

河川氾濫避難モデルの作り方

2007/10/09 (株) 構造計画研究所

1. はじめに

artisoc は、マルチエージェント・シミュレーションを世の中に広めるために様々な機能を提供しております。**artisoc2.0** では、道路やネットワークが簡単に描ける描画ツールを提供しており、交通モデルや避難モデルがさらに作りやすくなりました。

「河川氾濫避難モデル」は、地方自治体等で配布されているハザードマップと **artisoc** と **Google Earth** を利用することで、住民のミクロな視点での行動を表現し、避難計画の立案に役立てようという試みです。これは工学的な知見により作成されたハザードマップと複雑系の得意とするボトムアップのシミュレーションを組み合わせ、さらに 3D 可視化することで、専門知識がなくても『分かりやすい』シミュレーションを提供することができます。

なお、本ドキュメントはモデルの作成手順を解説することを主眼としておりますので、避難行動については厳密に定義するものではありません。本モデルを参考に、現実の避難行動を反映したモデルが作成されることを期待します。

2. 準備

避難モデルを作成するとき、まずは何を対象に避難をするかを決める必要があります。

ハザードマップと一口に言っても、洪水、内水、高潮、津波、土砂災害、火山など様々な種類があります。

本モデルは、福井県の足羽川の河川氾濫想定データを利用しております。

(参考)

地質情報整備・活用機構提供

ハザードマップ（災害予測図・危険範囲図など）の公式掲載サイト

<http://www.gupi.jp/link/link-b/hazard-index.html>

3. 道路情報の作成

避難モデルを作成する地点を決めたら、避難経路となる道路情報を用意する必要があります。ここで、**artisoc2.0** の新機能である描画ツールを利用します。描画ツールの詳細については「描画ツールの使い方」を参照してください。

描画ツールの使い方

<http://mas.kke.co.jp/cabinet/byoga-tool.pdf>

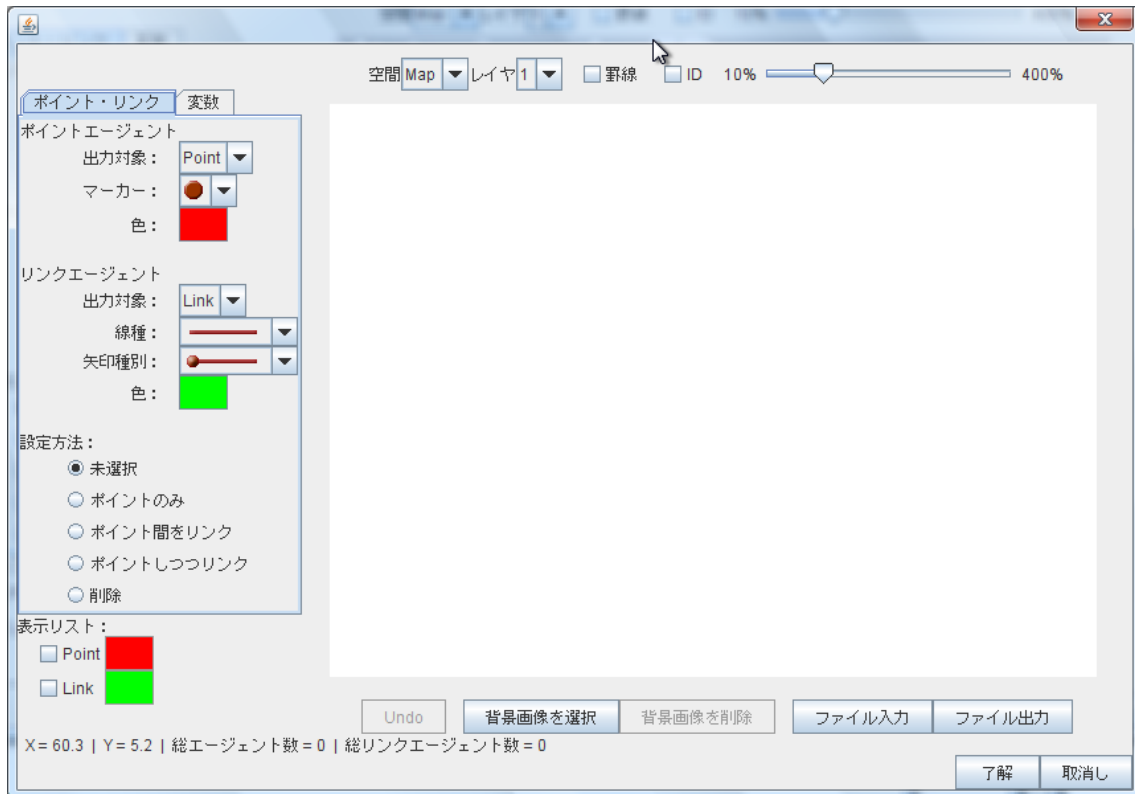
描画ツールを起動する前にツリーに次のコンポーネントを定義してください。

名前	変数型名	説明
Universe.Map	空間	空間を定義します ※本モデルは 80×60 の大きさで定義
Universe.Map.Point	エージェント	交差点を定義します
Universe.Link	エージェント	道路（交差点のリンク）を定義します
Universe.Link.Begin_Agt	エージェント型変数	道路の端点（始点）を定義します
Universe.Link.End_Agt	エージェント型変数	道路の端点（終点）を定義します

なお、「Begin_Agt」、「End_Agt」は描画ツールがリンクエージェントの対象を認識するための予約語ですので別名で定義できません。



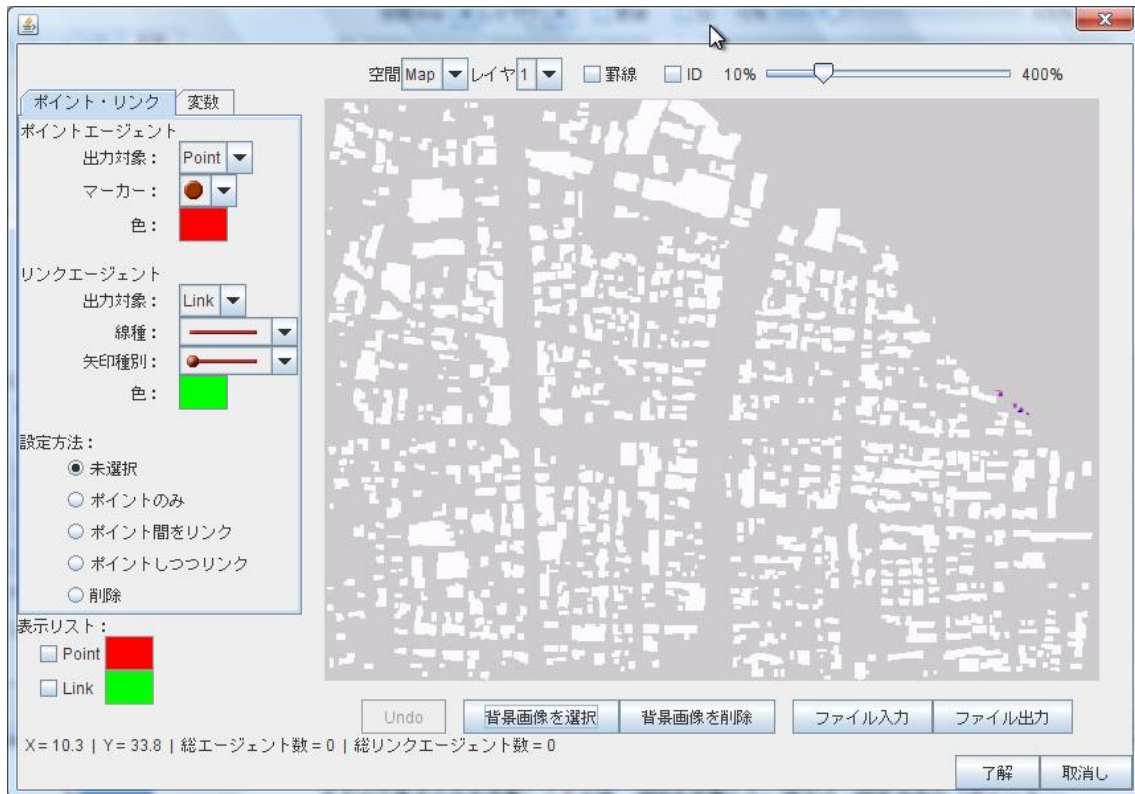
ここで、「Map」を右クリック（もしくは設定メニューを選択）して「初期値設定」をクリックすると描画ツールが表示されます。



画面左部にコントロールエリア、画面右部にキャンバスエリアが表示され、画面上部と画面下部にアシストエリアが表示されます。

ポイントエージェントの出力対象に「Point」、リンクエージェントの出力対象に「Link」が定義されていることを確認してください。また、設定方法は「未選択」であることを確認してください。

次に、該当エリアの画像を読み込みます。画面下部の「背景画像を選択」をクリックし、任意の画像を選択するとキャンバスに画像が読み込まれます。

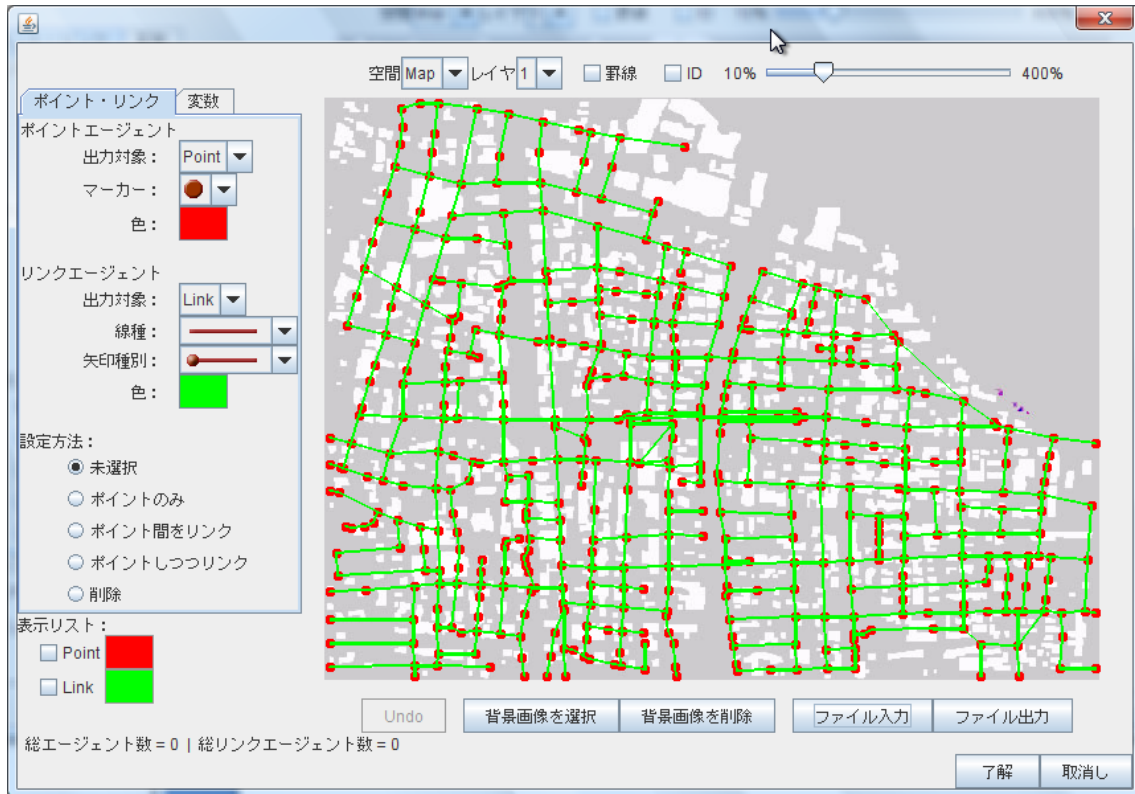


次に設定方法にて「ポイントしつつリンク」を選択してキャンバスをマウスで左クリックしていくと交差点と道路を連続して定義できます。道路の終点もしくは途切れる場合は、マウスで右クリックにより選択状態を解除できます。

誤って交差点や道路を定義した場合は、設定方法の「削除」を選択して任意の交差点や道路を左クリックすると削除できます。

まずは交差点のみを定義してその後、道路を定義したい場合は、設定方法の「ポイントのみ」を選択して交差点を定義した後に設定方法の「ポイント間をリンク」を選択して道路を定義すればよいです。

交差点と道路を定義すると次のような状態になります。



本モデルでは、ポイントエージェント「Point」を 484、リンクエージェント「Link」を 639 定義しております。

別途 GIS データ等で交差点や道路の位置情報がある場合は、CSV ファイルによる入出力を行うことができます。操作は画面右下部の「ファイル入力」「ファイル出力」を利用します。

CSV ファイルのファイル書式は前述の「描画ツールの使い方」を参照してください。

4. 最短経路探索

道路情報を定義した段階では、避難者が避難所までどのような経路を使って移動するかについては解決できません。

そこで本モデルでは、最短経路探索として一般的なダイクストラ法を利用することにより、避難者の経路の初期値を作成します。

ダイクストラ法を利用するには、次のコンポーネントを定義してください。

名前	変数型名	説明
Universe.Map.Point.DijkstraValue	実数型変数	ダイクストラ法計算用変数を定義します
Universe.Map.Point.Choice_Agt	エージェント型変数	ダイクストラ法計算用変数を定義します
Universe.Link.Cost	実数型変数	ダイクストラ法計算用に重み付け値として定義します
Universe.DijkstraLink	エージェント	最短経路を保持する道路（交差点のリンク）を定義します
Universe.DijkstraLink.Begin_Agt	エージェント型変数	道路の端点（始点）を定義します
Universe.DijkstraLink.End_Agt	エージェント型変数	道路の端点（終点）を定義します
Universe.DijkstraLink.Route	文字列型変数	最短経路を定義します
Universe.DijkstraLink.Value	実数型変数	ダイクストラ法計算用変数を定義します

前述で定義した「Universe.Link」の情報を元にダイクストラ法で計算した結果を「Universe.DijkstraLink」に格納します。具体的な最短経路については、「Universe.DijkstraLink.Route」で保持しますが、データの格納書式は、「Universe.Map.Point」のエージェント ID のカンマ区切りの配列です。これらのコンポーネントを利用した実際の計算は、「Universe」で定義される関数「DijkstraMethod」、「DijkstraCalculation」です。

ダイクストラ法はリンク間の重みを考慮して計算を行いますが、本モデルでは、距離を代入して計算を行っております。距離以外に道幅や傾斜、歩行困難度等を考慮する場合は、関数「set_cost_reverse」を修正してください。

5. モデル実行における 3 つのステップ

最短経路の計算とエージェント毎の避難経路（ルート）の定義は計算時間がかかるため、これは別途計算してファイル保存しておき、シミュレーション実行は計算済みのファイルを読み込み、初期値として利用しております。

以下、本モデル「fukui.model」を参照しつつ、読み進めてください。

本モデルのコントロールパネルでは「実行モード」と「氾濫流速」の 2 つの値を指定できますが、このうち「実行モード」がこれに当たります。

「実行モード」に「(最短経路計算)」を指定して実行を行う前に、「Universe.DijkstraLink」のプロパティを開き、エージェント数を「0」と指定してください。

はじめてモデルを実行する場合、「実行モード」に「(最短経路計算)」を指定して実行します。このとき、Univ_Init の 9 行目以降の処理が実行されます。

If StrComp(Universe.CalcMode, "(最短経路計算)") == 0 Then

ここではダイクストラ法を利用した最短経路の計算を行ってシミュレーションを終了します。

本モデルでは、交差点の数 484、道路の数 639 に対して、選択しうる交差点と交差点のパターン数は 233,772 であり、この最短経路を計算しております。計算結果については、「dijkstra.csv」に保存されます。

ここで、「Universe.DijkstraLink」のプロパティを開き、エージェント数を「233772」と指定してモデルを保存してください。

次に、「実行モード」に「(ルート取得)」を指定して実行します。このとき、Univ_Init の 23 行目以降の処理が実行されます。

ElseIf StrComp(Universe.CalcMode, "(ルート取得)") == 0 Then

ここでは、最短経路の計算結果をファイル入力し、すべての交差点に避難者 (Evacuee)

を1名ずつ生成し、ルートを定義してシミュレーションを終了します。初期状態として交差点ではなく、道路上に避難者を配置したい場合は `Universe` の関数「`create_Evacuee`」を修正してください。

最後に、「実行モード」に「避難実行」を指定して実行するとシミュレーションが実行します。このとき、上記の2ファイルのほか、浸水情報ファイル（`flood.csv`）と交差点追加情報ファイル（`point_info.csv`）を入力します。

浸水情報ファイルは「`Universe.Map.Point`」のエージェント ID と浸水する相対時間を格納し、交差点追加情報ファイルは「`Universe.Map.Point`」のエージェント ID と標高「`Universe.Map.Point.Z`」を格納します。

6. 浸水の再現

前述の浸水情報ファイル (flood.csv) より読み込んだ値は Univ_Step_Begin で適宜、浸水フラグ「Universe.Map.Point.Flood」を立てている。

7. 避難者 (Evacuee) の行動

本モデルでは避難者は次のルールに従って避難行動を行っております。

- 目的地「Target_Points」を目指して最短経路を移動します。このとき、「Target_Points」は「Universe.Map.Point」のエージェント ID のカンマ区切りの配列です。
- もし、渋滞しているときは、その混雑度に応じて減速します。
- 移動途中で交差点が浸水しているときはその交差点に接続する交差点を検索し、浸水していない交差点がある場合はそのいずれかを目的地として再定義して最短経路を検索して避難行動を行ないます。接続する交差点がすべて浸水している場合はその時点で停止します。
- 定義したすべての目的地に達したときは停止します。

Universe の関数「create_Evacuee」にて、避難者を歩行速度により遅い人（緑色）と速い人（赤色）の 2 種に分けています。浸水に遭遇した人は黄色、浸水で停止した人は黒色に変更されます。

8. シミュレーションの出力

ハザードマップの時系列の画像ファイルを準備できる場合は、これを下絵としてエージェントを動作させることができます。

本モデルでは、マップ出力「Map」にて背景画像の「ファイル指定変数名」に「Universe.BackGround_Filename」を定義し、Univ_Step_Beginにて変数を更新しております。

artisoc 単体での動作確認は、マップ出力「Map」で確認できますが、Google Earth との連携を行うために、Univ_Step_End で関数「ScreenShot」を利用して各ステップの実行結果を画像ファイル（1 から始まる連番をつけた GIF ファイル）として保存します。このとき、関数「ScreenShot」はアクティブウィンドウをキャプチャリングするため、マップ出力「Map」の実行ウィンドウがアクティブであることを確認してください。

なお、本モデルは計算すべき避難者の数が各ステップで異なるため、ステップ毎の計算時間が異なります。計算速度が極めて高速もしくは無視できる場合は、画像ファイル名を固定にしてダイレクトに Google Earth と連携することができます。

9. シミュレーションの再現と Google Earth との連携

「fukui.model」で作成した連番画像ファイルを「movie.model」を利用して与えられた時間ごとに「artisoc.gif」というファイル名に変更します。

「movie.model」を実行する前に最大ステップ数（用意した画像ファイルの最大の番号）と開始ステップ数を指定してください。

次に、「河川氾濫避難モデル.kml」をクリックします。すると Google Earth が起動し、「artisoc.gif」がレイヤ（イメージオーバーレイ）の1つとして表示されます。

なお、「河川氾濫避難モデル.kml」は画像ファイルの配置場所と更新頻度を定義しているのみです。詳細については、Google Earth の「場所」の「保留・浸水想定（福井県）」をマウスで右クリックしてプロパティを選択し、リンクが「artisoc.gif」になっていること、更新タブの「時間をベースに更新」の「実行間隔」が「定期的」「2 秒」になっていることを確認してください。

建物を 3D 表示にしたい場合は、Google Earth の「レイヤ」にて「建物の 3D 表示」を選択してください。

モデル実行中は 2 秒おきに画像が更新されますが、この間に Google Earth の視点をマウスで操作することができ、避難者の目線での避難行動や任意の点からの避難状況が表示できます。

10. 今後の展望

本モデルについては避難シミュレーションの基本編として提供しており、いくつかの拡張案を考えております。

- 避難行動をより詳細に記述したい。(人だけでなく車の動き、誘導者の存在、近所付き合いや地域コミュニティの影響等)
- 避難経路について、基本的に主要道路を通り、状況によって細い道路を通るようにしたい。
- GIS データをより簡便に取り込むようにしたい。
- 複数のハザードマップを重ね合わせた被害想定を考慮したい。(地震が発生したことによる津波と火災等)
- 大規模地域を想定した複数モデルの同時実行を行いたい。
- 全体的に高速化したい。