

GPGPU を用いた浅水流方程式に基づく数値解析の高速化

岡山大学大学院環境生命科学研究科

学生会員 ○ 田中 龍二

岡山大学大学院環境生命科学研究科

正会員 吉田 圭介

岡山大学大学院環境生命科学研究科

フェロー会員 前野 詩朗

1. 序論

河川や海洋といった広い水域の流れの数値解析において微地形の影響を詳細に検討する際には、局所的に細かい計算格子を用いる場合がある。その際には、格子数を増加させるだけでなく、通常は計算の安定化のために時間刻みを小さく設定する必要があるため、数値計算には多大な時間を要する。しかし、一般に水工学分野において数値計算を行う際には計算機資源や解析に要する時間に限りがある場合が多く、負荷の大きい計算が実際に行われることは少ない。

一方、近年登場したコンピュータグラフィックボード(GPU, Graphics Processing Unit)を用いた汎目的計算(GPGPU, General-purpose GPU)は、解析時間の大幅な低減が期待できる有望な手段の 1 つと考えられている。しかし、水工学分野では適用事例¹⁾は少なく、CPU(Central Processing Unit)を用いた従来型の数値計算に対する GPGPU の優位性は詳細には明らかにされていない。

本研究では 2 次元浅水流計算²⁾を対象に、精度と計算効率の観点から CPU と GPU の計算性能を比較検討した。本研究で用いた GPU は写真-1 に示した Tesla K20c (NVIDIA 社)である。この GPU は画像処理機能のない GPGPU 専用製品であり、倍精度演算が可能である。本研究の開発環境は、OS: Windows7 64bit, CPU: Xeon E5-1620 3.60GHz, メモリ: 16GB, コンパイラ: PGI コンパイラである。CPU での計算は GPU と同じ開発環境で行った。

本研究では 2 種類の数値実験を行った。まず、水際移動のない仮想的な単断面開水路において不等流を対象として平面 2 次元浅水流計算を行い、GPGPU の基本的な演算性能を確認した。その後、自然河川での水際移動の影響を考察するために、旭川・大原箇所を対象に出水時の定常流況を解析して

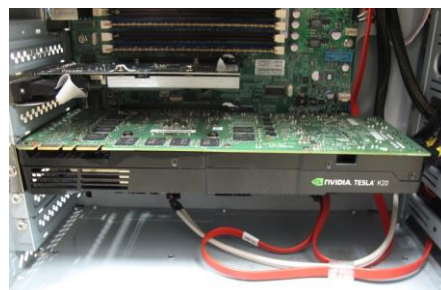


写真-1 研究で用いた GPU (差し込み例)
(上方 GPU: GeForce GT610 (出力用),
下方 GPU: Tesla K20c (GPGPU, 計算用))

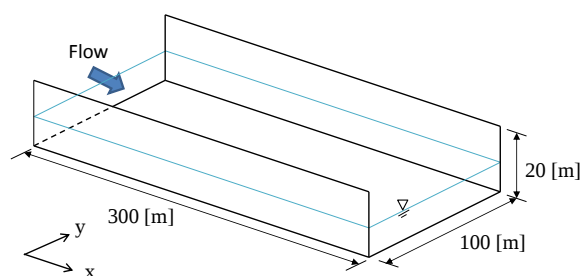


図-1 単断面開水路の概略図

CPU と GPU の演算性能を比較検討した。

2. 解析結果

(1) 単断面開水路への適用

図-1 に示す仮想的な矩形断面開水路に対して一定の流量を通水し、不定流条件下での GPU による計算結果と CPU による計算結果とを比較し、両者の差の程度について調べた。

a) 計算条件

流下距離 300m, 横断距離 100m, 水路勾配 1/600, 水路床のマニングの粗度係数 $n = 0.028 \text{m}^{-1/3}$ の単断面開水路に対して、上流端での定常流量 $Q = 400 \text{m}^3/\text{s}$ および下流端での定常水位 $H = 9.43 \text{m}$ を設定し、平面 2 次元浅水流方程式を用いて不等流計算を行った。計算時間 T は 2hr, 計算時間間隔 Δt は 0.01s であり、流下方向 ξ と横断方向 η に一様な幅 1m の計算格子を設定した。その結果、各方向の格

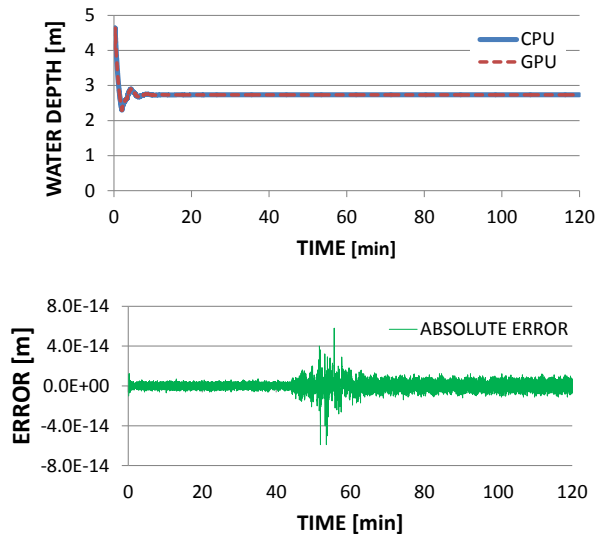


図-2 上流から 150m 地点の水路中央での水深の経時変化

(上:変動値 h_C と h_G , 下:絶対誤差 $\Delta h \equiv h_G - h_C$)

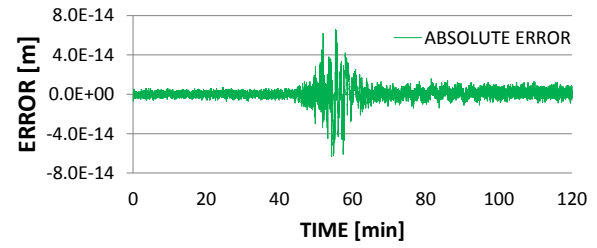
子数は $N_\xi = 300$, $N_\eta = 100$ となり, 総格子数 $N = N_\xi \times N_\eta$ は 30000 である. 浅水流計算手法の詳細は文献 2)を参照されたい.

b) 解析結果

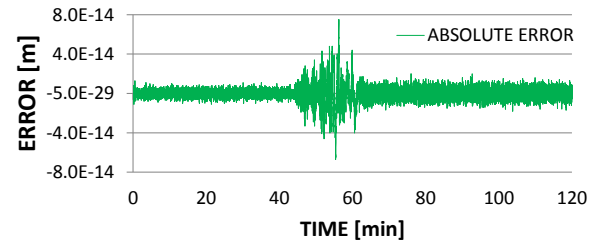
図-2 には上流から 150m 地点の水路中央の水深 h に関して, CPU の計算結果 h_C と GPU による計算結果 h_G を併示した. 図中, データの出力時間間隔は 0.4s である. 誤差 Δh はかなり小さいため, 目視では変動値から誤差は読み取れない. また, 両者の差 $\Delta h \equiv h_G - h_C$ を絶対誤差と定義して図示した. 絶対誤差は $1.0 \times 10^{-13} \text{m}$ 以内であり, 計算開始から約 50 分頃の一部の時間帯を除いてデータは安定している. 計算開始から約 45 分から約 60 分の間では比較的大きな誤差が生じているが, その後はほぼ計算開始直後の誤差と同程度となっている. この原因として, 上下流端の境界条件の設定²⁾の影響が伝搬して誤差を生じたものと推察される. 図-3 には流速値 u, v に関して図-2 と同様のデータを示した. ここで, u, v はそれぞれ流下方向と横断方向の流速の瞬間値である. 流速に関しても水深と同じ傾向であることが分かる.

(2) 実河川への適用

上記の結果で見られた誤差は非常に小さく, また従来の CPU を用いた数値解析が工学上十分な精度



(a) 流下方向流速の誤差 Δu



(b) 横断方向の誤差 Δv

図-3 上流から 150m 地点の水路中央での流速 u, v の絶対誤差 $\Delta u, \Delta v$ の経時変化

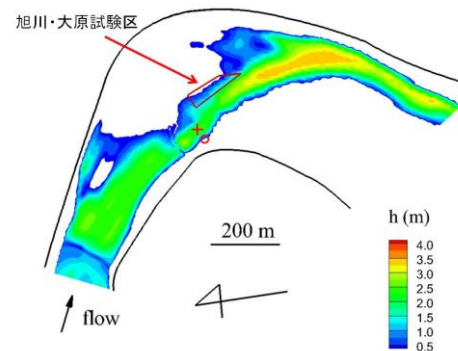


図-4 旭川・大原試験区を含む対象領域の水深分布 (着色部は水域, ×印及び○印は水深のモニタリング箇所)

を有する限り, 水際の移動がない開水路流れの解析を行う上では GPU の計算精度は十分高いものと考えられる. 一方, 実河川への適用を考える場合, この小さな誤差が原因で水際での水域・陸域判定²⁾において条件分岐が生じ, その結果 CPU と GPU の計算で比較的大きな誤差を生じることが懸念される. ここでは, 水際移動を含む実河川の定常流況解析を行い, 水深の経時変化を対象に CPU と GPU による計算結果を比較し, GPU による計算の精度と速度を調べた. ここでは, 図-4 に示す旭川・大原試験区を含む河道縦断 2km 程の範囲の流動解析に適用した.

表-1 各ケースの計算格子

	(ξ, η) 方向の 格子数	格子総数
Case1	69 × 29	2001
Case2	135 × 57	7695
Case3	205 × 85	17425
Case4	273 × 113	30849

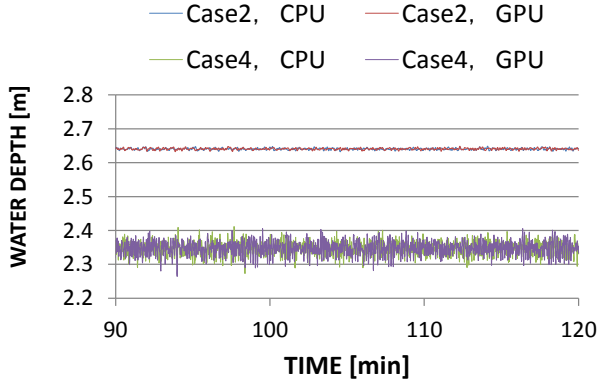


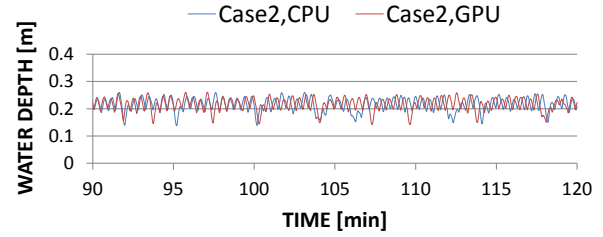
図-5 水域(×印)での水深の経時変化 (実時間 90 分から 120 分)

表-2 Case2 及び Case4 での水深変動の平均値と標準偏差

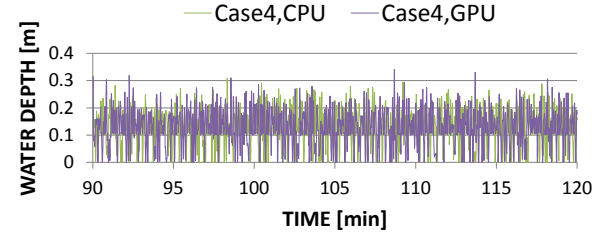
	Case2		Case4	
	CPU	GPU	CPU	GPU
平均値	2.6403	2.6406	2.3470	2.3475
標準偏差	0.0026	0.0026	0.0182	0.0195

a) 計算条件

現地の小規模出水時(年 2 回程度)の定常流況を想定し、上流端での定常流量 $Q = 400\text{m}^3/\text{s}$ 、下流端での定常水位 $H = 9.43\text{m}$ を設定して非定常計算を行った。計算時間 $T = 2\text{hr}$ 、計算時間間隔 $\Delta t = 0.01\text{s}$ 、陸域・水域境界の判定基準水深²⁾ $h_* = 0.1\text{m}$ であり、低水路、高水敷のマニング粗度係数 n については既往研究³⁾を参考にそれぞれ 0.028, 0.026 とした。解析では同一の計算領域に対して、流下方向 ξ と横断方向 η の格子分割数を変化させて、表-1 に示す 4 ケースで計算を行った。各ケースでの河床高は Case1 でのデータを基にして線形内挿して求めた。また、解析では図-4 の水域内の×印及び○印の地点に最も近い格子(最近傍格子)における水深 h の時間変化をモニタリングした。ここで、各ケースで最近傍格子点の座標や河床高は異なるため、ケース間で水深の時間平均値が異なることに注意する必要がある。



(a) Case2



(b) Case4

図-6 水際(○印)での水深の経時変化 (実時間 90 分から 120 分)

表-3 実時間 2 時間に要する解析時間及び高速化率

		CPU	GPU	高速化率
	Time[s]	594.40	586.90	1.01
Case1	換算	9.9 [min], 0.17 [h]	9.8 [min], 0.16 [h]	
	Time[s]	2535.09	656.05	3.86
Case2	換算	42.3 [min], 0.7 [h]	10.9 [min], 0.2 [h]	8.49
	Time[s]	6524.53	768.54	12.87
Case3	換算	109 [min], 1.8 [h]	12.8 [min], 0.2 [h]	
	Time[s]	12568.81	976.77	
Case4	換算	209 [min], 3.5 [h]	16.3 [min], 0.3 [h]	
	Time[s]			

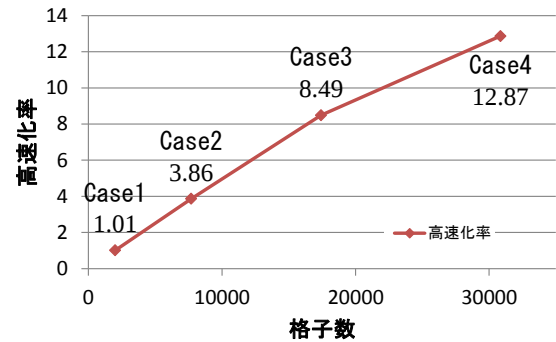


図-7 高速化率と計算格子数の関係

b) 解析結果

図-5 に Case2 及び Case4 での水域での水深の経時変化について、CPU による計算結果と GPU による計算結果を併示した。また、表-2 には流況がほぼ平衡状態に達した時間帯(解析時間 90 分から 120 分の間)における水深の平均値と標準偏差を示した。CPU と GPU の計算結果を比較すると、水深変動の挙動は比較的似ている。例えばこの時間帯の統計量で見た場合、Case2 では約 2.64m の水深に対

して 3mm 未満の変動であり、Case4 では約 2.34m の水深に対して 2cm 未満の変動を示している。

各ケースでの高速化率を表-3 に示す。ここで高速化率とは CPU による計算時間を、GPU による計算時間で割った値と定義する。高速化率はケースで異なり、Case4 において最高の 12.9 倍の高速化が達成された。高速化率が異なるのは、用いた計算格子数によるものと考えられる。つまり、格子数が少ない場合は GPU 内の計算コアの稼働率が低くなっているのではないかと推測される。また、Case4 の GPU を用いた場合の解析時間が Case2 の CPU のみの解析時間を大幅に下回っていることが分かる。つまり、本節で示した計算精度を維持しつつ、約半分の格子幅の計算格子を用いて大幅な解析時間の短縮が行えることを示している。

図-7 に高速化率と計算格子数の関係を示す。線形ではないが、高速化率は格子数の増加に伴い増加することが分かる。さらに格子数を増やした際この傾向が現れるかは不明である。なお、格子数が最も少ない格子幅約 20m の場合、高速化率が約 1.0 で CPU の解析時間と同程度になっており、全ての条件で高速化が行えるわけではない。

3. 結論

本研究では境界適合座標法を用いた、非定常平面 2 次元浅水流解析を対象に、計算時間と精度の観点から GPU の性能を検討した。得られた主要な結論を以下に示す。

- 1) 水際移動のない矩形単断面開水路の不等流計算を対象にした基本的な数値実験において、GPU による並列計算結果と CPU による計算結果を比較した結果、水路中央での両者の水深の差（誤差）は $1.0 \times 10^{-13} \text{m}$ 以内であり、GPU を用いて非常に精度よく計算が行えた。
- 2) 水際移動の生じる、旭川・大原箇所を含む実河川を対象にして両計算結果を比較した結果、GPU による計算結果は CPU による計算結果に対して上記 1) と同じ程度には一致しなかった。これは上記 1) の微小な誤差が原因で、水際において水域・陸域判定に関する条件分岐が生じ、水面変動やそれに基づく流れが生じるためである。ただし、水深変動の時間平均値や標準偏差値から判断すると、GPU による計算は CPU による計算と同程度の精度で行

われており、河川工学的な用途では十分な精度を有する。

- 3) GPU による計算の高速化率は計算格子数の増大に対してほぼ線形に増加し、総格子数が約 30000 の場合で約 12.9 倍の高速化が達成された。ただし、高速化率と計算格子の線形関係は、本研究で調べた GPU と格子数の範囲に限り指摘が可能である。他の GPU を使ったり、格子数をさらに増加させた際の特性は不明である。

今後の研究課題としては、1) GPU のメモリを有効に使えておらず、既往の研究で示されている高速化率に比べても、高速化率は低い値であること。また、2) 研究の第 1 段階として流れの計算のみを扱い、実河道での水理解析で必要となる植生や河床変動などは考慮されていないことが挙げられる。今後はこれらを考慮した解析を行い、計算精度及び高速化率の検証が必要である。

参考文献

- 1) 後藤仁志，堀智恵美，五十里洋行，Abba KHAYYER：GPU による粒子法半陰解法アルゴリズムの高速化，土木学会論文集 B，Vol.66，No.2，pp.217-222，2010。
- 2) 長田信寿：一般座標系を用いた平面 2 次元非定常流れの数値解析，水理公式集 例題プログラム集，2001。
- 3) 前野詩朗，吉田圭介，松山悟，藤田駿佑：旭川大原試験区における植生消長シミュレーションモデルの構築と検証，土木学会論文集 B1(水工学)，Vol.70，No.4，I_1369-I_1374，2014。