

分散網管理における動的負荷分散制御方式の提案

3U-8

吉原 貴仁

杉山 敬三

小花 貞夫

(株) KDD 研究所

1 はじめに

SNMP (Simple Network Management Protocol) などの網管理方式 (以下, 集中管理方式と呼ぶ) では処理が管理ノード (マネージャ) に集中し, また, 網管理情報収集の際のポーリングに必要な帯域が被管理ノード (エージェント) 数の増大とともに増大するため, 大規模網への適用に限界があるなどの問題が指摘 [1, 2, 3, 4, 5] されている. このため, 網管理処理の一部を記述したスクリプトをエージェントにダウンロードして実行する管理方式 [1, 2, 3] やエージェントに移動して網管理処理を自律的に行うモバイルソフトウェアエージェントを用いる管理方式 [4, 5] (以下, 両管理方式を分散管理方式と呼ぶ) の提案がある.

しかしながら, 従来の分散管理方式ではエージェントの処理負荷を考慮せずにスクリプトがダウンロードされるため, 特定のエージェントに処理負荷が偏り, 必ずしも網全体で処理負荷の均衡がとれない問題点がある. 本稿ではこの問題点を解決するため, マネージャやエージェントの処理負荷ならびにポーリングや通知などスクリプト実行時に必要な帯域に応じてマネージャやエージェントなどで処理を行わせ, 動的な負荷分散を図る, 動的負荷分散制御方式を提案する.

2 分散管理方式と動的負荷分散制御の必要性

2.1 分散管理方式

スクリプトのダウンロードおよびソフトウェアモバイルエージェントによる管理方式はいずれも分散管理方式に分類されるが, 以下本稿では前者の管理方式を対象とする.

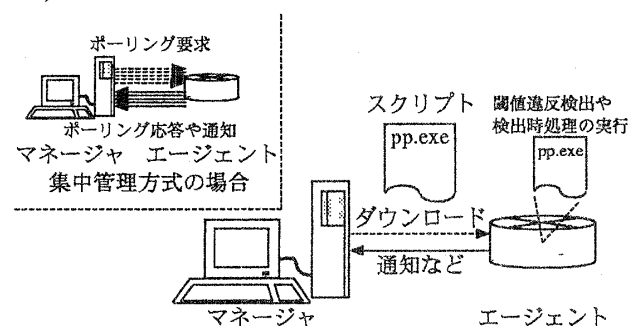


図 1: スクリプトのダウンロードによる分散管理方式.

マネージャは管理対象, 閾値, および閾値違反検出時の処理などを記述するスクリプトをエージェントにダウンロードする (図 1). エージェントではスクリプト実行により収集する管理情報を必要に応じて集約・加工し, 指定される情報だけをマネージャに通知する.

Proposal on Dynamic Load Balancing Scheme for Decentralized Network Management
 Kiyohito YOSHIHARA, Keizo SUGIYAMA,
 and Sadao OBANA
 KDD R&D Laboratories Inc.

2.2 動的負荷分散制御の必要性

従来の分散管理方式では, エージェントの処理負荷を考慮することなくマネージャからダウンロードされたスクリプトが実行されるため, 特定のエージェントに処理負荷が偏り, 必ずしも網全体としての処理負荷の均衡がとれない. ここでは, 例えば, ルータにおけるパケット転送処理性能の劣化などが懸念される. このため, 処理負荷に応じた動的な負荷分散制御が必須となる.

3 動的負荷分散制御方式の提案

以下に, マネージャやエージェントの処理負荷ならびにポーリングや通知などスクリプト実行時に必要な帯域に応じて処理を動的に分散させ, 網全体として処理負荷の均衡を図る, 動的負荷分散制御方式を提案する.

3.1 基本原理

(A) 動的な処理の切り替え

ポーリングや通知などスクリプト実行時に必要な帯域のみならずマネージャやエージェントの負荷に応じて, エージェントだけでなくマネージャやこれら以外のプロキシやメディエーション装置などの中間ノードで処理させるようマネージャが動的に切り替える.

(B) 切り替えの基準

処理負荷の偏りを各ノードの CPU 利用率と全ノードの平均 CPU 利用率との差とし, これと処理速度等の CPU 諸元およびスクリプト実行時に必要な帯域に基づく. マネージャは周期的に CPU 利用率の値を収集するためのスクリプトを事前に各ノードにダウンロードする. またスクリプトの記述から, マネージャやエージェントでそれぞれ実行する場合に必要な帯域をマニュアルで算出しマネージャに事前登録する. マネージャ, エージェント, および中間ノードが同一管理ドメインにある場合, 中間ノードで実行する際に必要な帯域は両者の和とする. また, 上記の基準だけでは切り替えられない場合 (後述の (C) 参照), マネージャやエージェントが事前に割り当てる優先度を用いる.

(C) 切り替え対象と切り替え先の決定

処理負荷の偏りの許容範囲 (許容偏在度 (%)) と処理に利用できる最大許容帯域 (許容帯域 (bps)) を設ける. 許容偏在度を超える CPU 利用率最大のノードにおける複数スクリプトから, CPU 利用率が許容偏在度を超えないノードへの切り替えにともなう必要帯域の減少が最大 (減少するものがなければ増加が最小) となるスクリプトを切り替え対象とする. また, CPU 諸元を加味した CPU 利用率が最小のノードを切り替え先に決定する. この際, 許容帯域を超える場合には切り替えずに当該ノードの過負荷を網運用者に通知し, 優先度の低いスクリプトの実行を停止する.

(D) 切り替えの実行

マネージャは上記 (C) で決定されたスクリプトを新

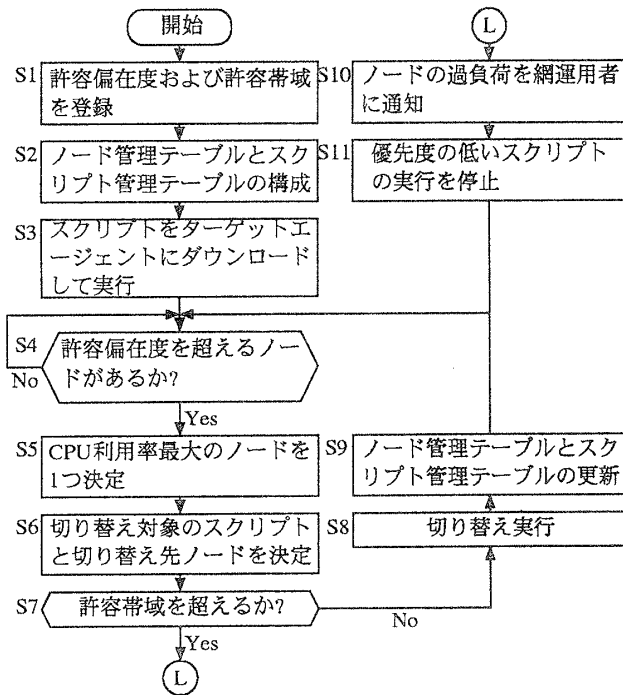


図 2: 提案方式のフローチャート。

たな切り替え先にダウンロードして実行を開始し切り替え元でのスクリプト実行を終了する。またこれにともない、スクリプトのダウンロード状況や起動状況など以降の切り替えに必要な情報 (3.2 節参照) の更新を行う。

3.2 提案方式の概要

図 2 に提案方式のフローチャートを示す。

はじめに許容偏在度および許容帯域の登録 (図 2 S1), ならびにスクリプト実行時に必要な帯域ならびに CPU 利用率等を登録・収集し, ノード管理テーブルとスクリプト管理テーブル (図 3) の構成 (図 2 S2) を行う。ノード管理テーブルは各ノードの CPU 利用率を管理する。また, 各ノード毎にスクリプト管理テーブルを提供し, スクリプト名, マネージャおよびエージェント上で実行する際に必要となる帯域, ならびに, スクリプトが管理情報収集の対象とするエージェント (以下, ターゲットエージェントと呼ぶ) を管理する。次いで, 必要帯域が最小となるようスクリプトを各ターゲットエージェントにダウンロードし実行する (図 2 S3)。

許容偏在度を超えるノードがある場合 (図 2 S4), ノード管理テーブルから CPU 利用率最大のノードを 1 つ決定し (図 2 S5), さらに 3.1 節 (C) にしたがって, スクリプト管理テーブルから切り替え対象と切り替え先を決定する (図 2 S6)。

切り替えにより許容帯域を超えるか否かを検査し (図 2 S7), 超えない場合には切り替えを実行し (図 2 S8), ノード管理テーブルとスクリプト管理テーブルの更新 (図 2 S9) の後, S4 の処理を継続する。許容帯域を超える場合には当該ノードの過負荷を網管理者に通知し (図 2 S10), 優先度の低いスクリプトの実行停止 (図 2 S11) の後, S4 の処理を継続する。

(a) ノード管理テーブル

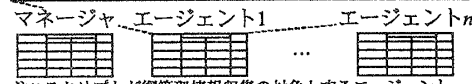
ノード	CPU利用率(%)
マネージャ	46
エージェント1	78
エージェント2	53
⋮	
エージェントn	52
平均	50

許容偏在度(%)	10
許容帯域(bps)	1.0*10 ⁵
消費帯域(bps)	3.2*10 ³

(b) スクリプト管理テーブル

エージェント1

スクリプト	帯域(bps)		ターゲットエージェント(注1)	優先度
	マネージャ	エージェント		
script A	864	346	エージェント1	5
script B	346	0(注2)	エージェント1	2
script C	172	14	エージェント2	7
⋮				



注1) スクリプトが網管理情報収集の対象とするエージェント
注2) 閾値違反等の場合のみマネージャへ通知するスクリプトの場合は0と規定

図 3: ノード管理テーブルとスクリプト管理テーブルの概略。

3.3 適用例

図 3 に示す例の場合, マネージャは図 2 S5 においてノード管理テーブルに基づき CPU 利用率が 40% = (50%(全ノード CPU 利用率の平均) + 10%(許容偏在度)) を超えるノード (エージェント 1) を決定する。次いで, 図 2 S6 においてスクリプト管理テーブルに基づき切り替えにともなう必要帯域の減少が最大の “Script C” (この場合, 172 + 14 (bps) から 172 (bps) へ減少) が切り替え対象に, CPU 利用率最小の (46%) のマネージャが切り替え先に決定される。

4 おわりに

本稿では分散網管理環境において動的にノードの処理負荷を分散制御する方式を提案した。本方式では, マネージャやエージェントの処理負荷ならびにスクリプト実行時に必要な帯域に応じて, 網全体で動的に処理負荷の均衡を図る。提案方式の実装を通じた実環境評価が今後の課題である。最後に日頃御指導頂く (株) KDD 研究所村谷拓郎所長ならびに鈴木健二副所長に感謝します。

参考文献

- [1] Y. Yemini, G. Goldszmidt, and S. Yemini. Network Management by Delegation. In *Proc. of IFIP ISINM '91*, pp.95-107, 1991.
- [2] J. Grégoire. Models and Support Mechanisms for Distributed Management. In *Proc. of IFIP ISINM '95*, pp.17-28, 1995.
- [3] A. Vassila, G. Pavlou, and G. Knight. Active Objects in TMN. In *Proc. of IFIP/IEEE IM '97*, pp.139-150, 1997.
- [4] T. Magedanz and T. Eckardt. Mobile Software Agents: A New Paradigm for Telecommunication Management. In *Proc. of IFIP/IEEE NOMS '96*, pp.360-369, 1996.
- [5] A. Bieszczad, B. Pagurek, and T. White. Mobile Agents for Network Management. *IEEE Comm. Surveys*, Vol.1, No.1, 1998.