時刻印方式における部分後退復帰

2H-1

仲 興国　上林 弥彦
(九州大学 工学部)

1. まえがき

分散データベースシステムにおける代表的な並行処理制御として、時刻印方式が知られている[1]。従来の時刻印方式においては、操作衝突が生じたとき、必ず時刻印の小さい処理単位が後退復帰されるようになっている。そのため、我々は後退復帰対象を動的に決定する時刻印方式を提案した[4]。[4]で提案した方式の一つの特徴として、処理単位の部分後退復帰[2]が可能となった。本稿では、分散データベースにおいて処理単位の部分後退復帰が必要であることを示し、[4]に基づく部分後退復帰の実現方法について検討する。

2. 時刻印方式

本節では、従来の時刻印方式および[4]で提案した時刻印方式について簡単に述べる。

(1) 従来の多重版時刻印方式　各処理単位T_iは到着順に時刻印t_iを割当てられる。データxの読み出し時刻印$t_r(x)$はxを読んだ処理単位の時刻印のうち最大のもの、xの書き込み時刻印$t_w(x)$はxを書いた処理単位の時刻印とする。書き込み操作はデータxの新しい版を生成する。xの過去の値は多重版として保持する。T_iの読み出しは$t_i > t_w(x)$を満たす有効な値を読む。T_iの書き込みは時刻印t_iが最大の$t_r(x)$および$t_w(x)$と比較して$t_i > \max(t_r(x), t_w(x))$を満たすときのみ行える。条件を満たさないときには$T_i$を後退復帰する。上記により、矛盾を生じたときに時刻印の小さい(従って先に生成された)処理単位が後退復帰される。

(2) [4]の時刻印方式　従来の時刻印方式を基礎とし、矛盾を生じたときにできる限り時刻印の大きい処理単位を後退復帰する。この方式は各データxの操作履歴を完全に記述することを必要とする。従来の多重版方式と同様にT_iがxを読み出すには矛盾を生じない。書き込みに矛盾を生じた場合に、できればT_iではなく、T_iの書き込み値を読むべきであった処理単位を後退復帰する。図1にデータxに対する操作履歴を記述している。各w_iやr_iのiは処理単位T_iによる操作に対応しており、iが大きいほど対応する時刻印も大きいと仮定する。処理単位T_1の$w_1(x)$で生成した値がある。それを読み出す操作に$r_2(x)$、$r_4(x)$がある。T_3の書き込み$w_3(x)$が来たときに、$r_4(x)$を行ったT_4が後退復帰される。従来の方法ではT_3が後退復帰させられた。

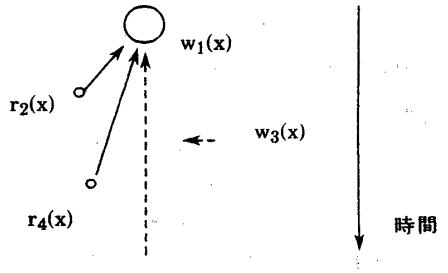

図1. 操作履歴の変更

3. 部分後退復帰の必要性

操作の衝突が起こったときに、処理単位全体ではなく、矛盾する操作を含む処理単位の一部分をやり直して済ませることは部分後退復帰と呼ばれている[2]。部分後退復帰の実現は難しいため、従来あまり重視されていない。しかし、分散データベースの場合に処理単位の部分後退復帰は特に重要である。何故ならば、全域的な処理単位は各局において、それぞれ一つの部分処理単位として扱われる。それらの部分処理単位は通信によって結合されている。処理単位の実行コストには(従って後退復帰のコストにも)データ通信が問題となっている。従って、一ヵ所での操作衝突はその局で局所的に扱えることが望ましい。即ち、部分後退復帰が実現できると、後退復帰しなければならない処理単位の実行したコストを減少するとともに後退復帰のための通信は不要にする。

従来の時刻印方式では、矛盾が起こったときに時刻印の小さい処理単位が後退復帰されるため、部分後退復帰の実現は原理的に不可能となっている。なぜならば、時刻印の大きい処理単位を実行してしまうため、時刻印の小さい処理単位に大きい時刻印を割り付けなければ、矛盾が解除できないからである。[4]の方式により処理単位の部分後退復帰は可能となった。次の節では、部分後退復帰について述べる。

4. 処理単位の部分後退復帰

分散データベースにおける全域的処理単位T_iは所在局におけるTM(処理単位管理者)で制御され、各局で実行する部分処理単位の列($T_i = T_{i,1}, T_{i,2}, \ldots T_{i,n_i}$)と見なされる。$T_i$の実行にあたって、各部分処理単位の初期状態を$T_i$を認証するまで保持しておく。矛盾を生じ

Partial Rollback in a New Timestamp Ordering Method
Xingguo Zhong and Yahiko Kambayashi
Kyushu University

たとき時刻印の大きい処理単位が既に認証された場合に、時刻印の小さい処理単位を後退復帰し、全体的後退復帰を行う。それに対して時刻印の大きい処理単位が認証されていない場合にそれを後退復帰し、部分後退復帰を行う。

(1) 後退復帰対象の決定　　分散データベースでは、処理単位の認証には二相認証[3]を用いるのが普通である。処理単位Tを認証するときに、それが生成される局S_Tから各部分処理単位の局へ、予備認証を通知する。各局で部分処理単位の予備認証を行う。予備認証が完了すると、局S_Tに返事する。すべての部分処理単位を予備認証したことをS_Tが受け取れば、認証の第二段階を行う。

後退復帰対象の決定は衝突が起きたときに時刻印の大きい処理単位が衝突する局において既に予備認証されたかどうかによる。即ち、衝突する操作の属する部分処理単位が予備認証されているかどうかのことである。予備認証されていないときに、それを後退復帰する。

(2) 部分後退復帰の制御　　$T_i(T_i=T_{i,1}, T_{i,2}, \ldots T_{i,n_i})$を後退復帰される時刻印の大きい処理単位とし、$T_{i,k}(1\leq k\leq n_i)$を矛盾を生じたときの部分処理単位とすると、$T_i$中に$T_{i,k}$以後の部分だけを後退復帰すれば良い。衝突が起きた時点で、$T_{i,k}$はまだ終了していないかもしれない。そのときに、$T_{i,k}$のみ後退復帰する。$T_{i,k}$が同一の局にあるので、後退復帰の処理には通信が不要である。$T_{i,k}$が既に終了し、T_iが他の部分処理単位を実行しているとき、後退復帰に通信が必要である。

(3) 操作履歴の変更　　データxの操作履歴は一般にxの書き込み時刻印の順に付けた複数の版とそれらを参照する読み出しからなる[4]。処理単位の部分後退復帰の場合に、操作のほとんどは同一の時刻印で再び実行するため、書き込みをデータの操作履歴から消去しなくてよい。読み出し操作は削除すべきである。

5. 部分後退復帰の問題点とその解決法

処理単位$T_i(T_i=T_{i,1}, T_{i,2}, \ldots T_{i,n_i})$において、部分処理単位$T_{i,k}(1\leq k<n_i)$上で操作の衝突が起きたときに、$T_i$の部分後退復帰$PR(T_i, k)$が行われる(図2を参照)。そのとき$T_i$は$T_{i,l}(k<l\leq n_i)$以後まで実行しているとする。その部分後退復帰を処理する間に、部分処理単位$T_{i,l}$上で更に操作の衝突が起き、部分後退復帰$PR(T_i, l)$が起きることがある。$PR(T_i, k)$によって$T_{i,l}$に対する後退復帰が先に行われたなら、$T_{i,l}$上で行われる読み出しを既に対応する操作履歴から削除しているため、$PR(T_i,l)$を起こらないが、$PR(T_i,l)$が先に生じたときには、$T_{i,l}$以後の部分処理単位に対する後退復帰は二重となる。同様に部分処理単位の後退復帰が多重に起こる可能性がある。

更に、$PR(T_i, l)$が$PR(T_i, k)$より先に起きることもありうる。部分後退復帰$PR(T_i, l)$が完了してからT_iが$T_{i,l}$から実行し続ける。その後$PR(T_i, k)$が起きるならば、$PR(T_i, k)$が単一な後退復帰となる。$PR(T_i, l)$の処理中に$PR(T_i, k)$が起きたときには$T_{i,l}$以後の部分処理単位の後退復帰が二重となる。

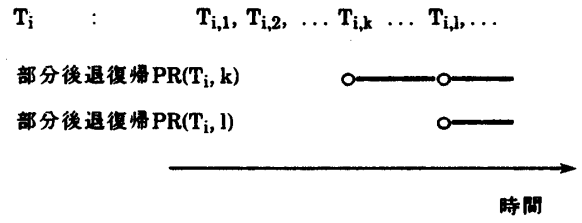

図2. 多重後退復帰

同じ部分処理単位に対する後退復帰が多重に起きると、正しく処理できない恐れおよび効率の低下がある。それに対する対策として次の方法を用いる。

衝突が起きてT_iを部分後退復帰するときには、衝突した局からT_iが生成された局S_{T_i}に通知する。S_{T_i}におけるTMがT_iに対する部分後退復帰を制御する。

TMが$PR(T_i, k)$(或いは$PR(T_i, l)$)を実行するときに$T_{i,k}$(或いは$T_{i,l}$)以後の各部分処理単位を並列に後退復帰する。$PR(T_i, k)$(或いは$PR(T_i, l)$)を処理する間に$PR(T_i, l)$(或いは$PR(T_i, k)$)が起きたときにTMがこの二つの部分後退復帰をまとめて処理を行う。まとめた結果、各部分処理単位に対して$PR(T_i, k)$のみが起きたときと同様である。処理単位の再実行は部分処理単位$T_{i,k}$からである

6. むすび

分散データベースの場合に部分後退復帰が重要であることを示し、[4]で提案した時刻印方式に基づく部分後退復帰の実現方法について述べた。代表的な並行処理制御である二相施錠方式における部分後退復帰[2]はすくみを解消することを目標とする。それに対して時刻印方式における部分後退復帰は時刻印の大きい処理単位の実行が早すぎた部分をやり直すことといえよう。

参考文献:

[1] Bernstein, P. A. and Goodman, N, : Timestamp-based Algorithms for Concurrency Control in Distributed Database Systems. Proc. Inter. Conference on VLDB. Oct. 1980.

[2] Fussell, D., Kedem, Z. M. and Silberschatz,A., : Deadlock Removal Using Partial Rollback in Database Systems, ACM Proc. SIGMOD, 1981.

[3] Gray, J. N. : Notes on Data Base Operating Systems, IBM Report, RJ2188,1978.

[4] 仲, 上林, : 後退復帰対象を動的に決定する時刻印方式, 情報処理学会, 第54回 データベースシステム研究会, 1986年7月.