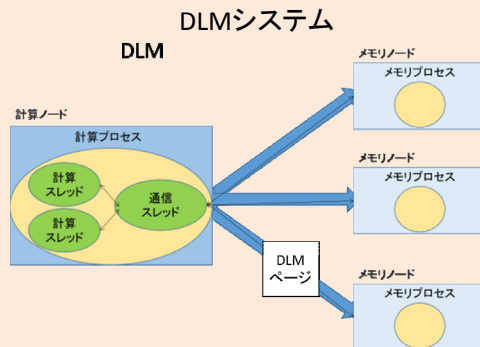


遠隔メモリ利用によるOut-of-Coreマルチスレッドプログラム向け 自動適応型ページサイズ可変機構の動作実験

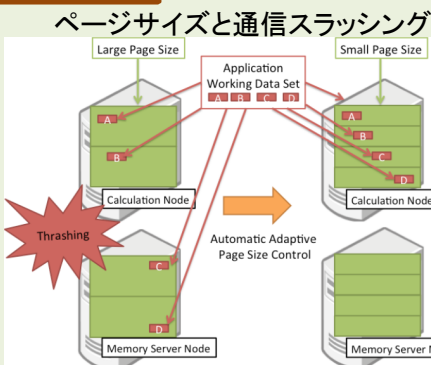
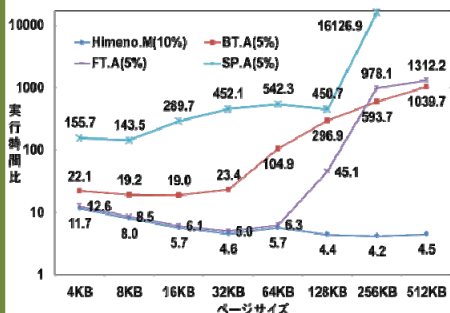
大浦 陽, 緑川 博子(成蹊大)

背景

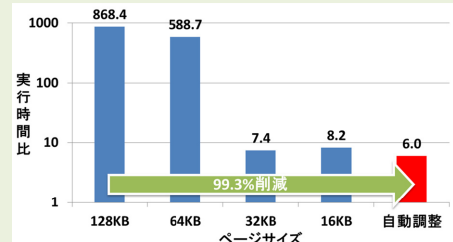
遠隔メモリ利用により、主メモリサイズを超える大きなデータを処理することが可能なDLM(Distributed Large Memory)[1]が開発され、実用上有効であることがわかっている。DLMでは、複数のメモリサーバと計算ノード間のデータ交換ページサイズが指定可能で、アプリケーション実行時の各ループ文のメモリアクセス特徴に応じたページサイズに自動制御することにより、メモリサーバとの通信スラッシングを抑制することが可能である[2]。本発表では、従来、シングルスレッドプログラム向けであった適応型自動ページサイズ制御(AAPC)[3]をマルチスレッドプログラムへ適用できるようにDLMを拡張し[4]、マルチスレッド向けのページサイズ制御方式を試作して稼働実験を行った。



ページサイズが応用の実行時間に及ぼす影響



適応型自動ページサイズ制御機構 AAPCの効果(逐次プログラム適用時)



NPB SP B ローカルメモリ率10%で実行

マルチスレッドプログラム向けページサイズ可変機構の制御方針

1. OpenMPを利用したマルチスレッドアプリケーションを想定
2. ページサイズ制御に用いるスワップ回数、各for文時間計測はマスタースレッドのみで行う
3. アプリケーションの全スレッドに対し、同一のページサイズ変更を適用

ページサイズ変更方式

1. ページサイズを小さくする (スラッシング抑制)
スワップ回数(WS推定値) > 計算ノードのページ数
→ 次のページサイズ: 計算ノードのメモリサイズ / スワップ回数
2. ページサイズを大きくする (通信効率の向上)
計算ノードのページ数 > スワップ回数 → ページサイズ * 2
3. ページサイズ変更後の性能がよい場合 **元のページサイズへ戻す**
変更後の実行時間 > 変更前の実行時間 * 1.1 → 元のページサイズへ
4. **選択的適応** (ページ可変機構オーバーヘッド削減)
実行時間の長いループ文にのみページサイズ変更を行う

自動調整関数

```

for(int i=0; i<n; i++){
    swapin_countstart(0);
    #pragma omp parallel for
    for(int j=0; j<l; j++){
        statements1;
    }
    swapin_refresh(0);

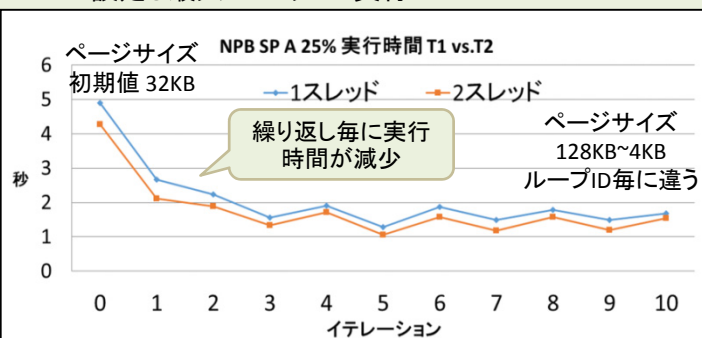
    swapin_countstart(1);
    #pragma omp parallel for
    for(int k=0; k<m; k++){
        statements2;
    }
    swapin_refresh(1);
}

```

- 繰り返し処理のページスワップ回数と実行時間を計測
- 計測箇所毎にループIDを設定

マルチスレッドプログラム向けページ制御動作実験

NPB2.3[5]のSPのAを、ローカルメモリをプログラムサイズの25%に設定し最大2スレッドで実行



マルチスレッドプログラム向けページ可変制御方針が有効であることを確認。今後、多スレッドによる安定稼働、複数応用における性能評価を行う予定

[1] 鈴木, 鷹見, 緑川: "マルチスレッドプログラムのための遠隔メモリ利用による仮想大容量メモリシステムの設計と初期評価", 情報処理学会, HPC研究会 Vol.2011-HPC-132, No.13, pp.1-6, (2011,11)
 [2] 古尾谷, 緑川: "リモートページングのためのページサイズ自動調整機構 - ループ文におけるワーキングセット推定と選択的制御の導入 -", HPCS2013論文集, (2013,1)
 [3] H. Midorikawa, J. Uchiyama: "Automatic Adaptive Page-size Control for Remote Memory Paging", proc. of 13th IEEE/ACM International Symp. on Cluster, Cloud and the Grid Computing CCGrid2012, pp.694-696, 2012-5
 [4] 上野, 長畑: "マルチスレッドプログラム対応遠隔メモリページングへのページサイズ可変機構の設計と実装", 成蹊大卒業論文(2015,2)
 [5] NAS Parallel Benchmark Omni
<http://www.hpcs.cs.tsukuba.ac.jp/omni-compiler/>