

分散処理環境における洪水シミュレーターの一実現方法

谷川 昌也 †

名城大学理工学部情報工学科 †

高橋 友一 ‡

名城大学理工学部情報工学科 ‡

1 はじめに

東海豪雨など、洪水に関する災害は無視できないものといえる。国土交通省のIT施策で「水情報国土」により個人レベルで雨量や川の水位情報を入手できる [1]。また、地方整備局では管轄の河川での氾濫シミュレーションの表示 [2] や、三重県尾鷲市を対象とした津波災害のシミュレーション結果などが公開されている [3]。

洪水による被害を最小限に抑える対策方法を事前に検討するため、想定されない場所での破堤や集中豪雨による洪水のシミュレーションへの展開が望まれる。

本論文では、洪水や地震などの災害時における避難勧告などによる人的損害への影響をふくめたシミュレータの構築を目指し、分散処理環境下で構築した洪水シミュレーターの一実現方法とその評価について述べる。

2 洪水シミュレータ

2.1 定式化

洪水シミュレータに対する要求項目を以下に挙げる。

1. 任意の出水個所の指定 (破堤, マンホールからの出水, 水道管の破裂)。
2. 救助活動 (修復工事, 排水ポンプの設置) の反映。
3. 避難先として建物。

本方法では計算地域を格子状に区切り, それぞれの格子を { 標高, 水位 } の情報を持つオブジェクト P_{ij} として取り扱う。洪水の広がりには拡散方程式を用いる (式 1)。

$$\frac{\partial w_{ij}}{\partial t} = \alpha \left(\frac{\partial^2 w_{ij}}{\partial x^2} + \frac{\partial^2 w_{ij}}{\partial y^2} \right) + S_{ij} \quad (1)$$

ここで, w_{ij} は格子オブジェクト P_{ij} における標高と水位の和を示す (図 1 左)。 α は拡散係数である。 S は増水や排水を示し, 増水個所では +, 排水個所では -, その他は 0 の値を取る。計算結果が標高に影響を与える場合にはプログラム上で制限する (図 1 右)。

A realization method of flood simulation
in distributed processing environment
† Masaya Tanigawa

Meijo University, Faculty of Science and Technology

‡ Tomoichi Takahashi

Meijo University, Faculty of Science and Technology

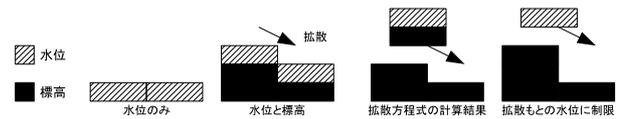


図 1: 洪水の拡散

2.2 震災総合シミュレーションシステム

本シミュレータは震災総合シミュレーションシステム (IDSS) のサブシミュレータとして開発した [4]。本シミュレータを IDSS のサブシミュレータとしてプラグインすることで以下を実現する。

1. 複数のパソコンで並列計算する分散処理。
2. 火災シミュレータなど, 他の災害を計算するシステムや, 市民や救急隊の動きを計算するエージェントなどとの統合。
3. 共有時空間データ (SSTD) でサブシミュレータ間の情報共有。

分散処理では SSTD を複数の地域に分割し, 各ホストに割り当てる。地域間の境界をまたいだシミュレーションを行うために, 隣り合った境界領域で互いのコンピュータがある程度相手の管轄領域を“のりしろ”として保持し交信する。

3 実験

3.1 実験環境

東海豪雨の洪水地域のマップ [6] を用意して実験を行った。使用したパソコンはすべて CPU:Pentium4 3GHz, RAM:2GB のもので, ネットワークはギガイーサーで構成した。

3.2 分散処理環境の実験

名古屋市天白区を対象とし, 実際の計算時間とシミュレーションの進行状態を比較し, 分散処理の効果を検証した。図 3 は計算時間とシミュレーション内の時間のグラフ, 図 4 は計算時間と洪水の広がり具合のグラフを示す。

計算は横 1956m × 縦 1498m の範囲で行った。標高は $z = 10\sin\frac{x}{200} + 10\cos\frac{y}{200}$ の関数で定義した。格子数は横 250 × 縦 192 とし, のりしろの長さは 58m とした。

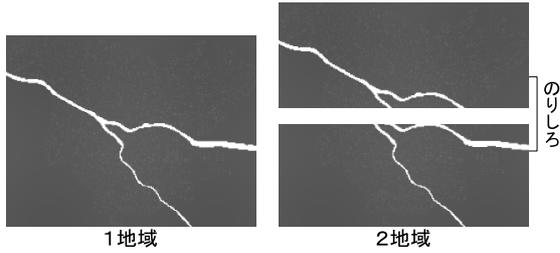


図 2: 地域分散

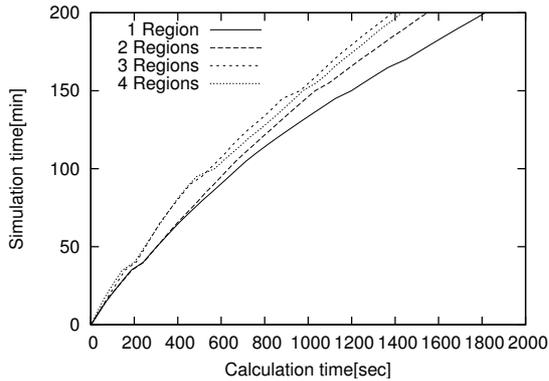


図 3: 実験結果 (シミュレーション内時間での比較)

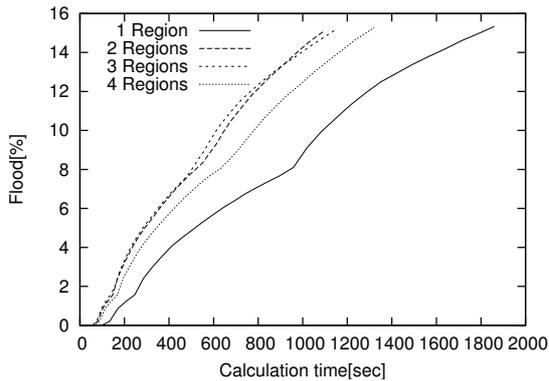


図 4: 実験結果 (洪水の広がりでの比較)

図 2 は 2 地域分散を示し, 3 地域, 4 地域の場合でも等間隔に分割を行った。

通信速度と計算速度のトレードオフから 3 地域分散で最も効果がある。

3.3 洪水状況の評価

名古屋市中川区を対象とし, 図 5 の浸水実績図と図 6 に示すシミュレーション結果との比較で実際の洪水との違いを見た。

計算は横 1821m× 縦 1043m の範囲で行った。標高は $z = \frac{y}{200}$ の関数で定義した。格子数は横 250× 縦 123 とした。計算地域内の各河川が増水したのとしてシミュレーションを行うが, 浸水実績図で浸水が酷い箇所近くの河川が増水が酷かったものとし増水量に違いを持たせた。

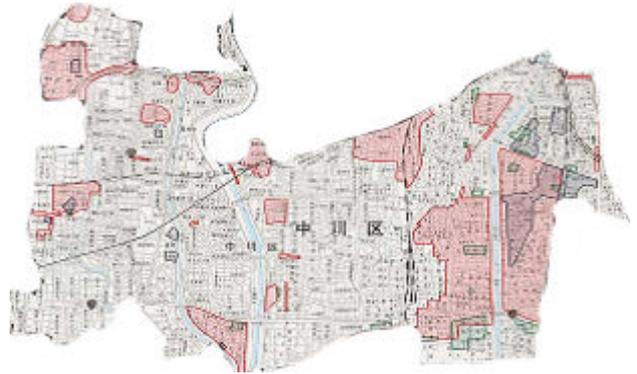


図 5: 浸水実績図 ([5] より)

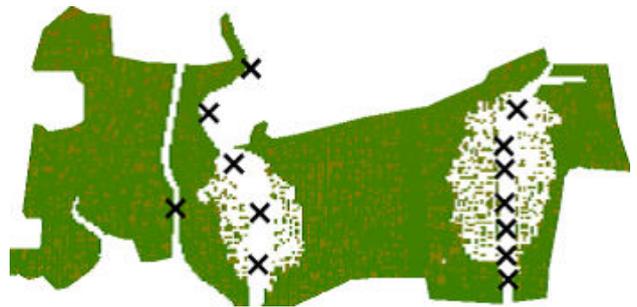


図 6: シミュレーション結果 (設定した増水箇所を×で示す)

マップの右側の部分はほぼ同等の結果が得られたが, 中央付近では違ったものになっている。図 6 の増水箇所の配置と洪水状況から, 増水箇所の配置条件によって結果が大きく左右されている。そのため, 実際の運用ではさまざまな条件を与えて実験を行う必要がある。

また, 図 5 では他の区から広がった洪水も見られるため, 十分広い範囲でのシミュレーションが必要である。

4 おわりに

今回は 1 つの区のみを計算範囲としており, 3 台以上での分散処理による計算速度の向上はなかった。しかしながら, 名古屋市全体などの大きな範囲で計算を行う際には, 計算速度だけでなくメモリ容量の限界からも地域分散が必要不可欠となる。

今後は実際の標高データを用いたシミュレーションで確認する。

参考文献

- [1] <http://www.mlit.go.jp/river/IDC/index.html>
- [2] <http://www.cbr.mlit.go.jp/kisojyo/rootup/top.html>
- [3] <http://www.ce.gunma-u.ac.jp/regpln/katada/kataweb/atop.htm>
- [4] <http://www.kedm.bosai.go.jp/>
- [5] http://www.city.nagoya.jp/13doboku/toukai_gouu/toukaigouu.htm
- [6] 高橋 谷川 高橋 公開地図データにおける建物の自動生成方法 [情報処理学会 1V-8]