Locality-Sensitive Hashing のスケーラブルなハードウェアアーキテクチャの FPGA 実装

A Scalable FPGA-based Architecture of Locality-Sensitive Hashing

定久 紀基 Tsunaki Sadahisa　山本 佳生 Kasho Yamamoto　金 多厚 Dahoo Kim　福田エリック駿 Eric S. Fukuda　浅井 哲也 Tetsuya Asai　本村 真人 Masato Motomura

北海道大学大学院　情報科学研究科
Graduate School of Information Science and Technology, Hokkaido University

１．背景

Locality-Sensitive Hashing(LSH)は、従来のハッシュ法とは異なり、類似したデータが近いハッシュ値となるハッシュ法である。その特性から大量のデータを高速に分類できるため、ビックデータの処理に応用できる（例えば、大規模データの検索時に高速にデータを分類して、不要なデータを捨てる事ができる）。またハッシュ値の計算に複雑な乗算、除算を含まないため HW に向いている。これまで、高速化のために LSH を HW 実装し活用する研究はあるが、特定のアプリケーションを想定して、決まった入力データを効率良く演算するよう演算部を最適化されている。したがって、特定の形式の入力データでしか使用できず、他のデータ形式に対応することは困難である。そこで本研究では、用途によって変わる様々な次元の入力データに容易に対応できる拡張性を持つ LSH アーキテクチャを提案する。

２．Locality-Sensitive Hashing の概要

本研究で実装するのは、Datar らによるユークリッド空間での LSH[1]であり、ハッシュ値を入力データと正規分布を持つ乱数列から選ばれたベクトルとの内積によって計算する。ハッシュ関数はデータ空間を直線または超平面で分割する。したがって、あるデータに近いデータを探すときには、そのデータからハッシュ値を計算しアクセスするだけで類似したデータを引き出すことができる。

３．提案するアーキテクチャ

本アーキテクチャでは多様なデータに対応するためのアイデアとして、Linear Feedback Shift Register(LFSR)によって作られた一様乱数を中心極限定理に従い足し合わせて正規分布を持つ乱数列を生成する（図１左側）。

LFSR は長い周期を持ちそれぞれの状態が異なる値を持つため一様乱数として良く、状態遷移が確定的であるため同じ初期状態を与えてやれば同じ乱数列を作ることができる。そのため、LFSR を計算のたびに同じ初期状態にリセットすることで、毎データ同じランダムベクタを用いた内積ができ、LFSR のビット幅を長めに用意することで多様なデータの形式に容易に対応することができる。

一様乱数を足しあわせて正規分布を作る際には乱数列を最低 4 つ程度用意すればよいことが分かっている[2]。LFSR は、出力線を入れ替えると全く相関が無い乱数列を得られるため、一つのモジュールの出力を並べ替えて複数の乱数列を用意できる。これらを足し合わせることで、メモリに所望の乱数列を保存しておくアプローチに比べ、桁違いに少面積で正規分布に従った乱数列を用意する方法を考案した。

18bitLFSR からこの方法によって作られた乱数列を使い、ハッシュ値計算、ハッシュテーブルでの照合といった LSH のアルゴリズムをハードウェア上で実装する（図１右側：ハッシュ計算部）。これらのアーキテクチャを Xilinx Virtex5　LX330T に実装するために予備評価を行った結果を表１に示す。

表１：モジュールごとの論理合成結果

	Reg	LUT	BRAM	Freq
ハッシュ計算部	138	445	4.644Mb	100.5MHz
LFSR	584	2,467	0	100.4MHz

文　献

[1] M. Datar, N. Immorlica, P. Indyk, and V. Mirrokni. Locality sensitive hashing scheme based on p-stable distributions. In Proc. ACM Symp. on Computational Geometry, pages 253–262, 2004.

[2] 脇本和昌，“正規乱数列のつくり方，”乱数の知識，pp.74-85，（社）森北出版，1970.

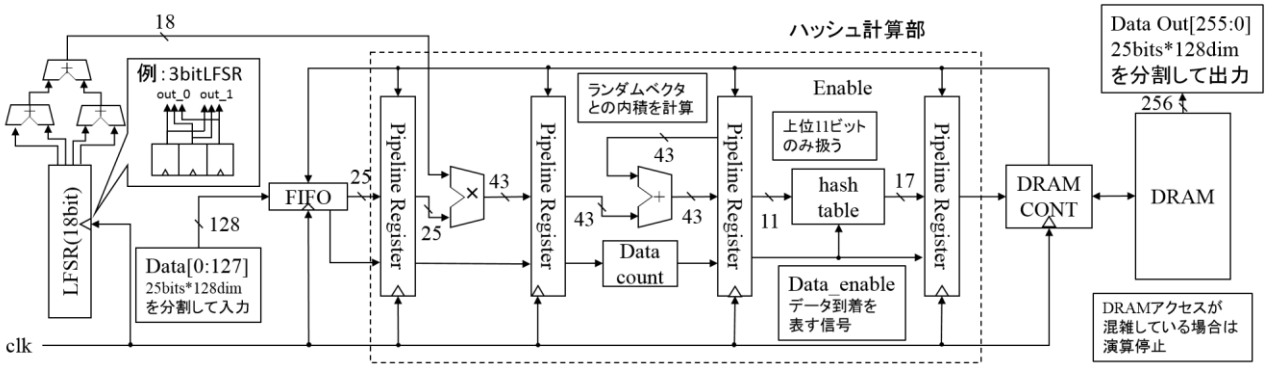

図１：提案するアーキテクチャ