

1 はじめに

SAR(Synthetic Aperture Radar:合成開口レーダ)は、日中、夜間、雲霧等の天候を問わずに、高い分解能で地表を撮像できるセンサである。SARでは、センサが収集したデータを信号処理し、人間が理解可能な画像を生成する処理が必要になる。この画像再生処理は、演算量が多く、並列処理等による高速化が求められる。

我々はこの画像処理高速化の手段として、汎用的なUNIXのSMP(Symmetric Multi-processor)マシン上でマルチスレッドプログラムによる処理の並列化を進めている。その中で、画像再生の1部分処理であるコーナーターンの並列化には困難があった。本稿では、キャッシュのヒットを考慮したコーナーターンの高速化方法の検討と、提案する高速化法での計測結果を示す。

2 コーナーターン

SAR 画像再生処理は、レンジ方向圧縮処理と、コーナーターンと、アジマス方向圧縮に分けることができる。レンジ方向はセンサがマイクロ波を照射する方向であり、アジマス方向はセンサを搭載したプラットフォームの進行方向である(図1参照)。レンジ方向とアジマス方向の圧縮処理では、各方向でFFTやIFFTといった信号処理を行う[3]。この圧縮処理では、処理前と処理結果の画像の領域を確保し、レンジまたはアジマス方向で画像を分割して並列実行することで、効率良く高速化を行える見込みを得ている[1]。

コーナーターンは、各方向での信号処理を効率良く行うために、画像のメモリ上の配置を変更する処理で、逐次処理の場合、全処理時間の10%強を占める。図1はレンジからアジマス方向への例である。

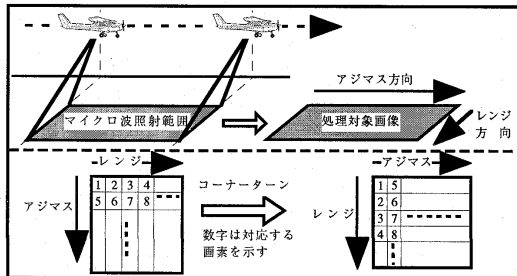


図1 レンジ・アジマス方向とコーナーターン

Parallelization of SAR Image Reconstruction on SMP: Optimization of cache access in Corner-turn  
 Information Technology R&D Center, Mitsubishi Electric Corporation, 5-1-1 Ofuna, Kamakura, Kanagawa, 247-8501, Japan

コーナーターンの並列化を行う時に、我々は次の問題点があると考えた。コーナーターン処理前の画像を基準に単純に並列化を行うと、書き込み側である処理後の画像側でキャッシュのライトミスを引き起こしやすい(図2参照)。逆に、コーナーターン処理後の画像を基準に単純に並列化を行うと、読み込み側である処理前の画像側でキャッシュのリードミスを引き起こしやすい。

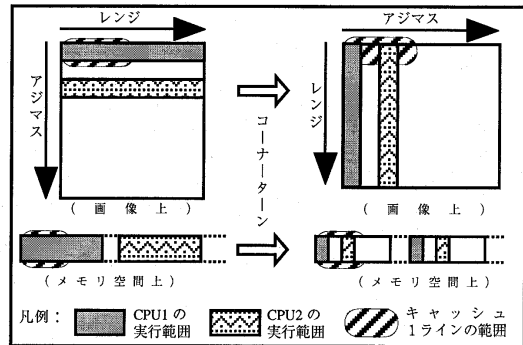


図2 問題となるアクセスパターン

3 並列化方法の検討

コーナーターンでのキャッシュミス問題を解決するために、我々は、キャッシュのラインにちょうど納まる幅の画素数を1辺とする正方形を“画像ブロック”として定義し、この単位で処理をプロセッサに分割してコーナーターンを並列実行する(図3参照)。

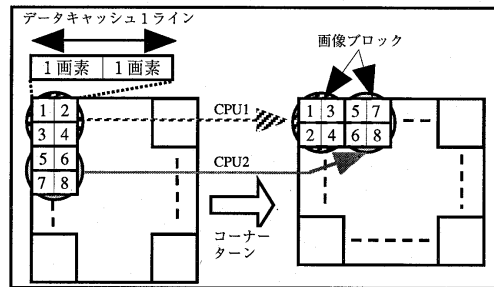


図3 画像ブロック単位でのコーナーターン

階層型のキャッシュの場合、まず、最下位のキャッシュで画像ブロックを算出する。その後、上位のキャッシュで、ラインにちょうど納まる幅の“1つ下位のキャッシュの画像ブロック”数を1辺とする正方形を上位の画像ブロックとして順次算出する。また、画像ブロック内とブロック間のアクセス順序は、キャッシュの総容量(ライン数)から決める。

今回対象としたシステムでは、1画素は8 byte (float × 2) であり、表1に示す値の1次と2次のキャッシュがある。

	Line size	Total Size
1st Cache (D-Cache)	16 byte	16 Kbyte
2nd Cache	64 byte	2 Mbyte

表1. キャッシュサイズ

従って、まず、1次キャッシュのラインに納まる2画素分のデータを1辺とする正方形を「1次画像ブロック」とする。次に、2次キャッシュのラインに納まる4個分の1次画像ブロック(8画素分)を1辺とする正方形を「2次画像ブロック」とする(図4参照)。なお、今回の対象システムでは、両画像ブロックの処理に対して十分なキャッシュ容量があり、画像ブロック内とブロック間でのアクセス順序が実行性能に与える影響は小さい。このため、性能評価対象から除いた。

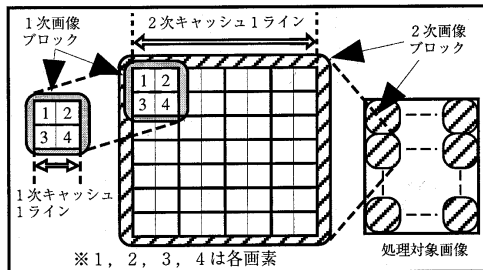


図4 2次画像ブロック

#### 4 性能計測

画像ブロック単位での並列実行法の効果を検証するため、次の3種類の方法でコーナーターンを行った時の性能を計測した。

1. キャッシュ考慮なし  
処理前画像を基準に順次実行。
2. 1次画像ブロック単位  
1次キャッシュのみを考慮した1次画像ブロック単位で実行。
3. 2次画像ブロック単位  
2次キャッシュまで考慮した2次画像ブロック単位で実行。

計測は、Ultra SPARC II(300MHz) × 4CPU である Sun Enterprise 450 上で行った。プログラムの並列化手段としては、我々が並列プログラミング支援環境の1ツールとして研究開発を進めている SPL(Simple Parallel Library) を利用した[2]。結果を図5に示す。

なお、計測では、並列処理(2スレッド以上の処理)のためのオーバーヘッド(スレッド生成やバリア同期時間)を含んでいる。

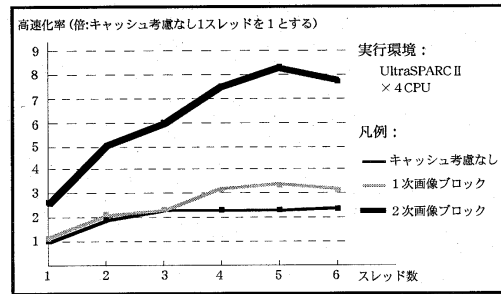


図5 計測結果

この結果、次のことを確認した。

- 2次画像ブロック単位で4CPUを使って並列化した場合に、単純な逐次処理に対して約8倍の高速化を得られること。
- 2次キャッシュを考慮することで、逐次実行でも約2.6倍の高速化を得られること。
- 並列処理の高速化率では、キャッシュ考慮なしでは約2.3倍だが、2次画像ブロック単位では約3.1倍の高速化を得られること。
- 2次キャッシュまで考慮した方が、1次キャッシュのみ考慮したケースよりも性能上の効果が大きいこと。

ここで、1次と2次のキャッシュ間で差が大きくなった原因は、2次キャッシュミスのペナルティが1次キャッシュと比較して大きいためと、1次画像ブロックに含まれる画素数が4個と少ないためと考える。ペナルティの差は、キャッシュのラインサイズ及び総容量の差に加え、1次キャッシュがライトスルー、2次キャッシュがライトバックのポリシーを採用していることも影響していると考えられる。

#### 5 まとめ

SAR 画像再生処理の並列化で問題となるコーナーターン部分処理について、画像ブロック単位での並列実行法を提案した。この方法の効果を確認するため、実行時間を計測した。この結果、4CPUを使って並列化した場合に、単純な逐次処理に対して約8倍の高速化を得た。また、並列処理の高速化率についても2.3倍から3.1倍へと改善できた。

#### 参考文献

- [1] 和泉 秀幸 他:SAR 画像再生処理への並列プログラミング支援環境の適用検討, 第59回 情処全国大会 5L-4, 1999.9.
- [2] 福地 雄史, 石塚 章子, 和泉 秀幸: マルチプロセッサ対応 UNIX 上での並列プログラム開発支援環境の開発, 第48回 情処全国大会 2G-9, 1994.3.
- [3] 飯坂讓二監修 日本写真測量学会編: 合成開口レーダ画像ハンドブック, 朝倉書店, 1998.5.