

並列オブジェクト指向言語 A-NETL の 言語処理系

4P-1

岩本 吉史 吉永 努 馬場 敬信
(宇都宮大学工学部)

1. はじめに

我々は、並列オブジェクト指向の概念に基づいたトータルアーキテクチャ A-NET (Actors NETWORK) を開発中である^[1]。この核言語として、オブジェクト間の通信を重視した並列オブジェクト指向言語 A-NETL、及び、その機械語の設計を行い、言語処理系を作成した。

本稿では、A-NETL、機械語の特徴と、その言語処理系について述べるとともに、オブジェクト間通信のコスト軽減についての検討を行う。

2. 並列オブジェクト指向言語 A-NETL

A-NETL では、比較的粒度の大きなオブジェクトが、互いに通信をしながら協調して問題を解決する。これらのオブジェクトは、状態変数とメソッドの記述からなる。

A-NETL には、以下の特徴がある。

1) オブジェクトの動的生成・消去

オブジェクトのひな型をもつクラスオブジェクトにメッセージを送ることにより、動的にオブジェクトを生成することができる。

また、不要となったオブジェクトの消去も、delete メッセージを送ることによって行うことができる。

2) 三種類のメッセージの型

ABC L / 1^[2]と同様、現在、過去、未来の三種類の型のメッセージがある。

3) 並列メッセージ送信

複数のオブジェクトに対して、同一のメッセージを同時に送ることができる(マルチキャスト)。

4) 複数メッセージ受信

二つ以上のメッセージを受信して起動されるメソッドを記述することができる。

5) メッセージの取り消し

オブジェクトを並列に動作させる環境では、必要な返答を得た後でも処理を続けているオブジェクトがある可能性がある。このオブジェクトに対して別の処理を依頼したいときなどに、必要なくなった処理の取り消しを行うことができる。

6) プリミティブメソッド

整数、文字列、配列等のデータオブジェクトに対して送られるメッセージは、プリミティブメソッドとして定義された機械命令により実行される。

7) グループ宣言

効率の良いオブジェクトの割り付けを行うために、次

の二種類のグループ宣言を設けた。

① group: 互いに通信量の多いオブジェクト群を近くに配置して、オブジェクト間の通信コストを下げるができる。

② collect: オブジェクト群を同一の PE に配置して、ルータを介さずに、遅延のないメッセージ通信を行い実行速度を低下させないようにすることができる。

3. 機械語の特徴

A-NET 機械語の特徴を以下に述べる。

1) 高機能な命令セット

オブジェクトをコンパクトなコードにコンパイルし、メッセージ送信などを高速に実行するために高機能な命令セットを定義した。メッセージ送信や、配列、リスト操作などのプリミティブメソッドは、基本的にソース文と 1 対 1 に対応する機械語によって実現される。

2) アドレス形式

バイト境界を持つ可変長命令をバイトアドレスで指定し、データは、一語 40 ビットを単位とするワードアドレスで扱う。

3) ベースレジスタ方式

再配置可能なコードにするために、変数へのアクセスは、ベースレジスタを介して行われる。このため、変数、リテラルを表すオペランドはベースレジスタからのオフセットとなっている。ベースレジスタには、状態変数用、一時変数用、リテラル用、の三種類がある。

4) 相対分岐

3) と同様の理由で、分岐命令は相対分岐とした。

4. A-NETL 言語処理系

A-NET 言語処理系は、大きく分けてコンパイラとローダからなる。

4.1 コンパイラ

コンパイラは、ソースプログラムを再配置可能なコンパイルドオブジェクトに変換する。このとき、別にオブジェクト・セレクト情報も出力する。

1) コンパイルドオブジェクト

このファイルには、ソースプログラムを翻訳した機械語と共に、各オブジェクトを実行するのに必要なオブジェクト辞書や、メッセージによって起動されるメソッドに必要なメッセージ辞書などの情報も付加されている。

Language Processing System for a Parallel Object-Oriented Language A-NETL

Yoshifumi IWAMOTO, Tsutomu YOSHINAGA, and Takanobu BABA

Utsunomiya University

2) オブジェクト・セクタ情報

このファイルには、オブジェクト情報、セクタ情報、グループ情報が記述されている。分割コンパイルを可能としているために、これらの情報は、各ソースファイルごとに生成される。

4. 2 ロード

ローダは、プログラムの実行に必要なコンパイルオブジェクトをリンクしてロードモジュールを生成する。このときに、各々のオブジェクト・セクタ情報を統合して、グループ情報を考慮しながらオブジェクトを割り付けるPEを決定する。そして、統合されたオブジェクト表とセクタ表をつくり、それにしたがって、コンパイルオブジェクトを書き換えてロードモジュールを生成する。

5. 高機能命令の使用頻度

A-NETLによる応用例のコンパイル結果を整理して、高機能命令である通信命令と、リスト、配列操作命令の使用頻度を示したのが図1である。各応用例は、以下のものである。

例1：NAND回路を組み合わせて、XOR回路をシミュレートするプログラム。

例2：最小値を持つオブジェクトを探索するプログラム

例3：オブジェクトのネットワークの中で中心のものを探索するプログラム。

例4：オブジェクトのネットワークの中で、あるオブジェクトをリードとして選び、そのことを他のオブジェクトに知らせるプログラム。

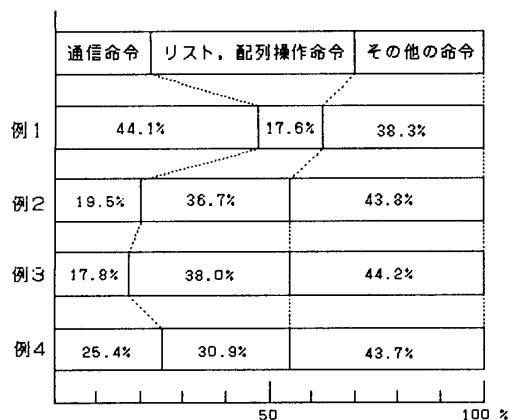


図1 応用例中の高機能命令の使用頻度

この図から判るように、通信命令とリスト、配列操作命令を合わせると、どの例でも50%を超えている。これは、A-NETLが、1) オブジェクト間通信を中心にして問題解決を行うこと、2) オブジェクトは、マルチキャストを行うために、他のオブジェクトをリストや配列の形で管理している場合が多いことなどが主な理由と考えられる。

6. 通信コスト軽減の重要性

このように、応用プログラムに占める通信命令の割合は、比較的高い。したがって、高速化を達成するためには、メッセージパッシングの高速化と、通信コストの軽減が必要になる。

そのために、A-NETでは、1) ファームウェアによる通信命令の高速化、2) グループ宣言による静的オブジェクト割り付けの最適化、3) 応用問題領域の要請に合わせたネットワークアーキテクチャの可変性、の三点から通信コストの軽減を図っている。

ここでは特に、言語処理系の立場から、オブジェクト割り付けの最適化について述べる。

オブジェクトの最適な割り付けは、PE間の距離を表す相互相関行列をD、グループ宣言による情報を含めたオブジェクト間の関連の深さを表す相互相関行列をRとすると、Rの要素を入れ換えて $\sum \sum d_{ij} r'_{ij}$ が最小になるような相互相関行列R'を見つけ出すことに相当すると考えられる。この方法だと、Dの要素を変えることにより、どんなネットワークアーキテクチャにも対応することができる。しかし、この最適化については、いくつかの問題点がある。

1) オブジェクト間の関係の定量化

オブジェクト間の通信量は、動的に変化するため、オブジェクト間の関係の定量化は難しい。A-NETLでは、グループ宣言によりオブジェクト間の関係をユーザに定義してもらい、この問題に対処している。

2) 計算量の増大

オブジェクトの最適化は、組み合わせの問題と考えることができる。したがって、単純に総当たり方式で最適解を見つけようとする、高並列を前提とした場合、計算量の増大は避けられない。

3) 動的オブジェクト

A-NETLでは、オブジェクトを動的に生成することができるため、応用例によっては、この事を考慮して割り付けを行わないと、意味のないものになってしまう危険性がある。

7. おわりに

A-NETLとその言語処理系について述べた。そして、応用例からオブジェクト割り付けの最適化の重要性を指摘し、検討した。今後、本稿で述べた問題点をふまえながら、オブジェクト割り付けの最適化を行うシミュレータを試作し、評価をしていく予定である。

謝辞 日頃貴重なご意見を頂く東工大・米澤明憲教授に感謝する。なお、本研究は、一部文部省科学研究費(課題番号62550255オブジェクト指向高度並列AIマシンに関する研究)の援助による。

参考文献

- [1]馬場 他：“並列オブジェクト指向トータルアーキテクチャA-NET”，並列処理シンポジウム'89，A4-1(1989)
- [2]米澤：“並列オブジェクト指向言語ABCL/1による並列処理記述とその枠組みの研究”，信学論，Vol. J71-D(1988)