4H-8

Cell Broadband Engineを用いた近傍への通知によるPSO演算の並列化

藤田　大地†　　三浦　康之†　渡辺　重佳†

湘南工科大学†‡

1.はじめに

粒子群最適化(ParticleSwarmOptimization, PSO)は医術や映画などと様々な場面で応用されている汎用的な最適化アルゴリズムの一つであり,遺伝的アルゴリズム(GeneticAlgorithm,GA)などと比べて収束が早いことが知られている.PSO は,多大な計算コストが必要となるものの,並列化に適したアルゴリズムであることから,マルチコアプロセッサなどを用いて効率的に並列化を行うことが可能と考えられる.

またこの分野に関して過去に様々な成果が発表されており,CellB.E.に関しては SIMD 演算と SPE の活用により,高い効率で並列化できることが知られている[1].しかしながら局所的な最適値を扱うものに関しては問題が複雑なことから,並列化に関する研究が少ないのが現状である.本研究ではローカルな最適値を Cell 上で演算するうえでのアルゴリズムの検討と検証を行う.具体的には,計算量を要する乱数の発生や評価関数に対して SIMD 演算の導入や SPE の活用による並列化を施し,各コア間とは mailbox により通信する.近傍と通信をする PSO 最適値に関しては多量のデータ通信が必要だが,DMA 通信機能によって解決していく.

2.粒子群最適化

PSO は,集団を構成する個々の情報を共有しながら最適な解を得る、鳥や魚の群れの行動などを単純化しシミュレーションすることによって開発された、最適化問題を解くための有力な手法の１つとして知られている [2].具体的には,群れの探索点において,それぞれが情報を共有しながら最適な解を探索する.各探索点は,位置ベクトル,速度ベクトル,各探索点の過去の最良値とその時の位置ベクトルを保持する.それと共に,全体では過去の最適値とその時の位置ベクトルの情報を共有する.

以下にPSOの処理手順を示す.

１．各粒子について位置ベクトル x と速度ベクトル v をランダムに初期化する.

２．各粒子の最適値を現在位置に初期化する.

３．全体の最適値を各粒子が算出した最適値に置き換える

４．各探索点の位置と速度を以下の数式によって演算する.

$x = x + v$…①

$v = \alpha v + \beta_1\omega_1(x' - x) + \beta_2\omega_2(x'_g - x)$…②

ここでα, β_1, β_2は定数で今回はすべて 0.9 としている.ω_1, ω_2は乱数である.

５．x'とx'_g, vを更新する.

６．３～５の手順を繰り返す.

3.Cell Broadband Engine

PlayStation 3 (PS3®)に搭載されている Cell（Cell Broadband Engine）はソニー・SCE・IBM・東芝によって開発されたマイクロプロセッサである. Cell には PowerPC ベースの CPU である PPE が１基, SIMD 演算機である SPE が８基搭載されている. 2 つはそれぞれ得意とする分野が違い, 有効に活用することで Cell の能力を発揮することができる[3].

4.環境

4.1 LINUX環境

PS3®には PowerPC ベースの CPU が搭載されており, LINUX をインストールすることができる. 我々の研究室では Fedora 11 を導入している.

4.2 Cell プログラミング環境

プログラミング環境として, libspe2 を利用する. Libspe2 は, (株)ソニー・コンピュータエンタテインメント, IBM 社及び(株)東芝の 3 社によって共同で開発され, GNU LGPL に基づき公開されている. Cell の SPE を扱うために必要なヘッダーファイルなどが含まれる. 環境構築は[4]を参照した.

Parallelization of Particle Swarm Optimization with Communication to Neighborhood by using Cell Broadband

†Daichi.Fujita †Yasuyuki Miura †Shigeyosi Watanabe

†Shonan Institute of Technology

Copyright ©2011 Information Processing Society of Japan.
All Rights Reserved.

5.並列化

5.1 大域な最適値による PSO アルゴリズムの並列化

複数の SPU への対応はまず SPU/探索点を各 SPU に割り当てた.

各 SPU からの最適値をまとめるためにはデータをイタレーション 1 回ごとに全体に通信する必要がある.CellB.E.には PPE と各 SPE との通信手段として DMA 通信と mailbox がある.今回扱ったデータ量は 32bit 以下であったため後者により PPE に各 SPU からの最適値を集中させた.同 PPE により最適値を計算したものを再び mailbox により各 SPU に返した.

5.2 近傍と通信する PSO アルゴリズムの並列化

付近の探索点と通信をする PSO のための Cell への実装方法を以下に示す.

5.2.1 DMA 転送

この手法は DMA 転送による多量のデータ通信が必要になる.そこで DMA 転送に要する時間を表 1 に示す.このように 1 回の DMA 転送に要する時間は 1 回のループの 3%以下と非常に短いため,各探索点が共有する情報はすべて DMA 転送で処理することで,ほぼ最高性能を引き出すことが可能になると考えられる.

表 1 DMA 占有率

探索点	①1ループ(us)	②1DMA(us)	②/①
64	8576	259	0.03
128	11274	264	0.023
256	13847	270	0.019

5.2.2 近傍と通信する PSO 並列化アルゴリズム

近傍と通信する際,リンクの接続関係の設定として,閾値の設定,距離の計算に探索点数 N に対して $O(N^2)$の計算量を要する.それぞれを SPE 数で分割し,距離の算出にはユークリッド距離法を使用し,並列化,SIMD 化を施すことにより高速化する.理論上,この処理は SIMD 化により 4 倍,さらに SPE 数分の速度向上が期待できる.

6.実験

今回は PLAYSTATION3(40GB)を使用して実験を行った。実験環境を表 2 に示す.

表 2.実験環境

CPU	Cell B.E.3.2GHz
Memory	XDR 256MB
OS	Fedora 11
コンパイラ	GCC
ライブラリ	CellSDK3.1 libspe2
評価関数	Griewank

表 3. 実験結果

	リンク全体	関数演算	合計
大域1コア(s)	0	3.02	3.02
大域 4 コア SIMD(s)	0	0.38 (7.94 倍)	0.38 (7.94 倍)
近傍1コア(s)	11.46	4.59	16.06
近傍 4 コア SIMD(s)	0.78 (14.69 倍)	0.8 (5.73 倍)	1.58 (10.16 倍)

Cell 上で 5.1,5.2 の並列化を施した PSO の計算を行ったときの性能を調べた.②式の$\alpha, \beta_1 \beta_2$はすべて 0.9,手順の反復は 1000 探索点数は 256 としている.表 3 はそれぞれの手法での 1 コアのみで実行したものと 4 コアと SIMD を施したものの速度を各演算に区切って示したものである.また速度向上率はそれぞれの 1 コアのみのものを基準としている.

リンク演算の速度向上率はほぼ理想値に近い値が出ている.評価関数演算では十分な速度向上が得られていない.これは条件分岐処理で時間がかかっているものと思われる.

7.まとめ

本稿では、CELLB.E.上で PSO を並列に実行するアルゴリズムを実験検証した。Cell の高速化手法を PSO アルゴリズムに適応したことにより,大域な最適値によるものは 7.94 倍,近傍と通信するものは 10.16 倍高速化された.リンクに関しては非常に効率が良く,理想値に近い値が出ている.計算部分に関しては双方ともさらに高速化が見込める結果となった.今後は評価関数部も理想値に近づけていく.

参考文献

[1] 五十嵐潤,園尾聡,古賀崇了, “Cell B.E. による SIMD-oriented Fast Mersenne Twister を用いた Particle Swarm Optimization, ” Cell Speed Challenge 2008 自由課題部門,2008.6

[2] 加藤 暢恵” Particle Swarm Optimization を用いた階層型ニューラルネットワークの一構成” 広島大学大学院教育学研究科紀要. 第二部, 文化教育開発関連領域 Vol.55 page.71-76,2007

[3] http://cell.fixstars.com

[4] PLAYSTATION 3 Linux 完全攻略ガイド 著者：塩田紳二, 安田絹子, 國司光宣, 平初, 川井義治, 古坂大地, 青柳信吾, 八重樫剛史 ISBN コード：978-4-8443-23891

Copyright ©2011 Information Processing Society of Japan.
All Rights Reserved.