

TR-AC-0055

003

適応的QoS制御方式における市場モデルを用いた  
ネットワークリソース割り当て法の検討

小菅 昌克 萩野 長生\*

\*KDD研究所

2001. 3.16

ATR環境適応通信研究所



# 適応的QoS制御方式における市場モデルを用いた ネットワークリソース割り当て法の検討

小菅 昌克<sup>1</sup> 萩野 長生<sup>2</sup>

<sup>1</sup>ATR環境適応通信研究所

〒619-0288 京都府相楽郡精華町光台二丁目2番地2

E-mail: kosuga@acr.atr.co.jp

<sup>2</sup>KDD研究所

〒356-8502 埼玉県上福岡市大原2丁目1番15号

E-mail: ogino@kddlabs.co.jp

あらまし 我々は、既に、端末およびネットワーク環境の変化に応じて、ユーザ要求を反映したQoS制御を行うためのマルチエージェント技術を応用した適応的QoS制御方式を提案している。今回、適応的QoS制御方式におけるドメインごとのネットワークリソース割り当て法として、市場モデルの適用を検討した。その際、ドメイン内に無線で接続された端末があることを想定し、帯域制約のある端末を含む場合についても最適なネットワークリソース割り当てが実現できるように市場モデルの拡張を行った。価格を調整するネットワークエージェントをドメイン内に設け、ネットワークエージェントがドメイン内の端末の帯域制約を管理して、市場モデルに帯域制約を反映させる。

## A Study on Market-Based Network Resource Allocation in Adaptive QoS Management Scheme

Masakatsu KOSUGA<sup>1</sup> Nagao OGINO<sup>2</sup>

ATR Adaptive Communications Research Laboratories.

2-2-2 Hikaridai Seika-cho Soraku-gun Kyoto

619-0288 Japan

E-mail: kosuga@acr.atr.co.jp

KDD R&D Laboratories Inc.

2-1-15 Ohara Kamifukuoka-shi Saitama

356-8502 Japan

E-mail: ogino@kddlabs.co.jp

**Abstract** We have proposed an adaptive QoS management scheme based on multi-agent technology, which can control QoS autonomously according to fluctuation of system states and variation of user request. We will discuss how to apply the market model to network resource allocation in each domain under the framework of this adaptive QoS management scheme. We assume a network including terminals accommodated using wireless access link with a restricted bandwidth, and extend the conventional market model. A network agent to adjust price in the market manages the restriction of link bandwidth in the domain, and consider the restriction in the network resource allocation.



## 目次

	頁
1 研究の背景	1
2 適応的QoS制御方式におけるネットワークリソースマネージメント	1
3 市場競争モデルを用いたネットワークリソース割り当て法	
3.1 基本的な市場競争モデルに基づくネットワークリソース割り当て法	2
3.2 市場競争モデルの拡張	3
3.3 交渉手順	4
4 シミュレーション実験の準備	
4.1 需要関数	5
4.2 供給関数	7
4.3 価格の調整	8
4.4 交渉の終了条件	9
5 シミュレーション結果	
5.1 シミュレーション1	10
5.2 シミュレーション2	12
5.3 シミュレーション3	13
6 まとめ	15
参考文献	15



## 1. 研究の背景

今後、モバイル環境下において、ユーザのパーソナリティをより重視したマルチメディア通信の普及が予想される。このようなマルチメディア通信においては、時々刻々と変化する通信環境に応じて、ユーザに意識させることなく、通信システム側で自律的に、通信アプリケーションの機能・構成、使用リソース、通信形態を制御する仕組みが要求される。このような観点から、我々はメディアストリームのQoSマネージメントを中心に、端末およびネットワーク環境の変化に応じて、ユーザ要求を反映したQoS制御を行うためのマルチエージェント技術を応用した適応的QoS制御方式を提案している[1][2]。

これまでは、メディアストリームのネットワークリソースマネージメント方式として、ATMネットワークやインターネットにおけるRSVPのような保証型の方式を想定し、ネットワークのアドミッション制御を通して、ネットワークリソースの割り当てを行うことを想定していた。しかしながら、サブネットワーク内のネットワークリソースは複数の端末で共有されているため、個々の端末で利用可能なリソース限度量はネットワークの状態により変動する。そこで、ネットワークの状況に基づいて、個々の端末が利用できるリソース量を決定し、より柔軟にリソース割り当てを行うための端末とネットワーク間での交渉が必要になる[3]。さらにホームネットワークのように、個々のユーザ要求の把握が困難なネットワークにおいては、ユーザ端末における自律分散的な処理に基づいて、ネットワークリソース割り当てが実現されることが望ましい。そこで、今回、適応的QoS制御方式におけるネットワークリソース割り当て法として市場モデルの適用を検討し、帯域制約のある端末を含む場合についても最適なネットワークリソース割り当てが実現できるように市場モデルの拡張を行った。

2章では、適応的QoS制御方式におけるネットワークリソースマネージメントについて述べ、3章では、拡張された市場競争モデルに基づくネットワークリソース割り当て法について説明する。最後にまとめと今後の課題を述べる。

## 2. 適応的QoS制御方式におけるネットワークリソースマネージメント

複数のマルチメディアストリームを同時に利用するマルチメディアアプリケーションを実現するためには、各メディアストリームのQoSを決定し、そのQoSに基づいて、端末およびネットワークのリソースを適切に割り当てる必要がある。我々が提案している、適応的QoS制御方式のフレームワーク[1][2]では、アプリケーションごとに生成されたエージェント（アプリケーションエージェント：AA）が交渉を行うことでQoSを決定する（図1-a）。端末内および端末間交渉では、それぞれのストリームに対するQoSパラメータ値とQoSパラメータ値に対するユーザの満足度を示すユーティリティ値の関係を表すユーティリティ関数を利用して、リソース制約条件のもとで全体的なユーティリティ値を最大にするようなQoSパラメータ値を、各ストリームに対して選択する。

ここでリソース制約条件の対象となるリソースは、端末リソースとネットワークリソースが考えられる。具体的には、端末リソースとしてCPU使用率、ネットワークリソースとしてネットワーク帯域を想定する。端末リソースは端末内の全アプリケーションで共有されるため、その利用可能限度量は、例えば、100%（実際はOS自身のオーバーヘッドがあるためそれ以下になる）としている。ネットワークリソースについては、エンドエンドでの保証型リソース割り当てを仮定し、利用可能限度量を、ネットワークの

最大利用可能帯域としている。端末内および端末間交渉によって決定されたネットワークリソース帯域が、実際に利用可能か否かはアドミッション制御によって確認することを想定していた。

しかしながら、1つのドメイン内のネットワークリソースは複数の端末で共有されているため、個々の端末で利用可能なリソースの限度量はネットワークの状態により変動する。そこで、ネットワークの状況に基づいて、個々の端末が利用できるリソース量を決定し、より柔軟にリソースの割り当てを行うための端末とネットワーク間での交渉（図1-b）が必要になる[3]。

ドメイン内でのネットワークリソースマネジメントとして、ポリシー制御型ネットワーク (Policy-based networking : PBN) が提案されている[4]。PBNでは、管理者がネットワークを運用する際に予め知識や経験として持っている運用ポリシーをネットワークシステム内で保持し、その保持情報に従って、自律的に運用・管理を行っていく。PBNは、企業ネットワーク等の利用者のポリシーが明確であり、ネットワーク管理者が全体の統括を行うようなネットワークには有効であると考えられる。

それに対して、ホームネットワークのようなネットワークでは、ネットワークを利用する人々の行動の制約が少なく、その利用目的が多種多様であり、計画性がないことが考えられる。従って、明確なポリシーを決定することも困難であると考えられる。そのようなネットワークにおいて効率的なネットワークリソースの割り当てを行うためには、ユーザ端末の自律分散的な処理に基づき、ボトムアップ的にネットワークリソース割り当てを行う仕組みが必要となる。近年、分散資源割り当てを実現するためのメカニズムとして市場モデルが注目され[5]、ネットワークリソース割り当てへの適用も試みられている[6]。

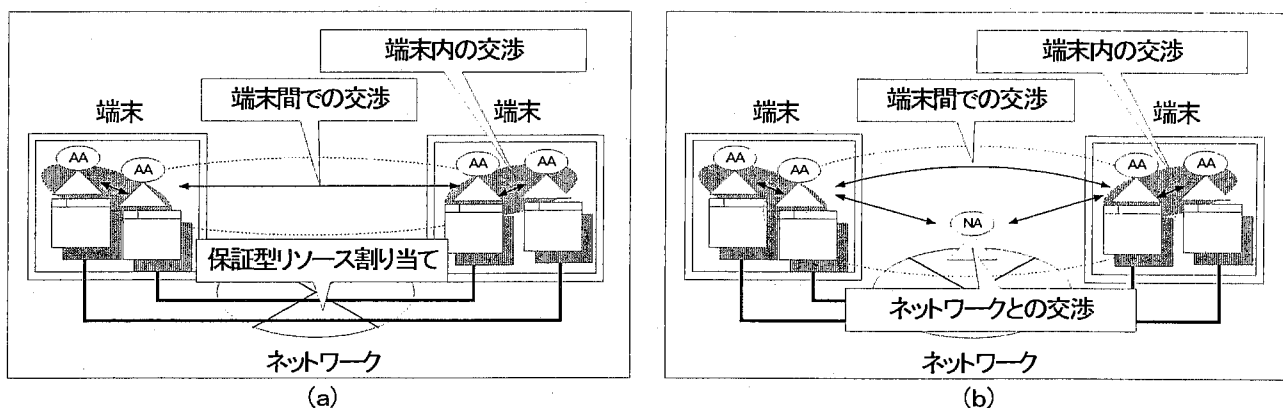


図1. 適応的QoS制御方式におけるQoS交渉モデル

### 3. 市場競争モデルを用いたネットワークリソース割り当て法

#### 3. 1. 基本的な市場競争モデルに基づくネットワークリソース割り当て法

市場モデルにおいて、各消費者が自らの消費者余剰利益（効用から財の購入費用を引いたもの）を最大にするように財の消費量を決定した場合、競争均衡においては、消費者余剰利益と生産者利潤の総和（社会厚生）が最大になることが知られている。この定理に基づく市場競争モデルを応用して、分散的にネットワークリソースを割り当てる手法が提案されている[7]。以下で詳しく述べる。

ストリーム  $s (s=1, \dots, n)$  が使用するネットワーク資源の量を  $x_s$  とし、その時の、ユーザのストリーム  $s$



に対するユーティリティを  $u_s(x_s)$  で表す。ネットワーク全体での資源の消費量の総和を  $X \equiv \sum_s x_s$  とし、ネットワーク資源が  $X$  だけ消費されたときに生じる遅延等のコストを関数  $c(X)$  で表す。この時、ユーザの満足度は、 $u_s(x_s) - c(X)$  で表される。ここで、ユーザ全体の満足度を最大にするように各  $x_s$  を決定したい。つまり、

$$\max_{\{x_s; s=1, \dots, n\}} \sum_{s=1}^n (u_s(x_s) - c(X)) \quad (1)$$

を解くことになる。この問題を、市場競争モデルを導入して分散的に解くため、資源の単位量あたりの価格を表すパラメータ  $p$  を導入すると、ユーティリティ関数  $u_s(x_s)$  およびコスト関数  $c(X)$  と価格  $p$  との関係は、

$$\frac{du_s(x_s)}{dx_s} = p \quad (2)$$

$$p = n \frac{dc(X)}{dX} \quad (3)$$

で表される。 $n$  はストリームの数である。それぞれを解く事で、ストリーム  $s$  の価格  $p$  での需要量を表す需要関数  $x_s = D_s(p)$  および、資源の供給量を表す供給関数  $X = S(p)$  を得る。各端末では、需要関数に基づき各自の需要量を分散的に計算する。交渉管理システムは、全ストリームの需要量の総和と供給量が一致するまで、価格  $p$  を動的に調整する。

### 3. 2. 市場競争モデルの拡張

前述の市場競争モデルを、我々の適応的 QoS 制御方式におけるネットワークリソース割り当てへ適用する。文献[7]では、価格  $p$  を動的に調整する交渉管理システムは、各端末内に存在し、端末間でのリソースの調整には予約型の制御が行われており、市場モデルは利用されていない。そこで我々は、価格を決定するネットワークエージェント (NA) をドメイン内に 1 つ設定し、市場モデルを端末間のリソース調整まで適用する (図 2)。また、我々は、ドメイン内に無線リンクによって接続される端末が存在することを想定する (図 2 の端末 3)。有線に比べると無線の帯域は狭く、帯域のギャップが存在する。例えば、ネットワークに PHS でダイヤルアップ接続する場合、無線リンクの帯域は 6.4 kbps となり、有線系と比較して大きな帯域のギャップとなる。このような帯域制約は、その通信リンクに接続された端末が知っている。そこで、端末側に通信リンクを管理するエージェントとして、ネットワークインターフェースエージェント (NIA) を設ける。NIA はユーザ要求を保持する AA と連携し、NA と市場モデルに基づいてネットワークリソース割り当てに関する交渉を行う。

帯域制約のあるようなネットワークでは、それらの制約を反映せずに交渉を行えば、結果として無線リンクに対してリンク帯域以上の帯域を要求してしまう可能性がある。これを防ぐために、市場モデルにそれらの制約を反映させる仕組みが必要となる

その方法として、以下の 2 つが考えられる。

1. N I A側で制約を反映させる
2. N A側で制約を反映させる

N I A側で制約を反映させる場合、通信先のリンク帯域制約を通信元が知る必要がある。しかし、通信元が通信先の帯域制約を知るには、通信先に問い合わせる必要があり、各端末が帯域制約を管理する方法では、手順が煩雑になる。一方、N Aがドメイン内のすべての端末に関する帯域制約を知るためには、端末がドメインに接続された時や通信帯域が変化した時（例えば、有線から無線へと繋ぎ変えた場合）に、N I AがN Aに通信帯域を通知すればよい。そこで、今回は、N A側で制約を反映させることとする。

制約をコスト関数に反映することも可能であるが、その場合、通信リンクごとにコスト関数を用意する必要があり、交渉モデルは複雑となる。そこで、コスト関数は共通帯域（図2の有線部）のもの1つとし、制約は交渉の打ち切りという形で反映する。つまり、価格の初期値を十分大きな値に設定し、常に下がる方向に新しい価格が調整されるようにする。そしてN Aが现阶段の需要に応じて新しい価格を決定し、それを全N I Aに通知する際、その段階で既にリンク帯域以上の帯域を要求した端末のN I Aに対しては、リンク帯域と等しい帯域を割り当てて交渉終了を通知する。

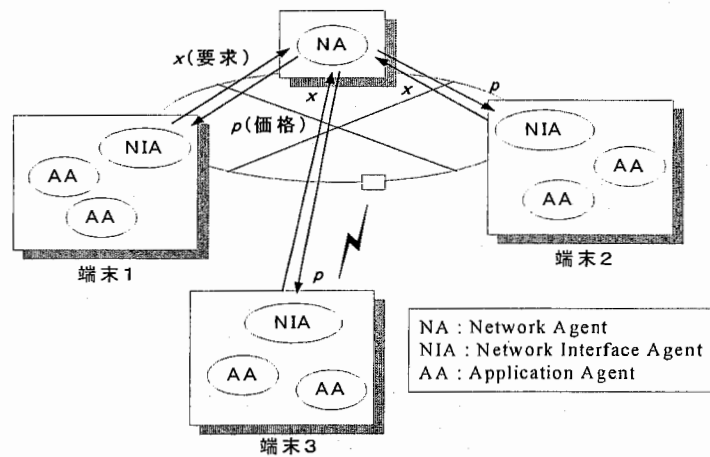


図2. 構成図

### 3. 3. 交渉手順

交渉は、1. アプリケーションが起動・終了した時、2. ユーザ要求が変化した時、3. ドメイン内の通信環境が変化したとき、に行われる。N AとN I AおよびA Aによる交渉手順は以下ようになる。ただし、ドメイン内の通信環境が変化した場合は、③から始まる。

- ① A AはN I Aに利用可能帯域を問い合わせる。
- ② N I AはN Aから価格の通知を受けた後、通知された価格に応じてリソース要求量を決定し、N Aに報告する。N Aは、
- ③ N Aは現在の需要に応じて新しい価格を決定し、全N I Aに通知する。その際、接続リンク帯域以上の帯域を要求する端末があり、かつ下げる方向に新しい価格を調整した場合には、その端末のN I Aには接続リンク帯域と等しい帯域を割り当てて交渉終了を通知する。
- ④ 価格が均衡するか、またはすべてのN I Aに交渉終了が通知されるまで、②～③が繰り返される。価格が均衡した場合には、N Aは残っているすべてのN I Aに割り当て帯域と共に交渉終了を通知する。

⑤ 交渉終了の通知を受けたN I Aは通知された帯域を利用可能帯域としてA Aに返答する。

#### 4. シミュレーション実験の準備

3章で述べた方式をシミュレーション実験にて検証を行う。まず、実験にあたり、

- 需要関数
- 供給関数
- 価格の調整方法
- 交渉の終了条件

についての詳細化と基礎実験を行う。その後、制約のある場合の実験結果を示す。

##### 4. 1 需要関数

(2)より需要関数を求めるには、ユーティリティ関数を決定する必要がある。様々なユーティリティ関数が想定可能であるが、規模による収穫の逓減という条件を満たす場合に競争均衡が存在することがわかっている[5]。そこで、この条件を満たす2つのユーティリティ関数を想定する。この2つのユーティリティ関数のグラフを図3に示す。

A) ユーティリティ関数1 :  $u_1(x)$

$$u_1(x) = u_{\max} - \frac{u_{\max}}{1 + \gamma x} \quad (4)$$

ここで、 $u_{\max}$ は最大ユーティリティ、 $\gamma$ は帯域の増加に対してユーザ満足がどれだけ向上するかを決定するパラメータである。この関数は、八槇氏の論文[9]におけるQoS生産者関数とも一致する。表1(a)に、 $u_{\max}=100$ 、 $\gamma=1/100$ の時の値を示す。これは、直感的にQoSパラメータセットを設定したものにQoSマッピングを行った結果[10](表2)と似通っている。

(2)、(4)式から $u_1(x)$ の需要関数 $D_{u_1}(p)$ は、

$$D_{u_1}(p) = \frac{1}{\gamma} \left( \sqrt{\frac{u_{\max} \gamma}{p}} - 1 \right) \quad (5)$$

となる。ただし、 $x$ が正になるのは、 $0 < p < 1$ の時である。

B) ユーティリティ関数2 :  $u_2(x)$

$$u_2(x) = a \cdot x^{(1/\beta)} \quad (6)$$

表1 (b) に、このユーティリティ関数の一例として、 $a=10$ ,  $\beta=3$ の時の値を示す。

(2), (6) 式から  $u_2(x)$  の需要関数  $D_{u_2}(p)$  は、

$$D_{u_2}(p) = \left( \frac{\beta}{a} p \right)^{\left( \frac{\beta}{1-\beta} \right)} \quad (7)$$

が得られる。

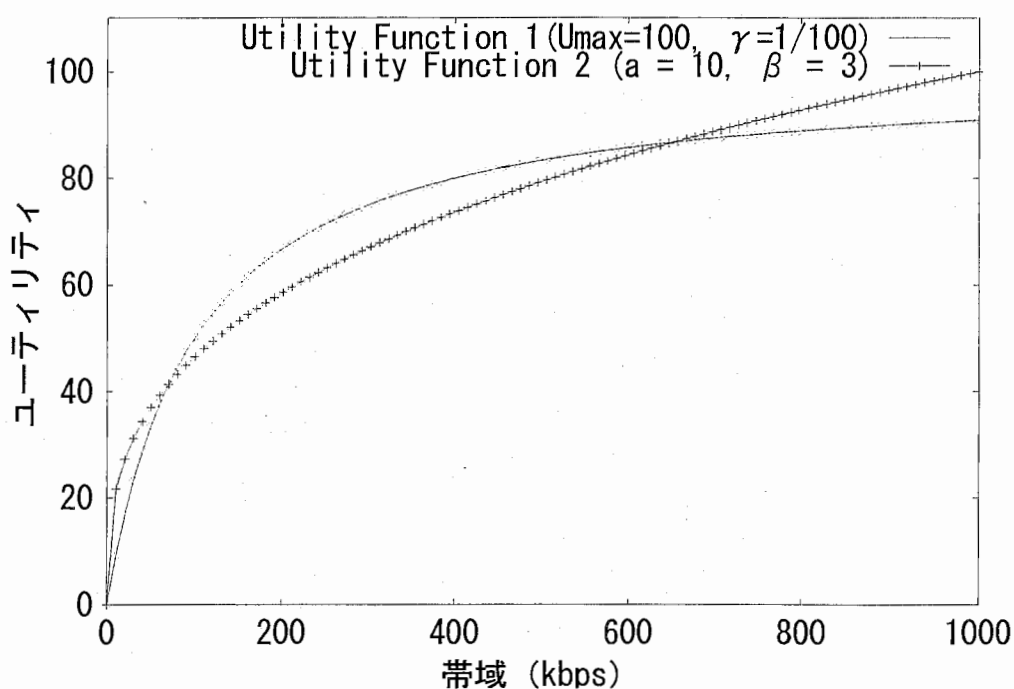


図3. ユーティリティ関数

表1. ユーティリティ関数の例

帯域	ユーティリティ
33.4	25.0
64.8	39.3
169.5	62.9
467	82.4
32555.9	99.7

(a) ユーティリティ関数1  
( $u_{max}=100$ ,  $\gamma=1/100$ )

帯域	ユーティリティ
33.4	32.2
64.8	40.2
169.5	55.3
467	77.6
1000	100

(b) ユーティリティ関数2  
( $a=10$ ,  $\beta=3$ )

表2. QoSパラメータセットの例

Candidate	Size	Frame rate	Quality	Bandwidth (Kbps)※	Utility
#1	640 x 480	12	90	32555.9	100
#2	320 x 240	8	90	467.0	82
#3	320 x 240	8	50	169.5	63
#4	320 x 240	3	50	64.8	39
#5	160 x 120	3	50	33.4	25

※ PentiumII 266MHz のマシンのデータからQoSマッピング[8]を用いて算出

#### 4. 2 供給関数

コスト関数は、割り当てられるネットワークの遅延などの特性を表したものである。今回は、コスト関数は単純なものを想定し、M/M/1 (∞) モデルでのパケット遅延時間を用いる。

$$c(X) = h \frac{X}{B - X} \quad (8)$$

Bはリソース帯域、hは平均サービス時間である。B=10240(kbps), h=1とした時のグラフを図4に示す。

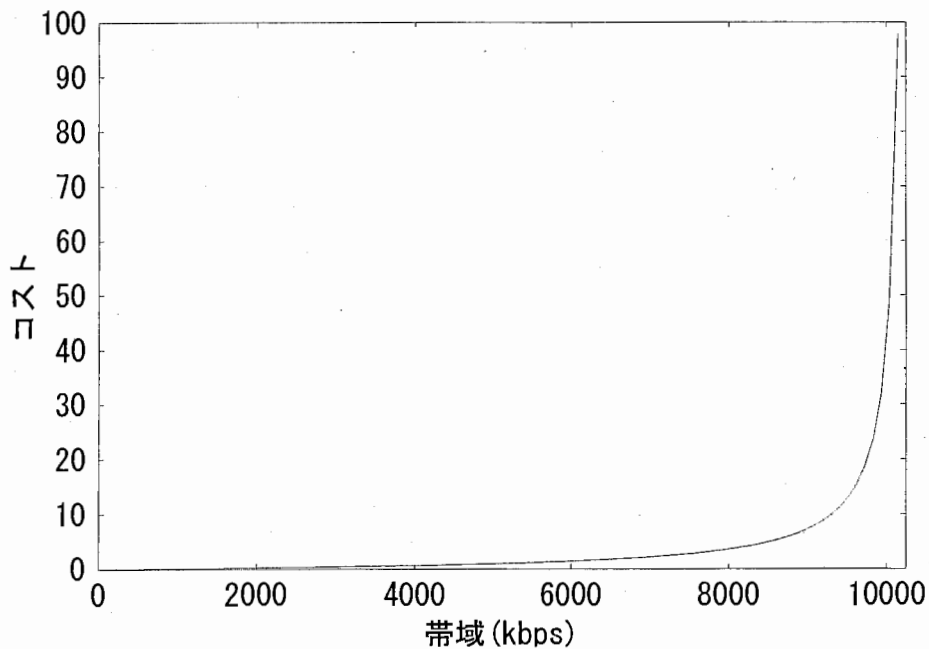


図4. コスト関数 (B=10240, h=1)

(3) (8) 式より供給関数は、

$$X = B - \sqrt{\frac{nhB}{p}} \quad (9)$$

となる。ただし、 $X$ が正になるのは、 $p > \frac{nh}{B}$ の時である。

今回の実験では、リソースの競合を起こりやすくするために $B=1024$ (kbps)とし、また単純化のために $h=1$ とした。

#### 4. 3 価格の調整

(5) もしくは(7)式と(9)式を用いて需要と供給が等しくなるように価格を調整する。つまり、

$$\sum_s D_s(p) = S(p)$$

となるように $p$ を調整する。このときの一回の交渉における $p$ の変化量は、

$$\frac{dp}{dt} = \varepsilon \left( \sum_s D_s(p) - S(p) \right)$$

とし、 $\varepsilon$ を適当に決めることで収束する。 $\varepsilon$ の式の形によって収束の速度が決まってくる。最悪の場合、収束点で振動してしまう可能性もある。特に今回の実験例は、

- 価格の初期値は十分大きく取るため、急速に収束点に近づかないと、収束は非常に遅くなることが予想される。
- 今回使用した需要関数と共有関数は、収束点で急激に変化している。(図5)

といった特徴をもつ。

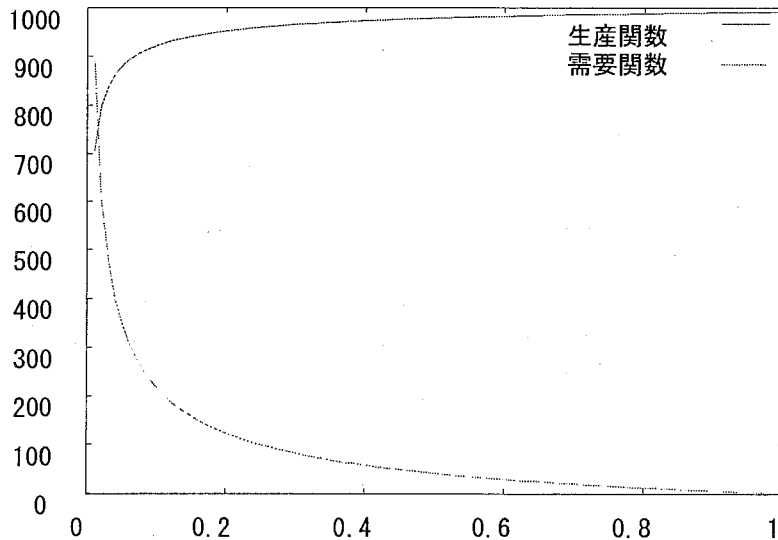


図5. 生産関数と需要関数

そこで $\varepsilon$ を

$$\varepsilon = \frac{a^* n}{\log(1+t)}$$

とした。 $a$ は定数、 $n$ は交渉リンク数、 $t$ は交渉回数である。何回かのシミュレーションの結果  $a=0.00002$  とした。

#### 4. 4 交渉の終了条件

総需要が利用可能帯域 ( $R_{max}$ ) 以下であり、総需要と供給の差が利用可能帯域のある割合 ( $A$ ) 以下となった場合、つまり、

$$\frac{\sum_s D_s(p) - S(p)}{R_{max}} < A$$

を満たす場合とした。

この  $A$  を決めるために、シミュレータ上で  $A$  の値を変化させて、交渉回数と総需要の変化を見た。 $A=0.01, 0.03, 0.05$  の交渉回数グラフ (図 6, 図 7) と総需要グラフ (図 8, 図 9) を以下に示す。

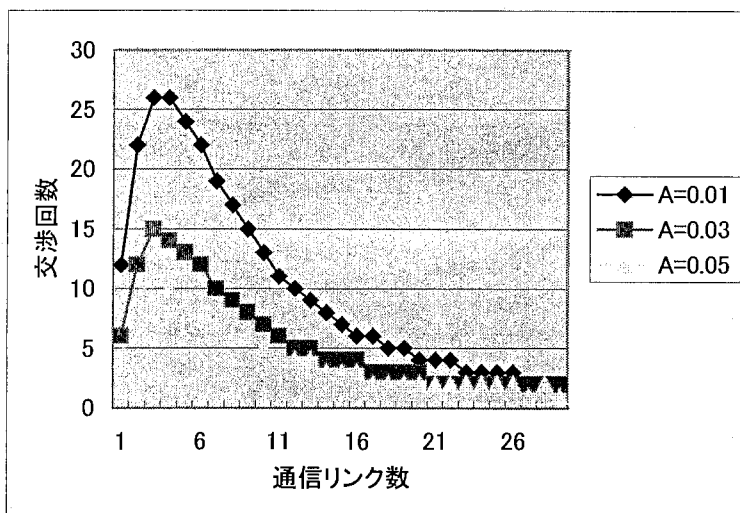


図 6. ユーティリティ関数 1 の交渉回数

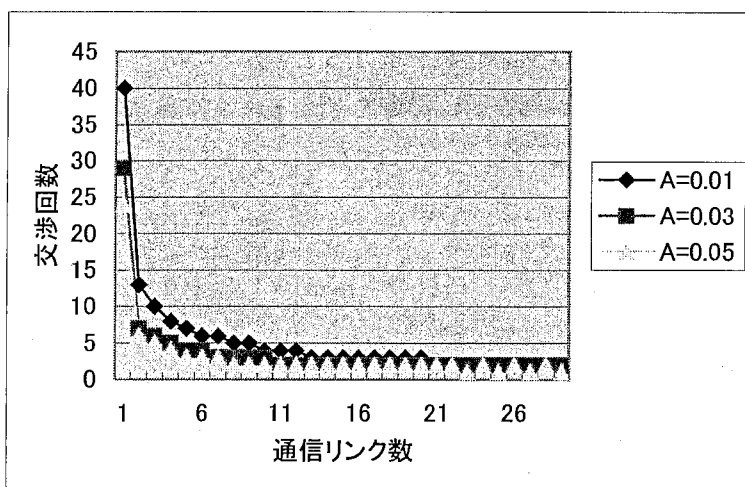


図 7. ユーティリティ関数 2 の交渉回数

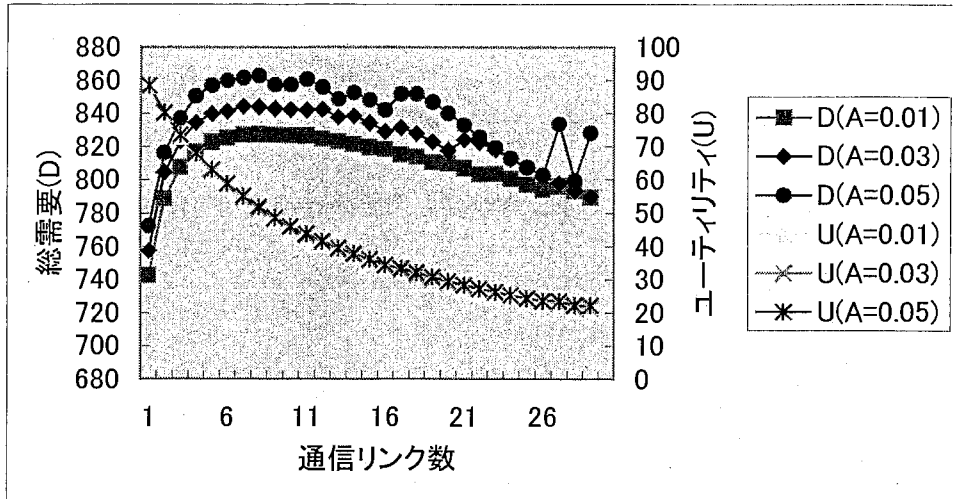


図 8. ユーティリティ関数 1 の総需要グラフ

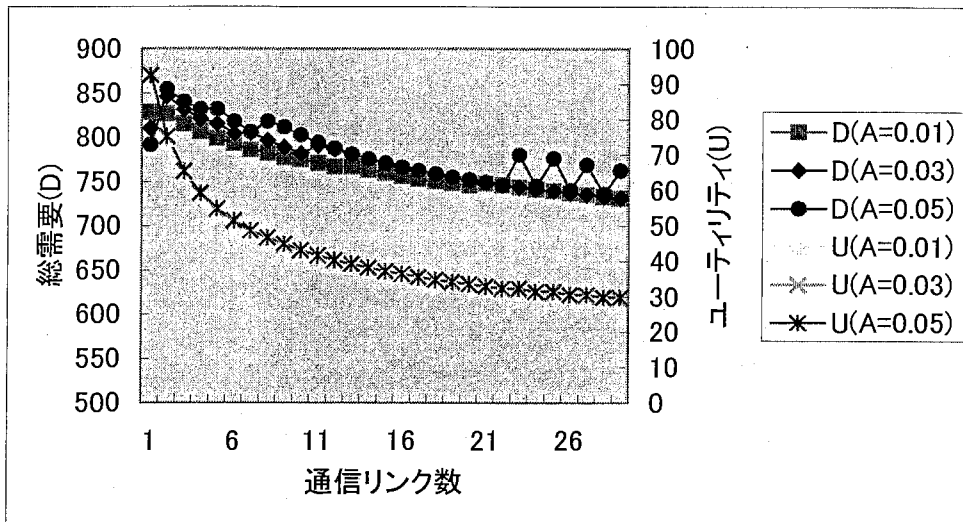


図 9. ユーティリティ関数 2 の総需要グラフ

図 6, 7 より、 $A$  の値が小さくなれば交渉回数が増えることがわかる。また、図 8, 9 より、 $A$  の値がある程度より大きくなると、その誤差が総需要に影響を与えるように見える。そこで今回の実験では、 $A = 0.03$  とした。

## 5. シミュレーション結果

### 5.1 シミュレーション 1

通信制約がある端末とない端末が 1 : 1 で存在するネットワークを想定し、通信リンクを一つずつ増やして交渉を行うシミュレーションを行った。ユーティリティ関数 1 での通信制約がある場合と通信制約がない場合の比較グラフを図 10, 11 に、ユーティリティ関数 2 での比較グラフを図 12, 13 に示す。



これらからわかるように、交渉回数は増えてしまうが、制約がある場合でも帯域を有効に割り当てられている。

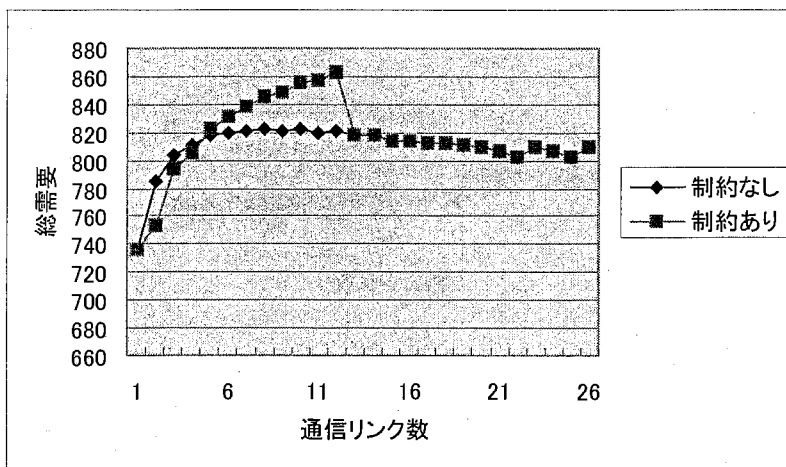


図10. ユーティリティ関数1での総需要

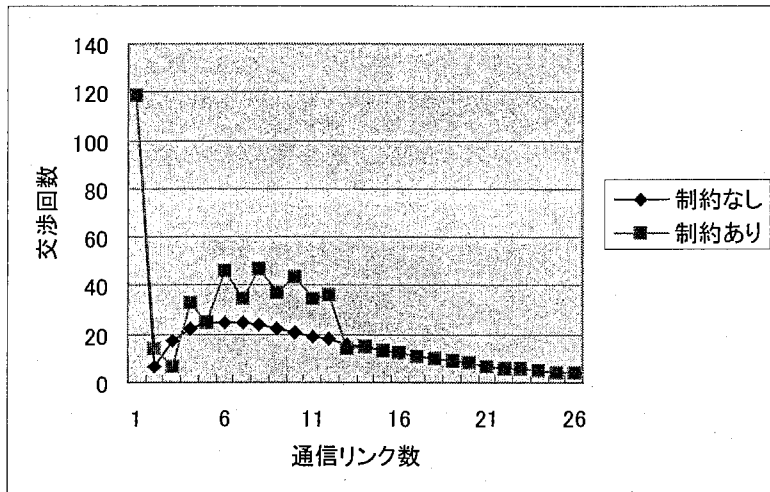


図11. ユーティリティ関数1での交渉回数

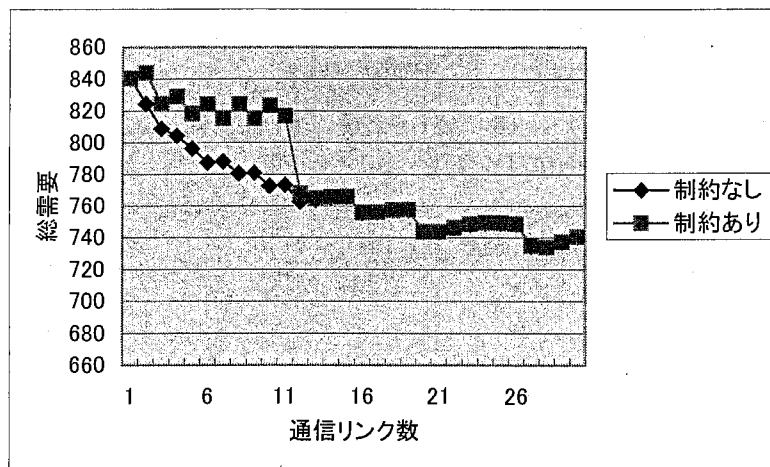


図12. ユーティリティ関数2での総需要

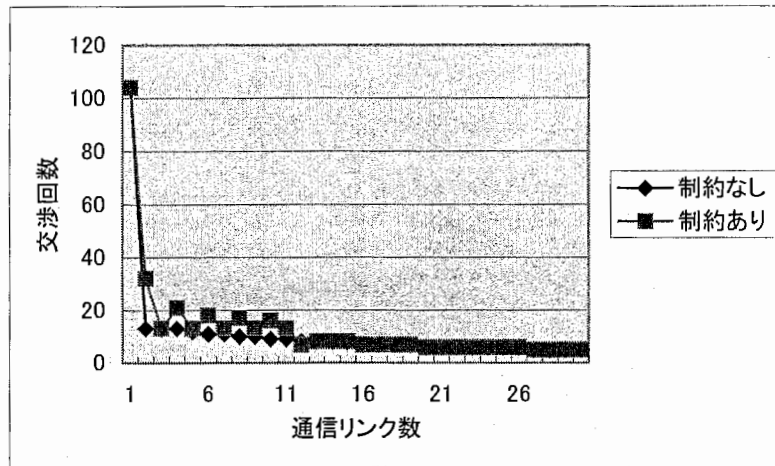


図13. ユーティリティ関数2での交渉回数

### 5.2 シミュレーション2

シミュレーション1と同様に通信制約がある端末とない端末が1:1で存在するネットワークを想定し、通信リンクを1つずつ増やして交渉を行うシミュレーションを行った。このシミュレーションでは、ユーティリティ関数1とユーティリティ関数2が混在している。この時の交渉回数と全端末でのユーティリティの総和のグラフを図14に、ユーティリティ関数ごとの需要量の変化を図15に示す。

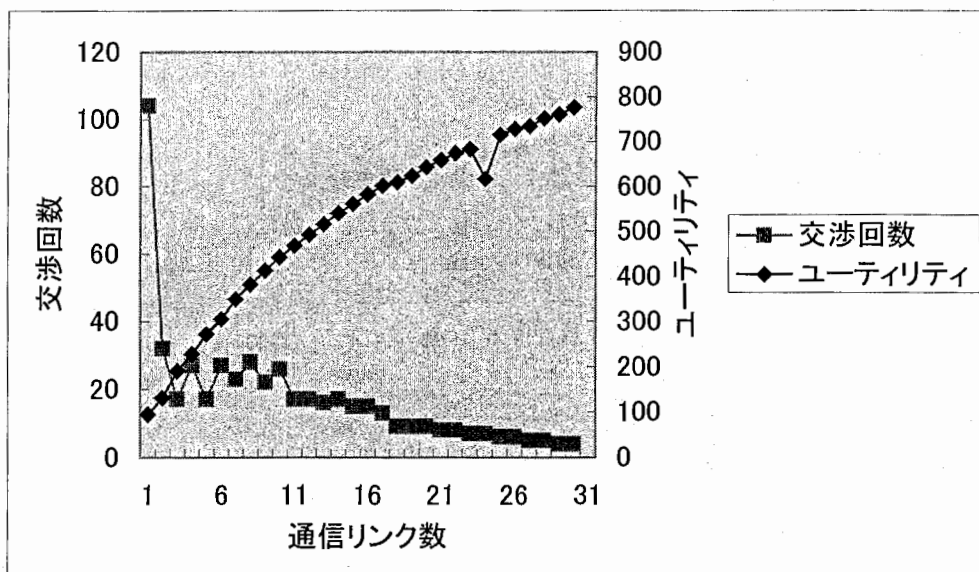


図14. 交渉回数とユーティリティ

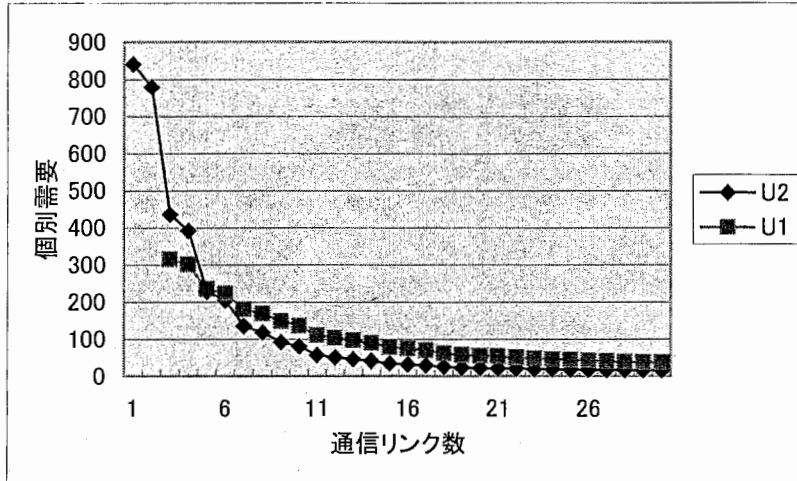


図 15. ユーティリティ関数ごとの需要量の変化

### 5.3 シミュレーション3

同一のユーティリティ関数ではあるがパラメータの違いを持つ端末が存在するネットワークを想定したシミュレーションを行った。パラメータの違いは、ユーティリティ関数が端末のリソース量からうける影響と考える。つまり、使用できるソース量の多い端末では、ユーザはよりよりQoSにてサービスを受けられると考えるために、リソース量の少ない端末よりユーティリティの値は低くなると想定する。例えば、ユーティリティ関数1において、端末リソースが多いケースとして $\gamma=1/100$ 、少ないケースとして $\gamma=1/50$ を想定する。

パラメータの違いを持つ端末が1:1で存在するネットワークを想定し、端末を1台ずつ増やして交渉を行い、最大30台まで行うシミュレーションを行った。結果を図16に示す。

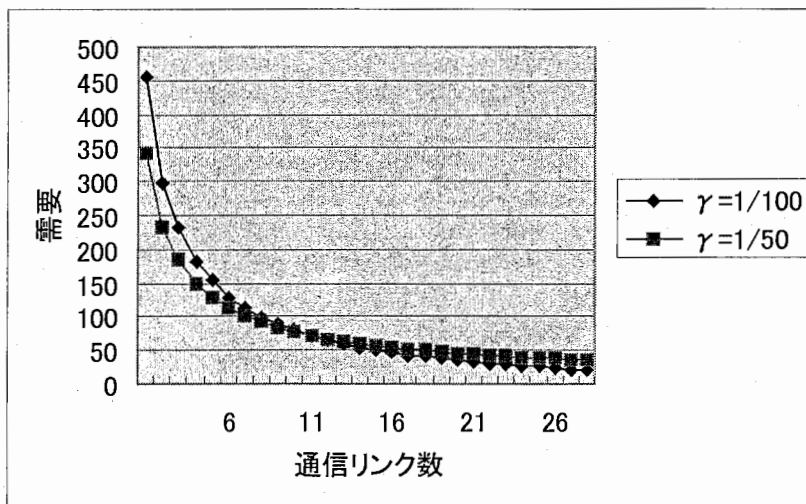


図 16. ユーティリティ関数1での比較

この場合、通信リンク数が13よりおおくなると、端末リソースが少ない端末に、多い端末より多くの通信帯域が割り当てられている。

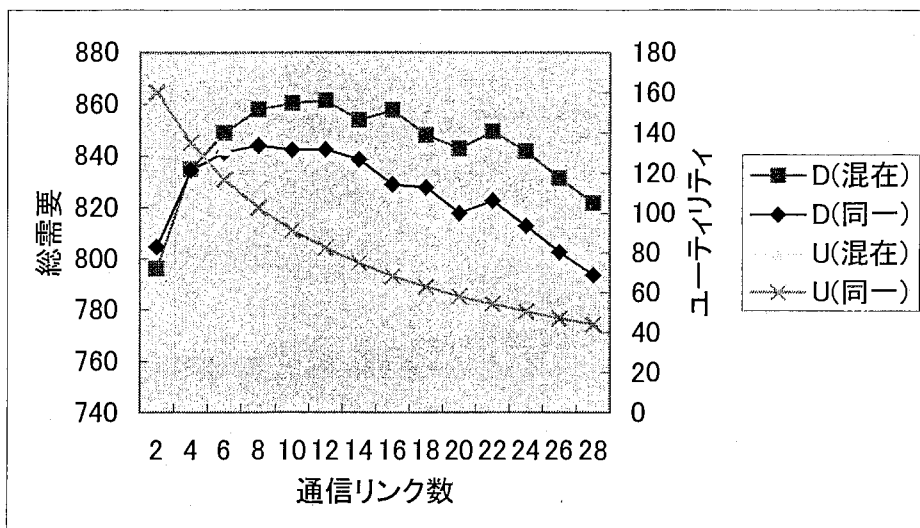


図17. ユーティリティ関数が混在時と同一時の比較

図17は、ユーティリティ関数が混在時と同一時の総需要とユーティリティのグラフである。ユーティリティは混在の場合は、異なるユーティリティ関数を持つ端末の、2つのユーティリティの和であり、同一の場合は各端末のユーティリティを単に2倍したものである。ネットワーク全体としては、より高いユーティリティを保持しており、総需要もより多くなっている。このことから、ユーティリティ関数の形によっては、リソース量の多い端末により多くのリソースを割り当てるのが、全体としてみたときに最適だとは言えないことがわかる。

ユーティリティ関数2で同様のことを行った。このユーティリティ関数では、端末リソースの大小を反映して帯域が割り当てられている。(図18)

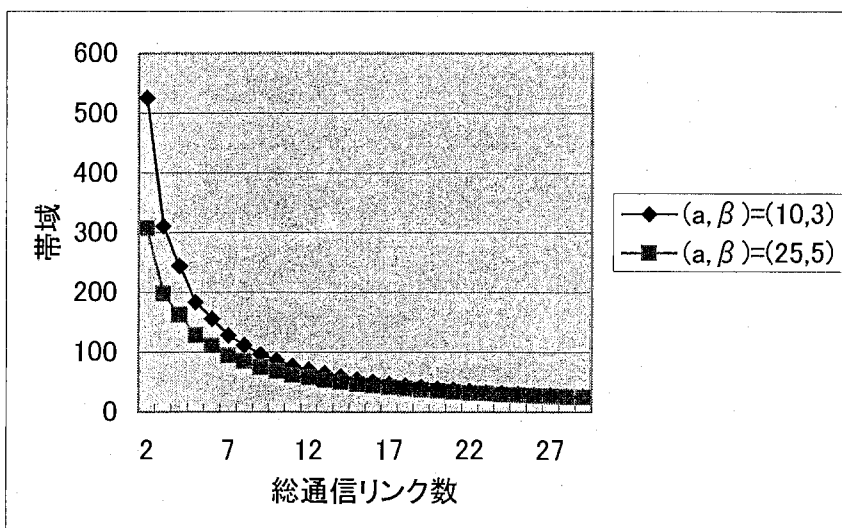


図18. ユーティリティ関数2での比較

## 6. まとめ

適応的QoS制御方式におけるネットワークリソース割り当て法として市場モデルの適用を検討した。また、ドメイン内に無線で接続された端末があることを想定し、帯域制約のある端末を含む場合についても最適なネットワークリソース割り当てが実現できるように市場モデルの拡張を行った。そこでは、各ドメイン内に価格を調整するネットワークエージェントが設けられ、このネットワークエージェントがドメイン内の端末の帯域制約を管理して、市場モデルに従った交渉手順の中に帯域制約を反映させる。この方式に基づいてシミュレーションを行い、基本的な動作の確認を行った。

今回のシミュレーションで用いた価格の調整関数は、今回適用したユーティリティ関数とコスト関数に依存しており、より一般的な調整関数もしくは調整メカニズムの検討が必要である。また、今後は、より一般的な条件でのシミュレーションを行っていく必要がある。

## 参考文献

- [1] 小菅, 山崎, 荻野, 松田, “マルチエージェントによる適応的QoS制御方式”, 信学会論文誌, Vol. J82-B No. 5, pp. 702-710, 1999
- [2] M. Kosuga, T. Yamazaki, N. Ogino, and J. Matsuda, “Adaptive QoS Management Using Layered Multi-Agent System for Distributed Multimedia Applications”, Proc. ICPP-99, pp. 388-394, Sept. 1999.
- [3] N. Ogino, M. Kosuga, T. Yamazaki, J. Matsuda, “A Model of Adaptive QoS Management Platform Based on Cooperation of Layered Multi-agents”, Globecom '99, 1999
- [4] R. Yavatkar, D. Pendarakis, R. Guerin, “A Framework for Policy-based Admission Control”, RFC2753, IETF, 2000
- [5] 石田, 片桐, 桑原, “分散人工知能”, コロナ社, 1996
- [6] 荻野, “市場経済モデルに基づく通信リソース割り当て”, 電子情報通信学会誌, Vol. 82 No. 9, pp. 967-976, 1999
- [7] 青木, 富川, “市場競争に基づく資源交渉システム”, 信学会論文誌, Vol. J80-B-I No. 6, pp. 408-415, 1997
- [8] 山崎, 松田, “スプラインニューラルネットワークによる動的QoSマッピング”, 信学総合大会, B-11-16, p. 549, 1999
- [9] 八槇, M.P. ウェルマン, 石田, “市場モデルに基づくアプリケーションQoSの制御”, 信学会論文誌, Vol. J81-D-I No. 5, pp. 540-547, 1998
- [10] 小菅, 山崎, 荻野, 松田, “適応的QoS制御方式を用いたマルチメディアアプリケーション”, 情報処理学会第59回全国大会演論文集, 3V-07, 1999

