

特別研究報告書

市場計算モデルを用いた
分散エージェントシステムの資源割り当て

指導教員 石田 亨 教授

京都大学工学部情報学科

沓 洋和

平成18年2月10日

市場計算モデルを用いた 分散エージェントシステムの資源割り当て

沓 洋和

内容梗概

現在, エージェントを用いた大規模なシミュレーションが行われるようになっていいる. また, 携帯電話でのモバイルエージェントの利用などが盛んになっていくと考えられる. これらは, 利用するエージェント数が多量になると複数のエージェントサーバで管理しなくてはならなくなる. こうした複数のエージェントサーバが分散して存在する環境の下で, 効率的にエージェントを動かすことが必要となる.

本研究は, エージェントシステムの利用者が所属している組織などによって, エージェントのグループ分けができる場合においてそのグループをコミュニティと呼ぶことにすると, コミュニティごとの目的に沿う形でエージェントを分散して配置したり, あるいは集中して配置したりすることが効率よくできる仕組みを構築することを目的としている.

既存の研究では, インタラクションの多さなどに着目して負荷を分散するという目的のために配置を効率化することが考えられている. しかし, 実際のシステムでは1つのサーバに集中して配置したい, もしくは逆に複数のサーバになるべく分散して配置したりしたい, などとコミュニティごとに異なる選好を持つと考えられる. あるいはエージェントサーバ側も負荷を少なくするためにインタラクション頻度が高いと考えられると考える同じコミュニティのエージェントをなるべく1つのサーバに集中して配置させたい, もしくは逆に複数のサーバに万遍なく配置したいという要求があると考えられ, それらを満たす必要がある.

本研究では, そうしたコミュニティごと, あるいはサーバごとに違う好みを反映しつつ効率の良いエージェント配置の仕組みを構築するという問題を解決するために, ミクロ経済学で用いられる一般均衡の理論を応用した計算機上に市場機構を構成する市場計算モデルまたは市場指向プログラミング (Market Oriented Programming) と呼ばれる手法の適用を提案する. 市場計算モデルはフロー制御やネットワーク QoS(Quality of Service) 制御や, 動画などのマルチメディアコンテンツのキャッシング制御に応用されている.

市場計算モデルを用いた場合の特徴として

- 市場の主体 (消費者または生産者) の選好 (preference) を効用関数として表すことができる。これにより、特定のサーバにエージェントを配置したいなどの選好を反映できる。
- 得られた解は、他者の効用を下げることなく自分の効用を上げることができないというパレート効率的な資源 (エージェントの配置) の配分となっており、無駄のない資源配分が可能である。
- 大規模なネットワーク上の運用を考えると、エージェントサーバごとに分散して計算を行うことが可能である。

といった点が挙げられる。

本研究ではこのような特徴に着目して先述の問題に対して市場計算モデルの適用を目指す、その際の課題として

- 本問題を市場計算モデルに当てはめたモデルの設計
- 本問題の意図するような選好を反映した効用関数の設計
- 実際のエージェントシステム上での実装の検討

が挙げられる。

これらの課題を解決するためのアプローチとして、市場計算モデルを実現するシミュレータを構成し、モデルの妥当性を検討するという方法を用いた。また、複数のエージェントサーバ間での運用を想定した実装を、追加の構成がを少なく実現できるような方法を考察し、以下のような成果を得た。

1. エージェントの配置問題に対し、コミュニティやエージェントサーバ毎に選好を反映する計算手法を提案した。
2. 市場モデルを用いた本問題のモデリングを行い、市場機構を実現したシミュレーションを行いモデルの妥当性を示した。
3. シミュレーションによりユーザの選好を実際に反映したエージェントの配置が得られることを示した。
4. 実際の分散エージェントサーバへの実装方法を提案した。

本研究によって、分散エージェントシステムでのエージェント配置問題に対し、コミュニティ(所属する組織などによるエージェント集合)の選好を反映しつつ、効率のよいエージェント配置を行う機構が実現可能となる。分散エージェントシステムを扱うアプリケーションでの決定手法の提案として情報学に貢献する。

Resource Allocation of Distributed Agent Systems based on Computational Markets

Hirokazu KUTSU

Abstract

Today, large-scale simulations by Massively Multi-Agent Systems have increased. In addition, it is thought that mobile-agent systems on portable telephones are more and more used. When the number of agents of these agent systems becomes large these agent systems must be composed of more than one agent servers. Such agent systems composed of distributed plural agent servers need to allocate the agents efficiently.

When it can group the agents with the organization and the like to which the user of the agent system has belonged, we define the group as community. The goal of this research is to construct a mechanism to allocate agent dispersed or centralized to satisfy the purpose of each community.

In existing research, optimizing the agent allocation to disperse its load has been studied. But, in the actual agent system, it is thought that each community has preference such as the community wants to allocate its agents dispersed to plural agent servers or centralized to only one agent server. Or the agent server may have the preference that it wants to allocate agents which belong to the same community centralized to one server. Because it is thought that their interaction frequency is high and allocating the agents to one server decreases the interaction cost.

In this research, to construct a mechanism to allocate the agents to satisfy those preferences different on each community or on each agent server efficiently, this thesis proposes the application of the technique called computational market or market oriented programming based on the general equilibrium theory on microeconomics to this problem. Market oriented programming is a technique constructing virtual market to solve resource allocation problem, and the technique has been applied to resource allocation such as network QoS(Quality of Service) and caches of multimedia contents.

This technique has the following characteristics.

- It establishes the method to define the preference of economic subjects

such as consumers or producers as utility function, and it can represent the preference such as it want to allocate agents to the particular agent server.

- The result given by this technique is Parete optimal. In Pareto optimal allocation, no consumer can improve its utility without decreasing the utility of other consumers.
- Thinking of the use on large-scale network, it is possible to calculate on dispersing every agent server.

Although this research proposes application of computational market to previously expressed problem, we have following problems to be solved.

- The design of the model which fits this problem in computational market.
- The design of utility functions which reflect the kind of preference which this problem is intended.
- Examination of mounting computational market on the actually agent system.

This research examines the validities of these models by forming the simulator that realizes the computational market as an approach to solve these problems. In addition, this research considers the method of actualizing the mounting which supposes the use between the group of agent servers with little addition of software. This research obtained following result.

1. The technique which reflects preference on every community and the agent server was proposed.
2. The modeling of this problem which uses the market model was done, the simulation which actualizes computational market was performed and the validity of the models was shown.
3. With the simulation, the fact that the allocation of the agent which really reflects the preference of the user can be obtained was shown.
4. Mounting method to the actually distributed agent server systems was proposed.

With this research, in agent allocation problem with distributed server system, we can obtain efficient agent allocation reflecting preference on every community and the agent server. This thesis contributes to information science by proposing a decisive method with applications which handle the distributed agent system.

市場計算モデルを用いた 分散エージェントシステムの資源割り当て

目次

第1章	はじめに	1
第2章	分散エージェントシステム	2
2.1	エージェントサーバとエージェント割り当て問題	2
2.2	エージェント集合を利用したエージェント配置	5
第3章	市場計算モデル	6
3.1	市場計算モデルと競争市場	7
3.2	競争均衡とワルラスの模索過程	7
3.3	消費者エージェント	10
3.4	生産者エージェント	10
3.5	競売人エージェント	11
3.6	市場計算モデルを用いたエージェント割り当て問題の解決	11
第4章	市場計算モデルを用いたシミュレーション	15
4.1	シミュレータの構成	15
4.2	シミュレーションの結果とその考察	17
第5章	実際のエージェントシステムへの実装の考察	20
第6章	終わりに	24
	参考文献	24
	付録：シミュレータのソースファイル	A-1

第1章 はじめに

現在, エージェントサーバシステムが様々な場面で用いられるようになってい
る. 災害時を想定した大規模な避難誘導を行うエージェントサーバシステムが
研究されているが, サーバ1台につき5万人誘導できたとしても, 100万人単位
の人が存在する都市では1台のサーバで誘導を行うことができないため, サーバ
同士をネットワークで結びつけて運用することが必要とされる. また, 携帯電話
などからインターネット上のリソースやサービス利用する際に, サーバへ必要
なリソースやサービスを自律的に取得することができるプログラムを送り通信
回数を減らしてネットワークのトラフィックを軽減することができるという, モ
バイルエージェントシステムの携帯電話への応用が考えられている. 携帯電話
と一対一に結びつくために扱うエージェント数が膨大になるため, 複数のサー
バをネットワーク上で運用することが考えられる.

災害時の避難誘導などは会社や学校など所属する組織などの単位でまとめて
行くと効率が良い. そうした場合, エージェントシステムでは組織などでグルー
プを設けてグループごとに効率的なエージェント配置を効率行える仕組みが必
要となる. モバイルエージェントでは同じリソースやサービスを利用するエー
ジェントならばリソースやサービスの結果を一部共有できるということがあれ
ば, 同じ興味を持つグループを作ればリソースやサービスを提供するサーバと
のインタラクションも減らすことができると考えられる. そのため, 避難誘導の
例同様にエージェントグループを設けての運用が効率的である.

これらのエージェントシステムの運用においてエージェントのグループをコ
ミュニティと呼ぶことにすると, コミュニティ毎にどのような配置が望ましいか
ということが異なってくると考えられる. 分散エージェントシステムの各サー
バに均等に分散して配置するのが良い場合もあれば, なるべく一つのサーバに
固まって配置するのが良い場合もあり, さらには特定のサーバに置きたい, ある
いは置きたくないといった選好が存在することが考えられる. また, サーバ側の
観点では同じコミュニティのエージェント同士ではインタラクションが多いと
考えられるのでなるべく同じサーバで扱いたいという選好が考えられる. こう
した選好に対し, 可能な限り多くの選好を満たす割り当てを見つけるという問
題が本研究の課題である.

本研究ではこうしたエージェントシステムでの資源の割り当て問題に対し, こ

うした選好を効用関数としてあらわし、市場計算モデルを用いて計算する方法を提案する。市場計算モデルとはミクロ経済学の一般均衡理論を応用した計算アルゴリズムであり、分散した資源の割り当て問題を解くことができ、トラフィック量の最適化 [1] やネットワーク QoS 制御 [2] や、動画などのマルチメディアコンテンツのキャッシング制御 [5] に応用されている。

市場計算モデルを用いた場合のこの問題のモデルの仕方を考え、シミュレーションを行うことでモデルの妥当性や配置の妥当性を検討する。その上で実際のエージェントサーバシステムへの実装を考察し、実効性について検討する。

第2章ではエージェントサーバシステムとその応用及びエージェントの割り当て問題について解説する。第3章では市場計算モデルと、その本題への適用について解説する。第4章では市場計算モデルを用いて本問題をシミュレーションし、その結果を考察する。第5章では実際のエージェントシステムへの応用を考察する。

第2章 分散エージェントシステム

本章では複数のエージェントサーバを用いた分散環境と、エージェントの割り当て問題について述べる。

2.1 エージェントサーバとエージェント割り当て問題

エージェントとは本来、自律的に状況を判断しながら与えられたシナリオに沿って目的を達成するために動作し続けるというものである。エージェントの一つの特徴としてユーザと1対1に結びついた動作をさせることにおいて、他のDBのみなどのアプリケーションサーバと比べてトランザクション処理の回数が減るために有用であると考えられている。この特徴を利用して、多数のユーザと1対1にサービスを行うことが有効な応用として期待されており、その一つの試みとして災害時の避難誘導を行うシステムがあげられる。また、携帯電話でのインターネット上のリソースやサービスの利用では、きちんとした通信プロトコルに沿った場合に通信コストや時間が多くかかってしまう。そこで、携帯電話から自分のほしいリソースやサービスを自律的に得てくれるというプログラムをサーバ上にアップロードして運用するというモバイルエージェントを応用したシステムが試みられている。

エージェントサーバの動作と、その性能の低下要因について述べる。エージェントサーバは図2のように、メモリ上のデータキャッシュと固定記憶上のデータベースをもち、登録されている全てのエージェントはデータベース上にデータが登録され、そのうち実際に動作しているエージェントに対してはデータキャッシュにデータがおかれる。エージェントサーバの動作は図1のようにまずユーザがエージェントサーバにエージェント配置を依頼し、その次にユーザのデータがオンメモリのデータキャッシュ上に存在しているかどうか調べ、存在していない場合固定記憶にユーザのデータが存在しているかを調べる。そこにも存在しない場合は新たにデータを作成する。そうして得られたユーザのデータをデータキャッシュ上にアップロードし、または最初からデータキャッシュにあった場合はそのまま、そのデータをエージェントに割り当てすることで配置が行われる。

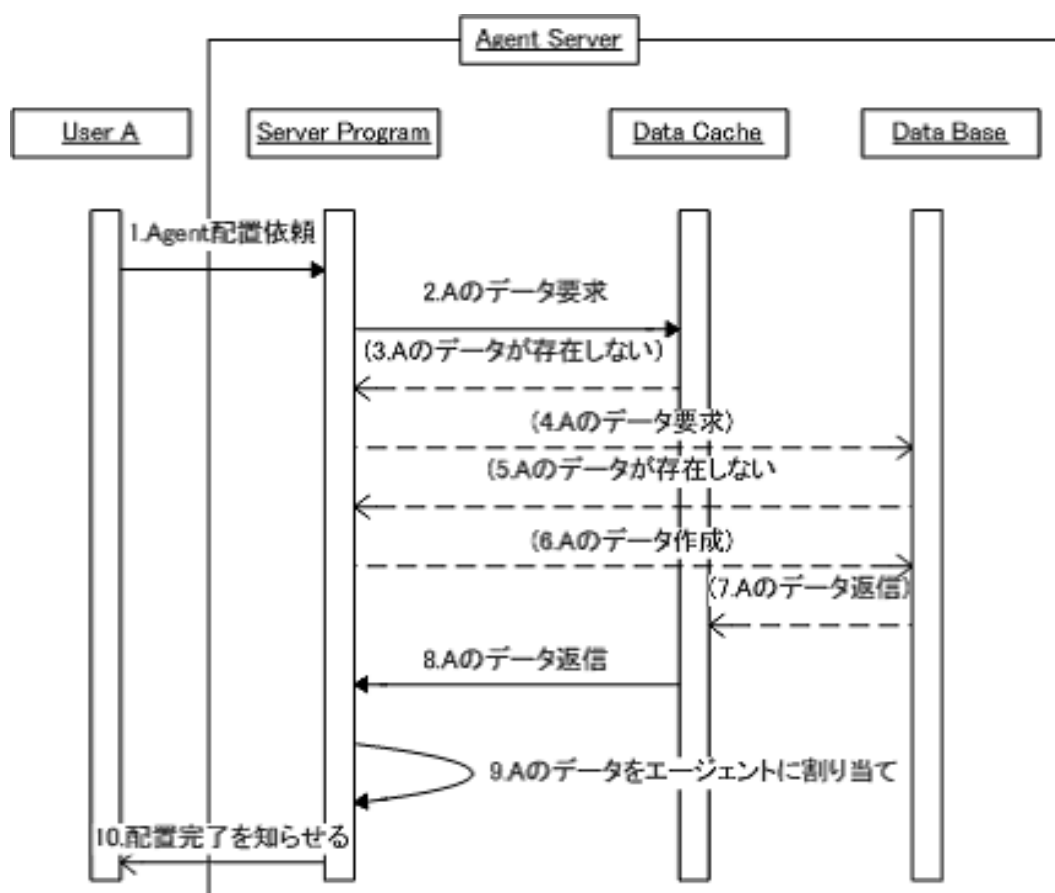


図1: エージェントの配置手順

エージェントサーバの性能低下要因として一つのエージェントサーバにある

許容量以上のエージェント利用が行われた時に図2に示されるようなスワッピングが発生する。スワッピングとは本来オンメモリ上で扱うことができるはずのエージェント数を超えた場合に、現在オンメモリのデータキャッシュ上で扱っているエージェントのデータを固定記憶に待避させて、新たにエージェントの配置を行わなければならないことである。固定記憶へのアクセスはオンメモリのデータへのアクセスに比べて大変にコストがかかるために、可能な限り少なくしなければならない。

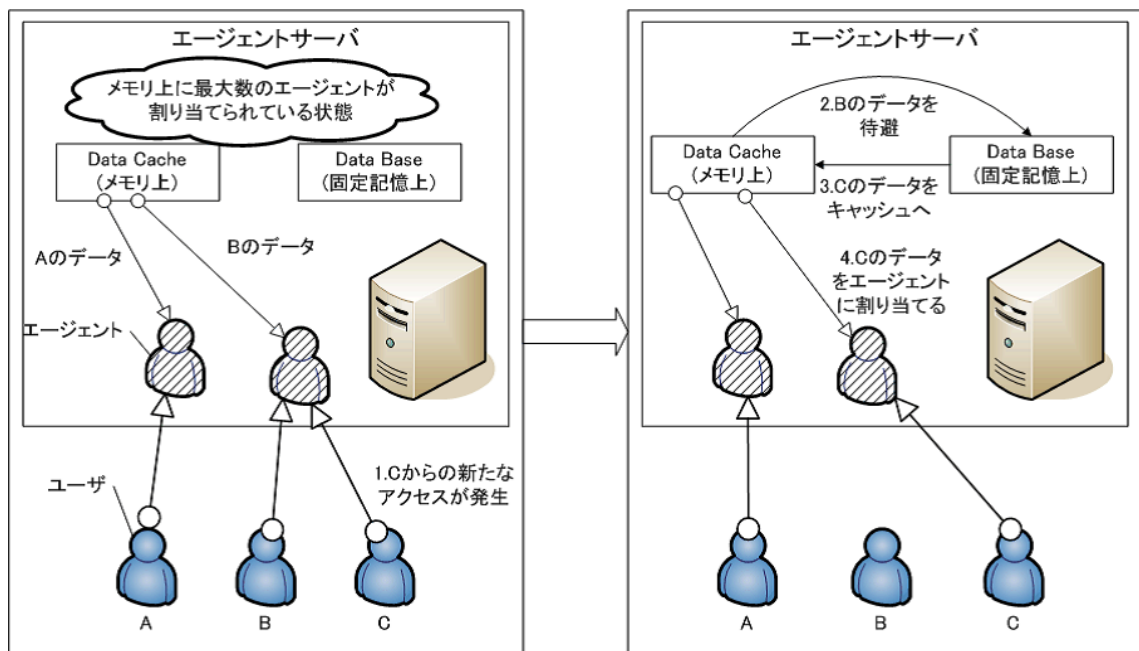


図2: スワップの発生

エージェントサーバを複数ネットワーク上で接続して用いるという分散エージェントシステムでの性能低下要因となるインタラクションについて述べる。

災害時などに従来用いられてきた安否確認のサービスは、実際の災害時になると、一つのサーバで動かしていたためにトラフィックが集中し、正常なサービスが受けられないことが多かった。そのため、避難誘導のエージェントサーバシステムにおいても一つのサーバで運用するのではなく、ボランティア的にP2Pでネットワークを結んだ複数のサーバでサービスを運用して負荷を分散することが研究されている [8]。

複数のエージェントサーバで運用する場合には図3に示すようにサーバの異

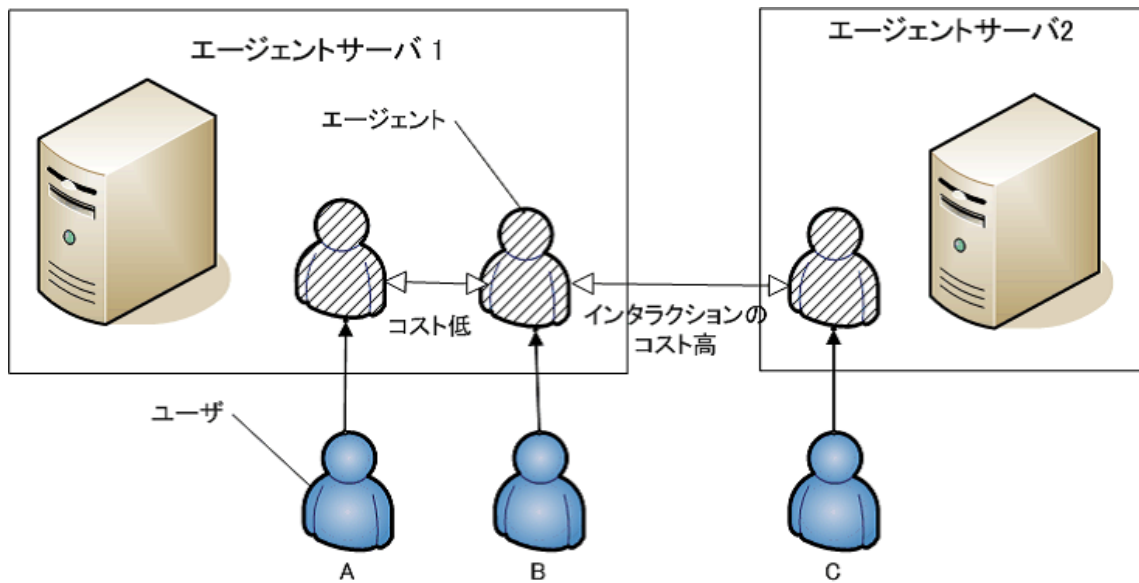


図3: エージェント間インタラクションのコスト

なるエージェント同士のインタラクションは同じサーバのエージェント同士のインタラクションよりもコストが高いことがわかっている。以上のことからエージェントサーバはなるべくインタラクションの多いエージェントの組を同じサーバに配置したいという選好をもっていると考えられる。

2.2 エージェント集合を利用したエージェント配置

1対1に結びつくことが特長として利用されているエージェントシステムだが、複数のエージェントをグループして扱うほうが効率が良い場合がある。避難誘導の例では、同じ組織に所属するもので安否確認や誘導を行うほうが効率が良い。また、グループを代表するエージェントのみに対して情報収集を行えば効率よくシステムが運用できると考えられる。エージェントをグループ分けした場合、グループごとに複数のサーバに分散して広く情報を集めたいといった事や、逆にトラフィックや通信コストの軽減をねらってなるべく同じサーバに配置されたいなどといった選好を持つことが考えられる。

携帯電話へのモバイルエージェントシステムの応用においても同じリソースやサービスの結果をモバイルエージェント同士で一部共有できる仕組みがあればトラフィックを軽減できて効率が良い。その場合、同じ興味を持つグループを構成すればグループ利用であるのでコストも軽減できると考えられる。よて、ト

ラフィックやコストを軽減するためにエージェントのグループを設け、グループごとに特定の興味に応じた選好や、特定のサーバにどうしても配置されたいなどの選好を設定してそれを満たす運用が考えられる。こうしたユーザおよび対応するエージェントのグループをコミュニティと呼ぶことにすると、図4のように分散エージェントシステムにおいてコミュニティごとの選好を反映したエージェントの配置を決定することが必要となっている。

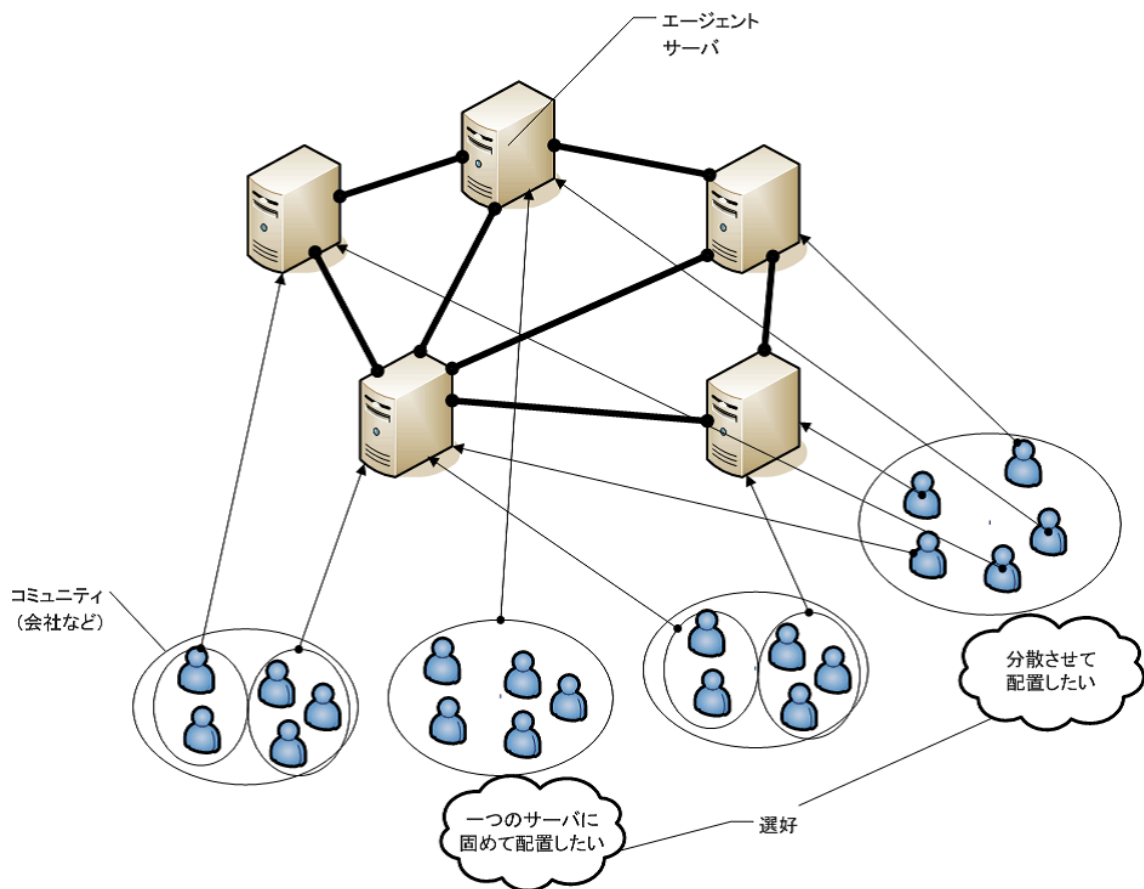


図4: コミュニティごとの選好を反映したエージェント割り当て

第3章 市場計算モデル

本章ではミクロ経済学の一般機構の理論を応用した市場計算モデルについて記述し、2章で述べたエージェントコミュニティの配置問題への応用について考察する。

3.1 市場計算モデルと競争市場

人間が実際に経済活動を行っている市場は、非集中制御の下で大規模な資源割り当てを行っていると言える。特に、理想的な競争市場において、需要と供給が一致する均衡状態では選好をどれだけ満たしているかをあらわしていると言える効用において、他者の効用を下げることなく自分の効用を上げることができないというパレート効率的な資源配分ができることが知られている。市場計算モデルとはこの事を利用して計算機上に仮想的な市場を構成し、競争均衡により決定される価格を計算してその価格に基づく資源割り当てを求めるという手法である。市場計算モデルでは、効用関数に対して利用者ごとの選考も反映しながら無駄のない資源割り当てが実現できることが知られており、トラフィック量の最適化 [1] やネットワークアプリケーション QoS 制御 [2]、動画などのマルチメディアコンテンツのキャッシュ制御 [5] に関する応用が研究されてきた。

少し詳細に定義すると、市場経済は財と経済主体から構成されており、経済主体は財の消費者と財の生産者から構成されている。財は市場において価格が決定され、その価格に基づいて自分の効用を最大化するように経済主体が行動する市場を競争市場という。競争市場において経済主体は自分で直接価格を操作することはできず、与えられた価格に基づいて効用を最大化するために価格受容者 (price taker) と呼ばれる。

3.2 競争均衡とワルラスの模索過程

価格を与えたとき、1つの財についての1つの消費者の需要を求めることができる。需要は予算制約を満足しつつ、効用を最大化する値となっている。1つの財の価格を変化させたとき、財の価格に対して効用を最大化するような財の量が決定でき、それが描く曲線を需要曲線と呼ぶ。すべての消費者についてのその財の需要曲線を求めて、その和のグラフを求めると、その財における市場全体の需要曲線が求まる。同様に1人の生産者についても、価格が与えられると、1つの財について利潤を最大化する生産量を求めることができ、それが描く曲線を供給曲線という。需要曲線と同様に市場全体の供給曲線を求めることができる。市場全体の需要曲線と供給曲線が交わる場所は需要と供給が一致しており、すべての財について需要と供給が一致している状態を競争均衡 (competitive equilibrium) と呼ぶ。

競争均衡が存在する条件は選好が凸であり、生産可能集合が凸であることである。2つの財について他の財の効用を固定して効用が同じ点を結んだ曲線を無差別曲線という。この接線の傾きは片方の財の変化分に対し同じ効用を得るためのもう片方の財の変化分を表し、この比を限界代替率という。選好が凸であるとはこの無差別曲線が下に凸であることを意味し、ある財を持っている量が多ければ多いほどその財の量が一定量増えたときに増える効用の量は少なくなることを意味する。これを限界代替率逓減の定理が成り立つという。生産可能集合が凸であるとは、生産量が単調増加で生産量の増加に従って生産による利益が減少するような状態を意味する。

競争均衡について厚生経済学の基本定理 (fundamental theorem of welfare economics) と呼ばれる以下に示す命題が知られている。

- 競争均衡における、財の需要と供給による配分は、経済主体の効用に対してパレート最適である。パレート最適であるとは、すべての経済主体において他者の効用を下げることなく自分の効用を上げることができない状態。
- 任意のパレート最適な割り当てを競争均衡の結果として導くような初期時の配分が存在する。

このような競争均衡状態をえるための仕組みを説明する。まず、中立的な競売機構が各財について初期価格を設定する。各経済主体は、与えられた価格の下で消費者は予算制約を満たしつつ自分の効用を最大に、あるいは生産者は生産可能な生産で利潤を最大にする。競売機構は需要が多いものの価格を上げ、供給が多いものの価格を下げる。このプロセスを繰り返し、各財の需要と供給がすべて一致すると、競争均衡となり実際の取引が行われる。この方法をワルラスの模索過程という。この方法で均衡状態を導くには、ある財の価格が上昇しても他の財の需要が減少しないという粗代替性が満たされなければならない。パンとジャムのようにパンの売り上げが落ちれば、ジャムの需要も減少するというような関係の場合粗代替性は満たされていない。

一般的な市場計算モデルでは経済主体である消費者と生産者にエージェントを割り当て、各エージェントが効用や利潤を最大にする戦略をとりながら競売機構に入札する。市場計算モデルにおいて生産者エージェントが生産するサービスなどの財、及びその財を購入するために消費者エージェントが支払う特別な財である対価 (factor of production) の関係を図5に示す。

図5において、左側の円は消費者エージェントを、右側の円は生産者エージェ

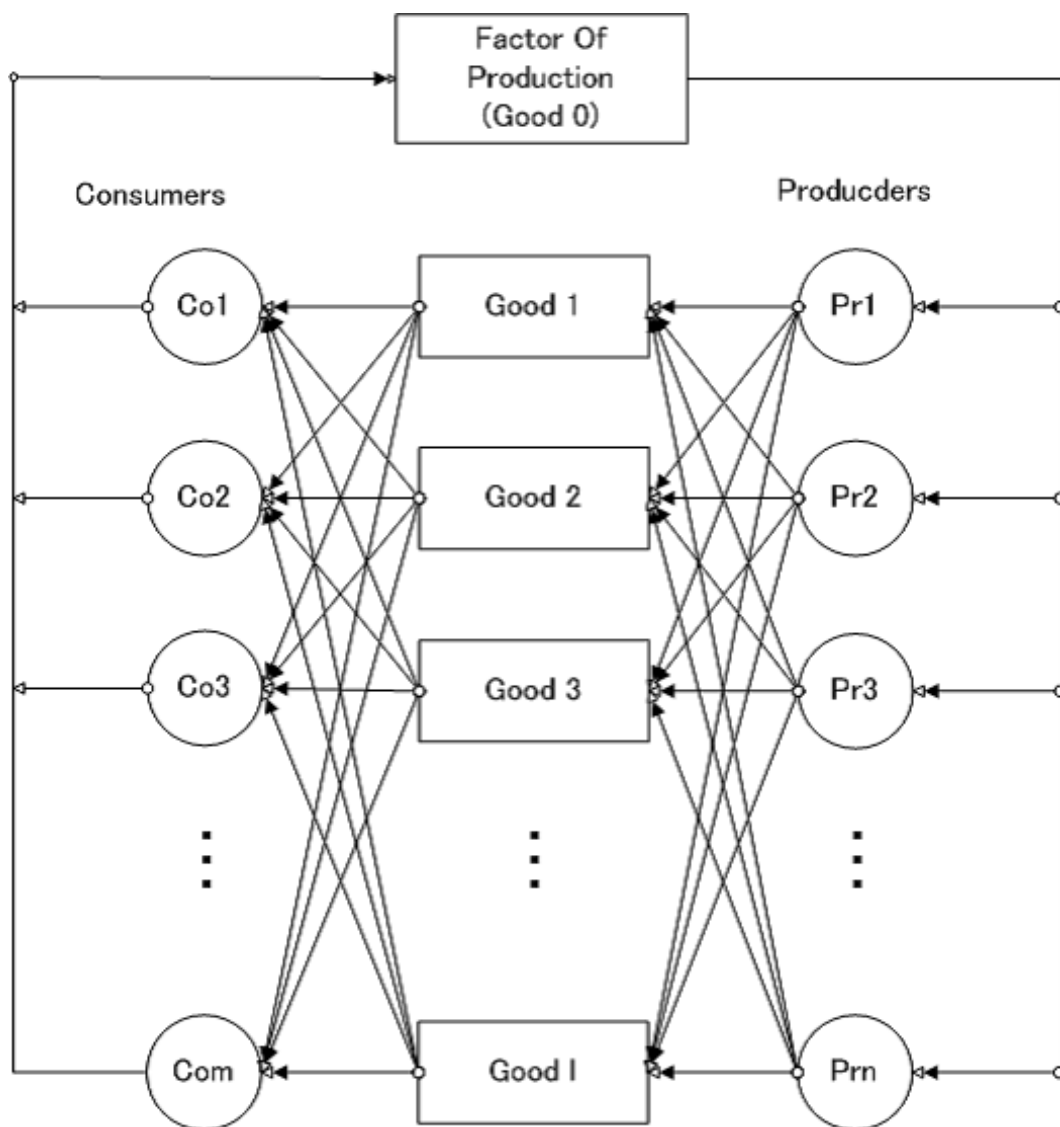


図 5: 一般的な市場計算モデル

ントを表し、真ん中の列の長方形は財を表して、一番上は対価となっている。財から消費者エージェントへの矢印は財を需要することを示し、消費者エージェントから対価への矢印は、代わりに対価を手放すことを示す。対価から生産者エージェントへの矢印は生産者が生産の材料として対価を得ることを意味し、生産者エージェントから財への矢印は、対価を用いて生産者エージェントが財を生産することを意味する。また、資源には価格が設定されており、価格は価格決定機構である競売人エージェントが決定する。価格に基づいて、消費者エージェントが需要を決定し、生産者エージェントが供給を決定するという過程を、均衡状

態になるまで繰り返すと、パレート効率的な資源配分となっている。

3.3 消費者エージェント

消費者エージェントは、ユーザの選好を反映した効用関数を外部から与え、価格に基づいて需要を決定する。市場に n 個の財があるとき、ある消費者において $X' = (x'_1, x'_2, x'_3, \dots, x'_l)$ より $X'' = (x''_1, x''_2, x''_3, \dots, x''_l)$ が望ましい。このとき、ある効用関数 $u(x_1, x_2, x_3, \dots, x_l)$ において $u(x'_1, x'_2, x'_3, \dots, x'_l) > u(x''_1, x''_2, x''_3, \dots, x''_l)$ となるように、効用関数を定める。任意の x', x'' において、上記の式を満たすような連続な効用関数は、選好がいくつかの条件を満たしていれば存在することが示されており、ユーザの選好を効用関数として記述することができる。

同一のサーバに固まって配置されるほうが望ましいという効用は、

$$U(X) = \sum_{i=1}^l \alpha_i x_i^\rho$$

という効用関数において $\rho > 1$ とすればそのようなそのような効用になると考えられる。ただし、 α_i は各サーバへのプリファレンスとする。逆に $\rho < 0$ とすれば、分散して配置されるのが望ましいという効用を表すことになると考えられる。

消費者エージェントは、消費者エージェントが初期に持つ予算を制約条件としてこのような効用関数を最大化するように行動する。すなわち、 p_i を財 i に付けられた価格、 e_k を消費者 k が持つ予算とすると

$$\sum_{i=1}^l p_i x_{ik} \leq e_k$$

を制約条件として上記の効用関数を最大化するように需要を決定する。

3.4 生産者エージェント

生産者エージェントは、ある資源を用いて他の資源へと変換する働きを持ち、消費者同様に、利生産者エージェントは、ある資源を用いてその他の資源を変換する働きを持つ。どのような変換が行われるかは、生産関数によって記述される。サービス i に対する生産者の生産関数を $f_i(z)$ とする。ここで、 z は生産に投入される資源ベクトルである。生産者 i によるサービス生産量を y_i で表すと、 $y_i = f_i(z)$ で表される。すなわち、生産関数は、生産者が複数の資源を含む資

源ベクトル Z を用いて最大 y_i のサービスを生産できることを示している．生産者は，生産に必要な資源のコストを支払い，生産したサービスの提供による収入を得る．従って，各資源の価格を要素に持つ価格ベクトルを P として，以下の利潤最大化問題を解き，供給を決定する．

$$\max \sum_{i=1}^l p_i f_i z - PZ$$

3.5 競売人エージェント

ある価格において，消費者エージェントが表明する需要量と生産者エージェントが表明する供給量は必ずしも一致しない．このため，需要量・供給量に対応して価格を調整する主体である競売人エージェントを定める．需要量よりも供給量が多ければ，価格を上げ，需要量を減少させ供給量を増加させる．逆に，供給量よりも需要量が多ければ，価格を下げる．

価格の調整を行った競売人エージェントは，新しい価格情報を消費者エージェント・生産者エージェントに伝える．このような価格調整過程は，需要関数や生産関数が凸である等の性質を満たす場合に収束する．

3.6 市場計算モデルを用いたエージェント割り当て問題の解決

上記のとおり計算機上に市場機構及び効用関数を構成すれば，ユーザの選好を満たしながらパレート効率的な解を見つけることができる．

エージェント割り当て問題に対して，以下のように定義する．

コミュニティ	$C_{01}, C_{02}, \dots, C_{0m}$
エージェントサーバ	$S_{e1}, S_{e2}, \dots, S_{en}$
エージェントサーバの容量に比例した CPU リソース	s_1, s_2, \dots, s_n
コミュニティ C_{0i} がサーバ S_{ej} で獲得した CPU リソース	x_{ij}
エージェントサーバ S_{ej} の単位あたりの CPU リソースの価格	p_{ij}
このとき，需要関数は Property を予算として次のようになる．	

$$x_i = \left(\frac{p_i}{\alpha_i}\right)^{\frac{1}{\rho-1}} * \frac{Property^{\frac{1}{\rho-1}}}{\sum_{i=1}^n \left(\frac{p_i}{\alpha_i}\right)}$$

エージェント割り当て問題に対して市場計算モデルを適用した場合として次の3つのモデルを考える

- モデル1

図6のようにコミュニティのみを消費者として扱うモデル. エージェントサーバから常に一定量のCPUリソースが提供されると仮定して生産者を設けず, 消費者は初期財として合計が総CPUリソースとなるように各コミュニティに与える. 消費者は効用関数にもとづいて需要関数を決定し, 需要を決定する. 競売人は供給一定の元で需要が供給と等しくなるように価格を調整する.

Consumers

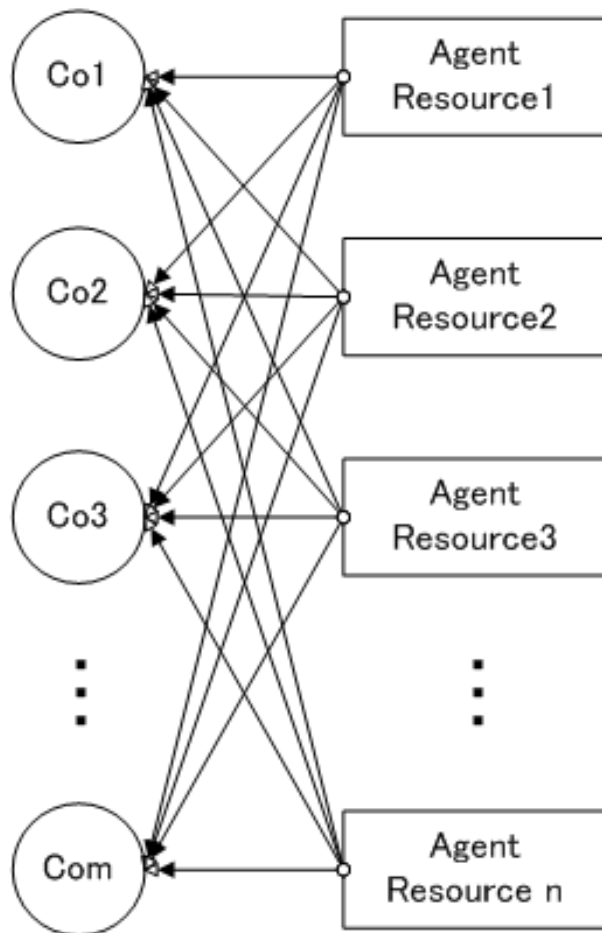


図6: コミュニティのみを消費者とするモデル

- モデル2

コミュニティとサーバをどちらも消費者とするモデル. ただし, コミュニティ側

の消費者は初期財として対価をあらわす財を持つとし、コミュニティ側の消費者はCPUリソースを得ようとする選好を持ち、エージェント側の消費者は初期財として自分のCPUリソースを持ち、対価を得ようとする選好を持つとする。競売人はサーバ側の支払う財を供給とし、コミュニティ側の消費者の需要と一致するように価格を調整する。

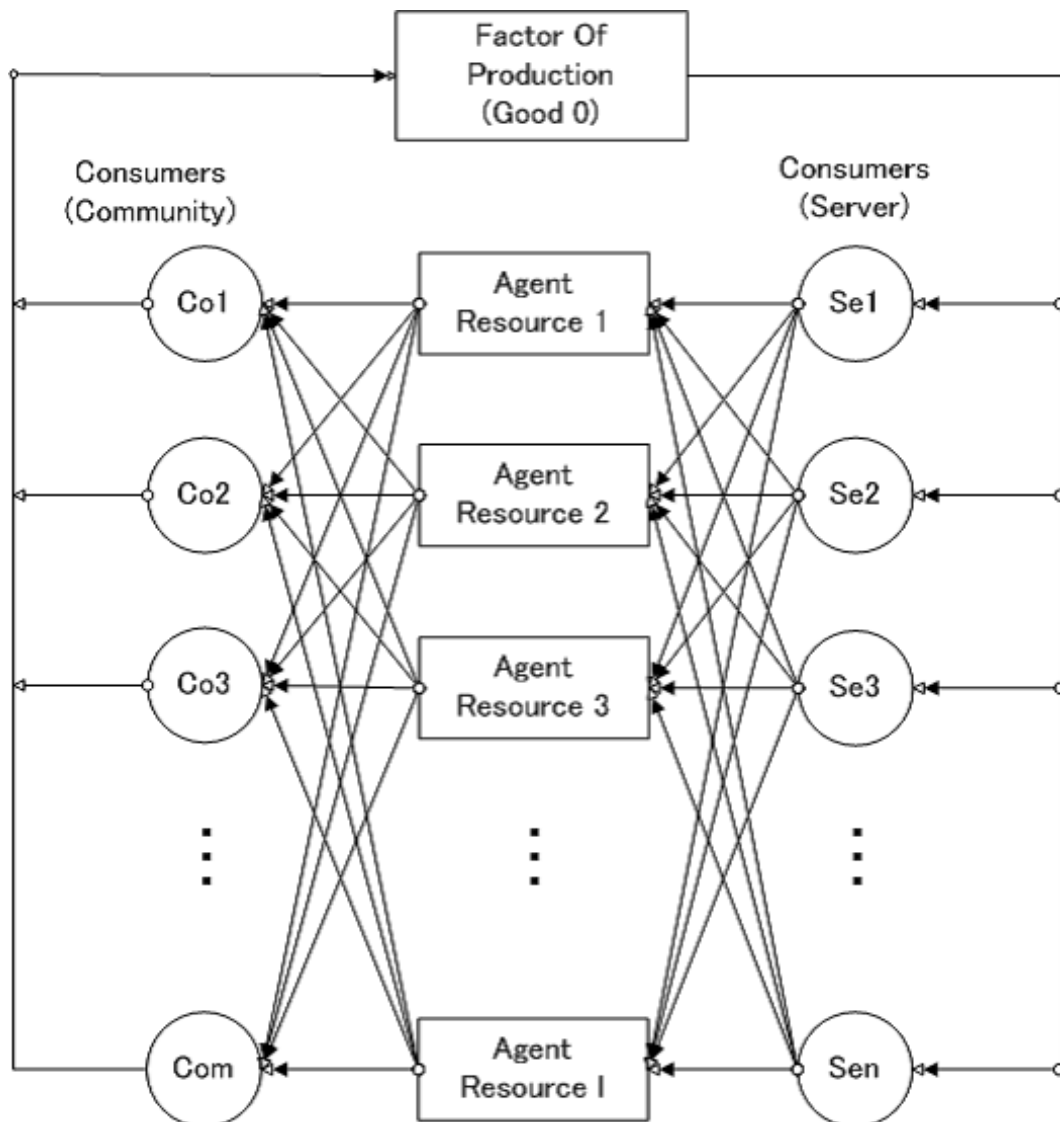


図7: コミュニティ, サーバともに消費者とするモデル

- モデル3

図8のようにコミュニティを消費者, サーバを生産者とするモデル. サーバの生

産者エージェントが対価の財を CPU リソースに変換する生産を行うと考える。モデル2と同様に消費者は対価の財を初期財としてもち、生産者は自分の所有するエージェントリソースを初期価格にて売却した分の資金を持つとする。競売人は需要と供給が一致するように価格を調整する。

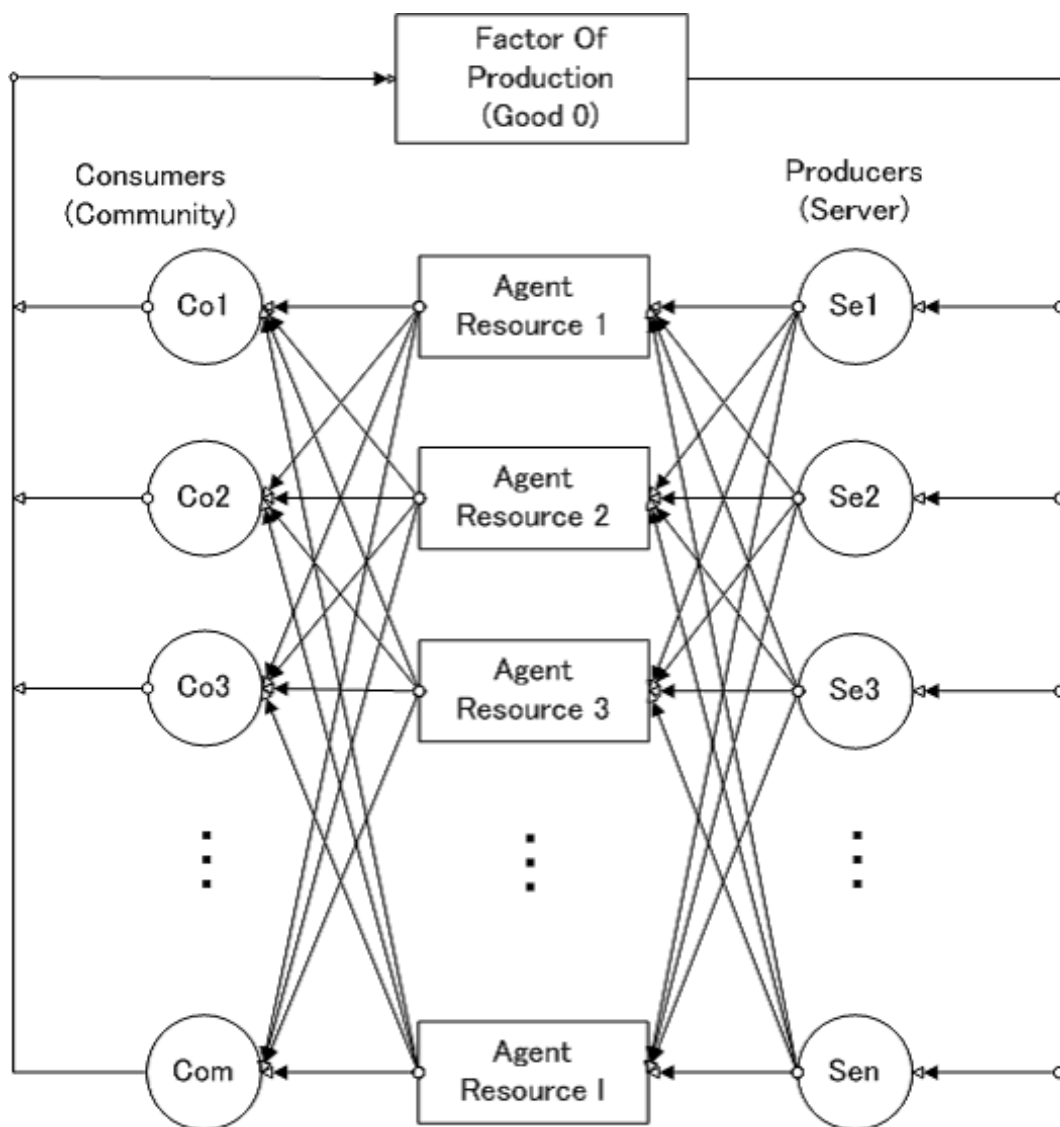


図8: コミュニティを消費者, サーバを生産者とするモデル

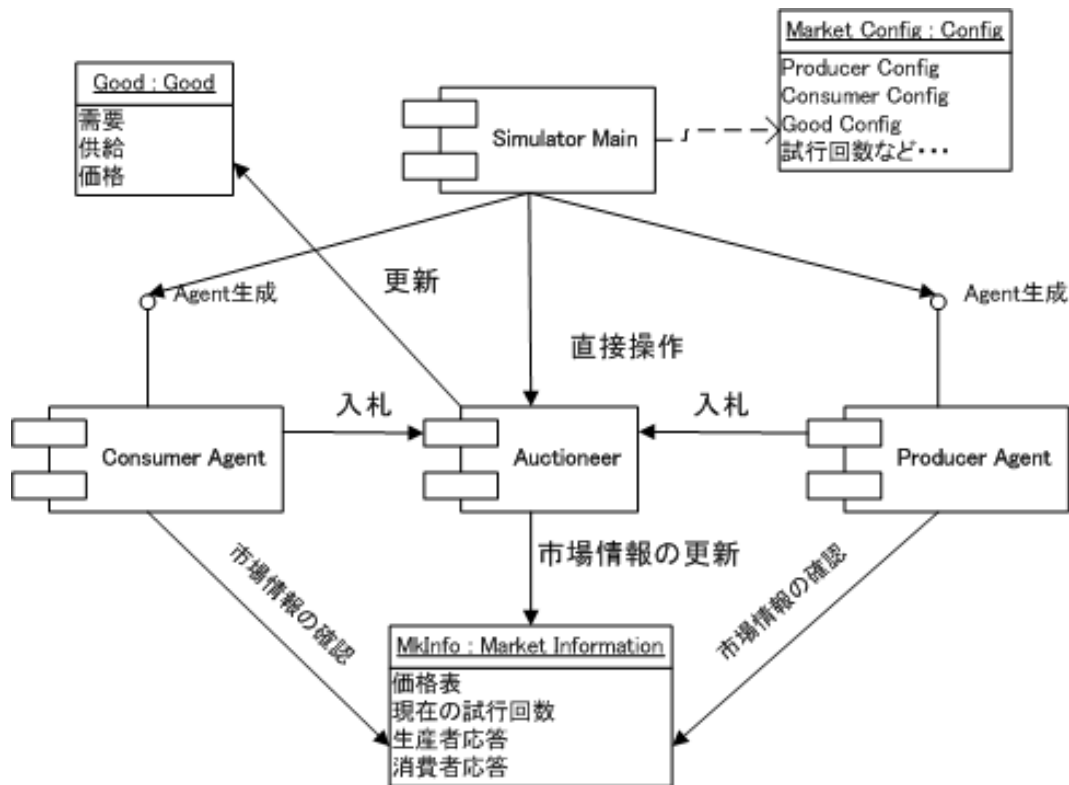


図 9: シミュレータの構成

第 4 章 市場計算モデルを用いたシミュレーション

4.1 シミュレータの構成

3章で示したモデルや、効用関数の妥当性の検証を行うために市場計算モデルを用いたシミュレータを構成する。図9にシミュレータの構成を示す。メインとなるプログラムが設定情報を参照して、生産者、消費者の各エージェントを生成する。競売人から各エージェントにメッセージを送ることはなく、市場の情報を更新することで各エージェントが自律的に価格の変更を知ることができる。

入力は各エージェントサーバのエージェント数、各コミュニティの所有する初期財、各コミュニティの各サーバへの好みをあらわすプリファレンス、分散させたいか集中させたいかを表すと考えられるの値となり、出力はエージェントの配置となる。

図 10 にシミュレータの動作を示す。

1. 前回の入札による需要と供給により決定した価格の変更を市場情報に対して行う。次に市場情報の試行回数を増やすことで間接的に各エージェント

- に価格の変更を知らせる.
2. 各エージェントはこの間に価格の変更がないかどうか試行回数を随時チェックする.
 3. 価格の変更を知った消費者エージェントはすべての財の価格情報を取得し, 生産者エージェントは自分の生産する財の価格情報を取得する.
 4. 価格に基づき自分の需要あるいは供給の計算を行う.
 5. そして, その需要および供給入札を行う.
 6. 各エージェントはエージェントの入札カウンタである生産者応答, 消費者応答をインクリメントして入札済みであることのフラグを立てる.
 7. 競売人エージェントは生産者応答, 消費者応答を調べることで入札数が最大になるのを待つ.

最大になった時点で全需要と全供給が求まるのでそれを用いて新たな価格を決定する. として、1~8までのループを繰り返す. 競売人は需要と供給の差に閾値を設け, 閾値よりも小さければ収束したものと判定してその時点での需要と供給で配置を決定する.

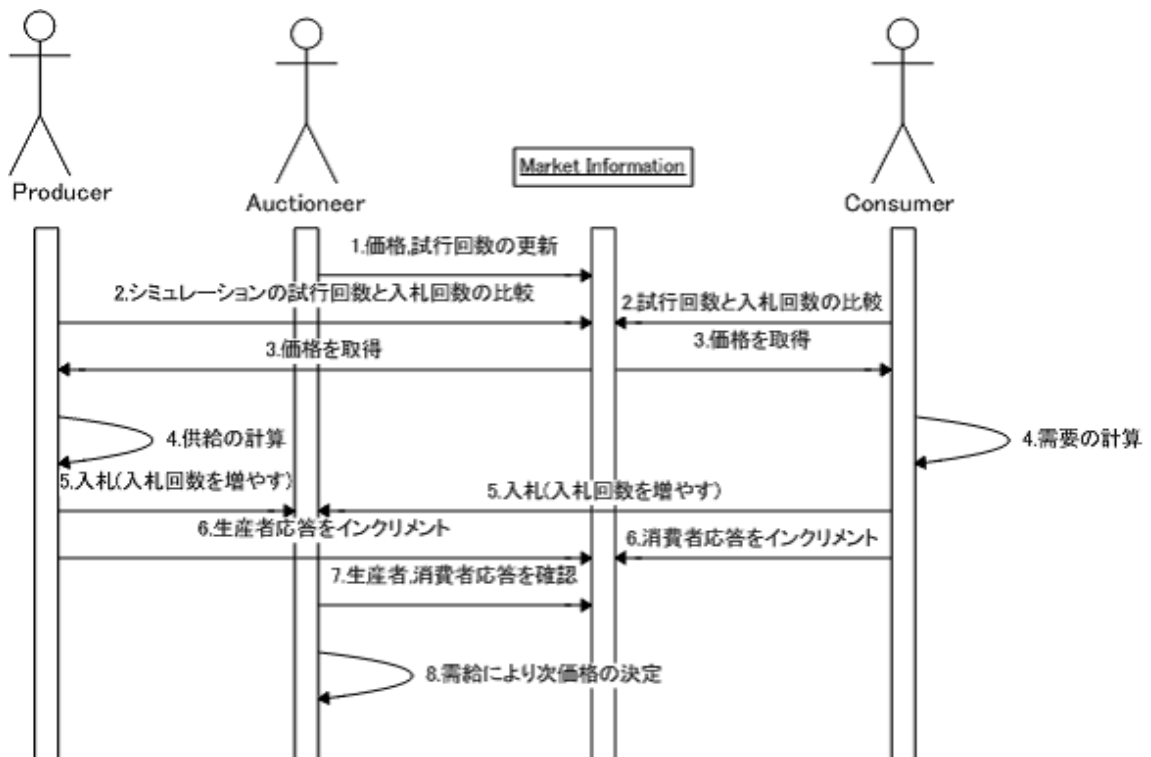


図 10: シミュレータの動作

4.2 シミュレーションの結果とその考察

コミュニティの総数 20, エージェントサーバの総数 10, 許容量は 200 から 1000 の間で, 実験を行った結果を以下に示す.

コミュニティのみを消費者として, サーバ側は一定量の供給をし続けるモデル 1 において $\rho = 0.2$ で実験した結果を, コミュニティ 0 から見た結果が図 11 となっている. グラフは横軸が入札回数, 縦軸が各サーバへの需要を表している. 初期値はほぼプリファレンスから決まっているが, 価格によって需要が上下して収束していることがわかる. ほぼ収束しており, ρ が 0 に近いために分散して各サーバに配置されていることがわかる.

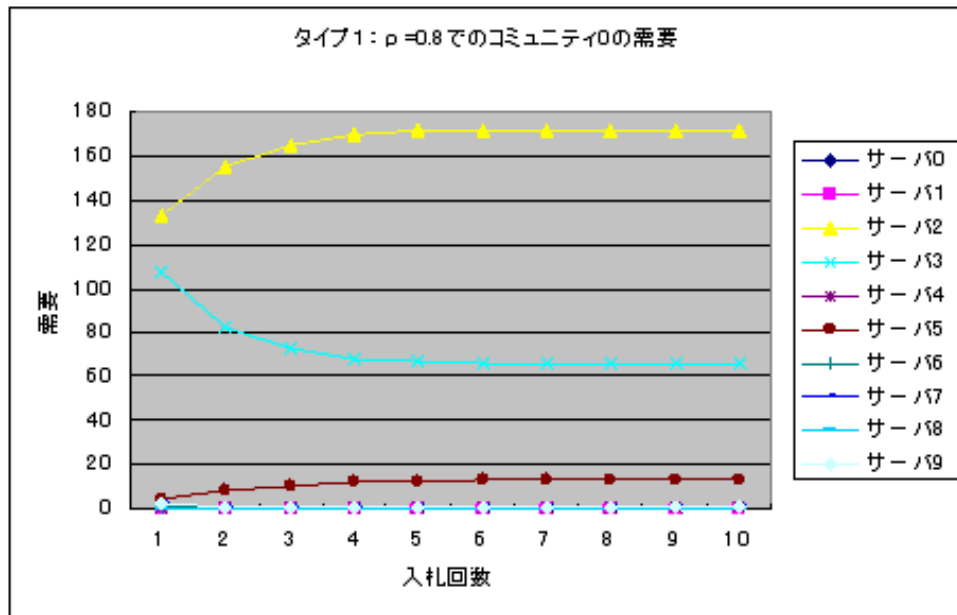


図 11: モデル 1, $\rho = 0.2$ でのシミュレーション結果

$\rho = 0.2$ のモデル 2, モデル 3 も同様に収束している. モデル 3 では生産者エージェントの生産関数によっては, 全部を生産しないことになるのでいくつかのサーバに対する需要はモデル 1 やモデル 2 と比べて低くなっている.

図 14 はモデル 1, $\rho = 0.8$ でのシミュレーション結果であり, $\rho = 0.8$ で 1 に近いことになるべく固まって配置されたいという選好をあらわしていると考えられる. 2 つのサーバ以外にはほとんど配置されていないため, 選好が反映されていると言える.

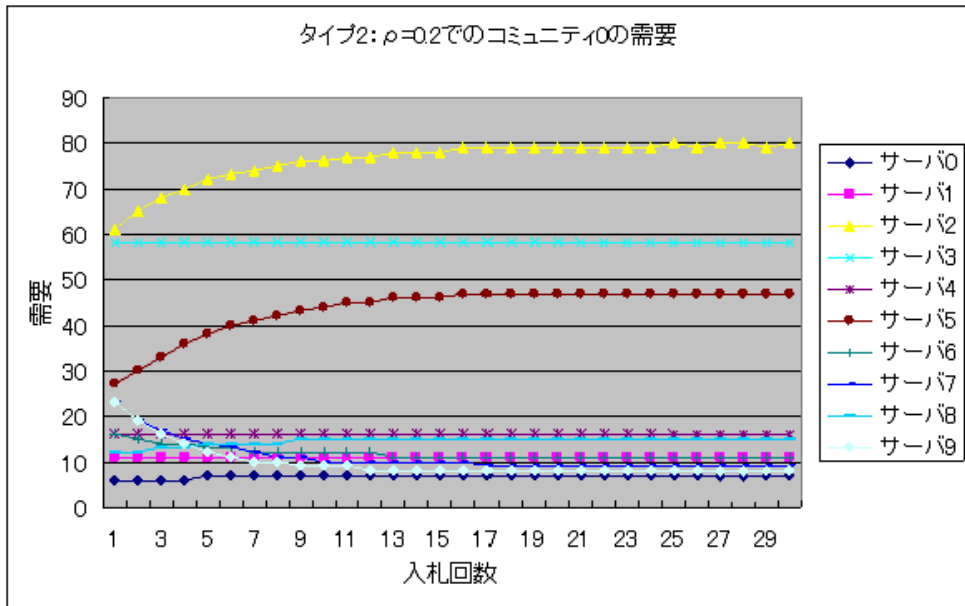


図 12: モデル 2, $\rho=0.2$ でのシミュレーション結果

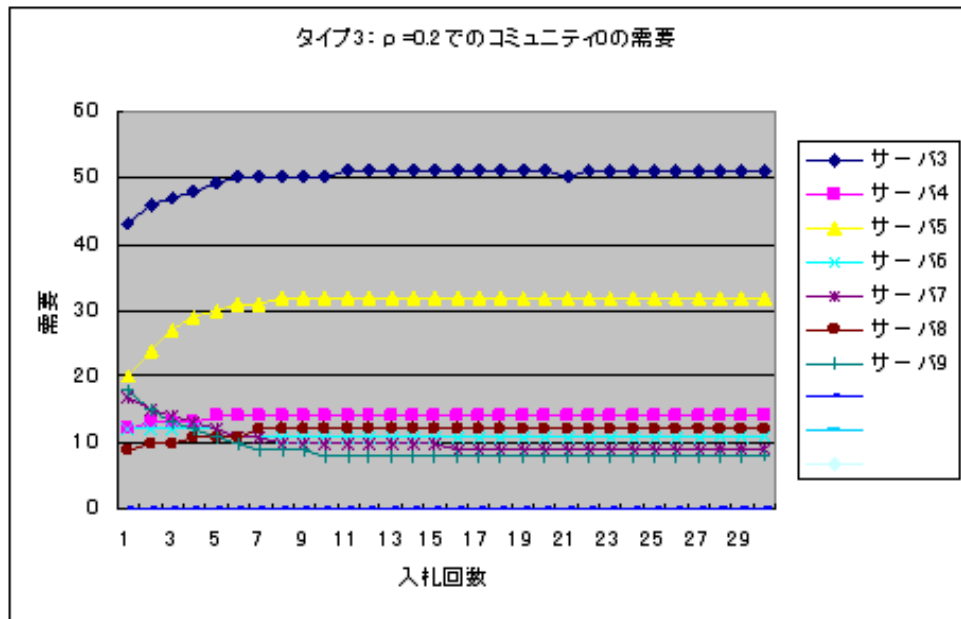


図 13: モデル 3, $\rho=0.2$ でのシミュレーション結果

図 15, 16 はモデル 3 の $\rho = 0.2$ での財 0 の価格と需給の変化を表す. モデル 3 は供給も変化しながら収束している様子が見られるが, モデル 1 や 2 では供給は変化せずに需要のみ推移していた.

どのモデルも収束はしたが, 数回にわたる試行の結果, モデル 1 の場合が一番

早く収束した。これは、サーバ側からの供給が一定であるためだと考えられる。モデル3の場合は全部のサーバリソースを供給しなくなることがあるため、他のモデルと比べた場合に需要及び供給が少ない状態で収束することがあった。また ρ の値を大きくしすぎると価格よりもプリファレンスのみで需要を決めてしまうことがわかった。

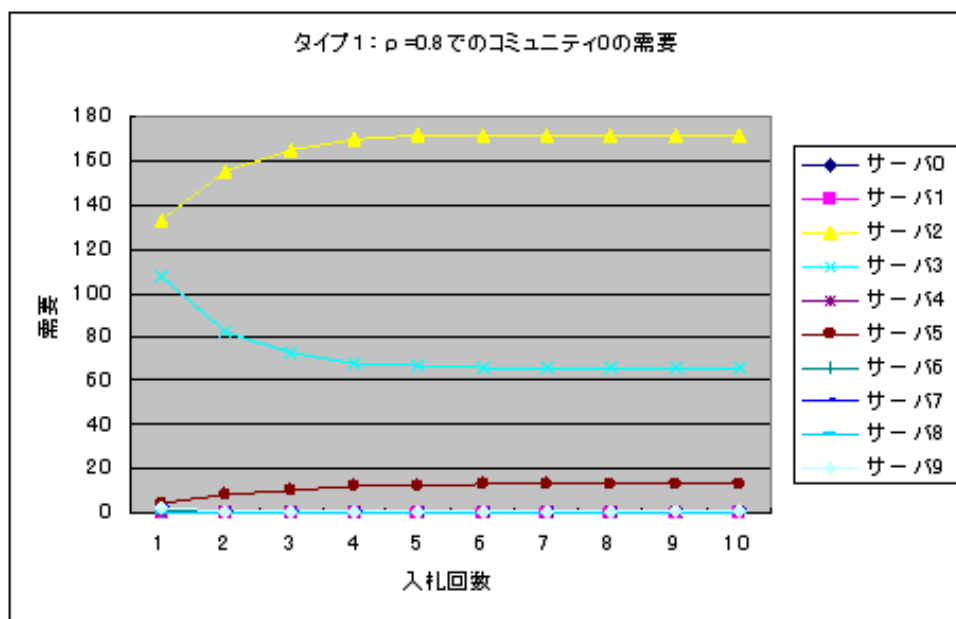


図 14: モデル1 $\rho = 0.8$ でのシミュレーション結果

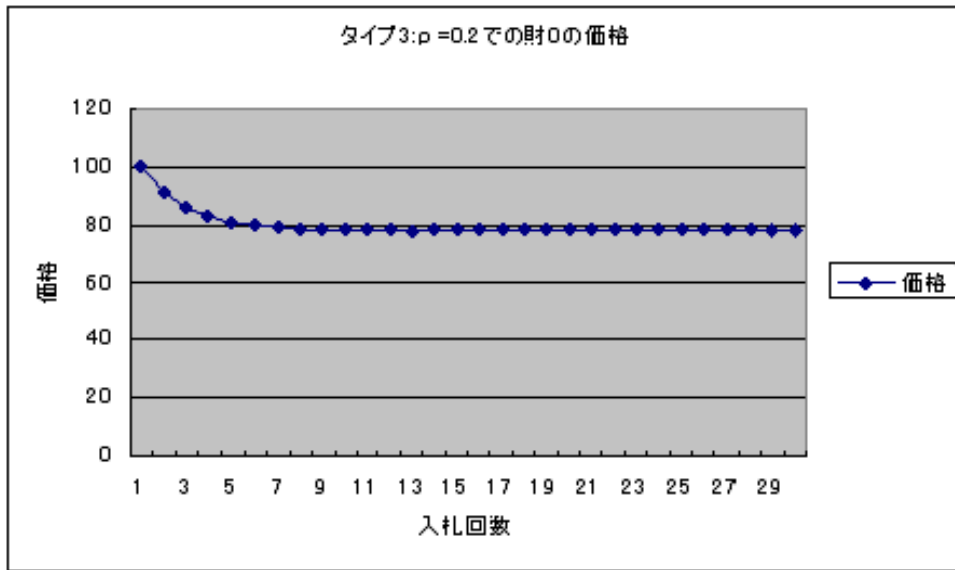


図 15: モデル3 $\rho = 0.2$ のシミュレーション結果 (価格)

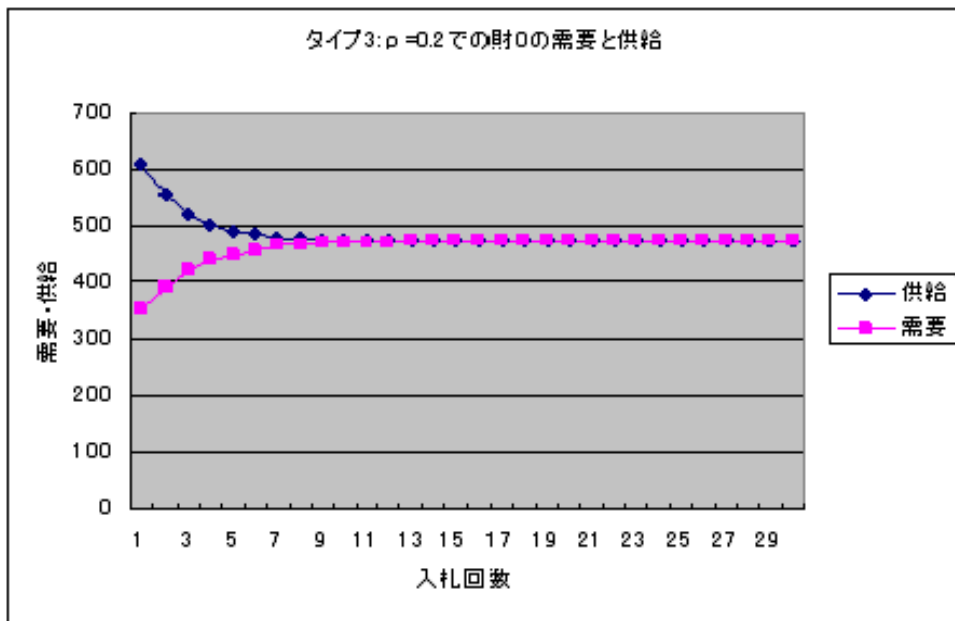


図 16: モデル3 $\rho = 0.2$ でのシミュレーション結果 (需要と供給)

第5章 実際のエージェントシステムへの実装の考察

図 17 のように実際のエージェントサーバに対してはサーバ1 台に対し1 人の競売人エージェント (Auctioneer) を用意すれば, 各サーバが物理的に独立であることを利用して, 分散して市場計算モデルに基づくエージェント配置が計算

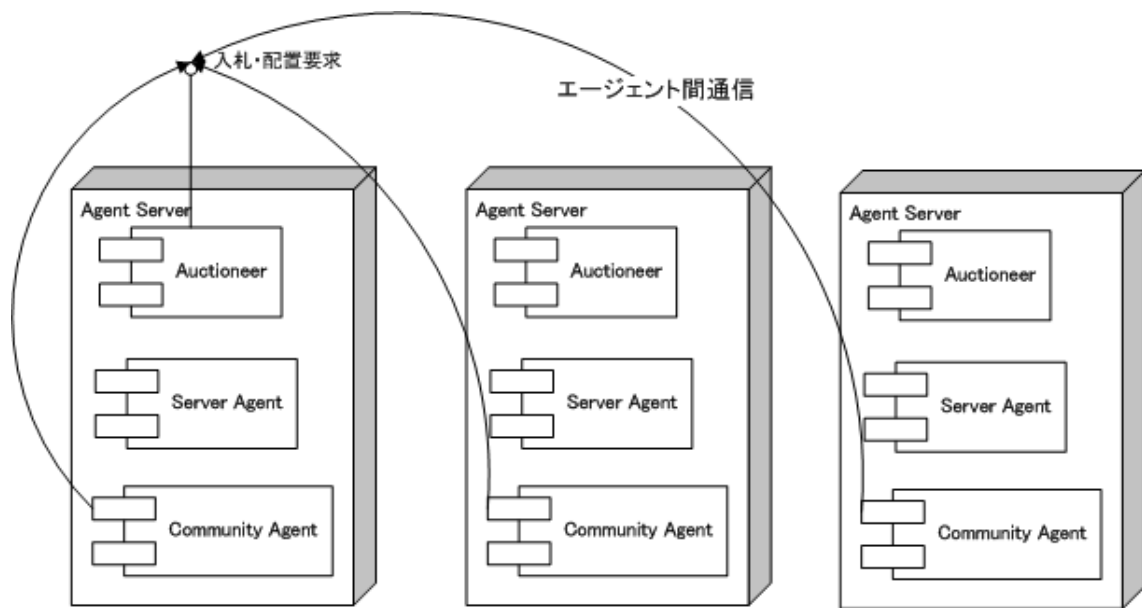


図 17: 実装案

できる。その際に、入札や配置要求の伝達はエージェント間通信を利用する。各サーバには複数のコミュニティエージェントが存在し、各サーバに1つしかない競売人エージェントに対して、すべてのサーバに対してエージェント間通信を用いて入札を行う。

図 18 により詳細な実装の様子を示す。図 19 のように1つのエージェントサーバには1つのエージェントサーバプログラムが動作している。また、エージェントの配置を指示するコントローラのプログラムもエージェントサーバ毎に1つ動作している。エージェントサーバプログラム上には1つの競売人エージェント、1つのサーバを代表するエージェントが存在しており、サーバを代表するエージェントは、3章で述べたモデル2の場合は消費者、モデル3の場合は生産者として振る舞い、自分のサーバに存在する競売人エージェントに対して入札を行う。コミュニティを代表するエージェントは各サーバに複数存在することがあるが、1つのコミュニティに対して全てのエージェントサーバ上に対して1つのコミュニティエージェントがわりあてられる。コミュニティエージェントは3章で述べたどのモデルでも消費者として振る舞い、自分のサーバ及び他のサーバの競売人に対して入札及び配置要求を行う。競売人エージェントはサーバエージェントとコミュニティエージェントの入札に従い、価格を調整して次の入札を指示する。また、一定数の入札後に送られるコミュニティエージェン

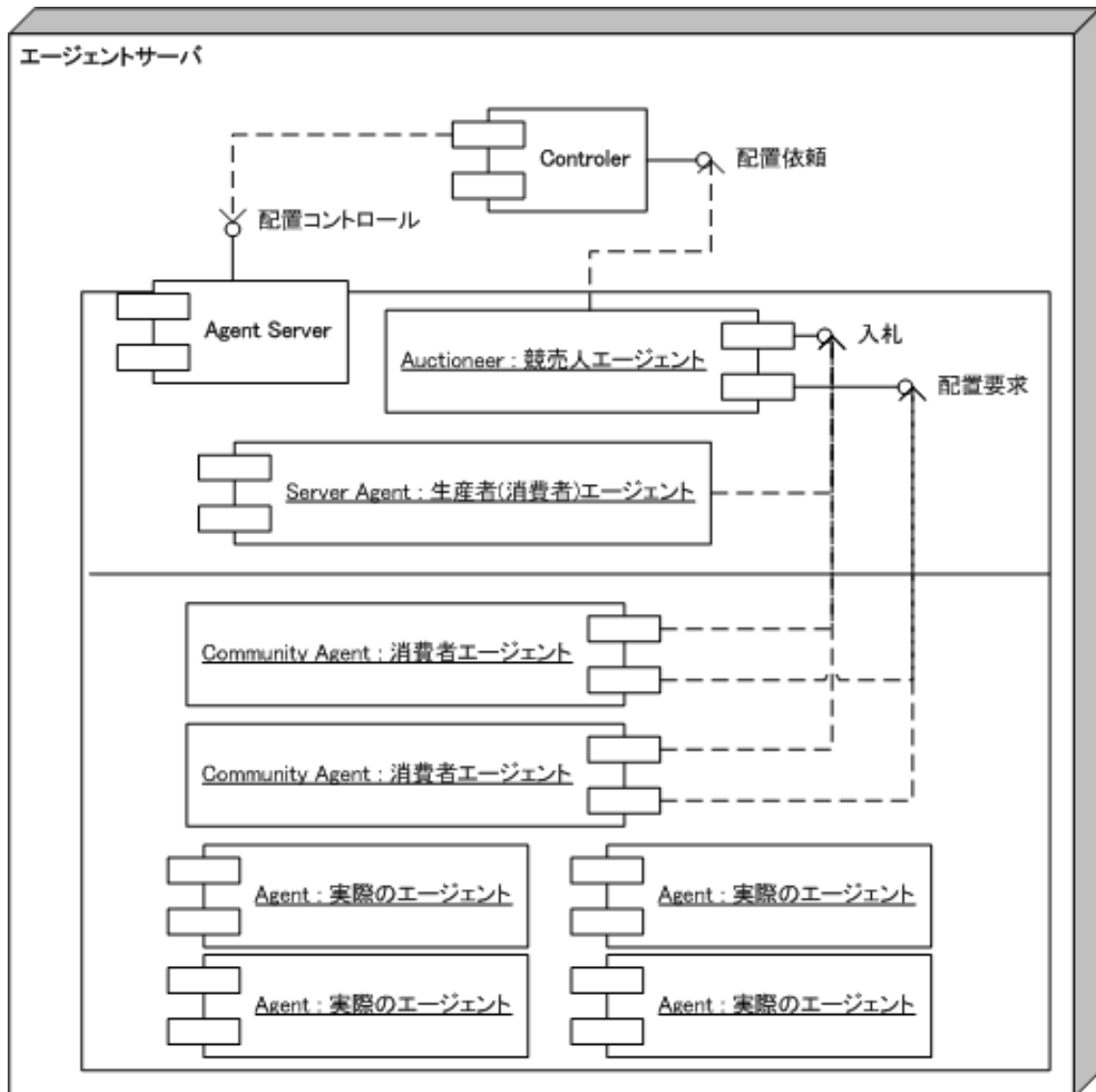


図 18: 各エージェントサーバの構成

トからの配置要求に従って、コントローラに対して配置を依頼する。依頼を受けたコントローラはエージェントサーバプログラムに対して実際のエージェント (各コミュニティのメンバーにあたる) の配置を指示する。

計算手順は以下のようになり、図 19 にその様子を示す。

1. 競売人エージェントは価格をサーバエージェント及びコミュニティエージェントに通知する。
2. 価格に基づいてサーバエージェントは供給を決定、コミュニティエージェントは需要を決定する。

3. 需要と供給を競売人に対して入札する.
4. 需要と供給から競売人は価格を調整し, 次の入札を行うために1にもどる.
5. 1~4を一定回数繰り返した後, コミュニティエージェントからそのときの需要に基づいてエージェントの配置要求が競売人に送られる.
6. 各コミュニティエージェントからの配置要求が集まると, 競売人エージェントはコントローラに対して配置依頼を送る.
7. コントローラがエージェントサーバに対し, エージェントの配置を指示する.

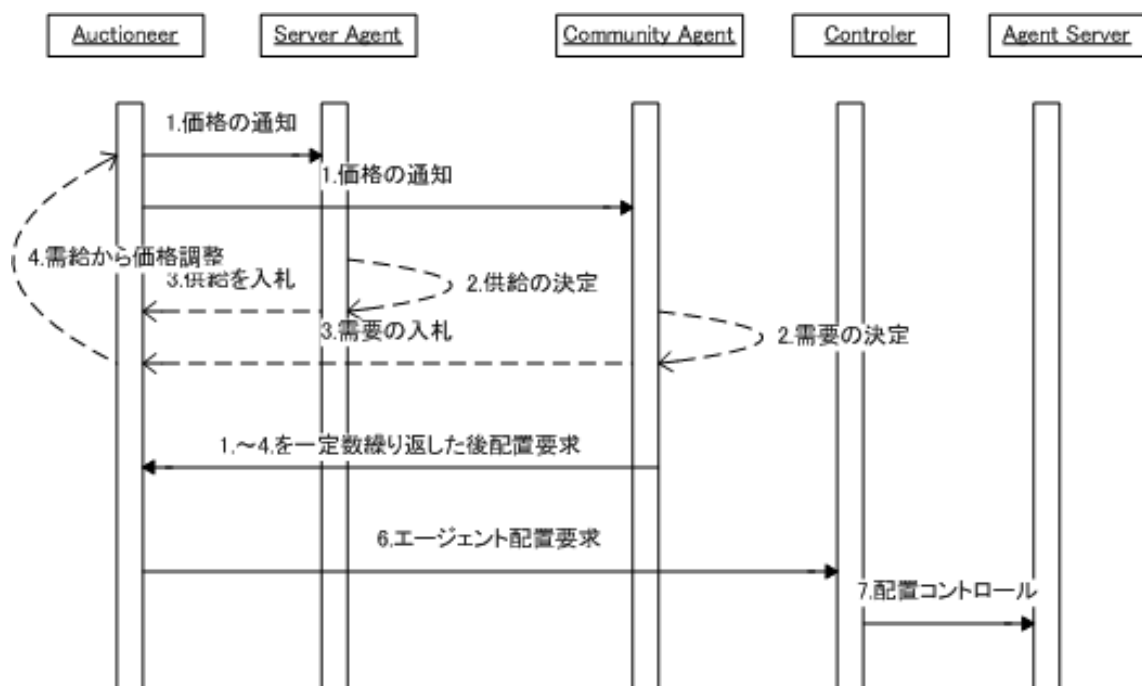


図 19: 実装時の計算手順

エージェントごとに一定の入札回数を手順に従って計算するとコミュニティは他のコミュニティの動作を気にせず, サーバエージェントは他のサーバエージェントを気にせず独立に計算できる. 同期を取るのではなく入札回数で動作するようにすればよいが, 入札するコミュニティ数を全てのサーバエージェントが把握していることが必要となる. 同期を取っていないために実際の配置で問題になることが考えられるが, 計算するのは自分のエージェントサーバを代表する競売人エージェントとサーバエージェントおよびそこに存在するコミュニティを代表するエージェントだけなので, 差はあまりでないと考えられる.

第6章 終わりに

本研究では、今後ますます応用が期待される分散エージェントシステムにおいて効率の良いエージェント配置を目指す仕組みを提案した。これによって大規模な避難誘導をエージェントを用いて行うシステムや、携帯電話と一対一に結びついてサービスを行うモバイルエージェントシステムなど、複数のエージェントサーバをネットワーク上で接続して運用するアプリケーションにおいて、ユーザの集合ごとの選好を反映しながらエージェントのリソースを無駄なく割り当てる手法の実現を市場モデルを用いて実現することを提案することができた。

現在の分散エージェントシステムにおいては利用者の集合ごとの選好を反映したエージェント配置は行われておらず、今後そうした配置が必要とされてくることが考えられる。本研究を用いて得られるエージェントリソースの配置は、市場計算モデルによって得られるためにパレート最適性を示し、無駄のない資源割り当てが可能であることが保障されており、有効な資源配分が可能となる。これによって、分散エージェントシステムを扱うアプリケーションでの決定手法の提案として情報学に貢献する。

参考文献

- [1] Wellman, M.: A Market-Oriented Programming Environment and its Application to Distributed Multicommodity Flow Problem, *Journal of Artificial Intelligence Research*, Vol.1 pp. 1-23(1993).
- [2] Yamaki, H., Wellman, M.P. and Ishida, T.: A market-based approach to allocating QoS for multimedia applications, *Proceedings of the First International Conference on Multi-Agent Systems (Lesser, V.(ed.))*, MIT Press(1995).
- [3] 八槇博史, 山内裕, 石田亨: 市場モデルにおけるアプリケーション QoS の制御:実装上のトレードオフ, *情報処理学会論文誌*, Vol. 40, No.1, pp. 142-149(1999)
- [4] 田中慎司: QoS Market: A Market-Based Network QoS Control System, *Master-thesis*(2002)
- [5] 中塚康介: Caching Policy for Content Delivery Networks Based on User

- Preferences, Master-thesis(2002)
- [6] 石田亨, 片桐恭弘, 桑原和宏: 分散人工知能, コロナ社, 並列処理シリーズ 11(1996)
 - [7] 奥野正寛, 鈴木興太郎 : ミクロ経済学 I, 岩波書店 モダンエコノミックス 1(1985)
 - [8] Ka-Po Chow, Yu-Kwong Kwok: On Load Balancing for Distributed Multi-agent Computing, IEEE Transactions on Parallel and Distributed Systems, Vol.13, No.8, pp. 787-801(2002)