データベースマシンGRACEの実行制御方式

3C-6

伏見信也*　喜連川優**　田中英彦***
(*三菱電機　計算機製作所　**東京大学　生産技術研究所　***東京大学　工学部)

1. はじめに

従来の並列関係データベースマシンの研究に於いては，個々の関係演算の並列処理アルゴリズムや実装方式に関するものが多く，複数関係演算の並列実行の制御に関するものは少ない。一方で，近年の研究により関係データベースマシンの環境ではこれら複数演算の実行制御に関するオーバヘッドが極めて大きく，マシンの性能はこれら実行制御ソフトウェアの構成，或いはその実装法に大きく左右されることが明らかとなった[1]。

関係データベースマシンGRACEでは，この問題に対し関係演算の実行をタスクと呼ばれる単位で管理，制御することにより，効率の良い実行制御ソフトウェアを実現している[2]。本稿では，その制御方式について述べ，またGRACEのソフトウェアシミュレータ上での実装方式について報告する。

2. 実行制御方式

2.1 タスク

GRACEでは，関係代数演算はタスク(task)として実行される。タスクはオペランドデータ全体と演算の実行環境（マシンのハードウェア資源の集合）から成る。例えば結合演算の実行を行う場合，オペランドデータを格納している記憶モジュール群(Source)，結合操作を行う処理モジュール群，結合結果を格納する記憶モジュール群(Sink)，及びこれらモジュール間でデータ転送を行う為に用いるチャネル群が結合演算の実行環境を形成する。

演算の実行に際しては，まずこれら演算の実行環境が割付けられる。必要なハードウェア資源の割付けに成功すると，制御モジュールからの発火命令によりオペランドデータが自律的に実行環境中を移動し，演算の実行が行われる。GRACEでは全ての関係演算はオペランドデータの流れに沿って線型時間で実行される[2]。

タスクは自律的な計算実体であり，マシンの制御モジュールの介入無しで与えられた関係演算を実行する。

2.2 制御ソフトウェアの構成

GRACEの制御ソフトウェアは上記のタスクの制御を中心として構成され[2]，最下位から順に(a)個々のモジュール内の処理プログラムから構成される基本アルゴリズム層，(b)各タスクの実行環境中のハードウェア資源間の協調制御を行うタスク内制御層，(c)タスクの発火制御やタスク間の協調制御を行うタスク間制御層，(d)投入されたトランザクションをタスク群に展開し，またこれらに対する障害回復制御を行うトランザクション管理層，からなる。

3. ソフトウェアシミュレータ

上記の実行制御方式の正当性，有効性の確認，更にその実現方式の検討を目的としてGRACEのソフトウェアシミュレータを作成し，その上にGRACEの制御ソフトウェアを実装した。本シミュレータは，GRACEの処理モジュール群，記憶モジュール群，制御モジュール，及びこれらモジュール群を接続する為のリングバス，及び各モジュールとリングバスを接続するリングバスインターフェースユニットをシミュレートし，全体が論理的なクロックで同期しながら動作する。

4. 制御ソフトウェアの実装

ソフトウェアシミュレータ上に基本アルゴリズム層，タスク内制御層，タスク間制御層を実装した。基本アルゴリズム層は，各モジュールのシミュレータ内のハードウェアシミュレーションプログラムそのものである。タスク内制御層はタスク内のハードウェアモジュール群の間でのデータストリーム制御プロトコルとして実装される。本プロトコルは各モジュールに付加されているリングバスインターフェースユニットの機能としてシミュレートされる。これら2つの層に関しては既に報告[2]しており，本稿ではタスク間制御層の実装について述べる。本層は主に制御モジュール上に実装される。

4.1 タスク間制御層の機能と構成

タスク間制御層の具体的な機能は以下の通りである。

(1)トランザクション管理層から送られてきた木構造のタスク群（トランザクションステップ）を内部構造に変換し，タスク制御テーブルに登録する。

(2)発火可能（子タスクが全て実行を終了しているタスク）を同定し，これにハードウェア資源を割付けてタスクの実行環境を設定する。

(3)走行可能（発火可能で且つ実行環境の割付けが終了して

Execution Control of Database Machine GRACE by
Shinya Fushimi*, Masaru Kitsuregawa**, and
Hidehiko Tanaka**
*MITSUBISHI Electric Co. ** University of Tokyo

いる）なタスクにデータ転送用のチャネル（データ収集チャネル及びデータ分配チャネル）を割付けて，これを発火する。

(4)走行中のタスクの実行状態を監視し，タスクの終了を検出すると，タスクから実行環境を解放する。また，その親タスクに対して子タスクの終了を通知する。

タスク間制御層のソフトウェアはこれらの諸機能を以下のソフトウェアモジュール群によって実装している。

(a)Resource Manager: 発火可能なタスクに対し，必要なモジュール群の割付けを制御する。

(b)Channel Allocation Manager: 走行可能なタスクに必要なデータ転送用チャネルを割付け，タスクの発火を制御する。

(c)Task Monitor: 走行中のタスクの実行状態を監視し，実行を終了したタスク内のモジュールに対し，タスクからの離脱を指示する。

(d)Channel Deallocation Manager: 実行を終了し，その実行環境を解放されたタスクから更にデータ転送用のチャネルを解放する。

これらの内，(1)及び(3)の為に特定のリングバスチャネル（資源管理チャネル，及びタスク制御チャネル）が用いられる。これらソフトウェアモジュールは，制御モジュールに次々と到着するチャネルの種別により動的に選択，駆動される。

4.2 タスク制御テーブル

マシン内のタスクはタスク制御テーブルのエントリとして登録される。タスク制御テーブルエントリは以下のフィールドを持つ。

(1)トランザクションID，トランザクションステップID:
タスクが属するトランザクション，及びトランザクションステップのIDを示す。

(2)データ収集サブタスクID，データ分配サブタスクID:
タスク内制御層で管理されるタスクの下位層（サブタスク）の制御に用いられる。

(3)親タスクID: タスクは一般に木構造をしており，本フィールドが親タスクのタスク制御テーブルエントリのポインタとなる。

(4)記憶モジュール数，及び処理モジュール数: タスクの実行に必要とされる記憶モジュールと処理モジュールの台数を示す。

(5)データ収集チャネル数，データ分配チャネル数:
タスク内のデータ転送に必要とされるこれら2種のチャネル数を示す。

(6)記憶モジュール数，及び処理モジュール数の現在値:
現時点でタスクが保有している記憶モジュール，処理モジュールの台数を示す。

(7)子タスクの総数，走行中の子タスク数，走行を終了した子タスク数: タスクの木構造に於ける子タスクの実行状態を示す。これにより当該タスクの発火可能性の判定を行う。

(8)タスクの状態: 次節で述べるタスクの状態遷移に於ける現在のタスクの状態を示す。

(9)優先度: タスクの実行順を定める為のpriorityを示す。

(10)次エントリ，前エントリ: タスクの状態に応じてタスク制御テーブルエントリはdoubly linked listを形成する。この為のポインタとして用いられる。

4.3 タスクの状態遷移

タスクは以下の状態を順に遷移する。

(1)Wait（子タスクの終了待ち）: タスクの初期状態であり，子タスクの実行終了待ちであることを示す。

(2)ResourceAlloc(モジュール群の割付け待ち): 全ての子タスクの実行が終了すると，タスクは本状態に遷移し，実行に必要な処理モジュール，記憶モジュール群の割付け待ちとなる。

(3)ChannelAlloc（チャネルの割付け待ち）: モジュール群が割付けられると，タスクはデータ転送用のチャネルの割付け待ちとなる。必要なデータ転送用チャネルが全て割付けられた瞬間にタスクは（データストリーム制御プロトコルにより）自動的に発火し，演算の実行を開始する。

(4)Running(走行中): タスクが演算を実行中であることを示す。

(5)StatCollect(演算結果の統計情報の収集): タスクが演算の実行を終了した後，演算結果に対する統計情報（結果タプルに対する動的クラスタリングの分布情報等）を収集していることを示す。

(6)Dealloc(資源の解放待ち): タスクがチャネルやモジュール群の解放を行っていることを示す。

(7)Done（終了）: タスクの実行が全て終了し，タスク制御テーブルからのエントリ削除待ちであることを示す。

5. おわりに

データベースマシンＧＲＡＣＥの実行制御方式について述べ，そのソフトウェアシミュレータ上での実装方式について報告した。本シミュレータによる詳細な性能評価等については稿を改めて発表したい。

参考文献

〔1〕DeWitt,D. et al, GAMMA - A High Performance Dataflow Database Machine, Proc. of VLDB 86.

〔2〕Fushimi,S. et al, An Overview of The System Software of A Parallel Relational Database Machine GRACE, ibid.