

FOSS4G を利用した水害時避難経路探索システムの構築

嘉山陽一

Introduction of flood evacuation route search system using FOSS4G

Yoichi KAYAMA

Abstract: In recent years, we can obtain accurate terrain data in laser measurement. Using such LiDAR data fine mesh units available data was calculated assuming a flood of large-mesh units. There are many powerful tools in open source software products for handling digital spatial data. Using fine mesh unit data and open source software(FOSS4G), I made a system for searching safe route to a shelter, assuming flood occur. I made a QGIS plug-in for this system to find capable route from a point on the map to shelter not using the submerged roads ,assuming flood.

Keywords: オープンソース, レーザ計測, 経路探索, pgRouting

1. はじめに

国土交通省,都道府県では平成 13 年から洪水予報河川及び水位周知河川において水防法に基づく浸水想定区域図を公表している。

また浸水想定区域を含む市町村長は防災計画を住民に周知するための印刷物の配布を行わなければならない(水防法第 15 条 4) .この印刷物として浸水想定区域図に避難場所や洪水時の避難に必要な事項を記載した洪水ハザードマップが作成されることが多い。

このように浸水想定区域データは水害対策の基礎データといえるものである.ただし想定データの氾濫シミュレーションは 250m メッシュ,標高は 50m メッシュを利用して計算が行われている(浸水想定区域図作成マニュアル国土交通省河川局治水課) .

近年,航空機レーザによる地形計測で高密度の標高データを作成し微細な地形をデジタルデータで表現できるようになった.50m メッシュ標高で作成していた浸水想定区域データをレーザ計測 2m メッシュの標高データで再構成すると従来に比べて微細な地形単位の想定浸水状況を把握することができ.

このような細かな浸水想定区域図データの新たな利活用方法を検討するプロトタイプとして本システムを作成した。

今回は想定浸水データと道路ネットワークデータを GIS システム上でオーバーレイして「浸水していない」道路のみを利用した経路探索を実行するための機能実装方法を考察した。

本機能実装を行うために pgRouting, PostGIS, QuantumGIS 等の複数の空間情報に関するオープンソースソフトウェア(Free and Open source software for geospatial 略称 FOSS4G)を利用し,それらの有用性を調査した。

嘉山陽一 〒350-1165 埼玉県川越市南台 3 丁目 1 番地 1

朝日航洋株式会社

Phone: 049-244-4032

E-mail: youichi-kayama@aeroasahi.co.jp

2. 利用データについて

本システムでは1次データとして以下のデータを利用した。

- ・想定浸水深データ（ラスタデータ）
- ・避難所位置データ（ベクタ ポイントデータ）
- ・道路ネットワーク（ベクタ ラインデータ）
- ・避難所区域データ（ベクタ ポリゴンデータ）
- ・背景図データ（ベクタまたはラスタ、航空写真、レーザ計測データ、基盤地図情報等）

想定浸水深データは想定破堤地点別に作成されている。本システムで利用する時は各地点別の浸水深を数値として持つ GeoTiff 形式のラスタデータ（Gとして準備されている。この GeoTiff ファイルは2m単位の区画を1ピクセルとして、区画単位の想定浸水深数値（m単位）をピクセルの値として保持している。このファイルは各想定破堤地点毎に10分単位の破堤後経過時間別に作成されている。想定破堤地点（河川名、左右、キロポスト）と破堤後経過時間（10分単位）を指定するとその地域の想定浸水状況データを特定することができる。

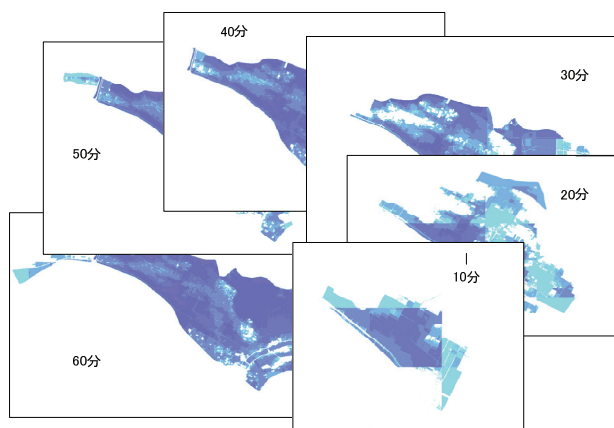


図-1 破堤後経過時間10分単位に作成された想定浸水ラスタデータ（浸水深データはダミー）

3. pgRouting を利用した経路探索

3.1 pgRouting とは

pgRouting とはスイスとフランスに拠点を置く CampToCamp 社 (<http://www.camptocamp.com/>) で開発された pgDijkstra という経路探索システムを日本のオークニー社で機能追加、改良したオープンソースの経路探索システムである。

オープンソースのデータベース管理システム PostgreSQL は PostGIS という拡張パッケージをインストールすると空間データを利用できる。pgRouting は PostGIS の機能を利用して経路探索を行うための関数群として提供されている。よって pgRouting の経路探索機能は PostgreSQL/PostGIS で構築した空間データベースに対して実行できる。pgRouting では Shortest Path Dijkstra, Shortest Path A*, Shortest Path Shooting Star を利用した経路探索ができる。さらに pgRouting では巡回セールスマン問題と到達圏検索を実行することができる。

3.2 pgRouting を利用した経路探索

pgRouting では PostGIS/PostgreSQL のライン、またはマルチラインジオメトリを持つテーブルを利用して経路検索を行う。この時各ラインジオメトリレコードはあらかじめジオメトリ間の位相構造を作成して結果を各レコード内カラムに保持している必要がある。

pgRouting の関数を含んだ PostgreSQL の SQL を発行することで pgRouting の最短経路検索を実行できる。この時、関数の引数として経路の開始点と終了点のノード番号を指定する必要がある。

経路検索の結果はジオメトリを含むレコードの集合として返される。pgRouting を利用した検索は SQL の検索条件として記述できる条件であるならば組み合わせることができる。

徒歩による水害時の避難経路として経路検索を行う場合は道路の水没条件以外に通行不能な条件が

ある道路を経路から除外する必要がある。たとえば鉄道や幹線道路等を潜り抜けるアンダーパスの箇所は水害ハザードマップでは通行すべきではない箇所としてマークされている。道路ネットワークに含まれる高速道路を徒歩の避難経路として利用すべきかどうか道路の構造等によって異なると思われる。このような場合 SQL で記述できる検索条件を作成できれば pgRouting を利用した経路検索に組み合わせることができる。

4. 空間条件付き経路探索の実装

各道路区間データは PostGIS が管理するジオメトリカラムを保有する PostgreSQL のレコードとして管理されている。各道路が浸水しているかどうかの判定は道路区間のラインストリングまたはマルチラインストリングジオメトリとラスタデータで表現されている浸水深の空間的な位置関係の判定で表現できる。

ただし PostGIS1.5 の通常インストール版ではラスタデータを扱うことができない。よってラスタで表現されている時間別浸水領域データを加工して浸水領域を表すポリゴンとして PostGIS/PostgreSQL データベースに格納した。

データの加工にはオープンソース GIS である GRASS6.4 を利用した。加工の手順を図-2 に示す。

作成した時間別浸水領域ポリゴンデータは経過時間をレコードフィールドに格納したうえで破堤箇所別に PostGIS/PostgreSQL のテーブルに格納した。この処理により道路区間のラインストリングと浸水領域ポリゴンの位置関係判定関数を利用して道路の浸水状況を判定することが可能になる。破堤後経過時間はフィールド minute に分単位で格納されている。例えば破堤後 30 分で浸水していない道路を抽出する SQL は次のとおりである。

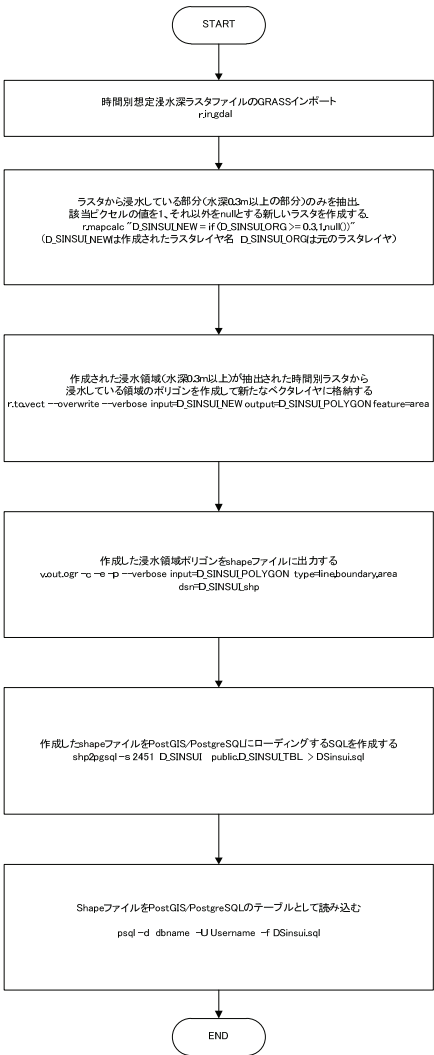


図-2 時間別浸水深ラスタデータの加工手順

```
SELECT *
FROM ways as a -- ways 道路区間テーブル
where a.gid not in(
    select b.gid from ways as b,
    suishin_polygon_all as c -- 浸水深 polygon
    where c.minute = 30 -- 破堤後 30 分経過
    and ST_Intersects(b.the_geom,
    c.the_geom)) -- ジオメトリの交差判定
```

ここでテーブル名 `ways` が道路区間ラインストリングのジオメトリをカラム `the_geom` で持つテーブル、テーブル名 `suishin_polygon_all` が浸水領域ポリゴンジオメトリをカラム `the_geom` 破堤後経過時間をカラム `minute` に保有するテーブルである。また `ST_Intersects()` はジオメトリ間の位置関係を判定する PostGIS の関数である。この関数は2つのジオメトリ同士で重なる部分がある場合真の値を返す。この SQL を利用して pgRouting の Shortest Path A*アルゴリズムで経路検索を行う SQL は以下のとおりである。

```
SELECT vertex_id , edge_id , cost
FROM shortest_path_astar( -- A* 関数
'SELECT a.gid as id, source, target, cost as cost,
  x1, y1, x2, y2
FROM ways as a where a.gid not in( select b.gid from
ways as b, suishin_polygon_all as c
where c.minute = 30 -- 破堤後 30 分経過
and ST_Intersects(b.the_geom,
c.the_geom))' -- ジオメトリの交差判定
, 4799, 3962, false, false)
/* source id, target id, directed, has reverse cost
*/
```

ここで `shortest_path_astar()` が Shortest Path A*アルゴリズムで経路検索を行う pgRouting の関数である。一番下の行の数値“4799,3962”は経路検索を行う道路ネットワークの始点と終点のノード番号である。この SQL を発行すると経路を構成するレコードの `id` とコスト値のリストが結果として返される。

5. QuantumGIS を使った機能統合

以上の方法で pgRouting, PostGIS/PostgreSQL 上で浸水した道路を避けた経路検索を実行することができる。ただしこの場合検索は PostgreSQL のコマン

ドを発行して、検索結果の文字列がキャラクター端末に表示されるのみである。そこで検索条件の指定と検索結果を地図上に表示するためにデスクトップ GIS から検索を行い、結果を表示するシステムを作成した。

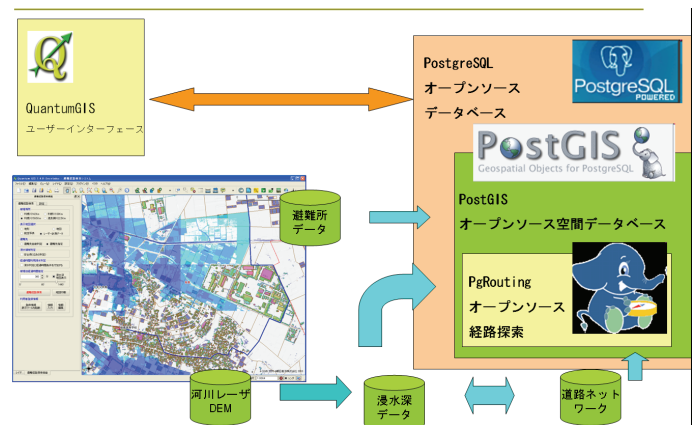


図-3 システム構成

QuantumGIS (QGIS) とはオープンソースの高機能マルチプラットフォームデスクトップ GIS である。QuantumGIS は2002年から開発が開始されているが正式リリース版である Version1.0 がリリースされたのは2009年の1月である。2010年8月には Ver1.5 がリリースされた。

Quantum GIS の基本機能は以下のとおりである。

- ・ データの表示
- ・ データの検索と地図の作成
- ・ データの作成編集、管理と変換
- ・ データ解析
- ・ プラグインを使った機能拡張
- ・ インターネット上の地図データの表示
- ・ インターネット公開用地図の作成

QuantumGIS では空間データライブラリ GDAL/OGR でサポートしている多数のラスタ、ベクタ形式データをレイヤとして利用できる。さらに

QuantumGIS では WMS,WFS,PostGIS,Delimited Text, Spatialite 等の様々な空間データを利用できる.

また QuantumGIS は外部プラグインの追加することでユーザー独自の機能を追加できる.外部プラグインは python または C++で記述することができる.今回は python プラグインとして避難経路検索機能を実装した.QuantumGIS のプラグインは Qt フレームワークで提供されている GUI フレームワークでユーザーインターフェースを記述できる.今回は図-4 のユーザーインターフェースを作成した.



図-4 避難経路検索システム操作パネル

図-4 のパネルで破堤箇所と破堤後経過時間,その他オプションを指定した後に,地図上で始点と避難所を指定して経路検索を実行する.処理が開始されると指定したパラメータで前記の SQL が発行される.この場合検索結果の経路が図-5 のように地図上に表示される.

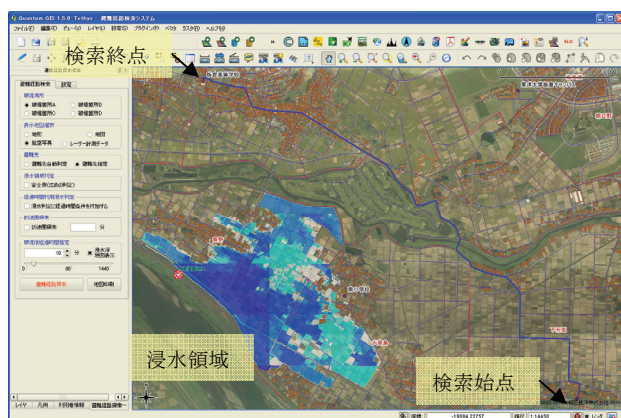


図-5 破堤後 10 分避難経路検索結果(浸水深はダミー)

同じ場所で破堤後経過時間を変更すると,浸水している領域が変化する.その状態で同一の経路検索を行う.今回は浸水していなかった道路でも今回の検索で浸水している場合はルートから除外した形で経路検索が行われていることが図-6 でわかる.

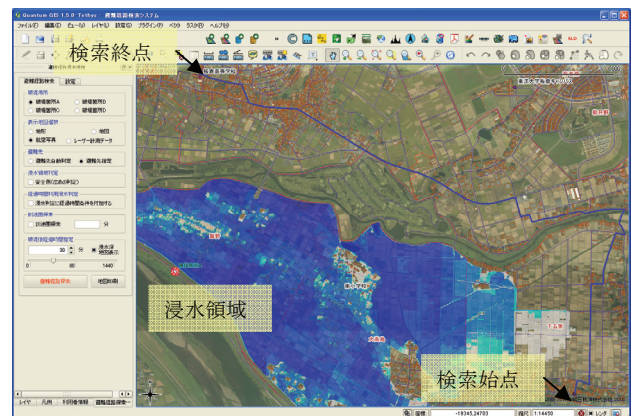


図-6 破堤後 30 分避難経路検索結果 (浸水深はダミー)

またある地点から指定時間内に避難できるルートを全て検索する到達圏検索を実行できる.この場合あらかじめ移動の想定時速を数値で定義しておくものとする.この検索においても浸水している道路は通らない.図7は指定地点から 30 分で到達できる範囲を検索したものである.

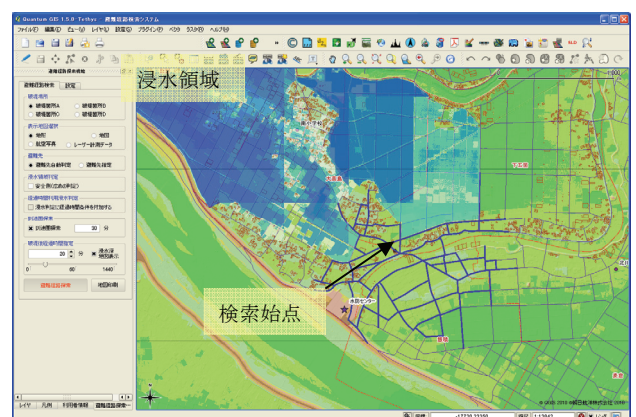


図-7 30 分で到達可能な範囲 (浸水深はダミー)

6. まとめ

レーザ計測データで精密化した想定浸水深データは細かい地物単位の浸水判定に利用できる。課題としてはデータの事前加工が必要である（道路ネットワークのトポロジ作成、浸水深データの事前加工）ためデータ更新時にデータ再構成作業に時間がかかることがあげられる。道路ネットワークについては最新のデータでなければ経路探索を行う意義が低下するので道路ネットワークデータ更新の仕組みが重要である。

今回利用した FOSS4G プロダクトは多彩な機能を持ち、組み合わせを行うことで高機能システムを短工期で作成することができた。近年空間情報に関するオープンソースシステムの機能向上は著しいものがある。FOSS4G コミュニティの世界的調整機関である The Open Source Geospatial Foundation の活動によってワールドワイドで FOSS4G の分散開発や利用が促進されている。国内外の FOSS4G コミュニティと開発・利用のための情報共有を盛んにすることでさらに有用な FOSS4G プロダクトができるであろう。

謝辞

本研究を「国土交通省関東地方整備局管内氾濫解析基礎資料作成業務」に利用し、関係する方々の指導・助言・協力をいただいた。QuantumGIS, GRASS は文部科学省宇宙利用促進調整委託費の「FOSS4G を利用した衛星データの利用のためのオープン・リソースの構築」の日本語化成果を利用した。pgRouting, PostGIS, PostgreSQL, GRASS, QuantumGIS は FOSS4G コミュニティが作成・維持している成果を利用した。関係する官庁、企業、コミュニティの方々に感謝します。

参考文献

- 国土交通省：浸水想定区域の指定・公表状況について, <http://www.mlit.go.jp/river/bousai/main/saigai/tisiki/syozaiti/sinsui/sinsui-ref1.html>.
- 国土交通省：浸水想定区域図・洪水ハザードマップ, <http://www.mlit.go.jp/river/bousai/main/saigai/tisiki/syozaiti/index.html>.
- 国土交通省 河川局治水課：浸水想定区域図作成マニュアル, H17, <http://www.mlit.go.jp/river/bousai/main/saigai/tisiki/syozaiti/index.html>.
- pgRouting Documentation, <http://pgrouting.postlbs.org/wiki/pgRoutingDocs>.
- Claude Philipona, Daniel Kastl, Frédéric Junod, Anton Patrushev, Web-based Routing: An Introduction to pgRouting with OpenLayers, <http://pgrouting.postlbs.org/wiki/WorkshopFOSS4G2007>, 2007.
- PostGIS 1.5.1 Manual, <http://postgis.refractor.net/docs/index.html>, 2010.
- Markus Neteler and Helena Mitsova, Open Source GIS A GRASS GIS Approach, Third Edition, 2008.
- QuantumGIS Coding and Compilation Guide Version 1.4 ‘Enceladus’, http://download.osgeo.org/qgis/doc/manual/qgis-1.4.0_coding-compilation_guide_en.pdf, 2010.
- FOSS4G を活用した衛星データ利用のためのオープン・リソースの構築, <http://www.osgeo.jp/foss4g-mext/>, 2010.
- Qt Online Reference Documentation, <http://doc.qt.nokia.com/>.