

自動分割システム RODS におけるデータフロー制御

6M-1

○高橋透 福田享子 松山実 横井利彰

武蔵工業大学

1. はじめに

近年、ネットワーク環境が普及してきたが、この環境を効果的に利用するソフトウェアの開発は容易でない。現在、ネットワーク環境で動作するソフトウェアの開発を支援するシステム RODS¹⁾ (Repetitive Optimum Dividing System)の開発を行っている。本報告では、RODSの作成するソフトウェア内におけるデータフロー制御について述べる。

2. RODSの構成

RODS上では、ソフトウェアはNDFD(Network Data Flow Diagram)というCASE技術のデータ・フロー・ダイアグラムを拡張した独自のデータ・フロー・ダイアグラムを用いて記述する。

NDFDエディタはソフトウェアをNDFD表記で記述し、NDFDソースとして出力する。

自動分割システム(Automatic Dividing System)は、NDFDソースを分割し、ネットワーク上の各マシンへ配送する。自動分割システムがNDFDソースを分割したものをNDFDピースと呼ぶ。

NDFD処理系は各マシンへ配送されたNDFDピースを、実行可能コードに変換する。

各マシン上で実行されたソフトウェアは、その実行履歴が記録される。履歴解析システム(Log Data Analyzer)は、この実行履歴を解析し、解析データを自動分割システムに送る。

自動分割システムは、送られた解析データを基に再分割・再配送を行う。これを繰り返すことにより、実行時間の短い、より最適な分割へと近づけることができる。

RODSの構成を図1に示す。

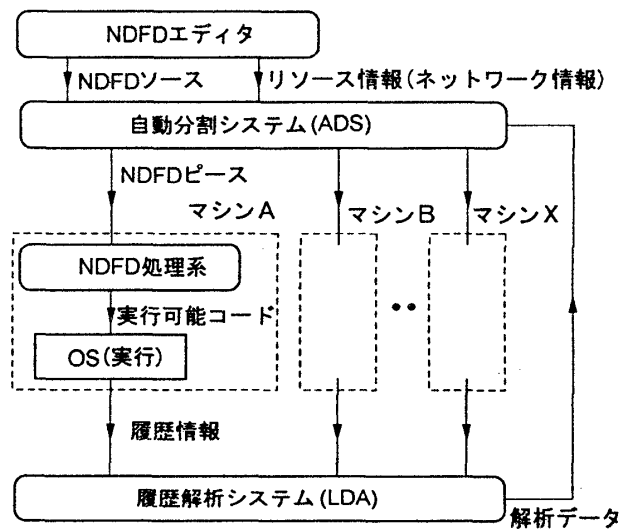


図1 RODSの構成

3. NDFD処理系

NDFD処理系はNDFDピースから実行可能コードを生成する。NDFD処理系は、NDFDからC言語ソースへとトランスレートするNDFDコンパイラとC言語処理系により構成される。

NDFD処理系は実行コード生成の際に、マネージャと呼ばれる4つのC言語関数群を組み込む。

1. メイン・マネージャ

他のマネージャの管理(起動, 初期化, 終了処理)を行う。

2. データフロー・マネージャ
ノードの状態の管理およびノード間のデータフローを制御する。

3. ネットワーク・マネージャ
ネットワークを介した際のデータ転送を受け持つ。

4. ログ・マネージャ
データフローおよびネットワーク・マネージャの実行状況を記録する。履歴解析システムはこの記録を解析する。

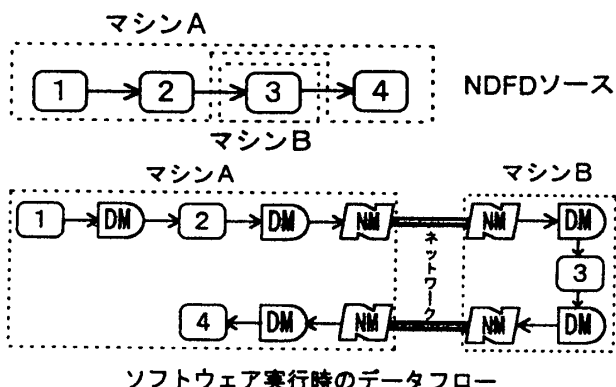
4. データフロー・マネージャ

NDFD は処理の最小単位をノードとし、ノードの動作に以下の5段階を定義している。

- 生成 ノードの初期化
- 待機 入力データが全て揃うまで待つ
- 実行 入力データの基に処理
- 送出 データが全て受け取られるまで待つ
- 消滅 ノードの動作終了

以上の定義を基に、ソフトウェアのデータフロー制御を行う。

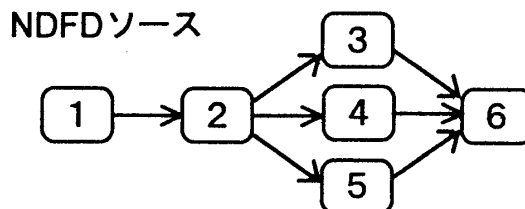
ソフトウェア実行の際にノードとノードのデータ転送はデータフロー・マネージャを介して行われる、さらにネットワークを介したデータ転送はネットワーク・マネージャが仲介する。(図2)



- DM** データフロー・マネージャ
- NM** ネットワーク・マネージャ
- データ

図2 ソフトウェア実行時のデータの制御

以上のデータフロー制御により、ネットワーク上の3台のマシンを用いて負荷分散を行いながら実行する場合と、マシン1台のみで実行する場合の実行時間の比較を行う。(図3)



ガントチャート

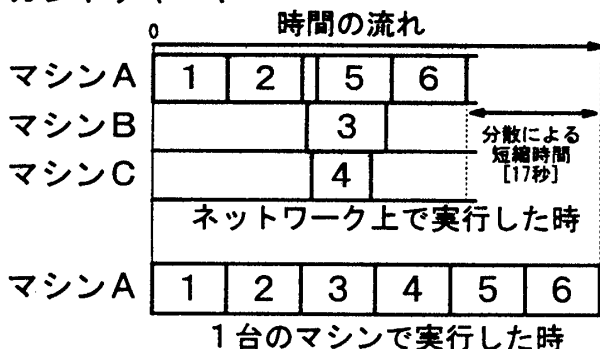


図3 ネットワーク上の実行とスタンドアローン実行の比較

各ノードは、マシンAで実行すると10秒程度の時間のかかる処理であり、負荷分散により約17秒の時間の短縮ができた。

5. まとめ

NDFD を用いたデータフロー制御により、ネットワーク環境に対応したソフトウェアを作成することができた。

ソフトウェアの処理をネットワーク上のマシンに分散して実行することにより、実行時間の短縮ができた。

今後、マシンの数・ノードの種類を増やした上でデータフロー制御を行う。作成されたソフトウェアを実行し動作検証を行う。

参考文献

[1] 山下利夫,松山実,横井利彰,“自動分割システム RODS における NDFD 処理系について”, 情報処理学会第46回全国大会, 講演論文集(分冊5),pp.261-262,1993