

カオスポルツマンマシンの デジタルハードウェア実装に関する検討

川島 一郎[†] 上ノ原 誠二[†] 加藤 孝史[†] 山口正登志[†]

鈴木 秀幸^{††} 森江 隆[†] 田向 権[†]

[†] 九州工業大学大学院生命体工学研究科

^{††} 東京大学大学院情報理工学系研究科

1. はじめに

近年、多層に積み重ねたニューラルネットワークを用いて学習を行うディープラーニングが注目を集めている[1]. ディープラーニングで用いられているニューラルネットワークの1つとして制限付きボルツマンマシンがある. しかし、制限付きボルツマンマシンが必要とする演算量は膨大で、コンピュータに多くの処理能力や電力を要求する. そこでFPGA(Field Programmable Gate Array)などのハードウェアに実装することで、より効率的にシステムを構築することが期待される.

2. カオスポルツマンマシン

制限付きボルツマンマシンで用いられているボルツマンマシンは確率的に動作する全結合のニューラルネットワークである[1]. あるニューロン i に他のニューロンからの入力 z_i が入ってきたとき、ニューロンが発火する、つまりニューロン i の状態 $s_i \in \{0,1\}$ が1になるかどうかは確率的に以下のように決定される.

$$p[s_i = 1] = \frac{1}{1 + \exp(-z_i/T)} \quad (1)$$

$$z_i = b_i + \sum_j w_{ij}s_j \quad (2)$$

ここで、 T はネットワークの温度と呼ばれる正の定数、 b_i はバイアス、 w_{ij} は他のニューロンとの結合の強さ s_j は他のニューロンの発火状態を表している. よってハードウェアでの実装の際には回路内部に乱数生成器を必要とする.

一方、カオスポルツマンマシン[2]は乱数の代わりにカオスダイナミクスを用いることで乱数生成器を必要としないハードウェア実装を実現することができる. カオスポルツマンマシンではニューロン i は内部状態 $x_i \in [0,1]$ を持つ. この状態値の変化量は以下の式によって表される.

$$\frac{dx_i}{dt} = (1 - 2s_i) \left(1 + \exp \frac{(1 - 2s_i)z_i}{T} \right) \quad (3)$$

また、ニューロンの発火状態は以下の式で表される.

$$\begin{cases} s_i := 0 & \text{when } x_i = 0 \\ s_i := 1 & \text{when } x_i = 1 \end{cases} \quad (4)$$

(3), (4)式により、内部状態 x_i は0と1の間を振動し続けることになる. このシステムによってボルツマンマシンのカオスダイナミクスが実現されている.

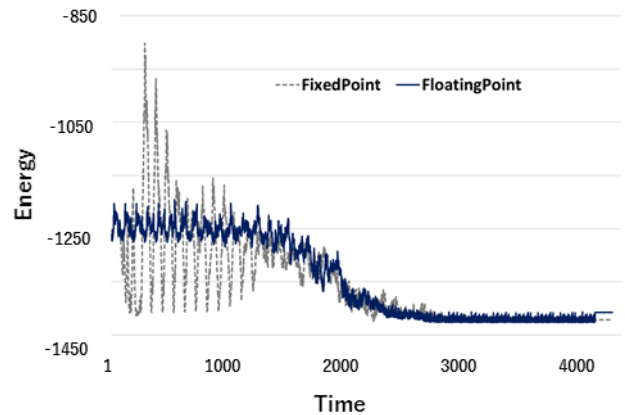


図1. カオスポルツマンマシンのエネルギー変化

3. 提案手法

本稿では、カオスポルツマンマシンの効率的なデジタル回路実装を目指し、固定小数点演算化を検討する. この手法により回路が単純化し、浮動小数点演算より少ないハードウェアリソースでの実装が可能になる.

4. 実験結果

図1にカオスポルツマンマシンを100ニューロンの最大カット問題[2]に適用した結果を示す. 浮動小数点演算(64bit)と固定小数点演算(整数部16bit, 小数部16bit)のボルツマンマシン内部のエネルギー変化を比較したところ、エネルギーの波形は異なるものの、最終的にほぼ等しい値に収束していることから、固定小数点演算が実現可能であることが分かった.

5. まとめ

本稿では固定小数点演算でもカオスポルツマンマシンが実装可能であることを示した.

今後は、ボルツマンマシンとしての特性やカオスの性質などに着目し、効率的なハードウェア実装についての検討を行う予定である.

参考文献

- [1] 岡谷 貴之, “深層学習,” 講談社, 2015.
- [2] Hideyuki Suzuki *et al.*, Chaotic Boltzmann Machines, Scientific Reports, 2013.