

河川網の入れ子構造を反映した 不定流計算のオブジェクト指向プログラミング

宇都宮大学 学生員 椎谷 成孝
宇都宮大学 正会員 池田 裕一

1. はじめに

1990年頃までは、土木工学分野で計算機の利用というと、数値計算がほとんどであった。ところが、IT革命が謳われるようになり、また社会需要の多様化に伴い、それ以降の土木工学分野での情報技術の活用・発展には目覚ましいものがある。

そのような情勢の中で、Javaは注目すべきコンピュータ言語のひとつである。インターネットとの親和性も高く、最近になって普及してきたオブジェクト指向プログラミング(OOP)もサポートしている。

OOPは、Fortranなど従来の構造化プログラミングとは異なり、プログラムを動作させたい世界がどのような「もの(オブジェクト)」で構成されるか検討していく手法で、人間の認識により近いアプローチで、複雑なシステムを扱うことができる¹⁾。

本研究で取り上げる開水路不定流は、支配方程式自身は複雑ではないが、上下流端の外部境界条件や、堰・横越流などのさまざまな内部境界条件を考慮しようとする、汎用的なプログラムはさほど単純ではなく、分合流を含めた河川網を扱うとなると、かなり複雑なものとなる。

そこで本研究では、この問題をOOPで扱うことによってプログラム全体を単純化し、保守・拡張の容易なアプリケーションの構築が可能であることを、具体的に検証することにする。

2. 河川網の構成要素

(1) 河川網の入れ子構造

河川網の構造を表す場合、単純に考えると、図1(a)のように境界条件となる点(図中の○)と開水路部

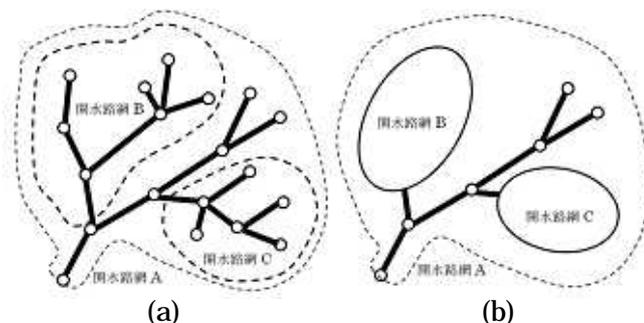


図1 河川網の入れ子構造

分(図中の太線)のみをつないで構成することになるが、構成要素の個数が多くなると河川網の記述が煩雑になってしまう。そこで、本研究では図1(b)のように、部分的な河川網も構成要素として扱い、階層的に河川網を構成できるようにした。そして、河川網の計算処理を行う際には、それに含まれる部分河川網のなかの構成要素の処理をも行うというように、再帰的な処理を組み込むことにした。

(2) 多様な断面オブジェクトによる計算処理

一般に河川網のような構造を扱う場合は、図2(a)のように境界条件を表す点(○)と開水路(太線)からなる木構造として表すことが多い。しかしそれでは、堰などの内部境界点を有する場合には、一つの河川をいくつかの開水路に分割して扱う必要が生じ、実務的に都合が悪い。そこで、本研究では、一つの河川を断面の連続であるとし、その断面の役割(流れの計算や境界条件など)に応じた処理を行う断面オブジェクトを、必要に応じて配置することにした(図2(b)参照)。このような計算処理はOOPでは「多態性の活用」¹⁾といて、きわめて簡便にプログラムすることが可能である。

3. 本研究でのクラス構成

本研究で考えたクラス構成を図3に、各クラスの役割を表1に示す。図1において、三角矢印はその先端に当たるクラスの特徴を根元に当たるクラスが受け継いでいること(継承)を示す。また菱形から伸びる矢印は、そのクラスを1つ以上保持していること(集約)を表す。

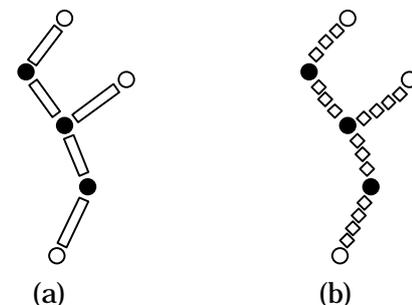


図2 多様な断面オブジェクトによる計算処理

keyword: 河川網, 入れ子構造, オブジェクト指向プログラミング, Java

連絡先 〒321-8585 栃木県宇都宮市陽東7-1-2 宇都宮大学工学部建設学科 TEL 028-689-6214 FAX 028-689-6230

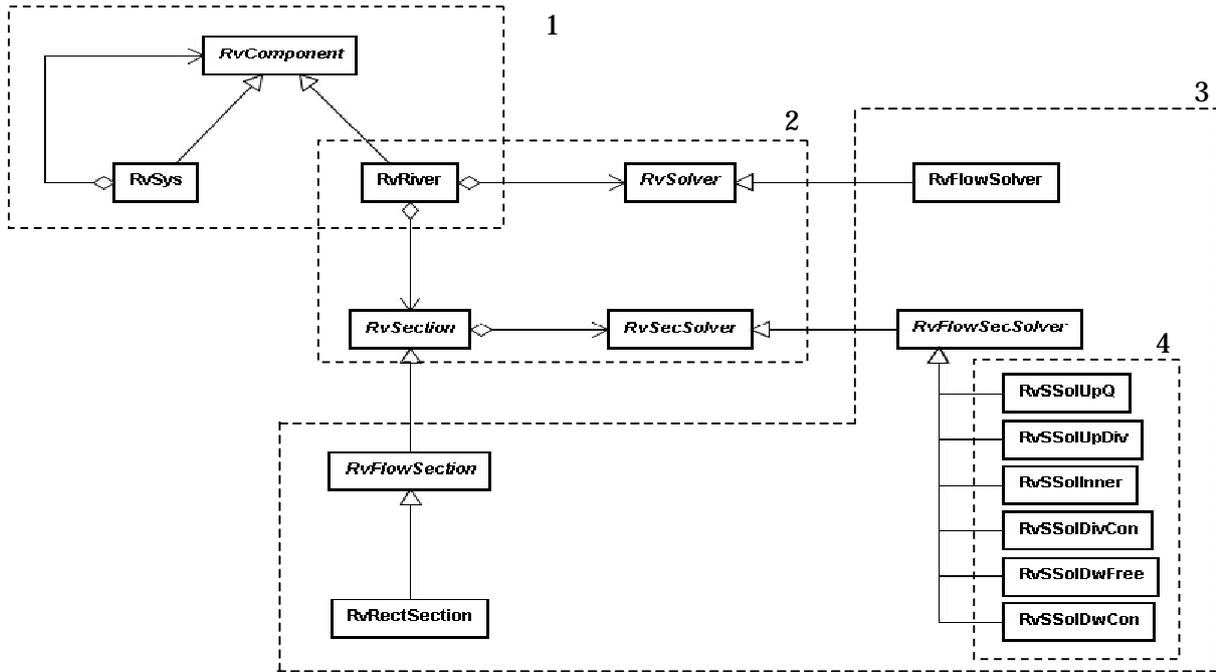


図 3 クラス構成図

(1) 河川網の入れ子構造 (図 3 の 1)

河川網の入れ子構造を実現するために、河川網と単河川の共通部分をもつクラス *RvComponent* をおいた。それを継承するクラスが河川網クラス *RvSys* と単河川クラス *RvRiver* である。

すなわち、河川網は *RvComponent* オブジェクトの集合体であるが、その一つ一つは *RvSys* オブジェクトであったり *RvRiver* オブジェクトであったりする。そのうち *RvSys* オブジェクトはその内部に *RvComponent* を複数保持することになるので、そのなかにさらに *RvSys* あるいは *RvRiver* オブジェクトが含まれることになり、河川網の入れ子構造を実装することができる²⁾。

(2) 単河川の構成と計算処理(図 3 の 2)

単河川を表す *RvRiver* オブジェクトは、断面を表す *RvSection* オブジェクトを必要なだけ保持する。これは実務で管理している断面などが相当する。*RvSection* オブジェクトにはさまざまな役割を与えることになるが、それはそのオブジェクトが保持する *RvSecSolver* オブジェクトを切り替えることで実現する。また *RvSection* の計算処理を単河川ごとにまとめて行う機能は、*RvSolver* オブジェクトが受け持つ。

(3) 具体的な計算方法の記述(図 3 の 3)

本研究で行った洪水計算は、それぞれ 2 部分から継承した *RvFlow* ~ の接頭辞のついた 3 部分のクラス群で行われる。土砂流量計算や濃度計算など、別の計算する際には 3 部分を交換することで行うことができる。

特に、断面オブジェクトでの計算処理については、共通する処理は *RvFlowSecSolver* 中に記述し、さま

表 1 クラス構成表

| | |
|------------------------|-------------------|
| <i>RvSys</i> | 河川網クラス |
| <i>RvRiver</i> | 単河川クラス |
| <i>RvComponent</i> | 抽象的な河川のようなもののクラス |
| <i>RvSolver</i> | 単河川の計算をするクラス |
| <i>RvSection</i> | 断面クラス |
| <i>RvSecSolver</i> | 断面の計算をするクラス |
| <i>RvFlowSolver</i> | 単河川の洪水計算をするクラス |
| <i>RvFlowSection</i> | 洪水計算に必要な変数をもったクラス |
| <i>RvFlowSecSolver</i> | 洪水計算をするクラス |
| <i>RvRectSection</i> | 長方形断面クラス |
| <i>RvSSolUpQ</i> | 流量を数式で与える上流端クラス |
| <i>RvSSolUpDiv</i> | 分流してくる上流端クラス |
| <i>RvSSolInner</i> | 通常の内断面クラス |
| <i>RvSSolDivCon</i> | 分合流する内断面クラス |
| <i>RvSSolDivFree</i> | 自由下流端クラス |
| <i>RvSSolDwCon</i> | 合流していく下流端クラス |

ざまな役割 (境界条件・流れ計算など) に応じて、さらに細かく *RvSSolUpQ* など図 3 の 4 部分に示すような、さまざまなクラスを利用することにする。

以上により、大規模な河川網、流域計算を容易に行うことができる。これらを用いた計算プログラムでの計算結果は発表会当日に発表する。

参考文献

- 1) 高橋麻奈：やさしい Java 第 2 版、ソフトバンクパブリッシング、551p, 2002.
- 2) 結城浩：Java 言語で学ぶデザインパターン入門、ソフトバンクパブリッシング、480p, 2002.