

一貫性情報に基づく処理単位の動的再構成

5 A A - 1

徐海燕 古川哲也 史一華

福岡工業大学 九州大学 福岡工業短期大学

1 まえがき

データベースの応用分野がCAD/CAM, オフィスオートメーション, CASEなどに拡大するにつれ, 協同作業支援機構をデータベースの機能として実現する必要性が高まってきている。CAD/CAMなどの応用分野とそれにおける協同作業には, 作業の長時間性や分担, 進行状況の予測不可能性などの特徴がある。

このため, 協同作業機構は高度な並行処理を支援できる制御機能と, 利用者に処理単位の動的再構成機能を提供する必要性がある。前者の要求のために, 著者らは状態を導入することによって, 作業手順や制約によって定義される状態の変化規則がデータベースの一貫性のための統一的な判定基準となるという結果を示している^[2]。さらに, 節で表される一貫性情報を利用することにより, 並行処理制御の正当性基準は従来の直列可能性から独立化可能性へ拡張できるという結果が得られている^[2, 3]。本稿では, 独立化可能性基準に基づく並行処理制御における処理単位の動的再構成機能について検討する。

2 一貫性情報に基づく正当性基準

処理単位 T は, 半順序関係 $<$ を持つ検索操作 $(R(x))$, 変更操作 $(W(x))$ の集合である。与えられた処理単位 T に対して, $WSET(T) = \{x \mid W(x) \in T\}$ と $RSET(T) = \{x \mid R(x) \in T\}$ でそれぞれ T によって変更, 検索されたデータ項目の集合を表す。複数の処理単位の並行実行をスケジュール H といひ, $<_H$ でその実行順序を表す。

独立化可能なスケジュールのクラスは, 各処理単位 T が次の3つの性質を満たす等価なスケジュールが存在するものからなる^[2]。

- 論理性: T によって操作されたデータ項目は, T が終了するまで, 他の処理単位によって変更されていない。
- 検索一貫性: T によって検索されるのは, ある一貫性制約を満たしたデータベースの部分集合である。
- 結果一貫性: 検索されたデータベースに対して, 単独実行した結果のデータベースは一貫性制約を満たす。

独立化可能性は, 各処理単位は一貫したデータベースに対して実行され変更した結果は一貫しているという

処理単位の正当性も, 一貫したデータベースにおける独立化可能スケジュールの実行結果は一貫したデータベースであるというデータベースの一貫性も保証できる^[2]。

スケジュール H と等価で各処理単位は論理性を満たすようなスケジュールからなる論理性のためのクラスについて, 直列可能スケジュールのクラスと比較した上で検討されている^[3]。2相施錠方式から専有施錠とその後施錠される共有施錠間の競合関係を無くす拡張によって得られた2相施錠の拡張版(2PLE)によって許可されたスケジュールは, 論理性のためのクラスに属する^[3]。

3 分割・合併操作とその性質

従来のデータベースにおいて, 処理単位の構成に対する操作は, 処理単位の開始, 終了, 中止 (abort) の3種類のみである。しかし, CAD/CAMなどの応用分野においては, 処理単位の進行状況は予測不可能である。終了できない場合に, 中止という対処方法しか取れなければ, 長時間に渡る作業結果の紛失という受け入れ難い事態を招く。直列可能性基準の下での処理単位の動的再構成機能も検討されている^[1]が, データベースの一貫性と処理単位の正当性にどのような影響を及ぼすかについては, 利用者の責任としている。本節では, 2PLE方式を用いる場合に動的再構成操作の基礎となる分割・合併操作の性質について検討する。

・処理単位の分割: 処理単位 T を次の2つの条件を満たす部分処理単位 T_i, T_j に分割する。

$$i) WSET(T_i) \cap WSET(T_j) = \phi,$$

$$ii) R(x) < W(x) \text{ である } R(x), W(x) \text{ は同じ部分処理単位に属す。}$$

条件 (2) より, 分割後のスケジュールも 2PLE スケジュールである。また, 検索一貫性の定義により, 分割によって検索一貫性上新しい問題が生じることはない。一方, 結果一貫性は, 一貫性制約の中で $WSET(T)$ の要素に関わる節が, 処理単位の結果値 $(WSET(T) \cup (WSET(T) - RSET(T)))$ において真であることを要求するので, 分割後データ不足によって結果一貫性の判定ができない場合がある。しかし, 分割された T_i の結果一貫性判定に $R(x)$ の結果が不足しているならば, $R(x)$ は T_i にも属するように分割すればよい。一方, $W(x)$ の結果が不足しているならば, $W_j(x) < R_i(x)$ である $R_i(x)$ を T_j に追加すればよい。

以上のように, 分割操作で検索一貫性の問題に対処できる。また, 分割された1つの処理単位を終了することによって一部のデータ項目は解錠され, 競合で専有施錠

Restructuring Transactions Dynamically Based on Consistency Information

Haiyan XU[†], Tetsuya FURUKAWA[‡], Yihua SHI^{††}

[†]Fukuoka Institute of Technology, [‡]Kyushu University,

^{††}Fukuoka Junior College of Technology

できないという論理性上の問題にも対処できる。さらに、結果一貫性を満たさない処理単位の問題となる一部の変更操作を1つの処理単位に分割して中止すれば処理単位全体が後退復帰することを避けられる。一方、直列可能性基準の場合では、条件(2)では分割後のスケジュールが2PLスケジュールであるとは保証できず、分割された T_i, T_j の結果一貫性の明確な保証方法もない。

・施錠相にある2つの処理単位の合併：次の条件を満たす T_i, T_j を1つの処理単位 T として合併する。

条件： $R_i(x) <_H W_i(x) <_H R_j(x) (R_j(x) <_H W_j(x) <_H R_i(x))$ である $R_i(x), R_j(x)$ は存在しない。

この条件を満たせば合併された処理単位 T は、各データ項目に対する検索、変更操作は1回のみという条件を満たす。これは2PLEスケジュールにおいて、共有施錠は常にでき、一旦施錠されたデータ項目には専有施錠できないためである。これより、合併後のスケジュールは2PLEスケジュールであることも保証される。また、結果一貫性の定義より、元の2つの処理単位が結果一貫性を満たすなら、合併後の処理単位も結果一貫性を満たす。ただし、合併後検索一貫性を満たさなくなることがある。

単なる合併操作を必要とする場合は、いままでの処理を協同作業者にまとめてやってもらうような場合に限られるが、分割操作と組み合わせると、分割された一部の処理単位の合併、合併後再び分割などのような利用例が上げられる。

4 競合に対処するための操作

本節では、合併、分割、終了などの操作の組合せで2PLE方式を用いた場合、競合により専有施錠ができないという具体的な場合の対策について検討する。

・ $W(x)$ に対する部分保存操作：

次の3つの操作からなる。

i) 処理単位 T を $R(x), W(x)$ および x の結果一貫性判定に必要な結果に対する検索操作からなる部分処理単位 $T_i (W(x)$ の結果一貫性判定に $R(y)$ の結果が必要ならば、 $R(y)$ は T_i にも属するように分割し、 $W(y)$ の結果が必要ならば、 $W(y) <_H R_i(y)$ である $R_i(y)$ を T_i に追加する)と、 $T_j = T - \{W(x)\}$ である T_j に分割する。

ii) 処理単位 $T_k = \{R_k(x)\}$ を実行させるとともに、 T_i を終了させる。

iii) 処理単位 T_k, T_j を再び合併する。

T_i が結果一貫性を満たさないなら、この操作は拒否されるが、この場合は元の処理単位 T の $W(x)$ に関する変更結果も結果一貫性を満たさないことを意味する。2PLE方式を用いる場合には、処理単位 T_k は常に共有施錠できる。さらに、 T が結果一貫性を満たすなら、(3)の合併時にも検索一貫性上問題は生じない。これは $R_k(x)$ の値が $W(x) \in T$ の値であるためである。

変更結果の保存を考えなくても良い場合には、次の2つの操作がある、

・ $W(x) \notin T$ である $R(x)$ を削除する操作：

処理単位 T を $\{R(x)\}$ と $T - \{R(x)\}$ に分割し、前者を中止する操作系列に相当する。

・ $W(x)$ を削除する操作：

処理単位 T を $\{W(x)\}$ と $T - \{W(x)\}$ に分割し、前者を中止する操作系列に相当する。これには、 $W(x)$ がすでに実行された場合と $W(x)$ のための専有施錠が待機中の2つの場合がある。

これで x がすでに専有施錠されたことによる競合は、 $W(x)$ に対する部分保存操作と $R(x)$ を削除する操作を順に行えば解決できる。一方、 x が共有施錠されたことによる競合は、 $R(x)$ を削除する操作で x は解錠される。ただし、結果一貫性判定に $R(x)$ の結果を必要とするなら、その前に部分保存操作を行えばよい。

競合処理の対策としては、専有施錠の処理単位はその時点で共有施錠している処理単位の終了までは待つが、待機している間に新たな共有施錠は許可しないという方法もある。

・ x に対する専有施錠予約操作：

x は共有施錠されており、予約している他の処理単位はないことが条件となる。

この予約操作を実行すると、他の処理単位は x に新たに共有施錠できなくなり、現在共有施錠中の処理単位が解錠すれば、予約した処理単位はすぐ専有施錠できるようになる。

このように、競合やすくみの場合には、動的再構成操作を利用すれば、後退復帰/部分後退復帰されることなくスケジュールの独立化可能性を保証できる。

5 むすび

本稿では、独立化可能性基準における処理単位動的再構成機能の性質をまとめた。直列可能性基準^[1]の場合と比べると、動的再構成による影響を完全に把握できるという特徴がある。協同作業支援機構をデータベースの機能として実現するためにはこの性質は不可欠である。

参考文献

- [1] Kaiser, G. E.: Dynamic Restructuring of Transactions, *Database Trans. Models for Advanced Applications*, Morgan Kaufmann Pub. (1992).
- [2] 徐 海燕, 古川哲也, 史 一華: 一貫性情報を用いたデータベースの並行処理制御, *情処論*, Vol. 35, No. 12 (1994).
- [3] 徐 海燕, 古川哲也, 史 一華: 並行処理制御方式による独立化可能クラスと直列可能クラスの比較, *情処論*, Vol. 37, No. 8 (1996).