

津波数値解析プログラムのオブジェクト指向での分析、クラス構造の確立、C++言語を用いたコーディング

学籍番号1130072 氏名 深津 宗祐
高知工科大学システム工学群 建築・都市デザイン専攻

現在使用されている津波数値解析プログラムをオブジェクト指向で分析を行い、クラス構造を作成した。作成したクラス構造をもとにUML(統一モデリング言語)を使用したクラス図での表記を行った。クラスの属性、メソッドからオブジェクト指向プログラミング言語であるC++を用いたコーディングを行い、作成したクラスが実装可能か検討した。

Keywords : 津波数値解析、オブジェクト指向、UML、クラス

1. はじめに

1.1 研究背景

南海トラフで地震が起こった際、発生した津波の波源から、スーパーコンピュータによる高速な津波の解析を行い、遡上域を求めることができれば、津波到着前にいち早く津波の遡上域を伝え、住民を無事に避難させることができる。

スーパーコンピュータは、GPGPU(画像処理用演算ユニット)を用いた並列計算を多量に効率的に行うことで高速化している。現在の津波数値解析プログラムのフォートラン言語は並列計算に適したプログラミング言語ではないためスーパーコンピュータの能力を活用することができない。

そのため、津波解析プログラムをオブジェクト指向で分析しクラス構造の確立をすることで、並列計算に適したプログラムになり、GPGPUを用いた演算の高速化ができる。

クラス構造にすることで、どれかのクラスにバグが見つかった場合でも、そのクラスを変更するだけでプログラムが成り立つ。クラスが自分の意図していない使われ方がされデバックが行われる、多人数でのプログラム作成が容易、などの利点もある。

1.2 研究手順

数値流体力学の解析法をオブジェクト分析する。そのためには、数値流体力学の知識とオブジェクト指向の理解が必要となる。

オブジェクト指向とは、認識できるモノどうしの関係性を抽象化し、形にしようという考え方で、これを使用したプログラミングをオブジェクト指向プログラミングといい、相互にメッセージを送りあうオブジェクトの集まりとしてプログラムを構成する。

津波数値計算で使用される数値流体力学には線

形長波理論、非線形長波理論(浅水理論)、分散波理論などがあるが、ここでは非線形長波理論(浅水理論)を扱う。

非線形長波理論(浅水理論)は水位を求める連続式と流量を求める運動方程式に分けられ、非線形長波理論(浅水理論)は水位を求める連続式と流量を求める運動方程式に分けられる。また、津波のような連続体は本来、数値シミュレーションでは扱うことができないため、それぞれ、空間格子、時間格子で差分化したものを用い、そのオブジェクト分析を行う。

オブジェクト分析により抽出した関係性をもとに、津波数値解析プログラムにふさわしいクラス構造を作成する。

作成したクラス構造をUML(統一モデリング言語)のクラス図を用いて表記する。UML(統一モデリング言語)は情報の共有を図る際に利便性の高いツールであり、ソフトウェア工学におけるオブジェクトモデリングのために標準化した仕様記述言語であり、グラフィカルな記述で抽象化したシステムのモデル(UMLモデル)を生成する汎用モデリング言語である。13種類の図(ダイアグラム)必要に応じて書き分ける。

クラス図を作成する際に気を付けなければいけないことは、クラスがどのような属性を持っており、どのようなメソッド(プログラム上での動き)を起こすのか。また他のクラスとの関係性も念頭に置いていなければならない。

作成したクラス構造をもとにオブジェクト指向プログラミング言語であるC++によるコーディングを行う。

2. 研究概要

2.1 オブジェクト分析

オブジェクト分析についての説明を行う。

はじめに、津波数値解析の基本となる算定式を以下に示す。

$$\frac{\partial \eta}{\partial t} + \frac{\partial M}{\partial x} + \frac{\partial N}{\partial y} = 0$$

式1 連続式

$$\frac{\partial M}{\partial t} + \frac{\partial}{\partial x} \left(\frac{M^2}{D} \right) + \frac{\partial}{\partial y} \left(\frac{MN}{D} \right) + gD \frac{\partial \eta}{\partial x} + \frac{gn^2}{D^{7/3}} M \sqrt{M^2 + N^2} = 0$$

式2 x方向の運動方程式

$$\frac{\partial N}{\partial t} + \frac{\partial}{\partial x} \left(\frac{MN}{D} \right) + \frac{\partial}{\partial y} \left(\frac{MN M^2}{D} \right) + gD \frac{\partial \eta}{\partial y} + \frac{gn^2}{D^{7/3}} N \sqrt{M^2 + N^2} = 0$$

式3 y方向の運動方程式

津波のような連続体はコンピュータによる数値シミュレーションでは扱うことができない。そのため、実際に数値解析を行うには、連続体を不連続的な点へ置き換えなければならない。

連続式は、空間格子での差分化を行い、運動方程式は時間格子での差分化を行う。

連続式及び運動方程式を差分化した水位算定式と流量算定式を以下に示す。

$$\eta_{i,j}^{k+1} = \eta_{i,j}^k - \Delta t \left(\frac{M_{i+\frac{1}{2},j}^{k+\frac{1}{2}} - M_{i-\frac{1}{2},j}^{k+\frac{1}{2}}}{\Delta x} + \frac{N_{i,j+\frac{1}{2}}^{k+\frac{1}{2}} - N_{i,j-\frac{1}{2}}^{k+\frac{1}{2}}}{\Delta y} \right)$$

式4 水位算定式

$$M_{i+\frac{1}{2},j}^{k+\frac{1}{2}} = \frac{1}{1 + \mu_{i+\frac{1}{2},j}^{k-\frac{1}{2}}} \left[\left(1 - \mu_{i+\frac{1}{2},j}^{k-\frac{1}{2}} \right) M_{i+\frac{1}{2},j}^{k-\frac{1}{2}} - gD_{i+\frac{1}{2},j}^{k-\frac{1}{2}} \frac{\Delta t}{\Delta x} (\eta_{i+1,j}^k - \eta_{i,j}^k) - \frac{\Delta t}{\Delta x} \left\{ \frac{\left(M_{i+\frac{1}{2},j}^{k-\frac{1}{2}} \right)^2}{D_{i+\frac{1}{2},j}^{k-\frac{1}{2}}} - \frac{\left(M_{i-\frac{1}{2},j}^{k-\frac{1}{2}} \right)^2}{D_{i-\frac{1}{2},j}^{k-\frac{1}{2}}} \right\} \right]$$

式5 流量算定式

現在の水位を求めるためには、前時間ステップの水位と周りの流量が必要なことがわかる。

現時間ステップの流量を求めるためには、隣接する水位、前時間ステップの同座標流量と異なる座標の流量、全水深、粘性係数が必要なことがわかる。

以上のことから、水位、流量は互いが自分を求めるために必要な要素となっていることがわかる。

現在の水位及び流量を求めるための要素を属性、実行する作業をメソッドとシクラスを作成する。

作成した水位と流量のクラスを以下に示す。

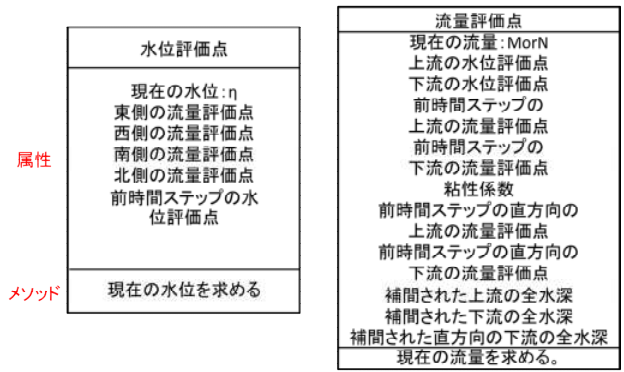


図1 水位・流量クラス

オブジェクト分析により、抽出した関係性をオブジェクトイメージを使い説明する。

以下にオブジェクトイメージを示す。



図2 オブジェクトイメージ

水位のオブジェクトモデルを以下に示す。

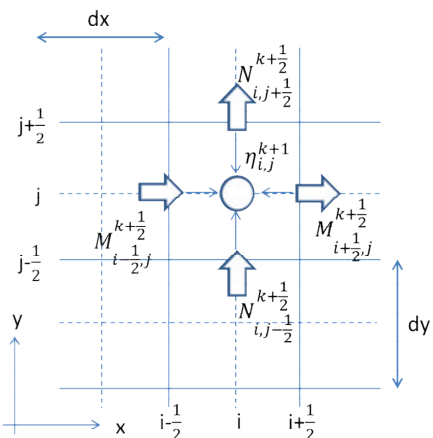


図3 水位 X-Y平面オブジェクトモデル

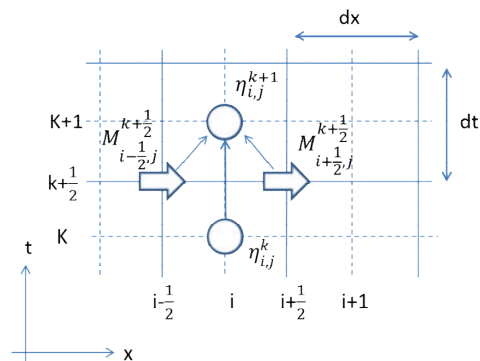


図4 水位 X-t平面オブジェクトモデル

図3より、現時間ステップの水位 η を求めるには東側の流量評価点、西側の流量評価点、南側の流量評価点、北側の流量評価点が必要だということが

わかる。

図4では、さらに前時間ステップの流量評価点が必要だということがわかる。

一方、流量評価点では、時間・空間補間された水位が必要となる。時間・空間補間の概念図を以下に示す。

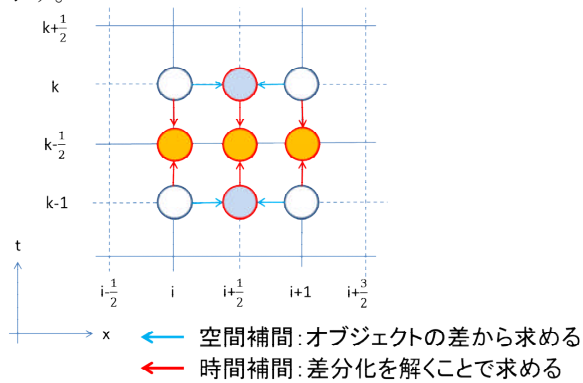


図5 時間・空間補間概念図

このような補間を行うことで、本来水位が存在しない点の水位を使うことができる。なお、流量も同様に空間・時間補間を行うことができる。

流量のオブジェクトモデルを以下に示す。

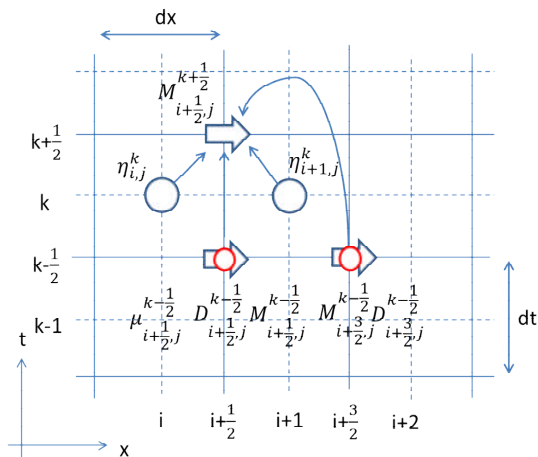


図6 流量オブジェクトモデル X-T視点

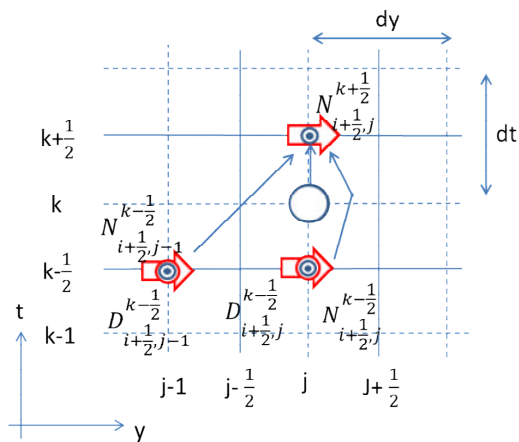


図7 流量オブジェクトモデル Y-T視点

図6より現時間ステップの流量を求めるには上流、下流の水位評価点、前時間ステップの上流、下流の流量評価点、補間された上流、下流の水位評価点が必要bなことがわかる。

図7では、直方向の上流、下流の流量評価点と、補間された上流、下流の水位評価点が必要bなことがわかる。

以上のことから津波算定式は水位及び流量がお互いに作用し合い求められていくことがわかる。

津波数値解析プログラムでは格子間隔の異なる複数の領域を用い計算しており、陸地に近づくほど格子間隔が小さくなっている。格子間隔の異なる領域のイメージを以下に示す。

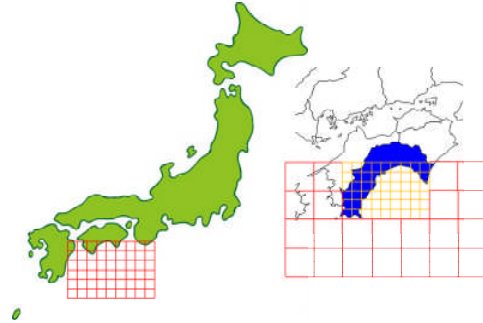


図8 領域イメージ

格子間隔の最も大きい領域では流体の境界外縁での反射を防ぎ、水の出入りを自由にするため、透過境界条件の設定を行っている。そのため上で示した算定式とは別の算定方法で水位の算定を行っている。

透過境界のオブジェクトモデルを以下に示す。

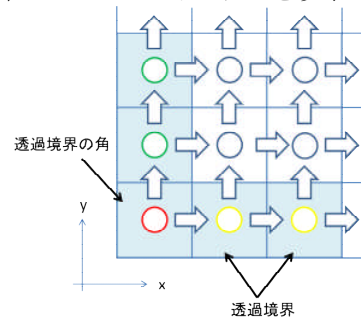


図9 透過境界オブジェクトモデル

上で示したように透過境界では水位一つに対し、流量が三つ、透過境界の角では水位一つに対し、流量が二つである。そのため、透過境界と透過境界の角で、異なるクラスを作る。

格子間隔の異なる領域を複数用い計算する際、計算結果に連続性を持たせるために、接続計算が必要となる。水位の接続は、小領域から大領域へ行われ、計算値に連続性を持たせている。大領域1マスに小領域のグリッドが9マス内包され、内包する小領域9マスの水位変化の平均値を受け渡す。

以下に水位接続境界のオブジェクトモデルを示す。

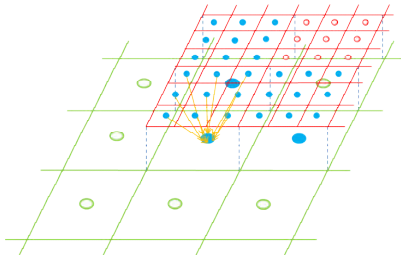


図10 水位接続境界オブジェクトモデル

流量の接続は、大領域から小領域へ行われるため、大領域において計算した流量から外挿計算、内挿計算を行い、時間格子間隔の小さい小領域に流量を与える。

流量接続境界のオブジェクトを以下に示す。

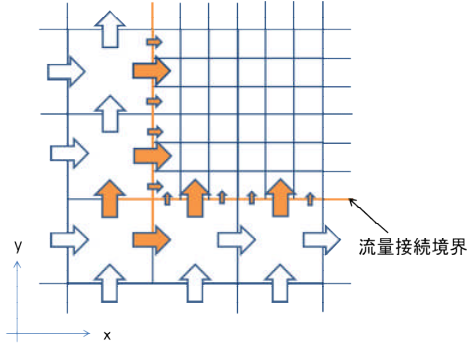


図11 流量接続境界オブジェクトモデル

2.2 クラス構造の確立

オブジェクト分析した津波数値解析プログラムからクラス構造の作成を行う。

水位の計算を行う点をHeight Point(水位評価点)、流量の計算を行う点をVelocity Point(流量評価点)とした。

2.1でも示したように、水位の計算は通常の津波算定式で行う場合や、透過境界条件を使用する透過境界及び、透過境界の角、水位接続境界に分けられるので、Height Point(水位評価点)には、Ordinary Height Point(通常の水位評価点)、Boundary Height Point(透過境界の水位評価点)、Corner Height Point(透過境界の角の水位評価点)、Connection Height Point(接続境界の水位評価点)を作成した。

Velocity Point(流量評価点)では、通常の場合と、接続計算を行う場合の2パターンなので、Ordinary Velocity Point(通常の水流量評価点)とConnection Velocity Point(接続境界の水流量評価点)を作成した。

Boundary Height Point(透過境界の水位評価点)及びCorner Height Point(透過境界の角の水位評価点)では、ポイントの存在する方向により算定に使用する流量が変わるので、Boundary Height Point、(透過境界の水位評価点)のサブクラスとしてEast Boundary Height Point(東の透過境界の水位評価点)、West Boundary Height Point(西の透

過境界の水位評価点)、North Boundary Height Point(北の透過境界の水位評価点)、South Boundary Height Point(南の透過境界の水位評価点)を作成し、Corner Height Point(透過境界の角の水位評価点)のサブクラスとして、NE Corner Height Point(北東の透過境界の角の水位評価点)、NW Corner Height Point(北西の透過境界の角の水位評価点)、SE Corner Height Point(南東の透過境界の角の水位評価点)、SW Corner Height Point(南西の透過境界の角の水位評価点)を作成した。これらをまとめたものをRegion(水位・流量評価帯)とし、複数の格子間隔の異なるRegionをまとめたものをAll Region(水位・流量変動分布)とした。現時点での計算結果はAll Region(水位・流量変動分布)から出力され、次時間ステップの算定に使用される。

2.3 UMLを用いたクラス図の作成

2.2で作成したクラス構造をもとにUMLを使用した津波数値計算プログラムのクラス図を以下に示す。

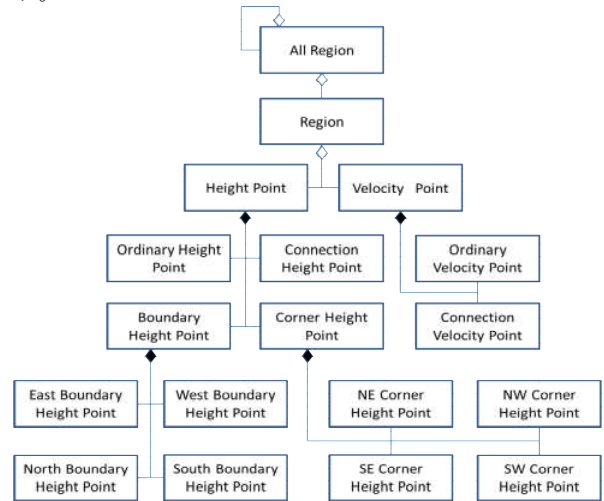


図12 UML クラス図

2.4 C++を用いたコーディング

作成したクラス構造をもとにオブジェクト指向プログラミング言語であるC++によるコーディングを行い作成したクラス構造がプログラムとして成り立っているか確認する。

今回はクラス同士の関係や動きを考えるためヘッダファイルの作成を行った。各クラスが自分の属性を持っているか、メソッドにクラスが行う処理を入れているか、一つのクラスに必要な以上の役割を持たせていないか確認する

3. 結論

オブジェクト指向プログラミング言語であるC++でヘッダファイルを作成し、今回作成したクラス構造が実装可能であることを確認することができた。

今後、解析を行い、作成したクラス構造が津波数値解析プログラムに適したものであるか検討する。