GeoInformatica 8:3, 263–280, 2004

Towards Temporal Dynamic Segmentation

BO GUO*
Maricopa County, Phoenix, AZ 85003, USA
E-mail: boguo@mail.maricopa.gov

CARL E. KURT
Department of Civil Environmental, and Architectural Engineering, The University of Kansas, Lawrence, KS 66045, USA
E-mail: ckurt@ku.edu

Received April 10, 2003; Revised March 25, 2004; Accepted March 25, 2004

Abstract

In recent years, there have been many research studies focusing on linear data modeling as well as on temporal GIS-T (GIS for transportation) implementations. However, what was fundamentally missing from the research circle was the study of a methodology for processing and representation of linearly referenced features in the temporal context, or temporal dynamic segmentation.

This paper dissects the functional specifications of temporal dynamic segmentation. The authors start by exploring the definition and characteristics of dynamic segmentation. The scope of dynamic segmentation is extended to include two functional categories and three essential functions. The paper then defines spatiotemporal segment and a spatiotemporal join operation, which are the building blocks and the key mechanism behind temporal dynamic segmentation. A set of metric criteria for identifying spatiotemporal segment topologies are proposed as an effective alternative to the more general, but more costly, frameworks for the identification of topological relationships. The authors finally present functional specifications of the three essential functions of dynamic segmentation.

Keywords: GIS for transportation, temporal dynamic segmentation, linear referencing, linear referencing system, spatiotemporal

1. Introduction

There have been many research studies focusing on linear data modeling in recent years. Some of these models, including the latest MDLRS model [1], incorporated the temporal dimension(s). There have also been some published research attempts, by Zhao et al. [2], and by Raafat and Yang [3], for example, in implementing temporal GIS for transportation (GIS-T) projects. However, the ''catchall'' models, conceptual or otherwise, often leave little guidance for practitioners to resolve their daily challenges; the implementation research studies were found confined to the area of spatiotemporal representations that were outside the linear referencing system (LRS) domain [4]. There appears to be a lack of research in the fundamental methodology that is required to handle linearly referenced data in the temporal dimension.

*Corresponding author.

Just as classic dynamic segmentation has made LRS a practical data model; any temporal LRS data model needs the support of temporal dynamic segmentation. The objective of this research is to extend the classic dynamic segmentation into temporal dynamic segmentation. This is a highly interdisciplinary field of study, understanding some key concepts is an essential prerequisite. However, due to limitations of the paper, some related fields of study such as RDBMS, time in databases, time in spatial databases, among many others are left out. Interested readers are encouraged to reference numerous research publications [9–16].

1.1. Dynamic segmentation

The most fundamental functions of GIS-T are those addressing issues relating to transportation networks and features associated with these networks. In this sense, what sets GIS-T applications apart from general GIS applications are the predominant needs and challenges to model linear objects representing transportation infrastructure and features (attributes or events) associated with such infrastructure.

Static segmentation is the traditional method for modeling features of network segments. Multiple features of a segment are directly (typically via a foreign key) linked to the geometry of the segment. As Hickman [5] put it, such method causes ''excessive and uncoordinated segmentation'' in network and prevents effective network analysis and data sharing. Fletcher [6] ushered in a new design paradigm in modeling transportation network, which is characterized by the separation between feature data and spatial objects representing the underlying network, and the separation between data representing different types of features. The feature-network separation allows sharing of network infrastructure and promotes standardization; the feature–feature separation reduces feature segment fragmentation and avoids redundancy.

However, the new linear modeling approach necessitated new tools to present and analyze data. Due to the feature-network separation, linear features can no longer be directly mapped. Similarly, due to the feature–feature separation, cross feature analysis is made difficult. This is where dynamic segmentation technology comes into play.

Dynamic segmentation is often referred to as the technique that computes coordinates of linearly referenced features based on the referenced network objects ''on-the-fly'', thus avoiding the need for explicit representation and storage of feature geometry with the features in a database [7]. ESRI [8] includes linear data processing into the definition: software that ''can store, display, query and analyze the information associated with linear features without modifying the underlying linear data coordinates.''

A more complete view of dynamic segmentation becomes clear when functionality required to support the new linear data design paradigm is examined. Dynamic segmentation involves in two functional categories: linear network preparation and linear feature data representation and processing. In the area of linear network preparation, dynamic segmentation needs to support the building and maintenance of traversal networks. In the category of linear feature representation and processing, dynamic segmentation needs to address three essential functions. The first is segment geocoding, a process in which geometry or locational information of linearly referenced features is derived and subsequently mapped to relevant networks. The second is network overlay, or segment overlay in this paper, which combines two or more feature data sets by joining them based on their common relationship: location. The segment maintenance function, which has not been addressed by any published studies, ensures segment consistencies when feature data are updated.

To help understand the need for a temporal dynamic segmentation, the implied assumption of the classic dynamic segmentation has to be revealed—feature data and network data share common temporal existence. As a result, traffic collisions occurred on old road alignments that no longer exist could be geocoded on the re-aligned road segments through the segment geocoding function. Through the segment overlay function, traffic collisions could be reported under a different jurisdiction if the road segments on which the collisions occurred had been redistricted. To overcome this limitation, a temporal dynamic segmentation, needs to reconcile temporal existence between features and between feature and network as key part of its operations.

Since the implementations of traversal network preparation are model-specific and vendor specific, discussion in this functional category is not provided in the paper. The authors will instead focus on providing temporal extension to the second functional category, specifically, the three essential functions of linear data representation and processing: segment geocoding, segment overlay, and segment maintenance.

2. Segment fundamentals

In the linear data domain, feature data records are often generalized as segments, and networks are considered as collections of connected segments. Therefore, understanding of the definition and operations of segments in the temporal context is of fundamental importance.

2.1. Feature classification and segment representations

2.1.1. Classification of linear feature data. Transportation-related features are spatiotemporal. In the spatial dimension, features can be spatially classified as point features or linear features. In the time dimension (valid-time) they can be classified as instant features or period features. Features can spatiotemporally be categorized as any of the four possible combinations: point-instant, point-period, linear-instant and linear-period. For example, traffic signs belong to the point-period feature classification while accidents typically are in the point-instant category. Pavement types and pavement sweeping activities are in linear-period and linear-instant feature classes, respectively.

2.1.2. Representation of spatiotemporal segment. *Spatial segment.* When temporality is not concerned, segment refers to the spatial attributes of a linearly referenced feature. It represents a continuous section bounded by a pair of begin-end measurements on a given traversal.

Mathematically, a spatial segment is defined as:

$$s_{LN} = \left\{v, [m_i^v, m_j^v) \middle| v \in \mathrm{dom}(V), m_i^v, m_j^v \in \mathrm{dom}\left(M^V\right), m_i^v \leq m_j^v\right\},$$

where dom () represents the domain of a given attribute: V, traversal; M^V, traversal measure; v, a traversal; and m^v, a measure on traversal v.

A point can be considered a special segment with identical measurements at its boundary. For simplicity, however, a point segment is defined with one measurement on the given traversal.

$$s_{\mathrm{PT}} = \left\{v, m^v \,\middle|\, v \in \mathrm{dom}(V), m^v \in \mathrm{dom}\left(M^V\right)\right\}.$$

Segments are topological; their location and or geometric features are in traversals being referenced.

Temporal segment. Period is the temporal equivalent of a spatial segment; it can be defined as:

$$t_{\mathrm{PER}} = \left\{\left[t_i, t_j\right) \,\middle|\, t_i, t_j \in \mathrm{dom}(T), t_i \leq t_j\right\},$$

where T is the time attribute, and $t \in \mathrm{dom}(T)$ denotes a time point.

The equivalent of a point segment is the time instant:

$$t_{\mathrm{INS}} = \{t_i \mid t_i \in \mathrm{dom}(T)\}.$$

Spatiotemporal segment and TTD. For a spatiotemporal segment, its temporal dimension denotes the life span of the segment while the spatial dimension is used to derive its spatial characteristics. Since a spatial segment and a temporal segment can be graphically represented as straight lines along their respective axes, a general spatiotemporal segment can be represented graphically as a rectangular in the 2-D space bounded by two orthogonal axes: traversal measurement and valid time. The traversal-time diagram (TTD) depicts such space where feature segments as well as different versions of a traversal can be represented in simple shapes. TTD space offers an effective means to understand the behavior of and topological relationships among features and traversals.

Based on their spatial and temporal properties, spatiotemporal features are classified into linear period, linear instant, point period or point instant features. Corresponding to the feature classifications are four types of spatiotemporal segments, each with different shapes in the corresponding TTD space as rectangles, horizontal lines, vertical lines or points (figure 1). The simplicity of spatiotemporal segments in TTD space has significant bearing on the segment operations to be discussed later.

Spatiotemporal segments, in their generalized or special forms, can be defined as:

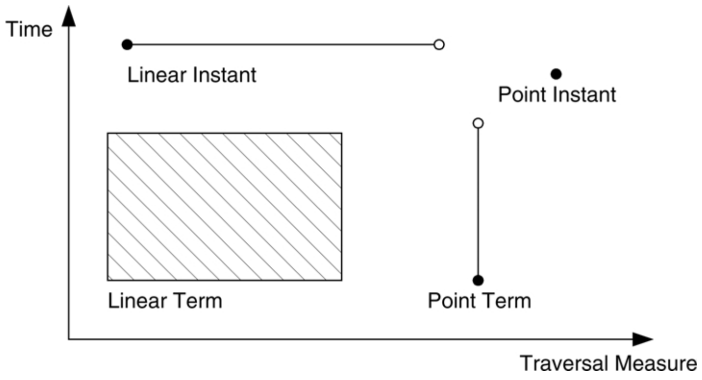


Figure 1. Traversal time diagram expresses feature spatiotemporality.

Linear period segment: $st_{\mathrm{LT}} = \{s_{\mathrm{LN}}, t_{\mathrm{PER}} \mid s_{\mathrm{LN}} \in \mathrm{dom}(S_{\mathrm{LN}}), t_{\mathrm{PER}} \in \mathrm{dom}(T_{\mathrm{PER}})\}$.

Linear instant segment: $st_{\mathrm{LI}} = \{s_{\mathrm{LN}}, t_{\mathrm{INS}} \mid s_{\mathrm{LN}} \in \mathrm{dom}(S_{\mathrm{LN}}), t_{\mathrm{INS}} \in \mathrm{dom}(T_{\mathrm{INS}})\}$.

Point period segment: $st_{\mathrm{PT}} = \{s_{\mathrm{PT}}, t_{\mathrm{PER}} \mid s_{\mathrm{PT}} \in \mathrm{dom}(S_{\mathrm{PT}}), t_{\mathrm{PER}} \in \mathrm{dom}(T_{\mathrm{PER}})\}$.

Point instant segment: $st_{\mathrm{PI}} = \{s_{\mathrm{PT}}, t_{\mathrm{INS}} \mid s_{\mathrm{PT}} \in \mathrm{dom}(S_{\mathrm{PT}}), t_{\mathrm{INS}} \in \mathrm{dom}(T_{\mathrm{INS}})\}$.

2.2. *Segment relationships*

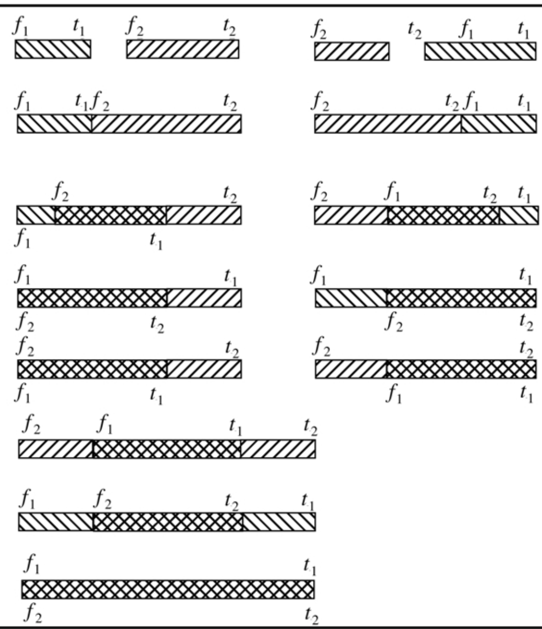
Identifying spatiotemporal segment relationships are of fundamental importance to various segment operations such as geocoding, maintenance and overlay (to be discussed later in the paper.) The application of the four-intersection model or DE-9IM model requires complicated computation not only for the derivation of interior, boundaries and/or exterior of segments involved, but their intersections as well. More information on the two models can be found in Egenhofer and Herring [18], Open GIS Consortium [19], Scarponcini [20].

Fortunately, due to the simplicity of segment shapes projected into the traversal-time space, segment metrics can be efficiently and intuitively used to determine topological relationships between segments. Table 1 presents metric criteria for identifying the eight relationships between two 1-D segments. The measurement comparison is simple and intuitive and is applicable to both spatial segments and temporal segments.

Topology of spatiotemporal (2-D) segments can be derived from its 1-D component topologies—spatial topology and temporal topology. Topological relationships of spatial and temporal components dictate the overall topological relationships between two spatiotemporal segments. Different component topology has different rank in terms of its influence on the combined 2-D topology. Such rank characteristic simplifies the

Table 1. Criteria for identifying topological relationships between two 1-D segments.

Relationship Type	*Metric Criteria*	*Topological Configurations between Two 1-D Segments*
Disjoint	$t_1 < f_2$ OR $f_1 > t_2$	
Meet	$t_1 = f_2$ OR $f_1 = t_2$	
Overlap	($f_1 < f_2$ AND $t_1 < t_2$ AND $t_1 > f_2$) OR ($f_1 > f_2$ AND $f_1 < t_2$ AND $t_1 > t_2$)	
Cover	($f_1 = f_2$ AND $t_1 > t_2$) OR ($f_1 < f_2$ AND $t_1 = t_2$)	
Covered_by	($f_1 = f_2$ AND $t_1 < t_2$) OR ($f_1 > f_2$ AND $t_1 = t_2$)	
Inside	($f_1 > f_2$ AND $t_1 < t_2$)	
Contains	($f_1 < f_2$ AND $t_1 > t_2$)	
Equal	$f_1 = f_2$ AND $t_1 = t_2$	



identification of 2-D segment topological relationship. Discussion on component topology rank and how it influences the overall 2-D topology can be found in Guo's dissertation [*4*].

2.3. Basic segment operations

Under the point-set theory, set-theoretical operations, such as set intersection, set union and set differences, can be used to depict the basic segment operations. For example, segment intersection operation results in segment(s) common to two sets of segments participates in the operation. Segment union operation results in a set of segments representing the space occupied by any segment participating in the operation.

The basic segment operations often result in decomposition of segments. The outcome of the operation depends on the topological relationship between the two participating sets of segments. To illustrate this, suppose there are two sets of segments, each containing one segment, Seg A and Seg B, respectively, and that Seg A and Seg B are on the same traversal. Figure 2 shows the two segments in overlap relationships with two slightly different topological configurations. Configuration I, where the current Seg A overlaps with the current Seg B, breaks the two segments into four non-overlapping segments, each accounts for the intersection or the differences of the original segments. In Configuration

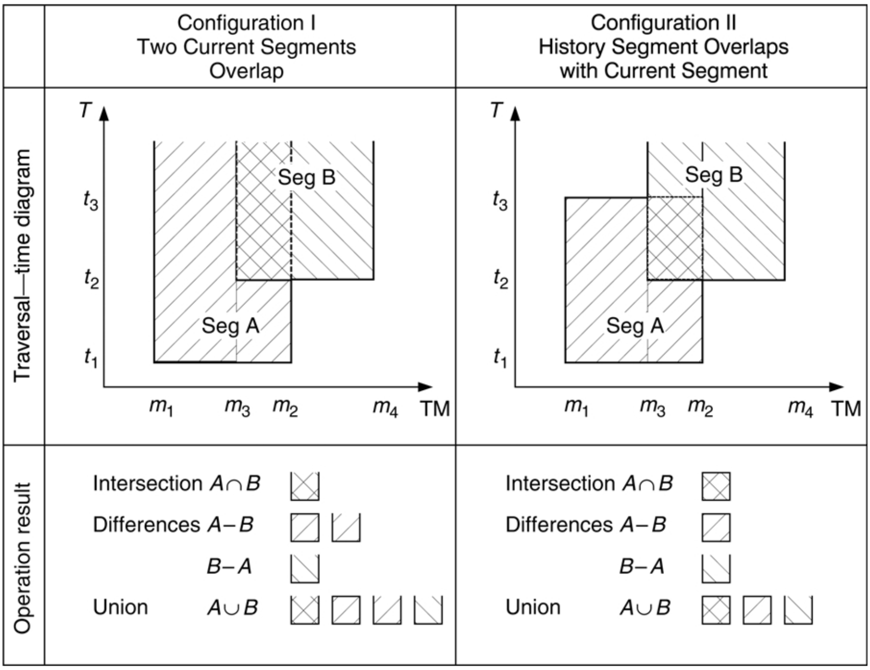


Figure 2. Basic segment operations illustration.

II, where the history Seg A overlaps with current Seg B, five segments are generated however.

The basic segment operations are binary operations where two sets of segments are involved. Similar to the corresponding traditional set operations, the basic segment operations are closed within the universe of segments (meaning that a segment operation always produces another set of segments). As a result, more than two sets of segments can be operated in a nested fashion.

2.4. *Segment join operations*

The basic segment operations focus on the topological aspect of segments involved in both sets. To bring qualified features from both sets together based on their spatiotemporal relationships, however, requires segment join operations. Just as join operations are the most important and certainly the most versatile operations to a RDBMS, so are segment join operations to dynamic segmentation.

Before expressing the temporal join operations, the following schema level notations are introduced. The notation set is based on what Gundadhi and Segev [13] had used in their temporal join optimization research.

At the schema level, let K_i, represent surrogate key with domain dom (K_i); A_{ij}, spatiotemporal-invariant attributes with domain dom (A_{ij}); V, traversal; M_B^V, begin measure on V for a linear feature; M_E^V, end measure on V for a linear feature; M_A^V, at measure on V for a point feature; T_B, begin valid-time for period feature; T_E, end valid-time for period feature; T_O, on valid-time for instant feature.

Further define $R_i = \{K_i, A_{i1}, \ldots, A_{im}, S, T\}$ as a schema for spatiotemporal segments; $S = \{V, M_B^V, M_E^V\}$ as linear sub-schema, or $S = \{V, M_A^V\}$ as point sub-schema; $T = \{T_B, T_E\}$ as a period
sub-schema, or $T = \{T_A\}$ as an instant sub-schema; $R_i' = R_i - T - S$ as the non-spatiotemporal sub-schema of R_i; $r_i(R_i)$ as a relation on schema R_i; x_i as a tuple in relation r_i; $x_i()$ as projection of x_i on given attribute(s), ϕ as the value of null; $x_i(S)$ as a spatial location (linear or point) for tuple x_i; $x_i(T)$ as temporal attribute (period or instant) for tuple x_i.

2.4.1. *Spatial join.* Spatial joins are behind spatial search and analysis operations such as search by radius or polygon overlay analysis, in GIS software. In the linear data domain, linear spatial join operations are behind segment overlay and segment geocoding functions of dynamic segmentation. A linear spatial inner join operation can be defined as:

$$
\begin{aligned}
r_1 S - \text{Join } r_2 = \Big\{ x_3 \mid x_3(R_1') &= x_1(R_1') \wedge \\
x_3(R_2') &= x_2(R_2') \wedge \\
x_3(V_3) &= x_1(V_1) \cap x_2(V_2) \wedge \\
x_3\left(M_B^{V_3}\right) &= \max\left(x_1\left(M_B^{V_1}\right), x_2\left(M_B^{V_2}\right)\right) \wedge \\
x_3\left(M_E^{V_3}\right) &= \min\left(x_1\left(M_E^{V_1}\right), x_2\left(M_E^{V_2}\right)\right) \wedge \\
x_3\left(M_B^{V_3}\right) &\leq x_3\left(M_E^{V_3}\right) \Big\}.
\end{aligned}
\tag{1}
$$

Typical segment overlay function is accomplished by inner-joining two sets of linear segment features. The result is a new set of segments that share segment spatial attributes common to the two feature sets and that contain non-spatial attributes from the two features sets. However, few dynamic segmentation implementations support any of the outer join types beside the inner-join operation.

2.4.2. *Spatiotemporal join operations.* A segment spatiotemporal join combines spatial join with temporal join. The following expression represent a spatiotemporal inner join.

$$\begin{aligned} r_1 ST - \text{Join } r_2 = \{x_3 \mid & x_3(R'_1) = x_1(R'_1) \wedge \\ & x_3(R'_2) = x_2(R'_2) \wedge \\ & x_3(S) = x_1(S) \cap x_2(S) \wedge \\ & x_3(T) = x_1(T) \cap x_2(T) \wedge \\ & x_1(S) \cap x_2(S) \neq \phi \wedge \\ & x_1(T) \cap x_2(T) \neq \phi\}. \end{aligned} \tag{2}$$

Spatial location is more selective than temporal location. After all, features reside on many different traversals while sharing the one-and-only time axis. Therefore, spatiotemporal join operations are considered as an extension to spatial joins as opposed to temporal joins. Accordingly, the spatial dimension is treated as the primary dimension. The spatial topological relationships are examined first before the temporal topological relationships when processing a spatiotemporal join operation. For example, if a segment in one relation does not spatially intersect with any segment in the other relation, the segment is destined for an appropriate outer join set. If a segment spatially intersects with one or more segments in the other relation, the segment's temporal relationships with the spatially intersected segment(s) will be evaluated to identify the existence of spatiotemporal intersection(s). Such existence will then lead to decomposition of the spatiotemporally intersected tuples in both relations by the 2-D region. The decomposed tuples sharing the same space are included in the inner join result set, while the decomposed tuples unique to each other will be collected in the appropriate outer join sets.

3. Temporal dynamic segmentation

Based on the discussion on definition segment and segment operations, and the discussion on segment topology identification criteria, this section proposes functional requirements for temporal dynamic segmentation in three essential functions: segment geocoding, segment overlay and segment maintenance. The three categories represent the linear data representation and processing capabilities of dynamic segmentation.

3.1. Segment geocoding

The feature-based modeling approach separates features from geometry. As a result, geometry of feature segments is not stored with the features; instead, it resides in the network to which the feature segments are referenced. Segment geocoding translates the referenced spatial attributes of features into physical geometry based on the geometry of the network to which the features are referenced. The resulting geometry, usually in Cartesian coordinate pair or pairs identifying point or linear features, can then be plotted.

However, all features cannot be geocoded due to data deficiencies in feature segment data and/or in traversal networks. Examples of such data deficiencies in dynamic segmentation are unmatched traversals or out-of-bound measurements. There can be three geocoding configurations: mappable, partially mappable, and un-mappable. A feature segment is mappable if its entire geometry can be can be mapped to the matching traversal. The partially mappable configuration occurs when geometry of a feature segment can be mapped partially to the matching traversal. Similarly, the un-mappable configuration occurs when feature geometry cannot be mapped to any traversal. The identification of the different mapping configurations is important for GIS-T practitioners, as it tells the truth in data. Most commercial implementations of dynamic segmentation do not provide such capability.

In temporal dynamic segmentation, geocoding of spatiotemporal segments requires the common temporal existence of features and their matching traversals. A feature segment is mappable only when its valid time period is within or equal to that of the matching traversal and its measurements are within or equal to the measurement range(s) of the traversal. This temporal restriction results in two mapping states, one-to-one mapping state and one-to-many mapping state, under the mappable or the partially mappable configuration. In one-to-one mapping state, a spatiotemporal feature segment is mapped onto one particular version of a traversal. In one-to-many mapping state, however, one spatiotemporal feature segment is mapped onto more than one versions of a traversal.

To illustrate the various mapping configurations in temporal dynamic segmentation, it is assumed that traversal *v* underwent a re-alignment and extension construction in 1980, which resulted in two different traversal versions. When the two versions of traversal *v* and the feature segments, identified by A through F, are mapped in the TTD space (figure 3), various geocoding configurations and states become clear. A summary is provided in table 2.

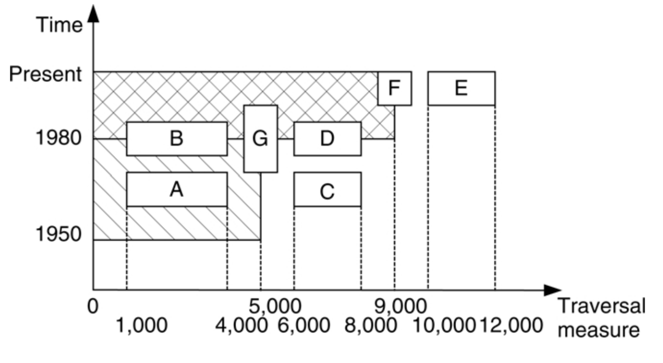


Figure 3. Different geocoding configurations and states.

Table 2. Summary of geocoding configurations and states.

Configuration	*State*	*Example*	*Classic Dynamic Segmentation*
Mappable	One-to-one	A	Feature space is contained in or covered by space of a traversal version
	One-to-many	B	Feature valid time period crosses valid time periods of two or more traversal versions, AND Feature space is contained in or covered by the combined space of two or more traversal versions.
Partially mappable	One-to-one	D	Feature space overlaps with space of a traversal version.
	One-to-many	G	Feature valid time period crosses valid time periods of two or more traversal versions, AND Feature space overlaps with the combined space of two or more traversal versions.
Un-mappable	NA	C, E	Feature space is jointed or disjointed with the space of ANY traversal version.

3.2. Segment overlay operation

Segment overlay function brings together different features that are stored in separate structures for analysis. It is implemented via spatial or spatiotemporal join operations.

Spatiotemporal segment overlay performs spatial join operations only on those feature segments that share common temporal existence. Figure 4 illustrates a simple spatiotemporal segment overlay operation between a surface type feature and a maintenance district feature on a traversal. There exist three different feature-value configurations. Segments 1 through 5 have both surface type and maintenance district values while segment 6 and segment 7 have only surface type or maintenance district value. Segment feature-value configurations are dependent on options used for the operation.

There are four options corresponding to the four join types: inner join, left outer join, right outer join and full outer join. Two more options, left difference and right difference, are added to the list as by-products of the join operations. Different options satisfy different analysis needs, and result in segments with different feature-value configurations.

3.2.1. Feature classifications and segment overlay options. A segment overlay operation is a closed operation. This property allows multiple (more than two) segment datasets to participate in the operation in a nested fashion. However, segments can have different spatiotemporal classifications, which affect outcomes of the overlay operations. Moreover, the applicability of certain overlay options is also affected by the relative position of two input segment datasets.

For example, in the case of accident incidents overlays on county-maintained roads, the first or left table represented 0-dimension point instant features and the second or right table represented 2-D linear period features. The semantically meaningful overlay options are inner-join and left options (left-join and left-difference). The output segments are zero-dimension accidents occurred on and/or off the county-maintained road system.

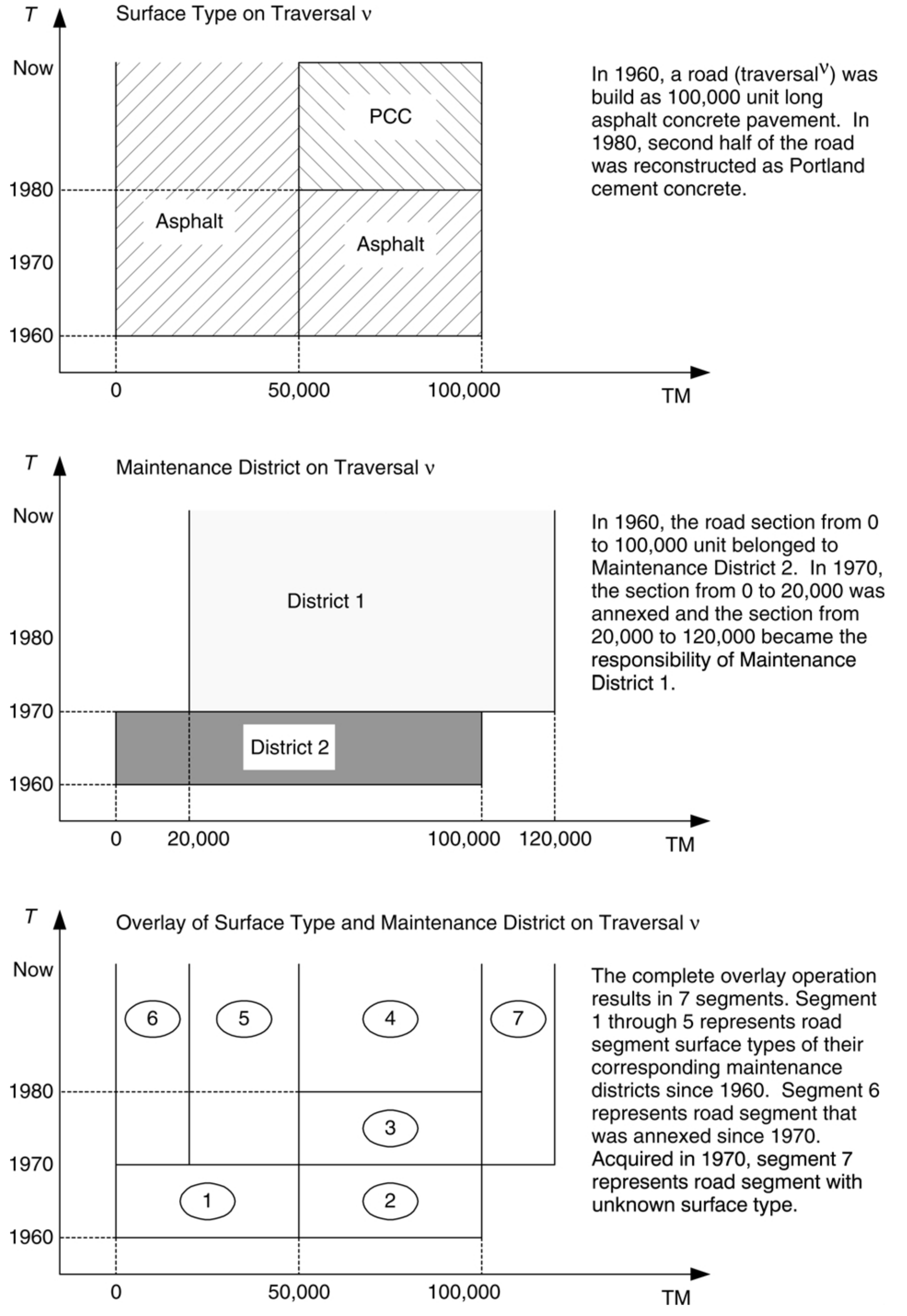


Figure 4. Spatiotemporal segment overlay illustration.

The following rules regarding spatiotemporal overlay analysis can be generalized:

1. If two overlaying segment data sets share an identical spatiotemporal class, the resulting segments are of the identical class. In addition, all six overlay options are applicable.
2. If two overlaying data sets are of different spatiotemporal class and the dimension of the two features are different, the resulting segments share the feature class of the data sets with the lower dimension. Overlay options are limited to the inner-join option and the left or the right options depending on the position of the data sets with the lower dimension.
3. If two overlaying data sets are of different spatiotemporal class and the dimensions of the overlaying data sets are equal, the resulting segments assume the lower spatial dimension in the spatial axis and lower temporal dimension in the temporal axis. Overlay options are limited to the inner-join and the left or the right options depending on the position of the data sets with the lower spatial dimension.

3.3. Segmentation maintenance operations

The maintenance of dynamically segmented feature data in RDBMS often requires more than the basic SQL data manipulation language (DML) commands of delete, update, and insert. This is because changes in an individual segment's dimension(s) often affect the neighboring segment(s) connecting to it. As a result, to maintain segment data set consistencies such as value-redundancy and value-contradiction [4], more complex maintenance operations are needed.

In the temporal database context, obsolete data are not physically deleted, but are retired, or logical deleted. Logical deletions are accomplished by assigning non-future values to the valid end-time column of the target data records.

3.3.1. Insertion. A qualified spatiotemporal insert segment can be directly inserted through SQL's `insert` command if it has spatiotemporal disjoint or meet relationship with any other segment on the traversal or it is the only segment on a traversal in the target data set (figure 5).

3.3.2. Deletion. The spatiotemporal segment deletion breaks the original segment into remaining segment(s) representing portion(s) of the original segment(s) left undeleted. In addition, the action generates a retired segment, representing the portion that was active before the date of the deletion. The retired segment can be considered as having been logically deleted from the dataset (figure 5).

3.3.3. Update. The update operation is triggered when the state or value of a segment or a portion of a segment has changed. Similar to the delete operation, the update operation breaks the original segment into remaining segments and a retired segment. The retired

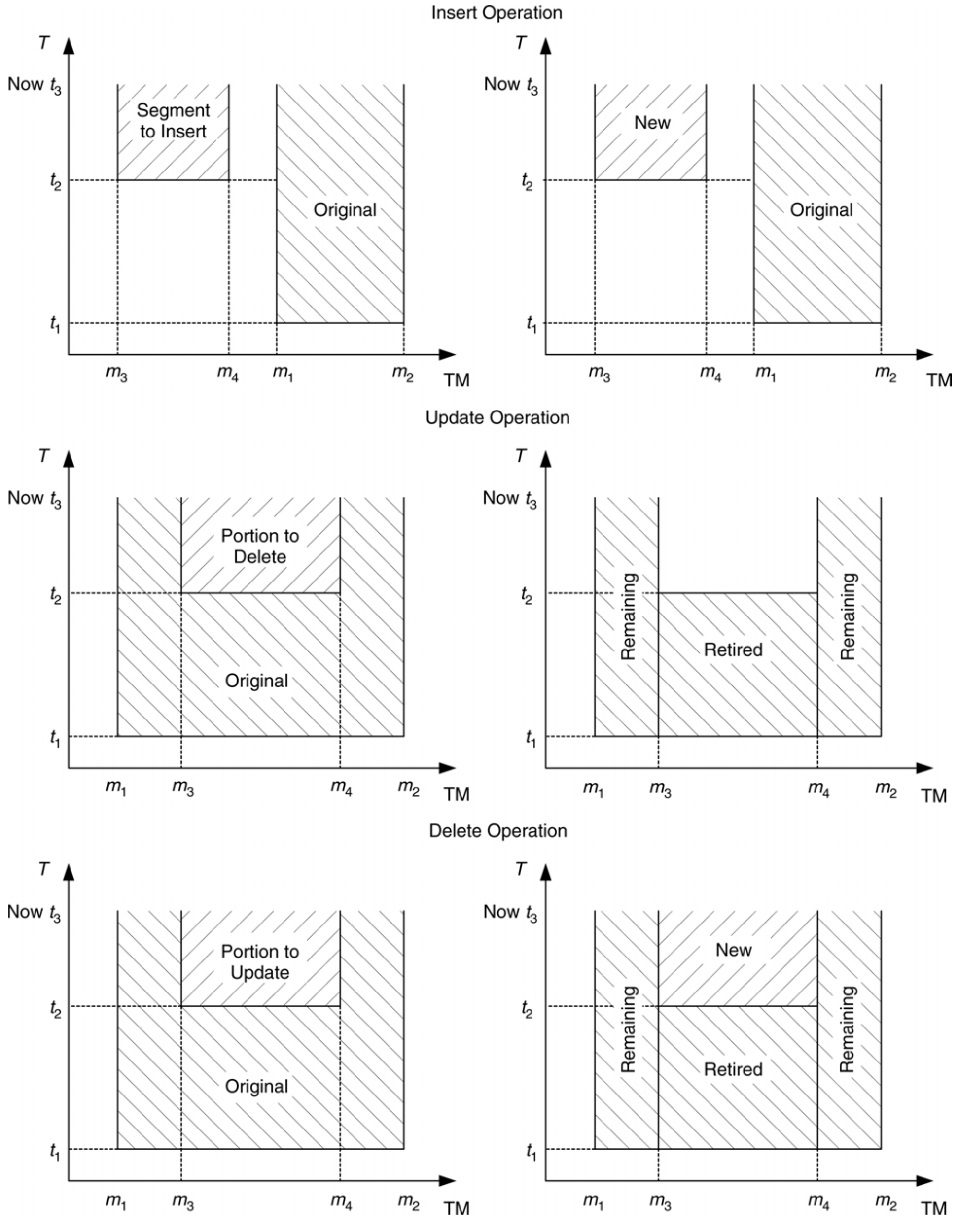


Figure 5. Spatiotemporal segment maintenance operations.

segment represents the state of the segment before the update operation occurs. The new segment is inserted representing the current state of the segment portion (figure 5).

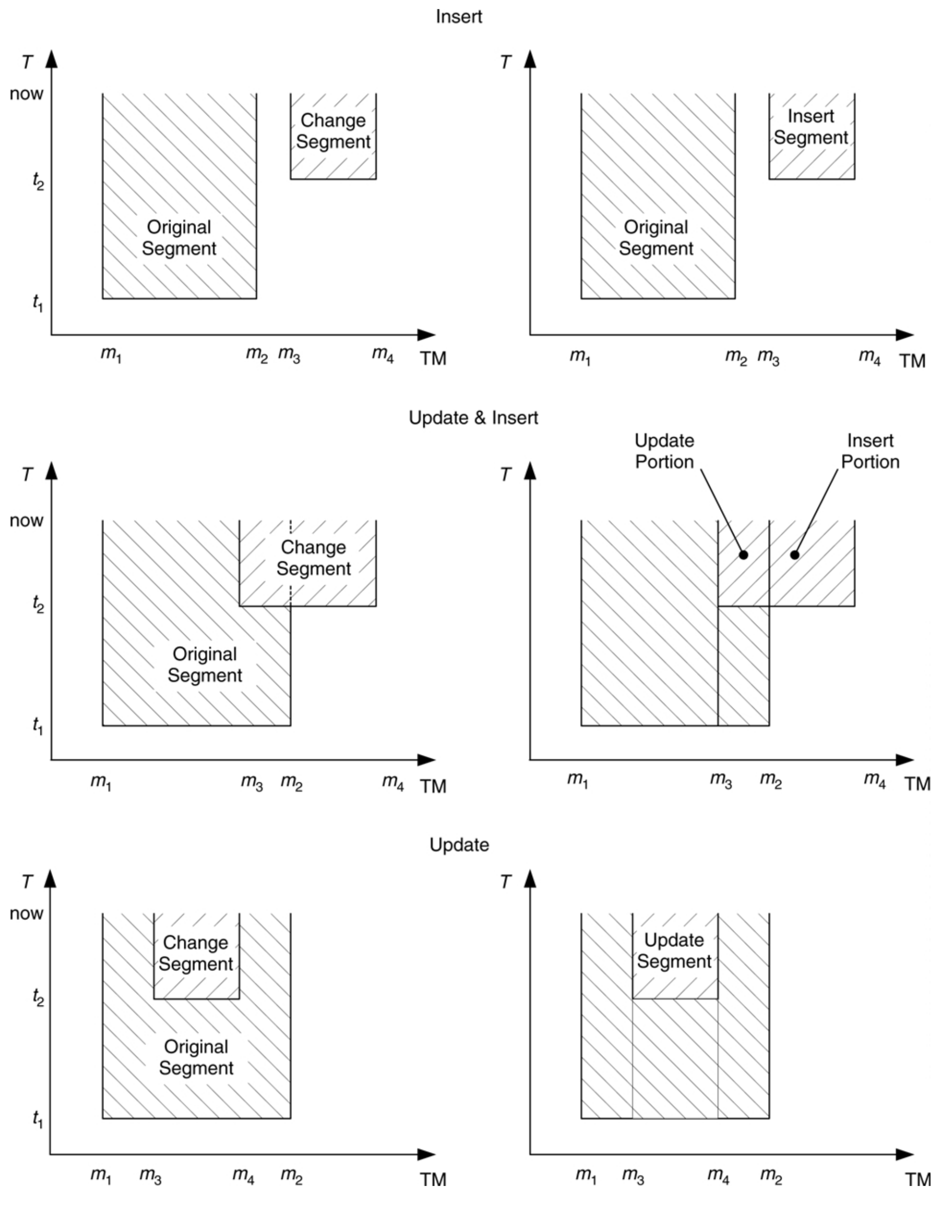


Figure 6. Segment topology and insert/update operations.

3.3.4. Segment topology and segment maintenance operations. One important characteristic of segment maintenance options is their dependencies on the topological relationships between a change segment and original segment(s). In fact, the topology often determines the applicability of both types and outcomes of the operations.

Depending on the relationships between a change segment and original segment(s), a segment deletion operation falls in one of the three scenarios. A null deletion occurs when the change segment is in disjoint or meet relationships with the original segment(s). A partial deletion occurs when the change segment is in an overlap relationship with the original segment(s) and only the portion common to both segments, or the overlapped portion is deleted. A full deletion occurs when the change segment is contained in, or covered by, or equal to the original segment. The number of resulting remaining segments is also depends on the topological relationships. For example, no remaining segment will be generated if a change segment forms an equal or contain relationships with the original segment(s).

Similarly, the topological relationships between a change segment and the original segment(s) dictates whether an insert operation, or an update operation or both should be performed. If the change segment and the original segment(s) form a disjoint or meet relationship, an insert operation is called for. If the relationship is is-contained or is-covered or equal, an update operation is assumed. Any other relationships (contain, cover or overlap) lead to a combination of update and insert operations. As illustrated in figure 6, the change segment is subsequently decomposed into two sub-segments. The sub-segment common to the change segment and original segment updates the original segment;

4. Conclusions

Proposing and developing temporal LRS data models may have resolved half of the puzzle in implementing successful temporal GIS-T projects; however, resolving the other half of the puzzle requires a dynamic segmentation technology that operates in the temporal domain. This paper has presented a temporal dynamic segmentation framework in terms of functional descriptions and specifications. The success of Maricopa County (Arizona) DOT's Roadrunner application that was built upon the technology should provide a practical reference to the proposed work. Background information as well as implementation details can be found in Guo's [4] dissertation.

5. Summary contributions

This paper proposed a temporal extension to dynamic segmentation. In the process, the following contributions have been made.

1. The paper gives a more complete definition of dynamic segmentation. Dynamic segmentation contains functions in the area of linear network preparation and maintenance and in the area of linear data representation and processing. In the former area, key functions should build and maintain traversal networks. In the latter area, functions for segment geocoding, segment overlay and segment maintenance should be provided.

2. The paper classifies linear features into four classifications based on their spatiotemporality: linear period, linear instant, point period and point instant.
3. The paper proposes TTD as a means to analyze interactions and topology between spatiotemporal segments.
4. The study has proposed simple metric criteria for identification of topology amongst spatiotemporal segments.
5. The paper has proposed a definition for spatiotemporal inner join operation.
6. Finally, the research put together a temporal extension to dynamic segmentation in the area of linear data representation and processing.

6. Limitations and future research

The main limitation of the proposed temporal dynamic segmentation is perhaps the lack of complete support for the types of feature dynamism identified by Segev and Shoshani [21] Shoshani and Kawagoe [22] which are step-wise constant, discrete, and continuous. In fact the limitation lies in the definition of segment. Being homogenous within its temporal bounds, the segment is suitable for step-wise constant features such as pavement types, or discrete features such as traffic collisions. It is not suitable for modeling continuous features such as moving vehicles or pavement condition rating. While the continuous feature dynamism could be simulated as a collection of discrete features under finer temporal resolutions within the proposed framework, such simulation can be difficult and costly to implement. It is, therefore, desirable to incorporate the support for the continuous feature dynamism in future research in the attempt to meet the demands of the transportation industry.

Acknowledgments

The authors would like to acknowledge the reviewers for their insightful comments and revision requests. Special thanks go to Maricopa County Department of Transportation where most of the research, development and verification of the technology took place.

References

1. T.M. Adams, N.A. Koncz, and A.P. Vonderohe. ''Guidelines for the implementation of multimodal transportation location referencing systems,'' *NCHRP Report 460*, TRB, National Research Council, Washington, D.C., 2001.
2. F. Zhao, L. Wang, H. Elbadrawi, and L. Shen. ''Temporal geographic information system and its application to transportation,'' *Transportation Research Record 1593*, TRB, National Research Council, Washington, D.C., 47–54, 1997.
3. H. Raafat, and Z. Yang. ''Relational spatial Topologies for historical geographical information,'' *International Journal of Geographical Information Systems*, Vol. 8(2)163–173, 1994.
4. B. Guo, ''A feature-based linear data model supported by temporal dynamic segmentation.'' Unpublished Doctoral Dissertation, 2001.

5. C. Hickman, ''Feature-based data model and linear referencing systems: Aid to avoiding excessive segmentation of network links.'' *Proceedings of the Geographic Information Systems for Transportation Symposium, Sparks, NV,* 89–113, 1995.
6. D. Fletcher, ''Modeling GIS transportation networks,'' *URISA Proceedings, Vol. II,* Washington, D.C., URISA, 84–92, 1987.
7. A. Vonderohe, L. Travis, R.L. Smith, and V. Tsai. ''Adaptation of geographic information systems for transportation,'' *NCHRP Report 359*, TRB, National Research Council, Washington, D.C., 1993.
8. Environmental Systems Research Institute, Inc., Dynamic Segmentation—Modeling Linear Features, Arc/Info 6.0 User's Guide, 2nd ed., 1992.
9. A. Silberschatz, H. Forth, and S. Sudarshan. Database System Concepts, 3rd ed. McGraw Hill, 1996.
10. J. Clifford and A. Croker. ''The historical relational data model (HRDM) and algebra based on lifespans,'' *Proceedings of the International Conference on Data Engineering*, 528–537, 1987.
11. R. Snodgrass, ''The temporal query language tquel,'' *ACM Transaction on Database Systems*, Vol. 12(2): 247–298, 1987.
12. A. Segev and H. Gundadhi. ''Event-join optimization in temporal relational databases,'' *Proceedings of the Fifth International Conference on Very Large Databases*, Amsterdam, 205–216, 1989.
13. H. Gunadhi and A. Segev. ''A framework for query optimization in temporal databases,'' *International Working Conference on Statistical and Scientific Database Management*, 131–147, 1989.
14. A. Segev, Join Processing and Optimization in Temporal Relational Databases. Temporal Databases: Theory, Design, and Implementation, Benjamin/Cummings Publishing Company, 356–387, 1993.
15. C. Leung and R. Muntz. Stream Processing: Temporal query Processing and Optimization. Temporal Databases: Theory, Design, and Implementation, Benjamin/Cummings Publishing Company, 329–355, 1993.
16. G. Langran. Time in Geographic Information Systems. Taylor & Francis, 1992.
17. R.T. Snodgrass. Developing Time-Oriented Database Applications in SQL. Morgan Kaufmann Publishers: San Francisco, CA, 2000.
18. M.D. Egenhofer and J. Herring. ''A mathematical framework for the definition of topological relationships,'' *Proceedings of the Fourth International Symposium on Spatial Data Handling, Zurich, Switzerland,* 803–813, 1990.
19. Open GIS Consortium, Abstract Specification, OpenGIS Project Documents 99–100 through 99–116, Wayland, Massachusetts, 1999.
20. P. Scarponcini, Spatial Data in Relational Databases. GIS-T'2000 Workshop Handouts, Minneapolis, MN, March 26, 2000.
21. A. Segev and Shoshani, A. ''Logical modeling of temporal database,'' *Proceedings of the ACM SIGMOD International Conference on Management of Data*, 454–466, May 1987.
22. A. Shoshani and K., Kawagoe ''Temporal Data Management,'' *Proceedings of the Twelfth International Conference on Very Large Databases, Kyoto, August,* 79–88, 1986.

Dr. Kurt is a professor of Civil, Environmental and Architectural Engineering at the University of Kansas. He has published widely in many areas of civil engineering including the application of GIS in Civil Engineering. His research interests include the development of efficient GIS algorithms to route and schedule vehicles in transportation and the support of data in the transportation field. He is a registered engineer in Kansas and Missouri.