

マルチエージェントによる
適応型ネットワーク管理技術の研究

Research on Adaptive Network Management
Technologies using Multi-agent Systems

2008年2月

早稲田大学大学院情報生産システム研究科

情報生産システム工学専攻 知能化ネットワーク研究

寺内 敦

目次

第1章	序論	1
1.1	研究の背景と動機	1
1.1.1	本論文の扱う課題	1
1.1.2	課題に対する本論文のアプローチ	2
1.2	要求条件と課題	2
1.3	本論文の構成と、各章の概要	3
1.3.1	第2章の概要	5
1.3.2	第3章の概要	5
1.3.3	第4章の概要	6
1.3.4	第5章の概要	6
第2章	マルチエージェントを利用した適応型ネットワーク管理の構成技術	7
2.1	広域ネットワーク管理のためのエージェント組織化支援技術	7
2.1.1	マルチエージェントシステムによるネットワーク管理技術	7
2.1.2	マルチエージェントシステムにおけるエージェント組織化技術	12
2.1.3	広域ネットワーク管理におけるエージェント組織化	13
2.1.4	広域ネットワーク管理のためのエージェント組織化に対する従来技術と課題	15
2.2	エージェント間協調動作管理技術	16
2.2.1	マルチエージェントシステムにおけるエージェント間協調	16
2.2.2	広域ネットワーク管理におけるエージェント間協調	19
2.2.3	エージェント間協調動作管理技術に関する従来技術と課題	21
第3章	広域ネットワーク管理のためのエージェント組織化支援技術	24
3.1	本論文における位置づけ	24
3.2	研究の背景	25
3.3	広域ネットワーク管理を行うマルチエージェントシステムの実例	27
3.3.1	AS間経路診断システム ENCORE	27
3.3.2	マルチエージェントによる広域動画配信アーキテクチャ	28
3.4	問題の所在とアプローチ	34
3.4.1	広域ネットワーク管理用マルチエージェントシステムにおけるエージェント組織化の問題	34
3.4.2	アプローチ	37
3.5	提案方式	39

3.5.1	エージェント組織化支援システム ARTISTE	39
3.5.2	ARTISTE による組織化支援	41
3.5.3	マルチエージェントシステムによる探索条件の送信	43
3.5.4	エージェントの探索手順.....	45
3.6	提案方式の評価.....	48
3.6.1	実験環境.....	48
3.6.2	実験結果.....	51
3.6.3	エージェントの配置方法.....	55
3.7	本章の総括	61
第 4 章	エージェント間協調動作管理技術	62
4.1	本章の位置づけ	62
4.2	研究の背景	63
4.3	問題の所在とアプローチ.....	64
4.3.1	問題の所在	64
4.3.2	アプローチ	66
4.4	提案方式.....	69
4.4.1	グループによる情報共有方式	70
4.4.2	個々のエージェントの信頼度に基づく情報共有方式	72
4.5	提案方式の評価.....	78
4.5.1	実装	78
4.5.2	動作例	80
4.5.3	実験結果.....	83
4.6	本章の総括	89
第 5 章	結論	91
5.1	広域ネットワーク管理のためのエージェント組織化支援技術	91
5.2	エージェント間協調動作管理技術	92
5.3	今後の展望	93
	謝辞.....	95
	図目次.....	96
	表目次.....	98
	参考文献.....	99

第1章 序論

【関連論文【A1】，【A2】，【A5】，【A7】，【A8】，【B1】 - 【B4】，【C1】】

1.1 研究の背景と動機

インターネットの広がりや留まることなく、ますます巨大化および複雑化が進んでいる。加えて、広帯域のネットワークおよび高性能の計算機の普及に伴い、高品質の映像配信などインターネット上でのサービスが高度化、多様化している。このようなサービスが多様化する状況においては、それぞれのサービス毎にネットワークに対する要求条件が異なっており、それらの要求条件を満たすようにインターネットを現在のように人手で適切に管理制御することはますます難しくなっている。

この問題を解決するため、インターネット管理の自動化、高度化の技術に対する期待は非常に大きい。マクロな視点で見た場合、インターネットは ISP (Internet Service Provider) など独立した主体が管理運用する AS (Autonomous System) が相互接続した巨大な分散システムと捉えられる。個々の管理主体は通常自分が管理する AS 内しか観測制御できず他の AS の状態を知ることができないため、複数の AS を跨る広域ネットワークを管理・制御するには AS(管理主体)同士による協調作業が必要である。このように、広域ネットワークの管理は本質的に複数主体間の分散協調作業であり、この特徴が作業の自動化を困難にしている。

1.1.1 本論文の扱う課題

そこで、人工知能の分野からのアプローチとしてインターネットと同様の分散構造を持つマルチエージェントシステムへのネットワーク管理への応用がこの問題の解決に有効であると考えられ、実際に多くの提案がなされている。一般にマルチエージェントシステムを使った作業ではエージェント間の協調が必要となるが、この協調を効果的に実行するためには実行する作業の内容やエージェントの状態などに基づいて選択された適切なエージェント群が協調することが重要である。このようにエージェントの機能や状態などを考慮して協調すべきエージェント群を選択する作業をエージェントの組織化と呼ぶ。ただし、集中管理機構を持たないインターネット上のマルチエージェントシステムでは組織化に必要な広範囲のネットワーク情報の観測が容易ではないため、適切な組織化に困難が伴う。

1.1.2 課題に対する本論文のアプローチ

このような背景の元で、本論文は複数の AS を跨る広域ネットワークへマルチエージェントシステムを適用することでネットワーク状態の変動に適応したネットワーク管理を実現する適応型ネットワーク管理技術を、特に、マルチエージェントシステムの動作の適応性、効率性を向上させるためのエージェント組織化支援技術および実際の広域ネットワーク管理作業にマルチエージェントシステムを安全に適用するためのエージェント間協調動作管理技術に着目して検討し、明らかにしている。具体的には、広域ネットワークを管理するマルチエージェントシステムを対象にして、ネットワーク上の位置など特定の条件を満たすエージェント群を動的に探索・同定することによりエージェントの組織化を支援するエージェント組織化支援技術を明らかにする(課題 1)、異なる AS 上に配置されたエージェントの協調動作に対して、各配置先 AS の持つポリシーを反映することでマルチエージェントシステムの安全な適用を実現するためのエージェント間協調動作管理技術を明らかにする(課題 2)。

本論文内で検討に用いているソフトウェアは、プロトタイプを構築して実際のネットワーク上で動作させて評価を行うことで、提案技術の高い信頼性を示している。これにより、提案技術が今後のインターネット管理における中核技術として利用可能な実用的技術であることを明らかにしている。

1.2 要求条件と課題

本節では、複数の AS を跨る広域ネットワークへマルチエージェントシステムを適用し適切に管理するための要求条件と、その課題について説明を行う。

- ① 一般にマルチエージェントシステムを使った作業ではエージェント間の協調が必要であり、この協調作業を効果的に実行するためには、実行する作業の内容やエージェントの状態などに基づいて協調相手のエージェント群を選択する組織化を適切に行う必要がある。特に広域ネットワーク管理へマルチエージェントシステムを適用する際はエージェントの組織化のために AS を跨る広範囲のネットワーク情報が必要であるが、集中管理機構を持たない分散システムであるインターネットに関して、そのような広範囲の情報を観測収集することは容易なことではない。そのため、このような広範囲のネットワーク情報の収集を行いながらエージェントの適切な組織化を行うことが広域ネットワーク管理へのマルチエージェントシステムの適用において不可欠である。

- ② マルチエージェントシステムによって広域ネットワークの管理を行う場合、複数のエージェントが各配置先で観測したネットワーク情報を交換することがある。エージェントが配置された AS(を運営する主体)間の関係はネットワークの正常な運用のために協力する関係にある一方、異なる ISP(Internet Service Provider)など利害が競合する関係にもある。そのため、実際には、たとえインターネットの管理という共通の利益のためでも、AS 間のネットワーク情報の交換にはそれぞれの AS のポリシーに基づき制限が加わることが通常である。よって、マルチエージェントシステムを実際のネットワーク管理に適用するためには、前述のような情報交換の制限などエージェントの配置先 AS のポリシーを反映してエージェント間での協調動作を適切に管理する方法を確立することが不可欠となる。

上記のような要求条件を満足するためには、以下に示すように、対応する 2 つの課題を検討する必要がある。

- **【課題 1】** 広域ネットワークを管理するマルチエージェントシステムに対して、ネットワーク上の位置など特定の条件を満たすエージェント群を動的に探索・同定することによりエージェントの組織化を支援するエージェント組織化支援技術の明確化
- **【課題 2】** 異なる AS 上に配置されたエージェントの協調動作に対して、各配置先 AS の持つポリシーを反映することでマルチエージェントシステムの安全な適用を実現するためのエージェント間協調動作管理技術の明確化

1.3 本論文の構成と、各章の概要

本論文は、マルチエージェントシステムにより複数の AS に跨る広域ネットワークの管理をネットワーク状態の変化に適応して行う適応型ネットワーク管理技術における構成技術に関する研究成果をまとめたものである。

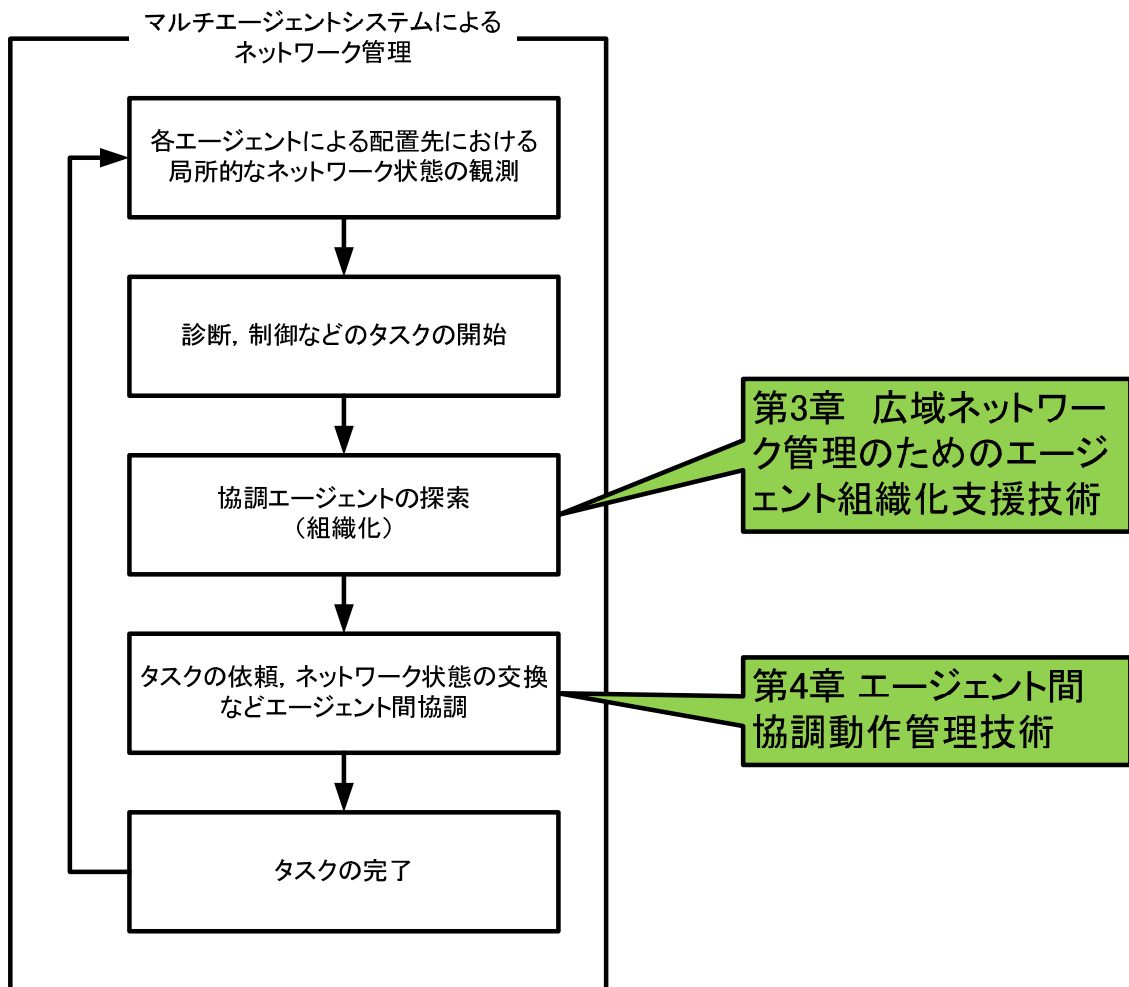


図 1-1 各章の関わり

一般的なマルチエージェントシステムを用いたネットワーク管理の流れと本論文の各章との関わりを図 1-1 に示す。本論文で明らかにするマルチエージェントによる適応型ネットワーク管理技術の構成要素は、大きくエージェントによるネットワーク状態の観測部分と、観測した状態に基づいたエージェント間の協調によるタスクの実行部分に分けられる。さらにタスクの実行部分は大きく適切な協調エージェントの探索部分と実際の協調作業部分に分けることができる。本論文では、これらの動作を広域ネットワーク上で行う際の問題を明らかにするとともに適切な解決技術の提案をそれぞれ第 3 章、第 4 章で行う。

本論文では、このマルチエージェントシステムによる広域ネットワーク管理技術における構成要素の実現技術を明らかにすることを目的としており、さらに提案技術を実際のネットワーク管理業務への適応する観点からの問題にも言及している。本研究の全

体構成を以下にまとめる。

1.3.1 第 2 章の概要

ここでは、近年の動向を踏まえて、マルチエージェントシステムを利用したネットワーク管理の構成技術についてこれまでの到達点を示している。構成技術とは、エージェント組織化支援技術およびエージェント間協調動作管理技術である。本章では、従来の構成技術が本論文で対象としているようなネットワーク管理に対して適用可能か否かの観点から整理を行うと共に、実システムへの適用性の観点から第 1 章で設定した課題の妥当性を示す。

1.3.2 第 3 章の概要

ここでは、課題 1 に関し、複数の AS を跨る広域ネットワーク管理のためのマルチエージェントシステムを対象に、ネットワーク上の位置や機能など特定の条件を満たすエージェント群を動的に探索・同定することにより組織化を支援するエージェント組織化支援技術を明らかにする。

具体的には、組織化のために必要な広域ネットワーク情報を複数のエージェントを用いて分散観測することで単一の AS における観測では獲得の難しい広域ネットワークのトポロジ情報などを利用できるエージェント組織化支援技術を提案した。さらに、広域ネットワーク管理を行う複数のマルチエージェントシステムの動作を分析した結果、これらの組織化作業や利用する情報の共通性があることを明らかにした。これらの知見を元にして複数のマルチエージェントシステムに組織化支援機能を提供することができる共通プラットフォームとしてエージェント組織化システム ARTISTE (Agents' Roles and Topological Information management System for agent TEamwork) を実装した。

実際のインターネットと同等の情報が流通している実験環境上で実装した ARTISTE の評価実験を行った結果、ARTISTE はネットワーク上の広域ネットワーク管理用マルチエージェントシステムのための組織化を正しく行え、結果として組織化を静的に行う場合と比べてマルチエージェントシステムの運用コストを下げることができると、実際のインターネット環境においても実用的な探索範囲内では 3 秒以内で探索が完了することを確認できた。さらに、ARTISTE を含めたシステムアーキテクチャを採用することで、広域ネットワーク管理用マルチエージェントシステムの構築のための工数が従来と比べて約 8.3% 削減できることを確認した。

1.3.3 第4章の概要

ここでは、課題2に関し、広域ネットワーク管理を行うマルチエージェントシステム内で行われる協調動作を各エージェントの配置先のポリシーに基づいて管理するためのエージェント間協調動作管理技術を明らかにする。

まず、実際のネットワーク管理作業を調査した結果を元にネットワーク管理におけるマルチエージェントシステムの協調動作に関する要求条件を明らかにした。この要求条件を満たすため、エージェント間の協調動作を管理するためにエージェントのグループ管理機構およびエージェントの信頼度に基づく情報共有機構を提案した。前者は複数のエージェントをグループとして管理することにより系列会社の運用するエージェントのみと協調するなどの動作管理を実現し、後者は各エージェント(の運営者)が他のエージェントに対する信頼度をデータとして設定することにより信頼度の違いによって協調の有無やその内容を動的に制御することを可能とする。これらの提案する技術を組み込んだネットワーク管理用マルチエージェントシステムを試作して実験環境で動作させて評価を行い、本技術の有効性を明らかにした。

1.3.4 第5章の概要

本論文では、マルチエージェントシステムを用いて複数のASを跨る広域ネットワークをネットワーク状態に適応して管理制御する際の、マルチエージェントシステムの動作の適応性、効率性を向上させるためのエージェント組織化支援技術および広域ネットワーク管理作業にマルチエージェントシステムを安全に適用するためのエージェント間協調動作管理技術を提案し、これらの提案技術の実装および評価を通じて、それらの実現性を検証した結果について総括する。さらに、本研究が将来のネットワーク管理技術の実用化に果たす役割について述べる。

第2章 マルチエージェントを利用した 適応型ネットワーク管理の構成技術

【関連論文 【A1】，【A2】，【A5】，【A7】，【A8】，【B1】-【B3】，【C1】，【C4】
- 【C7】】

ここでは、第1章であげたマルチエージェントシステムを利用した適応型ネットワーク管理技術の構成要素技術に関して、その概要および目的を示した後、各技術分野でとられている従来のアプローチを説明する。それらが本研究で想定するような実際の広域ネットワーク管理作業においてどのような効果を発揮するかについて解説を行う。さらに、本論文で取り扱う課題に適用する上での従来技術の到達点を説明することで、本論文で設定した課題の妥当性を明らかにする。

2.1 広域ネットワーク管理のためのエージェント 組織化支援技術

広域ネットワーク管理のためのエージェント組織化技術とは、広域ネットワーク管理を行うことを目的に動作するマルチエージェントシステムにおいて、エージェント間での協調が必要となった場合にタスクの内容やエージェントの状態などを考慮して協調作業にもっとも適したエージェントの集合を選択する技術である。本章では本技術の重要性および課題を明確に説明するため、はじめに本技術と関連の深い次の2つの技術、①マルチエージェントシステムによるネットワーク管理技術、および②マルチエージェントシステムにおける組織化(支援)技術について述べる。

2.1.1 マルチエージェントシステムによるネットワーク管理 技術

インターネットをマクロな視点で捉えると大学やISP (Internet Service Provider)な

どの管理主体が運営するネットワークである AS (Autonomous System)が相互接続されて実現されている。AS の内部では複数の LAN や別の AS などが相互接続されているため、結果として、インターネットは大量のネットワークが相互接続された巨大なネットワークとして機能している。そして AS 間で経路情報やトラフィックを交換することで全世界に渡る広域の通信を実現している。

インターネットを有効に利用するためには、ネットワークが常に安定した接続を提供し、かつ、十分に高速なデータ転送ができることが望ましいが、多数の機器やユーザが接続されたインターネットでは常にそのような状態が成り立つという保証はない。そのため、ネットワークの運営者ではそれぞれが管理するネットワークにおいて、上記のような望ましい状態に保つべく留意する必要がある。本論文では、このようにネットワークの状態を(運営者あるいはユーザの観点から)望ましい状態に保つ作業のことを「ネットワーク管理」と呼ぶことにする [Calvert Griffioen, 2006].

一般的なネットワーク管理は次の手順によって行われる。

- ① 管理者がツールなどを用いて自分の管理するネットワークの状態を収集する
- ② 収集した情報を分析し、必要に応じてルータなどの機器を操作しながらネットワークを望む状態になるように制御する

ネットワーク管理の流れの概略を図 2-1 に示す。

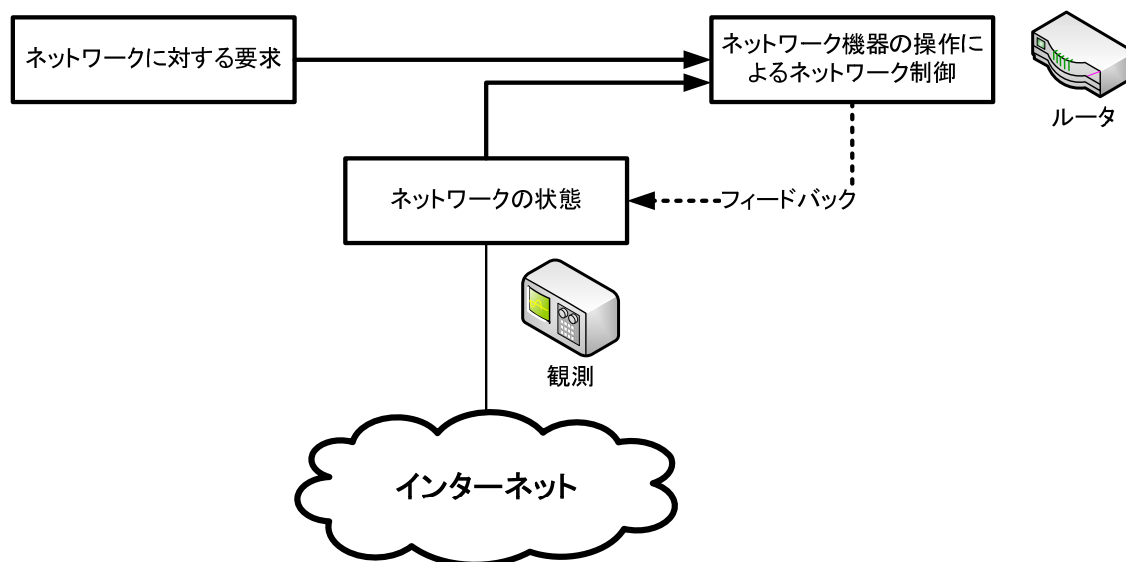


図 2-1 ネットワーク管理の流れ

ところが、LAN などの AS 内部のネットワークを対象にしたネットワーク管理と複

数のASを跨る広域ネットワークを対象にしたものとは全体の流れは同一であっても本質的に異なる点がある。その点を説明するためにまず、この2種類のネットワークの特長を表 2-1 に整理する。

表 2-1 ネットワークの規模による違い

LAN などの AS 内部のネットワーク	<ul style="list-style-type: none"> ● 管理主体は単独の組織 ● 管理主体はネットワーク全体の情報を比較的容易に取得可能 ● 配下のルータなどネットワーク機器の制御も制約なく実行可能
AS を跨る広域ネットワーク	<ul style="list-style-type: none"> ● AS 毎に異なる複数の管理主体が存在 ● それぞれの管理主体は自己の管理するネットワークの情報は比較的容易に取得可能である反面、他の主体が管理するネットワークの状態を知ることは困難 ● ルータなどネットワーク機器の制御についても同様に自己の管理する機器であれば制約なく実行可能だが、他の主体が管理するネットワーク機器は制御不能

表 2.1 が示すように、前者の AS 内部のネットワークに関しては単独の管理主体がネットワーク状態の観測から制御まで特に制約なく実行できるため、ネットワーク管理作業を主体が運営する集中型システムによって自動化、効率化することができる¹。これに対して、AS を跨る広域ネットワークでは複数の管理主体がネットワーク管理に関わ

¹ ただし、ネットワークの状態の観測はネットワークの複数地点において分散観測することがある。

っており、かつ、それぞれの主体が観測、制御できる範囲は自己の管理するネットワーク配下に限られる。例えば、インターネットで通信を行うためには経路情報を適切にネットワーク全体に伝搬させる必要があるが、この経路情報の伝搬は経路情報を受信した AS が、さらに別の AS に経路情報を転送するという AS 間の協調によって実現されている。各 AS における転送作業はそれぞれの AS のポリシーに従って行われており、他の AS がそれを観測し制御することはできない。そのため、経路情報の伝搬障害が起きたとしても特定の AS からの観測のみで原因を知ることはできず経路を伝搬した AS が協調して原因を調査する必要がある。別の例として、高品質の動画など大容量のコンテンツをインターネットで転送する場合、転送経路上の AS の 1 つでも十分な転送速度が確保できなければ転送の遅延などの問題が発生する。この場合も各 AS では自分以外の AS のトラフィックの状態を知ることはできないことから、原因となる AS の発見など問題解決には AS の協調が不可欠である。

このことを図 2-2 に示す状況を例に用いて説明する。

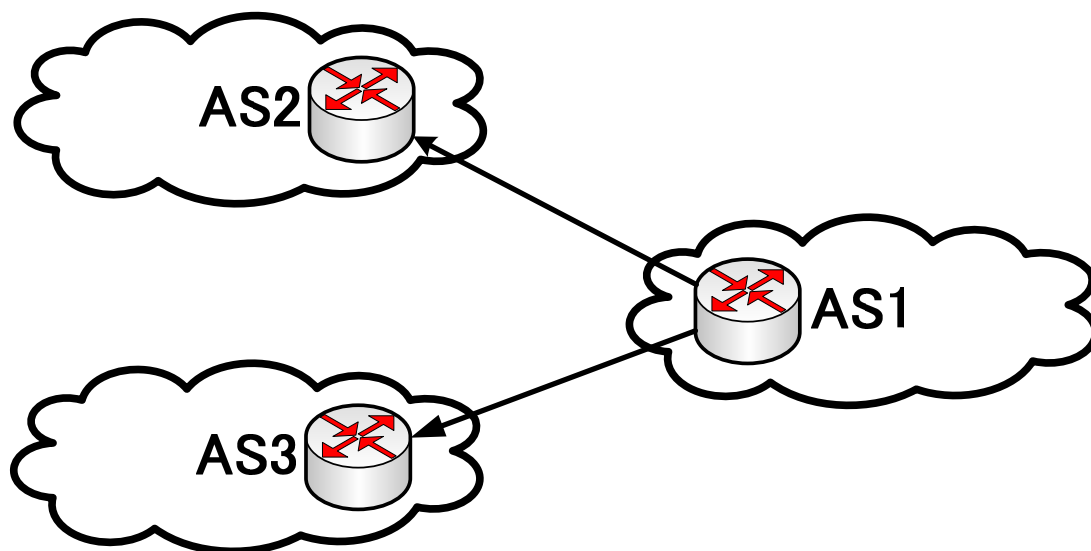


図 2-2 ネットワーク動作における AS 間の関連

図中の AS1 は AS2 と AS3 を経由してインターネット全体との通信を行っているものとする。このとき、AS1 は自 AS 内部からのトラフィックの増加に対処するため、より回線容量の大きい AS3 経由でのトラフィックを増加させるようにネットワークの構成を変更したとする。しかし、同時に AS3 は同じく自己の方針に基づき AS1 からのトラフィックを回線速度の遅い回線経由でしか外部に流さないようにネットワークの設定を変更したとする。この場合、元々の AS1 の要求であるトラフィック増加への対処は AS1 と AS3 の間という局所的な範囲では達成されるものの、AS3 から外部に出ていく回線がボトルネックとなるためインターネット全体で達成することは不可能になる。つまり、AS1 は自己の要求を満たすために自己のネットワークを制御することで対処

しようとするがネットワーク全体の動作にまで影響を与えることはできず、他の AS(この場合は少なくとも AS3)との協調が不可欠である。このように広域ネットワーク全体を管理制御するためには、ネットワークに含まれる AS 間での協調作業が不可欠であることが分かる。

このような AS 間の分散協調作業としてモデル化される広域ネットワークの管理を単一の地点で稼働するシステムによる集中型のアプローチによって自動化することは困難である。なぜならば、ある AS (ネットワーク) にインストールされたシステムは配置先 AS の状態を観測し、ルータなどを制御することは可能かもしれないが、他の AS (ネットワーク) に関する観測や制御を行うことは不可能である。前述の通り、広域ネットワークの管理ではネットワークに含まれる AS 全体の協調が不可欠であるため、結果として、単一システムの集中型アプローチでは広域ネットワーク全体の管理を行うことはできないことが分かる。

そこで、このように集中型システムでは十分に対処できない広域ネットワーク管理の高度化に対する人工知能の分野からのアプローチとしてインターネットと同様の分散構造を持つマルチエージェントシステム [E.H.Durfee, V.R.Lesser, D.D.Corkill, 1987] のネットワーク管理への応用が期待されている。マルチエージェントシステムは独立に動作可能な複数のエージェントが必要に応じて他のエージェントと協調しながらタスクを実行する分散協調型のシステムであり、従来の集中型システムでは対処できない問題を解決するためのアプローチとして電子商取引や Video-on-demand などさまざまな分野での応用が進んでいる。マルチエージェントシステムをネットワーク管理に応用する試みについてもすでに多くの提案がなされている [O.Akashi, A.Terauchi, K.Fukuda, T.Hirotsu, M.Maruyama, T.Sugawara, 2005], [寺内, ほか, 2004], [S.Willmott B.Faltings, 2000], [D.A.Tran, K.A.Hua, S.Sheu, 2003], [M.Sasabe, N.Wakamiya, M.Murata, H.Miyahara, 2003], [M.M.Hefeeda, B.K.Bhargava, D.K.Y.Yau, 2004].

これらの論文から明らかのように、一般的に、マルチエージェントシステムをネットワーク管理に応用する場合、本章の冒頭に記載した手順、すなわち、①ネットワークに関する情報を収集、分析、②収集した情報に基づきルータなどのネットワーク機器を制御、という流れをエージェントによって実行することになる。一方、マルチエージェントシステムのような分散協調型のシステムは中央集権型のシステムと異なりシステム全体を管理できないため、システム全体を効率よく動作させて目的を達成させるためにはタスクの効率的な分割とそれらを適当なエージェントへと割り当てる作業が重要になる。本研究ではそのようなマルチエージェントシステムの効率的な動作を実現するためのエージェントの組織化と呼ばれるプロセスに着目する。次章では、このエージェントの組織化技術について解説する。

2.1.2 マルチエージェントシステムにおけるエージェント組織化技術

一般にマルチエージェントシステムを使った作業は大きく以下のように分割できる。

- ① マルチエージェントシステムとして実行する作業（以下、タスク）を複数の副作業（以下、サブタスク）に分割する。
- ② 分割した各サブタスクをマルチエージェントシステム内のエージェントに対して割り当てる。
- ③ エージェントは個々にサブタスクを実行すると共に、必要に応じてエージェント間で協調することでマルチエージェントシステムとしてのタスクを実行する。

このとき、③の協調作業を効果的に実行するためには、②のサブタスクの割り当てにおいて、マルチエージェントシステムとして実行するタスクの内容やエージェントの状態などに基づいて選択された適切なエージェント群に対してサブタスクを割り当て、それらエージェントが協調作業を行うことが重要である [S.Willmott B.Faltings, 2000]。この選択が適切になされていない場合、例えば、協調相手のエージェントが依頼した作業を実行するための機能を持たない、問題解決に必要な情報を観測できない、高負荷などのため依頼された作業を実行できない場合には協調がうまくいかず結果として作業の実行が失敗する可能性がある。このようにマルチエージェントシステムにおいて要求されるエージェントの機能や配置を考慮しながら協調相手とすべきエージェント群を選択する作業をエージェントの組織化と呼ぶ。組織化はエージェント間の協調を行うすべてのマルチエージェントシステムが行うべき作業であり、それ故に適切な組織化を実現することはすべてのマルチエージェントシステムに共通した要求である。

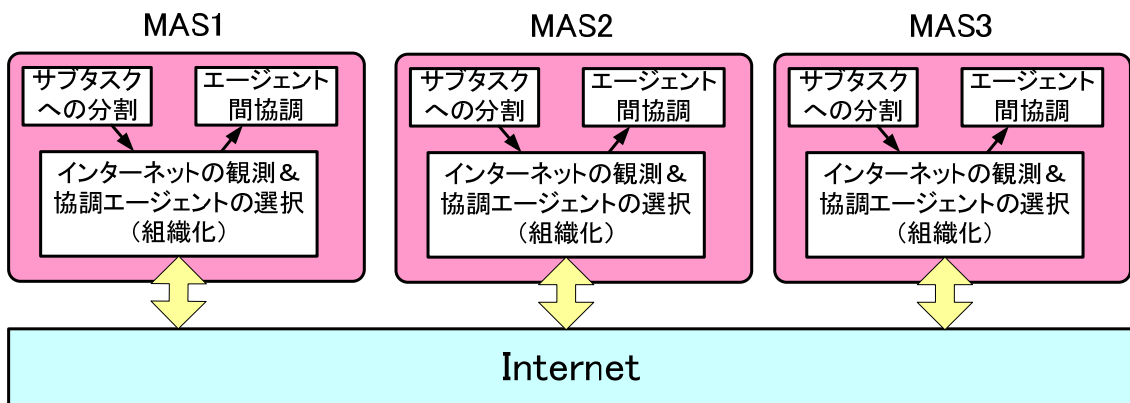


図 2-3 マルチエージェントシステムの動作

特に、インターネット上で動作するマルチエージェントシステムにおいてはマルチエージェントシステムの動作環境であるインターネットの状態が頻繁に変わることなどから環境の変化に適応した動的な組織化を行うための技術がマルチエージェントシステムの安定動作のために求められている。次章では、広域ネットワーク管理を行うマルチエージェントシステムにおける組織化技術の重要性を説明するとともに、当該技術がマルチエージェントシステムによる広域ネットワーク管理を実現する上で不可欠な技術であることを説明する。

2.1.3 広域ネットワーク管理におけるエージェント組織化

本章では広域ネットワーク管理にマルチエージェントシステムを適用する際のエージェント組織化手続きが不可欠であることを述べる。

通常、マルチエージェントシステムにおける組織化は可能性のある作業を想定した上で事前に完了させておくか、マルチエージェントシステムによる作業の実行中にエージェントによって動的に行うことが想定される。しかし、事前に組織化を行う場合、各エージェントは自分以外のエージェント情報を管理しなければならない。

筆者らは実際に広域ネットワーク管理用のマルチエージェントシステムを作成、運用しているが、このようなエージェント情報の維持管理やそれに伴うエージェントの組織化のコストの高さが大きな問題となっていた。筆者らが実際に運用しているマルチエージェントシステムのうち、[O.Akashi, A.Terauchi, K.Fukuda, T.Hirotsu, M.Maruyama, T.Sugawara, 2005]で提案されている AS 間経路診断システム ENCORE を例にしてこの問題について具体的に説明する。

図 2-4 に ENCORE における協調エージェントの管理方法を示す。図の左に示すように、現行の ENCORE では個々のエージェント毎に協調する可能性のあるエージェントのリストを設定ファイルとしてあらかじめ保持させておく必要がある。この設定として指定されるエージェントは ENCORE が行うタスクの内容や設定時のネットワークの状態などを考慮して ENCORE エージェントのオペレータによって決定される。ところが、図の右に示すようにインターネットの状態や他エージェントの状態が設定時より変化した場合には、その変化に合わせて設定ファイルを変更する必要がある。

例えば、図左に示すように、AS1 上のエージェント 1 において、タスク実行のために他エージェントとの協調が必要になったとする。このとき、エージェント 1 は協調相手として「自 AS(=AS1)からホップ数 3 以内で接続された AS 上のエージェント」を選択しており、図左の状態ではそれらは AS2,3,4 上のエージェント 2,3,4 であったとする。

その後、ネットワークの状態に変化がおき、AS間のホップ数が増えたり、あるいはエージェントのアドレス自体が変わってしまったとすると、エージェント1では図右に示すように設定ファイルを修正して、条件を満たさないエージェントをリストから削除したり、アドレスを修正しなければならない。

この設定作業量は系のエージェントの数が少ない場合には大きくないが、エージェントの数が増えるに従って増加し、かつ、変化が起きる度に変更を行わなければならないため、運用上大きな問題であることが筆者らの運用経験から明らかになった。

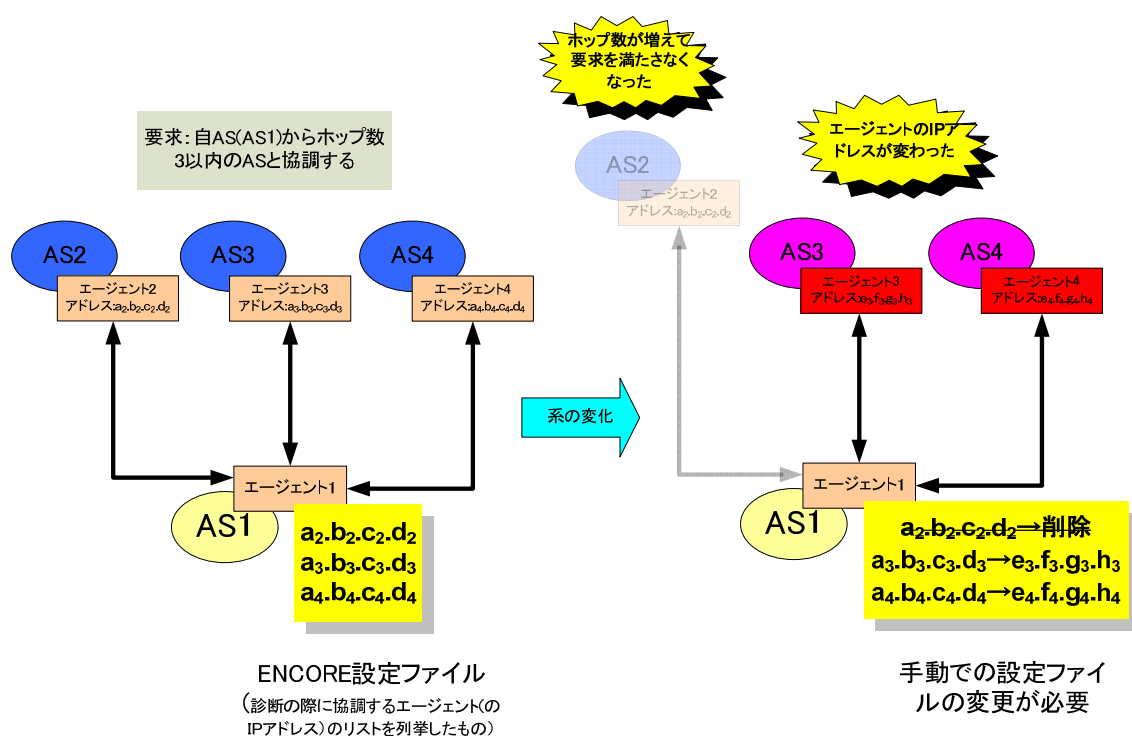


図 2-4 ENCORE における協調エージェントの管理

ここでは例として ENCORE における事例を示したが、この問題は ENCORE に限らずエージェントを静的に管理するすべてのマルチエージェントシステムにおいて発生する普遍的な問題である。さらに、広域ネットワーク管理にマルチエージェントシステムを適用する場合には、次のような理由から上に示したような手動での組織化はさらに難しくなっている。

- 協調エージェントの選択にネットワークの状態(トポロジなど)が深く関係するため、ネットワーク状態の変化の都度、新たな組織化が必要となる
- 広域ネットワーク管理の中にはさまざまな種類の作業が含まれており、それぞれの

作業に適した組織化を行わなければならない

以上より、組織化の手続きを自動化あるいは半自動化によって効率的に行う技術を開発することは、マルチエージェントシステムをネットワーク管理などに実際に適用する上で生じる障害を克服するために非常に重要であると言える。

2.1.4 広域ネットワーク管理のためのエージェント組織化 に対する従来技術と課題

マルチエージェントシステムにおける組織化の重要性は [Wooldridge, 2002]などにも見られるように、マルチエージェントシステムの研究の初期より指摘されてきた問題である。このようなエージェントの組織化を支援するシステムの代表的なものは [Tambe, Pynadath, Chauvat, 2000]で提案されている。これは、マルチエージェントシステムに含まれるエージェントの持つ能力や機能といった情報をデータベースとして管理し、エージェントが協調する際に各エージェントが協調相手に求める能力や機能をキーとしてこのデータベースを参照することにより、協調相手として適切なエージェントの情報を提供するというものであり、マルチエージェントシステムのエージェントに対するイエローサービスのようなものであった。ただし、[Tambe, Pynadath, Chauvat, 2000]の方法では、対象とするマルチエージェントシステムがインターネットのような動的な環境で動作することを想定しておらず、データベースに保存される情報も個々のエージェントの静的な情報のみに限られていた。

他方、ネットワーク管理を行うマルチエージェントシステムの分野では [今野, 吉村, 羽鳥, 岩谷, 阿部, 木下, 2005] [内山, 梅津, 安本, 東野, 2006]のように LAN や社内ネットワークを対象として、エージェントの協調によるネットワーク管理を行う手法がいくつか提案されている。ただし、これらの研究が想定している動作環境である LAN や社内ネットワークは基本的に単独の管理主体によって運営されているネットワークであるため、当該管理主体からは当該ネットワークの状態やトポロジといった詳細情報も容易に取得できる。そのため、このようなネットワークを対象にしたネットワーク管理用マルチエージェントシステムでは、マルチエージェントシステムは主に管理のために必要なネットワーク情報を分散収集するために用いられており実際の作業は集中的に行われている。また、単独の主体によってマルチエージェントシステムが運営されているため、マルチエージェントシステム中のエージェントに関する位置や状態などの情報を制限なく取得することが可能であり、エージェントの組織化を行うことも比較的容易である。

これら単独主体によるネットワークを対象にしたマルチエージェントシステムに対して、広域ネットワークで動作するマルチエージェントシステムとしては、動的なルーティング[S.Willmott B.Faltings, 2000][Willmott Faltings, 1999]、動画配信のためのトラフィック制御 [M.Sasabe, N.Wakamiya, M.Murata, H.Miyahara, 2003][D.A.Tran, K.A.Hua, S.Sheu, 2003]などの方法が提案されている。これらの方法では広域ネットワーク上でのエージェント協調を実現するために各エージェントによりノード(エージェント)間の距離やトラフィック情報などさまざまなネットワーク情報の取得を行っているが、基本的にはそれぞれのアプリケーションに必要とされる情報のみを取得しており、広域ネットワーク管理に必要な情報としては十分とはいえず、また、それぞれのシステムは特定のアプリケーションとして設計されているため、多様な作業内容が存在するネットワーク管理に対して適用可能とはなっていない。

このように、従来の手法では広域ネットワーク管理を行うためのマルチエージェントシステムの組織化支援技術を構成するには十分ではなく、これらの課題を解決した新たな手法の提案が広域ネットワーク管理におけるエージェントの組織課支援技術を構成する上で必須となる。

2.2 エージェント間協調動作管理技術

エージェント間協調動作管理技術とは、マルチエージェントシステムにおいてエージェント間での協調動作を行う場合に、各エージェントの運営者の持つ意図やポリシーをエージェントの動作に反映させることによってエージェント間の協調動作を運営者の意図通りに管理するための技術である。本章では本技術の重要性および課題を明確にするため、はじめにマルチエージェントシステムにおけるエージェント間協調技術の概略を述べた後、広域ネットワーク管理にマルチエージェントシステムを適用した際に生じうるエージェント協調に関わる問題を示す。最後に、本技術に関する従来手法とその限界点を示すことで本技術の開発の重要性を明らかにする。

2.2.1 マルチエージェントシステムにおけるエージェント間協調

マルチエージェントシステムは、複数のエージェントがそれぞれ自分に与えられたタスクを実行し、必要に応じて他エージェントとの間で協調を行うことにより、システム全体としての目標の達成を実現するシステムである。具体的にエージェント間でどのよ

うな協調作業が行われるか、協調の頻度はどのくらいか、などは個々のシステムが対象とする問題、システムにおける問題解決の手順などによって異なっているが、自分では達成できない、あるいは他のエージェントの方が効率よく実行できるタスクを適当な他のエージェントに対して依頼し、実行結果を取得するというタスクの相互依頼やタスクの実行のために必要な情報の交換などが一般的に行われるエージェント間の協調作業と考えられる。

ただし、このような協調作業はマルチエージェントシステムの運用形態によってその意味が変わることがある。この違いを表 2-2 に示す。

表 2-2 マルチエージェントシステムの運用形態による協調の違い

マルチエージェントシステムの運用形態	
システム全体(=全エージェント)が単一の主体によって運用されている場合	<ul style="list-style-type: none"> ● システムとしての目標が単一主体の意図に基づくため、明確 ● 全エージェントが同一の目標達成に向けて協力可能 ● エージェント間での利害の競合はほとんど起きない
システム中の各エージェントが別の主体によって運用されている場合	<ul style="list-style-type: none"> ● 複数の主体が関わるため、システムとしての共通の目標は存在しないか、不明確 ● 個々のエージェントはそれぞれの運用主体の意図に基づいて動作 ● 運用主体同士の関係によってはエージェント間の動作に競合が生じる

表に示した 2 つの場合の違いは、エージェント間の動作に何らかの競合が生じうるか否かということである。表中の前者の場合では、システムとしての目標が明確であり、システム中の各エージェントはその目標に向けて最大限の貢献をするように動作する

(ように設計されている)ことが想定される。よって、各エージェントは無条件に他のエージェントと協調的に振る舞うことができる。ただし、SETI プロジェクト [SETI@home]に代表される分散協調型の科学実験などエージェントの運営主体が複数であっても、システムとしての単一の目標達成にむけてすべての主体が無条件に協力することが明確である場合はこちらの場合に含まれる。

ところが、反対に、後者のようにエージェントが別々の主体によって運用されている場合では、システムとしての目標が明確に決まるわけではない。例えば、P2P 分散技術を用いたファイル共有では、個々のエージェントは自分の欲しいファイルを他と協調して取得したり、その逆にネットワークに対して流通させたりするが、エージェントは自分の意図に基づいて動作するのみで、ファイル共有システム全体としての目標(強いて言うならば大量のファイル転送をネットワークに負荷をかけずに効率的に行うこと)を意識しているわけではない。別の例として、電子商取引を行うマルチエージェントシステムを考える。各エージェントはその利用者の意図(求める商品や希望購入あるいは販売価格など)を保持して利用者の意図を最大限実現するために動作する。このとき、一般的な商取引と同様に、商品を購入しようとする利用者と、販売しようとする利用者との間ではその利害が競合するため、それぞれが所有するエージェントの利害も同様に競合することになる。図 2-5 を用いて説明する。

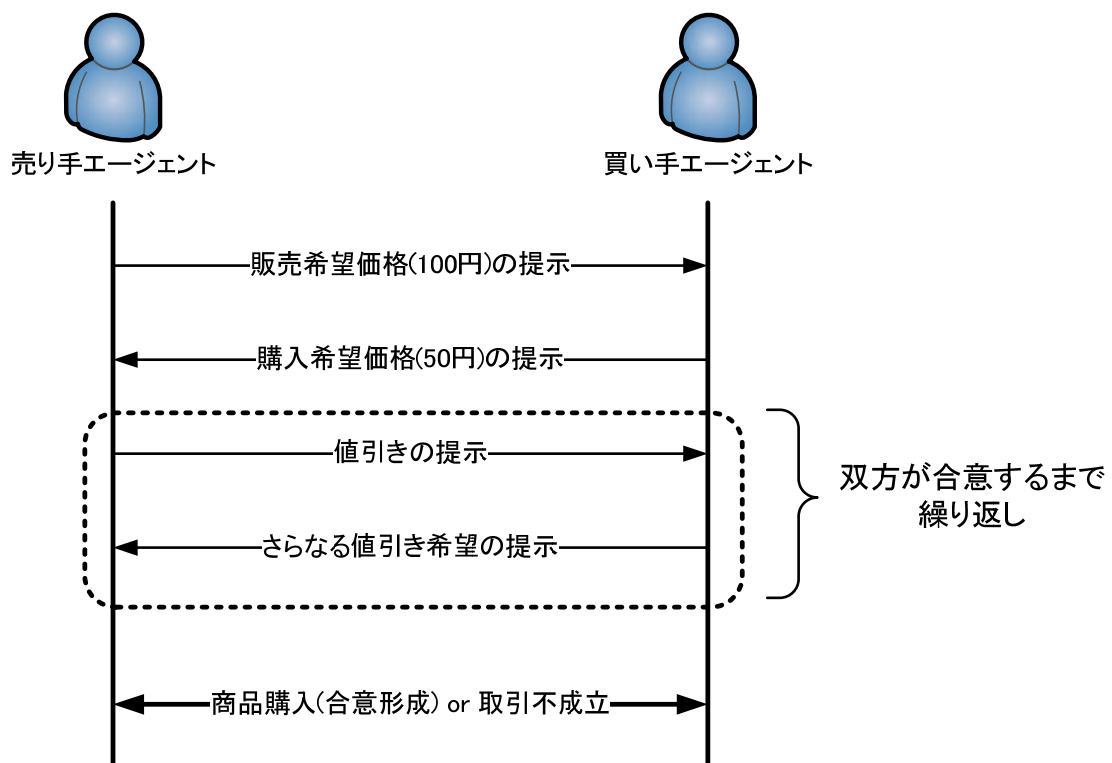


図 2-5 エージェント間での利害の競合の例

図中の売り手エージェント(の運用者)はある物品を 100 円で販売したいという要求を持ち、買い手エージェント(の運用者)は同じ物品を 50 円で購入したいという要求を持っている。このように、エージェント協調において双方の要求(意図)が競合する場合には何らかの形でこの競合を解消する試みがなされる。本例の場合であれば、例えば、売り手が販売価格を下げ、買い手が購入価格を上げることで互いの要求をある程度満たす価格を決定することで合意を形成することができる。もちろん、交渉を行っても合意に達しない場合もある。

このように、分散協調型システムであるマルチエージェントシステムでは、システムの運用形態によってエージェント間の利害の競合が生じる可能性があり、そのような運用がなされる場合、エージェントシステムではそれらを解消するための仕組みが必要であると言える。次章では、広域ネットワーク管理におけるエージェント間協調とそこで発生しうる利害の競合について説明する。

なお、マルチエージェントシステムの運用形態にかかわらず、システム全体で利用するリソース(記憶領域や計算資源など)が共通かつ有限である場合には、エージェント間でそれらのリソースをうまく共有して利用するための資源割り当ての問題が発生する。このような状況は、設計によっては問題に依存せずにマルチエージェントシステムに生じうる問題であるが、本研究では、このような資源割り当てに関わるエージェント間の競合に関してはエージェントシステムの設計の問題と捉え議論の対象外とする。

2.2.2 広域ネットワーク管理におけるエージェント間協調

広域ネットワーク管理を行うマルチエージェントシステムでは複数の AS 上に配置したエージェント間の協調を通じてネットワーク管理における観測や診断を実行する。

一般に、各エージェントの運営管理はエージェントが配置された AS を管理する主体によって行われることが想定される。AS の管理主体には大学や研究機関などの公的な機関も含まれるが、大半はインターネットの接続サービスを有料で提供する ISP (Internet Service Provider) である。ISP はユーザに対してインターネットの接続を提供することで利益を得ているため、各 ISP は自社の顧客に対してできるだけ高速で安定なネットワーク接続を提供することを目的としている。大学などインターネット接続ビジネスとは無関係な AS であっても、同様に安定で高速な接続を当然求めている。そのため、各 AS では自社の設備を増加させるなど自分自身の性能を向上させるとともに、他 AS との接続を状況に応じて随時変更しながら常に自 AS における接続性や安定性の向上に努めている。

ところが、AS が相互に接続されたインターネットにおいては、AS の互いの動作が影響を及ぼしあう関係にあるため、これら別々の AS の持つ要求がすべて同時に満たさ

れるとは限らない。このことを図 2-6 を用いて説明する。

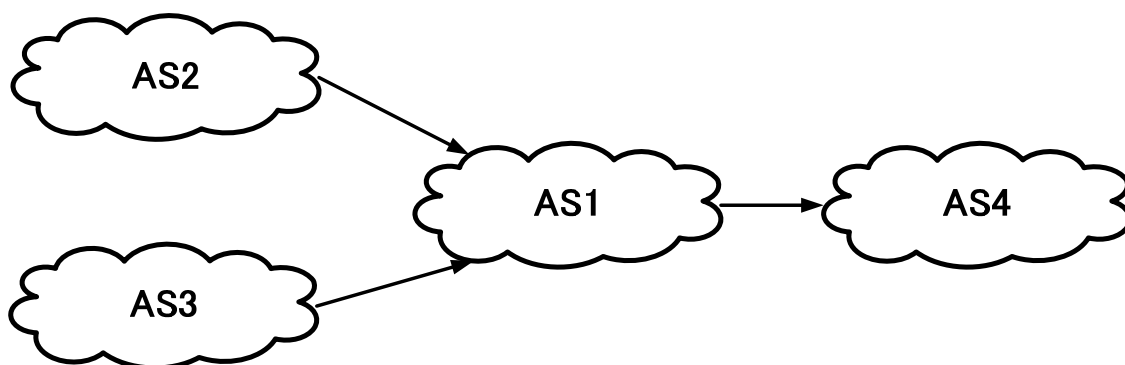


図 2-6 AS の要求が競合する例

図中の AS2, AS3 は AS1 など複数の AS(図には記載されていない)と接続することでインターネットとの接続を確保しているものとする。このとき、AS2 において AS4 へのトラフィックが増加したとする。この場合、AS2 ではトラフィックの増加に対応するために AS4 への接続を、よりホップ数の短い AS1 経由に変更する。AS1 でもその時点では(自 AS 内や AS4 との接続設備に余裕があるなどの理由から)、AS2 による経路変更を受け入れたとする。この状況下で AS3 においても AS2 と同様の AS4 へのトラフィックの増加が起きたとする。この場合、AS3 でもこの状況に対処するために同様の戦略、すなわち、AS4 向けのトラフィックを AS1 経由に変更しようとする。ところが、AS1 ではすでに AS2 からのトラフィック増加に対応したために設備的な制約などから AS3 からの新たな要求に対応できない可能性がある。このような場合、AS1 では AS3 から経路変更があったとしても直接 AS4 には流さずに別の(あまり高速でない)接続経路でトラフィックを転送する可能性がある。また、一般的な AS では本例のような自 AS をただ通過するだけのトラフィックを歓迎しない傾向があるため、AS1 の立場からすれば自 AS の運営ポリシーに反するため AS3 の要求を受けつけない可能性もある。

以上のように、インターネットにおける高速で安定した接続性の確立のためには AS 間の協調がなければならないにもかかわらず、その反面、AS の管理主体同士は利害が競合する関係にもあるため、無条件での協調作業が行われるという保証はないことが分かる。そして、このような AS 間のネットワーク管理へマルチエージェントシステムを適用する場合に当てはめると、各エージェントが観測や制御という作業を行えるのは、あくまで自分が配置されているネットワーク(AS)という限られた範囲内のみであって、AS 間での合意がなされていない限り他の AS の状態を観測したり他の AS 上のルータなどのネットワーク機器を直接操作することはできない。そのため、エージェント間の協調が不可欠である反面、エージェント間の利害の競合も必須であるということが言える。

このように、マルチエージェントシステムを広域ネットワーク管理に適用する場合にはエージェント間の競合解消機構が不可欠である。言い換えると、競合解消機構を持たずエージェント間協調を行うとエージェントの動作にASの意図が反映されない可能性があり問題が生じるおそれがある。この問題の一つとして考えられるのがエージェントを通じた情報流出である。各ASにおけるネットワーク構成などに関する情報はそれぞれのAS独自のものであり、ISPなどでは極秘情報として扱っているところも多い。一方、マルチエージェントシステムによるネットワーク管理作業においては各エージェントが観測したネットワーク状態などを協調の過程で相互にやりとりすることが行われる。このような情報の交換はあるエージェントが別エージェントに情報の公開を依頼するという形で行われるが、先に述べたとおり各エージェントが観測した情報は極秘扱いされることもあるため、ASによってはそのような情報を無条件に他エージェントに公開することはできないというポリシーを持つものもある。これは情報を求めるエージェントと与えたくないエージェント間での利害の競合が発生した状況といえ、適切な競合解消機構が必要となる。

筆者自身のネットワーク管理作業から得た知見だけでなく実際のISPからヒアリングを行った結果、この情報流出の問題は実際のネットワーク管理にマルチエージェントシステムを適用した場合の最も重要な問題の一つとしてISPが考えていることが分かった。

以上より、マルチエージェントシステムにおけるネットワーク管理を行う際の情報流出を抑えるためのエージェント間協調動作をエージェントの管理者であるASのポリシーに基づいて適切に管理する技術がマルチエージェントを利用した適応型ネットワーク管理技術を構成する上で不可欠である。

2.2.3 エージェント間協調動作管理技術に関する従来技術と課題

マルチエージェントシステムにおいて発生するこのような利害の競合という状況下において、エージェント間の交渉によって一定の合意を形成するための仕組みはマルチエージェントシステムの研究分野において従来からさまざまな方法が提案されている。その中の代表的な例が契約ネットプロトコルである。

契約ネットプロトコル(以下、契約ネット)は、タスクを独立したサブタスクに分割して、それらを複数のエージェントに割り当てるためのプロトコルである。契約ネットは次のような手順で実行される。

- ① 実行したいタスクを持つエージェント（マネージャ）は、必要であればそのタスクをサブタスクに分解し、各サブタスクに対して入札を求めるメッセージをエージェント全体に放送する。
- ② メッセージを受信したエージェントのうち提示されたタスクを実行可能なエージェントは、そのタスクに対して入札メッセージをマネージャに返信する。
- ③ マネージャは、送られてきた複数の入札メッセージを評価し、それらの中から最も適切と思われる入札を一つ選択して、それを送ってきたエージェントに落札メッセージを送る。この落札メッセージを得たエージェントが契約者である。
- ④ 契約者は与えられたタスクを実行して返答をマネージャに返す。
- ⑤ マネージャは、各部分タスクの実行結果のレポートを集め、最終的には各部分タスクに対する解の統合を行う。

このようにして、契約ネットを用いると、マルチエージェントシステムにおける階層的なタスク割り当てがマネージャを中心としてトップダウンに形成される。その他、経済学の視点を利用したアプローチとしてオークションやバーゲニングと呼ばれるタスク割り当ての方法も提案されている。

これらの契約ネットやオークションなどの競合解消およびタスク割り当ての方法は汎用的な理論であり、広域ネットワーク管理用マルチエージェントシステムにおけるエージェント間の協調動作の管理決定に適用することは可能と考えられる。しかし、これらの方法を利用するためにはプロトコルとしての動作の詳細（入札に対する応札を行うための戦略など）を決定するために、現実のネットワーク管理に適用するためには広域ネットワーク管理を行う際に生じるエージェント間の競合に関する分析や競合解消のための動作戦略に関する検討が必要となるが、従来このような検討そのものが行われていなかった。

一方、広域ネットワークで動作するマルチエージェントシステムとして提案された [S.Willmott B.Faltings, 2000][Willmott Faltings, 1999], [M.Sasabe, N.Wakamiya, M.Murata, H.Miyahara, 2003] [D.A.Tran, K.A.Hua, S.Sheu, 2003]などの方法では、

- マルチエージェントシステムを用いたアプリケーションの構成方法や手順の構成が議論の中心である
- エージェント全体が協力関係にあることが前提とされている

などの理由により、提案手法を現実のネットワークに適用した場合に生じるエー

エージェント間での利害の競合およびその解消という現実の問題については考慮されていなかった。

このように、これまでに提案されてきた研究では本章で説明したマルチエージェントシステムを広域ネットワーク管理に適応する際に問題になるエージェント間協調動作の管理について十分な検討がなされてきたとは言えない。実際のネットワーク管理において生じるエージェント協調に関する問題を分析し、それを解決する新たなエージェント間協調動作管理手法の提案が広域ネットワーク管理におけるエージェント間協調管理技術を構成する上で必須となる。

第3章 広域ネットワーク管理のためのエージェント組織化支援技術

関連論文 【A1】，【A2】，【A5】，【A7】，【A8】，【B2】 - 【B4】

3.1 本論文における位置づけ

本章では，マルチエージェントシステムによるネットワーク管理において，効率的かつ効果的なエージェント間協調を実現するためのエージェント組織化支援技術について明らかにする．本章の位置づけを図 3-1 に示す．

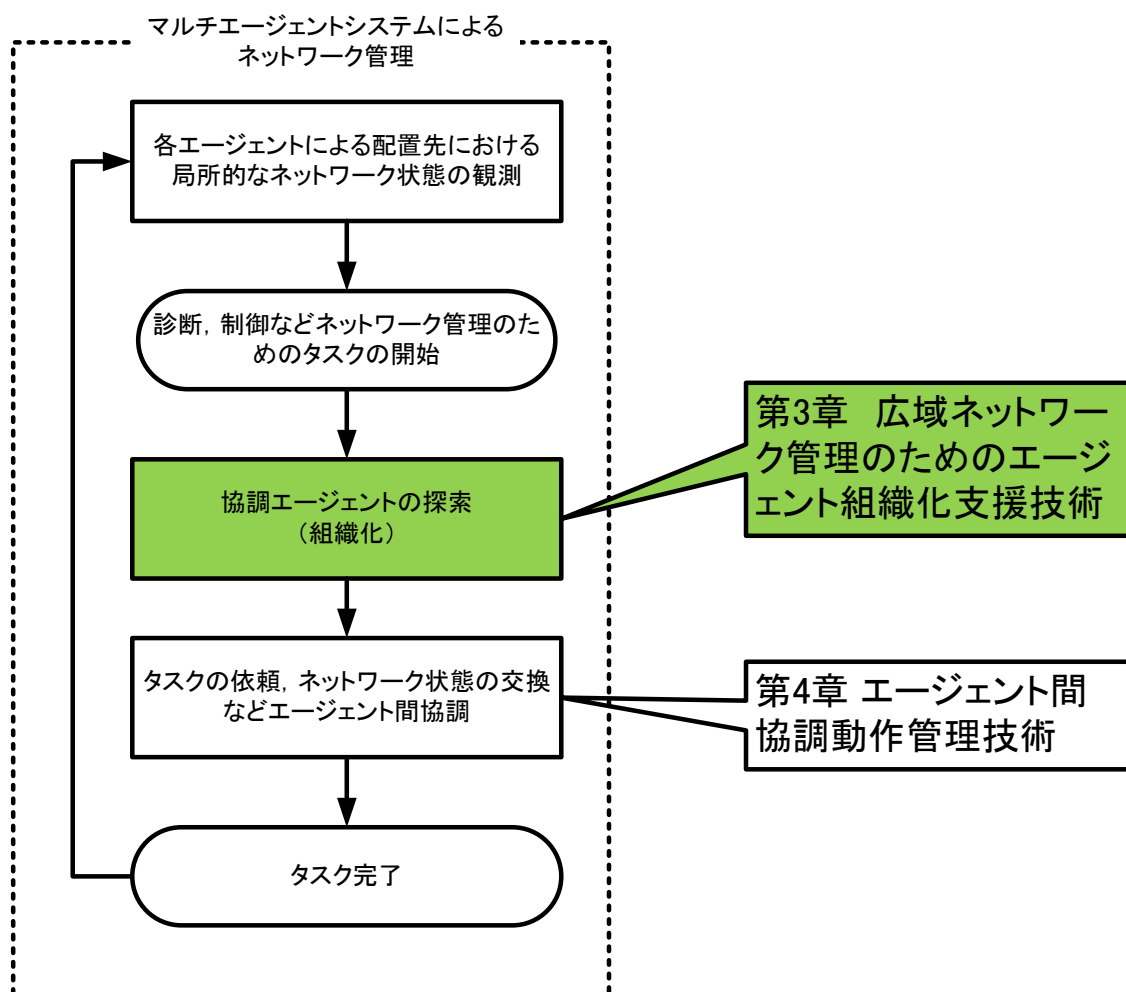


図 3-1 本章の位置づけ

はじめに、提案技術の重要性、必要性を明らかにするために広域ネットワーク管理における分散協調作業の重要性を実例を用いて述べるとともに、広域ネットワーク上で分散協調を行う際のエージェント間協調の困難さを明らかにする。そして、それらへの解決策となるエージェント組織化支援システムおよびそれを含むネットワーク管理用エージェントアーキテクチャを提案する。提案システムによるエージェント組織化支援の詳細を説明した後、実際に実装したエージェント組織化支援システムを複数の仮想 AS からなる仮想ネットワーク環境で動作させて得られた検証結果を示す。

3.2 研究の背景

インターネットは近年その巨大化および複雑化が進んでおり、アプリケーションやサービスの品質に対する要求の高まりもあって、インターネットを適切に管理制御することはますます難しくなっている。そのため、インターネットの管理の自動化、高度化への AI の応用が期待されるが、元来インターネットは集中的な管理主体を持たない分散システムであり、この特徴が管理の自動化を困難にしている。

マクロな視点で見た場合、インターネットは独立した主体が管理運用する AS(Autonomous System)が相互接続した巨大な分散システムと捉えられる。個々の理主体は通常自分が管理する AS 内しか観測制御できず他の AS の状態を知るができないため、複数の AS を跨る広域ネットワークを管理・制御するには AS 管理主体)同士による協調作業が必要である。例えば、インターネットで通信を行うためには経路情報を適切にネットワーク全体に伝搬させる必要があるが、この経路情報の伝搬は経路情報を受信した AS が、さらに別の AS に経路情報を転送するという AS 間の協調によって実現されている。各 AS における転送作業はそれぞれの AS のポリシーに従って行われており、他の AS がそれを観測し、制御することはできない。そのため、経路情報の伝搬障害が起きたとしても特定の AS からの観測のみで原因を知ることはできず経路を伝搬した AS が協調して原因を調査する必要がある。別の例として、高品質の動画など大容量のコンテンツをインターネットで転送する場合、転送経路上の AS の 1 つでも十分な転送速度が確保できなければ転送の遅延などの問題が発生する。この場合も各 AS では自分以外の AS のトラフィックの状態を知ることはできないことから、原因となる AS の発見など問題解決には AS の協調が不可欠である。

このように、本質的に AS 間の分散協調作業である広域ネットワークの管理の自動化を単一システムによる集中管理で実現することは不可能であり、分散構造を持つマルチエージェントシステム[E.H.Durfee, V.R.Lesser, D.D.Corkill, 1987]を用いたアプロー

チが特に有効であると考えられる。実際にこのアプローチに基づいて多くの提案がなされている[O.Akashi, A.Terauchi, K.Fukuda, T.Hirotsu, M.Maruyama, T.Sugawara, 2005], [寺内, ほか, 2004], [S.Willmott B.Faltings, 2000], [D.A.Tran, K.A.Hua, S.Sheu, 2003], [M.Sasabe, N.Wakamiya, M.Murata, H.Miyahara, 2003], [M.M.Hefeeda, B.K.Bhargava, D.K.Y.Yau, 2004].

一般にマルチエージェントシステムを使った作業ではエージェント間の協調が必要であり、この協調作業を効果的に実行するためには、実行する作業の内容やエージェントの状態などに基づいて選択された適切なエージェント群が協調作業を行うことが重要である [S.Willmott B.Faltings, 2000]. この選択が適切になされていない場合、例えば、協調相手のエージェントが依頼した作業を実行するための機能を持たない、問題解決に必要な情報を観測できない、高負荷などのため依頼された作業を実行できない場合には協調がうまくいかず結果として作業の実行が失敗する可能性がある。本稿では、マルチエージェントシステムにおいて要求されるエージェントの機能や配置を考慮しながら協調相手とすべきエージェント群を選択する作業をエージェントの組織化と呼ぶ。組織化はエージェント間の協調を行うすべてのマルチエージェントシステムが行うべき作業であり、それ故に適切な組織化を実現することはそれらのマルチエージェントシステムに共通した要求であると言える。

通常、組織化は可能性のある作業を想定した上で事前に完了させておくか、マルチエージェントシステムによる作業の実行中にエージェントによって動的に行うことが想定される。しかし、事前に組織化を行う場合、各エージェントは自分以外のエージェント情報を管理しなければならない。そのため、エージェントの数が大きい場合やエージェントの状態が頻繁に変化する場合などには、この情報の維持管理コストが大きくなるため、適切な選択が難しくなる。さらに、広域ネットワーク管理用のマルチエージェントシステムでは、エージェントの組織化のために他に AS を跨る広範囲のネットワーク情報が必要であるが、集中管理機構を持たない分散システムであるインターネットに関して、そのような広範囲の情報を観測収集することは容易なことではない。

我々は広域ネットワーク管理の高度化、自動化を目指して AS 間経路診断、低負荷によるコンテンツ配信などを行うマルチエージェントシステム[O.Akashi, A.Terauchi, K.Fukuda, T.Hirotsu, M.Maruyama, T.Sugawara, 2005], [寺内, ほか, 2004]の提案を行っている。これらの広域ネットワーク管理用のマルチエージェントシステムを運用する上でもエージェント情報の維持管理やネットワーク情報の収集の困難さに伴うエージェントの組織化のコストの高さが大きな問題となっていた。

そこで本稿では、[O.Akashi, A.Terauchi, K.Fukuda, T.Hirotsu, M.Maruyama, T.Sugawara, 2005], [寺内, ほか, 2004], [D.A.Tran, K.A.Hua, S.Sheu, 2003], [M.M.Hefeeda, B.K.Bhargava, D.K.Y.Yau, 2004]などの AS を超えた広域ネットワークを管理するマルチエージェントシステムを対象にして、ネットワーク上の位置など特定

の条件を満たすエージェント群を動的に探索・同定することによりエージェント間の組織化を支援するエージェント組織化支援システム ARTISTE(Agents' Roles and Topological Information management System for agent TEamwork) を提案する。

ARTISTE は次の特長を持つ。

- ① 広域ネットワーク管理用マルチエージェントシステムにおける組織化作業の共通性に着目してARTISTEをマルチエージェントシステムと独立した共通的基盤とすることにより、複数のマルチエージェントシステムに支援機能を提供することが可能である。
- ② 組織化のために必要な情報を複数のASで分散観測することで、単一のASでの観測では獲得の難しい広域ネットワークトポロジ情報などを利用した組織化が可能である。

ARTISTE を用いることにより、広域ネットワーク管理用マルチエージェントシステムでは組織化を静的に行う必要がある場合と比べてマルチエージェントシステムの運用コストを下げることができる。さらに、マルチエージェントシステム 本来の問題解決の処理とエージェントの組織化に関する情報や処理を分離できることからネットワーク管理用マルチエージェントシステムを構築するためのコストが低減される効果を明らかにした。

3.3 広域ネットワーク管理を行うマルチエージェントシステムの実例

本章では、広域ネットワーク管理を行うためのマルチエージェントシステムの実例を2つ挙げる。

3.3.1 AS 間経路診断システム ENCORE

広域 IP 網経路障害診断システム ENCORE は、AS 間において発生した経路障害を迅速に検知、診断を行うためのシステムである。経路障害の特徴として、あるASによって発行された経路情報はネットワークを伝搬する過程において各ASのポリシーに基づいて随時変更される可能性があり、そのため、経路の発行元のASですら自分の経路が一旦ネットワーク上に流れてしまった後の変化を把握することができず障害の内容

や原因を単独で推測することが困難である、という性質が挙げられる。このような性質を持つ経路障害に対応するために、ENCOREではネットワーク上の複数のASにエージェントを配置して、それらのエージェントが協調して経路情報を相互に観測し、情報交換することによって経路障害の検知や診断を行っている。ENCOREが診断の対象とする経路情報の伝搬障害とその診断手順の簡単な事例を図3-2で説明する。

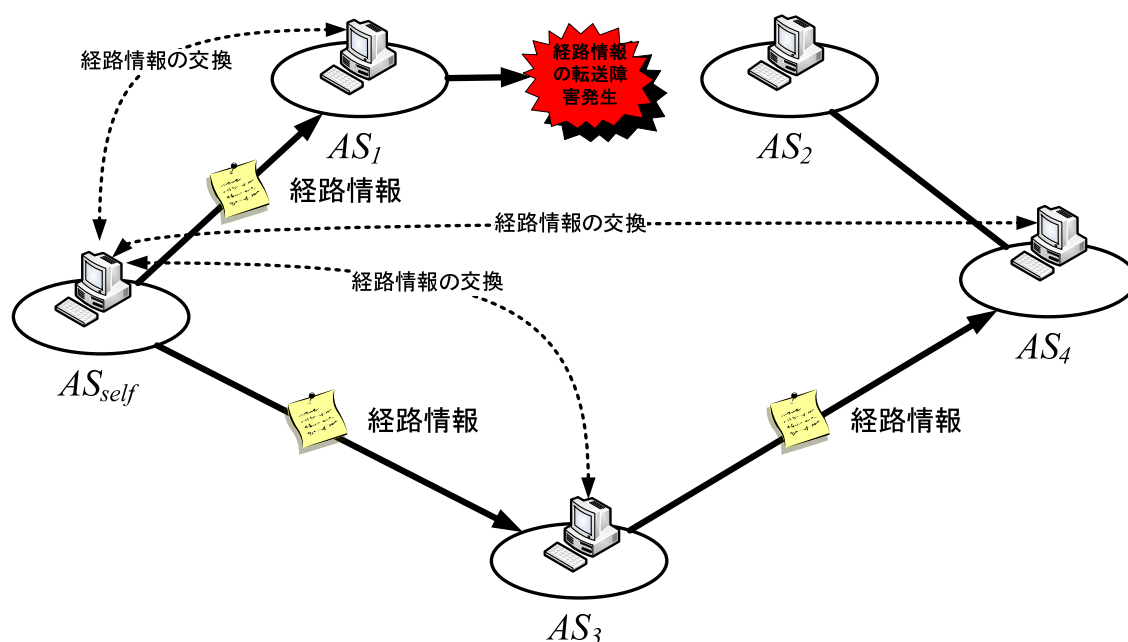


図 3-2 ENCORE による経路診断の例

図中において、 AS_{self} から advertise された経路情報は AS_1 , AS_2 あるいは AS_3 を経由して AS_4 に伝搬されるものとする。ここで、 AS_1 と AS_2 との間で AS_{self} に関するフィルタの設定に誤りが発生して、 AS_{self} に関する経路情報が AS_2 に流れなくなった状況を想定する。この状況を従来の方で、 AS_{self} から診断ツールなどを用いて調査を行っても、 AS_1 から先へとパケットが到達しないという以上の解析は不可能である。つまり、 AS_1 , AS_2 間の接続の障害なのか AS_2 自身のフィルタの設定誤りなのか判断できない。

これに対して、ENCORE を用いた診断では、 AS_{self} 中のエージェント R_{self} が AS_1 , AS_2 のエージェント R_1 , R_2 と通信し、それぞれの AS において観測された経路情報を送り返してもらう。その結果、 AS_1 では自分の advertise した経路情報が存在するのに対し、 AS_2 では AS_1 の経路情報はあっても自分の経路情報が含まれていないことが分かる。以上より、 AS_{self} では AS_1 と AS_2 間で AS_{self} の経路情報に関するフィルタの設定に誤りがあるということが推論できる。

3.3.2 マルチエージェントによる広域動画配信アーキテク

チャ

ADSL や FTTH といったブロードバンド環境が普及して、インターネットの高性能化、高品質化が進んでいる。しかし、インターネットのコンテンツもこれに伴い、従来の静止画中心から動画中心に移り変わっており、サーバの負荷やネットワークトラフィックの増加といったリソースの問題は依然として残っている。しかも、今後のインターネットにおける動画配信は、コンテンツの高品質化や多チャンネルによる複数コンテンツの同時配信、また短時間の動画から映画などの数時間に渡る動画まで多様化が進むと筆者らは予想しており、前記のリソースの問題は今後さらに顕著になる可能性がある。

インターネットによる多チャンネルの動画配信は、従来以上にユーザが on-demand でコンテンツを自由なスタイルで視聴することを可能にする。例えば、チャンネル(コンテンツ)を頻繁に変える、早送りや巻き戻しを行いながらコンテンツ中の好きなどころだけを少しずつ見る、現在、配信中のコンテンツを少し遅れてから視聴する、コンテンツの主要なところだけを飛ばし見しながらダイジェスト的に視聴する、といった多様な視聴スタイルが考えられ、これからの動画配信においては、このような多様性をサポートする必要があると考える。これに加えて、利用者の好みの傾向、短期的な人気、事件やイベントによる特定コンテンツへの過度の集中などのダイナミズムが配信系に発生する。しかし、インターネットでは全体を管理するメカニズムもなく、通常の負荷分散を利用するのが困難である。このような状況を考慮して、インターネットを用いて不特定多数のユーザに対して多種類の動画を配信するための方法が求められる。このような問題に対し、各種サーバを自律したエージェントとしてモデル化し、リソースの負荷の集中や偏りを防ぐとともに、動的に変化する利用者の要求に応えるメカニズムが提案されている。

まず、この方式では図 3-3 に示すような既存の CDN と同様の動画配信アーキテクチャを想定する。

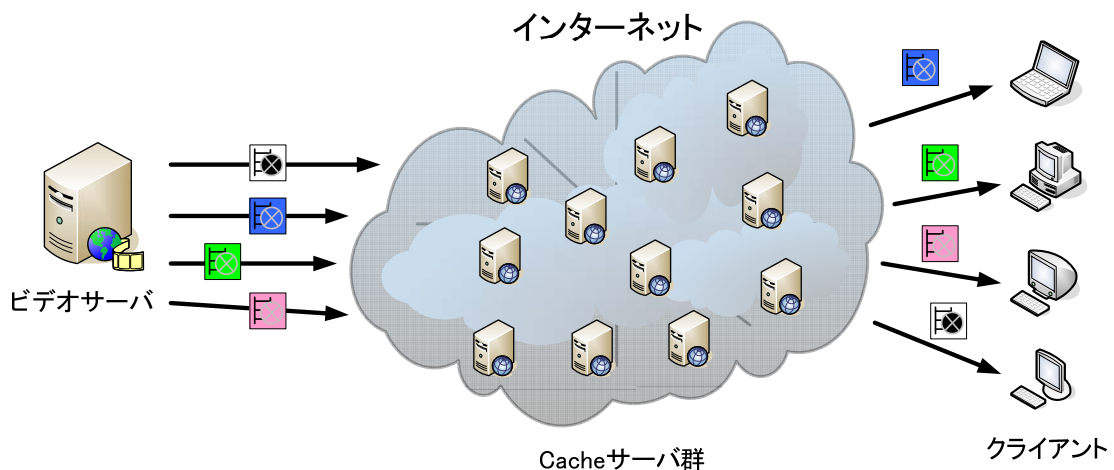


図 3-3 動画配信アーキテクチャ

このアーキテクチャでは、コンテンツのオリジナルを保持するビデオサーバと動画を視聴するクライアントの他に、コンテンツ配信業者、ISPなどが設置することを想定したキャッシュサーバが Internet 上に多数存在する。キャッシュサーバの初期配置は網の構成（帯域やクライアント数）などから決定する。設計においては既存の CDN で用いられている方法や知見を利用する。それぞれのキャッシュサーバは自律的にネットワークの状態や自己の状態を観測・解析し他のキャッシュサーバと協調することにより自己の動作を自律的に決定できる。なお、ビデオサーバでは、ディスクと関連するメモリの使用量は高いが、CPU 使用率は相対的に少ないと仮定する。また、エージェント間の協調のための通信コストはコンテンツと比べかなり少なく、配信には影響を与えないと考える。

各キャッシュサーバは動画を配信する際には、配信中の動画を一定サイズだけ自分のキャッシュスペースに保存しながら配信を行う。データはちょうど一定サイズの Window をずらすイメージで保存され、配信が進むにつれてキャッシュのデータも変わる。キャッシュのサイズはキャッシュサーバの storage の容量から決定するものとする。storage 容量に空きがなくなった場合はキャッシュのデータは最も古いデータから置き換わっていき、それ以外にも一定期間アクセスのなかったキャッシュデータはキャッシュスペースから削除される。

本アーキテクチャにおける基本的なキャッシュ制御の手順を以下に示す。

- ① クライアントはネットワーク中のキャッシュサーバの 1 つに対してリクエストを送信する。このとき、キャッシュサーバの選択には DNS redirection など既存の CDN で用いられている方法を利用することとする。
- ② クライアントからリクエストを受けたキャッシュサーバ(以降では他のキャッ

キャッシュサーバと区別するために **Mediator** と呼ぶこととする)は、以下の手順によりリクエストされた動画(の一部分)を所有するキャッシュサーバの集合を形成する。

- A) **Mediator** はまず自分の「近傍」を探索してリクエストされた動画(の一部)を保持するキャッシュサーバの集合を形成する。
 - B) 前ステップにおいてコンテンツをキャッシュしているキャッシュサーバが発見できた場合、**Mediator** は当該キャッシュサーバに対して対応する画像を近く利用するという要請を出す。要請を受けたキャッシュサーバは、自己のリソースを勘案し、この要求をコミットする。コミットを行ったキャッシュサーバは、実際にそのデータが使われるまで可能な限りそのデータを保持する。もし、リソースの関係でコミットメントを放棄しなくてはならない場合や、近隣で同じキャッシュ内容を持つ適切なエージェントが存在する場合には、新たなコミットメントを結ぶように **Mediator** に要請する。
 - C) **Mediator** の「近傍」の探索では、動画の一部あるいは全部が見つからないときは、最初の探索要求を受けたキャッシュサーバがさらに自分の「近傍」内で探索を行うことによりデータを取得できるか試みる。この、探索の **forwarding** は一定回数だけ行う。この場合も、データが見つかった場合は上記と同様に **Mediator** との間でコミットメントを確立する。
 - D) (c)においてすべてのデータを所有するキャッシュサーバの集合が形成できた場合は、この集合のリストをクライアントに返す。集合が形成できない場合はコンテンツの足りない部分はサーバから取得する。このとき、サーバからクライアントに直接データを転送するのではなく、サーバから **Mediator** を介してクライアントにデータを転送する。**Mediator** は転送の過程において自己のキャッシュにあるデータを随時更新する。
- ③ クライアントは返されたキャッシュサーバのリストを元にして、適宜キャッシュサーバを切り替えながら動画を受信する。もし配信の途中でリクエストが変更あるいはキャンセルされた場合には、**Mediator** はコミットメントを確立した各キャッシュサーバに通知をしてコミットメントを解消する。

次に、この動画配信方法においてエージェントを組織化する方法について述べる。本稿ではキャッシュサーバ(エージェント)の持つ役割を、各々がどのリクエストを処理するかという観点で捉える。この考えから、エージェントの役割の動的な割り当てという手続きを、個々のキャッシュサーバがネットワークおよび自己の状態や他者との協調によって自己のキャッシュの内容や処理するリクエストを動的に制御することと考え

る。この、協調による役割の割り当ての基本制御として以下の3つを提案する。

- ① 複製：他者に対して、自分の持っている役割と同一の役割の割り当てを依頼する手続きである。これは、例えば、あるキャッシュサーバに対するリクエストが大量になったときに利用する。このとき、別のキャッシュサーバに自分と同じ役割を持ってもらう、つまり、同じコンテンツをキャッシュしてもらうことで同種のリクエストを2者で分担して処理できるようになり、負荷分散が実現できる。

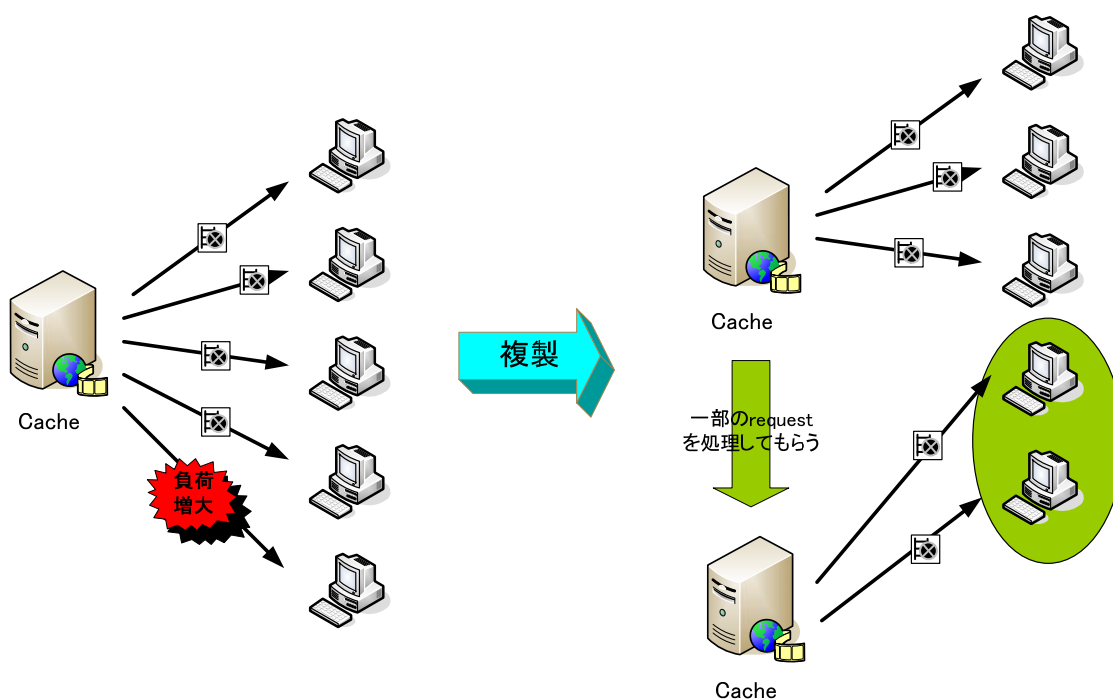


図 3-4 役割の複製

- ② 委譲：複製とは逆に、自分の現在の役割（の一部）を他者に割り当てて、自分はその役割の割り当てをやめる手続きである。あるコンテンツに対する人気さが下がりリクエストの量が減少してきた場合に、複数のキャッシュサーバで処理しているリクエストを処理能力のある1つのキャッシュサーバに集約して処理させるといった場合に行う。これにより他のキャッシュサーバのリソースを別の処理に使えるためリソースを効率的に利用できる。

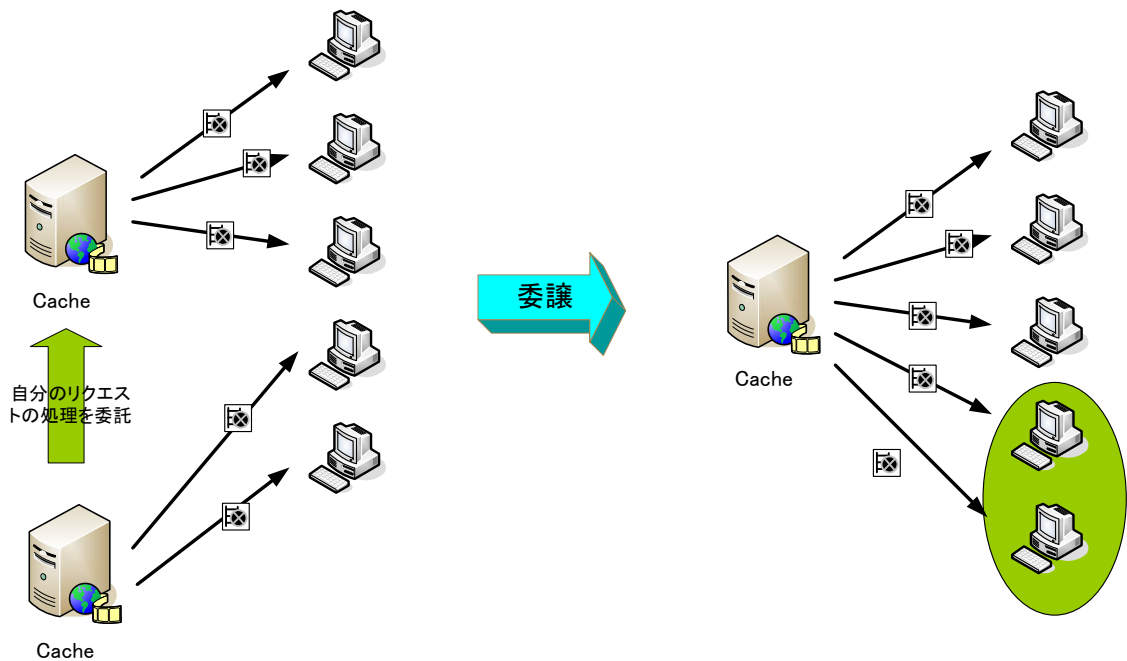


図 3-5 役割の委譲

- ③ 交換：委譲手続きの応用であり、自分の役割を他者に委譲してその代わりに他者の持つ役割の割り当てを受ける手続きである。例えば、あるキャッシュ A がコンテンツ A に対するリクエストを多く処理しているときコンテンツ B に対するリクエストが来たものとする。このとき、コンテンツ B に対するリクエストを多く処理しているキャッシュ B と協調して、コンテンツ A の処理という役割とコンテンツ B の処理という役割を交換する。多種類の動画を配信する場合、1 つのエージェントが複数のコンテンツのキャッシュを持つ、つまり、複数の役割を持つことが当然である。また、動画コンテンツはサイズの大きさから複数のキャッシュに分散して保存されるが分散されたキャッシュから部品を集める作業はネットワーク帯域を消費する。本手続きを用いると 1 個のコンテンツのキャッシュ をできる限り少ないサーバにキャッシュされるようにすることができリソースの効率的利用に有効である。

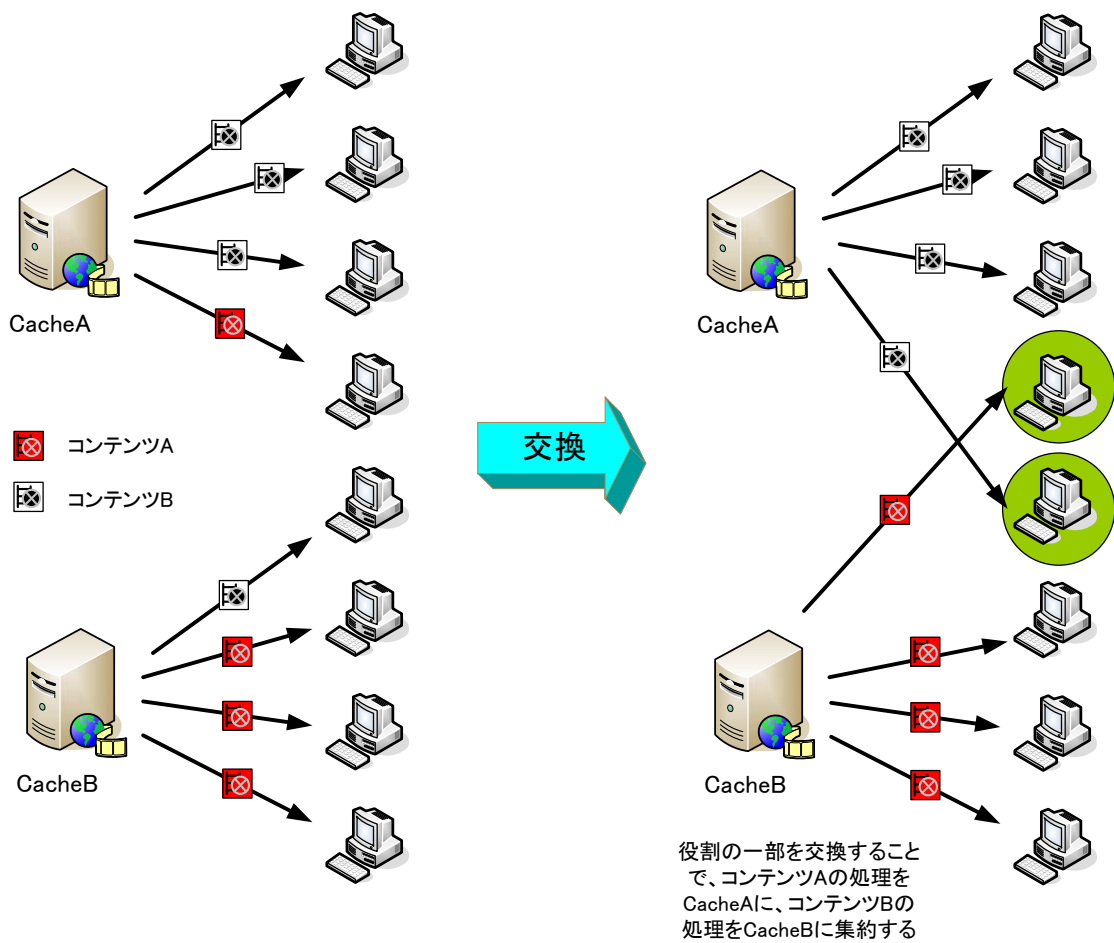


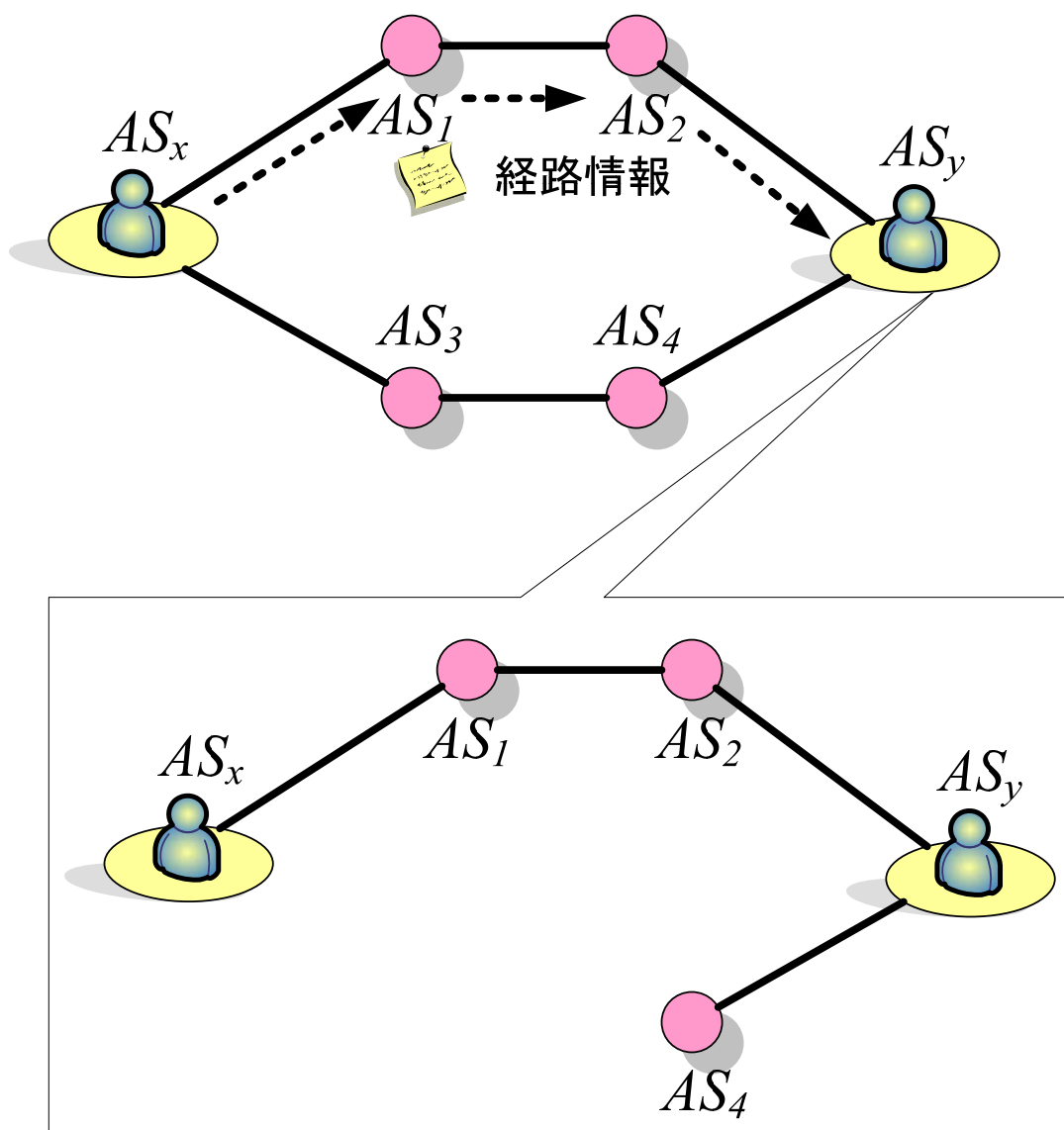
図 3-6 役割の交換

3.4 問題の所在とアプローチ

3.4.1 広域ネットワーク管理用マルチエージェントシステムにおけるエージェント組織化の問題

一般的なマルチエージェントシステムにおいて組織化を行う場合、エージェントの能力や保持する情報などを元にしてエージェントの選択を行うが、ネットワーク管理用マルチエージェントシステムにおいて組織化を行う場合、それらに加えてネットワークトポロジ、トラフィック、観測できる経路情報といった実ネットワークに関する情報に基づいて組織化を行うことが求められる。

特に、本稿で対象とする広域ネットワーク管理用のマルチエージェントシステムでは、広域ネットワーク管理が AS 間の協調作業であることから分かります。エージェントは離れた AS 上に分散配置されている必要がある。そのため、エージェントの組織化には各エージェントが動作している環境、すなわち地理的に離れた別の AS に関するネットワーク情報が必要であることになる。しかし、一般に AS は独立した組織によって運営されており、組織同士の利害関係やセキュリティ上の理由から AS 内部の情報を他 AS から観測することは許可していないことが多い。そのため、各々の AS に配置されたエージェントは他の AS で観測できる経路情報やトラフィックを知ることはできず、自分が配置された AS 周辺のネットワークに関する部分的な情報しか観測できない。



AS_y のエージェントが獲得できるトポロジ

図 3-7 単一のエージェントによるネットワーク情報の収集

個々のエージェントが部分的なネットワーク情報しか観測できないことを図 3-7 を用いて説明する。この例では AS_y が自分の近傍の AS 間トポロジを取得しようとしているものとする。トポロジの取得には AS 間での経路情報交換に使われる標準プロトコルである BGP(Border Gateway Protocol) (Rekhter & Li, 1995) でやりとりされる AS-Path 情報を利用する。AS-Path 情報とは経路情報が転送される際に経由した AS のリストである。実際のルータ(AS_y 上にあるものとする)から得られる AS-path 情報の出力例を以下に示す。

```
Address Metric Path
0x6122 0      ASy AS2 AS1 ASx
0x6213 0      ASy AS4 AS5 AS6
....
```

この例では各行が 1 つの経路情報に相当し、Path カラムがその経路の AS-path 情報である。最初の行の AS-path 情報は、当該経路が AS_x によって生成され、 AS_1, AS_2 をこの順番で経由して AS_y で受信されたことを示す。このことから、" $AS_y-AS_2-AS_1-AS_x$ " という AS 間の接続が存在することが分かる。2 行目以降についても同様に各行から AS 間接続情報が得られるため、それらをまとめることで特定の AS から観測した AS 間トポロジを取得できる。

このとき、 AS_x から AS_y へ経路情報が AS_1, AS_2 を経由して転送されているとすると、 AS_y は " $AS_y-AS_2-AS_1-AS_x$ " という Path が形成されていることを知ることができる。さらに、 AS_y は自分と直接接続された AS の存在は分かるので、 AS_4 の存在も把握できる。しかし、これらの情報だけでは AS_y から図中の AS_3 の存在を知ることができず、結果として図に示した部分的なトポロジしか獲得できないことが分かる。

このとき、例えば AS_y 上のエージェントが経路診断やトラフィック制御の目的で AS_x と直接接続された AS 上のエージェントと協調する必要があるとしても、自分が獲得できるトポロジ情報からは該当するエージェントが AS_4 上のエージェントであることしか把握できず、存在の分からない AS_3 上のエージェントとは協調が行えないことになる。

以上をまとめると、広域ネットワーク管理用マルチエージェントシステムにおいてはエージェントの協調相手を適切に選択するためには、エージェントの配置先である AS から観測できる局所的な情報に加えて、近隣の AS を含む、より広い範囲のネットワークに関する情報が求められるが、集中的な管理システムを持たないインターネット上ではこのような広い範囲の視点からの情報を得ることは容易ではない。

よって、これらの問題を解決して広域ネットワーク管理用マルチエージェントシステムの適切な組織化を行うためには、広域ネットワーク情報の収集機能を備えたエージェント組織化支援機構が必要である。

3.4.2 アプローチ

3.4.2.1 アーキテクチャ

エージェントの組織化機構は本来個々のマルチエージェントシステムに固有のものと考えられる。しかし、ネットワーク管理のようなマルチエージェントシステム本来の作業のための処理や知識に加えて、エージェント選択のための知識や手続きを混在させることはマルチエージェントシステムの構成を複雑化させ構築コストの増大および実行速度や保守性の低下を招く。

さらに複数の広域ネットワーク管理用マルチエージェントシステムを開発する場合を想定すると、それぞれのマルチエージェントシステムが独自に組織化機構を持つことになり非効率であると共に、それぞれのマルチエージェントシステムでも構築コストの増大や保守性の低下といった問題を抱えることになる。

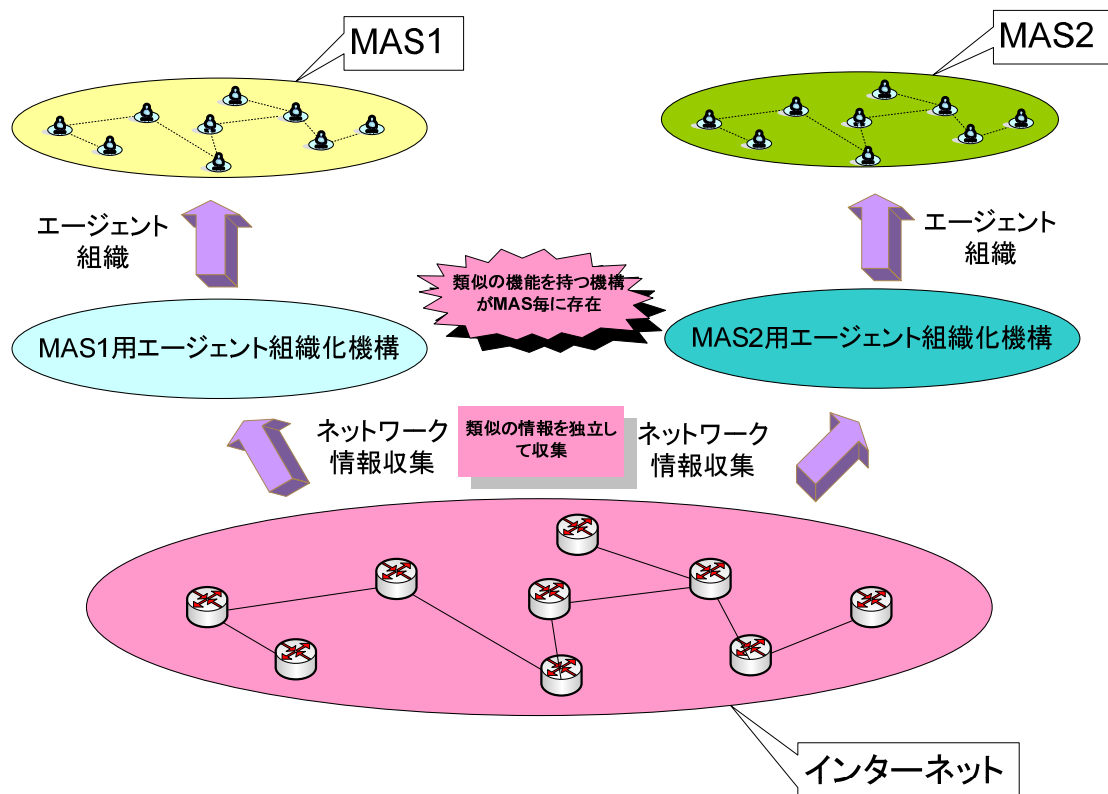


図 3-8 個々のマルチエージェントシステムによって組織化が行われる場合

組織化機構に求められる機能やアーキテクチャを明確にするために実際のマルチエ

エージェントシステムにおける組織化の方法や用いられる情報を調査した。本稿では検討対象のマルチエージェントシステムとして、AS 間経路診断システム ENCORE [O.Akashi, A.Terauchi, K.Fukuda, T.Hirotsu, M.Maruyama, T.Sugawara, 2005], 寺内らの動画配信システム [寺内, ほか, 2004], Tran らの動画配信システム [D.A.Tran, K.A.Hua, S.Sheu, 2003], hefeeda らの動画配信システム [M.M.Hefeeda, B.K.Bhargava, D.K.Y.Yau, 2004]を選択した。これらのマルチエージェントシステムではそれぞれ AS 間での経路診断, 広域ネットワーク上での動画配信といった作業を複数のエージェントにより実現するもので対象とするマルチエージェントシステムの要件を満たしている。これらのシステムでエージェントを組織化するために利用している情報を表 3-1 にまとめる。

表 3-1 マルチエージェントシステムの組織化において用いる情報

ENCORE	エージェントの状態(能力) AS 間トポロジ情報 各 AS における接続数
寺内らのシステム	エージェントの状態 (キャッシュの種類, 量) ノード間の距離(ホップ数, RTT)
Tran らのシステム	エージェントの状態(能力) ノード間の距離 ノードの負荷(トラフィック)
hefeed らのシステム	ノードの負荷(トラフィック) 配置先 AS 番号 配置先のネットワーク prefix

この表からエージェントを組織化するための情報として各マルチエージェントシステムともエージェントの情報に加えてトポロジなどのネットワーク情報を用いていることが分かる。そして、近隣のネットワーク機器との情報交換や遠方のエージェントとの連携により、必要とされるネットワークトポロジや経路情報の到達性, トラフィックなどの情報を収集し, 特定の条件の下にあるエージェントを動的に探索する作業はいずれのマルチエージェントシステムにおいても共通に行われることが分かった。

以上のように, 広域ネットワーク管理用マルチエージェントシステムのための組織化ではその手順や必要とする情報にある程度の共通性があることが分かった。我々はこの共通性に着目し, マルチエージェントシステムの組織化に必要な情報を一元的に収集管理して特定の条件下にあるエージェント群を動的に探索・同定する機能を複数の広域ネットワーク管理用マルチエージェントシステムに対して提供できる共通の基盤としてエージェント組織化機構を実現する。

3.4.2.2 広域ネットワーク情報の収集

3.4.1 で述べたとおり，広域ネットワーク管理用マルチエージェントシステムにおけるエージェントの組織化においては，特定の観測地点(AS)から観測できるネットワーク情報に加えて，より広い視点から観測したネットワーク情報が必要であるが，インターネットでは単一地点からの観測ではそのような広範囲のネットワーク情報を取得することは難しい．そのため，組織化機構を集中型の単一システムによる実現したのでは組織化に必要なネットワーク情報を十分に観測できないため，組織化機構に求められる条件を満たすことができない．

そこで，我々が提案する組織化機構はインターネットと同じく分散協調型の構造を持つマルチエージェントアーキテクチャによって実現することとした．そして，ネットワーク上に配置されたエージェントがそれぞれ局所的な情報を分散観測し，エージェントが協調することによって広域ネットワーク情報を取得するアプローチを採用した．

3.5 提案方式

3.5.1 エージェント組織化支援システム ARTISTE

以上の検討を元に，本稿ではエージェント組織化支援システム ARTISTE を提案する．ARTISTE を含むエージェントアーキテクチャを図 3-9 に示す．本モデルでは実際に広域ネットワークの管理作業を行うマルチエージェントシステム(図上部)と ARTISTE(図中部)が共にインターネット上で連携して動作する．以降，前者のマルチエージェントシステムを ARTISTE と区別するために，ネットワークマルチエージェントシステム(マルチエージェントシステム)と呼ぶ．

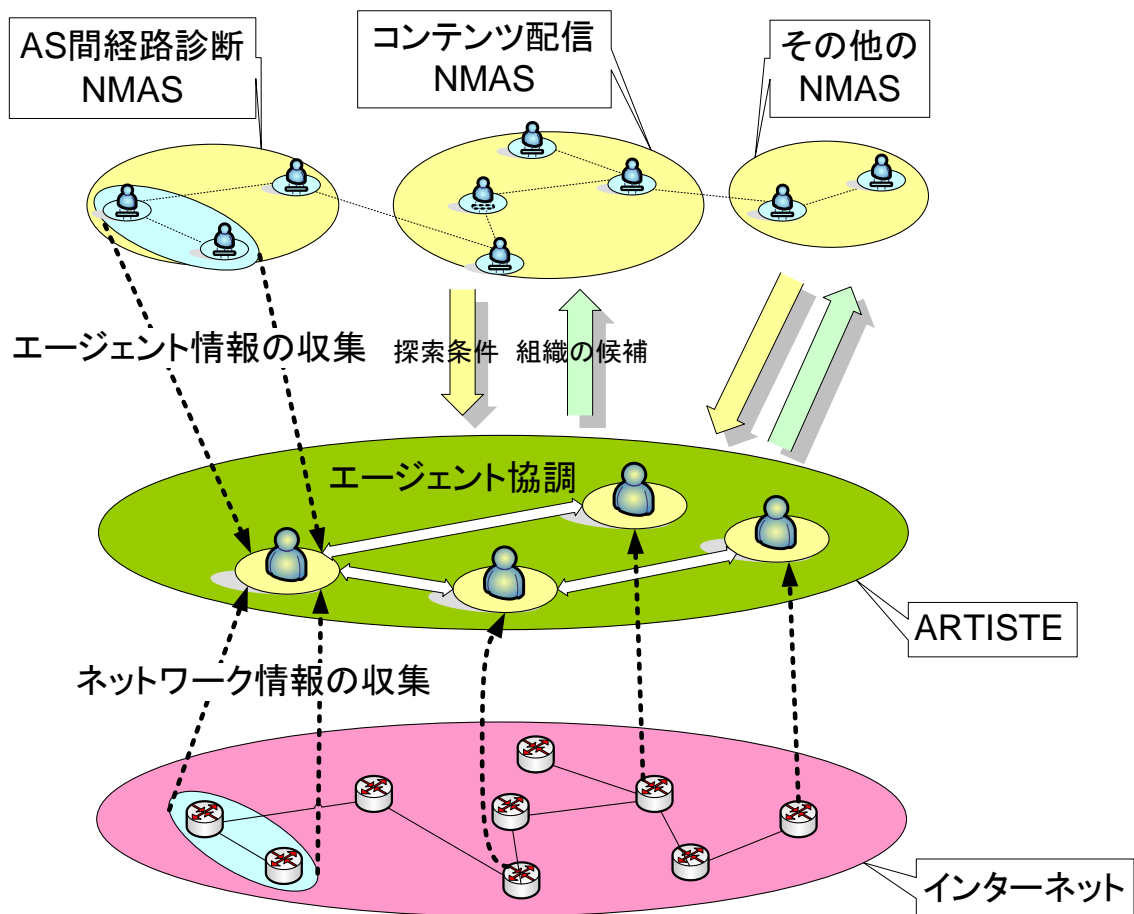


図 3-9 提案するエージェントアーキテクチャ

3.4.2.1 での検討から, ARTISTE はマルチエージェントシステムとは独立したシステムとして実現した. そして, マルチエージェントシステムと ARTISTE 間で連携するためのインタフェースとしてマルチエージェントシステムから ARTISTE に協調エージェントを探索するための条件を送信し, ARTISTE は送信された条件に合致するエージェントの情報を返答するという一般的な query-reply プロトコルを採用した. インタフェースを簡便かつ汎用的なものにすることにより, ARTISTE は多くのマルチエージェントシステムと連携可能になる. 見方を変えると, ARTISTE はマルチエージェントシステムに対して一種のメディアータシステムあるいはネームサーバとして動作すると言える.

さらに, 3.4.2.2 での検討に基づき, ARTISTE は複数の自律エージェントからなるマルチエージェントシステムとして実現し, ネットワーク上に分散配置したエージェント(以下, ARTISTE エージェント)によってネットワーク情報を複数地点から観測する. 3.4.1 で述べたように, 本稿で対象とするマルチエージェントシステムは AS 間での協調作業をサポートすることが目的であるから AS 毎にエージェントが配置されていると

してよい。さらに、同じく 3.4.1 で示したように特定の AS から他 AS の情報を観測することは困難であることから、ネットワーク情報およびマルチエージェントシステムのエージェント情報を最も効率よく収集するためには ARTISTE エージェントは AS 毎に配置されることが最適であると言える。以降では ARTISTE エージェントは AS 毎に配置されているものとして議論を進める。ARTISTE エージェントの最適な配置はマルチエージェントシステムの構成やネットワーク構造によって変化するため検討の必要な問題と考えられるが、本稿での議論の範囲外とする。

個々の ARTISTE エージェントは配置先 AS 内のネットワーク機器やマルチエージェントシステムエージェントの情報を取得する。各 ARTISTE エージェントは自分が配置された AS 以外の AS で観測されるネットワーク情報が必要なときは、適宜協調して情報の交換を行う。ARTISTE エージェント間での協調の際にはエージェントを探索する必要があるが、既存の P2P システムなどで提案されているエージェント探索のための手法、すなわち、集中型データベース、探索メッセージのフラッディング、DHT(Distributed Hash Table) (Stoica, Morris, Karger, Kaashoek, & Balakrishnan, 2001), (Zhao, Huang, Stribling, Rhea, Joseph, & Kubiatowicz, 2004), (Rowstron & Druschel, 2001)などの方法を利用するものとする。

3.5.2 ARTISTE による組織化支援

本章では ARTISTE を用いてマルチエージェントシステムのエージェントを組織化する手順を説明する。ARTISTE によるエージェントの組織化は次の 3 ステップに分けられる。

- ① ARTISTE による情報収集
- ② マルチエージェントシステムによる探索条件の送信
- ③ ARTISTE によるマルチエージェントシステムエージェントの探索

各ステップ毎の設計方針や機能要件を以下で述べる。

3.5.2.1 ARTISTE による情報収集

表 3-1 に示した通り、ARTISTE による組織化支援のためには対象とするマルチエージェントシステムのエージェント情報およびエージェントが配置された AS の局所的な情報および複数の AS を含む広域ネットワーク情報を収集する必要がある。それぞれの

情報の内容や具体的な収集手順を述べる。

3.5.2.2 エージェント情報

エージェントの状態や能力を表現する情報はマルチエージェントシステムによって異なるが、それらをすべて ARTISTE が収集すると ARTISTE が個々のマルチエージェントシステムに関する情報を持つ必要があり、ARTISTE の汎用性が失われる。

そこで、ARTISTE ではマルチエージェントシステムのエージェントが持つ情報のうち、複数のマルチエージェントシステムで共通的に利用される情報のみを収集する。具体的には、IP アドレス、接続先ポートなど静的な情報に加えて、各エージェントの動作状態を収集する。前者は ARTISTE エージェントに設定として与え、後者はマルチエージェントシステムエージェントと定期的に通信して取得する。

エージェントが実行できるタスクの種類などマルチエージェントシステムに依存する情報は ARTISTE の収集対象には含めない。このような問題依存の情報により協調相手のエージェントを選択したい場合、まず ARTISTE を用いて汎用的な条件により協調相手の候補を絞り込み、選ばれた候補の中から個々のマルチエージェントシステムが実際の協調相手を決定すればよい。

3.5.2.3 広域ネットワーク情報

個々の ARTISTE エージェントは自分の配置先 AS から観測できる局所的なネットワーク情報を、配置先 AS 上の BGP ルータ²から収集する。具体的には、ARTISTE エージェントがルータに接続し、適切なコマンドを実行することで AS-path 情報などのネットワーク情報を取得する。接続先のルータのアドレスやログインするために必要な認証情報は ARTISTE エージェントにあらかじめ与えておくものとする。

広域ネットワーク管理のための MAS におけるエージェント協調では、協調エージェントのネットワーク上の位置に大きな意味がある。例えば、AS 間の経路情報の診断では、自分の経路の状態を調べるため経路情報が実際に伝搬される別の AS のエージェントと協調する必要がある。また、検討対象とした動画配信では、クライアントからネットワーク上の近い位置にいるエージェントや広帯域のリンクを持つ場所に配置されたエージェントを優先的に配信に用いる。このようなエージェントを見つけるためには、局所的な視点だけでなく、広域の視点から見た情報が必要である。

このために ARTISTE では、エージェント間の情報交換によって他のネットワーク情報を獲得し、広域にわたるエージェント探索を可能とする。具体的な方法は後に例を使

² BGP を用いて他 AS と経路情報を交換しているルータ

って説明する．なお，エージェントの IP アドレスやネットワーク prefix のような局所情報は ARTISTE は AS 内から容易に収集できる．

次に，個々のエージェントが情報交換して実際に広域ネットワーク情報を獲得する方法を，3.4.1 と同様に AS 間トポロジを収集する場合を例にして示す．図 3-7 で示した状況下では AS_y 上のエージェントは AS_x と AS_y 間の部分的なトポロジしか観測できないことを 3.4.1 で説明した．ここで，図 3-10 に示すように AS_y からの経路情報が AS_4, AS_3 経由で AS_x に伝搬されている場合を考える．この場合も AS_x 上のエージェントは AS_2 の存在が分からず部分的なトポロジしか観測できない．このとき， AS_y から経路情報が AS_x に広報されているという条件を追加しても AS_y が観測できるトポロジには全く影響がないことに注意する．

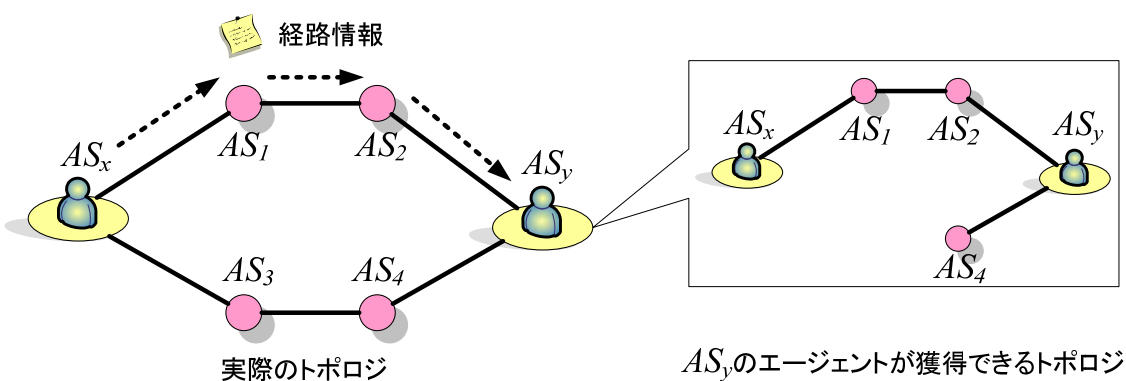


図 3-10 エージェントによる部分的なネットワーク情報の観測

ARTISTE ではこのような場合にエージェントが観測した情報を交換することで，より正確な広域ネットワーク情報を取得することを可能にしている．このトポロジの例では， AS_x と AS_y が観測したトポロジ情報を互いに交換する．それにより， AS_x では AS_2 ， AS_y では AS_3 を含む完全な近傍のトポロジを得ることができ，局所的な観測では得られない広域ネットワーク情報を獲得できる．

エージェント間の情報交換により他 AS の情報を獲得する手法はトポロジ情報以外の情報にも適用できる．例えば，AS におけるトラフィック量など外部 AS からは観測不可能な情報も，当該 AS 上のエージェントが観測した情報を交換することにより容易に得られる．

3.5.3 マルチエージェントシステムによる探索条件の送信

ARTISTE とマルチエージェントシステム間のインターフェースである探索条件につ

いて説明する. ここでの探索条件とはマルチエージェントシステムのエージェントの協調相手のエージェントが持つべき状態やネットワーク上の位置に関する情報を形式的に表現したものである. この探索条件はマルチエージェントシステムでの問題に依存するためマルチエージェントシステムで生成する. 作成方法や手順については各マルチエージェントシステム で自由に決定してよく, ARTISTE は関知しない. ARTISTE で用いる探索条件の一般的な形式を以下に示す.

```

query ::= query <condition>
condition ::= <attribute> | (and|or <condition>+)
attribute ::= <ip>|<asnum>|<neighbor>|
               <prefix>|<status>
ip ::= ipaddress <IP アドレス>
asnum ::= asnumber <AS 番号>
neighbor ::= neighbor <AS 番号> <ホップ数>
prefix ::= prefix <プレフィックス> <マスク長>
status ::= status <"in">|<"out">

```

各 attribute 節の意味は表 3-2 に示す.

表 3-2 探索条件に用いる attribute 節

名前	意味
ipaddress	指定された IP アドレスを持つエージェント
prefix	指定 prefix 上に配置されているエージェント(一つめのパラメータがネットワークアドレス, 二つめがマスク長を示す)
status	指定した status を持つエージェント(動作中であれば in, 停止していれば out)
neighbor	指定 AS から指定距離 (=ホップ数) 以内で接続された AS 上に配置されたエージェント(一つめのパラメータが AS 番号, 二つめが距離を示す)
asnumber	指定 AS 上に配置されたエージェント

この記法に従うと, 例えば (query (and (status in) (neighbor AS1 2)))

という探索条件は AS1 から 2 ホップ以内の AS に配置されていて、かつ、動作中のエージェントを探索することを意味する。

なお、表 3-2 には本稿で対象とする広域ネットワーク管理において頻繁に利用されるもののみを挙げている。これら以外にも適用する分野に応じて `attribute` 節を拡張することで、広域ネットワーク管理だけでなく AS 内ネットワークや LAN 管理といった異なる場面でも ARTISTE を適用可能になる。

3.5.4 エージェントの探索手順

本章では ARTISTE がマルチエージェントシステムから送られた条件に合致するエージェントを探索する手順を述べる。以下では、次の条件の元で ARTISTE およびマルチエージェントシステムが動作していることを前提とする。

- AS 上にはマルチエージェントシステムおよび ARTISTE エージェントが各 1 つずつ動作し、同一 AS 内でのエージェント間およびエージェントと BGP ルータ間の通信に必要な情報は各エージェントに予め与えられている。
- すべての ARTISTE エージェントの情報を登録した集中型データベース(以下、DB)があるものとし、ARTISTE エージェント間での協調の際には本 DB を使う。
- 前記 DB において、各 ARTISTE エージェントは AS 番号によって一意に特定できるものとする。

探索手順を以下に示す。

① 起動および待機

各 ARTISTE エージェントは起動時に設定として与えられた情報を読み込み、マルチエージェントシステムからの探索条件の待機状態に入る。マルチエージェントシステムエージェントの動作状況やネットワーク状態など変動する情報の収集は待機中でも定期的に繰り返す。

② マルチエージェントシステムから探索条件の受信

マルチエージェントシステムエージェントはエージェント協調が必要になった場合

に、同一 AS 上の ARTISTE エージェントに協調すべきエージェントの探索条件を送る。

③ 探索条件の解析

マルチエージェントシステムから探索条件を受信した ARTISTE エージェントは送られた探索条件を解析し、探索条件に含まれる `asnumber` 節や `prefix,ipaddress` 節の値を取得する。これらの値が自分の配置先と一致しない場合は他 AS のエージェントとの協調が必要と判断される。このとき、条件に `neighbor` 節が含まれている場合はステップ④に進む。含まれていない場合はステップ⑤に進む。

④ neighbor 節の処理

`neighbor` 節は直接ネットワークの位置を示していないが、3.4.1 や 3.4.2.2 での議論から分かるとおりに意味的には特定の AS の集合を表している。すなわち、`neighbor` 節は `asnumber` 節の集合に変換することが可能である。図 3-10 で示したネットワークを例に説明する。(query (neighbor ASx 1)) という探索条件を例にすると、この `neighbor` 節の表す「AS_xから 1 ホップで接続された AS」は図より AS₁, AS₃ である。よって、この探索条件は (query (or (asnumber AS1) (asnumber AS3))) と等価であることが分かる。変換の手順を以下に示す。

- A) `neighbor` 節の第一引数の AS 上の ARTISTE エージェントを集中 DB を使って検索し、結果の ARTISTE エージェントに対して `neighbor` 節が表す AS の集合を問い合わせる。
- B) 問い合わせを受けた ARTISTE エージェントでは自分の配置先 AS の BGP ルータから AS-path 情報を取得し、それを解析して合致する AS の集合を求めて返答する。
- C) 返答を受けた ARTISTE エージェントでは、結果を元に `neighbor` 節を `asnumber` 節の集合に変換した新たな探索条件を作る。

⑤ ARTISTE エージェントの探索

ステップ③および④で得た `asnumber` 節の値をキーにして DB を検索し、当該 AS に配置された ARTISTE エージェントの IP アドレスやポート番号など通信に必要な情報を取得する。

⑥ 探索条件の転送

ステップ⑤で得られた ARTISTE エージェント(群)に対して探索条件を転送する。

⑦ 転送先での検査

前ステップにおいて探索条件を転送された ARTISTE エージェントは、同一 AS 上のマルチエージェントシステムエージェントが送られた条件に合致するかを検査する。合致した場合はマルチエージェントシステムエージェントの IP アドレス、合致しなかった場合はその旨を示すメッセージを転送元の ARTISTE エージェントに返送する。

⑧ 検査結果の統合およびマルチエージェントシステムへの返答

転送元の ARTISTE エージェントでは転送先からの検査結果がすべて集まったら、それらをまとめて問い合わせ元のマルチエージェントシステムエージェントに対して探索結果として返答する。

手順の概略を図 3-11 に示す。

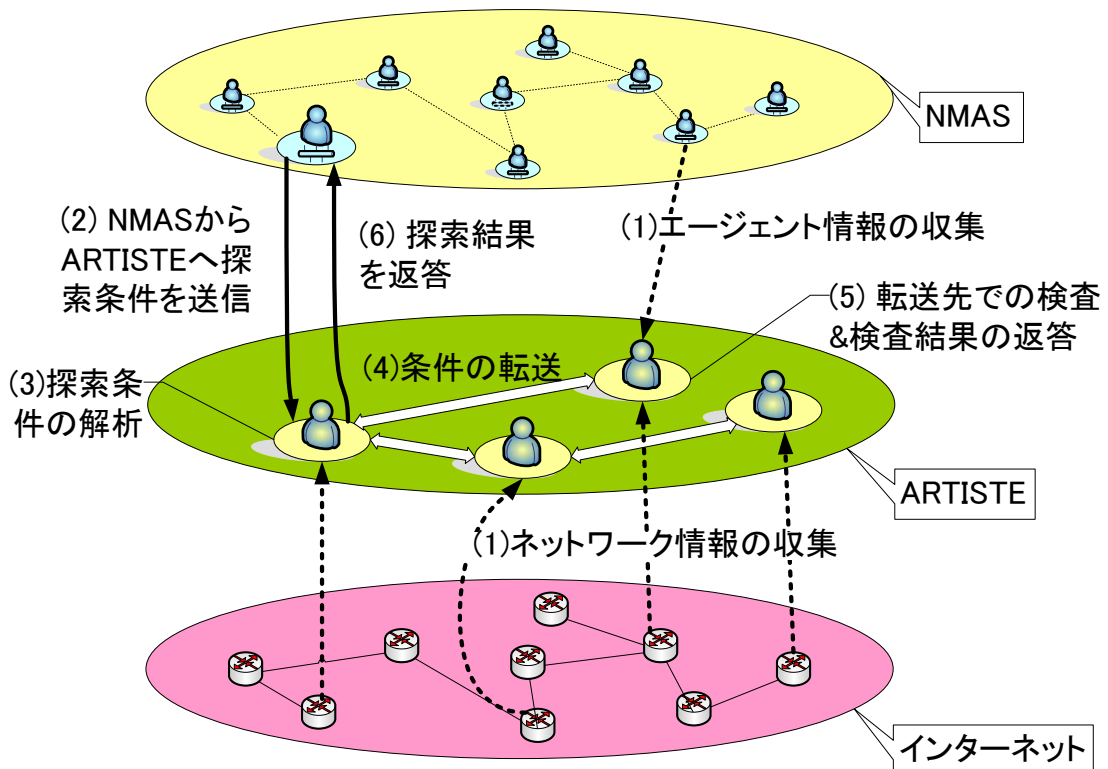


図 3-11 エージェントの探索手順

3.6 提案方式の評価

3.6.1 実験環境

3.6.1.1 システム構成

3.5 での検討を元に ARTISTE システムを実装した。そして、AS 間経路情報診断システム ENCORE を対象のマルチエージェントシステムとし、ENCORE(ARTISTE に対応させるために一部改造)と ARTISTE が連携動作する実験システムを構成した。実験システムの構成を図 3-12 に示す。

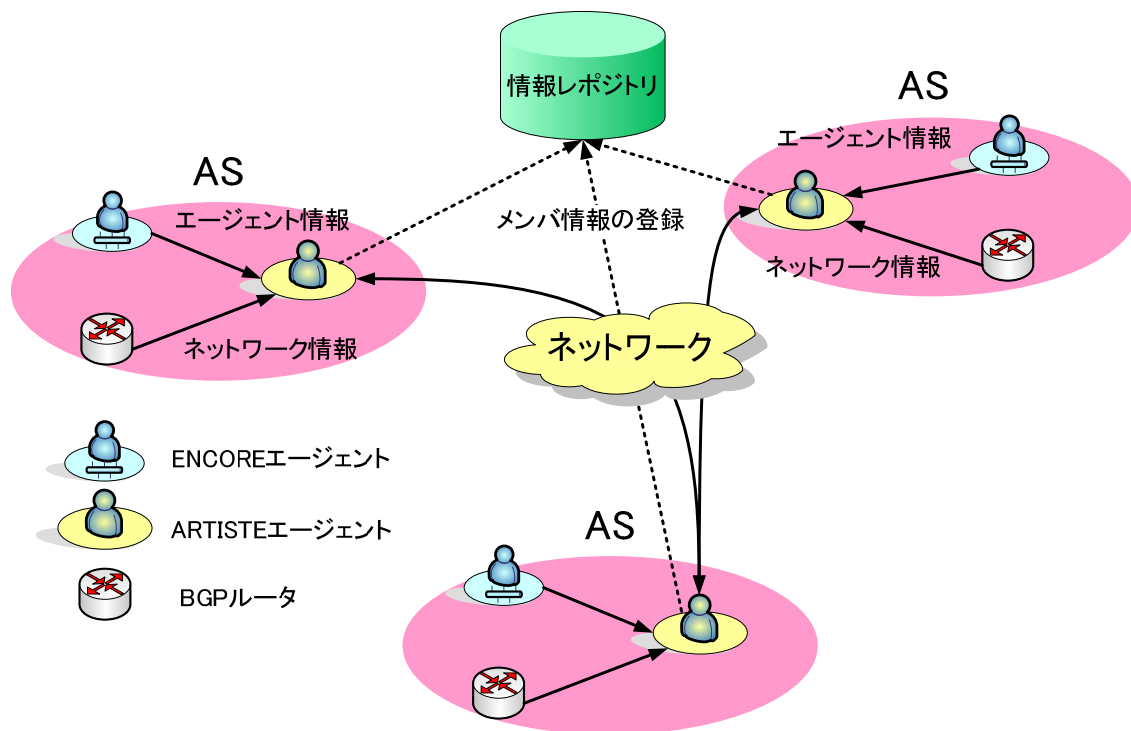


図 3-12 システム構成図

本システムは以下の 4 つの構成要素から成る。

- ① ARTISTE エージェント

商用の Lisp で実装され FreeBSD 6.1R 上で動作する。実装に当たっては ARTISTE エージェント間の通信処理は XML-RPC [XML-RPC Home page]を利用して実装した他、BGP ルータとの通信には telnet を用いるなど外部ソフトウェアとの連携部分はできるだけ汎用的な技術を用いるように留意した。ARTISTE エージェントのソフトウェア構成図を図 3-13 に示す。

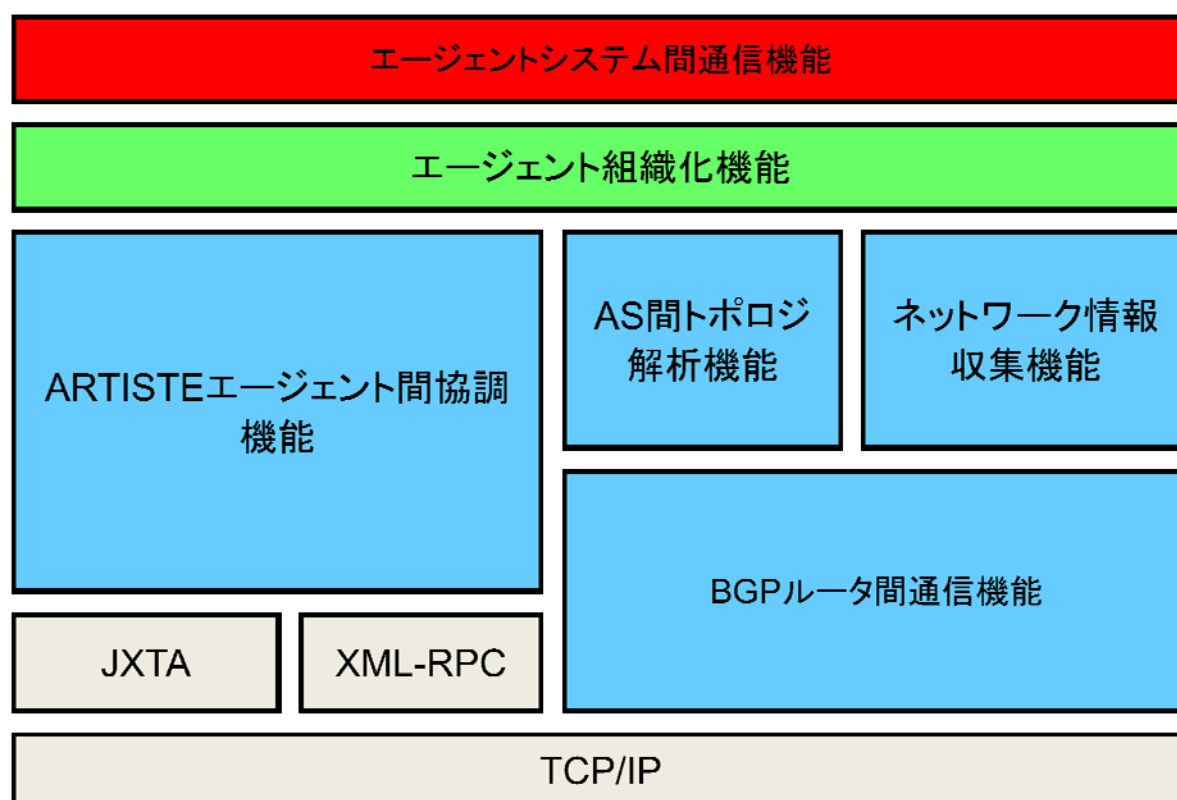


図 3-13 ARTISTE エージェントソフトウェア構成図

② 情報レポジトリ

全 ARTISTE エージェントの情報(IP アドレスや配置先 AS など)が登録されている。ARTISTE エージェント間で協調を行う際は、AS 番号などを検索キーとして ARTISTE エージェントを探索し、IP アドレスやポート番号などの必要な情報を取得した上で通信を行う。実験システムでは本レポジトリは LDAP を用いて実装した。

③ BGP ルータ

各 AS に配置されているルータで BGP により他の AS と経路情報を交換する。実験システムでは ARTISTE エージェントがルータに telnet でログインして適切なコマンドを実行することでネットワーク情報を取得できるように実装した。例えば、AS 間

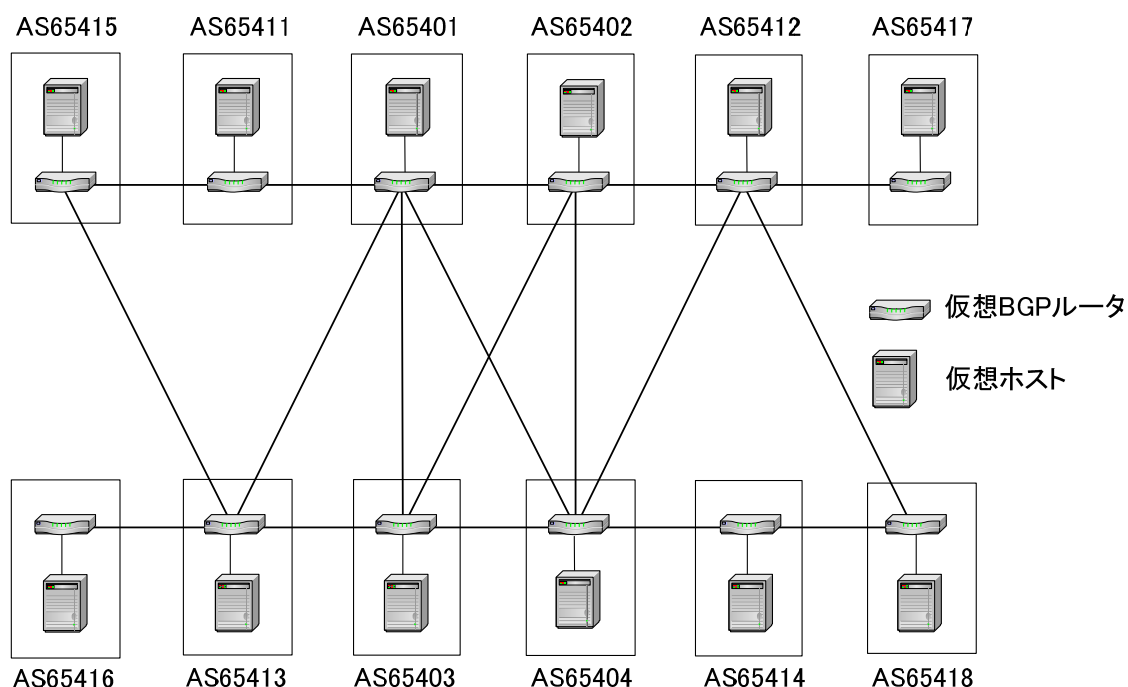
トポロジを使ったエージェントの探索を行う際に利用する AS-path 情報を取得するためには、管理者としてログインした後に ``show ip bgp``³ というコマンドを実行すれば AS-path 情報を含む BGP でやりとりされる情報が取得できる。

④ ENCORE エージェント

ARTISTE エージェントと同様に、商用の Lisp で実装され FreeBSD 6.1R 上で動作する。 ENCORE エージェント間の通信処理も同じく XML-RPC を用いて実装した。 ENCORE エージェントと ARTISTE エージェント間の通信も XML-RPC を利用して実装を行った。 XML-RPC は多数の OS や言語処理系に対応した実装が提供されているため、 ARTISTE とマルチエージェントシステム間の通信処理の汎用性を高めることができる。

3.6.1.2 実験ネットワーク

構築した実験システムを動作させるために市販の仮想マシンソフトウェアおよび同ソフトウェアの仮想ネットワーク機能を使って 12 個のプライベート AS からなる実験用仮想ネットワーク(以下、実験ネットワーク)を構築した。 実験ネットワークの構成を図 3-14 に示す。



³ BGP ルータが Cisco 社製である場合

図 3-14 実験ネットワーク構成図

実験ネットワーク上の各 AS は 2 つの仮想ホストから構成されている。1 つのホスト上では ARTISTE エージェントおよび ENCORE エージェントが 1 つずつ動作し、もう一つのホストはフリーのルータソフトウェア quagga (Quagga Routing Suite) を動作させることで BGP ルータとして機能させている。また、この実験ネットワークはインターネットとは接続していないが、実際のインターネットへの適用時の性能を検証するために実際のインターネットから取得したフルルート(約 16 万の経路情報)を BGP ルータ間で流通させている。

この実験ネットワークはラックマウント型の汎用 PC 4 台の上で動作している。これらの PC のスペックは 2 台が Pentium4 3GHz, メモリ 2GB で、残りの 2 台はデュアルの Xeon 2GHz, メモリ 4GB という構成である。また、仮想ソフトウェアが動作するホスト OS はすべての PC で Windows Server 2003 を用いており、仮想ホストおよび仮想 BGP ルータが動作するゲスト OS はすべて FreeBSD 4.11R を利用している。



図 3-15 実験ネットワーク構築用 PC

3.6.2 実験結果

作成した実験システムを使って ARTISTE の有効性の基本評価を行った。以下にその結果について述べる。

3.6.2.1 ARTISTE による組織化の有効性

ENCORE などのマルチエージェントシステムでエージェント間の協調を行う場合、

ARTISTE を使わないとすると、それぞれのエージェントでは協調相手のエージェントの IP アドレス、役割などの情報を協調の前に予め準備しておく必要がある。さらに、これらの情報はネットワークの状態や協調相手の変化に応じて随時変更しなければならない。ARTISTE を用いれば協調相手のエージェントを動的に探索することができるようになるため、協調相手エージェントの事前設定やその維持管理が不要になるという効果が得られる。これらの効果が実際に得られることを実験システムを使って検証した。

検証では、図 3-14 上の AS65416 上のエージェントが「AS65412 から 2 ホップ以内に接続された AS 上のエージェント(群)」と協調したいという要求を持っている場合を例にした。図 3-14 よりこの要求を満たすエージェントは AS65401, 65402, 65403, 65404, 65414, 65417, 65418 上で動作するエージェントであることが分かる。実際のマルチエージェントシステムでこれらを指定する際の設定を ENCORE と寺内らのストリーム配信システムの場合を例にして表 3-3 に示す。

表 3-3 協調エージェントに関する設定(ARTISTE 利用無)

ENCORE	寺内らのストリーム配信システム
(investigation . ("a.b.c.d:e" :asnumber 65401) ("f.g.h.i:j" :asnumber 65402))	<neighbors> <neighbor> <address>"a.b.c.d:e"</address> </neighbor> </neighbors>

これに対して、ARTISTE を用いる場合は、マルチエージェントシステムから前述の要求条件を 3.5.3 に示した記法に従って記述し ARTISTE に送信すればよいので表 3-4 に示した設定を行うのみでよい。この例のように、システムが全く異なっても、要求条件が同じであれば同一の処理でエージェントの組織化ができることは ARTISTE の汎用性、再利用性が十分に高いことを示している。

表 3-4 協調エージェントに関する設定(ARTISTE 利用時)

	ARTISTE 利用時
ENCORE	(query (neighbor 65412 2))
寺内らのストリーム配信システム	

実際に、本実験において協調エージェントに関する静的な設定を ENCORE から削除

し表 3-4 で示した条件を使って ARTISTE を使った探索を行うように変更したところ、ARTISTE から ENCORE への返答として AS65401, 65402,65403, 65404, 65414, 65417, 65418 上で動作するエージェントの IP アドレスが取得できることを確認した。

さらに、ネットワーク状態の変化に ARTISTE の組織化が追従できることを確認するため、AS65412 と AS65404 間と AS65404 と AS65402 間の接続を切断した後で同じ探索条件を用いてエージェント探索を行った。この場合は AS65401, 65402, 65403,65414, 65417, 65418 上のエージェントが条件に合致することが図 3-14 より分かる。このような変更があった場合、ARTISTE を使っていなければマルチエージェントシステムの要求が変化していなくても設定の変更が必要になる。これに対して ARTISTE を用いれば要求が同じであれば表 3-4 の設定を変える必要はないので、従来の設定変更コストが削減できる。実際に ARTISTE を用いて探索した結果、前述の AS 上のエージェントが正しく返答されることを確認した。

以上より、ARTISTE をマルチエージェントシステムと併せて利用することによって協調エージェントに関する設定の維持管理をしなくともネットワーク状態の変化に追従して適切な協調エージェントを動的に選択できることが分かり、ARTISTE による組織化支援がマルチエージェントシステムを運用する上で有効であることを明らかにした。

3.6.2.2 組織化機構の分離効果

ARTISTE を用いることによるもう一つの利点は、マルチエージェントシステム構築の際に経路診断などマルチエージェントシステム本来の問題解決に関わる処理と組織化の処理を分離できることである。例えば、マルチエージェントシステムが前述の「AS65412 から 2 ホップ以内に接続された AS 上のエージェント(群)」と協調する場合を考えると、ARTISTE を用いなければ問題解決のための協調を行う前に、

- AS65412 から観測した AS-path 情報の取得
- AS-path を解析して 65412 から 2 ホップ以内の AS の取得
- 前述の AS 上に配置されたマルチエージェントシステムエージェントの探索

という組織化の処理を行う必要がある。そのため、タスクのためのコードと組織化のためのコードが同一システム内に混在することになる。ARTISTE を導入すれば組織化のコードをマルチエージェントシステムから分離、省略することができるため、システムのコード量の削減効果が得られる。この効果を確認するために、作成した ARTISTE

システムおよびマルチエージェントシステムの一つである ENCORE の開発規模を調査した。その結果を表 3-5 に示す。

表 3-5 ARTISTE およびマルチエージェントシステムの開発規模

	組織化機構を分離	組織化機構を統合
ENCORE	約 39K	約 42.5K(=39K+3.5K)
ARTISTE	約 3.5K	

ARTISTE を導入せずマルチエージェントシステムが独自に組織化機構を持つとすると、マルチエージェントシステムにはそれぞれ本来の問題解決のコードに加えて組織化に関するコードを含めなければならない。このコード量は ARTISTE の規模に相当すると考えられるため、その場合は本来の問題解決のコードのみの場合に加えて約 3.5K ステップのコードが増えることが表 3-5 から分かる。逆に言うと、組織化機構を分離することによって、約 3.5K ステップのコードが削減できることになる。例えば ENCORE の場合を考えると、組織化機構を分離した場合は分離しない場合と比べて $3.5K/42K =$ 約 8.3%の工数削減が実現できたことになる。

ARTISTE は複数のシステムに対して適用できる再利用性もあるため、同様の探索機能を必要とするマルチエージェントシステムを開発する場合にも ARTISTE を導入することで同規模の開発規模削減効果が期待できる。さらに、組織化処理の分離を行うことはコード全体の見通しを向上させマルチエージェントシステムの構成の複雑化およびそれに伴う実行速度や保守性の低下を避けるという効果も期待できる。

以上より、ARTISTE の利用した組織化機能の分離はマルチエージェントシステムの構築する上での工数削減や構造の単純化による動作品質の向上といった有益な効果をもたらすものと言える。

3.6.2.3 性能

次に、ARTISTE によるエージェント探索のための応答時間の計測結果を表 3-6 に示す。

表 3-6 ARTISTE の応答時間

ホップ数(探索範囲)	1	2	3	4	5
応答時間(ms)	743	856	1967	6504	7734
ホップ数(探索範囲)	6	7	8	9	

応答時間(ms)	7732	7810	7758	7739	
----------	------	------	------	------	--

ARTISTE の応答時間の内訳を調べると、本実験の ARTISTE では集中型のレポジトリを利用していることもあり、他 ARTISTE エージェントの探索や通信には殆ど通信コストはかからず、ENCORE エージェントや BGP ルータとの通信も通信量が多くないため非常に短い時間で完了していることが分かる。この時間は ARTISTE エージェントの数が増えても殆ど変化しないものと考えられる。しかし、AS-path 情報から AS 間トポロジを作る過程およびそれを利用したエージェントの探索処理が処理時間の大半を占めていることが明確となった。

そこで、実際に実験システム上で AS-path 情報から AS 間トポロジを作成しエージェントの探索範囲、すなわち AS ホップ数を変化させてエージェントを探索した場合の探索時間を測定した。この実験では、実験システムに流通するフルルートにより AS 間トポロジを作っているため、ここでの測定結果は実際のインターネットでの測定値に近い。

表 3-6 から分かるように 3 ホップまでの範囲の探索では応答時間は 2 秒以内に収まっているが、それより広い範囲でエージェントを探索すると応答時間が大幅に増加する。よく知られているとおりインターネットは「べき法則」に従うネットワーク構造のため (Faloutsos, Faloutsos, & Faloutsos, 1999)、ホップ数の増加に伴うこの応答時間の急激な増加は合理的である。しかし、筆者らが実際のインターネットのトポロジを調査した結果によると、任意に選んだ AS から 4 ホップで接続される AS 数は殆どの場合約 10000 以上であることが分かっており、このような広大な範囲でエージェントを探索することは実際の適用では起きないと考えてよい。そのため、ARTISTE の応答時間は現実のネットワークに適用した場合でも問題ない水準と言える。

一方、(gavalas, Greenwood, Ghanbari, & O'Mahony, 2002)のように移動エージェントを利用したマルチエージェントシステムの場合、エージェント移動に伴って各 ARTISTE エージェントで収集管理する情報を変化させる必要がある。現状のフレームワークではそのためのコストを考慮していないため、エージェントの移動範囲が大きい場合、十分な性能が得られないことも考えられる。このような場合の対応については今後の課題である。

3.6.3 エージェントの配置方法

3.6.3.1 概要

ARTISTE では ARTISTE エージェントの協調によって、対象 MAS から与えられた条件に合致する対象 MAS 中のエージェントの探索を行うが、このとき、どの AS にどれだけの MAS エージェントがあるかを ARTISTE は観測によって知る必要がある。そのため、ARTISTE エージェントの集合はネットワークおよび MAS エージェントの全体の情報を把握していないと、要求に合致する MAS エージェント全体あるいは一部を発見することができない可能性がある。また個々の ARTISTE エージェントはネットワークおよび対象 MAS に関して局所的な情報しか獲得できないため、ARTISTE エージェントの集合により収集できる全体の情報は、ARTISTE エージェントの数と配置される場所に強く影響される。

前記のように MAS エージェントが AS 単位に配置されている場合、ARTISTE エージェントを配置する方法として最も簡便な方法はすべての AS 毎に ARTISTE エージェントを配置する方法である。各 ARTISTE エージェントが所属 AS から観測できるネットワーク情報および MAS エージェントの情報を管理するとすれば、原理的にはネットワークおよび対象 MAS 全体の情報を ARTISTE で管理できることになる。

ただし個々の AS は独立した組織が独自のポリシーで運営していることを考慮すると、全 AS に ARTISTE エージェントを配置するのではなく、一部の AS にのみ ARTISTE エージェントが配置される状況の方が現実的である。しかし、この場合には取得できるネットワークおよび対象 MAS の情報量が全 AS に配置できる場合と比べると減少するため、ARTISTE による組織化の品質が低下するという問題が生じる。本章では、ARTISTE やその他のマルチエージェントシステムを効果的に運用するためのエージェントの配置方法に関して考察を行う。

3.6.3.2 検証方法

本考察では、ARTISTE エージェントの配置に際して AS 間のトポロジ構造に着目する。インターネットの AS とその接続数(次数) k の間にはべき法則 $P(k) \propto k^{-\gamma}$ が成立すること (Faloutsos, Faloutsos, & Faloutsos, 1999)が知られている。つまり、大半の AS は少数の AS との接続しか持たないが、ごく一部の AS は多くの AS と接続している。後者の AS からは前者の AS に比べると広い範囲のネットワークが観測できるため、後者の AS に優先的に ARTISTE エージェントを配置することで、少数の ARTISTE エージェントで広範囲のネットワーク情報が取得できることが期待できる。

また、MAS エージェントが AS 毎に配置されているのであれば直接接続された AS が多い AS からは 1 ホップの通信で多くの MAS エージェントの情報が取得できるため、MAS エージェント情報を収集する際にも有利である。通信コスト以外にも、2 ホップ以上離れた AS 同士は未知である場合も多いため、2 ホップ以上先の AS (にいるエージェント)に関する情報を取得することは容易ではないと考えられる。よって、1 ホップで

多くの情報が取得できる AS は MAS エージェントの情報の収集に有利な場所であると言える。

以上の検討を元にして、ARTISTE において、

- ARTISTE エージェントはどこ(どの AS)に配置すべきか？
- 配置する ARTISTE エージェントの数はどのくらいあれば十分か？

という 2 つの問題に対して検証を行うこととした。これらの課題に対する定量的な検証を行うために、実際にインターネット上から取得した全経路情報（フルルート）を元に作成した AS トポロジを使って ARTISTE エージェントの配置と収集できる情報の範囲の変化をシミュレートする実験を行った。実験方法および結果の詳細を 3.6.3.3 および 3.6.3.4 章以降で述べる。

3.6.3.3 実験手順

- ① BGP ルータからフルルートを取得し、AS トポロジを作る。BGP で伝搬される経路情報にはそれぞれが伝搬する際に経由したすべての AS が AS-path 情報として次に示すように記載されている。

Address	Hash	Refcount	Metric	Path
0x6320abcd	0	8	0	AS1 AS3 AS6 AS2
0x6213efgh	0	3	0	AS1 AS3 AS6 AS4
0x614Fijkl	0	1	0	AS1 AS3 AS6 AS5 AS9 AS8
0x63F0mnop	0	1	0	AS1 AS3 AS6 AS7 AS9 AS10

この AS-path 情報を使って AS 間の接続を示すトポロジを作成する。例えば、この例で示した AS-path 情報をすべて統合すると図 3-16 に示すトポロジが得られる。

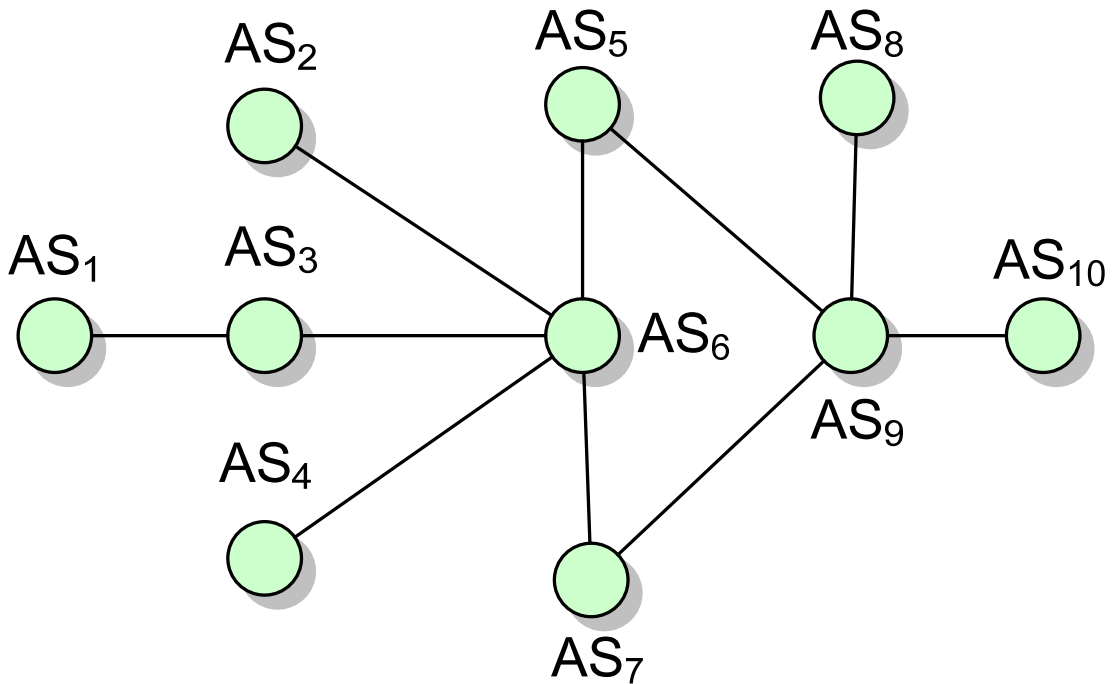


図 3-16 AS トポロジの例

- ② AS トポロジを使って、各 AS 毎に他 AS との接続数を調べる。例えば、図 3-16 からは AS₁ の接続数が 1 で AS₆ の接続数が 5 であることが分かる。
- ③ 次に全 AS の中から n 個の AS を選び、それらの AS に直接接続された近傍 AS の集合を求める。本実験での AS の選び方は 3.6.3.4 章に述べる。
- ④ 前ステップで得られる集合の要素数は、「選択した n 個の AS に ARTISTE エージェントを配置し、それらが相互に収集した情報を交換するとき、システム全体としてどのくらいの範囲のネットワークおよび MAS の情報を把握することができるか」という意味を持つ。そこで、提案する配置方法の評価に次の式で定義される「把握率」という値を用いる。

$$\text{把握率} = \frac{n\text{個のASの1ホップ近傍のAS数}}{\text{トポロジ中の全AS数}}$$

- ⑤ n および AS の選び方によって「把握率」がどのように変化するか調べる。

3.6.3.4 実験結果

本実験では世界中の4カ所のASにおいて取得したフルルートを使った. それぞれのフルルートから作成したASトポロジにおけるASの接続数の分布を表3-7および図3-17に示す.

表 3-7 ASの接続数の分布

接続数	1	2-10	11-100	101-1000	1000以上
トポロジ 1	13957	5132	372	30	3
トポロジ 2	14783	5050	382	33	3
トポロジ 3	14563	5246	395	31	4
トポロジ 4	14627	5217	382	29	2

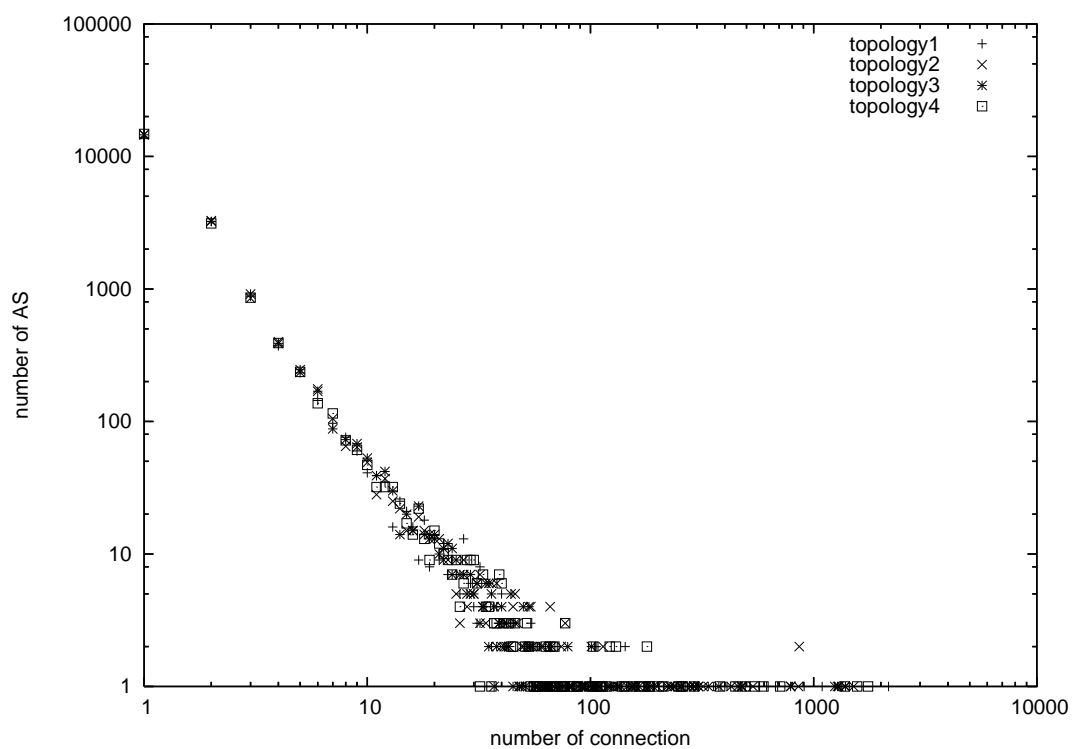


図 3-17 ASの接続数の分布

次に, ASの選び方と選ぶ個数による把握数の変化の様子を図 $\ref{fig:graph}$ に示す. ここでは, 以下の方法によりASを選択した場合のデータを示した.

- [data1] 接続数の多いASから順番に選択
- [data2] 接続数が2以上であるASからランダムに選択
- [data3] 全ASからランダムに選択

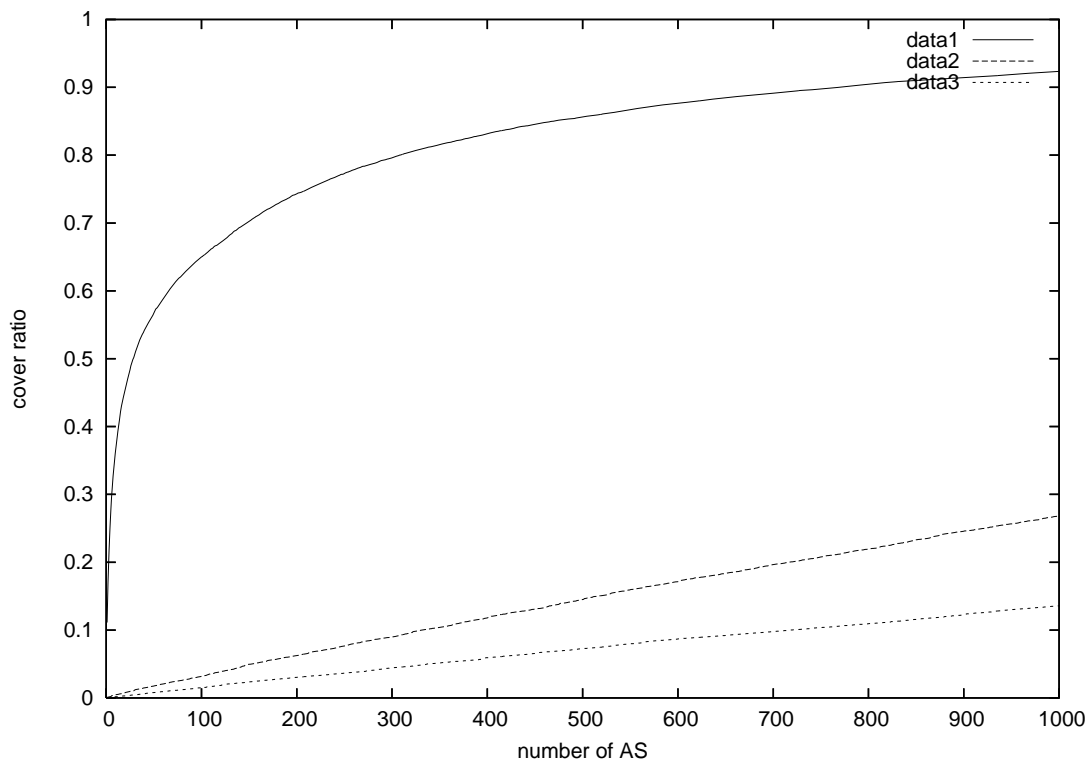


図 3-18 把握率の変化

3.6.3.5 考察

図 3-18 より，トポロジを考慮して優先的に ARTISTE エージェントを置く場合は，ランダムに ARTISTE エージェントを配置する場合と比べて，少ない ARTISTE エージェント数で高い「把握率」を得ることができると分かった．表 3-7 の AS の分布からも分かる通り，大半の AS は接続数が 1 であるため，トポロジを考慮せずランダムに ARTISTE エージェントを配置する AS を選べば接続数 1 の AS に配置される確率が高くなり，エージェントの数を増やしても把握率が伸びない．よって，提案手法のようにトポロジを考慮した配置が必須である．

本稿では，ARTISTE エージェントの配置に伴って，個々の ARTISTE エージェントが管理すべき MAS エージェントの情報が増加するという問題については触れなかった．例えば，今回の実験で使用した AS トポロジ中の AS の最大の接続数は約 2,100 であった．その AS に配置された ARTSITE エージェントは少なくともそれだけの数の MAS エージェントの情報を管理しなければならないとすると大きな処理能力が必要とされる．このように，配置場所によって ARTISTE エージェントに要求される能力にも大きな差が出る．トポロジを利用したサーバの配置と同時に負荷分散を行う方法については (Fukuda, Sato, Akashi, Hirotsu, Kurihara, & Sugawara, 2005)にも提案されているが，

ARTISTE エージェントの配置場所と要求される能力との関連については今後さらなる検討が求められる。

また、接続数の多い AS の間では相互の AS に対する接続が多いため、個々の AS とその近傍をクラスタと見なすとクラスタ間の重なりが多い。この重なりが小さくなるように配置を考慮すればさらに少ないエージェント数で多くの情報を得られる可能性がある。しかし、システムの **robustness** を考慮するとある程度のクラスタの重なり、すなわち、同じ情報を複数のエージェントで管理することが望ましいとも考えられる。この **trade-off** についても今後検討を進める予定である。

3.7 本章の総括

本稿では、マルチエージェントシステムの有効な適用分野であるネットワーク管理の中でも AS 間を跨る広域ネットワーク管理をマルチエージェントシステムで行う場合を対象にして、協調すべきエージェント群を動的に選択することによりマルチエージェントシステムにおけるエージェントの組織化を支援するエージェント組織化支援システム ARTISTE を提案した。

ARTISTE をマルチエージェントシステムとして実現することでネットワークの情報を複数地点から分散観測して情報交換することができ、広域のネットワークトポロジなどの単一地点からの観測では獲得の難しい情報を利用したエージェント選択が可能になることを示した。AS 間経路診断システム ENCORE と ARTISTE が連携動作する実験システムを作成して評価実験を行った結果、ARTISTE の導入によりエージェント協調のための情報を維持管理する必要がなくなりマルチエージェントシステムにおいて実際に運用コストの削減ができたこと、ARTISTE を共通基盤として実現することにより今後マルチエージェントシステムの開発において開発削減効果が期待できること、そして ARTISTE は実ネットワークへの適用に対しても十分な性能で動作することを示した。

以上より、提案方式は広域のインターネット管理に対してマルチエージェントシステムを適用する際に不可欠となる組織化作業を有効に支援することができることが明らかになった。その結果として、提案方式はマルチエージェントシステムの動作に環境に対する柔軟性と効率性をもたらすものであり、マルチエージェントによる適応型ネットワーク管理技術の構成技術であるエージェント組織化支援技術を明確化するものである。

第4章 エージェント間協調動作管理技術

関連論文 【A1】, 【A2】, 【A5】, 【B1】, 【C4】

4.1 本章の位置づけ

本章では, マルチエージェントシステムによるネットワーク管理において不可欠であるエージェント間の協調動作をエージェントの運用者のポリシーに基づいて適切に管理するための技術について明らかにしている. 本章の位置づけを図 4-1 に示す.

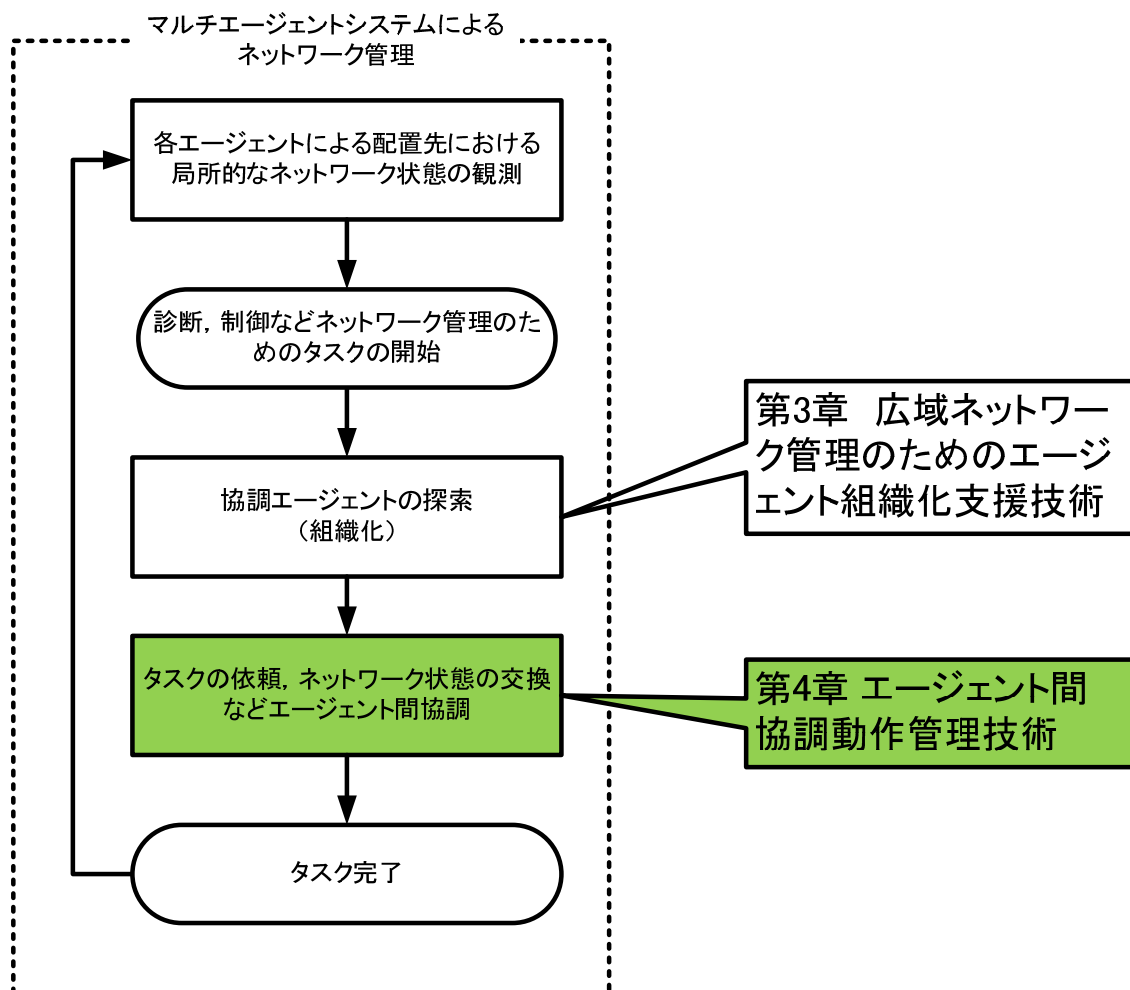


図 4-1 本章の位置づけ

はじめに、提案技術の重要性、必要性を明らかにするために実際の広域ネットワーク管理の作業を行う観点からマルチエージェントシステムのネットワーク管理への適用における問題を整理する。そして、筆者を含む実際のネットワーク管理業務に携わる技術者からのヒアリングなどを検討した結果としてエージェント間の協調動作管理に対する要求条件を整理した後、それらへの解決策となるエージェントのグループ管理機能および信頼度による協調動作管理機能を提案する。それぞれの機能の詳細を説明した後、実際にこれらの機能を実装したマルチエージェントシステムを複数の仮想ASからなる仮想ネットワーク環境で動作させた検証結果を示す。

4.2 研究の背景

インターネットは独立した管理主体により運用・管理される AS (Autonomous System) が相互接続された巨大な分散システムである。個々の管理主体は自分が管理する AS 内のネットワークしか観測制御できないため、複数の AS を跨る広域ネットワークの運用管理では AS 間での協調作業が不可欠である。例えば、インターネットの経路情報は複数の AS を経由しながらネットワーク全体に伝搬されるが、その伝搬過程において AS が任意に変更することができる。そのため、経路情報の転送状況を調査したい場合などは当該経路情報を生成した AS を始めとして当該経路を伝搬したすべての AS の協力が必要である。また、動画など大容量のコンテンツを多数のユーザにインターネットで配信する場合、サーバやクライアントといった終端のノードを制御するだけでは十分でなく、マルチキャストのように転送経路上のルータやネットワークを含めた広域にわたる協調制御が必要である。

このように、広域ネットワークの管理とは複数の AS による分散協調作業として捉えられる。そのため、広域ネットワークの管理を高度化、効率化するには単一システムによる集中型のアプローチでは十分でなく、分散配置された複数のエージェントが協調することでタスクの実行を行うマルチエージェントシステム [E.H.Durfee, V.R.Lesser, D.D.Corkill, 1987]を使ったアプローチが有効と考えられており、実際にマルチエージェントシステムをネットワーク管理に適用した事例も多く報告されている [O.Akashi, A.Terauchi, K.Fukuda, T.Hirotsu, M.Maruyama, T.Sugawara, 2005] [寺内, ほか, 2004] [S.Willmott B.Faltings, 2000] [D.A.Tran, K.A.Hua, S.Sheu, 2003] [M.M.Hefeeda, B.K.Bhargava, D.K.Y.Yau, 2004]。

一般に、マルチエージェントシステムがタスクを実行する際にはエージェント間の協調作業が不可欠である。具体的にエージェント間でどのような協調作業が行われるかは

マルチエージェントシステム毎に異なっているが、通常はエージェント間でのタスクの依頼や情報交換が行われる。広域ネットワークの管理を行うマルチエージェントシステムにおいても、ネットワーク上の複数地点に分散配置されたエージェントが局所的な作業に加えて必要に応じて他のエージェントと協調することによって、他地点で観測されるネットワーク情報を獲得し他のエージェントにネットワークの制御の依頼を行うことでシステムとしてのタスクを実行している。広域ネットワークの管理が複数の AS 間での分散協調作業としてモデル化されることから分かりますとおり、このような情報交換などの協調作業は広域ネットワーク管理においては不可欠なものであり、マルチエージェントシステムを使わずに人手でネットワーク管理作業を行う場合にも必要となる。

著者は、これまでにマルチエージェントシステムによるネットワーク管理システムの開発だけでなくそれらのシステムの適用性について検討してきた。その過程で実際に商用ネットワークを運用管理している複数の ISP とともに開発したシステムの適用実験なども行ってきたが、それらを通して以下の問題があることが分かった。

実際のネットワークの管理の現場においては、ある AS が自 AS のネットワーク情報を他の AS に公開することは顧客のプライバシー保護、特定攻撃の対象になる可能性があるなどの理由から特に注意が払われており、AS 間の情報交換については個々の AS が独自のポリシーを設定して、それに従って情報公開の可否を決定したり公開する情報の内容を制限するなどの運用がなされている。ところが、これまでのマルチエージェントシステムによるネットワーク管理システムでは、協調作業がシステムの実行に不可欠であるため、エージェント間での協調動作に関して制限を設けることは通常行われることはない。そのため、エージェント間の協調作業に関して何ら制限を持たない従来のネットワーク管理システムを実際のネットワーク管理に適用すると、エージェントの配置先のポリシーに反して AS 内のネットワーク情報がエージェントの協調を通じて他の AS に流出する可能性があった。

本章では、この問題を解決してマルチエージェントシステムの実際のネットワーク管理業務への適用を推進するためのエージェント間協調動作管理技術を提案し、さらに実際のネットワーク環境を用いて提案技術の有効性を検証する。

4.3 問題の所在とアプローチ

4.3.1 問題の所在

マルチエージェントシステムや P2P システムなど 1 つのシステム内に複数のプログラム(エージェント or ピア)が包含される分散システムの運用において大きな問題とな

るのがシステム内のプログラムの信頼性である。単一の主体がシステム全体を運用する場合など全プログラムの信頼性が保証されている場合、各プログラムはシステムとしての目的達成に向けた最適な動作を行うことができ、エージェント間の協調動作を含むシステムとしての動作には何ら制約を設ける必要はない理想的な環境と言える。

しかし、これらのシステムを実際のアプリケーションとして利用する場合、システムに含まれるプログラムがすべて信頼できるとは限らず、そのことに起因するさまざまな問題が想定される。そのような問題として例えば以下のようなことが考えられる。

- 悪意のある運用者によるプログラムにより、他の運用者に被害を与える可能性がある。
- 利己的な動作(依頼されたタスクを途中でやめてしまうなど)をするプログラムにより、システム全体の動作が制約を受ける可能性がある。
- 動作が遅い、あるいは不安定なプログラムが含まれることでシステム全体の動作が阻害されたりパフォーマンスが低下したりする。

このような問題を防ぐためにマルチエージェントシステムに関する研究分野においては、エージェント間のコミットメントという概念の導入などさまざまな理論的な研究が行われているが、実際の運用においてプログラムが全体の意図に反する動作を行った場合には対処ができないため、より具体的な対策が求められる。

一方、前記のような信頼できないエージェントがマルチエージェントシステムに含まれている場合に起こりうる問題を広域ネットワーク管理の場合に当てはめると以下のような可能性が考えられる。

- (悪意の有無に関係なく)エージェントによって観測されたネットワーク情報が他の AS に流出する恐れがある。
- ネットワークの制御や診断などのタスクが十分に実行できない可能性がある。

広域ネットワーク管理という作業の特性上、各エージェントを運用するのは AS を管理する必要のある ISP などの業者となるため、悪意を以て他の AS に関する情報を取得するという行為までは想定されないが、ビジネス的に利害が競合する他の ISP に対して上記のような問題が起こる可能性の存在は広域ネットワーク管理のマルチエージェントシステムの適用に際して大きな障害となりうる。特に 1 つめの情報流出の問題は実際に AS を管理する上で特に注意が払われている問題であり、この問題に対する対処は

不可欠である。そこで、本研究では広域ネットワーク管理を行うエージェント間の協調動作の中でも特にエージェント間で行われる情報共有の動作を対象にしてその動作を問題なく実行するため各エージェントの動作をエージェントの運営者の意図に応じて明示的に管理できる仕組みの検討を行う。

4.3.2 アプローチ

エージェント間で行われる協調動作とはすなわちエージェントの運営者間の協調動作を反映したものであるから、エージェント間の動作を管理する方法を検討するためにはエージェントの運営者間の協調動作についてまず分析する必要がある。

ISP などエージェントの運営者がネットワーク管理作業の際に他の ISP と協調する場合には作業の内容やネットワークの状態の他、相手の ISP が誰かということがその行動に大きく影響する。例えば、自分と協力関係にある ISP であれば相手に十分な情報を与えるだけでなく相手からも十分な情報や作業が得られることが期待できる。逆に、利害の対立する ISP が相手の場合はネットワークの動作に必要な最低限の協力をしないということも充分考えられる。このように、ISP 間での実際の協調が行われる場合には協調相手との関係がその動作を制御するために重要な要素であることが分かる。

そこで本研究では、エージェントの運営者である ISP 間の信頼関係をマルチエージェントシステム内のエージェント間の信頼関係としてモデル化する方法およびそのモデルに基づいて個々のエージェント間が適切な協調動作を相手によって選択する仕組みを導入することでマルチエージェントシステムによる情報流出の問題を解決する。これらの方法の元となるエージェント間の信頼関係として必要なモデルを抽出するために実際にいくつかの ISP から意見をヒアリングした結果、ISP 間で行う情報共有の制限方法は以下の 2 つに分けられることが分かった。

- ① 信頼できる相手によって構成される特定グループ内で情報共有
- ② 信頼度の異なるエージェント毎に異なる方式や内容の情報共有

それぞれの要求条件について詳細に述べる。

信頼できる相手によって構成される特定グループ内で情報共有

AS 間での通信障害や情報流出は自分や顧客だけでなくインターネット全体へも影響しうることから、AS の運営方針としてピアリングはもちろんトラフィック交換も十分に信頼できる相手としか行わないというケースも多い。その他、同一の企業グループに

属する ISP であれば相手が信頼できるので情報交換には制限を設けないなどということも実際によく行われていることである(図 4-2 参照). つまり, AS が他と協調作業を行う場合には信頼性できる(あるいは逆に信頼できない)複数の AS に対して決まった動作を取るということが多い. そこで, このような動作をマルチエージェントシステム上で実現するためにはエージェントを配置先 AS の関係を反映させたいいくつかのグループに分割して, そのグループに対して各エージェントが実行できるようにすることが有効であると考えた.

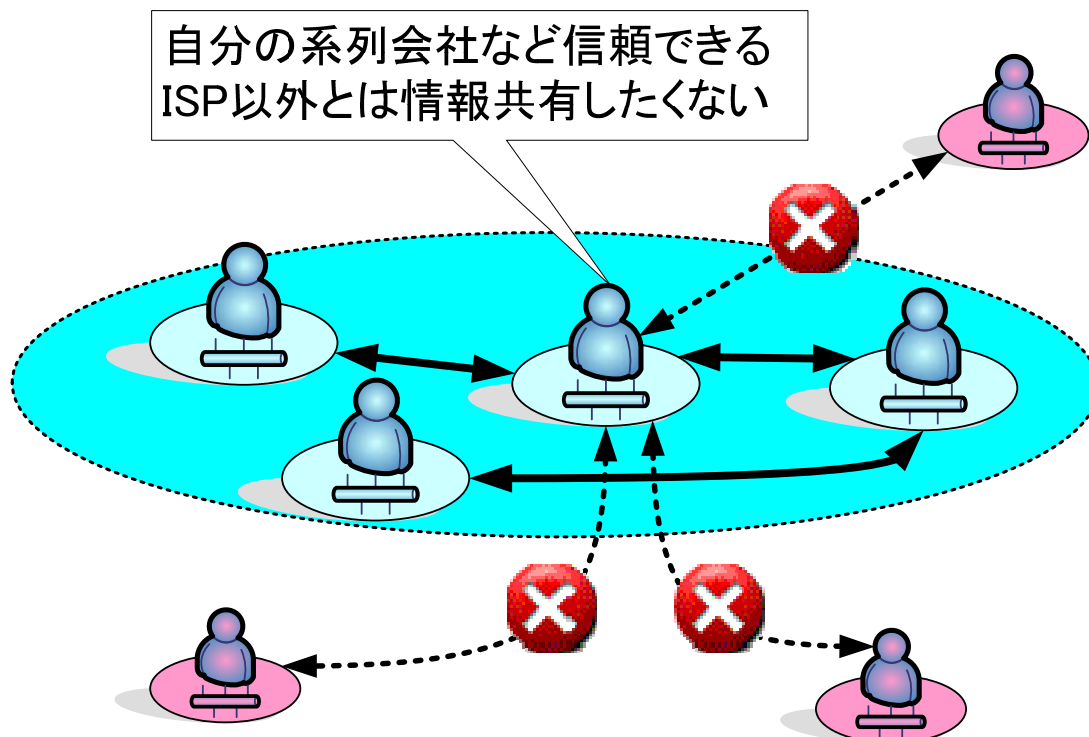


図 4-2 特定のグループ内での情報共有

そこで, 提案モデルではこのような実 AS 間に成り立つ信頼関係に基づいて形成される AS の集合を元に, それら AS に配置されたエージェントの集合をエージェントグループとして定義する. このエージェントグループの概念には以下の特長がある.

- AS 間の関係という基本的な情報であるため, マルチエージェントシステム(アプリケーション)に依存することなく成立する.
- 同一の企業グループに属する ISP など個々の AS の意図とは無関係に成立する.

これらの特性からグループの種類やメンバ情報などのエージェントグループに関する情報は個々のエージェント(AS)が持つのではなく、個別のマルチエージェントシステムとは独立した別システムによって共通的に管理することが必要である。例えば、筆者が第3章で提案したエージェントアーキテクチャにおいては、エージェント管理システム ARTISTE がその条件を満たすエージェントグループ情報を管理するシステムとして機能しうる。グループ情報を用いた情報共有の実現イメージを図 4-3 に示す。

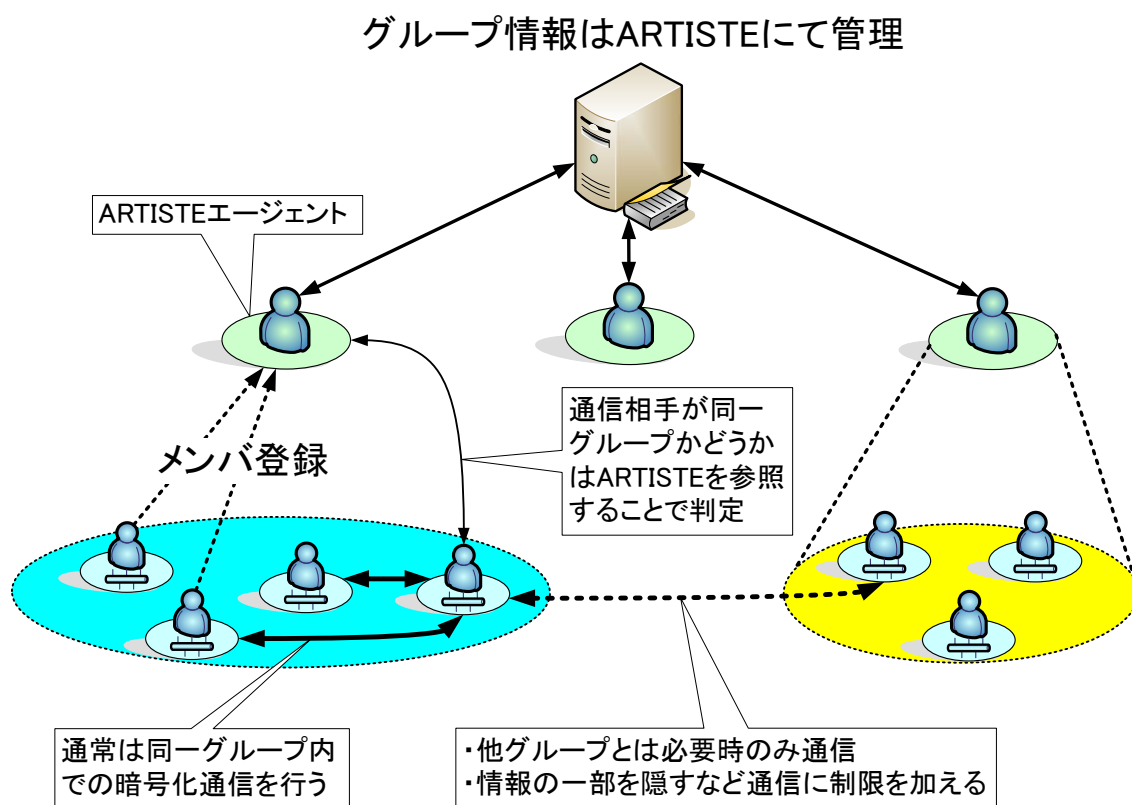


図 4-3 グループによる情報共有の実現イメージ

そして、マルチエージェントシステムでは、このエージェントグループ情報を随時参照しながら、協調相手に応じた動作の選択を行う。

信頼度の異なるエージェント毎に異なる方式や内容で情報共有

前述のグループによる情報共有方式は同一企業グループ内などエージェントを運用する個々の AS の意図に関係なく決まる global な情報を形式化したものであった。このような global な情報に基づく情報共有は AS やアプリケーションに依存することなく利用できるという利点がある反面、個々の AS の持つセキュリティや管理に対する独自の

ポリシーを反映できないという問題があった。本来、ASを運用するISPがどのような相手と協調していくかということは純粋に個々のASのポリシーによって決まるものであり、globalな関係に合致しないことも多い。例えば、同一の企業グループ内であっても特定のASには情報を制限して出したいとか、あるいはその逆の場合も当然考えられる。

そこで、このような要求に対応するため、本研究では個々のASが自分のポリシーに基づいて定義した他のASに対する信頼度をモデル化する方法とそれに応じた協調動作をエージェントが決定する情報共有方式を提案する。

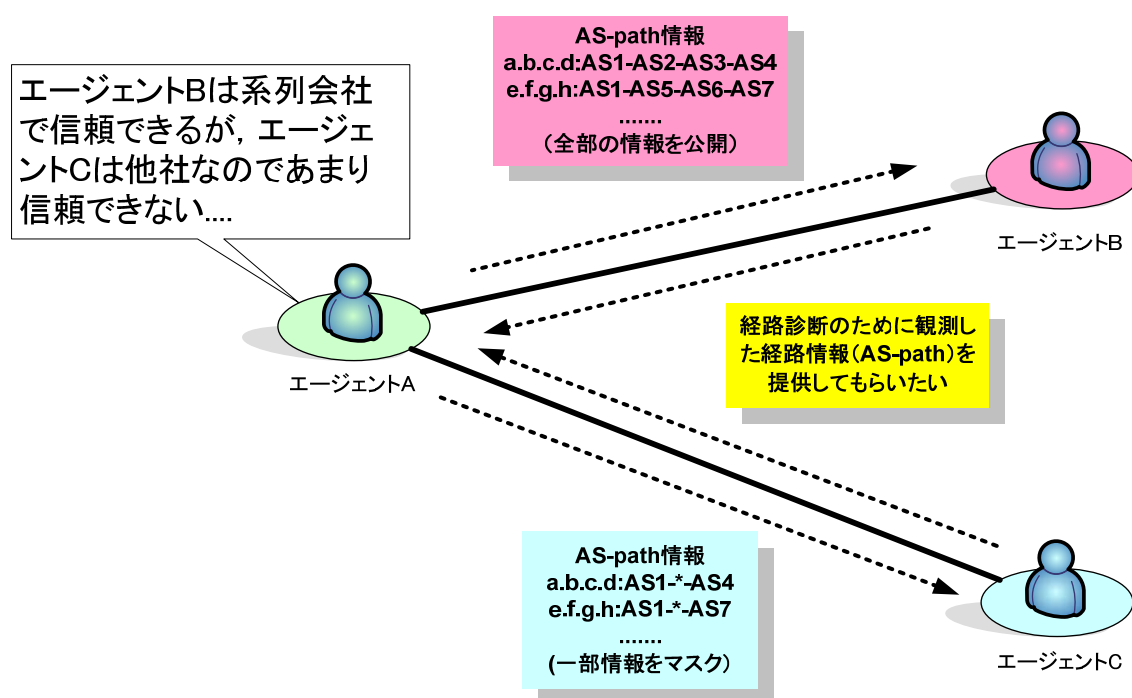


図 4-4 信頼関係に応じた動的な情報共有

4.4 提案方式

4.3.2 での検討に基づいて、広域ネットワーク管理用のマルチエージェントシステムにおけるエージェント間の情報共有を管理するための以下の2種類の方式を提案する。

- エージェントグループに基づく情報共有方式
- 個々のエージェントの信頼度に基づく情報共有方式

次章以降でそれぞれの方式の詳細について説明する。

4.4.1 グループによる情報共有方式

システム内のエージェントを適切にグループ分けするためには、はじめに各エージェントに対して適切な属性を付与する必要がある。例えば、エージェントを配置先 AS の企業グループによってグループ化したいのであれば、個々のエージェントに「企業グループ」などの属性データを付与し、同じ属性データを持つエージェントを 1 つのグループとして処理を行えばよい。そのため、ここで提案するグループによる情報共有方式を実現するためには次の 2 つの機能が必要である。

- エージェントの属性を付与、管理する機能
- 協調相手のエージェントの属性に基づいて協調動作を選択する機能

はじめに、属性を付与する方法について説明する。個々のエージェントの属性は[属性名, 値]という 2 項組によって定義され、1 つのエージェントは複数の属性を持つことができる。そして、各エージェントは同一の属性名と値を持つエージェント同士を 1 つのグループとして認識することで、1 つのグループに属する複数のエージェントに対して適切な動作を選択することができる。

属性は個々のエージェントの管理者やシステムの運営者によって作成、付与されることを想定している。エージェントに付与した属性を管理する方法は集中型のデータベースを利用して一元的に管理する方法、個々のエージェントが自分の属性を管理する方法などさまざまな方法が考えられるが、以下の説明では属性は前者の方法で管理されるものとする。すなわち、すべてのエージェントに付与された属性はエージェントとは別に用意されたネットワーク上のデータベース(以下、DB)に保存するものとし、すべてのエージェントは当該 DB に接続することで DB 内に保存されたエージェントの属性を参照できるものとする。DB を参照する際のキーはエージェントの IP アドレスなどエージェントを一意に特定できるものを利用する。なお、各エージェントが独自に属性を管理する方法では DB を使わないようにシステムを構成することも可能である。ただし、この場合は各エージェントが持つエージェントの属性情報に整合性が取れていることが保証される必要がある。

次に、付与した属性に基づいてエージェントが動作を選択する機能について説明する。はじめに各エージェントでは属性に基づく動作選択ルールを定義しておく。このルールは[属性名, 属性値, 動作]の 3 項組によって定義され、協調相手のエージェントがルー

ル中に記述された属性名および属性値と一致する属性を持つ場合に、ルール中の動作を実行する。このルールは属性とは異なり各エージェント内部に保存管理する。これは、動作選択ルールは他のエージェントから参照する必要がないためである。参照すべきルールや作業のどの時点でルールを参照するかなどは各エージェントが独自に定義する必要がある。具体的な選択手順を以下に示す。

- ① 各エージェントは他エージェントに通信を依頼する場合あるいは他エージェントから通信の依頼を受信した場合には、情報の交換を開始する前にDBに接続して相手エージェントの属性を参照する。このとき、参照の必要性の有無およびどの属性を参照するかは個々のエージェントに定義されているものとする。
- ② 属性を参照した後、エージェントでは自身のルールに基づいて別エージェントとの協調作業を実行し、その過程において必要であれば動作選択ルールを参照する。例えば、「所属」という属性値が自分と同一であるエージェントとは協調を行うが、それ以外の値を持つエージェントとは一切協調を行わないといったことを定義しておく。
- ③ ステップ①で得た相手エージェントの属性値と自身の動作選択ルールを照合することでルールに規定された動作を適宜選択しながら作業を行う。

協調エージェントが複数存在する場合は、これらの動作を必要回繰り返すことで協調相手のエージェント毎に動作を選択する。以上の動作手順を図4-5に示す。

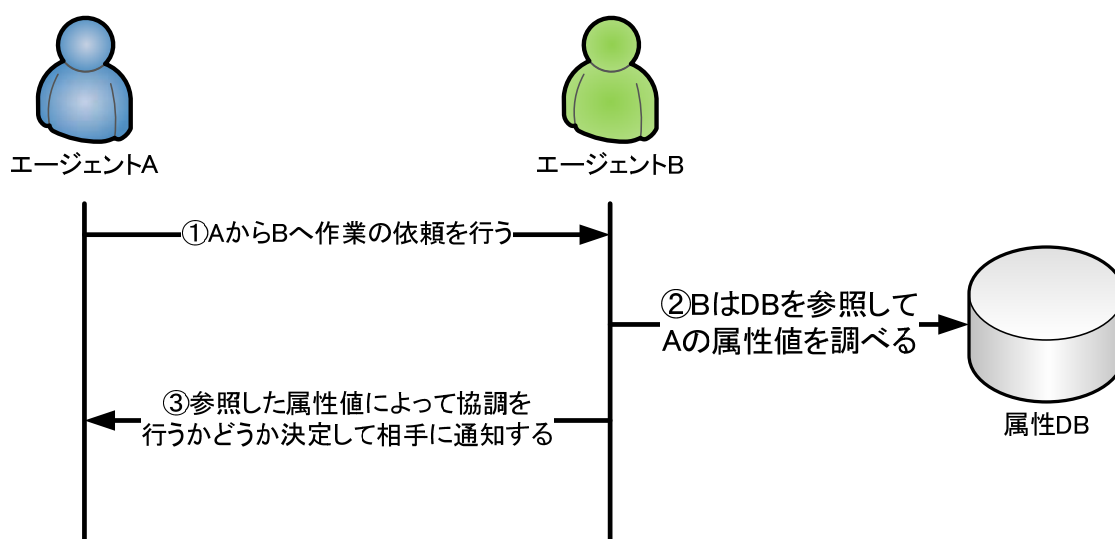


図 4-5 グループ通信の流れ

グループ通信を実際に利用する場合の例を以下に示す。

- 複数の組織が運営するエージェントが混在する環境において、自分と同じ組織により運営されているエージェントとのみ通信を行い、それ以外通信はすべて拒否する。
- エージェントの所属組織毎に交換する情報の内容を制限する。
- 本システムが有料で商用利用されている場合に、エージェントの運営者の支払った料金に応じて交換する情報の内容を制限する。

4.4.2 個々のエージェントの信頼度に基づく情報共有方式

本方式では、先のグループ通信とは異なり各エージェントは他エージェントに対する信頼度を自己(=配置先 AS)のポリシーに基づいて設定し、協調相手との信頼度を元にして適切な通信を行う。まず、各エージェントは他エージェントに対する信頼度を独自に設定し、エージェント内部に保存する。この信頼度は各エージェントのポリシーを反映したものであるため、DB 等エージェント外部ではなく各エージェントの内部に保存することが想定される。もちろん、エージェントとは別に用意した DB を用いて各エージェントの設定した信頼度を保存するような構成も可能である。また、この信頼度の設定は任意であり設定しなくともよい。

ここで注意しなければならないのは、信頼度の値はエージェント独自に決定することができるが、その値の決定基準に関してはシステム全体すなわち全エージェントで共通の合意がなければならない。このような基準が存在せず、各エージェントが独自に値を決定しただけでは相互評価の仕組みが有効に機能しないからである。このようなシステム共通の基準を決めるにはシステムの管理主体が決定する、各エージェントの運営者の合意によって決定するなどの方法が考えられる。この決定基準は例えば、

- 信頼度の上限は 100 とする。
- すべての情報を公開してよいエージェントには信頼度 80 以上をつける。
- 一切の通信を行わないエージェントには信頼度 0 をつける。

といったように決めておく。

次に、各エージェントが決めた信頼度を用いた通信を行うために各エージェントでは

他のエージェントに対する信頼度の設定と同時に、先のグループ通信と同様の動作選択ルールを定義する必要がある。これは[最低信頼度, 動作]の2項組からなり、協調相手のエージェントの信頼度が最低信頼度を上回った場合に規定された動作を行うように定義される。この動作選択ルールも外部に公開する必要はないので各エージェントが内部に保存する。

以上の準備の元に、信頼度を使った2者間の通信の手順を以下に示す。この説明ではエージェントAがエージェントBに通信を依頼するものとする。また、以下ではエージェントAが設定したエージェントBの信頼度を $ev(A,B)$ と表記する。

- ① Aは通信の依頼とともに自分が設定したBの信頼度、すなわち $ev(A,B)$ を送信する。
- ② Aからの依頼を受信したBは、自分が設定したAの信頼度、すなわち $ev(B,A)$ を参照して、 $ev(A,B)$ との最小値($=\min\{ev(A,B), ev(B,A)\}$)を計算する。この値が、このA-B間通信の信頼度となる。以降の説明ではA-B間通信の信頼度を $evcomm(A,B)$ と表記する。
- ③ Bは、Aから依頼された処理の実行可能信頼度と前ステップで決定したA-B間通信の信頼度とを比較して後者が前者を上回っている場合には処理を実行してその結果をAに返答する。それ以外の場合には信頼度の不足により依頼された処理を実行できない旨を返答する。

具体的な通信例を図4-6に示す。

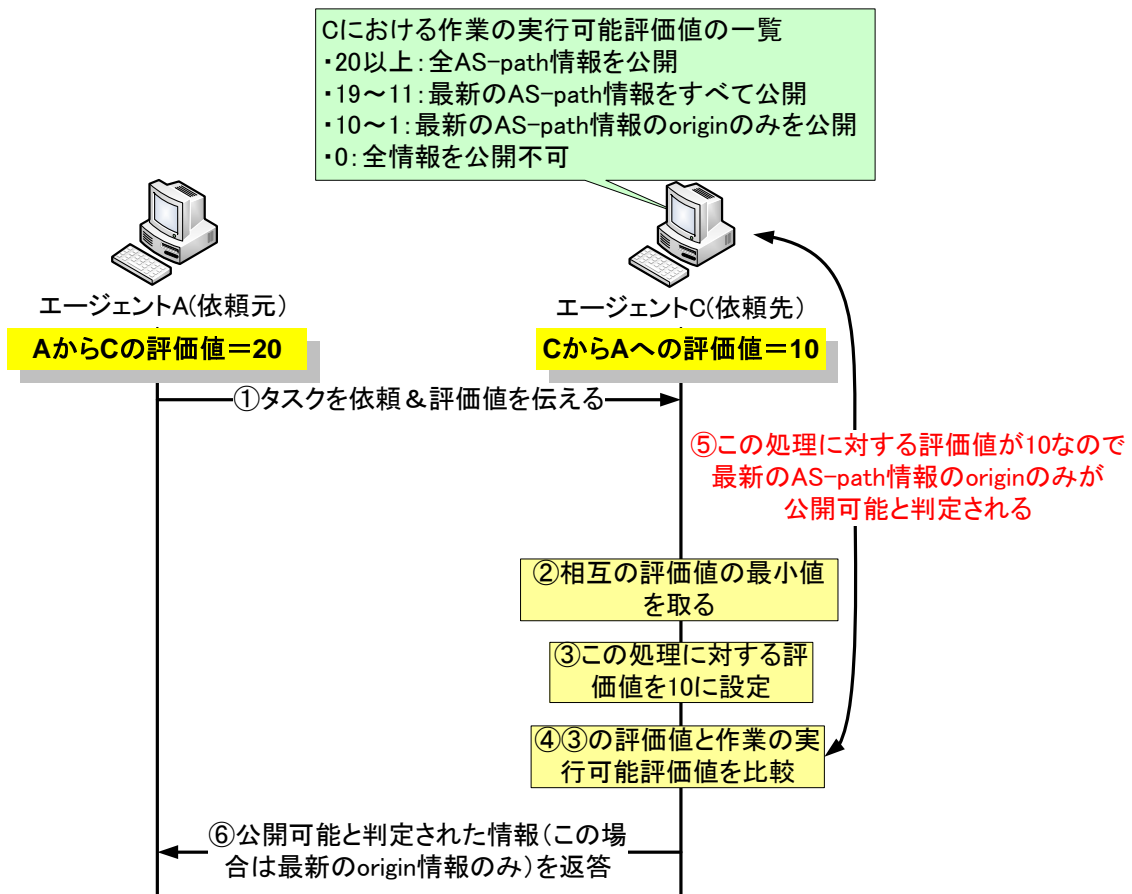


図 4-6 信頼度を用いた二者間通信の例

上記アルゴリズムのステップ②において信頼度を比較する意味は、悪意のあるエージェントが他のエージェントに処理を依頼するにもかかわらず自分自身では他エージェントの制限レベルを不当に低く設定して他エージェントからの処理の依頼を拒否するという通信の不公平性を回避することにある。この不公平性の概念を図 4-7 を用いて述べる。

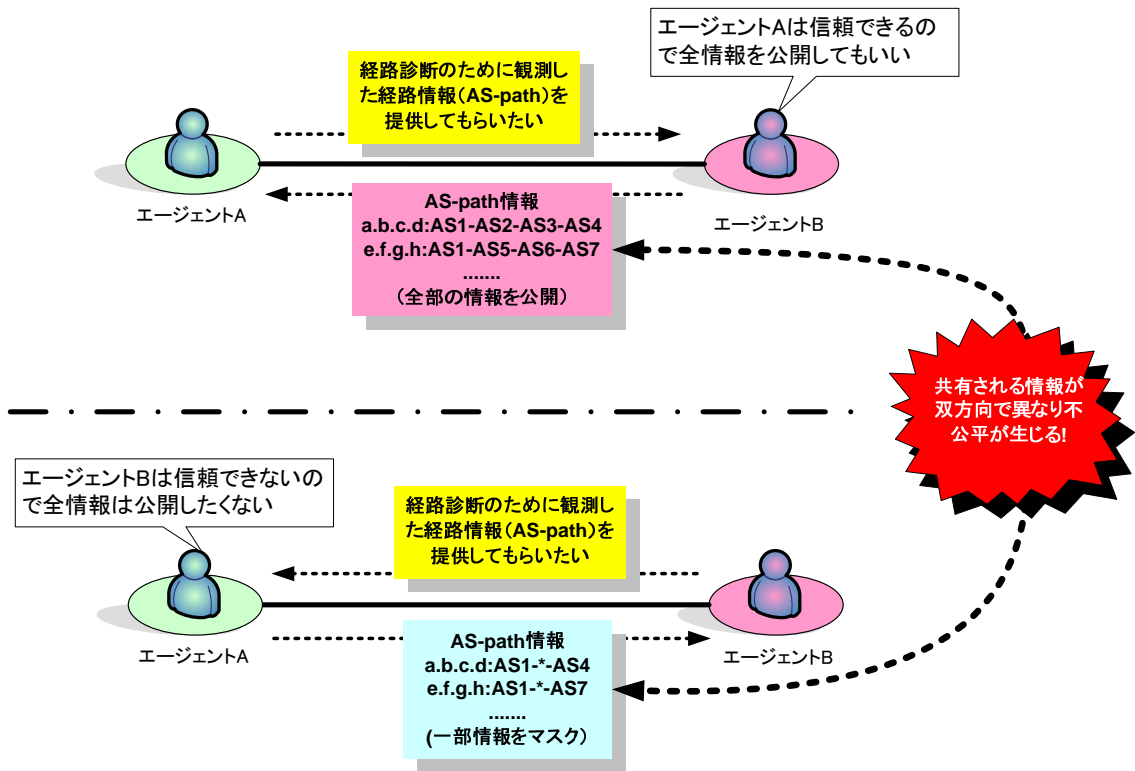


図 4-7 エージェントによる情報共有の不公平性

図中のエージェント A と B の互いの信頼度は次のようになっているものとする。

- B は A のことを全面的に信頼している。
- A は B のことをあまり信頼していない。

このとき、図上部に示すように A から B へと情報の要求があったとする。B は A のことを信頼しているため、A の要求に基づいて所有する全ての情報を公開することができる。しかし、逆に B から A へと同様の情報の要求があった場合を考えると、A は B のことを信頼していないため、所有する情報の一部しか公開しない。つまり、エージェント B から見れば自分は A に対してすべての情報を公開しているのに A からは一部の情報しか得られないという不公平が生じる。この問題をさらに悪用すると、悪意のあるエージェントによって自分の持つ情報は全く公開せずに他のエージェントから情報を無制限に受信することが可能になるというより深刻な問題を引き起こすため、適切な対策が必須である。

提案するアルゴリズムではステップ②で示した信頼度の交換および最小値の計算プ

プロセスを導入することで、情報共有のレベルが通信を行う 2 者の信頼度の低い側に設定されるので、他エージェントからの依頼を拒否するエージェントは同じ確率で他エージェントから処理の実行を拒否されることになる。そのため、通信に関わる一方にだけ情報が多く提供されるという不公平性を改善することが可能となっている。

エージェントを用いたネットワーク管理においては、処理を依頼されたエージェントがさらに別のエージェントに詳細調査など処理を依頼するというような、三者以上のエージェントが関わる通信が行われることがある。この場合には、先に説明した二者間通信を順に適用しただけでは問題が発生する。この問題を図 4-8 を用いて説明する。

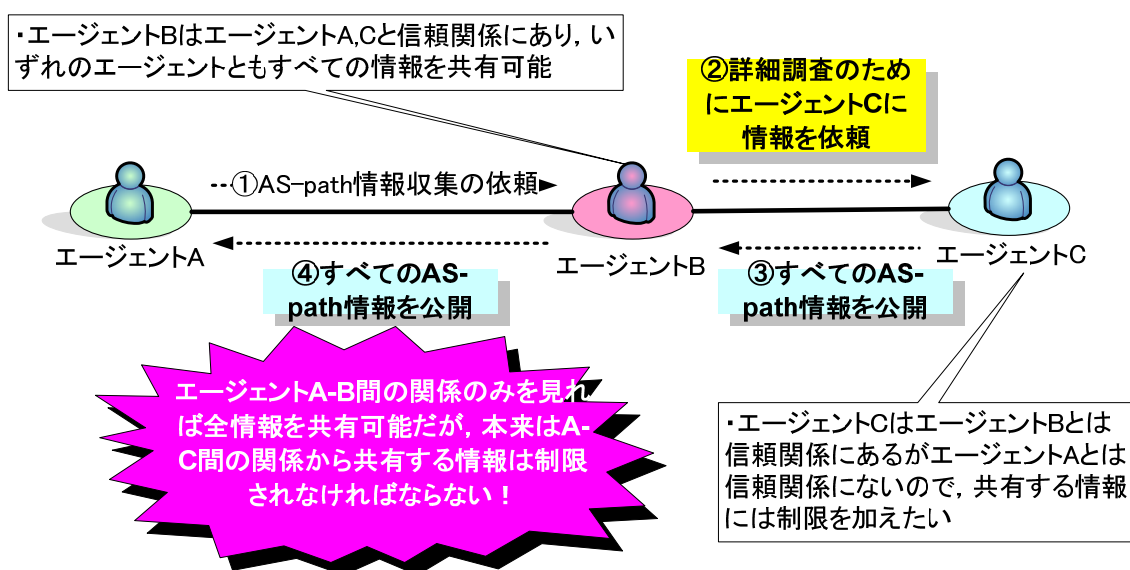


図 4-8 三者通信における問題

図中において、エージェント A から B に最初の情報公開の依頼があり、それに基づいて B が別のエージェント C に対して情報公開を依頼したものとする。そして、C から公開された情報は最初の要求元である A に B を経由して転送されるものとする。このとき、エージェント B は A および C と信頼関係にあるが、A と C は互いに信頼していないと仮定する。このとき、単純に二者間通信の繰り返りで情報公開を行うと次のようなことが起こる。

- C と B は信頼関係にあるため、C から B へはすべての情報を公開する
- 同様に A と B も信頼関係にあるため、B は C からもらった情報をすべて A に公開する。

ところが、C と A とは信頼関係にないので本来であれば C から(最終的に)A に送られる情報には制限が加わっていなければならず、このように B を介することですべての情報が公開されてしまうことはエージェント C のポリシーに反する。この問題は直接通信を行う二者に限って信頼度の適用を行ったことが原因であり、この問題を回避するためには通信に関わるすべてのエージェントの信頼度によって通信が制限されなければならない。

以上の検討を元に三者以上のエージェントの通信を信頼度によって制限する手順を以下に示す。この説明では前述の例と同様にエージェント A がエージェント B に通信を依頼し、B がさらにエージェント C に処理を依頼するものとする。

- ① A から B へ通信の依頼が行われると、A と B は二者間通信と同様に A-B 間通信の信頼度、すなわち相互の信頼度の最小値を求める。
- ② B が C へ通信を依頼する必要があると判定した場合、B は以下の信頼度を計算する。
- ③ 前ステップで求めた A-B 間通信の信頼度 ($=ev_{comm}(A,B)=\min\{ev(A,B),ev(B,A)\}$)

B が設定した C の信頼度($=ev(B,C)$)

- ④ B では通信の経路情報を作成する。作成の手順は以下の通りである。この例では A が通信の起点であるため、B により {A,B} という経路情報が作成される。
- ⑤ 通信相手(この場合は A)から経路情報が送られていない場合、通信相手が通信の起点と考えて、自分と通信相手を経路情報に加える。
- ⑥ 通信相手から経路情報が送られてきた場合、経路の最後尾に自分を追加する。
- ⑦ 前ステップで求めた信頼度と経路情報を B から C に送信する。
- ⑧ C は、B-C 間通信の信頼度 $ev_{comm}(B,C)$ を決めるために自分が設定した B の信頼度と、B が設定した C の信頼度との最小値を計算する。すなわち、 $ev_{comm}(B,C)=\min\{ev(B,C), ev(C,B)\}$ となる。
- ⑨ 次に、B-C 間通信の信頼度と、A-B 間通信の信頼度の最小値を取り、それを A-B-C 間通信の信頼度とする。この信頼度 $ev_{comm}(A,B,C)$ は次のように計算できる。

$$\begin{aligned}
 & ev_{comm}(A,B,C) \\
 & =\min\{ev_{comm}(A,B),ev_{comm}(B,C)\} \\
 & =\min\{\min\{ev(A,B),ev(B,A)\},\min\{ev(B,C),ev(C,B)\}\}
 \end{aligned}$$

$$=\min\{ev(A,B),ev(B,A),ev(B,C),ev(C,B)\}$$

- ⑩ 次に、Cは通信に関連するエージェントのリストを参照して、それらに含まれるエージェントに対する自分の信頼度の中の最小値を計算する。すなわち、 $\min\{ev(C,A),ev(C,B)\}$ を求める。
- ⑪ 最後に、A-B-C間通信の信頼度と前ステップのエージェント全体の信頼度の最小値を計算する。すなわち、以下の値が本通信の信頼度となる。

$$\min\{ev_{comm}(A,B,C),\min\{ev(C,A),ev(C,B)\}\}$$

$$=\min\{\min\{ev_{comm}(A,B),ev_{comm}(B,C)\},\min\{ev(C,A),ev(C,B)\}\}$$

$$=\min\{\min\{ev(A,B),ev(B,A),ev(B,C),ev(C,B)\},\min\{ev(C,A),ev(C,B)\}\}$$

$$=\min\{ev(A,B),ev(B,A),ev(B,C),ev(C,A),ev(C,B)\}$$

- ⑫ Cは前ステップで求めた信頼度と、依頼された処理の実行可能信頼度とを比較して前者が後者を上回っている場合には処理を実行してその結果をBに返答し、BはそれをAに返答する。それ以外の場合には信頼度の不足により依頼された処理を実行できない旨を同様にB経由でAに返答する。

本アルゴリズムを適用することにより、通信に関わるすべてのエージェントの中で最も信頼度の低いエージェントに合わせて情報共有が行われるため、本来公開すべきでない情報が公開されてしまうという事態を防ぐことができる。なお、本説明では三者間の通信を例にしたが、この手順は通信に関わるエージェントが三者以上になったとしても適用できる。

本章で説明した信頼度による情報共有と前章のグループによる情報共有とは互いに独立しており、二つを併用することもどちらか一方のみを使うことも可能である。

4.5 提案方式の評価

4.5.1 実装

提案する共有情報管理方式の有効性を検証するために、既存のネットワーク管理用マルチエージェントシステムである経路診断システム ENCORE に提案方式を組み込むことによって実装した。個々のエージェントは以下のモジュールから構成される。エー

ジェントの構成図を図 4-9 に示す。また、システム全体の構成例を図 4-10 に示す。

- ① エージェント間通信モジュール
- ② ネットワーク機器間通信モジュール
- ③ 障害診断・推論エンジン
- ④ 属性利用通信モジュール
- ⑤ 評価利用通信モジュール
- ⑥ 評価リスト

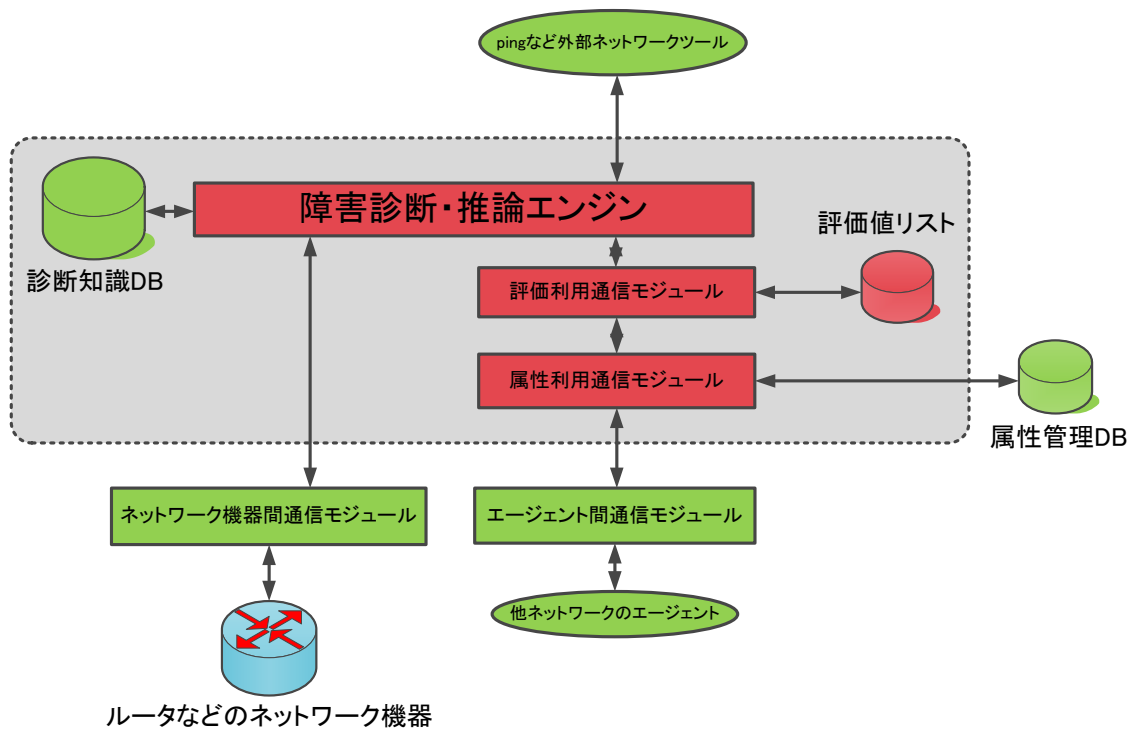


図 4-9 エージェント構成図

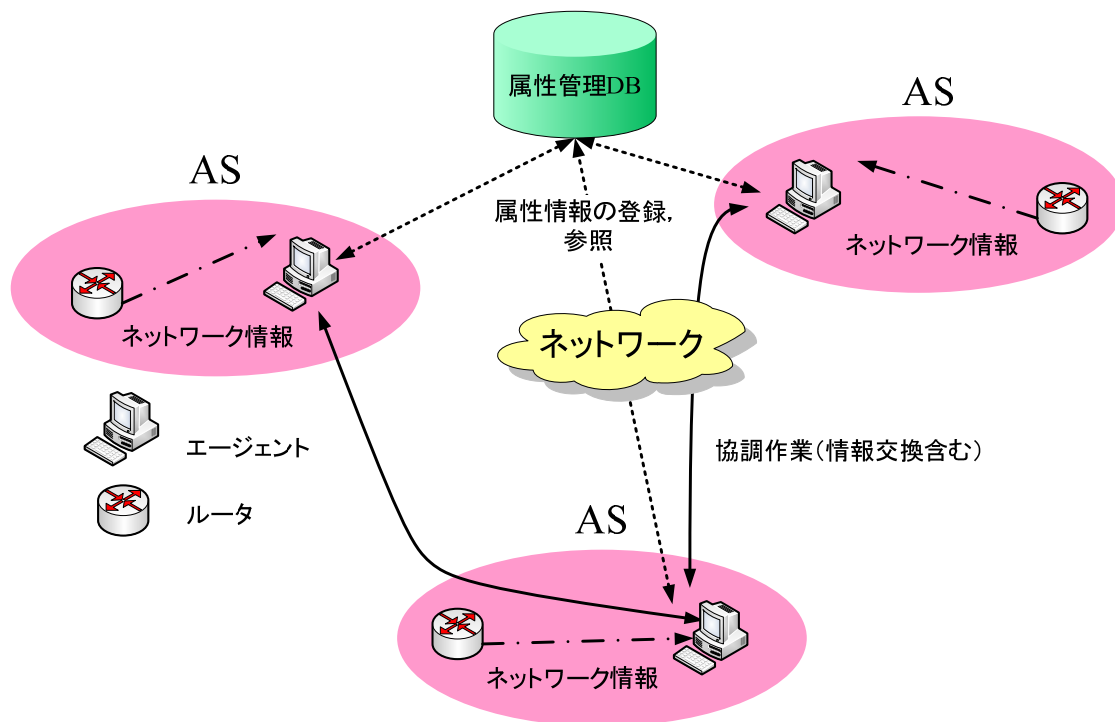


図 4-10 システム構成図

4.5.2 動作例

ここでは、ネットワーク管理用ある AS に配置された ENCORE エージェント（エージェント A とする）が別の AS に配置された 2 つの ENCORE エージェント（それぞれエージェント B,C とする）から AS-path に関する観測結果を要求する場合を例に取り、本発明による診断手順の実施例を説明する。また、以下では簡単のため、ENCORE エージェント A,B,C をそれぞれ A,B,C と表記する。

- ① A 中の推論エンジンが診断知識 DB 中の障害判定用知識と観測したネットワーク情報を照らし合わせることで経路情報がハイジャックされた可能性があることを検知したとする。
- ② 次に、A はこの仮説を検証するために実行すべき診断項目の集合および実行順序を導出する。この集合は各「仮説」に関する知識として診断知識 DB にあらかじめ組み込まれている。この診断項目の定義例（一部）を以下に示す。なお、この知識は Lisp 言語で記述されている。

```
;; 診断項目: BGP 経路が広報されているか
(def-rule BGP-advertised () ←ルール の定義
  (acquire (check-bgp-as-path))←記載された関数を実行
  (eval (BGP-advertised acq-result)))←関数の実行結果を評価し, 本項目の結果とする
```

```
;; 診断項目: BGP 経路が 10 分前に広報されていたか
(def-rule BGP-advertised-10min-ago ()
  (acquire (check-bgp-as-path-10min-ago))
  (eval (BGP-advertised acq-result)))
```

- ③ A の推論エンジンは前ステップで導出した診断項目を規定された順番に実行する。この例では、自分と同じ「所属組織」属性の値を持つエージェントから当該経路に関する AS-path 情報を取得するという項目を実行することとする。
- ④ そのため、A では属性利用通信モジュールを起動して属性管理 DB に対して検索を行い、「所属組織」属性の値が自分と同じエージェントのリストを取得する。今回の例ではそれがエージェント B,C であったとする。
- ⑤ 次に、A は B,C に対してハイジャックされた当該経路に関する最新の AS-path 情報および保存されている過去の AS-path 情報の履歴をすべて要求する。
- ⑥ ここで、プログラム間の信頼度の関係および B,C における AS-path 情報提供というタスクの実行レベルリストが以下の通り定義されていると仮定する。

表 4-1 信頼度の関係

	A から見た評価	B から見た評価	C から見た評価
対 A		20	10
対 B	20		20
対 C	20	20	

表 4-2 エージェント B,C のタスクの実行レベルリスト

信頼度	実行可能タスク
20 以上	全 AS-path 情報を公開
19~11	最新の AS-path 情報をすべて公開
10~1	最新の AS-path 情報の origin のみを公開
0	全情報を公開不可

- ⑦ A は信頼度利用通信モジュールを起動して B,C との通信を開始し, AS-path 情報の公開を依頼する.
- ⑧ まず, B に対してはタスク, つまり AS-path 情報の公開の依頼と共に自身の B に対する信頼度の値が 20 であることを送信する.
- ⑨ 次に, B では IP アドレスより通信を依頼してきたのが A であることが理解できるため, 自分の信頼度の一覧を参照して A に対する信頼度の値の 20 を求め, 前ステップで送信された A からの信頼度との最小値を計算してタスクの実行レベルを求める. この例では共に信頼度が 20 なので実行レベルも 20 になる.
- ⑩ 次に, B では自身のタスク実行レベルのリストを参照して, 依頼された AS-path 情報の公開というタスクのどのレベルまで実行可能かを判定する. この例では前ステップで実行レベルが 20 であることが分かっているので表 4-2 より全 AS-path 情報が公開可能であると判定される.
- ⑪ 前ステップの判定に基づき, B は A に対し要求のあった経路に関する全 AS-path 情報を送信する.
- ⑫ C の場合も同様であるが, 表 4-1 より C から見た A の信頼度が 10 であることからタスクの実行レベルは 10 となる. そのため, C から A に対しては表 4-2 より最新の AS-path 情報の origin のみが公開可能であると判定され, それらの情報のみが送信される. 以上の通信の流れをを図 4-11 に示す.

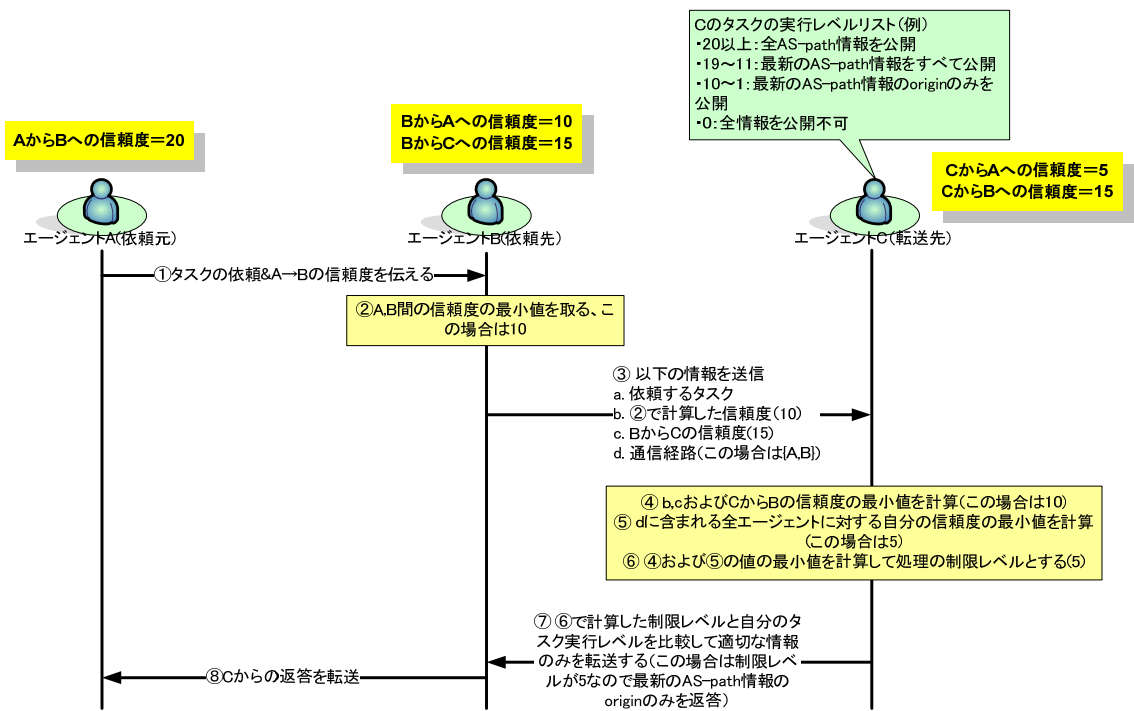


図 4-11 三者間通信の動作例

4.5.3 実験結果

図 4-9 および図 4-10 で示した実験用マルチエージェントシステムを動作させるために市販の仮想マシンソフトウェアおよび同ソフトウェアの仮想ネットワーク機能を使って 12 個のプライベート AS からなる実験用ネットワークを構築した。実験ネットワークの構成図を図 4-12 に示す。

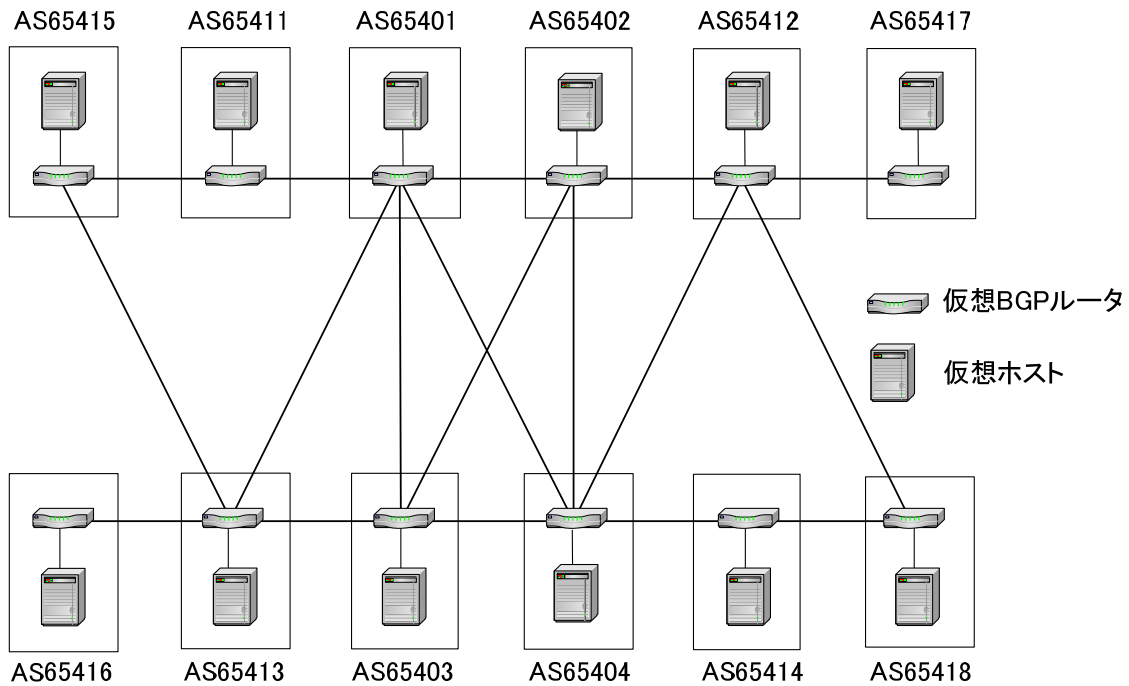


図 4-12 実験ネットワーク構成図

実験ネットワーク上の各 AS では仮想ホストが各々2つ動作しており、一つのホスト上ではネットワーク管理用のエージェントが一つ動作している。もう一つのホスト上でフリーのルータソフトウェア **quagga** を動作させることで BGP ルータとして機能させるように構成した。また、この実験ネットワークはインターネットとは接続していないが、実際のインターネットへの適用時の性能を検証するために実際のインターネットから取得したフルルート(約 16 万の経路情報)を BGP ルータ間で流通させている。

次に、提案する情報共有方式の有効性を検証するために、この実験ネットワーク上で以下のシナリオに基づくネットワーク管理を実行した。

- ① 準備として AS65416 上のエージェントは自分が広報しているネットワークプレフィックス **192.168.16.0/24** がネットワーク内に正しく伝搬されていることを確認するために、ネットワーク内の別 AS65417 上で動作するエージェントにあらかじめ同ネットワークの監視を依頼しておく。依頼を受けた AS65417 上のエージェントは自 AS のルータから BGP 情報を定期的を取得することで依頼されたネットワークプレフィックスの情報の伝搬状況を把握する。
- ② このとき、実験ネットワーク内の AS65412 が AS65416 が広報しているネットワークプレフィックス **192.168.16.0/24** を自分の origin とする新たな経路情報をネットワーク内に広報したもとする。

- ③ AS65412 によって広報されたネットワーク情報は複数の AS を経由しながらネットワーク内に広がっていき、各 AS では新たな情報に基づいてルータ上の経路情報を更新する。
- ④ このような広報がなされたことにより 192.168.16.0/24 宛のパケットが AS65412 に送られることになり、ネットワーク上では以下のような接続障害が発生する。
- A) AS65416 以外の AS から 192.168.16.0/24 宛に送られたパケットはすべて AS65412 に送られるが実際には AS65412 には 192.168.16.0/24 というネットワークが存在しないため当該パケットはすべてルータによって破棄される。そのため、AS65416 以外の AS から 192.168.16.0/24 への接続が不可能になる。
 - B) AS65416 内の 192.168.16.0/24 に含まれるホストから他のネットワークへ通信を行う場合、送信パケットは正しくネットワークに到達するが、それに対する返答のパケットは誤った経路情報に基づいて AS65412 に転送されてしまう。そのため、192.168.16.0/24 内のホストには他からの返答パケットが届かなくなるので、結果として当該ホストから他のネットワークへの接続が不可能になる。
- ⑤ このとき、AS65417 上のエージェントは AS65416 からの依頼に基づいて 192.168.16.0/24 に関する経路情報を監視しているため、AS65417 に 192.168.16.0/24 に関する新たな経路情報が伝搬されてきた場合に、origin が本来の情報と異なっていることをすぐに検知できる。そのため、AS65417 上のエージェントは AS65412 上のエージェントにその旨を通知すると同時に原因および影響範囲の特定のために周辺の AS 上のエージェントと通信して情報収集を試みる。
- ⑥ そして、このとき各周辺エージェントにおける信頼度の設定をのように設定した場合にエージェントに送られる出力結果を以下に示す。
- A) table vega4: spica2 = 10 それ以外 = 20
 - B) table spica3: vega4 = 30 deneb = 15 それ以外 = 20
 - C) table deneb: vega4 = 8 spica2 = 18 それ以外 = 20
 - D) table alioth: vega4 = 19 それ以外 = 20
 - E) table bellatrix: vega4 = 10 それ以外 = 20

- spica3(192.168.17.15)からの報告

信頼度の設定から vega4 と spica3 は特に制限は加わっていないので、この 2 者間では全情報が取得可能である。しかし、

- A) deneb は vega4 を 8 と設定している
- B) alioth は vega4 を 19 と設定している
- C) bellatrix は vega4 を 10 と設定している

ことから

- A) deneb(65401) は AS-PATH 情報が全てフィルターされる
- B) aliph(65402) は過去の経路情報がフィルターされる
- C) bellatrix(65404) は AS-PATH 情報は origin のみ公開する

という制限が出力結果に加えられていることがわかる。

```
hijack detected by: 192.168.17.15:4333 (65417)
```

```
current path (hijacked): 65412
previous path: 65412 _65404_ _65401_ _65413_ _65416_
prefix: 192.168.16.0/24
```

```
encore: 192.168.17.15:4333 (65417)
```

```
192.168.16.0/24
```

```
-32 to NOW sec: 65412
```

```
(中略)
```

```
-98471 to -6274 sec: 65412 65404 65401 65413 65416
```

```
encore: 192.168.1.13:4333 (65401)
```

```
192.168.16.0/24
```

```
-11051756 to NOW sec: (as-path information is omitted)
```

```
(historical data are omitted)
```

```
encore: 192.168.2.11:4333 (65402)
```

```
192.168.16.0/24
```

```
-27 to NOW sec: 65412
```

(historical data are omitted)

encore: 192.168.3.23:4333 (65403)
192.168.16.0/24
-11051658 to NOW sec: 65413 65416

encore: 192.168.4.21:4333 (65404)
192.168.16.0/24
-24 to NOW sec: 65412 (only origin is reported)
(historical data are omitted)

encore: 192.168.13.22:4333 (65413)
192.168.16.0/24
-7170042 to NOW sec: 65416

encore: 192.168.14.24:4333 (65414)
192.168.16.0/24
-16 to NOW sec: 65404 65412
(中略)
-98479 to -6285 sec: 65404 65401 65413 65416

encore: 192.168.16.26:4333 (65416)
192.168.16.0/24
-11052558 to NOW sec:

encore: 192.168.18.16:4333 (65418)
192.168.16.0/24
-29 to NOW sec: 65412
(中略)
-98488 to -6285 sec: 65414 65404 65401 65413 65416

- spica2(192.168.14.24)からの報告
- 信頼度の設定から以下のように出力が制限される.

- A) - spica2(65414) は AS-PATH 情報は origin のみ公開する
- B) - deneb(65401) は AS-PATH 情報が全てフィルターされる
- C) - alipth(65402) は過去の経路情報がフィルターされる
- D) - bellatrix(65404) は AS-PATH 情報は origin のみ公開する

出力結果から上記の制限が加えられていることが確認できる。

```
hijack detected by: 192.168.14.24:4333 (65414)
current path (hijacked): 65412 (only origin is reported)
previous path: (as-path information is omitted)
prefix: 192.168.16.0/24
```

```
encore: 192.168.14.24:4333 (65414)
```

```
192.168.16.0/24
```

```
-29 to NOW sec: 65404 65412
```

(中略)

```
-98492 to -6298 sec: 65404 65401 65413 65416
```

```
encore: 192.168.1.13:4333 (65401)
```

```
192.168.16.0/24
```

```
-11051774 to NOW sec: (as-path information is omitted)
```

(historical data are omitted)

```
encore: 192.168.2.11:4333 (65402)
```

```
192.168.16.0/24
```

```
-42 to NOW sec: 65412
```

(historical data are omitted)

```
encore: 192.168.3.23:4333 (65403)
```

```
192.168.16.0/24
```

```
-11051675 to NOW sec: 65413 65416
```

```
encore: 192.168.4.21:4333 (65404)
```

```
192.168.16.0/24
```

```
-41 to NOW sec: 65412 (only origin is reported)
```

(historical data are omitted)

encore: 192.168.13.22:4333 (65413)

192.168.16.0/24

-7170058 to NOW sec: 65416

encore: 192.168.17.15:4333 (65417)

192.168.16.0/24

-52 to NOW sec: 65412

(中略)

-98491 to -6294 sec: 65412 65404 65401 65413 65416

4.6 本章の総括

本章では, マルチエージェントシステムを実際の広域ネットワーク管理に適用することを想定した検討を行い, それにより, 従来のマルチエージェントシステムによるネットワーク管理システムでは取り扱われていなかったエージェント間の協調動作に起因する情報流出の可能性などの問題の存在を明らかにした. そして, 実際のネットワーク管理における情報共有の実態を分析した結果から以下の 2 種類の情報共有方式をエージェント間協調の制御方式として提案することで課題の解決を行った.

- エージェントグループに基づく情報共有方式
- 個々のエージェントの信頼度に基づく情報共有方式

提案するこれらの方式は, エージェント間の動作に関してレベルの違うカスタマイズを可能とするものであり, エージェントの運用者の要求やシステムの運用形態に合わせた柔軟な運用を行うことができる. さらに, 信頼度の交換によって情報共有の不公平性を解消する手段や 2 者以上のエージェントが介在する通信にも適用可能な汎用性の高いアルゴリズムを考案することにより, 広域ネットワーク管理の中でも特に重要な問題とされる情報共有を安全に行うために十分な効果をもたらすことが期待できる.

そして, 提案方式の効果を検証するためにこれらの方式を組み込んだネットワーク管理用マルチエージェントシステムを実際に試作し, フルルート情報を流通させた実験用ネットワーク上で稼働させて実験を行った. 実験の結果, 提案方式によってエージェント相互の信頼度に応じた適切な情報共有が行えることが確認できた.

以上より、提案方式はエージェント間の協調動作に対して高いカスタマイズ性と安全性をもたらすものであり、広域ネットワーク管理にマルチエージェントシステムを安全に適用するために不可欠であるエージェント間協調動作管理技術を明確化するものである。

第5章 結論

本論文は、マルチエージェントシステムを用いて複数の AS を跨る広域ネットワークをネットワーク状態に適応して管理制御するためのマルチエージェントによる適応型ネットワーク管理技術を明らかにするものである。

インターネットの広がりや留まることなく、ますます巨大化および複雑化が進んでいる。加えて、広帯域のネットワークおよび高性能の計算機の普及に伴い、高品質の映像配信などインターネット上でのサービスが高度化、多様化している。このようなサービスが多様化する状況においては、それぞれのサービス毎にネットワークに対する要求条件が異なり、それらの要求条件を満たすようにインターネットを現在のように人手で適切に管理制御することはますます難しくなっている。

マクロな視点で見た場合、インターネットは ISP (Internet Service Provider) など独立した主体が管理運用する AS (Autonomous System) が相互接続した巨大な分散システムと捉えられる。個々の管理主体は通常自分が管理する AS 内しか観測制御できず他の AS の状態を知ることができないため、複数の AS を跨る広域ネットワークを管理・制御するには AS(管理主体)同士による協調作業が必要である。このように、広域ネットワークの管理は本質的に複数主体間の分散協調作業であり、この作業を効率化するために同じ分散協調型のアプローチで問題解決を行うマルチエージェントシステムのネットワーク管理への適用が有効なアプローチであると考えられている。

本章では、本論文で提案し、実証した、マルチエージェントによる適応型ネットワーク管理技術の構成要素であるマルチエージェントシステムの動作の適応性、効率性を向上させるための「広域ネットワーク管理のためのエージェント組織化支援技術」および広域ネットワーク管理作業にマルチエージェントシステムを安全に適用するための「エージェント間協調動作管理技術」に関して、明らかにした結果について総括する。さらに、本研究が将来のネットワーク管理技術の実用化に果たす役割と今後の研究の展望について述べる。

5.1 広域ネットワーク管理のためのエージェント組織化支援技術

マルチエージェントによる適応型ネットワーク管理技術を構成するための課題 1 に対して、複数の AS を跨る広域ネットワーク管理のためのマルチエージェントシステム

を対象に、ネットワーク上の位置や機能など特定の条件を満たすエージェント群を動的に探索・同定することにより組織化を支援するエージェント組織化支援技術を明らかにした。

具体的には、組織化のために必要な広域ネットワーク情報を複数のエージェントを用いて分散観測することで単一の AS における観測では獲得の難しい広域ネットワークのトポロジ情報などを利用できるエージェント組織化支援技術を提案した。さらに、広域ネットワーク管理を行う複数のマルチエージェントシステムの動作を分析した結果、これらの組織化作業や利用する情報の共通性があることを明らかにした。これらの知見を元にして複数のマルチエージェントシステムに組織化支援機能を提供することができる共通プラットフォームとしてエージェント組織化システム ARTISTE (Agents' Roles and Topological Information management System for agent TEamwork) を実装した。

実際のインターネットと同等の情報が流通している実験環境上で実装した ARTISTE の評価実験を行った結果、ARTISTE はネットワーク上の広域ネットワーク管理用マルチエージェントシステムのための組織化を正しく行え、結果として組織化を静的に行う場合と比べてマルチエージェントシステムの運用コストを下げることができると、実際のインターネット環境においても実用的な探索範囲内では 3 秒以内で探索が完了することを確認できた。さらに、ARTISTE を含めたシステムアーキテクチャを採用することで、広域ネットワーク管理用マルチエージェントシステムの構築のための工数が従来と比べて約 8.3%削減できることを確認した。

本技術により、マルチエージェントシステムによってネットワーク管理を行う際に、ネットワークの状態や問題解決の文脈に適応してネットワーク上に分散配置されたエージェントの中から動的に最適なエージェントを探索して組織化することができるようになるため、マルチエージェントシステムの動作の効率性が向上する。また、提案する ARTISTE システムアーキテクチャの中立性、汎用性から別のネットワーク管理システムに対しても同等の効果をもたらすことが期待できる。

5.2 エージェント間協調動作管理技術

マルチエージェントによる適応型ネットワーク管理技術を構成するための課題 2 に対して、異なる AS 上に配置されたエージェント間で行われる協調動作に対して、各配置先 AS の持つポリシーを反映した動作管理を行うことでマルチエージェントシステムの安全な適用を実現するエージェント間協調動作管理技術を明らかにした。

まず、実際のネットワーク管理作業を調査した結果を元に、広域ネットワーク管理においてマルチエージェントシステムの協調動作に伴って生じる可能性がある問題点として AS 内部のネットワーク情報が他 AS に流出する可能性があることを明らかにした。

その上で、実際のネットワーク管理作業を行う ISP からのヒアリングに基づき、エージェント間で行う情報共有の方式として「信頼できる相手によって構成される特定グループ内での情報共有」と「信頼度の異なるエージェント毎に異なる方式や内容で情報共有」という 2 方式が求められていることを明らかにした。

それぞれの要求条件を満たすために「エージェントグループに基づく情報共有方式」および「個々のエージェントの信頼度に基づく情報共有方式」を提案した。前者は同一の属性を持つ複数のエージェントをグループとして管理することにより、系列会社の運用するエージェントのみと協調するなどの動作管理を実現する。後者は各エージェントが他のエージェントに対する信頼度をデータとして設定し、通信の際に双方の信頼度を考慮することによって協調の有無やその内容を動的に制御することを可能とする。信頼度を双方で交換することによって情報共有の不公平性を解消する機構や 2 者以上のエージェントが通信に関与する場合においても不当に情報が流出する恐れのない通信プロトコルを提案した。さらに、これらの提案する技術を組み込んだネットワーク管理用マルチエージェントシステムを試作して実験環境で動作させて評価を行い、本技術の有効性を明らかにした。

本技術は、マルチエージェントシステムを実際のネットワーク管理に適用する際に考慮すべき問題の一つであるエージェントを通じた情報流出の問題を解決するための技術である。この問題は従来の関連研究においては十分な検討がなされてきておらず、マルチエージェントシステムの実作業への適用に向けた障害となっていた。

本技術によって、エージェントの動作に運営者の持つポリシーや意図を正確に反映することができるようになるため、エージェントが運営者の意図に反した動作をすることで秘密情報の流出やセキュリティ上の脅威を引き起こすことを防止できる。提案した 2 つの情報共有方式は特定のアプリケーションに依存しない汎用的なものであるため、将来開発される新たなマルチエージェントシステムへの組み込みも容易であると期待できるため、マルチエージェントシステムによるネットワーク管理の普及に向け大きな効果が期待できる。

5.3 今後の展望

本論文では、マルチエージェントシステムによって広域ネットワーク管理を行う際の基盤技術であるエージェントの組織化支援技術とエージェント間協調動作管理技術の確立のための検討を行った。これらの技術はそれ自体が具体的なネットワーク管理作業を行っているものではないが、エージェントの組織化支援技術はマルチエージェントシステムの動作の効率性および環境変化への適応性の向上、エージェント間協調動作管理技術はマルチエージェントシステムのネットワーク管理への安全な適用を促すという

点で、どちらもマルチエージェントシステムによる広域ネットワーク管理を実現する上で欠かせない。そのため、これらの技術をアプリケーションの動作を向上させ支援するための基盤技術としてプラットフォーム化したエージェントアーキテクチャの提案も併せて行った。今回提案した技術をエージェントプラットフォームとして共通基盤化することで複数のマルチエージェントシステムがエージェント組織化機能あるいは協調動作管理機能を利用することができる。このため、システムの動作に不可欠な機能を開発コストを掛けずに導入することが期待でき、結果としてマルチエージェントシステムによるネットワーク管理技術の導入に大きく寄与するものとする。

ただし、マルチエージェントシステム技術を実際のネットワーク管理業務に適用する際には十分な注意が必要であることも筆者らが行ってきた ISP などとの共同実験などから明らかになっている。本論文で提案したエージェント間協調動作管理技術はそのような問題の一つである情報流出の防止に寄与する技術であるが、もう一つの提案技術であるエージェント組織化支援技術および当該技術を実装した ARTISTE システムについても実際の適用については十分な注意が必要であると考えられる。ARTISTE はネットワーク全体から広くエージェントを探索するシステムだが、運営者によっては自己のエージェントの存在や状態などに関する情報を公開することを望まない場合も考えられる。今後、ARTISTE の実用化に向けてはこれらの問題を考慮した上でシステムの改良を行っていくことも必要であると考えている。

本論文では提案技術の検討に際して、筆者らが開発したマルチエージェントシステムなどいくつかの代表的なマルチエージェントシステムを対象にし、それらに共通的に必要な機能の抽出を行うアプローチを採った。今回実現した 2 つの機能はほとんどすべてのネットワーク管理用マルチエージェントシステムにおいて必要と考えられる代表的な機能であるが、もちろん、これらの機能だけで必要十分というわけではない。例えば、今回提案した ARTISTE では AS 間という広域トポロジ情報を利用したエージェント探索機能を実現したが、アプリケーションの種類や適用範囲によってはルータレベルのより詳細なトポロジ情報を使った探索が必要になる場合もある。

このように、本論文で行った検討は実際の広域ネットワーク管理へのマルチエージェントシステム適用に向けての足がかりとなるものである。そして、マルチエージェントシステムの適用による広域ネットワーク管理の高度化という目的を達成するために、今後も実際のネットワーク管理作業から実際的な課題抽出を行いながら実用的なネットワーク管理システムの開発を行うとともに、多種多様なマルチエージェントシステムに利用可能な機能を提供する汎用性の高いエージェントプラットフォームの実現という 2 方向からの検討を続けていく予定である。

謝辞

本研究は、筆者が NTT ソフトウェア研究所, NTT 未来ねっと研究所, 早稲田大学情報生産システム研究科において, 1999 年から行ったものです。本論文執筆に関して懇切なご指導, ご鞭撻を賜った早稲田大学大学院情報生産システム研究科教授 小柳恵一博士に心からの深謝の意を表します。

上司として本研究を行なう機会を与えて頂いた京都大学 情報学研究科 通信情報システム専攻 守倉 正博 教授, NTT 未来ねっと研究所 萩本 和男 所長, 滝川 好比郎 部長, 藤井 哲郎 前部長, 丸山 充 グループリーダーに深く感謝致します。

また研究遂行にあたり, 日々, 有益なご討論, ご助言を頂いた, NTT 未来ねっと研究所 特別研究員 明石修 氏, 早稲田大学 菅原俊治 教授, 国立情報学研究所 福田健介 准教授, 豊橋技術科学大学 廣津登志夫 准教授, 大阪大学 栗原聡 准教授をはじめとする NTT 未来ねっと研究所, 早稲田大学大学院小柳研究室の方々に心から感謝申し上げます。

最後に, 家族をはじめ多くの方々の協力を得て研究成果をまとめることができましたことを深く感謝いたします。

目次

図 1-1 各章の関わり	4
図 2-1 ネットワーク管理の流れ.....	8
図 2-2 ネットワーク動作における AS 間の関連.....	10
図 2-3 マルチエージェントシステムの動作	13
図 2-4 ENCORE における協調エージェントの管理.....	14
図 2-5 エージェント間での利害の競合の例	18
図 2-6 AS の要求が競合する例	20
図 3-1 本章の位置づけ	25
図 3-2 ENCORE による経路診断の例.....	28
図 3-3 動画配信アーキテクチャ	30
図 3-4 役割の複製.....	32
図 3-5 役割の委譲.....	33
図 3-6 役割の交換.....	34
図 3-7 単一のエージェントによるネットワーク情報の収集.....	35
図 3-8 個々のマルチエージェントシステムによって組織化が行われる場合.....	37
図 3-9 提案するエージェントアーキテクチャ	40
図 3-10 エージェントによる部分的なネットワーク情報の観測.....	43
図 3-11 エージェントの探索手順	47
図 3-12 システム構成図.....	48
図 3-13 ARTISTE エージェントソフトウェア構成図.....	49
図 3-14 実験ネットワーク構成図.....	51
図 3-15 実験ネットワーク構築用 PC	51
図 3-16 AS トポロジの例.....	58
図 3-17 AS の接続数の分布	59
図 3-18 把握率の変化	60
図 4-1 本章の位置づけ	63
図 4-2 特定のグループ内での情報共有	67
図 4-3 グループによる情報共有の実現イメージ	68
図 4-4 信頼関係に応じた動的な情報共有.....	69
図 4-5 グループ通信の流れ.....	71
図 4-6 信頼度を用いた二者間通信の例	74

図 4-7 エージェントによる情報共有の不公平性.....	75
図 4-8 三者通信における問題	76
図 4-9 エージェント構成図.....	79
図 4-10 システム構成図.....	80
図 4-11 三者間通信の動作例	83
図 4-12 実験ネットワーク構成図.....	84

表目次

表 2-1 ネットワークの規模による違い	9
表 2-2 マルチエージェントシステムの運用形態による協調の違い	17
表 3-1 マルチエージェントシステムの組織化において用いる情報.....	38
表 3-2 探索条件に用いる ATTRIBUTE 節.....	44
表 3-3 協調エージェントに関する設定(ARTISTE 利用無).....	52
表 3-4 協調エージェントに関する設定(ARTISTE 利用時).....	52
表 3-5 ARTISTE およびマルチエージェントシステムの開発規模	54
表 3-6 ARTISTE の応答時間	54
表 3-7 AS の接続数の分布	59
表 4-1 信頼度の関係	81
表 4-2 エージェント B,C のタスクの実行レベルリスト.....	81

参考文献

- 【1】 Calvert, K. L., & Griffioen, J. (2006). On Information Hiding and Network Management. *SIGCOMM '06*. Pisa, Italy: ACM.
- 【2】 Condie, T., Kamvar, S. D., & Garcia-Molina, H. (2004). Adaptive Peer-To-Peer Topologies. In *Peer-to-Peer Computing* (pp. 53--72).
- 【3】 D.A.Tran, K.A.Hua, & S.Sheu. (2003). A New Caching Architecture for Efficient Video-on-Demand Services on the Internet. *IEEE Symposium on Applications and the Internet (SAINT 2003)*, (pp. 172-181).
- 【4】 E.H.Durfee, V.R.Lesser, & D.D.Corkill. (1987). Cooperation Through Communication in a Distributed Problem. In M. Huhns, *Distributed Artificial Intelligence*. Pitman.
- 【5】 Faloutsos, M., Faloutsos, P., & Faloutsos, C. (1999). On power-law relationships of the internet topology. *ACM SIGCOMM Computer Communication Review*, 29(4), 251--262.
- 【6】 gavalas, D., Greenwood, D., Ghanbari, M., & O'Mahony, M. (2002). Hierarchical network management: a scalable and dynamic mobile agent-based approach. *Computer Networks*, 38(6), 693--711.
- 【7】 Hua, K. A., Tran, D. A., & Villafane, R. (2003). Overlay Multicast for Video on Demand on the Internet. *ACM SIGAPP Symp. on Applied Computing (SAC 2003)*, (pp. 935--942).
- 【8】 Ishida, T., Gasser, L., & Yokoo, M. (1992). Organization Self-Design of Distributed Production Systems. *IEEE Transactions on Knowledge and Data Engineering*, 4(2), 123--134.
- 【9】 Ishida, T., Yokoo, M., & Gasser, L. (1990). An Organizational Approach to Real-Time Production Systems. *The 1990 National Conference on Artificial Intelligence (AAAI-90)*, (pp. 52--58).
- 【10】 Lv, Q., Ratnasamy, S., & Shenker, S. (2002). Can Heterogeneity Make Gnutella Scalable? In *Lecture Notes In Computer Science (Revised Papers from the First International Workshop on Peer-to-Peer System)* (Vol. 2429, pp. 94--103).
- 【11】 M.M.Hefeeda, B.K.Bhargava, & D.K.Y.Yau. (2004). A hybrid architecture for cost-effective on-demand media streaming. *Computer Networks*, 44(3), 353-382.

- [12] M.Sasabe, N.Wakamiya, M.Murata, & H.Miyahara. (2003). Scalable and Continuous Media Streaming on Peer-to-Peer Networks. *The third International Conference on Peer-to-Peer Computing (P2P'03)*, (pp. 92-99).
- [13] O.Akashi, A.Terauchi, K.Fukuda, T.Hirotsu, M.Maruyama, & T.Sugawara. (2005). Detection and Diagnosis of Inter-AS Routing Anomalies by Cooperative Intelligent Agents. In *Lecture Notes in Computer Science(Proc. of DSOM'05)* (Vol. 3775, pp. 181-192).
- [14] *Quagga Routing Suite*. (n.d.). Retrieved from <http://www.quagga.net>
- [15] Ratnasamy, S., Handley, M., Karp, R., & Shenker, S. (2002). Topologically-Aware Overlay Construction and Server Selection. *IEEE Infocom 2002*, (pp. 1190--1199).
- [16] Rekhter, Y., & Li, T. (1995). A Border Gateway Protocol 4 (BGP-4). *RFC1771*.
- [17] Rowstron, A., & Druschel, P. (2001). Pastry: Scalable, decentralized object location, and routing for large-scale peer-to-peer systems. In *Lecture Notes in Computer Science* (Vol. 2218/2001, pp. 329--350).
- [18] S.Willmott, & B.Faltings. (2000). The Benefits of Environment Adaptive Organisations for Agent Coordination and Network Routing Problems. *ICMAS2000*, (pp. 333-340).
- [19] *SETI@home*. (n.d.). Retrieved from <http://setiathome.berkeley.edu/index.php>.
- [20] Stoica, I., Morris, R., Karger, D., Kaashoek, M. F., & Balakrishnan, H. (2001). Chord: A scalable peer-to-peer lookup service for internet applications. *SIGCOMM '01* (pp. 149--160). ACM Press.
- [21] Tambe, M., Pynadath, D. V., & Chauvat, N. (2000). Building dynamic agent organizations in cyberspace. *Internet Computing*, 4(2), 65-73.
- [22] Touch, J. (2001). Dynamic Internet Overlay Deployment and Management Using the X-bone. *Computer Networks*, 36(2), 117-135.
- [23] Tran, D. A., Hua, K. A., & Do, T. T. (2003). ZIGZAG: An Efficient Peer-to-Peer Scheme for Media Streaming. *IEEE INFOCOM 2003*, (pp. 1283--1292).
- [24] Tu, Y.-C., Sun, J., Hefeeda, M., & Prabhakar, S. (2005). An Analytical Study of Peer-to-Peer Media Streaming Systems. 1(4), 354--376.
- [25] Willmott, S., & Faltings, B. (1999). Active Organisations for Routing. In *Lecture Notes in Computer Science(Proc. of IWAN'99)* (Vol. 1653, pp. 262-273).
- [26] Wooldridge, M. (2002). *An Introduction to Multiagent Systems*. John Wiley

& Sons, Ltd.

- 【27】 XML-RPC Home page. 参照先: <http://www.xmlrpc.com/>
- 【28】 Zhao, B. Y., Huang, L., Stribling, J., Rhea, S. C., Joseph, A. D., & Kubiawicz, J. D. (2004). Tapestry: a resilient global-scale overlay for service deployment. In *IEEE Journal on Selected Areas in Communications* (Vol. 22, pp. 41--53).
- 【29】 今野, 吉村, 羽鳥, 岩谷, 阿部, 木下. (2005). 能動化された情報資源に基づくネットワーク管理支援方式. *情報処理学会論文誌*, 46 (2), 493-505.
- 【30】 寺内敦, 明石修, 丸山充, 菅原俊治, 福田健介, 廣津登志夫, ほか. (2004). エージェントの組織化による広帯域ストリーム向け適応型配信アーキテクチャの提案. 第6回インターネットテクノロジーワークショップ(WIT2004).
- 【31】 内山, 梅津, 安本, 東野. (2006). 広域ネットワークにおける自律的なグループ形成機構を用いた分散ネットワークモニタ方式の提案. *情報処理学会論文誌*, 47 (2), 446-454.

研究業績一覧

論文

- 【A1】 寺内敦, 明石修, 丸山充, 菅原俊治, 福田健介, 廣津登志夫, 栗原聡, 小柳恵一 “広域ネットワークに対応したマルチエージェント組織化支援システム”, 人工知能学会論文誌, 22 卷 5 号 pp.482--492, 2007 年 6 月
- 【A2】 Terauchi, H. Suzuki "Multi-agent System for Efficiently Propagating Information through and across Conjunctive Communities", 情報処理学会論文誌, Vol.40, No.1, pp.197-207, Jan. 1999.
- 【A3】 明石修, 森保健治, 寺内敦 “カプセル化機構を用いた情報流通方式:FleaMarket” 情報処理学会論文誌, Vol.39, No.2, pp.458-465, 1998 年 2 月
- 【A4】 金井敦, 寺内敦, 山中顕次郎 “通信ソフトウェアの畳込開発法” 情報処理学会論文誌, Vol.34, No.5, pp.1166-1176, 1993 年 5 月

国際会議・シンポジウム (査読のあるもの)

- 【A5】 O. Akashi, A. Terauchi, T. Hirotsu, T. Sugawara “Lisp-based Platform and Applications for the Inter-domain Network Management” International Lisp Conference 2007, Cambridge, U.K., Mar., 2007.
- 【A6】 O. Akashi, A. Terauchi, K. Fukuda, T. Hirotsu, M. Maruyama, and T. Sugawara “Detection and Diagnosis of Inter-AS Routing Anomalies by Cooperative Intelligent Agents” Lecture Notes in Computer Science (Proc. of DSOM'05), Vol.3775, pp.181-192, Oct., 2005
- 【A7】 A. Terauchi and O. Akashi and M. Maruyama and K. Fukuda and T. Sugawara and T. Hirotsu and S. Kurihara, “ARTISTE: Agent Organization Management System for Multi-agent Systems”, 8th Pacific Rim Int'l Workshop on Multi-Agents (PRIMA'05), Kuala Lumpur, Malaysia, Sep., 2005.
- 【A8】 A. Terauchi, K. Yamanaka, J. Kato, “Efficient specification description for communications software using message sequences and rules”, IEEE GLOBECOM'94. , Volume 2, San Francisco, U.S., pp.805 – 810, Nov., 1994.

講演

- 【B1】 寺内敦, 明石修 "広域ネットワーク上のエージェントシステム管理手法の提案" 鹿児島, 日本, 第 14 回 マルチメディア通信と分散処理ワークショップ, pp.157-162, 2006 年 11 月
- 【B2】 寺内敦, 明石修, 丸山充, 福田健介, 廣津登志夫, 栗原聡, 菅原俊治"ネットワーク情報を利用したインターネットエージェントの配置" 沖縄, 日本, 第 13 回 マルチメディア通信と分散処理ワークショップ, pp.103-108, 2005 年 11 月
- 【B3】 寺内敦, 明石修, 丸山充, 福田健介, 栗原聡, 菅原俊治"経路障害診断システム用エージェント管理システムの提案", 第 12 回 マルチメディア通信と分散処理ワークショップ, 石川, 日本, pp. 299-304, 2004 年 12 月
- 【B4】 寺内敦, 明石修, 丸山充, 菅原俊治, 福田健介, 栗原聡, 廣津登志夫 "エージェントの組織化による広帯域ストリーム向け適応型配信アーキテクチャの提案" 石川, 日本, 第 6 回インターネットテクノロジーワークショップ(WIT2004), 2004 年 12 月
- 【B5】 寺内敦, 鈴木英明 " コミュニティを利用した情報伝達方式", 第 57 回後期情報処理学会全国大会, 名古屋, 日本, pp.465-466 , 1998 年 10 月
- 【B6】 寺内敦, 鈴木英明, " コミュニティを利用した情報伝達方式", マルチメディア, 分散, 協調とモバイルシンポジウム(DiCoMo '97), 北海道, 日本, 1997 年 7 月
- 【B7】 山本具英, 寺内敦, 曾根岡昭直 "電子商取引における電子割引券の提案", 1997 年電子情報通信学会総合大会, p.253, 1997 年 3 月
- 【B8】 森保健治, 明石修, 寺内敦 "情報流通システムにおける鍵配送通信の実装", 情報処理学会 マルチメディア通信と分散処理研究会報告, Vol.1996, No.108, pp.81-86, 1996 年 11 月
- 【B9】 寺内敦, 森保健治, 三宅延久, 明石修 "電子決済システムの実装と評価", 情報処理学会 マルチメディア通信と分散処理研究会報告, Vol.96, No.40, pp.97-102, 1996 年 5 月
- 【B10】 明石修, 森保健治, 三宅延久, 寺内敦 "FleaMarket 方式による情報流通システムの実装", 情報処理学会 マルチメディア通信と分散処理研究会報告, Vol.96, No.40, pp.103-108, 1996 年 5 月
- 【B11】 寺内敦, 森保健治, 明石修 "情報流通システムにおける課金方式" 第 4 回マルチメディア通信と分散処理ワークショップ, pp.251-258, 三重, 日本, 1995 年 10 月
- 【B12】 三宅延久, 奥山浩伸, 寺内敦, 金井敦 "CD-ROM 情報流通システムにおける情報オーサリングに関する一考察" 1995 年電子情報通信学会総合大会, p335, 1995 年 3 月
- 【B13】 寺内敦, 金井敦 "MSC とルールを併用した通信サービスの効率的開発法", 第 49 回後期情報処理学会全国大会, pp.263-264, 1994 年 9 月
- 【B14】 寺内敦, 山中顕次郎, 加藤順 "通信サービスにおけるエラー処理の自動生成", 第 45 回後期情報処理学会全国大会, pp.159-160, 1992 年 9 月

その他

NTT 機関誌

- 【C1】 寺内敦，明石修 “広域ネットワーク向けエージェント管理支援システム”，NTT 技術ジャーナル，2006年10月，Vol.18，No.10，pp.44-46
- 【C2】 明石修，森保健司，寺内敦 “インターネットを用いた情報流通プラットフォーム Infoket-I”，NTT R&D，1997年2月，Vol.46，No.2，pp.107-114.
- 【C3】 鈴木英明，三宅延久，清水亮博，寺内敦 “Infoket を用いた電子出版サービスの公正と評価”，NTT R&D，1997年2月，Vol.46，No.2，pp.115-123.

特許

- 【C4】 寺内敦，明石修 “ネットワーク管理システム，ネットワーク診断装置およびネットワーク診断プログラム”，特願 2007-150503，2007年6月
- 【C5】 寺内敦，明石修，丸山充 “ネットワーク管理システムおよび方法”，特願 2006-124612，2006年4月
- 【C6】 寺内敦，明石修，丸山充，福田健介，菅原俊治 “コンテンツ配信方法”，特願 2004-359341，2004年12月
- 【C7】 寺内敦，明石修，丸山充，福田健介，菅原俊治 “ネットワーク管理方法およびネットワーク管理システム”，特願 2004-346160，2004年11月
- 【C8】 寺内敦，桑名栄二，藪本康之 “課金決済業務代行手数料算出方法並びに同システム”，特許第 3800586 号，2006年5月
- 【C9】 佐藤直之，鈴木英明，寺内敦，桑名栄二，庵祥子 “証明書を用いたサービス提供方法”，特願平 11-374118，1999年12月
- 【C10】 寺内敦，鈴木英明 “情報取得伝達方法および装置”，特願平 9-231167，1997年8月
- 【C11】 森保健治，明石修，寺内敦 “暗号鍵内容を隠蔽した情報流通方法”，特願平 7-272521，1995年10月
- 【C12】 寺内敦，森保健治，明石修 “情報販売方法およびシステム”，特願平 7-274256，1995年10月
- 【C13】 明石修，森保健治，寺内敦 “情報流通方法およびシステム”，特願平 7-274324，1995年10月
- 【C14】 森保健治，明石修，寺内敦 “端末プログラム改ざん検出方法および情報センタ”，特許第 3393521 号，2003年1月
- 【C15】 中山隆二，金井敦，三宅延久，寺内敦 “情報の暗号化および復号化システム”，特願平 7-2415，1995年1月
- 【C16】 奥山浩伸，森保健治，金井敦，三宅延久，寺内敦 “暗号鍵配送システムおよび方法”，特許第 3348753 号，2002年9月
- 【C17】 奥山浩伸，森保健治，金井敦，三宅延久，寺内敦 “ソフトウェア解析保護方法”，特許第 3463963 号，2003年8月
- 【C18】 寺内敦，金井敦，三宅延久，森保健治，奥山浩伸 “情報流通装置および方法”，特許第 3412709 号，2003年3月

【C19】 寺内敦，山中顕次郎，加藤順 “メッセージシーケンス生成方法”，特願平 4-302288, 1992 年 11 月

【C20】 山中顕次郎，寺内敦，加藤順 “エラー処理合成装置”，特願平 4-282685, 1992 年 10 月