

ステンシル計算に基づく科学技術シミュレーションのハードウェアアクセラレーション

Hardware-based Acceleration of Scientific Simulation with Stencil Computing

佐野健太郎

Abstract

数値流体力学計算などの科学技術シミュレーションでは、ある決まった近傍領域（ステンシル）内のデータを参照しながら計算格子を繰り返し更新するような計算を行う場合が多い。そのような計算は、バッファメモリを用いて参照データを再利用するストリーム処理専用ハードウェアにより高速化できる。本稿では、ステンシル計算のためのハードウェアアクセラレーション技術について概説する。

キーワード：科学技術シミュレーション，ステンシル計算，ストリーム処理，専用ハードウェア，高性能計算

1. 背景

流体力学計算，電磁界解析計算，分子動力学計算，量子化学計算，多体問題計算をはじめとする数値シミュレーションは，ものづくり分野に加え，次世代医療・材料・エネルギー・防災減災・宇宙物理学といった科学技術分野において重要かつ不可欠な技術として認識されている。大規模かつ複雑なシミュレーションの実行には速度と規模の面で高性能な計算が必要のため，通常，多数の計算ノードをネットワークにより相互に接続した並列計算機であるスーパーコンピュータが用いられる。しかしながら，スーパーコンピュータの規模は数万ノード以上にも達しており，システム全体の消費電力を現実的な範囲に抑えながら高性能計算を実現するために，計算性能のみならず電力性能の高いシステムが求められている。

現在の一般的なスーパーコンピュータは，CPU のみのホモジニアスアーキテクチャか，または CPU に GPU や Xeon Phi 等のメニーコアアクセラレータを組み合わせたヘテロジニアスアーキテクチャに基づいている。近年のマルチコア CPU は，従来のシングルコア CPU と

異なりコア単体の性能を無理に向上させずに複数コア全体の性能を高めるような設計に傾きつつあるものの，シングルスレッドの実行レイテンシ短縮を目標としたままでは電力性能比の大幅な改善は困難であり，チップの電力制約が性能向上の壁となっている。一方，メニーコアアクセラレータは，複数スレッドによるレイテンシ隠蔽により全体の処理スループットを向上させるように設計されており，並列性の高い特定の計算問題に対して優れた性能と電力性能比を実現可能である。特に，近年のアクセラレータは，数千もの計算コアに加え 1 TByte/s に迫る広帯域の外部メモリにより高いピーク性能を有している。

しかしながら，そのようなアクセラレータを用いたとしても様々な計算問題に対して高い実効性能を実現するのは困難であり，場合によってはピーク性能の僅か数%しか得られないことがある。このような非効率性は外部メモリの帯域不足やメモリ階層システムの不適合により引き起こされるコアの稼働率低下に起因する。計算カーネルにおける外部メモリ参照データ当りの演算回数を演算密度と呼ぶ⁽¹⁾。演算密度がプロセッサのメモリ帯域・ピーク性能比よりも低い場合，演算器へのデータ供給が不十分となりコアの稼働率が低下する。ループアンローリングやテンポラルブロッキングといったキャッシュメモリを有効利用し演算密度を高める工夫が可能ではあるものの，計算において求められる様々なデータ移

佐野健太郎 正員 東北大学大学院情報科学研究科情報基礎科学専攻
E-mail kentah@caero.mech.tohoku.ac.jp
Kentaro SANO, Member (Graduate School of Information Sciences, Tohoku University, Sendai-shi, 980-8579 Japan).
電子情報通信学会誌 Vol.100 No.2 pp.82-86 2017年2月
©電子情報通信学会 2017

```

1: for(n=0; n<T; n++) { // T 反復 (時刻ループ)
4:   for(i=0; i<X; i++) // 2次元格子の走査の
3:     for(j=0; j<Y; j++) { // ためのループ
5:       // 次の時刻の格子点データ vn[i][j] を計算
5:       vn[i][j] = F(v[i+1][j], v[i-1][j],
5:                   v[i][j+1], v[i][j-1], v[i][j])
6:     }
8:   swap(v, vn) // 格子データの入れ換え
7: }

```

図1 反復ステンシル計算の擬似コード 現在の時刻の格子データ v から次の時刻のデータ vn を計算する。格子点 (i, j) の計算では、 3×3 スターステンシル内の五つの格子点を参照している。

動に対してメモリ階層は必ずしも適しているとは限らない。例えば、異なるコア間のデータ移動が、共有キャッシュメモリや外部メモリを経由するために非効率となる場合がある。

これらの問題は、様々な計算を対象とした汎用な設計により生じたものである。特に、外部メモリや多数の演算器・データパスの間におけるデータ移動はそれぞれの計算問題に対して最適化されておらず、コアの稼働率低下や無駄な電力消費の原因となっている。これに対し、回路再構成可能デバイスである FPGA は、対象問題に適合した効率の良いデータパスやメモリシステムを構成することにより、汎用のアクセラレータよりも高性能かつ低電力の計算処理が期待できる。

かつてシステム開発の脇役として回路エミュレーション等に利用されていた FPGA は、大規模化やアーキテクチャの改良が進み、近年では、最先端の半導体技術により製造されシステムの中核を占める高性能半導体デバイスとして扱われている。特に、現在の FPGA は、論理要素や配線資源のほかに、メモリブロック、整数演算器である DSP ブロック、高速シリアル伝送のための I/O ブロック、メモリコントローラ等の様々な固定回路 (ハードマクロ) を多数搭載しており、プログラマブルシステムオンチップ (SoC) とも呼べる構造を持つ。特に、最新世代の FPGA には浮動小数点演算にも対応した DSP ブロックが搭載されており、その結果、1~10 TFlops ものピーク演算性能を持つと言われている。また、パッケージ内で 2.5 次元実装された HBM (High-Bandwidth Memory) により 1 TByte/s もの DRAM メモリ帯域を持つ次世代 FPGA も発表されるなど^{(2),(3)}、GPU の後じんを拝していた数値計算性能や外部メモリ帯域の点でも魅力的なデバイスとして再び脚光を集めている。

しかしながら、FPGA の潜在能力を引き出すには、対象とする計算問題に対してオンチップハードウェア資源を効率的に利用する適切なハードウェアアーキテク

チャが求められる。本稿では、科学技術シミュレーションの代表的な計算カーネルである反復ステンシル計算を対象として、ストリーム計算に基づく専用ハードウェアの設計とその事例を紹介する。

2. ステンシル計算とそのストリーム化

2.1 反復ステンシル計算

反復ステンシル計算は、時間進行法による数値流体物理学計算や連立方程式の反復解法等の科学技術計算においてよく見られる代表的計算の一つである。二次元格子データによる反復ステンシル計算の擬似コードを図1に示す。これは、時刻 n のデータを用いた計算により次の時刻 $(n+1)$ のデータを得るという簡単な反復計算の例であり、格子データを走査するための二重ループと、時刻の反復のための最外ループから成る。

ループボディの関数 $F(\)$ は、現在の時刻のデータ $v[][]$ を参照して格子点 (i, j) における次の時刻のデータ $vn[][]$ を求める計算を表す。この計算は、各格子点に対しステンシルと呼ばれる近傍領域内のデータを参照して行われる。このような計算をステンシル計算と呼ぶ。一般に、ステンシルは格子点の位置によらず不変である。格子点 (i, j) に対するステンシルを $S_{i,j}$ と表す。図1の擬似コードは、図2のような中央と上下左右の5点から成る 3×3 スターステンシル $S_{i,j} = \{v_{i,j}, v_{i+1,j}, v_{i-1,j}, v_{i,j+1}, v_{i,j-1}\}$ による計算を表している。

2.2 計算のストリーム化

逐次的に供給されるデータ列をデータストリームと呼び、データストリームの各要素に対し次々と計算を行う方式をストリーム計算と呼ぶ。ストリーム計算は、規則的かつ連続したアドレスの読出しによりメモリ帯域を効率的に利用できる。

反復ステンシル計算は、ステンシルバッファと呼ばれる循環バッファを利用することによりストリーム化でき

る。図3は、 $X \times Y$ の大きさの二次元格子データを j 方向、 i 方向の順番に走査して得られるデータストリームに対し 3×3 スターステンシル計算を行う様子を表す。図では、格子点 (i, j) のデータ $v_{i,j}$ がデータストリームとして入力されている。直近の $2Y$ 個の入力データを図中において黄色で示した循環バッファに格納することにより、バッファに含まれるステンシル $S_{i-1,j-1}$ のデータを参照して、格子点 $(i-1, j-1)$ のステンシル計算を行うことができる。これは、格子点データを逐次読み出すごとに1格子点のステンシル計算を並行して行えることを表しており、複数格子点を参照する計算のストリーム化を意味する。

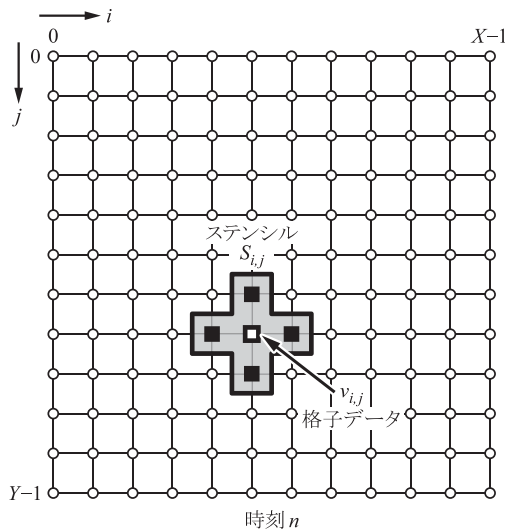


図2 二次元格子と 3×3 スターステンシル $S_{i,j}$

また、時刻 n におけるステンシル $S_{i-1,j-1}$ の計算結果 $v_{i-1,j-1}$ は次の時刻の計算に対するデータストリームとなることから、同様に、時刻 $(n+1)$ におけるステンシル $S_{i-2,j-2}$ の計算を並行して行うことができる。これは、連続する時刻 $n, (n+1), (n+2), \dots$ のそれぞれにおける $S_{i,j}, S_{i-1,j-1}, S_{i-2,j-2}, \dots$ のステンシル計算をパイプライン処理可能であることを意味する。パイプライン初段と最終段でのみ外部メモリとのデータ入出力を行うため、一度のストリーム計算で扱う時刻数を増やすことにより、メモリ帯域への要求を一定としながら計算の並列度を増やすことができる。

3. 専用ハードウェアアーキテクチャ

ストリーム化された反復ステンシル計算に対し高スループットのデータ入出力及び演算を実現する手段として、FPGAを用いた専用ハードウェアが有望である。高スループット計算の実現には、多数の演算を並行して実行可能なデータフロー型のハードウェアが適している。本研究では、データフローにおける演算器単位の細粒度並列性とパイプライン単位の粗粒度並列性の両方を備えた、図4のストリーム計算要素(SPE)アレーアーキテクチャを提案している。各SPEは、ステンシルバッファと計算パイプラインから構成される。各パイプラインは、浮動小数点演算等を単位として要素計算のデータフローをパイプライン化した回路である。複雑な問題の場合には、SPE内でも複数のバッファとパイプラインが使用される。

各SPEは、 n 個のパイプラインを持つことにより空間的並列性を利用して同時に複数の格子点データを計算

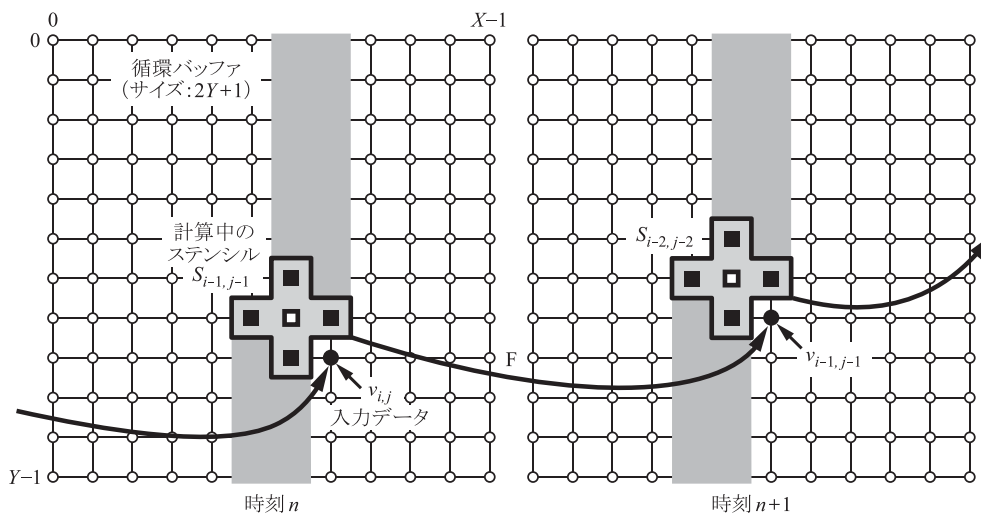


図3 ストリーム化された反復ステンシル計算 入力された格子点データを2列分の大きさの循環バッファに格納することにより、ストリームデータに対してスループット1でステンシル計算を実行できる。計算結果を次の時刻の計算へのデータストリームとみなすと、異なる時刻の計算をパイプライン的に実行できる。

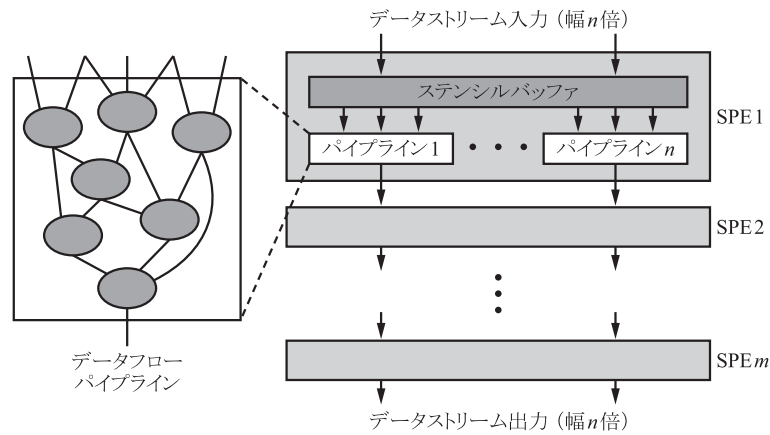


図4 ストリーム計算要素 (SPE) のアレー 時間と空間の両方に対し、各パイプラインはデータフローに基づく細粒度の並列性を、SPE アレーは粗粒度の並列性を利用している。

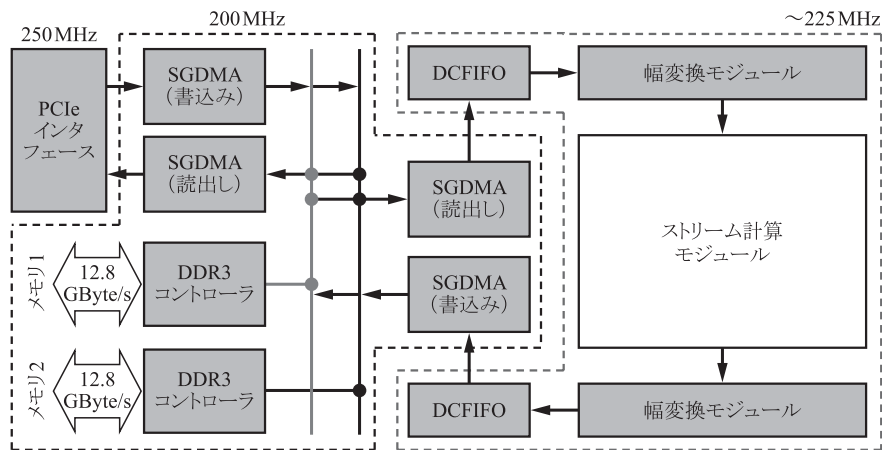


図5 FPGA に実装したアクセラレーションフレームワーク 問題ごとに書き換わるのはストリーム計算モジュールである。二つの DDR3 メモリはそれぞれ 12.8 GByte/s のピーク帯域を持つ。

する。更に、 m 個の SPE を数珠つなぎにすることにより、時間的並列性を利用して複数の時刻の計算をパイプライン実行する。データストリームが十分に長い場合、動作周波数を f とすると本アーキテクチャの計算性能は $O(nmf)$ となる。高性能を実現するには利用可能なハードウェア資源を利用して総パイプライン数 nm を最大化することが求められる。ただし、空間的並列性には n 倍のメモリ帯域が必要となるため、利用可能なメモリ帯域の範囲内で n を増加させた後に m を増やすという戦略が基本となる。

4. 事例

浮動小数点演算対応の DSP ブロックを搭載した最新の Arrial0 FPGA (10AX115N3F45I2SG) を用いて、格子ボルツマン法に基づく二次元流体シミュレーション⁽⁴⁾

と津波シミュレーション^{(5),(6)}それぞれのハードウェアアクセラレータを開発した。FPGA に実装したフレームワークシステムを図5に示す。DDR3 メモリコントローラを含むメモリバス部分は 200 MHz で動作し、SGDMA (Scatter-Gather DMA) によりデータ転送が行われる。二つの DDR3-1600 メモリのピーク帯域はそれぞれ 12.8 GByte/s である。計算部分は 225 MHz で動作し、クロックドメインをまたぐために DCFIFO (Dual-Clock FIFO) を使用している。

ストリーム計算モジュールの実装には、本研究において開発している高位合成コンパイラ SGen (Stream Processor Generator)⁽⁷⁾を用いた。SGen は、独自のストリーム処理記述言語により階層的に書かれた数式や関数呼出しの記述から、図4のハードウェアを表す Verilog-HDL (Hardware Description Language) コード群を生成する。これを FPGA CAD ソフトウェアを用いて

表1 演算器数, 実装 SPE 数及び最大実効性能

	SPE 当り演算器数					実装 SPE 数	最大性能 (GFlops)
	加算	乗算	除算	平方根	合計		
流体	70	60	1	0	131	18	525
津波	147	121	12	2 (8 換算)	288	6	383

コンパイルすることにより, FPGA に書き込み可能なビットストリームが生成される。

表1に, 流体及び津波シミュレーションのそれぞれに対する単精度浮動小数点演算器数, 単一 FPGA に実装された SPE 数, 測定された最大実効性能を示す。パイプライン段数に対してデータストリーム長が不十分なために生じる性能低下は1%程度であり, SPE 数にほぼ線形に比例する性能向上が得られた⁽⁵⁾。その結果, 流体と津波のそれぞれに対して525 GFlops と383 GFlops の実効性能が得られた。実装に使用した Arria10 FPGA を搭載する TERCASIC 社製 DE5A-NET ボードの電力を測定したところ, 流体と津波計算の電力当り性能はそれぞれ9.47 GFlops/W と8.41 GFlops/W であった。特に津波シミュレーションに関しては, 単精度浮動小数点演算のピーク性能が4 TFlops のミドルクラス GPU である AMD 社 Radeon R9 280X GPU に同等の実装を行ったところ, Arria10 FPGA の計算性能及び電力当り性能はそれぞれ1.7倍及び7.0倍高いことが明らかとなった。

5. おわりに

浮動小数点演算器が多数搭載された最新の FPGA は, 高性能かつ低電力な科学技術計算を実現する手段として有望である。データフロー及びストリーム計算に基づく反復ステンスル計算アクセラレータは, 二次元流体シミュレーション, 及び津波シミュレーションに対して数百 GFlops もの実効性能を達成し, 特に津波計算の場合には GPU を上回る計算性能と電力性能比を実現した。

次世代の Stratix10 FPGA は単精度で10 TFlops に迫るピーク演算性能を持つと言われており, 更なる性能向上が期待できる。一方, OpenCL コンパイラが提供されるなど環境は改善されつつあるものの, 依然としてプログラミングの生産性向上が FPGA 利用の大きな課題であり, 今後ますますの研究開発が求められている。

文 献

- (1) S. Williams, A. Waterman, and D. Patterson, "Roofline: An insightful visual performance model for multicore architectures," Commun. ACM, vol. 52, no. 4, pp. 65-76, 2009.
- (2) M. Langhammer and B. Pasca, "Floating-point DSP block architecture for FPGAs," Proc. the ACM/SIGDA International Symposium on Field-Programmable Gate Arrays, pp. 117-125, 2015.
- (3) D. Lewis, G. Chiu, J. Chromczak, D. Galloway, B. Gamsa, V. Manohararajah, I. Milton, T. Vanderhoek, and J.V. Dyken, "The stratix 10 highly pipelined FPGA architecture," Proc. the ACM/SIGDA International Symposium on Field-Programmable Gate Arrays, pp. 159-168, 2016.
- (4) K. Sano, Y. Kono, H. Suzuki, R. Chiba, R. Ito, K. Koizumi, and S. Yamamoto, "Efficient custom computing of fully-streamed lattice Boltzmann method on tightly-coupled FPGA cluster," ACM SIGARCH Computer Architecture News, vol. 41, no. 5, pp. 47-52, 2013.
- (5) K. Nagasu, K. Sano, F. Kono, and N. Nakasato, "Performance and power evaluation of FPGA-based tsunami simulator using floating-point DSPs," Proc. IEEE Symposium on Low-Power and High-Speed Chips (COOLChips XIX), poster paper#17, 2016.
- (6) K. Nagasu, K. Sano, F. Kono, and N. Nakasato, "FPGA-based tsunami simulation: Performance comparison with GPUs, and roofline model for scalability analysis," J. Parallel Distrib. Comput., to appear.
- (7) K. Sano, "DSL-based design space exploration for temporal and spatial parallelism of custom stream computing," Proc. the Second International Workshop on FPGAs for Software Programmers, 2015, <http://arxiv.org/abs/1509.00040>

(平成28年8月25日受付 平成28年9月15日最終受付)



さの けんたろう (正員)
佐野 健太郎 (正員)

2000 東北大大学院情報科学研究科博士課程了。2001 同助手。2005 同助教授。現在同准教授。2006 インベリアルカレッジロンドン客員研究員。博士 (情報科学)。カスタム高性能計算機に関する研究に従事。情報処理学会, IEEE, ACM 各会員。国際シンポジウム HEART ステアリング委員。