移動体トランザクションの概略的同期方式

濱田 賢、滝沢 誠
東京電機大学理工学部
e-mail {hama, taki}@takilab.k.dendai.ac.jp

鉄道、飛行機等のシステムでは、運行計画が、集中的に静的に決定されている。車のナビゲーションシステム等の開発により、個々の自動車が道路の状況により、運行計画をたてることが可能となってきている。運行計画の決定は計算量が大きくなり、事故や渋滞等により計画の修正、変更は容易ではない。そのために、運行計画を分散的に、かつ動的に決定する方式について考察する。本論文では、列車や自動車等の移動体の移動を移動空間をロックするトランザクションとしてモデル化する。移動体が、後退可能かどうか、停止できるかどうかといった移動体の性質に基づいたロックとその解放方式について論じる。

1 はじめに

鉄道、自動車等の交通システム、工場等の部品の配送システム、通信網の経路選択 [16] 等は、移動体が空間を移動するシステムとしてモデル化できる。例えば、道路交通システムでは、自動車が移動体となり、高速道路等が移動空間となる。これらを、抽象化したシステムを、移動体システム [3, 5] とする。こうした移動体システムでは、移動体をどのように移動させるかが問題である。移動計画を詳細に決定することは、考慮すべき要因が多く、計算量が大きくなる。このため、[3, 5] は、移動空間を、複数の領域オブジェクトに階層的に分割し、移動経路を概略的に決めて、移動しながら詳細化していく方法が論じられている。概略的な経路とは、領域空間内の、より上位のオブジェクトについての経路である。各オブジェクト毎に管理者が存在し、管理者は領域の状態、例えば、混雑状況を把握しているとする。オブジェクト管理者からの情報を基にして、移動体が移動空間を移動するためには、移動空間内のオブジェクトを確保する必要がある。本論文では、移動体を空間内のオブジェクトをロックするトランザクションとしてモデル化する。[5] は、空間の階層に基づいてロックを行なう方式を論じている。飛行機は停止できず、高速道路での車は、後戻りできない。このように移動体は、その性質について種々の性質を持つ。本論文では、各種の移動体の性質に対するロック方式とロックの解放方式を検討する。自動車のように、移動体を消去できない場合、デッドロック等により、移動できない場合には、移動体を現在いるオブジェクトから他に移動させる必要がある。移動方法には、通過してきた経路、あるいは、他の経路を戻る等がある。本論文では、移動体の性質とロックの解放方式と後戻りの関連について検討する。

まず、第2章で、システムのモデルを示す。第3章では、概略的な経路から詳細な経路を求める方法について論じる。第4章では、移動体が移動するためのオブジェクトの獲得方法について述べる。

2 システムモデル

移動体システム T は、移動体集合 V と移動空間 S から構成される。移動体は、S 内を起点から終点まで移動する。

Broad Synchronization of Vehicle Transactions
by Satoshi HAMADA and Makoto TAKIZAWA (Tokyo Denki University)

2.1 移動空間

移動空間Sは、オブジェクトから構成される。各オブジェクトoは要素オブジェクト$o_1, \ldots, o_n$に分割される。ここで、oをo_jの親とする。このように、Sは木構造であり、これを領域木とする。任意のオブジェクトo_iとo_jの共通の先祖の中で、一番葉に近いものを、$lca(o_i, o_j)$とする。oのレベル$level(o)$を、根からoまでの路内のオブジェクト数とする。$level(o_i) < level(o_j)$のとき、o_iはo_jより上位にあり、$level(o_i) = level(o_j)$のとき、o_iとo_jは同じレベル($o_i \equiv o_j$)にあるとする。$depth(o)$を、oから子孫の葉までのレベル数で最大のものとする。各葉が同一レベルにある領域木を、均衡木とする。本論文では、均衡木を考える。

各オブジェクトoは、入力口集合$\{ip_1, \ldots, ip_k\}$($k \geq 1$)と出力口集合$\{op_1, \ldots, op_h\}$($h \geq 1$)を持つ。ip_iから移動体vを受け付け、op_jからvを出力することにより、vのoでの移動を示す。$o\rangle$と$\langle o$を、各々oのある出力口と入力口とする。$\langle o\rangle$は、oのあるip_iからop_jまでの経路を示し、これを基本経路とする。vが、$\langle o\rangle$を移動するためには、oの要素オブジェクトを通過する必要がある。どの要素オブジェクトを通過するかは、oがダイクストラ法[7]等により、最適なものを決定する。

$o_j\rangle$と$\langle o_k$が結合されているとき、$o_j\rangle \rightarrow \langle o_k$と書き、$o_j$から$o_k$まで、移動体を移動できる。ここで、各$\langle o$と$o\rangle$に対して、$o$のある要素$o_i$と$o_j$の1つの$\langle o_i$と$o_j\rangle$が各々対応すると仮定する。$o_1$から$o_n$までの経路$o_1\rangle \rightarrow \langle o_2\rangle \rightarrow \ldots \rightarrow \langle o_n$を$\langle\langle o_1\rangle, \langle o_2\rangle, \ldots, \langle o_n\rangle\rangle$と書く。同じ親$o$のオブジェクトのみからなる経路を$o$内の経路とする。

[例] 工場の搬送システムについて考える。工場は、地区、棟、ブース、通路の順で階層化されている。図1に工場の1地区を示す。$A, B, \ldots$は棟を、$\alpha, \beta, \ldots$はブースを、$t_1, t_2, \ldots$は入出力口を示す。二重線は各搬送路を示す。搬送路を、搬送台車が移動する。搬送台車は通路を両方向に移動できる。棟Tとブースλに対して、$T\lambda$はT内のブースλを示すとする。これの領域木を図2に示す。□

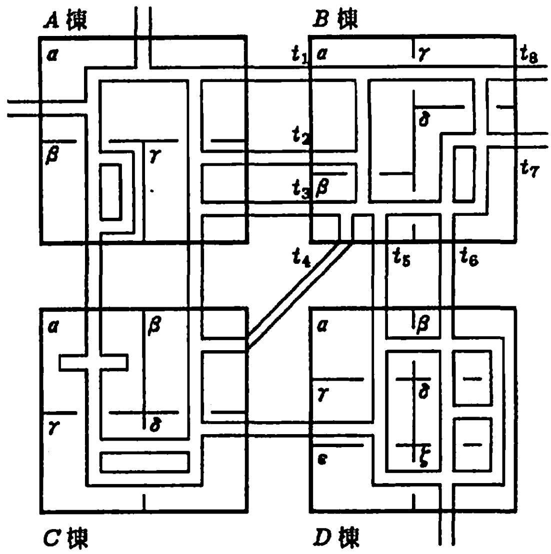

図1: 移動空間

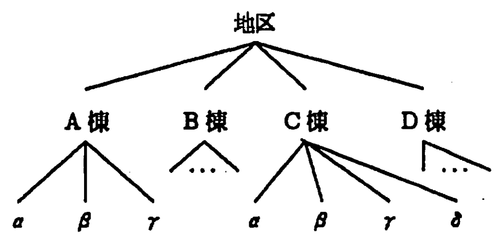

図2: 領域木

図1で、AからB内をt_1からt_6に移動し、Dに移動する場合を考える。B内の経路としては$\langle\langle\alpha\rangle, \langle\beta\rangle, \langle\delta\rangle\rangle$、$\langle\langle\alpha\rangle, \langle\gamma\rangle, \langle\delta\rangle\rangle$等がある。これを$\langle B\rangle^1$とする。一般に、$\langle o\rangle$の$i$レベル詳細化を$\langle o\rangle^i = \langle\langle o_1\rangle^{i-1}, \ldots, \langle o_n\rangle^{i-1}\rangle$ ($i \geq 1$)、$\langle o\rangle^0 = \langle o\rangle$とする。$\langle o\rangle^i$は1つ以上存在するので、$o$が移動時間を最小とするものを決定する。

$p_1 = \langle\langle o_1\rangle, \ldots, \langle o_n\rangle\rangle$と、$p_2 = \langle\langle o_1\rangle, \ldots, \langle o_{i-1}\rangle, \langle o_i\rangle^j, \langle o_{i+1}\rangle, \ldots, \langle o_n\rangle\rangle$ ($j \geq 1$)に対して、p_1はp_2よりも概略的(p_2はp_1よりも詳細)であり、$p_1 \ll p_2$とする。また、$\ll$は推移的関係である。経路pに対して、$p \ll q$で$q \ll r$なるrが存在しない経路qをpの最概略経路とする。最詳細経路も同様に定義される。移動体は、最終的には、最詳細経路を通過する。ある経路から、より概

略的な経路を求めることを概略化、この逆を詳細化とする。p を経路 $\langle\langle o_1\rangle, \ldots, \langle o_m\rangle\rangle$ とする。$o_1 < \ldots < o_m$, $o_1 \equiv \ldots \equiv o_m$, $o_1 > \ldots > o_m$ のとき、p を各々、上昇、平坦、下降経路とする。

[例] 図1で、$p_3 = \langle\langle C\gamma\rangle, \langle C\alpha\rangle, \langle A\rangle, \langle B\rangle\rangle$ は、$p_4 = \langle\langle C\gamma\rangle, \langle C\alpha\rangle, \langle A\beta\rangle, \langle A\alpha\rangle, \langle B\alpha\rangle, \langle B\gamma\rangle\rangle$ よりも概略的である。□

オブジェクト s と d 間の経路 p と q に対して、$p \ll q$ なる q が存在しない p を s から d への最概略経路とする。s と d の先祖で、$o = (s, d)$ の要素を各々 o_s と o_d とし、最概略起点と最概略終点とする。$\langle\langle o_s\rangle, \ldots, \langle o_d\rangle\rangle$ を s と d 間の最概略同位経路とする。図1で、$\langle\langle C\rangle, \langle A\rangle, \langle B\rangle\rangle$ は、$C\gamma$ と $B\delta$ 間の最概略同位経路である。C は $C\gamma$ の最概略起点で、B は $B\delta$ の最概略終点である。

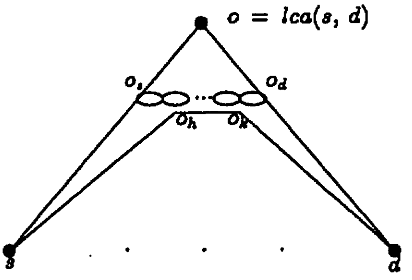

図 3: 最概略起点と終点

2.2 移動体の種類

移動体が、どのように移動できるかについて考える。例えば、自動車は停止できるが、飛行機は停止できない。このように、移動体は、移動空間内の移動についての制限がある。以上から、移動体は、以下の性質を持つ。

(1) 移動体の消滅 (消滅可能又は不可能)。

(2) 停止距離 (停止可能又は不可能)。

(3) 移動方向 (後退可能又は不可能)。

まず、移動中の、移動体を消滅できるか、できないかが問題となる。鉄道や自動車等の交通システムでは、移動体を消滅できないが、通信網では移動体のパケットを消滅できる。デッドロック等の障害に対して、移動体を消滅できない場合には、他の移動経路を見つけて移動させる必要がある。

次に、移動体の停止距離の問題がある。停止距離が小さく、すぐに停止できる移動体として、工場の搬送台車、市内電車等がある。停止距離が大きい移動体には、新幹線や高速道路を走行中の自動車等がある。また、飛行中の飛行機は停止できない。停止距離の長い移動体に対しては、移動前に、通過するオブジェクトをより詳細なレベルで獲得し、移動できることを保証する必要がある。本論文では、議論を簡単にするために、停止距離のあるものとないものとの2種を考える。

移動方向が一方向の移動体には、高速道路を走行中の自動車、航空路を飛行中の飛行機等がある。移動方向が両方向の移動体には、工場の搬送台車等がある。ここで、移動体が、移動してきた経路をそのまま戻るとき、これを後退とする。移動体には、後退可能と不可能の2種がある。通過したオブジェクトに戻ることを後戻りとする。後退では、移動してきたのと同経路を用いて後戻りする。

これら、移動体の性質により、移動空間を移動していくときの移動方式について考える。

3 経路決定戦略

起点から終点までの最適な詳細経路の決定は、計算量の点から困難な場合がある。また、各オブジェクト o の状況は、経路を決定した時点と、事故等の予測できない事象により、移動体 v が o に到着した時点では、異なる場合がある。従って、出発前に、終点までの詳細経路を決定するかわりに、概略的経路を決定し、移動するに従って、経路を逐次詳細化する戦略を用いる。v が、s から d に移動しようとする。このとき、v は、以下の手順 [3, 5] により最概略経路を決定する。

[最概略経路決定手順](v は s から d に移動)

(1) 最概略起点 o_s と最概略終点 o_d を求める。$o = lca(s, d)$ は、最概略同位経路 $p = \langle\langle o_s\rangle, \langle p_1\rangle, \ldots, \langle p_m\rangle, \langle o_d\rangle\rangle$ を決定する。

(2) s から $\langle p_1$ までの上昇経路の決定を o_s に、

$p_m\rangle$ から d までの下降経路の決定を o_d に、依頼する。

以上によって得られた経路 $b = \langle\langle o_1\rangle, \ldots, \langle o_n\rangle\rangle$ が最概略経路である [図 4]。次に、b の詳細化を行う。

[詳細化]　b から、$\langle\langle o_1\rangle^{i_1}, \langle o_2\rangle^{i_2}, \ldots, \langle o_h\rangle^{i_h}, \ldots, \langle o_k\rangle^{i_k}, \ldots, \langle o_n\rangle^{i_n}\rangle$ が得られる。□

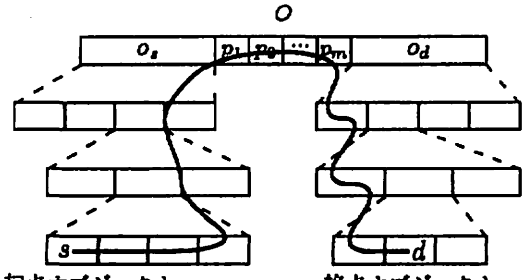

図 4: 最概略経路

b 内の各 $\langle o_j\rangle$ の詳細化レベル i_j について考える。詳細化レベルは、移動体の停止距離に関している。前述したように、移動体には、停止可能と、停止不可能なものとがある。停止不可能移動体に対しては、これから移動する経路はなるべく詳細化して、確実に移動できることを保証する必要がある。これに対して、停止可能移動体では、停止できるので、これから移動する経路を、停止不可能移動体より詳細化する必要はない。このことから、v の種類により、以下の詳細化手順により、各 $\langle o_j\rangle$ の i_j を決定する。ここで、$h_j = depth(o_j)$ とする。

[詳細化手順][図 5]

(1) v を、停止可能移動体とする。H とはどの位遠方まで深く詳細化するのかの指標である。

$$i_j = h_j \times (1-(j/n)^{\alpha}) \ (\text{for } i \le H)。$$
$$i_j = j/n \ (\text{for } H < i < n)。$$

(2) v を停止不可能移動体とする。I とは、v が移動し続けられる距離である。J は I より先をどの程度まで詳細化するのかの指標である。

$$i_j = h_j \ (\text{for } i \le I)。$$
$$i_j = h_j * (1-(j/n)^{\beta}) \ (\text{for } I \le i \le J).$$
$$i_j = j/n \ (\text{for } J < i < n)。□$$

停止可能移動体は、移動体自身が詳細化の度合を決定できる距離が長い。停止不可能移動体では、$\langle o_1\rangle$ から $\langle o_I\rangle$ を最詳細化し、移動できることを保証する。このため、移動体自身が詳細化の度合を決定できる距離は短い。停止不可能移動体では、停止可能移動体より短い距離で、最上位のオブジェクトから、下位のオブジェクトまで詳細化のレベルを変化させるため、$\alpha \le \beta$ である [図 5]。

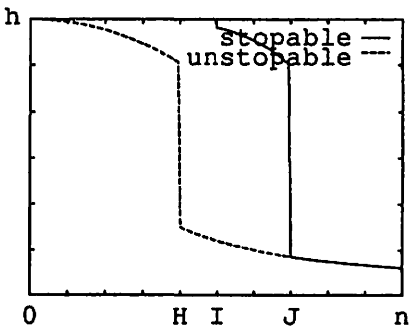

図 5: 詳細化のレベル i_j

4　同期方式

移動体は、経路の決定時と移動時に、移動空間内のオブジェクトのロックを行なう。

4.1　オブジェクトの状態

移動体 v は、オブジェクト o に対して、$\langle o\rangle$ の移動を要求する。o は、以下に述べるオブジェクトの状態情報を基にして、経路の決定を行なう。本論文では、各移動体の移動速度を自身で制御できないものとする。各基本経路 $t = \langle o\rangle$ は、以下の属性値を持つ。$cap(t)$ を t 内を同時に通過できる移動体の総数。$hold(t)$ を t の通過を要求 (通過中も含む) している移動体の総数。$time(t)$ を t を移動体が通過するときの最小移動時間。これは、t 内が空であるときの移動時間である。$cap(t)$ を t の容量、$time(t)$ を移動時間とする。t の混雑度

$cong(t) = hold(t) / cap(t)$ と移動時間 $etime(t) = time(t) / (1 - cong(t))$ とする。$etime(t)$ は、移動体の t の通過時間の目安を与える実効移動時間である。$cong(t) = 1$ のとき、t は移動体をこれ以上受け付けれず、t を通過しようとする移動体は待つことになる。経路 p に対して、$cap(p)$ と $time(p)$ を各々 p 内の基本経路の容量と移動時間の総和とする。

o と連接する o_i と o_j に対して、o_i から o_j に移動するための $o_i\rangle \rightarrow \langle o\rangle \rightarrow \langle o_j$ なる基本経路 $\langle o\rangle$ の集合を考える。この集合内の基本経路で、$cap(t)$ / $time(t)$ が最大の t を、o_i から o_j への最代表基本経路とする。最代表基本経路は、高速道路のような幹線を示している。

o と、その要素 $o_1, \ldots, o_n$ の関係について考える。o の基本経路 t の詳細経路のうち、最代表基本経路によるものの集合を P_t とする。t の容量と移動時間は $cap(t) = \sum_{p \in P_t} cap(p)$ 、$time(t) = (\sum_{p \in P_t}(cap(p) \times time(p))) / \sum_{p \in P_t} cap(p)$ とする。

4.2 ロック方式

移動体 v の移動を、オブジェクトのロックを行なうトランザクション [1, 4] と考える。$\langle o\rangle^1 = \langle\langle o_1\rangle, \ldots, \langle o_n\rangle\rangle$ とする。$\langle o\rangle$ を移動する v は、$\langle o_i\rangle$ を移動する部分トランザクション v_i $(i = 1, \ldots, n)$ から構成される。さらに、v_i は、o_i の各経路についての部分トランザクションから構成される。この意味で、移動体トランザクションは階層型 [11, 12] である。はじめに、v は、以下の手順により、o に $\langle o\rangle$ の移動を要求する。

[ロック方式]

(1) v は $\langle o\rangle$ をロックする。

(2) $v_1, \ldots, v_n$ を実行する。即ち、$\langle o\rangle$ が詳細化される。□

$v_1, \ldots, v_n$ が完了したとき、v は $\langle o\rangle$ を移動したことになる。v が o を移動した後に、$\langle o\rangle$ のロックをどのように解放するかを考える。このとき以下の3つの方式がある。

[解放方式]

(3-1) v が移動体 (即ち、階層の根) のとき、全てのロックを解放する。

(3-2) $o_1, \ldots, o_n$ についてのロックを解放するが、o についてのロックは解放しない。

(3-3) o を解放する。□

ここで、(3-1)、(3-2)、(3-3) を各々、閉、半開、開方式とする。閉方式では、v が目的地に到着するまで、通過したオブジェクトのロックが保持されるので二相ロック方式 [4] である。開方式では、v が、o を通過すると、直ちに o を解放する。半開方式では、v が o を通過したとき、o の下位のオブジェクトを解放するが、o のロックは保持される。

前述したように、移動体 v には、後退可能と後退不可能の二種がある。v は、o を通過後に、o のロックを解放すると、空間の利用効率が上がる。しかし、事故等により、予定していた経路を v が移動できなくなる場合を考える必要がある。まず、後退可能移動体を考える。v が予定した経路を移動できないとき、通過したオブジェクトのロックを保持しておくならば、後退を行なえる。即ち、閉方式を用いると、いつでも必ず後退できる。閉方式では、目的地に着くまで通過したオブジェクトのロックが保持されるので、移動空間の利用効率が低下する。これに対して、o 内の経路として $p_1 = \langle\langle o_1\rangle, \ldots, \langle o_n\rangle\rangle$ を、v が移動したとする。閉方式では、$\langle o_1\rangle, \ldots, \langle o_n\rangle$ のロックに加えて、$\langle o\rangle$ が解放される。これに対し、半開方式では、$\langle o_1\rangle, \ldots, \langle o_n\rangle$ のロックが解放されるだけであり、$\langle o\rangle$ のロックは保持される。従って、o 内を後戻りしようとするとき、$\langle o\rangle$ を詳細化することにより、後戻りのための o 内の経路 q を得れる。但し、q は、p と同一とは限らない。$\langle o_1\rangle, \ldots, \langle o_n\rangle$ のロックは解放されているので、これを、他の移動体が利用できるので、閉方式よりも空間の利用効率は向上する。

これに対し、後退不可能移動体 v は、後退できない。このために、オブジェクトを通過したのち、オブジェクトのロックは全て解放できる。即ち、

v は開方式を用いれる。事故や渋滞等で、予定通り移動できない場合には、あるオブジェクト、例えば、起点まで戻り、他の経路を見つけなければならない。このため、これから移動するオブジェクトをロックするときに、同時に、現在いるオブジェクトまで、戻れるような経路もロックする。この経路を補償経路とする。ここで、v が s から d に移動することを考える。v は s から d までの経路 $p_1 = \langle\langle s\rangle, \ldots, \langle d\rangle\rangle$ のほかに s から d までの補償経路 $p_2 = \langle\langle d\rangle, \ldots, \langle s\rangle\rangle$ もロックすることとする。経路 p の補償経路を $\bar{p}$ と書く。

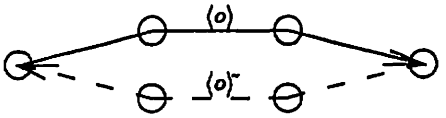

図 6: 補償経路

[例] 図1で、経路 $p = \langle\langle C\delta\rangle, \langle D\gamma\rangle, \langle D\alpha\rangle, \langle B\beta\rangle\rangle$ の補償経路の例として、$\bar{p} = \langle\langle B\beta\rangle, \langle A\beta\rangle, \langle C\delta\rangle\rangle$ がある。□

4.3 後退可能移動体

後退可能移動体 v の $t = \langle o\rangle$ についてのロックと半開解放手続きについて考える。

[ロック手続き] $hold(t) < cap(t)$ ならば、$hold(t) = hold(t) + 1$; v が根で、t が最代表基本経路の場合には、t を含む上位の基本経路 u に対して、$hold(u) = hold(u) + 1$ とする。□

即ち、上位のオブジェクトから詳細化に行い、下位のオブジェクトにロックがなされるように下降していく。t をロックすることは、下位の詳細経路の1つを不特定に獲得したことである。従って、t の詳細化は、下位の詳細経路を特定化することである。

ロックの解放は以下のようにして行なう。

[ロックの解放] $hold(t) = hold(t) - 1$; v が根の場合には、t を含む上位の基本経路 u に対して、$hold(u) = hold(u) - 1$ とする。□

ロックと反対に、下位からも上位はオブジェクトが解放されていく。半開方式では、v が $\langle o\rangle$ を通過した後に、$\langle o_1\rangle, \ldots, \langle o_n\rangle$ のロックは解放されるが、$\langle o\rangle$ のロックは保持される。

4.4 後退不可能移動体

後退不可能移動体 v の $\langle o\rangle$ についてのロックと解放手続きについて考える。ここで、$t = \langle o\rangle$ とする。

[ロック手続き] $hold(t) < cap(t)$ かつ、t を後戻りする補償経路 $\tilde{t}$ について、$hold(\tilde{t}) < cap(\tilde{t})$ ならば、$hold(t) = hold(t) + 1$; v が根で、t が最代表基本経路の場合には、t を含む上位の基本経路 u に対して、$hold(u) := hold(u) + 1$; t の上位オブジェクト $\neq \tilde{t}$ の上位オブジェクトなら、$hold(\tilde{t}) = hold(\tilde{t}) + 1$; $\tilde{t}$ を含む上位の基本経路 $\tilde{u}$ に対して、$hold(\tilde{u}) = hold(\tilde{u}) + 1$ とする。□

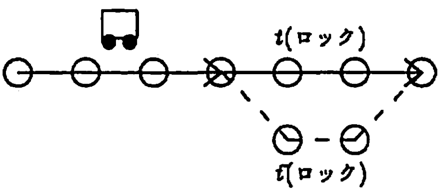

図 7: 後戻り不可能な移動体のロック

後退不可能移動体のロックでは、移動経路の他に補償経路もロックするので、空間の利用効率は低下する。但し t と $\tilde{t}$ の上位のオブジェクトで同一のものがあれば、そのオブジェクトを重複してロックしないため、空間の利用効率はあまり低下しない。

ロックの解放は以下のようにして行なう。

[ロックの解放] if v が次に移動するオブジェクトのロックが成功、かつ t を通過ならば, $hold(t) = hold(t) - 1$; t が根の場合には、t を含む上位の基本経路 u に対して、$hold(t) = hold(t) - 1$; t の上位オブジェクト $\neq \tilde{t}$ の上位オブジェクトなら、$hold(\tilde{t}) = hold(\tilde{t}) - 1$; $\tilde{t}$ を含む上位のオブジェクト $\tilde{u}$ に対して、$hold(\tilde{u}) = hold(\tilde{u}) - 1$ とする。□

本方式では v が、事故、渋滞等である所まで戻る場合には、$\tilde{t}$ によってロックされている経路を移動して現在いる所まで戻ることになる。

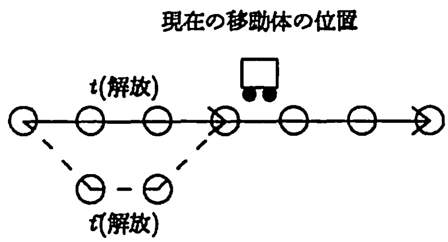

図 8: 後戻り不可能な移動体のロックの解放

4.5 デッドロック

複数の移動体がオブジェクトを待ち合ってデッドロックとなる場合がある。システムの状態は、移動体を節点とした待ちグラフ [6, 9] により示せる。移動体 v が、他の移動体 w によりロックされているオブジェクトを待つ時、v から w に有向辺を設ける。この待ちグラフ内に巡回閉路が存在する場合に、デッドロックが生じている。

[例] 4つの移動体 v_1, v_2, v_3, v_4 が、各々オブジェクト a, b, c, d をロックしている。次に、各々 b, c, d, a をロックしようとすると、デッドロックとなる。この状態を示す待ちグラフを以下に示す。□

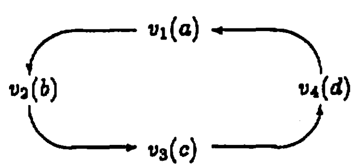

図 9: デッドロック

デッドロック閉路内のある移動体 v を選択して、v ロックしているオブジェクトを解放させることにより、デッドロックを解除できる。従来のトランザクションと異なり、移動体を消去 (アボート) できない。このために、v を $\langle o \rangle$ とは別の経路に移動させて、$\langle o \rangle$ のロックを解放する必要がある。この移動方法としては、移動体の種類により 2 種がある。

(1) v が後退可能移動体のとき、v を後退させる。

(2) v が後退不可能移動体の場合には、v は、後退できないので、目的地への新しい経路を見つけて、その経路に従って移動する。

この移動方法は、前節で示したロック方法と関連している。

5 おわりに

本論文では、目的地までの経路を動的に決定していく方式について述べた。本方式では、まず、目的地までの経路を概略的に決定し、移動してから詳細化する方式を示した。移動体の移動を、移動空間内のオブジェクトをロックする階層型トランザクションとしてモデル化し、その同期方式を示した。とくに、移動体の種類を検討し、その種類毎の同期方式を示した。停止不可能移動体に対しては、経路をなるべく詳細化し、確実に移動できるようにする方式を示した。後退不可能移動体に対しては、オブジェクトの移動後、直ちにオブジェクトを解放するが、このかわりに、後退するための補償経路をロックする方式を示した。後戻り可能移動体には、半開方式により、移動後、下位のオブジェクトのロックは解放するが、上位のオブジェクトは保持する方式を示した。これにより、これまで通過してきた経路を同じでもないかもしれないが後退できる。本方式は、一般的な移動体問題に適応できる。

参考文献

[1] Ahmed, K. E.: *Database Transaction Models for Advanced Applications*, Morgan Kaufmann, 1992.

[2] Deen, S. M.: *Cooperating Agents - A Database Perspective, Proc. of International Working Conf. on Cooperating Knowledge Base Systems*, (1990).

[3] Deen, S. M., Hamada, S., and Takizawa, M.: *Broad Path Decision in Vehicle System, Proc. of International Conf. on Database and Expert Systems Applications (DEXA92)*, pp. 8–13 (1992).

表 1: ロックの解放方式とその特徴

解放方式	同時実行度	後戻り	補償方法	適合する移動体例	
				後退可	後退不可
閉	×	○	後退	○	×
半開	△	△	上位層で補償 経路をロック	○	×
開	○	×	葉レベルで補償 経路をロック	×	○

○は適している、△中程度である、×は好ましくない、を示す。

[4] Eswaren, K.P., Gray, J., Lorie, R. A., and Traiger, I. L.: *The Notion of Consistency and Predicate Locks in Database Systems, CACM*, Vol.19, No.11, pp. 624–637 (1976).

[5] 濱田 賢、長谷川 正裕、滝沢 誠：協調的交通システム、情報処理学会、マルチメディア通信と分散処理研究会, DPS–52, pp. 77–84 (1992).

[6] Holt, R. C.: *Some Deadlock Properties on Computer Systems, ACM Computing Surveys*, Vol.14, No.3, pp. 303–328 (1972).

[7] 石畑 清：アルゴリズムとデータ構造、岩波講座ソフトウェア科学, 岩波書店, Vol.3, (1990).

[8] 岸本 了造：分散型協調マルエージェント型知的通信網、情報処理学会「B-ISDN 時代におけるマルチメディア通信と分散処理シンポジウム」論文集, pp. 37–49 (1991).

[9] Knapp, E.: *Deadlock Detection in Distributed Databases, ACM Computing Surveys*, Vol.19, No.4, pp. 303–328 (1987).

[10] Korth, F. Henry.: *Locking Primitives in a Database System, JACM*, Vol.30, No.1, pp. 55–79 (1983).

[11] Lynch, N. and Merritt, M.: *Introduction to the Theory of Nested Transactions, MIT/LCS/TR 367*, 1986.

[12] Moss, J. E.: *Nested Transactions: An Approach to Reliable Distributed Computing, The MIT Press Series in Information Systems*, 1985.

[13] Steeb, R., Cammarata, S., Hayes-Roth, A. F., Thorndyke, W. P., and Wesson, B. R.: *Distributed Intelligence for Air Fleet Control, Readings in Distributed Artificial Intelligence*, pp. 90–101 (1987).

[14] Takizawa, M., Hasegawa, M., and Deen, S. M.: *Interoperability of Distributed Information System, Proc. of the International Conf. on Interoperable Database Systems (IMS'91)*, pp. 239–242 (1991).

[15] Takizawa, M. and Deen, S. M.: *Lock Mode Based Resolution of Uncompensatable Deadlock in Compensating Nested Transactions, Proc. of the 2nd Far–East Workshop on Future Database Systems, Kyoto*, pp. 168–175 (1992).

[16] Tanenbaum, A. S.: *Computer Networks, Prentice-Hall International, Inc* (1989).