九州大学学術情報リポジトリ
Kyushu University Institutional Repository

数値地図データセットからの情報抽出および汎用地図ツールへの転用

松島，啓二
九州大学応用力学研究所技術室

石井，大輔
九州大学応用力学研究所技術室

https://hdl.handle.net/2324/14014

出版情報：九州大学応用力学研究所技術職員技術レポート. 9, pp.1-12, 2008-03. Research Institute for Applied Mechanics, Kyushu University
バージョン：
権利関係：

数値地図データセットからの情報抽出

および汎用地図ツールへの転用

九州大学応用力学研究所　技術室
松島　啓二　・　石井　大輔

１．はじめに

国土地理院が有償提供している数値地図データセットとして、「数値地図 25000（行政界・海岸線）」および「数値地図 50m メッシュ（標高）」がある。前者は、日本全国における行政界・海岸線の緯度・経度の数値データ、後者は、約 50m 四方毎に区切った日本地図の各メッシュ中央における標高の数値データで構成され、両者ともに当該データを読み込んで画像閲覧するための専用ビューアが付属されている。

これらの専用ビューアは描画および距離・面積計測等の機能があるものの、あくまでもビューアなので画像編集や文字挿入などはできない。また、元となるデータの書式が独特であるので、これらの専用ビューアでないと描画ができず、画像編集や文字挿入といった処理も簡単には行えない。換言すれば、他の処理が行えるプログラムへ当該データセットをそのまま転用することが困難である。

そこで、今回は「数値地図 25000（行政界・海岸線）」および「数値地図 50m メッシュ（標高）」の各データセットから緯度・経度・標高（数値地図 50m メッシュ（標高）のみ）の数値データを抽出し、他のプログラムへ転用できるように、書式を簡素化して出力する各変換プログラムを作成したので、ここに報告する。

２．数値地図 25000（行政界・海岸線）

２．１．「数値地図 25000（行政界・海岸線）」について

「数値地図 25000（行政界・海岸線）」は、2 万 5 千分 1 地形図の都道府県・市区町村等の行政界や海岸線の数値データ（以下、海岸基本データと略記）、および専用ビューアから構成される。

海岸基本データは、以下に分類される情報に基づいてベクトル形式で数値化されたものである。

（１）行政区域

行政界・海岸線で分節された一つの面のみで構成された最小単位の区域の情報である。その面の ID・名前、分節する行政界・海岸線の ID、代表点の ID・緯度・経

度が主な内容である。

（２）行政界

都道府県、支庁、都市区町村の行政的な境界を表す線の情報である。その線の ID・種類、線の始めと終わりの節点の ID、その線を構成する点の緯度・経度が主な内容である。

（３）海岸線

陸部と海部の境界を表す線の情報である。その線の ID・種類、始めと終わりの節点の ID、その線を構成する点の緯度・経度が主な内容である。

（４）行政界節点

行政界・海岸線の端点に位置し、他の行政界・海岸線と接続する点の情報である。その点の ID・種類、緯度・経度が主な内容である。

なお、線 ID における最初の 4 文字から線の種類を判別できる。線の種類と線 ID における最初の 4 文字の対応関係は、以下のとおりである。

都道府県界	AL01
支庁界	AL02
郡市または東京都特別区界	AL03
町村または指定都市区界	AL04
海岸線	CL05
湖沼の水涯線	WL02

「数値地図 25000（行政界・海岸線）」に付属の専用ビューアを用いて描画した大阪湾周辺の地図を図 1 に示す。また、海岸基本データの一例として、上記（３）における海岸線情報の一部を図 2 示す。図 2 から、線 ID（CL0500000008）・種類（海岸線）、始節点 ID（pAN00000001）、線を構成する点の緯度・経度（34.312021　135.093285 など）が記述されているのが分かる。

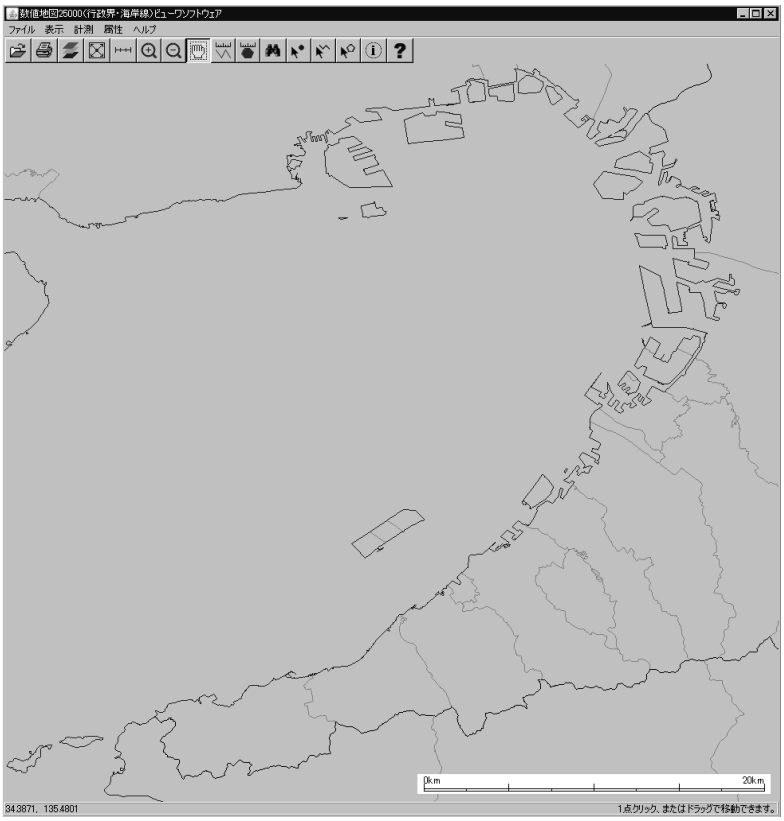

図１．「数値地図 25000（行政界・海岸線）」に付属の専用ビューアを用いた大阪湾周辺の地図

```
<海岸線 id="CL0500000008">↵                                  ← 線のIDと種類
^   <線 idref="cCL00000008" />↵
^   <辺 idref="eCL00000008" />↵
</海岸線>↵
<jps:GM_Curve id="cCL00000008">↵
^   <jps:GM_Primitive.proxy idref="cCL00000008" />↵
^   <jps:GM_Primitive.proxy idref="_cCL00000008" />↵
^   <jps:GM_OrientablePrimitive.orientation>+</jps:GM_OrientablePrimitive.orientation>↵
^   <jps:GM_OrientablePrimitive.primitive idref="cCL00000008" />↵
^   <jps:GM_Curve.segment>↵
^   ^   <jps:GM_LineString>↵
^   ^   ^   <jps:GM_CurveSegment.interpolation>linear</jps:GM_CurveSegment.interpolation>↵
^   ^   ^   <jps:GM_LineString.controlPoint>↵
^   ^   ^   ^   <jps:GM_PointArray>↵
^   ^   ^   ^   ^   <GM_PointArray.column>↵
^   ^   ^   ^   ^   ^   <jps:GM_Position.indirect>↵
^   ^   ^   ^   ^   ^   ^   <GM_PointRef.point idref="pAN00000001" />↵   ← 始節点ID
^   ^   ^   ^   ^   ^   </jps:GM_Position.indirect>↵
^   ^   ^   ^   ^   </GM_PointArray.column>↵
^   ^   ^   ^   ^   <GM_PointArray.column>↵
^   ^   ^   ^   ^   ^   <jps:GM_Position.direct>↵
^   ^   ^   ^   ^   ^   ^   <DirectPosition.coordinate>34.312021 135.093285</DirectPosition.coordinate>↵
^   ^   ^   ^   ^   ^   </jps:GM_Position.direct>↵
^   ^   ^   ^   ^   </GM_PointArray.column>↵                          ← 点の緯度・経度
^   ^   ^   ^   ^   <GM_PointArray.column>↵
^   ^   ^   ^   ^   ^   <jps:GM_Position.direct>↵
^   ^   ^   ^   ^   ^   ^   <DirectPosition.coordinate>34.311900 135.093449</DirectPosition.coordinate>↵
^   ^   ^   ^   ^   ^   </jps:GM_Position.direct>↵
^   ^   ^   ^   ^   </GM_PointArray.column>↵
^   ^   ^   ^   ^   <GM_PointArray.column>↵
^   ^   ^   ^   ^   ^   <jps:GM_Position.direct>↵
^   ^   ^   ^   ^   ^   ^   <DirectPosition.coordinate>34.312021 135.093618</DirectPosition.coordinate>↵
^   ^   ^   ^   ^   ^   </jps:GM_Position.direct>↵
^   ^   ^   ^   ^   </GM_PointArray.column>↵
^   ^   ^   ^   ^   <GM_PointArray.column>↵
^   ^   ^   ^   ^   ^   <jps:GM_Position.direct>↵
^   ^   ^   ^   ^   ^   ^   <DirectPosition.coordinate>34.312266 135.093881</DirectPosition.coordinate>↵
^   ^   ^   ^   ^   ^   </jps:GM_Position.direct>↵
^   ^   ^   ^   ^   </GM_PointArray.column>↵
```

図２．「数値地図 25000（行政界・海岸線）」データファイルにおける書式の一例

２．２．行政界・海岸線データ変換プログラムの作成

今回作成した変換プログラムは、「数値地図 25000（行政界・海岸線）」の海岸基本データから、ユーザの選択した種類（都道府県界、支庁界、郡市または東京都特別区界、町村または指定都市区界、海岸線、湖沼の水涯線のいずれか）の線 ID とその線を構成する点の緯度・経度を抽出し、ファイルに出力するものである。

例えば、ユーザの選択が海岸線なら、図 2 のような ID に“CL05”を含む線の情報を見つけ、その ID と点の緯度・経度の数値を読み取っていき、ファイルに出力する。

出力ファイルの書式は、ある線の ID がある 1 行に記述され、次の行から、その線を構成する点の緯度・経度データが 1 行につき 1 点ずつ順に記述され、全ての点が記述されると、次の行から、次の線データが記述されるという簡素なものである。図 3 は大阪府の海岸基本データから海岸線を選択した場合における出力の一例（ID=CL0500000008 の序盤）である。図 2 に記述されている緯度・経度（最初の 4 点）が正しく出力されているのが分かる。

```
>#ID=CL0500000008
34.312021,  135.093285
34.311900,  135.093449
34.312021,  135.093618
34.312266,  135.093881
34.312211,  135.093995
34.312357,  135.094093
34.312870,  135.094438
34.312968,  135.094356
34.313521,  135.094799
34.313821,  135.094761
34.313986,  135.094684
34.314203,  135.094400
```

図３．行政界・海岸線データ変換プログラムによる出力結果の一例

２．３．汎用地図ツールを用いた描画

今回、描画するための汎用地図ツールに GMT（The Generic Mapping Tools）を用いた。GMT は、ハワイ大学で開発され、ライセンスフリーで提供されているツールであり、地図を描画できる（パッケージ内に全球の地図データセットを含む）だけでなく、2 次元/3 次元図や散布図など、様々な図をユーザが自由に作成できる（専用言語でスクリプトを構築し実行する必要あり）。

図 4 に、GMT パッケージ内の大阪湾周辺の地図データを GMT で描画した結果を示す。同図は、外枠、目盛線、緯度・経度の軸ラベルを描き、座標内を灰色で塗りつぶし、その上に陸部を白色で重ねて描画している（灰色が残っている部分が海部となる）。

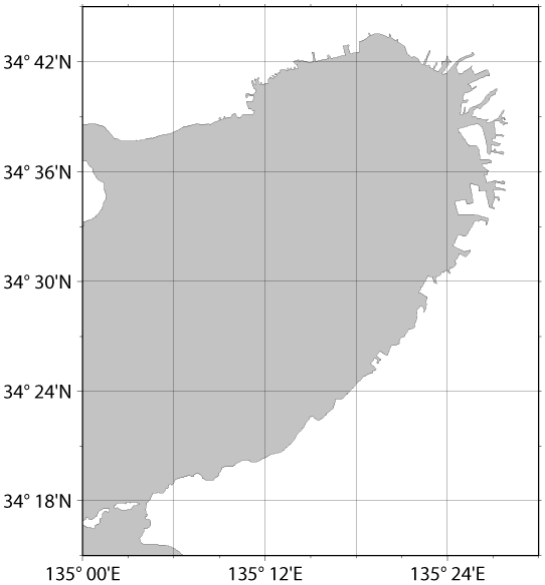

図４．GMT パッケージ内の地図データに基づいた大阪湾周辺の地図

GMT パッケージ内の地図データには、最新の埋め立てデータが含まれていないため、図1と比較しても分かるように、図4には一部の埋め立て地が描かれていない。

今回は、作成した変換プログラムの出力を用いて、最新の地形情報（埋め立て地や関西国際空港など）を加えて、GMT で大阪湾周辺の地図を描画したい。変換プログラムの出力は、「数値地図 25000（行政界・海岸線）」のデータを抽出したものなので、これを用いて GMT で描画させれば海岸線図には最新の埋め立て地や空港などが描けると予想し、以下に試みた。

なお、変換プログラムの出力を用いて描画する場合、GMT は入力ファイル（変換プログラムの出力ファイル）の線データ（緯度・経度データによって構成される）を、ファイルに列記されたデータ順に上から読み込んだ上で、座標上の点と看做しそれらを結ぶ線を描画し、線の始点と終点を結んで囲まれる領域の内側/外側を塗り潰すという動作をすることになる。このとき、GMT は線データを線 ID 毎に区切って処理する動作と、線 ID で区切らずに全ての線データを単一の線として処理する動作をスクリプト内で指定できる。

２．４．GMT を用いた描画における問題

２．２．で説明した変換プログラムの出力を用いて、GMT によって海岸線を描画し、陸部を塗りつぶす試みでは、不要な線が描かれたり、必要な線が描かれなかったり、海部が陸部の色で塗られたりして、期待する結果が得られなかった。

以下の（１）から（３）にその詳細を述べる。以下の図5から図7は、変換プログラムの出力を用いて、図4に重ねて描画したものである。

（１）各線を線 ID で区切らず、全ての線データを単一の線として入力順に一筆で結んだ場合、陸と島がつながってしまう。島を構成する線データを分離し、残りの海岸線全てを単一の線として一筆で描画することも考えられた。しかし、変換プログラムの出力における線の順番と地図における線の順番が異なっていたために、不要な線が描かれることが分かった。

変換プログラムは、海岸基本データにおける線情報（２．１．（２）、（３））の領域に記述された順にデータを出力したが、海岸基本データにおける線情報の領域に記述された線の順番は、地図上では連続でなかった。そのため、海岸基本データにおける記述の順に描画された線は地図上で散在し、接続すべきでない各線の間をつないでしまうことになる。

（２）線を ID 毎に描画するコマンドを実行した場合を図5に示す。同図では、単一の線で構成される閉ループの一部（例：同図-ア）、線 ID はあるが点データのない線（例：同図-イ）、ある線とそれに隣接する線の間（例：同図-ウ）が描画されなかった。

これは、変換プログラム上で線の始終における節点データを出力させなかったためである。閉ループは、始節点と終節点が同一なのでループとなる。よって節点がなければ、線は閉じない。点データのない線は始節点と終節点だけの情報を持つ線なので、節点がなければ描かれない。隣接する線の間は節点によって接続されているので、やはり節点がなければ、穴が空いてしまう。

（３）線を ID 毎に描画し、かつその線の囲む範囲内を白く塗るコマンドを実行した場合、GMT は 1 つの線の始点と終点を自動的に結び、その中を塗りつぶす。これにより、節点データがなくとも、閉ループは正しく閉じることになる。その結果を図 6 に示す。

単一の線で構成される閉ループは正しく描画されたが、他の線と接続して海岸線を形成するような線（本来、自身の始点と終点が直接つながらない）に対しては始点と終点をつなぐ不要な線が描画され、その不要な線で囲まれた海部が白く塗られた（例：同図-ア）。また、点データのない線などは描かれなかった（例：同図-イ）。

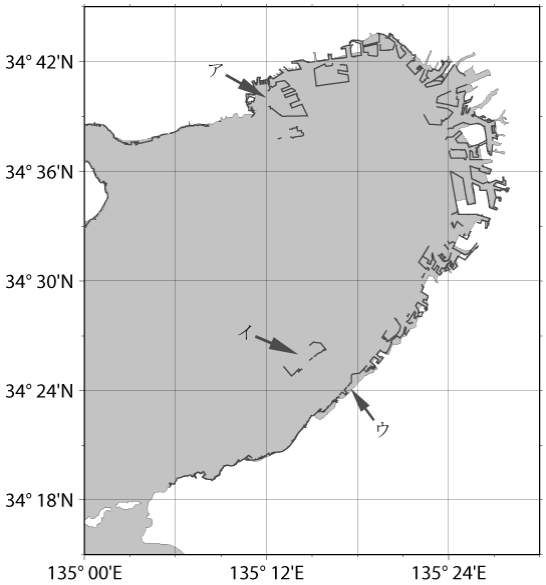

図５．線を ID 毎に描画した場合の図

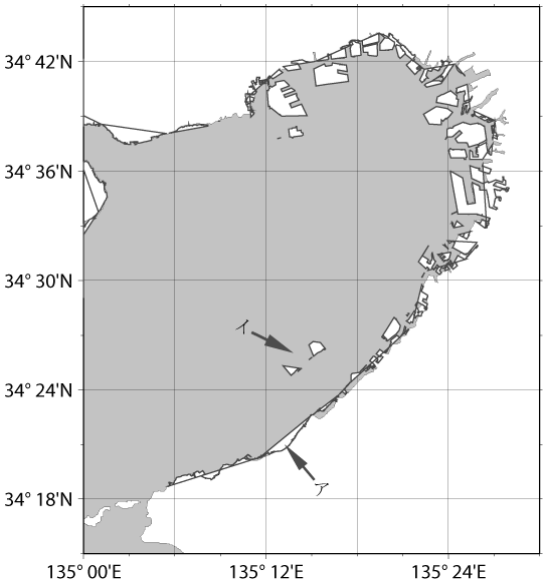

図６．線を ID 毎に描画し、かつ閉ループ内を塗りつぶした場合の図

２．５．変換プログラムの改良とその結果

前節の問題を解決するために、変換プログラムに改良を施した。改良点（Ａ）～（Ｃ）、および改良後の変換プログラムの出力を用いて GMT で描画した結果について述べる。

（Ａ）海岸基本データの行政区域情報（２．１．（１））に記述された線 ID（その行政

区域となる面を分節する線の ID）は、いくつかを調べたところ、データ上の順番と地図上の順番が一致していたので、ここから線 ID を読み込んで、出力の際にこの順で出力するようにした。

（Ｂ）海岸基本データの行政界情報（２．１．（２））および海岸線情報（２．１．（３））には､その線における始終の節点 ID が記述されている｡これと行政界節点情報(２．１．（４））の緯度・経度データを基に、出力の際、ある線データの始めと終わりに各節点の緯度・経度データを出力するようにした。

（Ｃ）出力ファイルを、単一の線で閉ループとなるもの（island）と、他の線と接続して海岸線となるもの(CL)の 2 つに分離した。GMT では、island を ID 毎に分離して結んで白く塗りつぶし、CL を ID 毎に分離して結んだ。

上記（Ａ）によって線データを適切な順に出力できることが期待されたが、行政区域情報における線 ID の順番と地図上の順番が一致しているものは一部しかなく、改良された変換プログラムの出力における線データの順も地図における線の順に一致していなかった。

上記（Ｂ)、(Ｃ）により、GMT が入力ファイル（CL）に基づいて ID 毎に線を描画しても、欠けた部分や不要な部分は現れなかった。また、入力ファイル（island）に基づいて ID 毎に線を描画し、かつその線の囲む領域内を白く塗るコマンドで、閉ループは正しく島として描画された。その結果を図 7 に示す。図 1 と比較しても正しく海岸線が描画できているといえる。図 7 で、海岸線よりも陸側であるのに灰色になっている部分（例：同図-ア）は、埋立地である。同図内における中央下寄りの長方形に似た島（同図-イ）は、関西国際空港である。関西国際空港は、島であるが island でなく CL に含まれるため、塗りつぶされていない。これは例外として処理しなければならず、以降、CL から除外して考える。

海岸線の描画および island の塗りつぶしは達成されたが、CL の外側に広がる陸部は完全には塗りつぶされていない。一部の埋め立て地に当たる部分（例：同図-ア）が灰色のまま残っている。ここを白色で塗るためには、CL によって囲まれる領域の外側を塗りつぶす動作を指定すればよい。しかし、その場合、入力データ（CL）における線データの順番と地図上で海岸線が接続する順番が一致しなければならない。なぜなら、GMT の動作は、点データを入力ファイルに列記された順に上から読み込み、それらを結ぶ線によって囲まれる領域の内側/外側を塗りつぶすものだからである。この動作によって CL の外側を塗りつぶすためには、CL における全ての線を、線 ID で区切らず単一の線として、CL に記述された順に一筆で結ぶ必要がある。このとき、CL における線の順番が適切でなければ、隣接しない各線の間を接続する不要な線が描かれ、その不要な線によ

って囲まれる領域が作られてしまい、適切に描画することができない。

線データが適切な順に並んだ CL を得ることは、今後の課題である。

その他の改良箇所として、変換プログラムが入力ファイル（海岸基本データ）の同じ部分を 2 度以上参照しないようにし、また、長大なデータ配列のためにインデックスを作成し、インデックスの配列には冗長性を持たせ、シーケンシャルな参照を極力排することで、処理の高速化を図ったことと、データ数（1 個の入力ファイルにおける緯度・経度データの総数）の上限が不定のため、可変長配列を用い、大きな入力に備えたことが挙げられる。

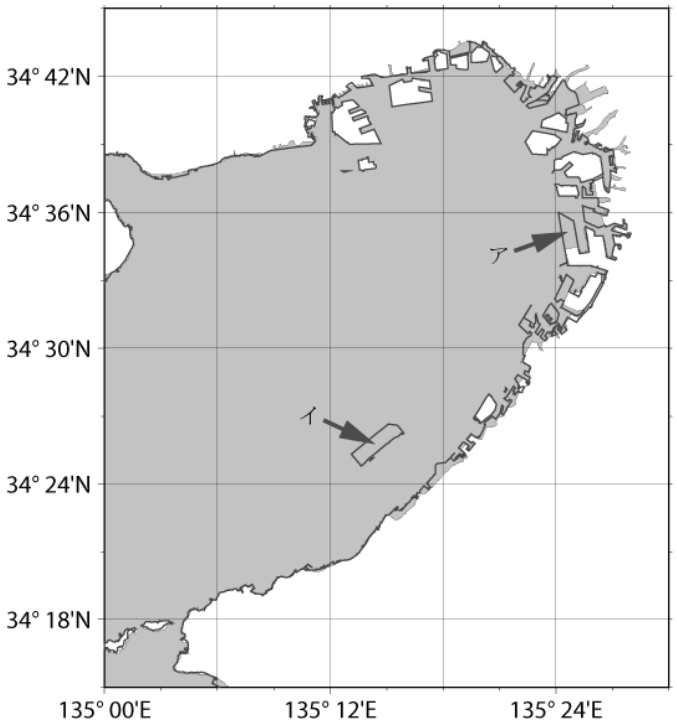

図７．改良した変換プログラムの出力データを用いて描画した図

３．数値地図 50m メッシュ（標高）

３．１．「数値地図 50m メッシュ（標高）」について

「数値地図 50m メッシュ（標高）」は、2 万 5 千分 1 地形図(2 次メッシュ)を経度方向および緯度方向に、それぞれ 200 等分（つまり 200×200 = 4 万等分）して得られる各区画（1/20 細分メッシュ、2 万 5 千分 1 地形図上で約 2 ㎜×約 2 ㎜）の中心の標高のデータ（以下、標高基本データと略記）と、標高を色によって表現するビューアからなる。標高点の間隔は緯度方向で 1.5 秒、経度方向で 2.25 秒となり、実距離で約 50m×約 50m となる。

図 8 は、「数値地図 50m メッシュ（標高）」付属の専用ビューアによる九州の地図である。格子は 2 次メッシュを表す。図 9 は図 8 の段彩色の設定である。

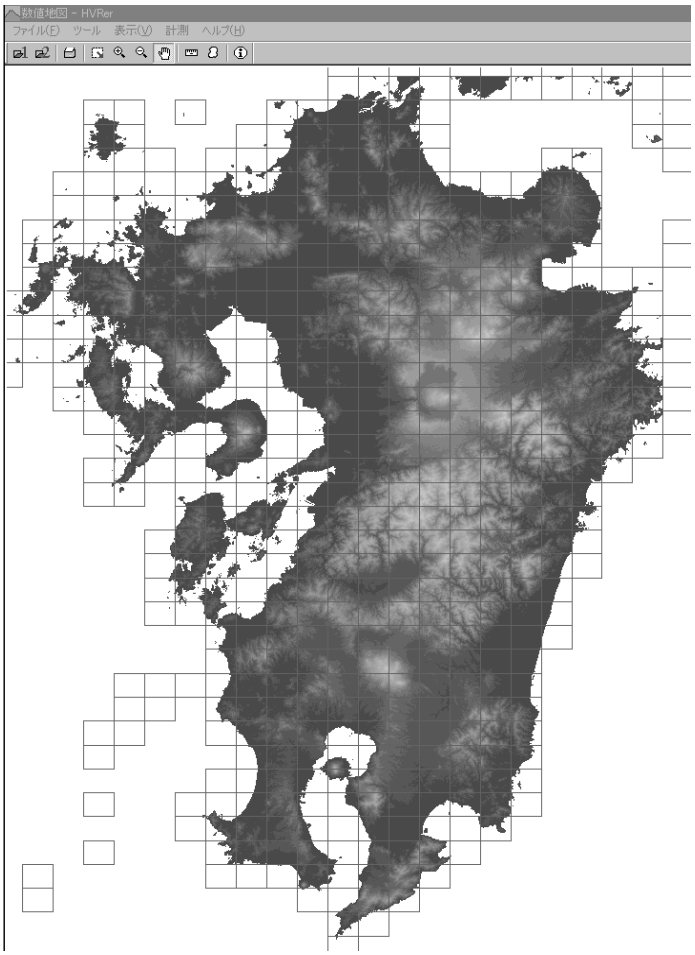

図８.「数値地図 50m メッシュ（標高）」付属の専用ビューアを用いて描画した九州の地図

図９．図 8 の段彩色の設定

標高基本データは、1 個のファイルが 1 個の 2 次メッシュデータとなる。1 個のファイルは、1 行目のヘッダと、2 行目以降のレコードからなるテキストデータである。ヘッダには 2 次メッシュの各頂点における緯度・経度情報等が記述されている。レコードは、1 個の 2 次メッシュにつき 200 個あり、1 行につき 1 個が記述される。標高値は 1 個のレコードにつき 200 個記述される。標高値は 0.1m 単位であり（しかし、0.1 の位は全て 0 であり、実質は 1m 単位である)、海部は”-9999”と記述される。図 10 は 503130 番の 2 次メッシュデータの一部である。レコードには、最初 6 バイトに 2 次メッシュ番号、続く 3 バイトにレコード番号、その後 5 バイトずつ標高値が記述されている。

```
5031302500019711986199620020003335001310000033400013107301椎田□□□□□□□□□1
503130001004100040000390003800040000400004000039000390003900038000400003600033 0
503130002004100039000370004100045000410003900039000390003900040000400003900036000340
503130003004300041000430004600044000410003900039000390003900040000400003900037000370
503130004004400044000450004400041000390004000039000400004000040000390003800039 0
503130005004400045000440004300040000400003900039000400004000040000390003900039000400
503130006004400045000450004400042000400003900045000430004200041000390003900039000400
503130007004500045000440004400043000420004600045000440004400040000390003900039000430
503130008004500045000440004400044000450004500044000440004700039000380003900046 0
503130009004500045000440004500046000450004500045000450004500038000390004200049 0
503130010004500045000460004700047000450004500045000450004400039000390004700051 0
503130011004700048000500005000048000450004500044000440004200039000390005100058 0
503130012004900049000500005000048000440004400045000430004000038000450005400060 0
503130013004900050000500004900046000450004500045000440004000039000460005800061 0
503130014005000049000490004900048000460004500045000440004200042000440005300062 0
```

図１０.「数値地図 50m メッシュ（標高)」データファイルにおける書式の一例

３．２．標高データ変換プログラムの作成

作成した変換プログラムは、標高基本データを読み、次の 3 点に基づいて、ある標高点の緯度・経度の値を算出し、その標高値と組み合わせて、ファイルに出力するものである。

- ある 2 次メッシュの左上頂点における緯度・経度（ヘッダより読み取れる）
- その 2 次メッシュ内のある標高データが何番目のレコードの何個目のデータか
- 標高点の間隔は緯度方向で 1.5 秒、経度方向で 2.25 秒である

出力ファイルの書式は、ある 2 次メッシュの番号（標高基本データのファイル名）がある 1 行に記述され、次の行から、その 2 次メッシュ内の標高点における緯度・経度・標高の数値データが 1 行につき 1 点ずつ順に記述され、全ての点が記述されると、次の行から､次の 2 次メッシュについて記述されるという簡素なものである｡図 11 は､503130 番の 2 次メッシュデータを変換した出力の一部である。

変換プログラムによって出力された九州のデータを GMT で描画した図を示す(図 12)。図 8 と比較して、同等の図が問題なく出力・描画できている。

```
#503130.MEM
33.669764,   130.997920,   0041.0
33.669764,   130.998551,   0040.0
33.669764,   130.999173,   0039.0
33.669764,   130.999796,   0038.0
33.669764,   131.000427,   0040.0
33.669764,   131.001049,   0040.0
33.669764,   131.001671,   0040.0
33.669764,   131.002302,   0039.0
33.669764,   131.002924,   0039.0
33.669764,   131.003547,   0039.0
33.669764,   131.004178,   0038.0
33.669764,   131.004800,   0040.0
33.669764,   131.005422,   0036.0
33.669764,   131.006044,   0033.0
33.669764,   131.006676,   0034.0
33.669764,   131.007298,   0037.0
33.669764,   131.007920,   0036.0
33.669764,   131.008551,   0036.0
33.669764,   131.009173,   0037.0
```

図１１．標高データ変換プログラムによる出力結果の一例

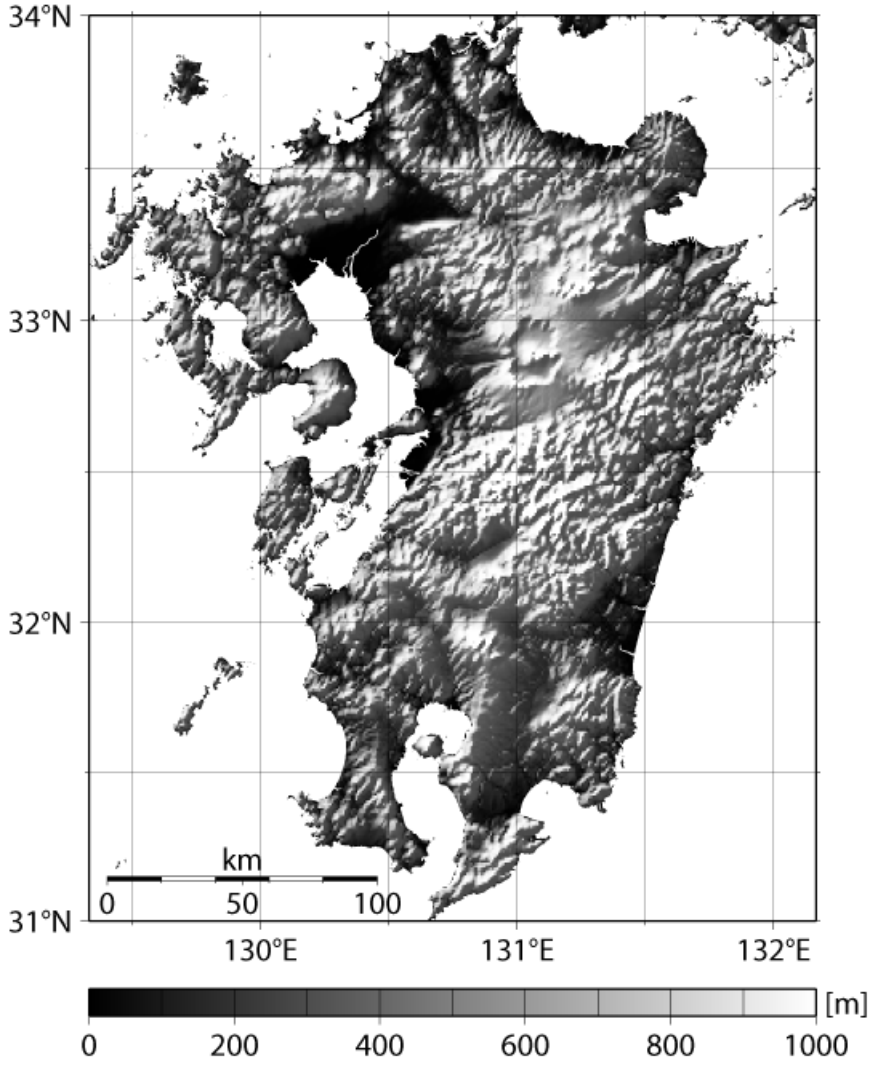

図１２．変換プログラムの出力データを用いて GMT で描画した九州の地図

４．おわりに

今回、「数値地図 25000（行政界・海岸線）」および「数値地図 50m メッシュ（標高）」のデータセットから各数値データを抽出し、他のプログラムで利用しやすいよう簡素な書式に変換して出力する変換プログラムとして、行政界・海岸線データ変換プログラムおよび標高データ変換プログラムを作成した。また、今回汎用地図ツールとして利用した GMT によって扱える書式に変換できるように、行政界・海岸線データ変換プログラムの改良を試みた。これにより、海岸線について専用ビューアで地図を閲覧するだけでなく、GMT を用いて描画できるようになり、画像編集や文字挿入などが可能となった。しかし、GMT による描画では、一部の埋め立て地や空港を陸部として塗りつぶすことができなかった。さらに、標高データ変換プログラムを用いて変換されたデータを用いて、GMT による描画が可能となった。

行政界・海岸線データ変換プログラムについては今後も改良を加え、GMT を用いて所望の地図が描けるようにしたい。

参考文献

B.W.カーニハン, D.M.リッチー： プログラミング言語 C　第 2 版　ANSI 規格準拠, 石田晴久訳, 共立出版株式会社, 2004