

## 標高勾配付き道路トポロジーデータの構築

蒋 湧

### 1 道路と道路データ

道路とは、人や車が通過するために整備された通路を指す（広辞苑）。道路には地域の交通機能のみならず、都市の空間機能や市街地形成機能など、表 1 に示したような複数の機能を担うことで、多くの地域研究の対象になってきた。本稿は、道路の通行機能に着目し、道路交通事故の数値解析に必要な道路データ構造とその実装方法を論じる。具体的には、道路間の接続、道路沿線の標高勾配と道路周辺の交通事故を含め、複数の関連データを空間データベースに統合し、GIS（地理情報システム）の解析に適する道路データ構造の構築方法を提示する。

道路データとは、道路の位置、形状と属性、3つの基本要素を備えたデータセットを指す。例えば、国土交通省の「国土数値情報」ダウンロードサイトに公開している道路データにおいて（図 1）、「場所：GM\_Curve」には道路の位置と形状の情報があり、それによって道路が地図上に描ける。また、「道路種別」と「路線名」などは道路の属性としてデータセットに付与されている。

道路の基本要素だけで構成された道路データを道路の基礎データと呼ぶ。通常、国土数値情報などから入手した道路の基礎データをベースに、研究に必要な情報を属性に追加するとともに、自らの研究に特化したデータ構造を設け、

表 1 都市道路の機能

機能の区分		内 容	
①交通機能	通行機能	人や物資の移動の通行空間としての機能	
	沿道利用機能	沿道の土地利用のための出入、自動車の駐停車、貨物の積み降ろし等の沿道サービス機能	
②空間機能	都市環境機能	景観、日照、相隣等の都市環境保全のための機能	
	都市防災機能	避難・救援機能	災害発生時の避難通路や救援活動のための通路としての機能
		災害防止機能	火災等の拡大を遅延・防止するための空間機能
	収容空間	公共交通機関の導入空間機能	地下鉄、都市モノレール、新交通システム、路面電車、バス等の公共交通機関の導入のための空間
		供給処理・通信情報施設の空間	上水道、下水道、ガス、電気、電話、CATV、都市廃棄物処理管路等の都市における供給処理及び通信情報施設のための空間
		道路付属物のための空間	電話ボックス、電柱、交通信号、案内板、ストリートファニチャー等のための空間
③市街地形成機能	都市構造・土地利用の誘導形成	都市の骨格として都市の主軸を形成するとともに、その発展方向や土地利用の方向を規定する	
	街区形成機能	一定規模の宅地を区画する街区形成	
	生活空間	人々が集い、遊び、語らう日常生活のコミュニティ空間	

(出典：実務者ための新都市計画マニュアル)

<<featureType>> 道路	<<codeList>> 道路種別コード
+ 場所: GM_Curve	+ 高速道路 = 1
+ 道路種別コード: 道路種別コード	+ 一般国道 = 2
+ 路線名: CharacterString	+ 主要地方道 = 3
+ 線名: CharacterString [0..1]	+ 一般都道府県道 = 4
+ 通称: CharacterString [0..1]	+ 特例都道 = 5
	+ 市町村道 = 6
	+ 私道 = 7

図 1 道路のデータ構造 (出典：国土数値情報)

実装することが必要になる。本稿は、道路交通事故の数値解析を目的とし、道路間の接続属性、道路沿線の標高情報、研究対象の事故個票情報、3つの要素を入れた道路データ構造を設け、その実装について解説する。本稿では名古屋市中心部を対象に、表 2 のデータと QGIS + PostgreSQL の環境を用いて、道路データ構造を実装する。

## 標高勾配付き道路トポロジーデータの構築

表 2 使用する主なデータ

No	データ	出典
1	道路データ	ArcGIS Geo Suite 詳細図 2020、ESRI ジャパン
2	標高データ	国土交通省、基盤地図情報、数値標高モデル 5m
3	2009 年愛知県交通事故個票	愛知県警察本部

## 2 道路トポロジーデータの作成

本研究は、ESRI ジャパン社が提供する ArcGIS Geo Suite 詳細図 2020 の道路基礎データを利用する。ArcGIS Geo Suite の道路データには、道路 (road) と細道 (nroad) の 2 種類のデータがあり、筆者はそれらを 1 つの道路データセットにまとめ、図 2 のようなデータベース構造を実装した。テーブル



図 2 道路基礎データの属性と構造

表 3 道路の類別 (tb\_road\_type)

layercode	road_type	layercode	road_type
1	高速自動車国道	22	一般都道府県道 (有料区間)
2	高速自動車国道 (トンネル区間)	25	主要一般道
5	都市高速道路	26	主要一般道 (トンネル区間)
6	都市高速道路 (トンネル区間)	27	主要一般道 (有料区間)
9	一般国道	30	一般道路
10	一般国道 (トンネル区間)	31	一般道路 (トンネル区間)
11	一般国道 (有料区間)	32	一般道路 (有料区間)
14	主要地方道	35	細道路
15	主要地方道 (トンネル区間)	36	細道路 (トンネル区間)
16	主要地方道 (有料区間)	42	フェリー航
20	一般都道府県道	43	その他道路
21	一般都道府県道 (トンネル区間)		



図3 道路基礎データのマッピング結果

ル `stb_network_data` には、識別子 `id`、ジオメトリカラム `geom` と道路類別の識別子 `layercode`、3つの属性を持ち、道路の類別はテーブル `tb_road_type` を参照している。

表3は、テーブル `tb_road_type` にある道路の類別を示し、図3は道路基礎データのマッピング結果を示す。

## 2.1 PostGIS を用いた道路トポロジーデータの作成

通常、道路上の交通事故は、人々が何らかの目的で、出発地点から到達終点までの道路を通過する（自動車走行、あるいは徒歩）際に生じる。その通過路を「経路」と呼ぶ。経路は、利用者の目的（通勤、買い物など）と目標（始点・終点）などにより現れ、経路上の道路の性質（形状、車線、幅員など）や交通事情（交通量、走行速度）などは事故発生の原因になる。従って、経路は事故原因の解明に研究すべき空間対象である。経路とは、一般的に経由した複数道路を指す。経路の基本情報として、道路と道路の接続、つまり、前の道路の終点と次の道路の始点をつなぐ情報は欠かせない。しかし、テーブル `stb_`

## 標高勾配付き道路トポロジーデータの構築

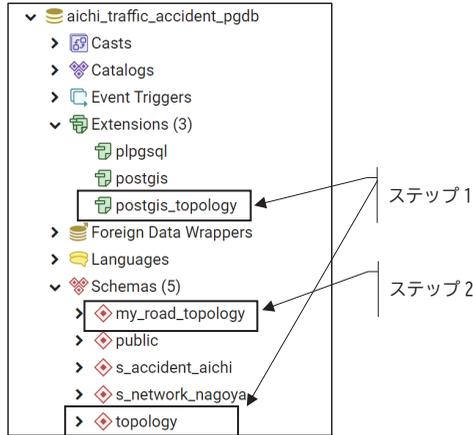


図 4 トポロジー関連のデータベース構造

network\_data の 3 つの属性には、経路に関わる情報は無い。道路のジオメトリ情報 geom に、道路の接続情報を加えると、道路のトポロジーデータになる。次は、テーブル stb\_network\_data にトポロジー情報を加える実装方法の概要を解説するが、詳細については文献「1」で参照できる。

本研究は、データベース PostgreSQL の PostGIS を用いたトポロジーを構築する。図 4 はトポロジー関連のデータベース構造を示す。トポロジーの実装は、図 4 と図 5 に示した 4 つのステップを踏んで行われる。ステップ 1 は、PostgreSQL の拡張 Extensions に postgis\_topology の拡張パッケージを追加する。これをおこなうと Schemas に topology というトポロジー管理のスキーマが現れる。次のステップでは、SQL 構文で本研究専用のトポロジースキーマ my\_road\_topology を新規作成する。ステップ 3 においては、SQL 構文でテーブル stb\_network\_data にトポロジーカラム topogeom を新規作成する (図 5 の右図)。最後のステップは「データ移入」である。ジオメトリ geom の道路位置情報を、道路の接続情報へ変換し、その結果を新規の topogeom に書き込む。「データ移入」作業は、PostGIS の SQL 構文で実行される。

情報	テーブル	プレビュー	クエリ(traffic_acci
	id	geom	layercode
1	1	LINestring	30
2	2	LINestring	30
3	3	LINestring	30
4	4	LINestring	30
5	5	LINestring	30
6	6	LINestring	30

情報	テーブル	プレビュー	クエリ(traffic_accident) ×
	id	geom	layercode topogeom
1	1	LINestring	30 (1,1,1,2)
2	2	LINestring	30 (1,1,2,2)
3	3	LINestring	30 (1,1,3,2)
4	4	LINestring	30 (1,1,4,2)
5	5	LINestring	30 (1,1,5,2)
6	6	LINestring	30 (1,1,6,2)

図5 データ移入前後の stb\_network\_data の属性表

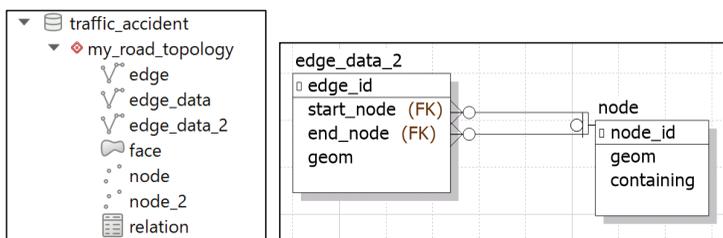


図6 my\_road\_topology スキーマ内の edge と node

図5に示したように、道路基本データはジオメトリ geom の空間位置情報のみ存在していたが (図5の左図)、トポロジー加える topogeom の接続情報が加えられている (図5の右図)。トポロジー構築の主な結果は my\_road\_topology スキーマに格納されている (図6)。テーブル edge は道路セグメント、node は道路セグメント間の接点、face は近隣の道路セグメントで囲まれるポリゴンを指す。テーブル edge\_data には、接点 node を経由した接続情報と face により隣接情報が記述されているが、筆者は edge\_data から接続情報だけを抽出し、新たに edge\_data\_2 を作成した (図6の右図)。

図7は接続情報の実例を示す。例えば edge\_id 3の start\_node と end\_node は、それぞれ5と6 (図7右図に下線を引く数値)。次の edge\_id 4は node 5から node 7へ、つまり、node 5で edge\_id 3と edge\_id 4がつながっている。図8は edge\_data\_2 と node のマッピング結果を示す。

## 標高勾配付き道路トポロジーデータの構築

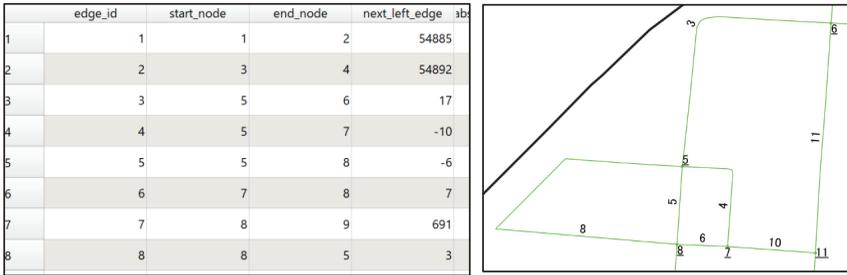


図7 edge と node による道路の接続情報

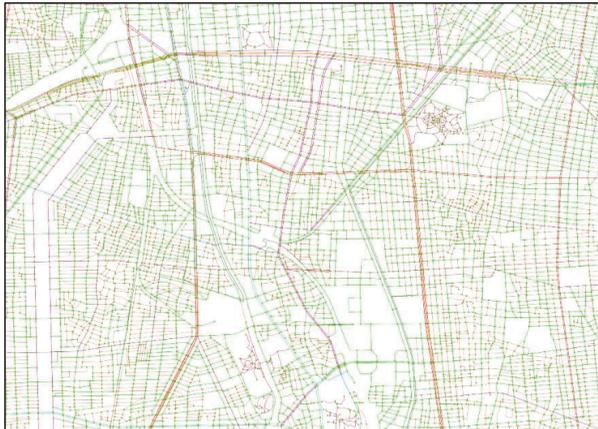


図8 edge\_data\_2 と node のマッピング結果

この節の最後に、トポロジー構築に関する注意すべき3点を述べる。  
 start\_node と end\_node は単なる edge 端点の名称であり、有向 edge の意味ではない。現時点高架道路と交差する一般道路の対応はできないので、手動で修正する必要がある。PC の計算負荷が大きい。本研究は名古屋市を対象にしているが、道路基礎データ stb\_network\_data には 66,462 件のデータがある。それをベースに構築した edge\_data と node の行数は、それぞれ 113,980 と 69,333 件にのぼる。筆者は、Intel (R) Core (TM) i7-7820HQ CPU

@2.90GHz、32GB メモリの PC で計算時間は約 1 時間 10 分を要した。

## 2.2 道路トポロジーデータに道路類別属性の追加

前節で完成した `edge_data` には、本来道路基礎データ `stb_network_data` が持つ道路の類別 `layercode` はない。今後の分析に備え、次は PostGIS の SQL 構文で `edge_data` に `layercode` の属性を追加する。

### SQL 構文 1

1	<code>select edge_id, ST_LineInterpolatePoint (geom, 0.50) as geom</code>
2	<code>into my_road_topology.stb_edge_middle_point</code>
3	<code>from my_road_topology.edge_data</code>
4	<code>order by edge_id</code>

### SQL 構文 2

1	<code>select a.edge_id, b.layercode</code>
2	<code>from my_road_topology.stb_edge_middle_point a, s_network_nagoya.stb_network_data b</code>
3	<code>where st_intersects (a.geom, b.geom)</code>
4	<code>order by a.edge_id</code>

### SQL 構文 1 と 2 の解説

SQL 構文 1 は `edge_data` のラインに中間点を作成する。スキーマ `my_road_topology` から `edge_data` を選び (行 3)、その中から `edge_id` と `geom` を選択し、PostGIS の関数 `ST_LineInterpolatePoint (geom, 0.50)` を用いて `edge_data` の中間点を求める (行 1)。その結果を、`edge_id` の順番で (行 4)、新規テーブル `stb_edge_middle_point` へ書き出す (行 2)。SQL 構文 2 は、`edge_data` の中間点と道路基礎データ `stb_network_data` の空間参照 (over layer) で、`edge_id` と `layercode` の関連を抽出する。スキーマ `my_road_topology` から `stb_edge_middle_point` を、スキーマ `s_network_nagoya` から `stb_network_data` を選択し、それぞれ `a` と `b` の別名で定義する (行 2)。次

## 標高勾配付き道路トポロジーデータの構築

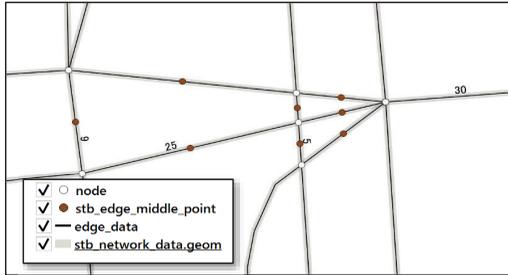


図9 egde\_data の中間点と stb\_network\_data のオーバーレイ

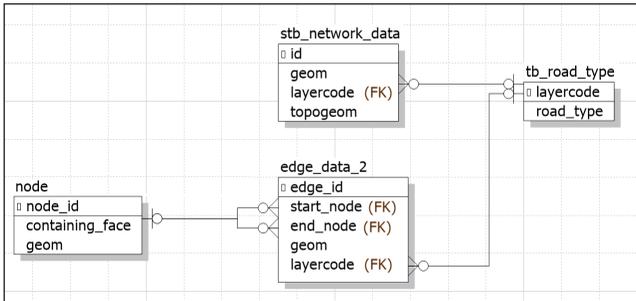


図10 道路トポロジーを入れたデータ構造

は、a と b が交差している条件で、関数 `st_intersects (a.geom, b.geom)` が成り立つ場合 (行3)、a からは `edge_id` を、b からは `layercode` を抽出し (行1)、その結果を `edge_id` の順で並べ替える。

図9 は `egde_data` の中間点と `stb_network_data` のオーバーレイの様子を示し、図10 は道路トポロジーを含めた道路データ構造の拡張を示す。

### 3 標高データの取り入れ

道路交通事故と道路沿線の標高勾配の関連性を調べるために、道路データ構造に標高の情報を取り入れる。標高は、国土交通省の基盤地図情報サイトで公

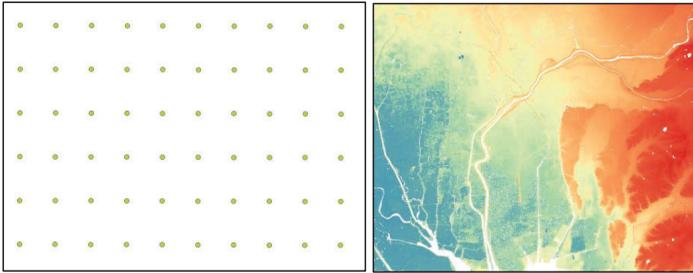


図 11 標高データ：左図 5m 間隔ポイント、右図ラスタ形式データ

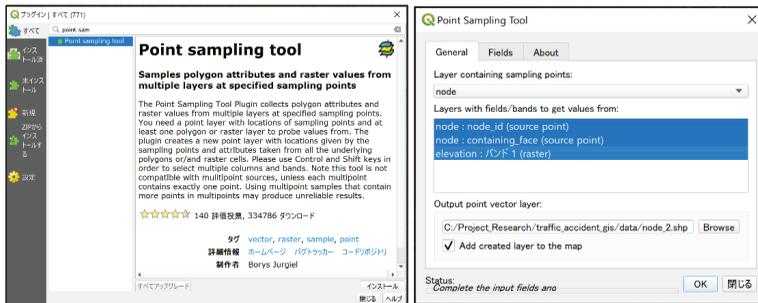


図 12 QGIS の Point sampling tool プラグインとその使い方

表されている数値標高モデル 5m のポイントデータを使用した (図 11 の左図)。計算負荷を考慮し、本稿は 4 つの 5 次メッシュ範囲のデータ (523666、523667、523656、523657) を取り入れ、5m 間隔の標高ポイント数は 12,572,400 に達した。それを QGIS の計算ツール「ベクタのラスタ化 (rasterize)」でラスタ形式へ変換した (図 11 の右図)。

次に、QGIS のプラグインツール Point sampling tool (図 12 の左図) を使って、トポロジーの node データと標高のラスタデータ elevation をオーバーレイし、node ごとに標高情報を取得する。その結果を node\_2 に保存すると、図 13 の標高情報を取り入れたデータ構造になった。

トポロジー node に標高値を入れることで、edge 両端点の標高値を用いて経

## 標高勾配付き道路トポロジーデータの構築

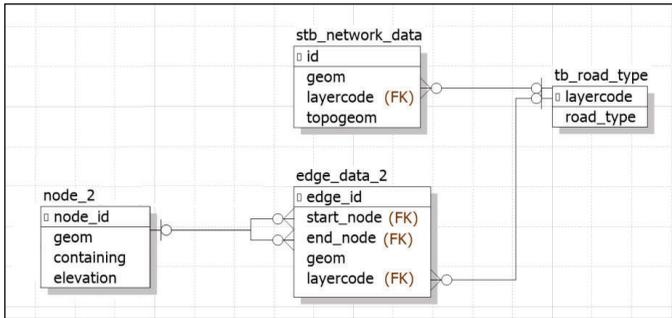


図 13 標高と道路トポロジーを入れたデータ構造

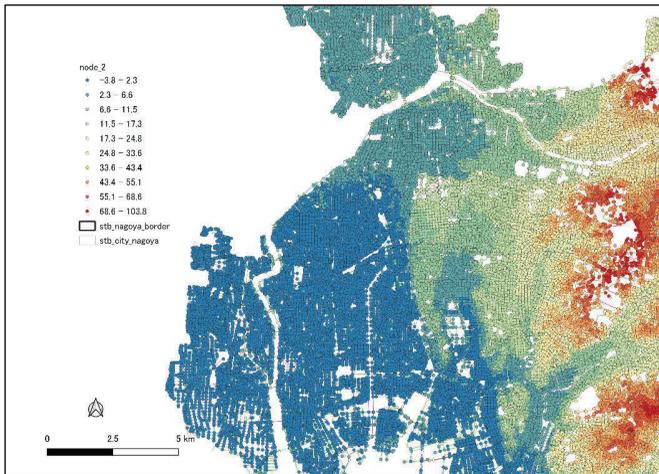


図 14 トポロジー node の標高分布

路の標高勾配が計算できるようになった。図 14 はトポロジー node の標高値で描いた道路標高の分布図を示す。



図 15 2009 年愛知県交通事故統計情報のマッピング

#### 4 交通事故個票データとの関連付け

次は 2009 年愛知県道路交通事故の統計データ（図 15）を用いて、下記の SQL 構文 3 と 4 で事故発生地点と edge\_data\_2 の最短直線を求め、両者を関連付ける。

SQL 構文 3

```

1  select a.id, b.edge_id, st_shortestline (a.geom, b.geom) as geom,
   st_length (st_shortestline (a.geom, b.geom)) as length
2  into s_network_nagoya.stb_accident_point_to_edge_5m_all
3  from s_network_nagoya.stb_accident_2009_nagoya a,
   my_road_topology.edge_data_2 b
4  where st_dwithin (a.geom, b.geom, 5)
5  order by a.id

```

SQL 構文 4

```

1  select a.id, a.edge_id, a.geom, a.length
2  into s_network_nagoya.stb_line_acci_to_edge_2009
3  from s_network_nagoya.stb_accident_point_to_edge_5m_all as a

```

## 標高勾配付き道路トポロジーデータの構築

4	inner join (select id, min(length) as min_length from s_network_nagoya.stb_accident_point_to_edge_5m_all group by id) as b
5	on a.id=b.id and a.length = b.min_length
6	order by a.id

### SQL 構文 3 と 4 の解説

まず、SQL 構文 3 は道路 edge\_data\_2 周辺に 5m のバッファを発生させる。そのバッファ範囲内に含まれる事故発生地点（ポイント）と各々の道路 edge\_data\_2 の間の最短直線を求める。次の SQL 構文 4 は、前の結果から、事故発生地点にとって、最寄りの edge\_data\_2 との最短直線を抽出し、事故 id と最寄り edge\_id を関連付ける。同じ計算を道路周辺 15m バッファまで拡大したところ、全ての事故 id と最寄りの edge\_id に関連付けることになった。

SQL 構文 3 では、まず、交通事故データ stb\_accident\_2009\_nagoya を a に、道路 edge\_data\_2 を b に定義する（行 3）。事故 a が道路 b 周辺 5m のバッファに含まれる条件の下で、つまり、関数 st\_dwithin (a.geom, b.geom, 5) が成り立つの条件で（行 4）、事故 id、道路 edge\_id、事故発生地と道路の間の最短直線 st\_shortestline (a.geom, b.geom) とその長さを抽出する（行 1）。その結果を事故 id の順で中間結果 stb\_accident\_point\_to\_edge\_5m\_all に格納する。

SQL 構文 4 は、複合 SQL 構文を持って中間結果 stb\_accident\_point\_to\_edge\_5m\_all から、各々の事故 id にとって最寄りの edge\_id との最短直線を抽出する。行 1 と行 3 は stb\_accident\_point\_to\_edge\_5m\_all から全項目を抽出し、その結果を a とする。一方、行 4 の複合 SQL 構文は、同じ中間結果から事故 id と事故 id ごとの直線長さの最小値を抽出し、その結果を b とする。行 5 の条件のもとで、行 1 の抽出を行い、その結果を事故 id の順に（行 6）、最終結果 stb\_line\_acci\_to\_edge\_2009 に書き出す。

図 16 は、SQL 構文 3 と構文 4 の実行結果を示し、図 17 はそれに伴う道路

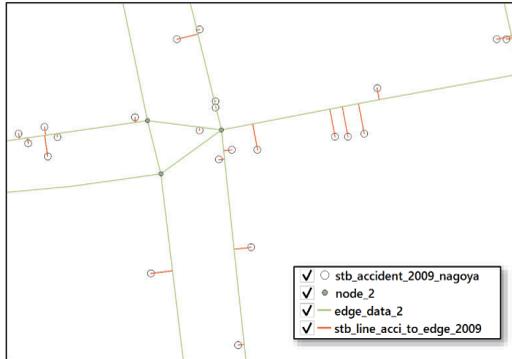


図 16 道路交通事故と道路 edge の関連付け

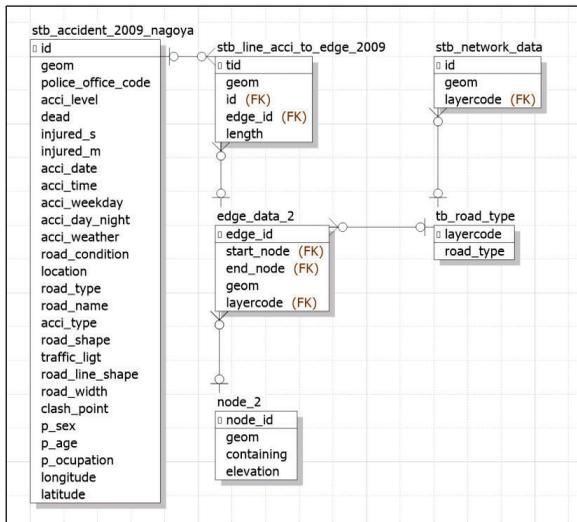


図 17 道路トポロジー、道路類別、標高と事故を含めたデータ構造

データ構造の拡張を示す。このデータ構造を用いて、交通事故を道路のトポロジー構造、沿線の標高勾配、道路の類別など多角的な視点から分析が可能になる。

## 5 応用事例について

本節では、提示した道路のデータ構造を用いた研究事例を解説する。図 18 の背景には 2009 年の交通事故の発生件数に基づいたカーネル密度の分布を表し、可視化された事故多発交差点から研究対象を特定する。また、研究対象の交差点から、QGIS のネットワーク解析のサービスエリアの機能を使って、対象交差点を通過し、100m 範囲で到達可能な経路を抽出することができる。これらの経路は事故分析の対象になる。このように提案した道路データ構造を利用し、道路の性質、標高勾配、事故の属性を含めた統合的な数値解析をすることができる。これまでの広範囲交通事故を対象に、メッシュ単位を用いた研究と比べ、本研究は経路を対象とし、道路データの空間トポロジー性質を活かしたより局所的な視点をもつことが特徴であると言えよう。

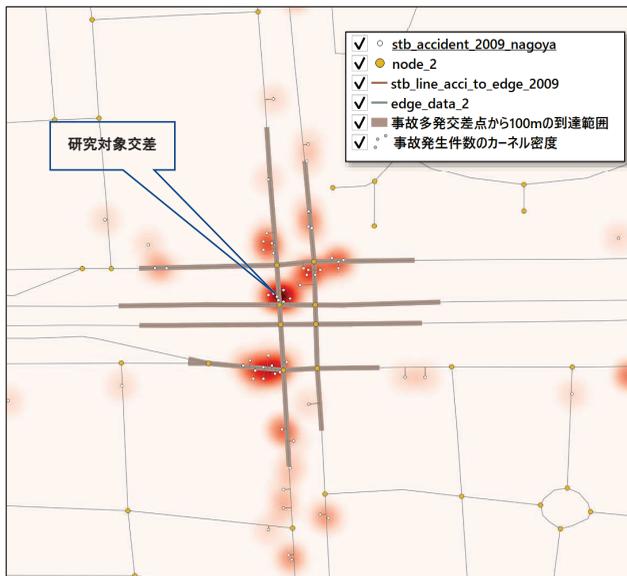


図 18 道路データ構造の応用事例

## まとめ

本稿は、道路交通事故を解析するための道路データ構造を提案した。通常の道路基礎データと PostGIS Topology パッケージを使って、道路のジオメトリ geom 情報からトポロジー topogeom 情報を引き出すことができる。道路のトポロジーデータは、edge と node で構成され、node には道路沿線の標高値を付与させ、edge には周辺で発生した交通事故を関連付ける。提案した道路のデータ構造は、主に経路を対象とした交通事故の分析に使われる。例えば、通勤時間帯の主な通勤経路、主な産業集積地区間の物流経路、大型ショッピングセンターとそれをつなぐ周辺の経路、交通事故多発の交差点とそれをつなぐ周辺の経路など、様々な地域社会の視点から交通事故の特徴を分析することが考えられる。

また、提案された道路のデータ構造は、道路交通事故の研究に限らず、津波避難行動、障害者車イスの走行ルート、競輪・ランニングの走るコースなどを含め、様々な研究分野への応用が期待される。

## 謝辞

本研究は、2020 年度愛知大学経営総合科学研究所の研究プロジェクト「人工知能と地理情報システムを用いた愛知県の死亡事故データ分析」の研究補助を受けた。

## 参考文献

- [1] 蔣湧、湯川晴敏、駒木伸比古、飯塚隆藤、村山徹、小川勇樹、「地域研究のための空間データ分析入門」、古今書院、2019.