

ソフトウェア自動分割システム RODS における 履歴解析システムの開発

5L-4

○福田享子 高橋 透 松山 実 横井利彰

武蔵工業大学

1 はじめに

現在我々が開発中のソフトウェア自動分割システム RODS^[1] (Repetitive Optimum Dividing System) は、ネットワーク環境で動作するソフトウェア開発を支援するシステムである。本システムの基本構成を図1に示す。

RODSで開発したソフトウェアを各コンピュータ上で実行すると、そのソフトウェアの実行履歴がファイルに記録される。履歴解析システムでは、この実行履歴を一定期間ごとに解析し、より確かな負荷の期待値やソフトウェアの分割が適切に行なわれていない部分を修正の必要な箇所として出力する。この解析結果は自動分割システム^{[2][3]}で参照され、ソフトウェアの再分割が行なわれる。そのためRODSでは、ソフトウェアの実行を重ねるごとに、より効果的な分割が行え、ネットワーク環境の変化にも対応することが可能となる。

今回は、履歴解析システムの解析手法について述べる。

2 ソフトウェアの実行履歴

NDFD処理系^[4]では、以下に挙げる4つのマネージャと呼ばれるC言語の関数群を用いる。

- メイン・マネージャ
- データフロー・マネージャ
- ネットワーク・マネージャ
- ログ・マネージャ

ソフトウェアの実行履歴は、データフロー・マネージャ及びネットワーク・マネージャを介してログ・マネージャによって記録される。記録される実行履歴を以下に示す。

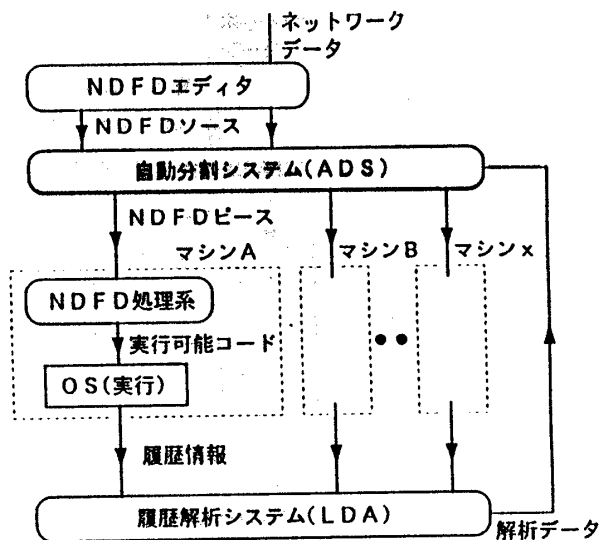


図1: RODSの基本構成

2.1 データフロー・マネージャによる履歴

データフロー・マネージャがデータの配送を行なう際、各ノードごとに以下に挙げる時刻を測定する。

- A: 最初のデータが入力された時刻
- B: 最後のデータが入力された時刻
- C: 処理が実行される時の時刻
- D: 処理が終了した時の時刻

これらの時刻より、以下の3つの時間を求め、履歴としてファイルに記録する。

- (B - A): データが全て揃うまでの時間
- (C - B): データが揃ってから処理が開始されるまでの時間
- (D - C): ノードの実行時間

2.2 ネットワーク・マネージャによる履歴

ネットワークを介した処理では、共有化分割(2台以上のコンピュータで実行可能な同一ノードが存在する

Development of LDA (Log Data Analyzer used in RODS)
Michiko FUKUDA, Toru TAKAHASHI, Minoru MATSUYAMA, Toshiaki YOKOI
Musashi Institute of Technology

ような分割)したノードを実際に実行したコンピュータ名を記録する。

また、実際にネットワークを介してノード間のデータ転送を行った場合、転送したデータの量を記録する。

さらに、一定時間ごとの各コンピュータの負荷(現在実行キュー内の平均ジョブ数を用いている)の状態を記録する。

3 履歴解析システム

ログ・マネージャにより記録されたそれぞれの実行履歴を、履歴解析システムによって解析を行なう。

3.1 データが揃うまで及び処理が開始されるまでの時間

これらの処理に時間がかかるということは、そのノードで処理が滞っていることになる。そこでノードごとに処理の滞る割合を求め、その原因をコンピュータの負荷と共に解析し、滞っているノード及び解決策を出力する。解決策としては以下の3つが挙げられる。

- 分割の見直し…(A)
- ノードの共有化分割…(B)
- コンピュータの性能評価の見直し…(C)

具体例を表1に示す。

表1: 解析例

ノード	滞る割合	負荷	共有化分割	対策
1	33	0.50	no	—
2	100	0.51	yes	C
3	50	0.08	yes	A
4	100	0.88	no	B
5	0	—	yes	—

3.2 ノードの実行時間

最初から正確な負荷の期待値を用意することは困難である。そこで、各ノードごとに平均実行時間を求め、この値を自動分割システムが分割する際に参照する各ノードの負荷の期待値とする。

3.3 共有化分割したノードの実行コンピュータ

共有化分割されたノードには、実行される確率の最も高い代表ノードというノードがあらかじめ設定されている。そこで、そのノードを実際実行した割合をコンピュータごとに求める。そして、代表ノードが適切であるかを解析し、代表ノードが不適切な場合、設定の見直しを求める。

3.4 ネットワークを介したデータ転送量

イーサネットを介したデータの転送量は、1回当たり1024バイトであり、多量のデータ転送は好ましくない。そこで、データ転送回数が複数回行なわれる割合の多いノードを修正必要箇所として出力し、データを出力するノードと入力されるノードが、同一コンピュータになるよう分割の見直しを要求する。

3.5 コンピュータの負荷測定

ネットワークに接続された各コンピュータの平均負荷を求め、負荷がほぼ平均的に分割されているかを解析し、負荷の著しい偏りがある場合、分割の見直しを求める。

4 おわりに

以上の解析により、より正確な負荷の期待値や分割の不具合な箇所を出力することができる。現在履歴解析システムは、シミュレーションデータによる解析を行なっているため、今後は、他のシステムとの融合をはかると共に、この解析結果が自動分割システムにどのように有効であるかの検証を行う方針である。

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] 石井浩一, 山下 利夫, 松山実, “ネットワーク環境における負荷分散の研究,” 情報処理学会第48回全国大会講演論文集(分冊5), pp.187-188, 1994.
- [3] 石井浩一, 春木篤幸, 松山実, 横井利彰, “ネットワーク上の分散処理におけるスケジューリング,” 情報処理学会第49回全国大会講演論文集(分冊1), pp.295-296, 1994.
- [4] 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.