3F－6

命令トレースとロック取得状況測定モニタによるSMP性能評価

関口 知紀* 庄司 直史* 円光 豊* 高崎 繁夫† 新井 利明*
*日立製作所システム開発研究所　†同ソフトウェア開発本部

1 はじめに

主記憶共有型マルチプロセッサ(SMP)のソフトウェアオーバーヘッド増加要因に，ロック競合によるステップ数の増加がある．これに関しては従来より解析的手法，シミュレーション予測[1]，稼働時のロック統計測定[2][3]を用いた性能評価が報告されている．しかし，これらの解析的手法では前提としてロック獲得頻度，保持時間を指数分布としており，この結果と実測値を比較した報告はない．

本研究では，命令トレースによる静的ロック解析と，稼働時のロック統計測定を組合せた解析により，静的解析の限界の見極めと高多重時の性能見積もりを実施した．

2 命令トレースによる静的解析

2.1 マルチプロセッサ対応トレーサの開発

ロック取得情報を得るため，プロセッサの実行命令列を取得するトレーサ[4]をマルチプロセッサ(MP)向けに拡張した．

(1) 疑似マルチプロセッサモードトレース

通常のOSでは，シングルプロセッサとマルチプロセッサで動作が異なる．ロック解析のためにはトレースもMPモードで取得しなければならない．しかし，全プロセッサのトレースを取得するのは工数，および，データ量が多くなり実際的でない．そこで，MP構成時にはトレースを取得するプロセッサをのみをMPモードで動作させ，他をアイドルさせる簡略版MPトレーサを開発した．

(2) I/O割り込み遅延機能

一般に，トレース下では命令実行が遅くなりタイミングに依存する処理が実稼働時異なる．特に，OSでは割込み契機とした処理が多いため，見かけ上のI/O完了時間が変化すると，プロセススケジュールの命令列が実稼働時と異なってしまう．そこで，トレーサで外部割込をある期間保留にし，その後割込をエミュレートし，実稼働時のタイミングでトレースを採取可能とする方式を開発した．これにより，ロック取得の契機と保持時間をより実際に近い形で収集できる．

Performance measurement of SMP by step trace and lock measurement.
T. Sekiguchi, N. Shoji, Y. Enkou, S. Takasaki, T. Arai.
Systems Development Lab., Hitachi Ltd., Software Development Center, Hitachi Ltd.

2.2 ロック取得情報の抽出

まず，静的解析として前記トレーサで取得したトレースからロック取得情報を抽出した．ロックワードへの操作は特定モジュール内のメモリ操作で実現されている．そこで，分岐命令情報でロック操作モジュールへの分岐を検出，メモリ参照命令で対象ロックを特定し(図1)，以下を取得するツールを開発した．

(1) 操作対象ロック．
(2) 各ロック獲得について，保持ステップ数．
(3) ロック獲得/解放要求を発行したモジュール名．

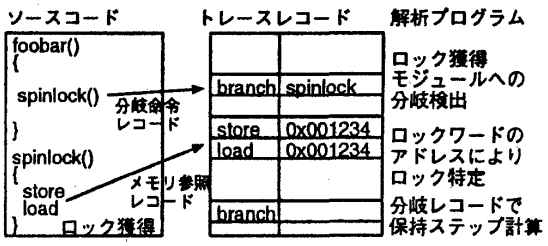

図1 トレースによるロック情報抽出

ロック保持ステップは，分岐命令情報に含まれる直前の分岐命令からの実行命令数を利用して計算した．

2.3 測定結果

あるトランザクション処理の命令トレースよりロックの取得状況を抽出した．解析結果の一部を表1に示す．

表1 トレース解析の結果

全処理ステップ数	165K
処理中に獲得したロック数	26
ロック獲得回数	229

ロック	獲得回数	ロック保持ステップ比	競合伸び予測(ステップ) 4way	16way	32way
lock1	7	0.7%	12.2	83.5	182.2
lock2	2	2.2%	137.6	968.0	2194.4
lock3	26	0.4%	3.9	26.4	56.7

解析の結果，保持ステップの長いロックはスケジュール関連，割り込み処理関連のロックであるが，処理に占める保持ステップの割合は数％未満であり，解析的手法による予測によれば，これらのロックでは競合は1%未満であるという結果となった．

3 モニタによるロック統計測定

3.1 ロックモニタの開発

静的解析の結果を検証するためロックモニタを開発した(図2)．本研究ではロック保持/待ち時間の平均値だけでなく，それらの分布がロック解析および高多重時の予測に有効であると考え，簡単な分布情報も取得することとした．また，測定オーバーヘッド削減のためキャッシュラインを考慮してデータを配置し，測定オーバーヘッドは1%未満となった．各ロックについて以下のデータを取得する．

(1) 獲得/競合回数．
(2) 保持/獲得待ち時間の合計．
(3) 保持/獲得待ち時間の簡略分布．

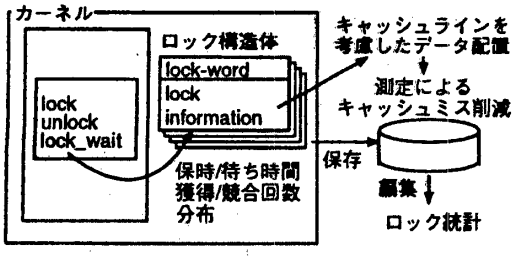

図2 ロックモニタの構成

時間情報はプロセッサへのクロックを単位として測定している．また，保時/待ち時間を適当な区間に区切って，それぞれの区間に含まれる保持/待ちの回数を，簡略分布として記録している．

3.2 ロックモニタによるロック統計測定

トレース解析で用いたものと同一のトランザクション処理のロック統計を測定した．測定ではプロセッサ数を1～4台と変え，1秒毎に統計を記録した．プロセッサ4台の時の測定結果をトランザクション当たりで正規化した結果の一部を表2に示す．

表2 モニタによる測定の結果(4プロセッサ時)

ロック	獲得/tr (回)	競合/tr (回)	保持/tr (step)	競合伸び/tr (step)
lock1	7.95	1.49	3884	1252
lock2	1.76	0.13	3460	222
lock3	24.49	0.50	212	44

4 測定結果の比較と考察

ロック競合について，静的解析の4wayの競合予測値とモニタによる実測値を比較すると大きく異なる値になっているロックが多い．その原因としては以下が考えられる．

(1) ロック保持と獲得頻度間の依存関係

プロセッサ間で通信が必要な時に，ロック獲得頻度がランダムと仮定できない場合がある．ここで対象としたOSでは全プロセッサでスケジューラが動作し，割込みにより互いのスケジュール事象を通知する．このため，I/O完了後等に複数プロセッサで同時にスケジュール操作が起動される傾向があり，スケジュール関連のロック獲得間隔をランダムと仮定できない．

(2) 複数ロックの同時保持

一般に，OSでは複数のロックを同時に保持する場合が多くある．このため，単純に待ち行列モデルを適用すると，下位のロックの競合を過大に見積もってしまう傾向がある．これに対しては，トレース解析を用いればロック獲得パターンを知ることができ，どのロックの予測が過大評価になりうるかの目安を得られる．

(3) 高度なロック機構

一般に，OSは単純なビジーウェイト型のロック，サスペンド型のロック，これらを組み合わせたロックを使い分けている．サスペンド型ロックはプロセススケジュールと密接に関連しており，スケジュール関連のロック獲得パターンに影響を与える．実際，本稿で示したlock1はこの影響を受けて獲得待ちが増加している．

5 おわりに

命令トレースを用いたロック解析と実稼働時ロック統計測定を実施した．これらの比較より，OS内のロックに対する待ち行列モデルの適用の限界とその原因を示した．

参考文献

[1] 星合 隆成:“密結合マルチプロセッサシステムにおける共通データアクセス競合の近似解析”, 信学論(D-I), J77-D-I, 1, pp.53–65(1994-01)
[2] M. D. Campbell 他: “Lock-Granularity analysis tools in SVR4/MP”, IEEE Software, Mar. 1993, pp.66–70
[3] J. P. CaraDonna 他: “Measuring Lock Performance in Multiprocesor Operating System Kernels”,
[4] 高崎 繁夫他:PA-RISC用ステップ解析システムの開発, 情報処理学会第47回全国大会, 4G-7(1993)