

ネットワーク環境における負荷分散に関する研究

7H-4

石井浩一 山下利夫 松山実
武蔵工業大学

1. はじめに

ネットワーク環境で協調動作するソフトウェアの開発を支援するシステムRODS^{[1][2]}(Repetitive Optimum Dividing System)を我々は開発している。本システムの基本構成はFig.1である。

RODS上では、DFD(Data Flow Diagram)をRODS用に拡張したNDFD(Network Data Flow Diagram)を用いて記述する。NDFD表記を用いたソフトウェアは専用のエディタであるNDFDエディタにより作成され、その内容はNDFDソースとして出力される。

ネットワーク情報と履歴解析システム(Log Data Analyzer)の解析結果を参照して自動分割システム(Automatic Dividing System)がNDFDソースを分割し、ネットワーク上の各マシンへ分配する。自動分割システムがNDFDソースを分割したものをNDFDピースとする。

各マシンに分配されたNDFDピースはそれぞれのNDFD処理系であるNDFDコンパイラにより実行可能なコードにコンパイルされる。現在はNDFDピースをC言語へトランスレートすることにより実行可能コードを生成する。

各マシン上で実行されるソフトウェアは実行履歴を取られる。この実行履歴は履歴解析システムにより解析される。解析結果は自動分割システムで参照され、再分割・再分配される。これにより実行時間が短い最適な分割が可能となる。

本稿は、自動分割システムについて述べる。

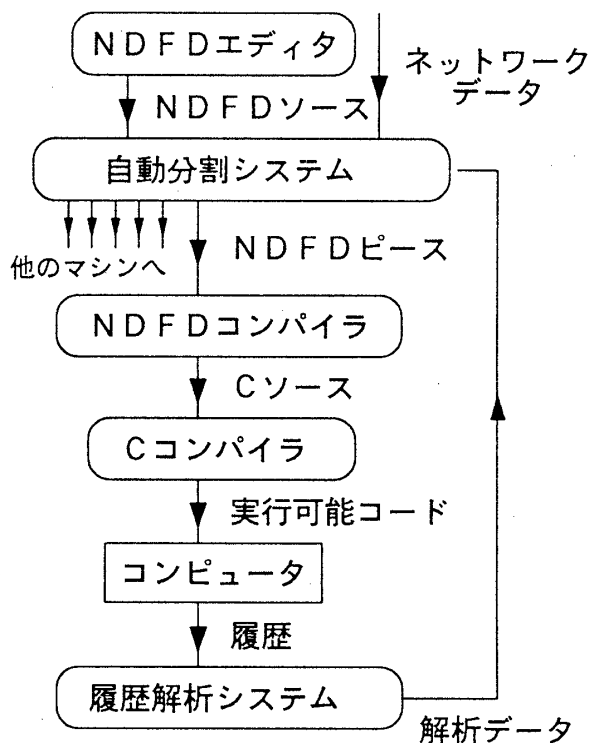


Fig.1 RODSの基本構成

2. 自動分割システム

自動分割システムはNDFDソースをNDFDピースに分割し各マシンに分配する。NDFDはノードという処理単位と矢印で記述される。ノードは分割不可能なプリミティブノードと階層構造を持つノードがある。本稿ではプリミティブノードのみを扱うものとする。

また分割はスケジューリングを考慮することで、実行時間がより短くなるようにする。分配後実行時に、実行するマシンの変更ができない静的なスケジューリングと、実行時マシンの変更が可能な動的なスケジューリングとに分け

ることができる。スケジューリングパターンを検索する際に各ノードの実行負荷を用いてシミュレートするが、この時使用される実行負荷は期待値として事前に与えられているものとする。

2.1 静的スケジューリング

実行時にマシンの変更ができないということは一部もしくは全てのノードが排他的であることを意味する。つまり一つのノードは唯一のマシンのみに存在し、他のマシンには存在しない。本稿では静的なスケジューリングの場合、全て排他的であるものとする。

最も基本的な静的スケジューリングのパターン検索方法として総当たり検索法がある。さらに検索時間を短縮できる最長経路法^[3](Critical Path Method)がある。総当たり検索法により最適解を求めることができるが、検索に多くの時間を要する。 n 個のノードを m 台のマシンに分割する方法は式(1)のような検索回数を要する。

$$\text{検索回数} = m^n \cdots \cdots (1)$$

また最長経路法による検索の場合、最適解と一致することが多く、最適解が求まらないときでも必要十分解として扱えることが多い。

2.2 動的スケジューリング

実行時にマシンの変更ができるという点からノードが重複的であることを意味する。つまり一つのノードが複数のマシン上に存在する。

この場合も最も基本的なスケジューリングパターン検索は総当たり検索法である。しかし、検索時間は静的スケジューリングに比べ飛躍的に増大する。 n 個のノードを m 台のマシンに分割する方法は式(2)のような検索回数を要する。

$$\text{検索回数} = 2^{m \cdot n} \cdots \cdots (2)$$

またノードが複数存在するため、最長経路が多重に存在する。このため最長経路法による検索は長い検索時間を必要とするため実用的では

ない。そこで遺伝的アルゴリズム^[4](Genetic Algorithms)を用いた検索を行なう。遺伝的アルゴリズムは遺伝子の発生や突然変異の過程によりランダム検索の能力を持ち、必要十分解が速く求まる。ただし、遺伝子型や評価方法、ローカルミニマムへの対処などの問題点が残る。

3. まとめ

以上のアルゴリズムにより静的及び動的な負荷分散が可能である。現在、自動分割システムは独立したシミュレーションとして研究を行っている。今後、他のシステムと協調させることにより、基準となる負荷単位の設定をする。さらに、階層構造を付加したノードを扱うアルゴリズムを開発すると同時に、他のアルゴリズムなども検討し、より適切な動的スケジューリングを実現する。

参考文献

- [1] 山下利夫, 松山実, 横井利彰, “ソフトウェア自動分割システムRODSの概要”, 情報処理学会第45回全国大会講演論文集(分冊5), pp. 319-320, 1992
- [2] 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
- [3] 富田真治, “並列処理マシン”, オーム社, 1989. 5
- [4] Goldberg. D. E, “Genetic algorithms in search, optimization and machine learning”, Addison-Wesley(1989).