数理解析研究所講究録
556巻 1985年 59-70

データベースシステムの同時処理制御における直列可能性のいくつかのクラスについて

京大工学部 木庭 淳 (Jun Kiniwa)
京大工学部 室章治郎 (Shojiro Muro)
京大工学部 長谷川利治(Toshiharu Hasegawa)

1. まえがき

データベースシステムにおいて複数利用者が参照命令(read access)と更新命令(write access)から成るトランザクション(transaction)を各々発生させるような環境下で，どのように制御すればデータが論理的に正しい処理となるか，すなわちデータの首尾一貫性(consistency)が保たれるか，という問題は，同時処理制御の中心的な課題である．いまトランザクションを一つの単位とみて一つずつ実行させる直列(serial)な系列(sequence)を考えると、これは明らかに正しい処理となるが非常に効率が悪い。そこで処理のスピードアップを計るため，複数利用者の要求を同時に受付けることが必要となり，複数個のトランザクションのアクセスのかみ合わさった(interleaved)系列を実行することになる．ところがこのような系列を実行するとデータの首尾一貫性が失われるおそれがあるため，直

列可能性(serializability)の概念が導入された．これはあたかも直列な系列を実行したと同等な効果を残すような系列を正しいものと認めよう、という考え方である．

現在までの研究で同時処理制御の分野で多くのことが明らかにされてきた．すなわち、直列可能性を実現するための施錠方式(locking)，同時性を高める多版方式(multiversion)、あるいは直列可能かどうかの判定が多項式時間で可能となる部分クラスの研究等がそれである．

ところで最近、Brzozowski，室は直列可能性を簡潔な数学的表現で表わし、従来の研究で扱ってきた直列可能性の概念は、必ずしも一致したものではないことを明らかにし、それらを区別して五つの直列可能性のクラスを定義した[2][3]．すなわち二つの実行系列 e_1 と e_2 の間で、データベースの最終状態が等しくなる場合を δ-等価 $(e_1 \sim_\delta e_2)$ とし、各トランザクションの得る情報が等しくなる場合を τ-等価 $(e_1 \sim_\tau e_2)$ とする．e_2 が直列であるとき、e_1 をそれぞれ δ-直列可能、τ-直列可能という．また $e_1 \sim_\delta e_2$ かつ $e_1 \sim_\tau e_2$ で e_2 が直列なら、e_1 を r-直列可能という．五つのうち重要なものはこの三つと思われる．

本稿ではある系列が r-直列可能なクラスに属するかどうかの判定が、TIO(transaction input/output)グラフ上の DITS (disjoint-interval topological sort)の概念によりなされること[5]をもとに、

τ-直列可能性，δ-直列可能性を判定するためのτ-TIOグラフ、δ-TIOグラフを提案した。また，各々の直列可能性のもつ意義を検討した上で、これらのグラフを用いてτ-直列可能，δ-直列可能な系列に対し適当な処理を施せば、r-直列可能と同じ効果を残す系列として受理できること、さらに多版方式に関し版数をふやすことにより各直列可能性のクラス間の包含関係はどうなるか、について調べた。なお定理の証明など詳しい内容は、文献[7]を参照されたい。

2．諸定義

データベースシステムはデータ項目の集合$D=\{d_1,\cdots,d_{|D|}\}$，トランザクションの集合$A_J=\{T_1,\cdots,T_n\}$及びスケジューラから成っている。一つのトランザクションT_iはデータ項目$d\in D$に対するread命令$r_i(d)$，write命令$w_i(d)$及びそれらの間に定められた順序関係$\leq_i$から成る。$\leq_i$には二つの意味があり、一つは任意のT_iの命令a,bにつき，$a\leq_i b$ならaはbの前に実行されなければならないこと、他の一つは$r_i(d)\leq_i w_i(d')$，$d,d'\in D$の場合には$w_i(d')$の値はT_iがその前に読んだデータ項目dの値に依存することである。

複数個のトランザクションがスケジューラに到着したとき、その命令の到着順に並べたものを系列という。各系列は、各命令が$\leq_i$と矛盾しないよう無作為に、左から右への全順序に

並べたものとして与えられる。例えば次の系列

$e_0 = r_1(a)r_2(a)w_1(a)w_2(a)$, $r_1(a) \leq_1 w_1(a)$, $r_2(a) \leq_2 w_2(a)$

について、各トランザクション内での$\leq_1$と$\leq_2$の順序関係は、e_0を左端の$r_1(a)$から右へ順に一つずつ実行するとき、保存されている。

系列eを実行させたとき、$d \in D$の最終値を$d(e)$とし、$\delta(e) = (d_1(e), \cdots, d_{|D|}(e))$を系列$e$の実行後の各データ項目の最終値とする。また、$T_i$が読込んだ$d$の値を$d(e)_i$とし、$T_i$が読込んだすべてのデータ項目の値を$\tau_i(e) = (d_1(e)_i, \cdots, d_k(e)_i)$で表わす。そしてすべてのトランザクションが読込んだ値を$\tau(e) = (\tau_1(e), \cdots, \tau_n(e))$と表わす。

$w_i(d)$により$d \in D$に特定の値が書込まれるとき、版(version)が生じたといい、各$d \in D$に対し最新版の存在のみを許すものを<u>単版方式</u>(single-version)、最新版のみならず、<u>過去に書込</u>んだ版の存在も許すものを多版方式(multi-version)、版数をk個に制限したものを<u>k-版方式</u>(k-version)という。$w_i(d)$による版を$r_j(d)$が読込むとき、T_j reads-d-from T_iといい、すべてのreadに対しどの版を割当てるか決めた系列を<u>履歴</u>(history)という。多版方式、k-版方式は履歴において考察する必要がある。

$r_i(d_1, d_2, \cdots)$の後に$w_i(d_3, \cdots)$が続くという規則をもつトランザクションを<u>2-ステップモデル</u>、この規則に加えて、writeす

るデータ項目は必ずその前にreadするという制限をもつトランザクションを制限付き2-ステップモデル、またこれらの制限の無いタイプを一般モデルといい、以下ではまず一般モデルについて議論し、5章で他のモデルにも考察を及ぼす。

3. 直列可能性

Eを任意の系列の集合とし、次の等価関係及び直列可能性を定義する。

定義1[2]：すべての $i\ (1 \leq i \leq n)$、すべての $e, f \in E$ に対し

$$e \underset{\delta}{\sim} f \iff \delta(e) = \delta(f)$$

$$e \underset{i}{\sim} f \iff \tau_i(e) = \tau_i(f)$$

$$e \underset{\tau}{\sim} f \iff \forall i.\ e \underset{i}{\sim} f$$

s が直列な系列、すなわち、$s = T_1 T_2 T_3 \cdots$ と表わせるとき、

$e \underset{\delta}{\sim} s$ ならeをδ-直列可能、$e \underset{\tau}{\sim} s$ ならeをτ-直列可能、

$e \underset{\delta}{\sim} s$ かつ $e \underset{\tau}{\sim} s$ ならeをr-直列可能という。 □

いま次の3つの系列を考える。

$e_1 = r_1(a)\,w_2(b)\,w_1(a)$, $r_1(a) \leq_1 w_1(a)$

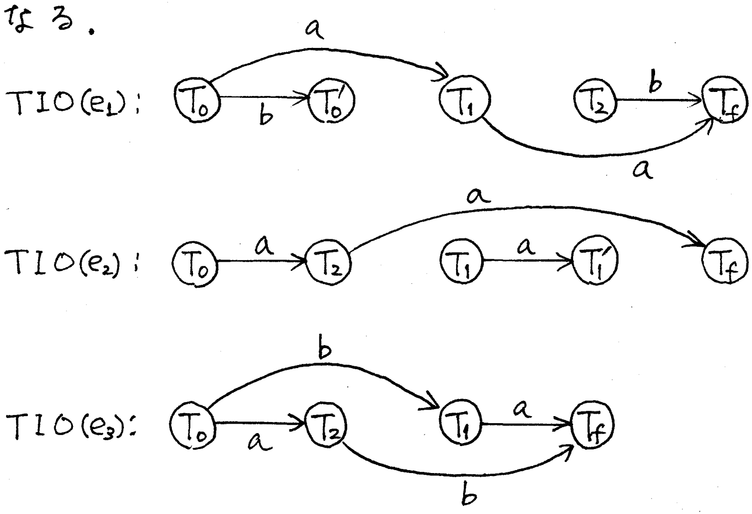
$e_2 = r_2(a)\,w_1(a)\,w_2(a)$, $r_2(a) \leq_2 w_2(a)$

$e_3 = r_2(a)\,w_1(a)\,r_1(b)\,w_2(b)$; $w_1(a) \leq_1 r_1(b)$, $r_2(a) \leq_2 w_2(b)$

ここで各系列の実行前にはすべてのデータ項目の初期値を書込むトランザクション T_0 があり、実行後にはすべてのデータ項目の最終値を読込むトランザクション T_f があるものとする。

各トランザクションに対し1つの節点をつくり、T_j reads-d-from T_i ならばラベル付有向枝 $T_i \xrightarrow{d} T_j$ をつけ、$(T_i, T_j):d$ と表わす。また $w_i(d)$ の値を読込むトランザクションが無いとき、$w_i(d)$ を無効書込み(useless write)といい、$(T_i, T_i'):d$ をつける。系列 e に対するこうしたグラフを TIO(e) と名付ける[5]と、TIO(e_1), TIO(e_2), TIO(e_3) は以下のようになる。

ある節点から発し、同じラベルをもつ枝の集合を interval という。TIOグラフの全節点を左から右へ枝方向に矛盾が生じない様に一列に並べるとき、有向閉路が無く、2つ以上の interval が重ならないならば、節点の全順序に関し DITS が存在するという。



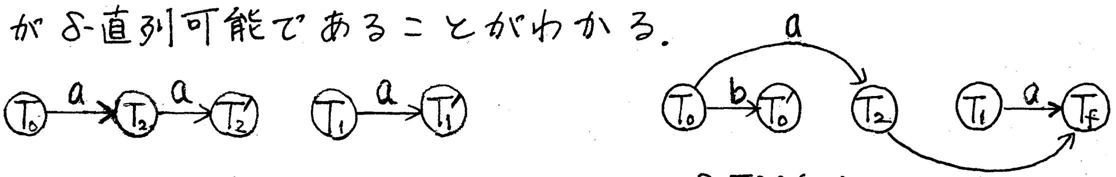
一般に系列 e が r-直列可能であるための条件は、TIO(e) に T_0 を最初とし T_f を最後とするような DITS が存在することであることが知られている。TIO(e_1) はこの条件を満たすが TIO(e_2) では $T_2 \quad T_1 \xrightarrow{a} T_1' \quad T_f$ のため、TIO(e_3) では $T_0 \quad T_2 \quad T_1 \quad T_f$ のため r-直列可能とはならない。TIOグラフの $(T_i, T_f):d$ をすべて $(T_i, T_i'):d$ にかえて節点 T_f を除いたグラフを t-TIOグラフということにすると次の定理が成立つ。

<u>定理1</u>：系列 e が τ-直列可能であるための必要十分条件は τ-TIO(e) に T_0 を最初とするDITSが存在することである。□

$r_j(d) \le_j w_j(d')$ となる $w_j(d')$ が存在しないような $r_j(d)$ を<u>無効読込み</u>(useless read)、T_i から T_f に有向路(path)が存在しないとき $r_i(d)$ を<u>不活性読込み</u>(dead read)といい、系列 e から無効読込み及び不活性読込みを除いた場合のTIOグラフを δ-TIO(e) と呼ぶ。

<u>定理2</u>：系列 e が δ-直列可能であるための必要十分条件は δ-TIO(e) に T_0 を最初とし T_f を最後とするDITSが存在することである。□

τ-TIO(e_2)，δ-TIO(e_3) をかくと、定理より e_2 が τ-直列可能、e_3 が δ-直列可能であることがわかる。



τ-TIO(e_2)　　　　δ-TIO(e_3)

<u>4. τ-，δ-直列可能性と r-直列可能性の意義と相互関係</u>

まず各直列可能性の正当性に関して検討を加えてみる。δ-直列可能という概念はPapadimitriou[8],[9]により導入されている。これは $\le_i$ の依存関係により各 $r_f(d)$ $(d \in D)$ に影響を及ぼす活性(live)なread, writeを逆にたどっていき、これら活性操作のみを考慮し、活性でないread, writeについては関知しないという立場である。その結果、あるシステムの利用者が論理的に首尾一貫していないデータ値を読込む可能性があり、問題が生じる場

合が考えられる．τ-直列可能性についてはBernstein, et al.[1]がその特性について論じている。つまり、実際にはシステムが動き出してからは次々とトランザクションが入力されてくるから、T_fは便宜上のものと考えられる．従って利用者に首尾一貫したデータ値が提供されてさえいればデータベースシステムとして充分機能を果していると考えられ、その点でτ-直列可能性の意義づけができる。またT_fについての別の見方として、これを文字通りの最終に実行するreadと考えるのではなく、データベースの状態を調べるために、同時にトランザクションが実行されていない任意の時点に挿入されるreadとみなした場合、システムの故障等を考慮し、保守的な立場からデータベースを首尾一貫した状態に保つという観点から、δ-直列可能性を見直す必要がある．一方、r-直列可能性はEswaran, et al.[4]等で用いられている．制約の強さから現実的には同時処理能力は少々劣るが、論理的には最も問題のない直列可能性の定義と考えられる．以上の定義のうちどれが現実的に最適かということに関しては、それぞれの立場で異なってくると思われる．ところで少なくともτ-直列可能またはδ-直列可能の性質をもつ系列ならば，適当な処理により，与えられた系列の順序を変えることなく，r-直列可能と同じ効果を残す系列となることが明らかになった。

なお以下において τ-, δ-, r-直列可能な系列のクラスを各々 $\mathcal{T}$, $\mathcal{D}$, $\mathcal{R}$ とし、多版 τ-, δ-, r-直列可能な系列のクラスを各々 $\hat{\mathcal{T}}$, $\hat{\mathcal{D}}$, $\hat{\mathcal{R}}$ と表わす。

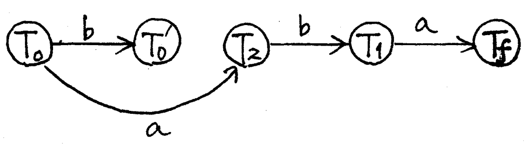
定理3：$e \in \mathcal{T} \cap \overline{\mathcal{R}}$ のとき、TIO(e)で $(T_i, T_f):d$ と $(T_k, T_e):d$ が重なっているなら、そのようなすべての $d \in D$ につき、$\tilde{e} = e * w_k(d)$ を繰返せば $\tilde{e} \in \mathcal{R}$。

系3：多版方式で T_f がどの版を読んでもよいなら $\hat{\mathcal{T}} = \hat{\mathcal{R}}$ □

ここで $e * w_k(d)$ は e の後に $w_k(d)$ を行なうという意味であり、系3は定理3を言い換えたものであるといえる。

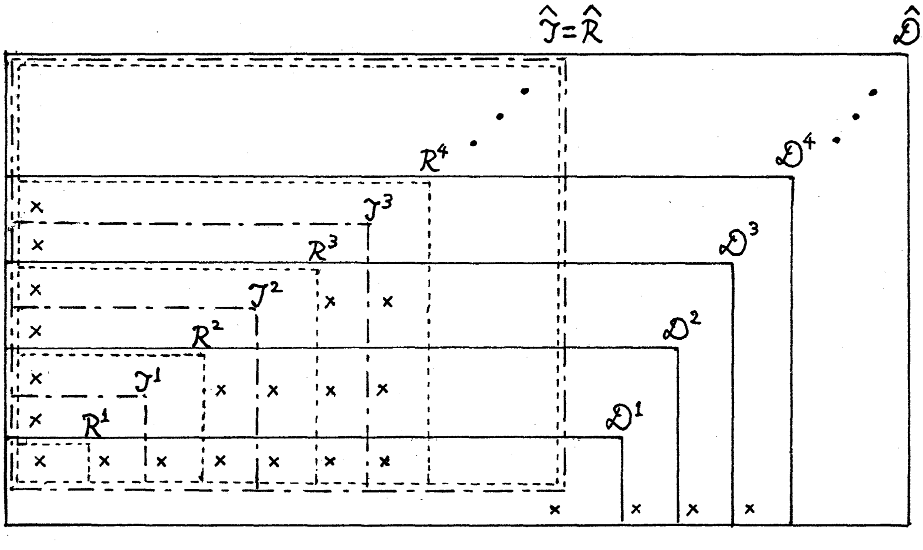
定理4：$h \in \hat{\mathcal{D}} \cap \hat{\mathcal{T}}$ なる履歴 h について、δ-TIO(h)で T_0 を最初とし T_f を最後とするDITSにおいて、τ-TIO(h)で $(T_i, T_j):d$ と $(T_e, T_m):d$ が重なるものとする。(但し、$T_i \ll T_e$) このとき、τ-TIOグラフにDITSが存在するまで $\tilde{h} = h * r_j(d_e)$ を繰返せば $\tilde{h} \in \hat{\mathcal{R}}$ が成立つ。

ここでの $r_j(d_e)$ は T_j reads-d-from T_e という意味である。また≪はグラフの節点の全順序で、$T_i \ll T_e$ なら T_i の方が T_e より左に並ぶものとする。e_3 についてこの定理4を適用すれば、$\tilde{h} = h * r_1(d_2)$ (hは単版と同じ割当て)で、TIO($\tilde{h}$)は下のようになる。



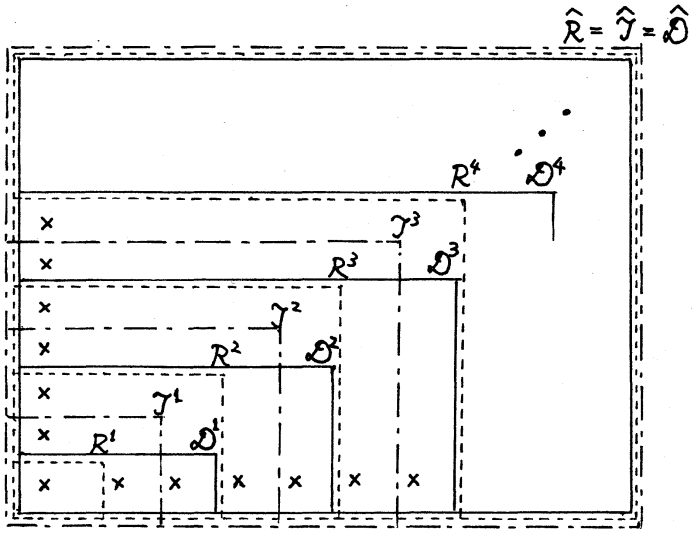
5. トランザクションモデルと版数に関する クラスの包含関係

トランザクションの一般モデルにおいて 1-版、2-版,… と版数を増していくと、δ-直列可能となるクラスは階層的に拡大していく[9]。このことはτ-, r-直列可能となるクラスについてもいえることがわかった。さらに2-ステップモデル、制限付き2-ステップモデルについても、これらの直列可能性のクラスは階層的に拡大していく。以上をもとに、一般モデル及び2-ステップモデルにおける各l版直列可能性のクラス(l= 1, 2, …)の包含関係は次のようになる。ここで例えば、k-版で直列可能なクラスを$\mathcal{T}^k$と表記する。



図中の×印は、その部分が空集合でないことを表わす。

次に、制限付き2-ステップモデルにおいて、各l版直列可能性のクラス(l=1,2,…)の包含関係は以下のようになる。



これらから、2-ステップモデル及び一般モデルでは、多版でτ-,γ-直列可能で受理できないが、1-版でδ-直列可能として受理できる系列があること。制限付き2-ステップモデルでは、k-版直列可能なクラスは(k-1)版以下の直列可能なクラスをすべて含むこと等が観察できる。

6. あとがき

本稿では文献[2],[3]で定義されたいくつかの直列可能性のクラスについて、文献[5]で導入されたTIOグラフを変形したグラフによって特性づけることを試みた。それらの議論を通して、直列可能性各クラスのもつ性質がより明らかになった。さらに各直列可能性のクラスの包含関係、階層性を調べた。

今後の課題としては、一度受付けた要求は却下を行なわないオンライン同時処理制御アルゴリズム[6]に、直列可能性のクラスのなかで有効に利用できるものがないかを調べること。

また各直列可能性の現実面からみて有意義な部分クラスで、しかもある系列がそのクラスに属するかどうかの判定が多項式オーダーの計算の手間で可能なものはないかを考えること等が挙げられる。

謝辞

有意義な助言をいただいた神戸商科大学加藤直樹助教授に深く感謝いたします。

References

[1] Bernstein,P.A., and Goodman,N., "Multiversion Concurrency Control-Theory and Algorithms", ACM TODS, Vol.8, No.4, Dec. 1983, pp.465-483.

[2] Brzozowski,J.A., "On Models of Transactions", Technical Report #84001, Dept. Appl. Math. Phys., Kyoto Univ., Kyoto, Japan, April 1984.

[3] Brzozowski,J.A., and Muro,S., "On Serializability", Technical Report #84012, Dept. Appl. Math. Phys., Kyoto Univ., Kyoto, Japan, July 1984.

[4] Eswaran,K.P., Gray,J.N., Lorie,R.A., and Traiger,I.L., "The Notions of Consistency and Predicate Locks in a Database System", Comm. ACM, Vol.19, No.11, Nov. 1976, pp.624-633.

[5] Ibaraki,T., Kameda,T., and Minoura,T., "Serializability with Constraints: DITS and Its Applications", Proc. 9th Int. Conf. on VLDB, Florence, Italy, Oct. 31 - Nov. 2, 1983, pp.49-51.

[6] Katoh,N.,Ibaraki,T.,and Kameda,T.,"Cautious Transaction Schedulers with AdmissionControl", TR84-1, Dept. Comput. Sci., SimonFraser Univ.,1984.

[7] Kiniwa,J., "On Several Serializability Classes in Database Systems", Master Thesis, Kyoto Univ., Feb. 1985.

[8] Papadimitriou,C.H., "The Serializability of Concurrent Database Updates", J.ACM, Vol.26, No.4, Oct. 1979, pp.631-653.

[9] Papadimitriou,C.H., and Kanellakis,P.C., "On Concurrency Control by Multiple Versions" ACM TODS, Vol.9, No.1, March 1984, pp.89-99.