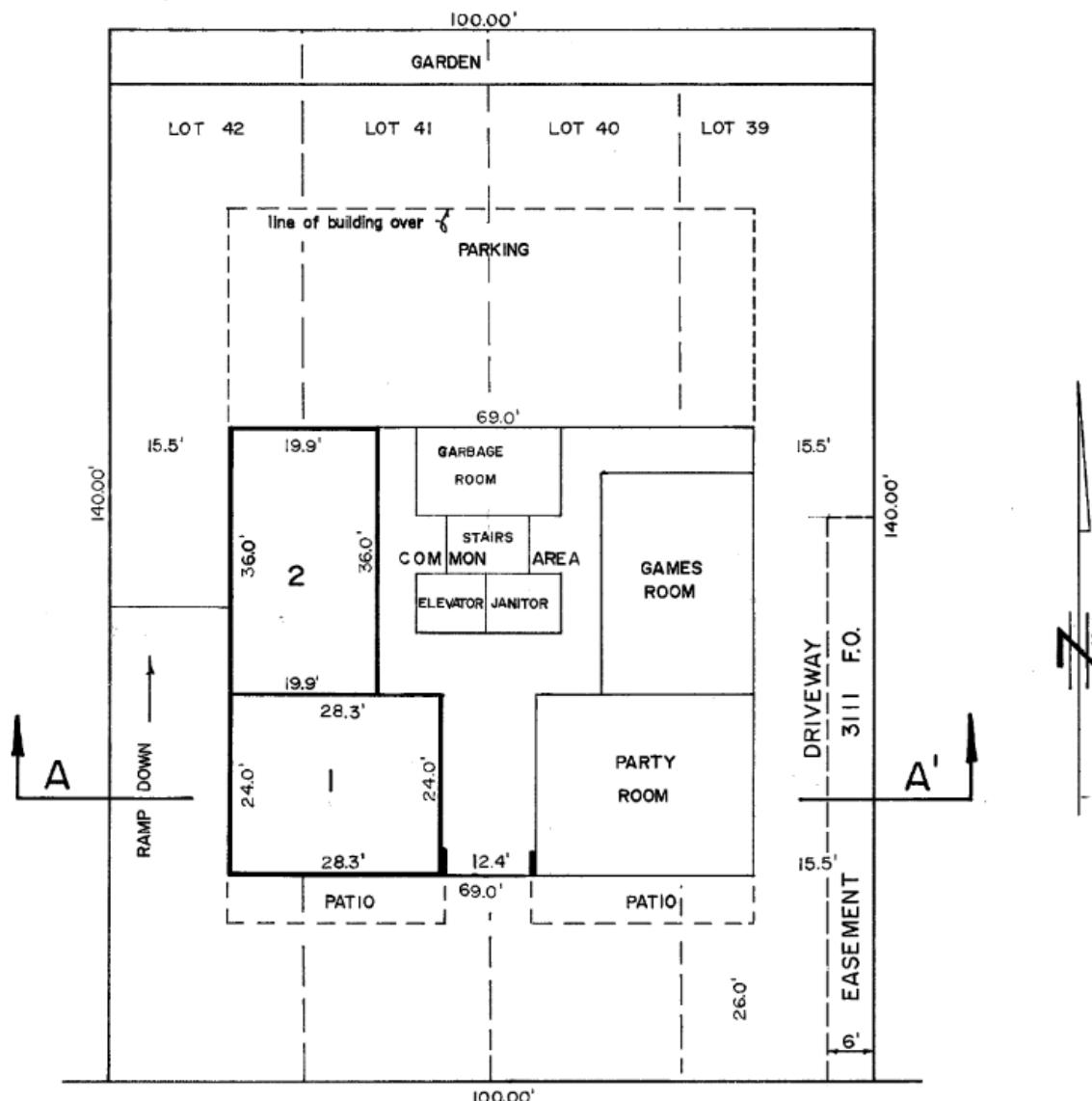


Figure 6.18: Condominium Experiment Survey Site Layout

Figure 6.19 below includes two legal survey plan drawings that were used to calculate 3-D boundary coordinates for the two cadastral units considered here.

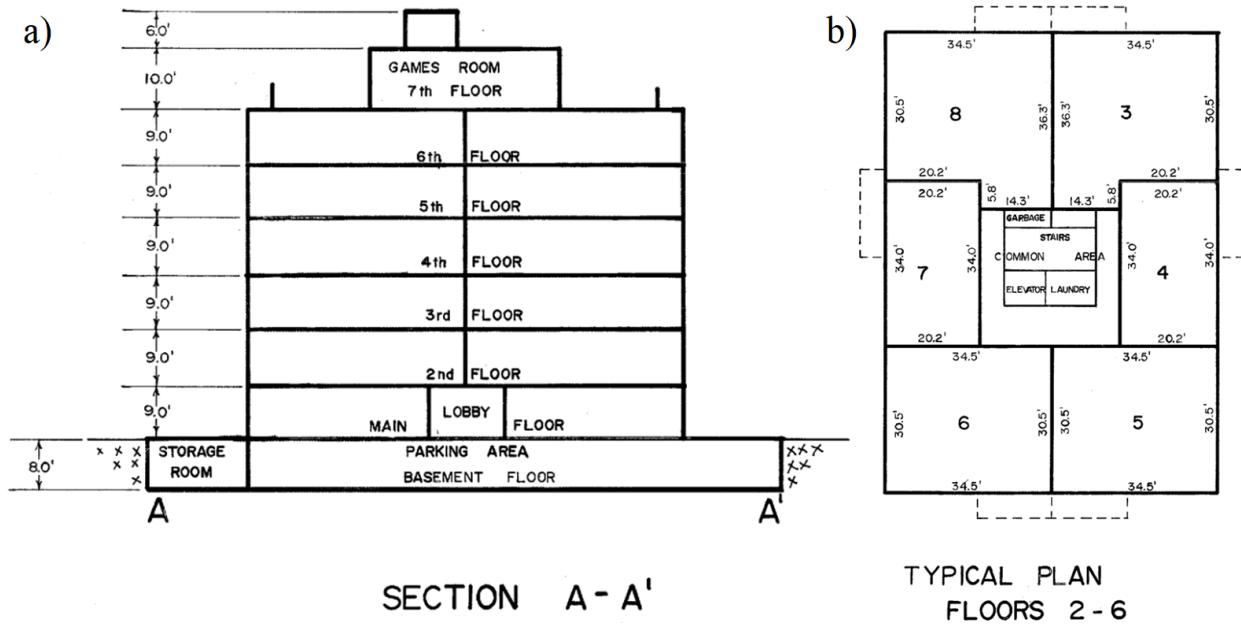


Figure 6.19: Condominium Building Cross Section and Typical Floor Plan

Figure 6.19a shows the side view of section A-A' from the site plan (Figure 6.18). This was used to derive (z) elevations for boundary point coordinates. Figure 6.19b shows the layout for floors 2 – 6 of the condominium building. Measurements on this layout were used to calculate the (x, y) horizontal boundary point coordinates. Adjacent 3-D cadastral units 5 and 6 were chosen for this experiment from the second floor of the condominium building. Relevant horizontal survey plan measurements and derived 3-D boundary point coordinates are shown in Figure 6.20 below.

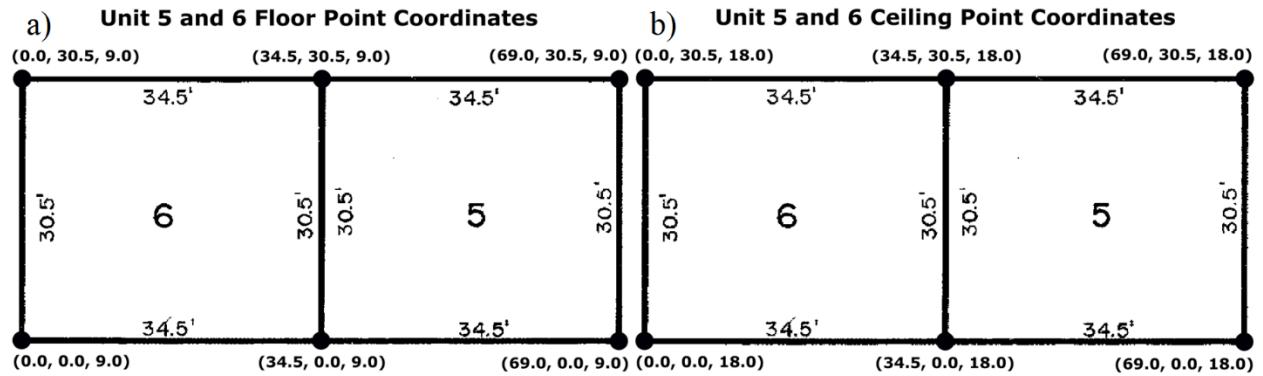


Figure 6.20: Calculated Floor and Ceiling Boundary Point (x, y, z) Coordinates

The first step in generating parameters for the boundary components of these units was to calculate the 3-D boundary points associated with each unit. A local coordinate system was used here, where (0,0,0) was the x, y, z coordinate of the southwest corner of the 1st floor of the building. Figure 6.20 shows how the boundary point coordinates associated with the floor and ceiling of each 3-D cadastral unit were calculated using measurements included in the survey plan. Point heights were calculated using the elevation measurements seen in Figure 6.19a above.

Boundary lines and planes were generated like the previous experiments run through the code so that plane lines had clockwise rotation when looking at each plane from the inside of each unit. Unit 6 was run through the code as 3-D cadastral object A and unit 5 was ran through the code as 3-D cadastral object B. Visualizations for the boundary points, lines, and planes of each unit can be seen in Figure 6.21 below.

Condo Test Overview		Boundary Points (BP)	
a)	NTS	b)	NTS
Boundary Lines (BL)		Boundary Planes (BPL)	
c)	NTS	d)	NTS

Figure 6.21: Condominium Testing Dataset Visualization

The visualization drawings in Figure 6.21 are not drawn to scale. Figure 6.21a shows an overview of 3-D cadastral units A and B used in the condominium experimental test. The main overlap between the two 3-D cadastral units in this experiment can be defined by the full plane touch existing between them. A boundary plane of Unit A (AAA2) fully touches a boundary plane of unit B (BBB4) at a plane segment. There are no intersecting boundary planes in this experiment. Figure 6.21b shows the names and 3-D coordinates of all boundary points, Figure 6.21c shows the names and topological directions of all boundary lines, and Figure 6.21d shows the names of all boundary planes.

Tables 6.31 – 6.33 below show the object input storage parameters for the 3-D cadastral units A and B used to create the condominium experimental testing dataset. Table 6.31 shows the boundary point storage parameters, Table 6.32 shows the boundary line storage parameters, and Table 6.33 shows the boundary

plane storage parameters used to define the boundaries of 3-D cadastral units 6 (A) and 5 (B). These parameters were generated using the processes described in Section 5.1.

Table 6.31: Condominium Testing Dataset - 3-D Boundary Point Coordinates

Unit 6 (A) Boundary Points							Unit 5 (B) Boundary Points						
Point ID	Point Name	X (e1)	Y (e2)	Z (e3)	no	ni	Point ID	Point Name	X (e1)	Y (e2)	Z (e3)	no	ni
1	A_1	34.5	30.5	9	1	1100.75	1	B_1	69	30.5	9	1	2886.125
2	A_2	34.5	0	9	1	635.625	2	B_2	69	0	9	1	2421
3	A_3	0	0	9	1	40.5	3	B_3	34.5	0	9	1	635.625
4	A_4	0	30.5	9	1	505.625	4	B_4	34.5	30.5	9	1	1100.75
5	A_5	34.5	30.5	18	1	1222.25	5	B_5	69	30.5	18	1	3007.625
6	A_6	34.5	0	18	1	757.125	6	B_6	69	0	18	1	2542.5
7	A_7	0	0	18	1	162	7	B_7	34.5	0	18	1	757.125
8	A_8	0	30.5	18	1	627.125	8	B_8	34.5	30.5	18	1	1222.25

Table 6.32: Condominium Testing Dataset - 3-D Boundary Line Storage Parameters

Unit 6 (A) Boundary Lines							Unit 5 (B) Boundary Lines												
Line ID	Line Name	Start Point ID	End Point ID	CGA 3-Blade Parameters						Start Point ID	End Point ID	CGA 3-Blade Parameters							
				e1^no^ni	e2^no^ni	e3^no^ni	e1^e2^ni	e1^e3^ni	e2^e3^ni			e1^no^ni	e2^no^ni	e3^no^ni	e1^e2^ni	e1^e3^ni	e2^e3^ni		
1	AA_1	1	2	0	30.5	0	-1052.25	0	274.5	1	BB_1	1	2	0	30.5	0	-2104.5	0	274.5
2	AA_2	2	3	34.5	0	0	0	310.5	0	2	BB_2	2	3	34.5	0	0	0	310.5	0
3	AA_3	3	4	0	-30.5	0	0	0	-274.5	3	BB_3	3	4	0	-30.5	0	1052.25	0	-274.5
4	AA_4	4	1	-34.5	0	0	-1052.25	-310.5	0	4	BB_4	4	1	-34.5	0	0	-1052.25	-310.5	0
5	AA_5	1	5	0	0	-9	0	310.5	274.5	5	BB_5	1	5	0	0	-9	0	621	274.5
6	AA_6	5	6	0	30.5	0	-1052.25	0	549	6	BB_6	5	6	0	30.5	0	-2104.5	0	549
7	AA_7	6	2	0	0	9	0	-310.5	0	7	BB_7	6	2	0	0	9	0	-621	0
8	AA_8	2	1	0	-30.5	0	1052.25	0	-274.5	8	BB_8	2	1	0	-30.5	0	2104.5	0	-274.5
9	AA_9	2	6	0	0	-9	0	310.5	0	9	BB_9	2	6	0	0	-9	0	621	0
10	AA_10	6	7	34.5	0	0	0	621	0	10	BB_10	6	7	34.5	0	0	0	621	0
11	AA_11	7	3	0	0	9	0	0	0	11	BB_11	7	3	0	0	9	0	-310.5	0
12	AA_12	3	2	-34.5	0	0	0	-310.5	0	12	BB_12	3	2	-34.5	0	0	0	-310.5	0
13	AA_13	3	7	0	0	-9	0	0	0	13	BB_13	3	7	0	0	-9	0	310.5	0
14	AA_14	7	8	0	-30.5	0	0	0	-549	14	BB_14	7	8	0	-30.5	0	1052.25	0	-549
15	AA_15	8	4	0	0	9	0	0	-274.5	15	BB_15	8	4	0	0	9	0	-310.5	-274.5
16	AA_16	4	3	0	30.5	0	0	0	274.5	16	BB_16	4	3	0	30.5	0	-1052.25	0	274.5
17	AA_17	4	8	0	0	-9	0	0	274.5	17	BB_17	4	8	0	0	-9	0	310.5	274.5
18	AA_18	8	5	-34.5	0	0	-1052.25	-621	0	18	BB_18	8	5	-34.5	0	0	-1052.25	-621	0
19	AA_19	5	1	0	0	9	0	-310.5	-274.5	19	BB_19	5	1	0	0	9	0	-621	-274.5
20	AA_20	1	4	34.5	0	0	1052.25	310.5	0	20	BB_20	1	4	34.5	0	0	1052.25	310.5	0
21	AA_21	8	7	0	30.5	0	0	0	549	21	BB_21	8	7	0	30.5	0	-1052.25	0	549
22	AA_22	7	6	-34.5	0	0	0	-621	0	22	BB_22	7	6	-34.5	0	0	0	-621	0
23	AA_23	6	5	0	-30.5	0	1052.25	0	-549	23	BB_23	6	5	0	-30.5	0	2104.5	0	-549
24	AA_24	5	8	34.5	0	0	1052.25	621	0	24	BB_24	5	8	34.5	0	0	1052.25	621	0

Table 6.33: Condominium Testing Dataset - 3-D Boundary Plane Storage Parameters

Unit 6 (A) Boundary Planes						Unit 5 (B) Boundary Planes							
Plane ID	Plane Name	Polyline ID Numbers	CGA 4-Blade Parameters				Plane ID	Plane Name	Polyline ID Numbers	CGA 4-Blade Parameters			
			e1^e2^no^ni	e1^e3^no^ni	e2^e3^no^ni	e1^e2^e3^ni				e1^e2^no^ni	e1^e3^no^ni	e2^e3^no^ni	e1^e2^e3^ni
1	AAA_1	1_2_3_4	-1052.25	0	0	-9470.25	1	BBB_1	1_2_3_4	-1052.25	0	0	-9470.25
2	AAA_2	5_6_7_8	0	0	274.5	9470.25	2	BBB_2	5_6_7_8	0	0	274.5	18940.5
3	AAA_3	9_10_11_12	0	310.5	0	0.00	3	BBB_3	9_10_11_12	0	310.5	0	0
4	AAA_4	13_14_15_16	0	0	-274.5	0.00	4	BBB_4	13_14_15_16	0	0	-274.5	-9470.25
5	AAA_5	17_18_19_20	0	-310.5	0	9470.25	5	BBB_5	17_18_19_20	0	-310.5	0	9470.25
6	AAA_6	21_22_23_24	1052.25	0	0	18940.50	6	BBB_6	21_22_23_24	1052.25	0	0	18940.5

There are 13 different types of overlap relationship classifications between 3-D cadastral units A and B in this condominium experimental test that are known to exist *a priori* to testing. These can be seen in Figure 6.22 below.

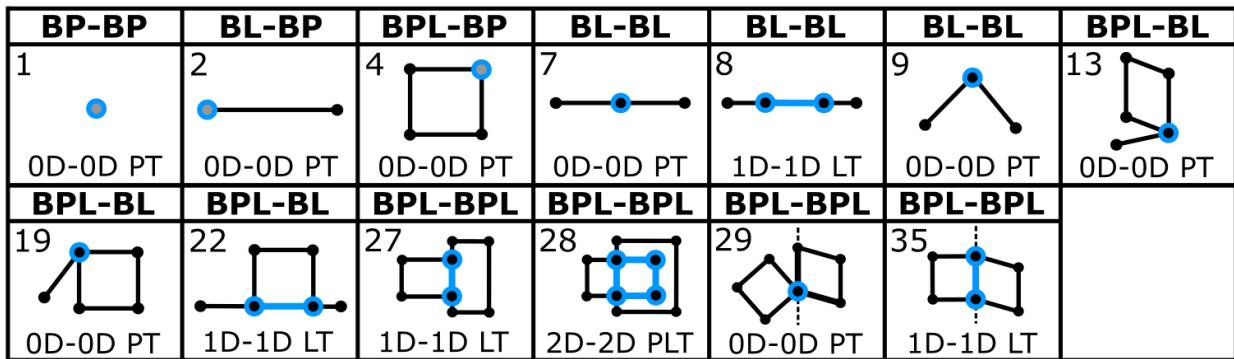


Figure 6.22: Condominium Testing Dataset - Existing Overlap Relationship Classifications

This condominium experiment results in numerous occurrences of each of the relationship classifications shown in Figure 6.22 above. Table 6.34 below shows a compiled sample of the program output for the condominium experimental testing dataset. It is organized into relationships between the six different component pair sets. Each component pair producing an overlap relationship classification was manually verified through visual inspection to be one of the *a priori* relationships shown in Figure 6.22 above. This was done by comparing the main classification algorithm program output from Table 6.34 with the condominium experimental test scenario visualizations shown in Figure 6.21.

Table 6.34: Condominium Testing Dataset – Sample of Program Output

Condominium Test - Component Pair Overlap Classification Results															
BP-BP					BL-BL										
Component Pair	Projected Relationship	Relationship Description	Overlap Classification	Geometry	Component Pair	Projected Relationship	Relationship Description	Overlap Classification	Geometry						
A1-B4	N/A	OD-OD PT	1	[34.5,30.5,9]	AA10-BB10	Collinear (C)	OD-OD PT	7	[34.5,0,18]						
A2-B3	N/A	OD-OD PT	1	[34.5,0,9]	AA10-BB11	Coplanar	OD-OD PT	9	[34.5,0,18]						
A5-B8	N/A	OD-OD PT	1	[34.5,30.5,18]	AA12-BB11	Coplanar	OD-OD PT	9	[34.5,0,9]						
A6-B7	N/A	OD-OD PT	1	[34.5,0,18]	AA12-BB12	Collinear (C)	OD-OD PT	7	[34.5,0,9]						
BL-BP					AA18-BB14	Coplanar	OD-OD PT	9	[34.5,30.5,18]						
Component Pair	Projected Relationship	Relationship Description	Overlap Classification	Geometry	AA18-BB18	Collinear (C)	OD-OD PT	7	[34.5,30.5,18]						
					AA19-BB15	Collinear (C)	1D-1D LT	8	[34.5,30.5,18] [34.5,30.5,9]						
					AA19-BB16	Coplanar	OD-OD PT	9	[34.5,30.5,9]						
					AA19-BB16	Collinear (C)	1D-1D LT	8	[34.5,30.5,9] [34.5,0,9]						
BPL-BP					AA20-BB20	Collinear (C)	OD-OD PT	7	[34.5,30.5,9]						
Component Pair	Projected Relationship	Relationship Description	Overlap Classification	Geometry	AA23-BB14	Collinear (C)	1D-1D LT	8	[34.5,0,18] [34.5,30.5,18]						
					AA5-BB15	Collinear (R)	1D-1D LT	8	[34.5,30.5,9] [34.5,30.5,18]						
					AA6-BB14	Collinear (R)	1D-1D LT	8	[34.5,30.5,18] [34.5,0,18]						
					AA7-BB11	Collinear (C)	1D-1D LT	8	[34.5,0,18] [34.5,0,9]						
BPL-BL					AA8-BB16	Collinear (R)	1D-1D LT	8	[34.5,0,9] [34.5,30.5,9]						
Component Pair	Projected Relationship	Relationship Description	Overlap Classification	Geometry	AA9-BB11	Collinear (R)	1D-1D LT	8	[34.5,0,9] [34.5,0,18]						
					BPL-BPL										
					Component Pair	Projected Relationship	Relationship Description	Overlap Classification	Geometry						
						OD-OD PT	—	—	[34.5,0,9]						
BP-BP						AAA1-BBB1	Coplanar	1D-1D LT	27	[34.5,0,9] [34.5,30.5,9]					
BL-BL						OD-OD PT	—	—	[34.5,30.5,9]						
BPL-BP						AAA1-BBB3	L Intersect	OD-OD PT	29	[34.5,0,9]					
BPL-BL						AAA1-BBB4	L Intersect	OD-OD PT	—	[34.5,30.5,9]					
BPL-BPL						OD-OD PT	—	—	[34.5,0,9]						
BP-BP						AAA1-BBB4	1D-1D LT	35	[34.5,30.5,9] [34.5,0,9]	[34.5,30.5,9]					
BL-BL						OD-OD PT	—	—	[34.5,0,9]						
BPL-BP						OD-OD PT	—	—	[34.5,0,9]						
BPL-BL						OD-OD PT	—	—	[34.5,0,18]						
BPL-BPL						OD-OD PT	—	—	[34.5,0,18]						
BP-BP						OD-OD PT	—	—	[34.5,0,18]						
BL-BL						OD-OD PT	—	—	[34.5,0,18]						
BPL-BP						OD-OD PT	—	—	[34.5,0,18]						
BPL-BL						OD-OD PT	—	—	[34.5,0,18]						
BPL-BPL						OD-OD PT	—	—	[34.5,0,18]						
BP-BP						OD-OD PT	—	—	[34.5,0,18]						
BL-BL						OD-OD PT	—	—	[34.5,0,18]						
BPL-BP						OD-OD PT	—	—	[34.5,0,18]						
BPL-BL						OD-OD PT	—	—	[34.5,0,18]						
BPL-BPL						OD-OD PT	—	—	[34.5,0,18]						
BP-BP						OD-OD PT	—	—	[34.5,0,18]						
BL-BL						OD-OD PT	—	—	[34.5,0,18]						
BPL-BP						OD-OD PT	—	—	[34.5,0,18]						
BPL-BL						OD-OD PT	—	—	[34.5,0,18]						
BPL-BPL						OD-OD PT	—	—	[34.5,0,18]						
BP-BP						OD-OD PT	—	—	[34.5,0,18]						
BL-BL						OD-OD PT	—	—	[34.5,0,18]						
BPL-BP						OD-OD PT	—	—	[34.5,0,18]						
BPL-BL						OD-OD PT	—	—	[34.5,0,18]						
BPL-BPL						OD-OD PT	—	—	[34.5,0,18]						
BP-BP						OD-OD PT	—	—	[34.5,0,18]						
BL-BL						OD-OD PT	—	—	[34.5,0,18]						
BPL-BP						OD-OD PT	—	—	[34.5,0,18]						
BPL-BL						OD-OD PT	—	—	[34.5,0,18]						
BPL-BPL						OD-OD PT	—	—	[34.5,0,18]						
BP-BP						OD-OD PT	—	—	[34.5,0,18]						
BL-BL						OD-OD PT	—	—	[34.5,0,18]						
BPL-BP						OD-OD PT	—	—	[34.5,0,18]						
BPL-BL						OD-OD PT	—	—	[34.5,0,18]						
BPL-BPL						OD-OD PT	—	—	[34.5,0,18]						
BP-BP						OD-OD PT	—	—	[34.5,0,18]						
BL-BL						OD-OD PT	—	—	[34.5,0,18]						
BPL-BP						OD-OD PT	—	—	[34.5,0,18]						
BPL-BL						OD-OD PT	—	—	[34.5,0,18]						
BPL-BPL						OD-OD PT	—	—	[34.5,0,18]						
BP-BP						OD-OD PT	—	—	[34.5,0,18]						
BL-BL						OD-OD PT	—	—	[34.5,0,18]						
BPL-BP						OD-OD PT	—	—	[34.5,0,18]						
BPL-BL						OD-OD PT	—	—	[34.5,0,18]						
BPL-BPL						OD-OD PT	—	—	[34.5,0,18]						
BP-BP						OD-OD PT	—	—	[34.5,0,18]						
BL-BL						OD-OD PT	—	—	[34.5,0,18]						
BPL-BP						OD-OD PT	—	—	[34.5,0,18]						
BPL-BL						OD-OD PT	—	—	[34.5,0,18]						
BPL-BPL						OD-OD PT	—	—	[34.5,0,18]						
BP-BP						OD-OD PT	—	—	[34.5,0,18]						
BL-BL						OD-OD PT	—	—	[34.5,0,18]						
BPL-BP						OD-OD PT	—	—	[34.5,0,18]						
BPL-BL						OD-OD PT	—	—	[34.5,0,18]						
BPL-BPL						OD-OD PT	—	—	[34.5,0,18]						
BP-BP						OD-OD PT	—	—	[34.5,0,18]						
BL-BL						OD-OD PT	—	—	[34.5,0,18]						
BPL-BP						OD-OD PT	—	—	[34.5,0,18]						
BPL-BL						OD-OD PT	—	—	[34.5,0,18]						
BPL-BPL						OD-OD PT	—	—	[34.5,0,18]						
BP-BP						OD-OD PT	—	—	[34.5,0,18]						
BL-BL						OD-OD PT	—	—	[34.5,0,18]						
BPL-BP						OD-OD PT	—	—	[34.5,0,18]						
BPL-BL						OD-OD PT	—	—	[34.5,0,18]						
BPL-BPL						OD-OD PT	—	—	[34.5,0,18]						
BP-BP						OD-OD PT	—	—	[34.5,0,18]						
BL-BL						OD-OD PT	—	—	[34.5,0,18]						
BPL-BP						OD-OD PT	—	—	[34.5,0,18]						
BPL-BL						OD-OD PT	—	—	[34.5,0,18]						
BPL-BPL						OD-OD PT	—	—	[34.5,0,18]						
BP-BP						OD-OD PT	—	—	[34.5,0,18]						
BL-BL						OD-OD PT	—	—	[34.5,0,18]						
BPL-BP						OD-OD PT	—	—	[34.5,0,18]						
BPL-BL						OD-OD PT	—	—	[34.5,0,18]						
BPL-BPL						OD-OD PT	—	—	[34.5,0,18]						
BP-BP						OD-OD PT	—	—	[34.5,0,18]						
BL-BL						OD-OD PT	—	—	[34.5,0,18]						
BPL-BP						OD-OD PT	—	—	[34.5,0,18]						
BPL-BL						OD-OD PT	—	—	[34.5,0,18]						
BPL-BPL						OD-OD PT	—	—	[34.5,0,18]						
BP-BP						OD-OD PT	—	—	[34.5,0,18]						
BL-BL						OD-OD PT	—	—	[34.5,0,18]						
BPL-BP						OD-OD PT	—	—	[34.5,0,18]</td						

The main classification algorithm program output a total of 16 BL-BL sets that produced the ‘Collinear 0D-0D Point Touch’ relationship (overlap classification 7), 16 BL-BL sets that produced the ‘Collinear 1D-1D Line Touch’ relationship (overlap classification 8), and 96 BL-BL sets that produced the ‘Coplanar 0D-0D Point Touch’ relationship (overlap classification 9). A sample of four, eight, and four of these relationships are presented, respectively.

The main classification algorithm program output a total of 48 BPL-BL sets that produced the ‘Point Intersect 0D-0D Point Touch’ relationship (overlap classification 13), 32 BPL-BL sets that produced the ‘Coplanar 0D-0D Point Touch’ relationship (overlap classification 19), and 32 BPL-BL sets that produced the ‘Coplanar 1D-1D Line Touch’ relationship (overlap classification 22). A sample of four of each of these relationships is presented.

The main classification algorithm program output a total of four BPL-BPL sets that produced the ‘Coplanar 1D-1D Line Touch’ relationship (overlap classification 27) and one BPL-BPL set that produced the ‘2D-2D Plane Touch’ relationship (overlap classification 28) consisting of four ‘Coplanar 1D-1D Line Touch’ segments. These are all presented in the table. The program output a total of eight BPL-BPL sets that produced the ‘Line Intersect 0D-0D Point Touch’ relationship (overlap classification 29) and eight BPL-BPL sets that produced the ‘Line Intersect 1D-1D Line Touch’ relationship (overlap classification 35). A sample of two of each of these relationships is presented.

The relationship classification results from this experimental test that was derived from a 3-D condominium survey in Alberta, Canada show how the methods developed in this study can be applied to a real cadastral scenario. These methods can be applied for validating defined legal boundaries before a survey plan is registered, or during the registration process.

6.4 Chapter Summary

The relevance this chapter has to the primary research objective is that it tests and validates the modelling processes developed in this study using experimental datasets, each consisting of two 3-D cadastral units. This chapter describes the results that were generated from performing research activity #8: “Test proposed computational procedures with simulated datasets to show the extent that they can work under” and research activity #9: “Test proposed computation procedures on two 3-D cadastral units derived from a real 3-D cadastral survey to show how the developed procedures can be applied to a practical 3-D boundary problem scenario”.

Section 6.1 presented a summary of the relationships that can be classified using the methods developed in Chapter 4 and Chapter 5 of this study. There is a total of 53 distinct relationships that can be classified between the six types of boundary component pairs. 38 of these classifications are non-disjoint (touch and intersect) relationships and 15 of these classifications are disjoint relationships. Two relationships can be classified between point-point boundary component pairs, four relationships can be classified between line-point boundary component pairs, five relationships can be classified between plane-point boundary component pairs, 10 relationships can be classified between line-line boundary component pairs, 14 relationships can be classified between plane-line boundary component pairs, and 18 relationships can be classified between plane-plane boundary component pairs. In total, this chapter presented 8 sets of experiments that were used to validate the data modelling processes developed in this study. All datasets consisted of two cube-like 3-D cadastral units that were each represented by their point, line, and plane boundary components. The relationships between all sets of 3-D cadastral units were known *a priori* to running the experiments and the classification results were validated manually to check if the program was producing the correct results.

Section 6.2 presented the first seven experiments consisting of simulated datasets that were created specifically so that each of the 53 distinct relationships could be classified at least once to show that the model could properly classify them. Simulated datasets 1-6 were used to show how the developed classification algorithms could be effective in properly classifying the 38 distinct ‘Touch’ and ‘Intersection/Overlap’ relationships while simulated dataset 7 was used to show how the developed classification algorithms could be effective in properly classifying the 15 distinct ‘Disjoint’ relationships.

Section 6.3 presented experimental results from a dataset consisting of two adjacent 3-D cadastral units that were derived from a centerline boundary condominium survey registered in Alberta, Canada. This experiment was included to show how the developed classification algorithms could be applied to real 3-D boundary data for a practical example in the surveying field. Two adjacent units were chosen from the second floor of the condominium building and boundary point coordinates were derived using a local coordinate system to the condominium building. The main implementation program output the correct relationship classifications for each of the six boundary component pair sets and was able to identify the shared wall existing between the two units.

All relationships for the experiments chosen were correctly identified and each of the 53 relationship classification types that were described in Chapter 4 and summarized in Figure 6.1 and Figure 6.2 above was classified at least once throughout the experimental tests. The results presented in this chapter suggest that classification processes that apply CGA algorithms can be used effectively in classifying relationships between the boundary components associated with 3-D cadastral units.

Chapter Seven: Discussion and Conclusions

This chapter summarizes the work that was performed, concludes on how the work completed here contributes to knowledge associated with 3-D cadastre development and the professional surveying and land administration fields, and analyzes how the work done in this study compares to previous research and discusses how it can be expanded upon further in future research. It is presented in four sections.

Section 7.1 briefly summarizes the work that was completed in each chapter of this study while linking it to different aspects of the primary research objective. Section 7.2 presents the conclusions that were deduced through completing this study. It discusses how this study contributes to knowledge, both theoretically and practically in applications related to 3-D land management and the professional land surveying profession. Section 7.3 presents a discussion on how the methods developed and applied in this study relate and compare to the existing theory and relationship modelling approaches that were presented in Chapter 2 and Chapter 3. Section 7.4 discusses some limitations of the methods developed here and provides recommendations for future work related to aspects of this study that can be expanded upon and explored further.

7.1 Research Summary

The cadastral literature search performed in Chapter 2 defined important concepts related to different research topics of 3-D cadastre development and was used to create a definition for 3-D cadastral boundaries and their boundary components. The 3-D cadastral units modelled in this study were derived from closed boundary volumetric 3-D units that could be registered through strata or condominium survey plans in Alberta, Canada. The data storage structure defines 3-D boundaries as the collection of their lower-dimensional 3-D point, line, and plane boundary components.

The mathematical literature search performed in Chapter 3 identified existing frameworks for defining and organizing relationships between spatial objects and discussed a few approaches to distinguish these relationships in practice. The relationships modelled in this study between 3-D boundary component pairs were limited to boundary components touching, intersecting, or being disjoint from each other. Formal theory regarding Conformal Geometric Algebra was included in this chapter along with the equations that were used in this study to implement topological checks between 3-D boundary components.

Data flow processing algorithms were developed and presented in Chapter 4 that can be used to classify various relationships between the 3-D boundary components considered in this study. The data formats that were used to define and digitally model 3-D cadastral boundaries were presented in this chapter as well.

The overall relationship classification approach that involved classifying the relationships between six sets of 3-D boundary component pairs were presented in this chapter and described in detail.

The approach that was followed to implement these classification algorithms using written MATLAB functions in combination with GAViewer files was described in Chapter 5. Methods for creating the 3-D cadastral boundary datasets were described, along with the methods that were applied to implement the main and secondary relationship classification functions that were presented in Chapter 4.

Results from the datasets in Chapter 6 showed that by applying the proposed processes for each of the six sets of boundary component pairs, a total of 53 distinctive relationships could be classified properly between the boundaries of two 3-D cadastral units. Thirty-eight of these were associated with overlap (touch or intersect) relationships while 15 were associated with disjoint relationships. The experiments showed that the methodological processes developed for the six types of boundary component pairs that apply a combination of CGA operations and 3-D point-point distance evaluations can be used to correctly evaluate geometric and topological relationships between six sets of boundary component pairs.

7.2 Conclusions and Contribution to Knowledge

The study developed and tested computational data modelling processes that applied CGA theory, objects, and operations, as well as 3-D point-point distance evaluations, to classify geometrical and topological relationships that exist between the (point, line, and plane) components defining the boundaries of two 3-D cadastral units. Data flow classification algorithms were developed to correctly classify various types of disjoint, touch, and intersect relationships between six sets of 3-D boundary component pairs. The six classification algorithms were validated by testing them on seven simulated experimental datasets and on an experimental dataset that was derived from a condominium cadastral survey registered in Alberta, Canada.

The initial hypothesis was that CGA theory and operations alone could be applied to classify all the topological relationships between 3-D cadastral boundary components. This was influenced by previous research presented in Section 3.2.2 that showed different ways of how CGA theory could be applied to model 3-D cadastral objects and to analyze relationships between their 3D boundaries. While it may be possible to reach similar results using CGA alone, additional techniques were applied in this study in combination with CGA operations to develop a solution that addressed the primary objective of this study.

The developed solution that was implemented to address the primary objective of this study used a combination of CGA operations and additional 3-D point-point distance evaluations to classify relationships between 3-D boundary components. This is not to say that it isn't possible to reach similar results by using CGA operations only, rather that additional operations were required to reach them using the methodology developed in Chapter 4. CGA allows for many other techniques that could possibly be applied to reach similar results, and these should be explored further with respect to classifying topological relationships between 3D cadastral units and their 3-D boundary components.

The CGA theory and methods as they were applied were used to correctly identify which projected relationship was occurring between 3-D boundary component pairs. Projected relationships could result in two boundary components being parallel, skew, collinear, coplanar, or intersecting at a point or line. Additional operational techniques that mainly calculated and evaluated various 3-D point-point distances were leveraged to both distinguish which of the final relationship classifications were occurring for each of the projected relationship scenarios and to gather geometric point coordinate sets which described the extent of any relationship classification that had some form of overlap between the two boundary components being considered.

This study has both theoretical and practical contributions to research associated with 3-D cadastres and 3-D land management. The theoretical contribution is that data flow processes and algorithms were developed and tested to classify 53 distinct relationship types between six sets of 3-D boundary component pairs. Simulated experimental datasets were set up so that each of the 53 relationships could be classified correctly at least once to validate the methodological theory that was developed and implemented to classify it. The experimental results from the datasets that were tested support how the methods that were developed using CGA along with 3-D point-point distance evaluations can be applied to correctly classify topological relationships between 3-D cadastral boundaries.

The practical contribution is that it showed how the methods developed here can be applied to solving a practical 3-D cadastral boundary problem example in the land surveying field, specifically towards validating a shared boundary between two adjacent 3-D cadastral units as intended on the survey plan prior to registering it. An experiment using a real cadastral example was derived using measurements included in a 3-D condominium survey plan registered in Alberta, Canada to validate the shared plane boundary between two adjacent 3-D cadastral units. While this type of relationship analysis can be done through visual inspection of survey plans, the methods developed here are more mathematically rigorous.

None of the experimental results from the datasets tested show the methods developed here to be false (falsification). Experimental results still needed to be validated through visual inspection as an independent method to verify that the results were correct. The methods developed here were not verified as being universally correct with all 3-D cadastral units, but rather have been validated with the simple cube-like cadastral experiments chosen.

7.3 Discussion on Methodological Approach

The primary objective of this study addressed how 3-D boundaries that are registered using survey plans could be digitally represented in a spatial DBMS using 3-D boundary point, line, and plane storage tables. The primary objective also addressed how various topological relationships could be organized and classified between the boundaries of two 3-D cadastral units.

This section discusses the data format that was chosen to represent 3-D boundaries. It discusses how the relationships classified here compare with that of the 9IM in 3-D space and discusses the limitations and advantages of the approach followed. Lastly, it discusses the advantage of outputting a geometrical description associated with each overlap relationship classified.

7.3.1 Discussion on Data Format Chosen

Even though the legal foundations and property definitions are not uniform among different cadastral jurisdictions, the processes developed and tested in this study that were used to organize and classify relationships between 3-D boundary components could be applied to 3-D cadastral data models that are being developed in jurisdictions that use a similar data format.

The data format used here to represent the boundary of a 3-D cadastral unit like that of Ying et al. (2015) was chosen to best reflect how cadastral boundaries can be registered using a strata or condominium survey plan in Alberta, Canada. This format could apply to other jurisdictions that use boundary points, lines, and planes to register 3-D cadastral boundaries as well.

While other data formats such as LiDAR point clouds have been considered to record 3-D cadastral boundaries in other studies, there are often challenges associated with them such as uncertainties with data quality and missing data (Koeva & Oude Elberink, 2016). The advantage of the point, line, plane format used here is that the parameters can be derived through relatively simple calculations using recorded distances and angles that are included in a survey plan. The experimental test involving two 3-D cadastral

units derived from a condominium survey plan (see Section 6.3) showed how real cadastral datasets could be digitized. This was done by first calculating 3-D point coordinates for all vertex points in the plan, followed by creating a topological structure for lines using two end point references and planes using at least three boundary line references.

Recalling the three different ways that 3-D property can be represented with 2-D data in a cadastral system (see Section 2.5.1), this study assumed a mixed combination of both 2-D and 3-D parcels in a cadastral database. The limitation of this approach is that 3-D parcels cannot be compared directly with adjacent 2-D parcels. The advantage of this approach is that 2-D parcels do not need to be converted into 3-D parcels. Condominium surveys in Alberta, Canada must include a drawing that shows how any condominium buildings included on the plan are within the limits of the 2-D parcel that they are registered to. This can be done using existing 2-D relationship analysis approaches between the parcel boundary and the condominium building footprint. An advantage to using this approach is that 3-D analysis only needs to be performed between 3-D cadastral units that are registered to the same 3-D survey plan and are contained within the same building. This allows for a local building coordinate system to be used which can simplify the boundary coordinate calculation process significantly, as was shown in Section 6.3.

7.3.2 Relationship Classifications Compared to the 9IM in 3-D Space

When comparing the relationships that can be classified using the methods proposed in this study with that of the theoretical models reviewed in Section 3.1.1, there are some key differences worth mentioning. One of these differences is that this approach generalizes relationships between boundary components into pairs that are ‘Disjoint’ from each other, ‘Touching/Meeting’ at one or more of their exterior boundary limits, or ‘Intersecting’ at both of their interior boundary extensions. Other topological classification distinctions associated with overlap such as ‘Inside/Contains’, ‘Covers/Covered by’, and ‘Equal’ were generalized into one of these groupings. These overlap relationships have minor distinctions between them such as the exteriors of objects touching or not and were not considered here.

This generalization was applied here because the expanded classifications were assumed to have limited additional advantages to analyzing relationships between 3-D cadastral boundaries, especially when the geometry of the overlap is returned with the classification. Applying this generalization results in a smaller number of relationships that can be distinguished in some cases using this approach than what is theoretically conceptualized using the 9IM in 3-D space (see Table 3.4). Examples of this are explained using the line-line and plane-line component pair relationship outcomes discussed below.

This approach can classify the same two distinct relationships between point-point boundary component pairs as all other models that were discussed in Section 3.1.1 and Section 3.1.2, being the ‘Disjoint’ and ‘0D-0D Point Touch’ classifications.

This approach can classify four distinct relationships between line-point boundary component pairs, compared to three distinct relationships that are conceptualized using the 9IM framework in 3-D space (see Zlatanova (2000) and Table 3.4). This approach can classify the same three relationships between line-point pairs, being ‘Disjoint’, ‘0D-0D Point Touch’ at a (0D) line end limit point, and ‘1D-0D Point Touch’ at a (1D) line interior extension point. The approach followed here makes an additional distinction for line-point disjoint relationships as a point can be ‘Disjoint’ and collinear or ‘Disjoint’ and not collinear.

This approach can classify 10 distinct relationships between line-line boundary component pairs (13 when distinguishing between common or reverse collinear), compared to 33 distinct relationships that are conceptualized using the 9IM framework in 3-D space (see Zlatanova (2000) and Table 3.4). This lower number is partially due to the generalization of topological descriptors described above that classifies the ‘Inside/Contains’, ‘Covers/Covered by’, and ‘Equal’ topological classifications as the same touch relationship. Another reason for this lower number of relationships is that Zlatanova (2000) could be considering relationships between lines that can have multiple segments.

This approach can classify five distinct relationships between plane-point boundary component pairs, compared to three distinct relationships that are conceptualized using all other models that were discussed in Section 3.1.1 and Section 3.1.2. This approach can classify four distinct relationships between plane-point boundary component pairs that are coplanar, being the ‘Disjoint’, ‘0D-0D Point Touch’ at a (0D) plane vertex point, ‘1D-0D Point Touch’ at a (1D) plane edge line, and ‘2D-0D Point Touch’ at a (2D) plane interior extension point. The approach followed here makes an additional distinction for plane-point disjoint relationships as a point can be ‘Disjoint’ and coplanar or ‘Disjoint’ and not coplanar.

This approach can classify 14 distinct relationships between plane-line boundary component pairs, compared to 31 distinct relationships that are conceptualized using the 9IM framework in 3-D space (see Zlatanova (2000) and Table 3.4). Zlatanova (2000) considers topological plane-line scenarios where the line being considered can have a single segment or multiple segments. When the examples with multiple line segments are removed there are a total of 10 distinct relationships that can be conceptualized. Part of the reason for the higher number of relationships resulting from the approach followed here comes from making a distinction of a plane-line pair having a coplanar, parallel, or intersection projected relationship. For example, there can be a ‘0D-0D Point Touch’ classification that has the coplanar projected relationship

or the intersecting at a point projected relationship. These would both be considered the same relationship under the 9IM framework.

This approach can classify 18 distinct relationships between plane-plane boundary component pairs, compared to 38 distinct relationships that are conceptualized using the 9IM framework in 3-D space (see Zlatanova (2000) and Table 3.4). Zlatanova (2000) considers topological scenarios where the surfaces (planes) being considered can have a single plane segment or complex surfaces with multiple plane segments. When the examples of surfaces with multiple plane segments are removed there are a total of 14 distinct relationships that can be conceptualized. Similar to that of the plane-line relationships discussed above, part of the reason for the higher number of relationships resulting from the approach followed here come from making a distinction of a plane-plane pair having a coplanar, parallel, or intersection projected relationship. For example, there can be a ‘1D-1D Line Touch’ classification that has the coplanar projected relationship or the intersecting at a line projected relationship. These would both be considered the same relationship under the 9IM framework.

7.3.3 Geometrical Representations for Relationship Classifications

Looking back to Section 3.1.2, one of the main requirements for topology in 3-D GIS applications identified by Ellul & Hacklay (2006) was that each topological element should also have some form of geometric representation so that the topological analysis between objects can be reconstructed and visualized in a front-end system.

One benefit of the approach developed in this study is that topological classifications that result in some form of overlap (touch or intersection) relationship between 3-D boundary component pairs are accompanied by a geometric description of the relationship. If the boundary components being considered touch or intersect at a point, a 3-D [x, y, z] point coordinate is returned with the topological classification. If the boundary components being considered touch or intersect at a line, two 3-D point coordinates are returned with the topological classification to represent the associated line overlap geometry. If the boundary components being considered touch at a plane, a set of 3-D point coordinates are returned with the topological classification to represent the associated plane overlap geometry. While this study did not visualize the output classifications for the experimental datasets, the output results could be loaded into a spatial DBMS using the point set geometry included with each topological classification for visualization purposes.

7.4 Recommendations for Future Research

This section discusses areas where the current methodology presented in this study could be expanded upon in future research. Section 7.4.1 discusses how the methods developed here could be expanded to incorporate additional topological descriptors to increase the number of relationships that could be classified between 3-D boundary components. Section 7.4.2 discusses how the approach followed here could be expanded to include solid 3-D bodies as boundary component types. This distinction would allow for additional topological relationships such as containment to be classified between two 3-D cadastral units. Section 7.4.3 discusses how the approach followed here could be expanded to consider additional 3-D cadastral boundary definitions and additional data types. Section 7.4.4 discusses how Vector Algebra could potentially be used as an alternate approach to the CGA methods applied here to determine projected relationships between 3-D boundary component pairs. Section 7.4.5 discusses the limitations of how the code used to implement the methods was written and suggests how it could be optimized.

7.4.1 Expanding Topological Descriptors

As mentioned in Section 7.3.2 above, topological classifications between 3-D cadastral units were generalized into their boundary components either being ‘Disjoint’ from each other, ‘Touching/Meeting’ at one or more of their exterior boundary limits, or ‘Intersecting’ at both of their interior boundary extensions. By expanding the classification processes developed here to include distinctions for ‘Inside/Contains’, ‘Covers/Covered by’, and ‘Equal’ topological descriptors, a larger number of distinct relationship classifications could be distinguished between 3-D cadastral boundaries. Two examples of how the methods developed here could be expanded to achieve this are described below using relationships between line-line and plane-line boundary component pair scenarios.

All classifications for common and reverse collinear line-line boundary component pairs (see Figure 4.11 and Figure 4.13) were classified as either ‘Disjoint’, 0D-0D Point Touch’, or ‘1D-1D Line Touch’ even though there were 13 different conditions evaluated. Out of these conditions, four could be otherwise classified as ‘Covers/Covered by’, two could be otherwise classified as ‘Inside/Contains’, and one could otherwise be classified as ‘Equal’. If these additional distinctions were considered along with the common and reverse collinear projected relationship distinctions, the processes developed here could be used to evaluate 33 distinct relationship classifications between two boundary lines instead of the 10 that are described in section 4.5.

Relating to plane-line boundary component pairs, overlap relationships 22 and 23 (see Figure 6.1) are classified when there is a 1D-1D line touch and a 2D-1D line touch, respectively. While both relationships are generalized as line touch classifications, they could be further expanded to increase the number of relationships that can be classified. The 1D-1D line touch (overlap relationship 22) could be expanded to have a meet or a covers/covered by relationship classification if the line did not extend past the boundary of the plane. The 2D-1D line touch (overlap relationship 23) could be expanded to have an overlap, covers/covered by, or contains/inside relationship classification. Both examples could be implemented by adding additional evaluations between the boundary line end points and the boundary plane. If these distinctions were taken into consideration, the methods developed and tested here would be able to classify 17 distinct relationships instead of the 14 that are described in section 4.7.

7.4.2 Expand Methods to Include Solid Bodies as Boundary Components

A limitation of the relationship classification approach as it is currently implemented is that the relationship classifications evaluated are not between the 3-D cadastral objects in whole, but rather between their lower-dimensional 3-D point, line, and plane boundary components. Relationships between all boundary component pairs need to be interpreted together to describe the relationship that is occurring between two 3-D cadastral units.

This limitation can result in cases where even though there is no relationship occurring between the boundary components (i.e. their boundary surfaces are not interacting), there could still be a relationship occurring between two 3-D cadastral units in whole. An example of this contradiction would be if a unit A was completely contained within a unit B and their boundaries were not overlapping. All boundary component pair relationships would result in the ‘Disjoint’ relationship classification individually even though unit A is completely contained within unit B.

While the methods put forward in this study have been proven to be effective for describing many relationships between two 3-D cadastral units effectively, there are still possible situations where additional information is required. To fix potential ambiguities and to increase the number of relationships that can be classified between 3-D cadastral units and their boundary components, future research should expand the classification algorithms proposed here to include additional processes for evaluating relationships between body-point, body-line, body-plane, and body-body boundary component pairs. Additional evaluations between CGA planes and various point and line sets could be applied to determine if boundary components were inside or outside the surface of a 3-D cadastral unit.

7.4.3 Applying Methodology to Different Data Formats

This research could be expanded to include classification processes that could be used to identify relationships between different types of 3-D cadastral unit boundary definitions, such as boundaries defined by cylindrical surfaces and boundaries defined using open surfaces that can exist in strata units.

Aside from flat boundary planes that were modelled in this study, 3-D boundaries in Alberta, Canada can also be defined using curved cylindrical faces to represent the surface of a legal boundary. This study did not consider cadastral units with these curved boundary surfaces and future research should consider developing additional processes that can incorporate them.

All cadastral objects modelled in this study have closed boundaries. However, strata survey plans in Alberta, Canada can register 3-D units that have open surface boundaries. This type of boundary definition is usually applied when physical structures that overlap in 3-D space are owned by separate entities, or where there is some other change in RRR use. In these scenarios, boundaries are defined by the surfaces where these changes in rights occur and can result in boundary planes that extend upwards or downwards to infinity but are constrained in a horizontal direction.

The methodological processes developed in this study were designed while only considering closed boundaries that are defined by a set of flat planar surfaces. Future research should consider 3-D cadastral objects that are defined using the types of open boundaries that can be included in strata survey plans and should develop relationship classification processes that can analyze them.

7.4.4 Alternate Approach for Determining Projected Relationships

The general approach to classifying topological relationships between 3-D boundary component pair sets was to first determine the projected relationship that was occurring between the two boundary components being considered, and then to use various 3-D point-point distance evaluations to reach the final relationship classifications.

Even though this study used CGA theory and operations to determine which projected relationship was occurring between 3-D boundary component pairs, it should be possible for other methods to be used to achieve this. One of the alternative methods that would be worth considering for future research is Vector Algebra mathematics. Vector Algebra is commonly used to represent points, equations for lines, and equations for planes in 3-D space. Vector Algebra can also be applied to determine intersections between these points, equations for lines, and equations for planes. For these reasons, future research should try to

implement the general approach followed here using Vector Algebra operations and procedures. A potential benefit of using other mathematical methods such as Vector Algebra to achieve similar results would be that there are more mathematical packages available for coding when compared to that of CGA which might be useful in optimizing the classification algorithms or applying the methods to work in a spatial DBMS environment.

An example of how Vector Algebra could be applied to achieve this is presented using the line-line boundary component pair. When trying to determine the projected relationship between a line-line boundary component pair, equations could be generated for both lines in 3-D space. If the equations were the same, or the same with opposite signs, the lines being considered could be determined to have the common or reverse collinear projected relationships, respectively. If the intersection between two lines resulted in a unique point solution, the lines being considered could be determined to have the coplanar projected relationship. If the intersection between the two lines had no solution, the lines being considered could be determined to have the parallel or skew projected relationship. Similar derivations using Vector Algebra should be possible for determining the projected relationships between the other boundary component pair types considered. Vector Algebra was not considered here as CGA was the area of inquiry for this study.

7.4.5 Optimizing Implementation Code

The implementation programs that were written to test and validate the classification processes developed in this study were written to perform all calculations in batches (see Chapter 5). It also required data to be passed back and forth between MATLAB and GAViewer. This process required the manual opening of files into GAViewer and saving them back into a specific folder so that the updated files could be opened using MATLAB. This implementation approach resulted in the need to parse and sort through large text files that contained CGA operation results which took longer to process than if the calculations could be processed in real-time on an individual relationship calculation basis.

To speed up the classification processes, it would be ideal if any future implementation of these processes was written using code that could integrate the CGA operations utilized using GAViewer with the various 3-D point-point distance calculations and evaluations that were followed to classify topological relationships.

7.5 Concluding Remarks

This study showed how CGA and other mathematical methods could be used to classify geometrical and topological relationships between sets of 3-D boundary components. The results from experiments presented here suggest that the methods that were proposed and tested in this study can be useful for describing relationships between the boundaries of two 3-D cadastral units. These methods could be applied by professional land surveyors and land titles officials to assist with quality control checks associated with 3-D boundaries before submitting and registering 3-D survey plans. While there are limitations to this approach associated with data formats and topological descriptors, the methods developed in this study are promising for future research related to classifying relationships between 3-D cadastral boundaries.

References – General

- Aien, A., Kalantari, M., Rajabifard, A., Williamson, I., & Wallace, J. (2013) ‘Towards integration of 3D legal and physical objects in cadastral data models’, *Land Use Policy* 35: p.140–154. Available at: <https://linkinghub.elsevier.com/retrieve/pii/S0264837713001063> [Accessed March 19, 2021].
- Aien, A., Rajabifard, A., Kalantari, M., & Shojaei, D. (2015) ‘Integrating Legal and Physical Dimensions of Urban Environments’, *ISPRS International Journal of Geo-Information* 4(3): p.1442–1479. Available at: <http://www.mdpi.com/2220-9964/4/3/1442> [Accessed March 19, 2021].
- Atazadeh, B., Kalantari, M., Rajabifard, A., Champion, T., & Ho, S. (2016) ‘Harnessing BIM for 3D Digital Management of Stratified Ownership Rights in Buildings’, *FIG Working Week 2016*. Available at: http://www.gdmc.nl/3dcadastres/literature/3Dcad_2016_03.pdf [Accessed March 19, 2021].
- Billen, R., Zlatanova, S., Mathonet, P., & Boniver, F. (2002) ‘The Dimensional Model: A Framework to Distinguish Spatial Relationships’, *Advances in Spatial Data Handling*: p. 285–298. Berlin, Heidelberg: Springer Berlin Heidelberg Available at: http://link.springer.com/10.1007/978-3-642-56094-1_21 [Accessed March 19, 2021].
- Breunig, M., & Zlatanova, S. (2011) ‘3D Geo-Database Research: Retrospective and Future Directions’, *Computers & Geosciences* 37(7): p.791–803. Available at: <https://linkinghub.elsevier.com/retrieve/pii/S0098300411000069> [Accessed March 19, 2021].
- Clementini, E., & Di Felice, P. (1995) ‘A Comparison of Methods for Representing Topological Relationships’, *Information Sciences - Applications* 3(3): p.149–178. Available at: <https://linkinghub.elsevier.com/retrieve/pii/106901159400033X> [Accessed March 19, 2021].
- Clementini, E., Felice, P., & van Oosterom, P. (1993) ‘A Small Set of Formal Topological Relationships Suitable for End-User Interaction’, *International Symposium on Spatial Databases*: p. 277-295. Available at: <http://delab.csd.auth.gr/~alex/sdb/artSSD93.pdf> [Accessed March 19, 2021].

Dorst, L., Fontijne, D., & Mann, S. (2007) ‘Geometric Algebra for Computer Science (Revised Edition): An Object-Oriented Approach to Geometry’, *The Morgan Kaufmann Series in Computer Graphics, Elsevier Science*. Available at: <https://www.elsevier.com/books/geometric-algebra-for-computer-science-revised-edition/dorst/978-0-12-374942-0> [Accessed March 19, 2021].

Drobez, P., Fras, M.K., Ferlan, M., & Liseč, A. (2017) ‘Transition from 2D to 3D Real Property Cadastre: The Case of the Slovenian Cadastre’, *Computers, Environment and Urban Systems*, 62: p. 125-135. Available at: http://www.gdmc.nl/3dcadastres/literature/3Dcad_2017_28.pdf [Accessed March 19, 2021]

Egenhofer, M. J. (1989) ‘A Formal Definition of Binary Topological Relationships’, *Third International Conference on Foundations of Data Organization and Algorithms (FODO), Paris, France, Lecture Notes in Computer Science*, 367: p. 457-472. Available at: <https://www.cise.ufl.edu/research/SpaceTimeUncertainty/STU-internal/STURefs/Ege89.pdf> [Accessed March 19, 2021].

Egenhofer, M. J. & Herring, J. R. (1990) ‘Categorizing Binary Topological Relations Between Regions, Lines, and Points in Geographic Databases’, *Tech. Rep. 90-12. National Center for Geographic Information and Analysis, University of California, Santa Barbara, CA*. Available at: http://www.dpi.inpe.br/gilberto/references/egenhofer_binary_topological_relations.pdf [Accessed March 19, 2021].

Egenhofer, M.J., Sharma, J., & Mark, D.M. (1993) ‘A Critical Comparison of the 4-Intersection and 9-Intersection Models for Spatial Relations: Formal Analysis’, *R. McMaster and M. Armstrong (eds), Autocarto 11*. Available at: http://www.dpi.inpe.br/gilberto/references/max_comparison_4_9_intersection.pdf [Accessed March 19, 2021].

El-Mekawy, M., Paasch, J.M., & Paulsson, J. (2015) ‘Integration of Legal Aspects in 3D Cadastral Systems’, *International Journal of E-Planning Research*, 4(3), p. 47-71. Available at: https://www.researchgate.net/publication/280614334_Integration_of_Legal_Aspects_in_3D_Cadastral_Systems [Accessed March 19, 2021].

Ellul, C. & Haklay, M. (2006) ‘Requirements for Topology in 3D GIS’, *Transactions in GIS* 10(2): p.157–175. Available at: <http://doi.wiley.com/10.1111/j.1467-9671.2006.00251.x> [Accessed March 19, 2021].

FIG International Federation of Surveyors (1998) ‘FIG Statement on the Cadastre’, *FIG Publication No. 11*. Available at: <https://www.fig.net/resources/publications/figpub/pub11/figpub11.asp> [Accessed March 19, 2021].

Fontijne, D. (2010) ‘GAViewer Documentation Version 0.84’, *University of Amsterdam*. Available at: <https://geometricalgebra.org/downloads/gaviewer.pdf> [Accessed March 19, 2021].

Fu, L., Yin, P., Li, G., Shi, Z., Liu, Y., & Zhang, J. (2018) ‘Characteristics and Classification of Topological Spatial Relations in 3-D Cadasters’, *Information* 9(4), 71. Available at: <http://www.mdpi.com/2078-2489/9/4/71> [Accessed March 19, 2021].

Góźdż, K.J. & van Oosterom, P. (2016) ‘Developing the information infrastructure based on LADM – the case of Poland’, *Survey Review* 48(348): p.168–180. Available at: <http://www.tandfonline.com/doi/full/10.1179/1752270615Y.0000000018> [Accessed March 19, 2021].

Gulliver, T., Haanen, A., & Goodin, M. (2016) ‘A 3D Digital Cadastre for New Zealand by 2021: Leveraging the Current System and Modern Technology’, *FIG Working Week 2016*: p. 473-490. Available at: https://fig.net/resources/proceedings/2016/2016_3dcadastre/3Dcad_2016_36.pdf.pdf [Accessed March 19, 2021].

Hildenbrand, D. & Oldenburg, R. (2015) ‘Geometric Algebra: A Foundation of Elementary Geometry with Possible Applications in Computer Algebra based Dynamic Geometry Systems’, *The Electronic Journal of Mathematics and Technology*, 1(1). Available at: <http://www.gaalop.de/wp-content/uploads/eJMT-Hildenbrand.pdf> [Accessed March 19, 2021].

ISO International Organization for Standardization (2012) ‘ISO 19152:2012 Land Administration Domain Model (LADM)’, *Geographical Information*. Available at: <https://www.iso.org/standard/51206.html> [Accessed March 19, 2021].

Jaljolie, R., van Oosterom, P., & Dalyot, S. (2018) ‘Spatial Data Structure and Functionalities for 3D Land Management System Implementation: Israel Case Study’, *ISPRS International Journal of Geo-Information* 7(1), 10. Available at: <http://www.mdpi.com/2220-9964/7/1/10> [Accessed March 19, 2021].

Kalantari, M. & Kalogianni, E. (2018) ‘Towards LADM Victoria Country Profile – Modelling the Spatial Information’, *Proceedings of the 6th International FIG 3D Cadastre Workshop*: p. 483-498. Available at: http://www.gdmc.nl/3dcadastres/literature/3Dcad_2018_26.pdf [Accessed March 19, 2021].

Kara, A., Kathmann, R., & van Oosterom, P. (2019) ‘Towards the Netherlands LADM Valuation Information Model Country Profile’, *FIG Working Week 2019*. Available at: <https://research.utwente.nl/en/publications/towards-the-netherlands-ladm-valuation-information-model-country-> [Accessed March 19, 2021].

Kitsakis, D., Paasch, J. M., Paulsson, J., Navratil, G., Vucic, N., Karabin, M., Tenorio Carneiro, A. F., & El-Makawy, M. (2016) ‘3D Real Property Legal Concepts and Cadastre: A Comparative Study of Selected Countries to Propose a Way Forward’, *5th International FIG 3D Cadastre Workshop*: p. 1-24. Available at: http://fig.net/resources/proceedings/2016/2016_3dcadastre/3Dcad_2016_11.pdf.pdf [Accessed March 19, 2021].

Koeva, M. & Oude Elberink, S.O. (2016) ‘Challenges for Updating 3D Cadastral Objects using LiDAR and Image-based Point Clouds’, *5th International FIG 3D Cadastre Workshop*: p. 169-182. Available at: <https://research.utwente.nl/en/publications/challenges-for-updating-3d-cadastral-objects-using-lidar-and-image-2> [Accessed March 19, 2021].

Lee, B.-M., Kim, T.-J., Kwak, B.-Y., Lee, Y., & Choi, J. (2015) ‘Improvement of the Korean LADM country profile to build a 3D cadastre model’. *Land Use Policy* 49: p.660–667. Available at: <https://linkinghub.elsevier.com/retrieve/pii/S0264837715003099> [Accessed March 19, 2021].

Lee, P.-C., Wang, Y., Lo, T.-P., & Long, D. (2018) 'An Integrated System Framework of Building Information Modelling and Geographical Information System for Utility Tunnel Maintenance Management', *Tunnelling and Underground Space Technology* 79: p.263–273. Available at: <https://linkinghub.elsevier.com/retrieve/pii/S0886779818300312> [Accessed March 19, 2021].

Lemmen, C., van Oosterom, P., & Bennett, R. (2015) 'The Land Administration Domain Model', *Land Use Policy* 49: p.535–545. Available at: <https://linkinghub.elsevier.com/retrieve/pii/S0264837715000174> [Accessed March 19, 2021].

Pouliot, J. & Girard, P. (2016) 'Subsurface Utility Network Registration and the Publication of Real Rights: Towards Full 3D Cadastre', *FIG Working Week 2016*. Available at: http://www.gdmc.nl/3dcadastres/literature/3Dcad_2016_01.pdf [Accessed March 19, 2021].

Seifert, M., Gruber, U., & Riecken, J. (2016) 'Multidimensional Cadastral System in Germany', *FIG Working Week 2016*. Available at: http://www.gdmc.nl/3dcadastres/literature/3Dcad_2016_07.pdf [Accessed March 19, 2021].

Selig, J.M. (2000) 'Clifford algebra of points, lines and planes', *Robotica* 18(5): p.545–556. Available at: https://www.cambridge.org/core/product/identifier/S0263574799002568/type/journal_article [Accessed March 19, 2021].

Shojaei, D., Olfat, H., Faundez, S.I.Q., Kalantari, M., Rajabifard, A., & Briffa, M. (2017) 'Geometrical data validation in 3D digital cadastre – A case study for Victoria, Australia', *Land Use Policy* 68: p.638–648. Available at: <https://linkinghub.elsevier.com/retrieve/pii/S0264837717303150> [Accessed March 19, 2021].

Stoter, J. E. (2004) '3D Cadastre', *Delft: NGC Netherlands Geodetic Commission*. Available at: <https://ncgeo.nl/downloads/57Stoter.pdf> [Accessed March 19, 2021].

Vucic, N., Markovicovi, D., & Mi, B. (2013) 'LADM in the Republic of Croatia – Making and Testing Country Profile', *5th Land Administration Domain Model Workshop*: p. 329-344. Available at: https://www.researchgate.net/publication/270818383_LADM_in_the_Republic_of_Croatia_-Making_and_Testing_Country_Profile [Accessed March 19, 2021].

Xu, D. & Zlatanova, S. (2013) ‘An Approach to Develop 3D Geo-DBMS Topological Operators by Re-using Existing 2D Operators’, *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences* II-2/W1: p.291–298. Available at: <https://www.isprs-ann-photogramm-remote-sens-spatial-inf-sci.net/II-2-W1/291/2013/> [Accessed March 19, 2021].

Ying, S., Guo, R., Li, L., Van Oosterom, P., & Stoter, J. (2015) ‘Construction of 3D Volumetric Objects for a 3D Cadastral System’, *Transactions in GIS* 19(5): p.758–779. Available at: <http://doi.wiley.com/10.1111/tgis.12129> [Accessed March 19, 2021].

Yu, Z., Luo, W., Yuan, L., Hu, Y., Zhu, A., & Lu, G. (2016) ‘Geometric Algebra Model for Geometry-oriented Topological Relation Computation’, *Transactions in GIS* 20(2): p.259–279. Available at: <http://doi.wiley.com/10.1111/tgis.12154> [Accessed March 19, 2021].

Yuan, L., Yu, Z., Chen, S., Luo, W., Wang, Y., & Lu, G. (2010) ‘CAUSTA: Clifford Algebra-based Unified Spatio-Temporal Analysis’, *Transactions in GIS* 14: p.59–83. Available at: <http://doi.wiley.com/10.1111/j.1467-9671.2010.01221.x> [Accessed March 19, 2021].

Yuan, L., Yu, Z., Luo, W., Zhou, L., & Lü, G. (2011) ‘A 3D GIS Spatial Data Model based on Conformal Geometric Algebra. *Science China Earth Sciences* 54(1): p.101–112. Available at: <http://link.springer.com/10.1007/s11430-010-4130-9> [Accessed March 19, 2021].

Yuan, L., Yu, Z., Lu, G., Luo, W., Yi, L., & Sheng, Y. (2012) ‘Geometric algebra method for multidimensionally-unified GIS computation’, *Chinese Science Bulletin* 57(7): p.802–811. Available at: <http://link.springer.com/10.1007/s11434-011-4891-3> [Accessed March 19, 2021].

Zhang, F., Jiang, X., Zhang, X., Wang, Y., Du, Z., & Liu, R. (2016) ‘Unified Spatial Intersection Algorithms Based on Conformal Geometric Algebra’, *Mathematical Problems in Engineering* 2016: p.1–10. Available at: <https://www.hindawi.com/journals/mpe/2016/7412373/> [Accessed March 19, 2021].

Zhang, J., Yin, P., Li, G., Gu, H., Zhao, H., & Fu, J. (2016) ‘3D Cadastral Data Model Based on Conformal Geometry Algebra’, *ISPRS International Journal of Geo-Information* 5(2), 20. Available at: <http://www.mdpi.com/2220-9964/5/2/20> [Accessed March 19, 2021].

Zlatanova, S. (2000) ‘On 3D topological relationships’, *Proceedings of 11th International Workshop on Database and Expert Systems Applications*: p. 913–919. Available at: <http://ieeexplore.ieee.org/document/875135/> [Accessed March 19, 2021].

Zulkifli, N. A., Rahman, A. A., Jamil, H., Teng, C. H., Tan, L. C., Looi, K. S., Chan, K. L., & van Oosterom, P. (2014) ‘Towards Malaysian LADM Country Profile for 2D and 3D Cadastral Registration System’, *FIG Congress 2014 – Engaging the Challenges, Enhancing the Relevance*. Available at: http://www.gdmc.nl/3dcadastres/literature/3Dcad_2014_01.pdf [Accessed March 19, 2021].

References – Legislation and Government Documents

Province of Alberta (2013) ‘Surveys Act R.S.A. 2000 Chapter S-26’, *Alberta Queen’s Printer 2013*. Available at: <https://www.qp.alberta.ca/documents/Acts/S26.pdf> [Accessed March 19, 2021]

Province of Alberta (2017) ‘Land Titles Act R.S.A. 2000 Chapter L-4’, *Alberta Queen’s Printer 2017*. Available at: <https://www.qp.alberta.ca/documents/Acts/L04.pdf> [Accessed March 19, 2021]

Province of Alberta (2020) ‘Condominium Property Act R.S.A. 2000 Chapter C-22’, *Alberta Queen’s Printer 2020*. Available at: <https://www.qp.alberta.ca/documents/Acts/c22.pdf> [Accessed March 19, 2021]

Province of Alberta (2020) ‘Law of Property Act R.S.A 2000 Chapter L-7’, *Alberta Queen’s Printer 2020*. Available at: <https://www.qp.alberta.ca/documents/Acts/L07.pdf> [Accessed March 19, 2021]

Service Alberta (2011) ‘An Introduction to Alberta Land Titles’, *Service Alberta*. Available at: <https://open.alberta.ca/publications/5649897> [Accessed March 19, 2021]