Master Thesis

Analyses of Contractors' Nash Equilibria on the Problem Solving Mechanism by Crowdsourcing

Supervisor Associate Professor Shigeo Matsubara

Department of Social Informatics
Graduate School of Informatics
Kyoto University

Masanori Hatanaka

February 2, 2010

Analyses of Contractors' Nash Equilibria on the Problem Solving Mechanism by Crowdsourcing

Masanori Hatanaka

Abstract

The crowdsourcing is considered as a new problem solving method using outside agents. In the crowdsourcing market, contractees announce their tasks, and then each of contractors chooses a task and reports the result if he/she completed the task. The contractee receiving reports from the contractors selects the best solution, and then pays reward to the contractor who reported the best solution.

However, there is not enough accumulation of knowledge about the operation and the management of the crowdsourcing market because of the novelty. When contractors are interested in their own utility, they select tasks selfishly and may gather the specific task. As a result, the allocation of tasks might be inefficient. If the impact of the property of tasks and the type of contractors on the social efficiency is revealed, we can get the guideline for dealing of tasks in the crowdsourcing market, and build a new problem solving mechanism based on the knowledge.

This paper considers the mechanism of crowdsourcing, and analyzes the social efficiency of it. Issues in the analysis of the social efficiency are listed as follows.

Modeling of the task allocation in the crowdsourcing market In many researches of resource allocation, it is assumed that each agent strategically reports its information, and discuss the determination mechanism of the allocation calculated by the reported information. Meanwhile in crowdsourcing, there is no top-down allocation because contractors autonomously select the task and carry it out. Therefore, we need to build a new model focusing on the strategic task selection.

Analysis of the social efficiency as a market In crowdsourcing market, the probability of successfully completing tasks becomes high and the social surplus is expected to be improved because tasks are carried out in parallel by contractors. However, the social inefficiency is also conceivable because

too many contractors might carry out the specific task. we need to know the impact of the asymmetry of tasks and the asymmetry of contractors on the efficiency as task allocation mechanism.

To solve the above issues, this paper formalizes the strategic task selection of contractors in crowdsourcing based on the congestion game. This paper defines the surplus ratio as the ratio between the social surplus in equilibrium and the optimal social surplus, and analyzes what extend the surplus ratio declines. This paper approaches the social efficiency analyses through two methods: one is the analysis of average impact of the asymmetry of tasks or contractors, and the other is that in the worst case of crowdsourcing models.

The contributions of this research are as follows.

Efficiency analyses in the case of asymmetric tasks and contractors
It is sufficient to analyze the surplus ratio for each combination of task allocations if the pure Nash equilibrium exists. This paper showed that a crowdsourcing with single type contractors had at least one pure Nash equilibrium, and that it was possible to generate an algorithm finding pure Nash equilibrium in the case of two tasks and two types, and so on. This paper analyzed the class having pure Nash equilibrium from these results, and revealed the parameter-dependent impact on the surplus ratio.

Theoretical analyses in the worst case equilibrium Previous congestion games assume that the amount of cost monotonically increases with the number of players, that is, the social surplus monotonically decreases. However, it might increase in crowdsourcing market because mass participation of contractors might boost the probability of successfully completing the tasks. This paper considers a crowdsourcing model as a new congestion model, and theoretically analyzes the efficiency in the worst case equilibrium. This paper showed that the social surplus in the worst case equilibrium was higher than half of the optimal social surplus in the restricted version of crowdsourcing game by applying "valid games".

The above contributions gave us a useful guideline on building a new problem solving market by crowdsourcing, and gave us a basic knowledge for designing an efficient task allocation mechanism in the future.

クラウドソーシングによる問題解決手法のタスク請負者に関する均衡分析

畠中 将徳

内容梗概

近年，Web を利用したクラウドソーシングという仕組みが注目を集めている．クラウドソーシングでは，タスクの依頼者が Web を通じてタスクの詳細や成功報酬を公開し，Web にアクセス可能な請負者が，提示されたタスクの中から自律的にタスクを選択，実行する．そして，個々の請負者はタスクの実行結果を依頼者に報告し，依頼者は最適の解を提示した請負者に報酬を支払う．

しかしながら，クラウドソーシングという仕組みは新しく，その効率性に関して十分な知見が得られていない．例えば，請負者が自分の効用のみに関心がある場合を想定すると，請負者の利己的なタスク選択により，特定のタスクに請負者が集中し，効率的なタスク割当が行われない可能性が考えられる．タスクの性質や請負者の能力による効率性への影響が明らかであれば，クラウドソーシングを用いて新たな問題解決市場を構築する際に，扱うタスクの性質に関するガイドラインを与えることが出来る．そこで，本研究では，クラウドソーシングのメカニズム自体に着目し，その社会的効率性について分析することを目的とする．クラウドソーシングの社会的効率性を分析する上では，次のような課題がある．

クラウドソーシングにおけるタスク割当のモデリング オークションなどの多くの競争的割当問題のモデルでは，タスクの請負者が自分の能力などについて何らかの情報提示を行い，それに基づく割当の決定方法について議論している．一方クラウドソーシングでは，請負者が自律的にタスクを選択，実行するため，トップダウンの割当が行われるわけではない．したがって，請負者の戦略的なタスク選択行動を考慮したモデルを構築する必要がある．

クラウドソーシングの市場としての社会的効率性 クラウドソーシングでは，複数の請負者によって並行にタスクが実行されるため，タスクの成功確率が高くなり，より高い社会的余剰が得られることが期待される．しかしながら，市場全体の効率性を考えると，特定のタスクへの過剰参入による非効率が起こっている可能性が想定できる．そこで，市場に存在するタスクの非対称性や，個々の請負者の能力の非対称性と，タスク割当の効率性の関

係について分析する必要がある．

以上の課題を解決するために，本研究では，クラウドソーシングにおける請負者の戦略的タスク選択行動を，混雑ゲームに基づき定式化する．本研究では，ナッシュ均衡における社会的余剰と，トップダウンに最適割当を計算したときの社会的余剰の比率を余剰比として定義し，どの程度低下するのかについての考察を行う．また，タスクや請負者の非対称性が均衡における効率性に与える平均的な影響の解析と，メカニズムとしての最悪ケースにおける効率性の解析という 2 つのアプローチを用いる．

本研究における主な貢献は次の 2 点である．

非対称なタスクや請負者が効率性に与える影響の解析　もし純粋ナッシュ均衡が存在するならば，各タスクに対する割当の組み合わせだけ計算すればよいことになる．そこでまず，クラウドソーシングモデルにおける純粋ナッシュ均衡の存在可能性について解析を行った．結果として，請負者のタイプが単一の場合には，必ず純粋ナッシュ均衡が存在すること，また，2 タスク，2 タイプという限定的な場合には，純粋ナッシュ均衡を発見するアルゴリズムが生成できること等を示した．これらの結果に基づき，実際に純粋ナッシュ均衡が存在するクラスの問題を解析し，各パラメータの変化による余剰比への影響を明らかにした．

最悪ケースの均衡における効率性の理論的解析　均衡が複数存在するような場合，実社会では，どの均衡が実現されるのかは未知である．従来の混雑ゲームでは，費用の総和が参加人数に対して単調増加，つまり，余剰の総和が単調減少であることを仮定していたが，クラウドソーシングでは，多くの請負者が参加することにより，タスク成功の可能性が高くなる可能性がある．効率性を理論的に保証するために，本研究では，クラウドソーシングモデルを新たな混雑ゲームのモデルであると想定し，最悪ケースの均衡における効率性について理論的な解析を行った．本研究では，既存の枠組みである「妥当ゲーム」の考え方を応用することで，クラスを限定したクラウドソーシングゲームにおいて，均衡における社会的余剰が，最適な社会的余剰の半分より大きくなることを示した．

上記 2 点の貢献により，クラウドソーシングを用いた新たな問題解決市場の構築における有用なガイドラインを得ると共に，将来より効率的なタスク割当方式の設計を行うための基礎的な知見が得られた．

Analyses of Contractors' Nash Equilibria on the Problem Solving Mechanism by Crowdsourcing

Contents

Chapter 1 Introduction

So far, activities such as R & D, marketing are closed to an organization. According to the development of information networks, we can find these activities are done by peoples co-creation through huge information network beyond the boundary of the organizations. One of these activity is crowdsourcing.

Howe defined crowdsourcing as the act of taking a job traditionally performed by a designated agent (usually an employee) and outsourcing it to an undefined, generally large group of people in the form of an open call[1]. Notable examples of the model include InnoCentive and Amazon Mechanical Turk. InnoCentive[1] is a global, online marketplace where organizations in need of innovation including companies, academic institutions, public sector, and non-profit organizations can utilize a global network of over 160,000 of the world-wide problem solvers. Amazon Mechanical Turk[2] is a crowdsourcing marketplace that enables computer programs to co-ordinate the use of human intelligence to perform tasks which computers are unable to do.

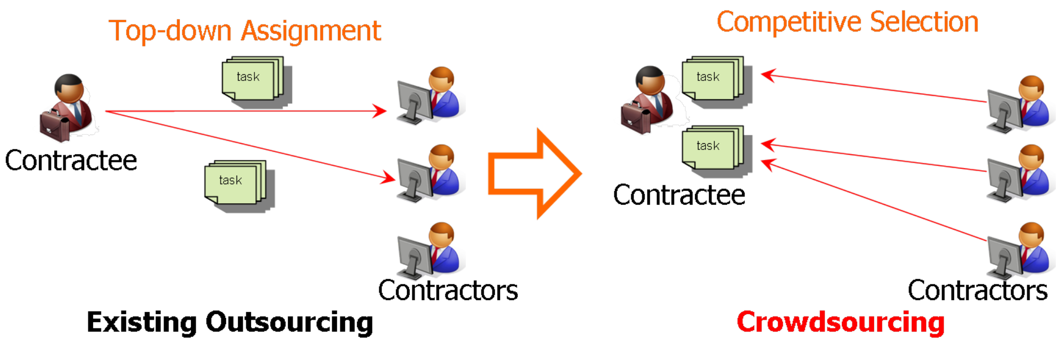


Figure 1: Model of Crowdsourcing Market

Consider the merit of utilizing crowdsourcing market compared to existing outsourcing market. The difference of task allocation mechanism between existing outsourcing and crowdsourcing is shown as Figure 1. In the existing outsourcing, tasks are allocated to contractors with high capabilities by con-

[1] http://www.innocentive.com/
[2] https://www.mturk.com/mturk/welcome

tractees. This is top-down allocation system. It is costful for contractees to evaluate many contractors. Contractees usually select some of potential contractors, evaluate them, allocate the task to the most efficient contractor, and pay the contractor. However, the probability of successfully completing a task increases when some contractors carry it out in parallel. It is more efficient for contractees to allocate a task which is valuable and low incurred cost. In crowdsourcing market, we can expect to realize more satisfactory task allocation because contractors autonomously select the task and they carry it out in parallel.

Crowdsourcing is promising as a new method for a problem solving. However, the social efficiency of it is questionable. If contractors behave selfishly, numinous number of contractors may carry out the specific task. Thus, efficient task allocations may be not realized. Furthermore, sufficient knowledge about its operation has not been accumulated. For example, how contractor's task selection and the payment method affect the contractor's incentive and the social efficiency in the problem solving is not clear. There is a critique that contractees and contractors are interested in their own utility and they are not interested in social efficiency. However, if its problem solving method is socially inefficient, crowdsourcing itself cannot become a persistent service. Thus, the viewpoint of social efficiency is important. If we understand the impact of the difference of tasks or that of contractors' abilities on the social efficiency, we can get the guideline for dealing of tasks in the crowdsourcing market, and build a new problem solving mechanism based on the knowledge. Therefore, it is quite important to understand the crowdsourcing from the viewpoint of mechanism design.

This research investigates a task allocation method used in InnoCentive and examines its social efficiency. In InnoCentive, a contractee announces the research problem, then contractors in the network try to solve the problem. They report the results if they can obtain solutions. The contractee examines these reports, select the best solution, and then pay reward to the contractor who reported the best solution. If the contractee does not satisfy any reports, the contractee does not have to pay anything. That is, the contractors autonomously select the task to tackle. This is different from traditional task allocation methods that the contractee evaluates the contractors' ability and selects the best contractor. Therefore, economics incentive of contractors should be examined in the different way from the existing studies about task allocation methods.

An auction might be considered as an efficient method for this task allocation problem. However, an auction seems difficult to find an efficient task allocation because of the nature of crowdsourcing. In crowdsourcing the contractee is not asked to pay more than pre-specified payment at the beginning. Even if more than one contractors find solutions, the contractor is sufficient to pay reward only to the contractor who reported the best solution, that is, is not required to pay reward to all the contractors who reported solutions. In addition, a contractee prefers that his/her task is chosen by as many contractors as possible because it increases the probability to find the better solution. Thus, the contractee does not have an incentive to limit the number of contractors. On the other hand, a contractor prefers trying to complete a task to doing nothing if it gives a positive expected utility. Thus, a contractor does not have an incentive to stop trying to complete the task in order to maximize the social surplus. Therefore, employing an auction seems difficult to find an efficient agent allocation. Employing Gale-Shapley algorithm[2] might be possible if we consider the crowdsourcing as a two-sided matching between tasks and contractors. However, it seems difficult to apply because of the same reason as an auction. Therefore, we need to analyze the mechanism of crowdsourcing with different model from existing auctions or two-sided matching mechanisms.

I suppose that contractors in the crowdsourcing market choose the task considering the trade-off between the expected reward for carrying out the task and the incurred cost. Researchers in traffic network fields analyze the equilibrium of networks with the congestion game model, so the same approach seems to be applicable to the problem of crowdsourcing.

This paper formalizes the strategic task selection of contractors in crowdsourcing based on the congestion game. Researches of congestion game models mainly refer the existence of pure Nash equilibrium or the efficiency in the worst case equilibrium. This paper defines the surplus ratio as the ratio between the social surplus in equilibrium and the optimal social surplus, and analyzes what extend the surplus ratio declines. This paper approaches the social efficiency analyses through two methods: one is the analysis of average impact of the asymmetry of tasks or contractors, and the other is that in the worst case of crowdsourcing models.

The remainder of this paper is as follows. First, in Chapter 2, this paper refers the related works about congestion games and equilibrium analyses. In Chapter 3, this paper describes the crowdsourcing model for rigorous discussion, and formalizes it as a crowdsourcing game. If the pure Nash equilibrium always exists in crowdsourcing games, it is sufficient to evaluate only for each combination of task allocations. Thus, this paper first examines the existence of pure Nash equilibrium in crowdsourcing games in Chapter 4. From those results, the average impact of the property of contractors and tasks on the social efficiency about the class having pure Nash equilibrium is analyzed in Chapter 5 and Chapter 6. Chapter 5 considers single type of contractors, while multiple types of contractors are considered in Chapter 6. This paper also analyzes the worst case equilibrium of crowdsourcing models, which is called the price of anarchy, in Chapter 7, and concludes in Chapter 8.

Chapter 2 Related Works

In this chapter, I will describe some previous works related to the congestion games or efficiency analyses.

2.1 Congestion games

Congestion game models have been actively studied in traffic research and communication network research. The beginning of this research was the seminal work by Rosenthal and they proposed the class having pure Nash equilibrium[3].

The basic model is described as follows. A set of m facilities and a set of n players exist, and the cost functions is defined as $c_{m_j}(n_{m_j})$. The cost function is a monotonically increasing function as the number of players who select the same facility m_j. Which facility players choose is their strategy. Let s_i be the facility chosen by player i, and Let $s = (s_1, s_2, ..., s_n)$ be a profile of all players' strategies. a profile s is pure Nash equilibrium if and only if the following inequality is satisfied for all players.

$$c_{s_i}(n_{s_i}) \leq c_{s'_i}(n_{s'_i} + 1), \forall s'_i \neq s_i$$

Rosenthal proved that this class of game always has pure Nash equilibrium. In fact, the process in which each player repeatedly choose the facility which minimize his/her cost converges the pure Nash equilibrium.

In the field of congestion game, some extensions are discussed. Fotakis, et al. studies weighted congestion games in which each player is weighted about the cost function[4]. They show that the pure Nash equilibrium also exists in singleton weighted congestion games, in which each player is allowed to choose only one facility. Milchtaich studies the player-specific congestion game in which each player has different cost function about the same facility. The difference between these models and the crowdsourcing model is described at length in later section.

There are few researches about application of the congestion game to the task allocation mechanism. One of the examples is the formalization of a wireless network caching problem as the market sharing game, which is the special case of congestion game[5]. In the market sharing game, there are some markets and players, and players choose the most profitable market. The whole profit of each market is given, so players in the same market share the profit equally. However, this model does not assume the failure of players in the market and the asymmetry of players. In a crowdsourcing market, contractors might fail to complete tasks, and they might have different probabilities of success for tasks. Therefore, this model is insufficient to analyze the efficiency of crowdsourcing.

2.2 Efficiency analyses

A previous research about efficiency of crowdsourcing mechanism is formalization of it as one of special auctions. Dipalantino, et al. consider the crowdsourcing as the all-pay auctions, and analyze it as a system of competitive contests[6]. They discuss the relation between reward and participation rate. It gives us the guideline about the setting of rewards. However, this paper discusses the social efficiency of the crowdsourcing as a task allocation system. It is different from their discussions.

Crowdsourcing mechanism is closely related to the fault tolerant mechanism in that the realization of the efficient allocation and the high probability of success for the task. Zhao, et al. discuss caching mechanisms in order to guarantee the fault tolerance of data or services in the information network[7]. They discuss how to maintain the high probability of service providing. However, they do not deal with strategic actions of agents. Porter, et al. study about resource allocation mechanisms in which they consider not only the execution cost of each agent, but also the success probability of tasks[8]. They deal with the individual rationality of agents, and discuss the mechanism to maximize the social surplus. However, they assume that a centric designer aggregates information about agents, and determine the resource allocation. They discuss mechanisms in which agents are honest and report their types. In the crowdsourcing, contractors autonomously select tasks. They don't need to report their types, so top-down assignment by contractees are not carried out. Therefore, we must deal with more distributed resource allocation problems. In crowdsourcing, the same tasks can be allocated more than two contractors in parallel unlike to the fault tolerant mechanism.

Stemming from research of Arrow, et al.[9], various models about allocation of resources for invention have been studied in the field of microeconomics. Barzel points out the possibility of overinvestment by the tragedy of commons[10]. Gilbert and Newbery insist in their paper that the investment for invention have the role as a strategy for preventing new entry. However, they deal with models in oligopoly situation by a few contractors, and discuss the rational expenditures for invention. This research considers that large indefinite numbers of contractors exist, and focus on the efficiency by contractors' decision making processes in selecting tasks. Therefore, it is different from the analysis of deciding the amount of investments for invention.

The "price of anarchy" of the class of congestion game have been actively studied about the efficiency of the model. The price of anarchy is defined as the ratio between the optimal value of the social utility function and the value in the worst case equilibrium.

There are two main models in congestion game fields which is different from the definition of the social utility function.

One is called the **KP** model[11]. The **KP** model has considered a simple network consisting of m parallel links, from source to destination; each link bears a capacity . Selfish traffic is modeled as a finite collection of users, each bearing an unsplittable traffic, and shipping it using a mixed strategy. In a Nash equilibrium, no expected individual cost can be unilaterally decreased. The social utility function is the expectation of the maximum amount of cost for each link; the optimum is the least possible maximum link. **KP** model is mainly utilized as the computational resource allocation model, in which each computational resource has different capability of computation.

The other is called the **W** model by Wardrop[12]. The **W** model has considered arbitrary multicommodity networks (that is, networks with multiple sources and destinations), with a cost function for each link. Selfish traffic is modeled as a splittable flow. The individual cost for a path (from source to destination), is the sum of the costs incurred on its links. In a Wardrop equilibrium, all (used) paths have the same individual cost. In the **W** model, players may be thought of as (infinitely many) non-atomic entities, each carrying infinitesimal traffic. The social utility function is the sum of individual costs. The optimum is the least possible, over all flows, sum of costs. Roughgarden discusses the selfish routing of agents in the network routing[13]. The ratio between the cost of equilibrium flows and that of an optimal flow are evaluated in this field. However, in the area of resource allocation systems by crowdsourcing, the comparison between the optimal allocation and the allocation in equilibrium has not been discussed.

This paper assumes that tasks are independent, so it is reasonable to consider the crowdsourcing as a simple network consisting of m parallel links such as the **KP** model. However, we are interested in the social efficiency of crowdsourcing, so it is also reasonable to consider the social utility function as the sum of participants' utilities as evaluated in the **W** model. Recently, the hybrid model was studied[14][15]. However, they and these models mentioned above evaluate the efficiency about the costs.

In the crowdsourcing market, the contractor will not participate if the expected reward of the task is lower than the cost. Therefore, we cannot apply the discussion of the efficiency about the costs directly to the discussion of the efficiency in the crowdsourcing market. In addition, previous congestion game models assume only the case in which the sum of costs increases as the number of players increases, while the crowdsourcing assumes that the sum of expected utilities can increase as the number of players increases because of the increment of the probability of successfully completing the task.

Chapter 3 Model

In this chapter, I will present a formal model to enable rigorous discussion. The concept model that we assume in this paper is shown as Figure 2.

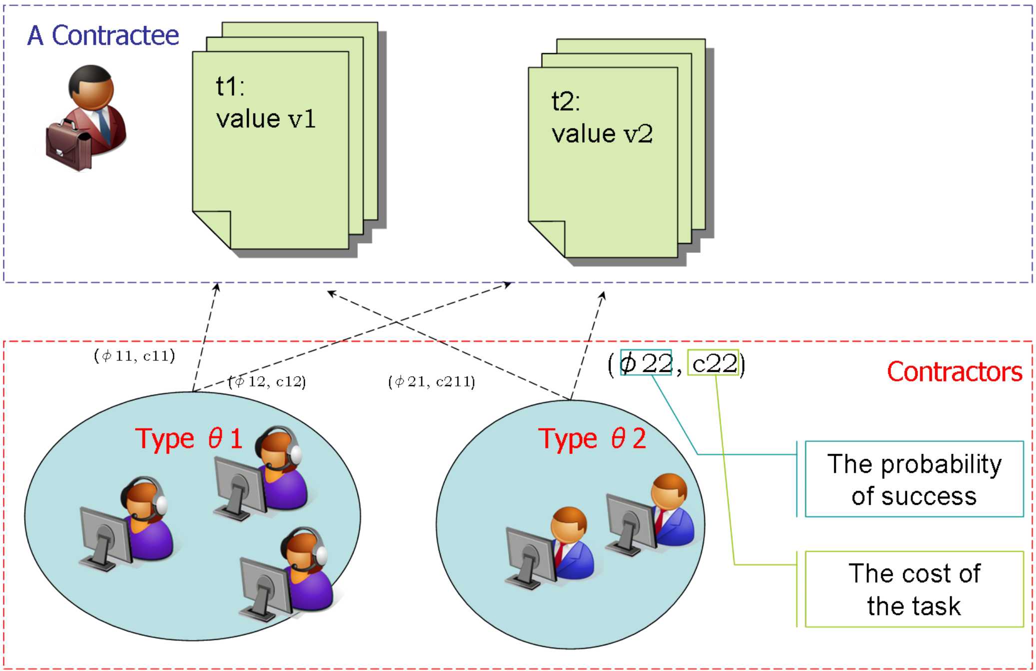


Figure 2: Model of Crowdsourcing Market

In a crowdsourcing market, there exist a contractee R, n contractors $\{A_k\}$, and a marketplace operator M. The contractee has tasks t_j $(j = 1, \cdots, l)$. This paper focuses on investigating contractors' behaviors. Thus, assuming a single contractee is sufficient for discussions. If task t_j is successfully completed by the contractor, the contractee enjoys the utility of v_j. Completing each task is independent from completing the other tasks. This paper assumes that $r_j (0 \leq r_j \leq v_j)$ is announced as the amount of reward for completing task t_j before a deadline to report the result. This means that the amount of reward is given and no strategic manipulation on it is assumed. For the sake of simplicity, I assume that $v_j = r_j$ in the following sections, and utilize the unified notation v_j. A market operator is assumed to be interested in maximizing social surplus.

On the other hand, contractors are characterized as type θ_i ($i = 1, \cdots, m$). A contractor A_k's type is denoted by $typeof(A_k)$. If a contractor of type θ_i chooses task t_j, he/she can successfully complete the task with a probability of $\phi_{ij}(0 \leq \phi_{ij} \leq 1)$, incurs the cost of $c_{ij}(c_{ij} \geq 0)$. It may happens that if a contractor invests more resources, which incurs a larger cost, the probability of success ϕ_{ij} increases. However, this paper assumes that the amount of cost is a constant value of c_{ij} if the type of the contractor θ_i and task t_j are given. In addition, this paper assumes that a contractor of type θ_i knows its probability of success ϕ_{ij} and its cost of c_{ij}. It may be also considered that a contractor gain a private value by carrying out a task. Creators in iStockPhoto market[1] and programmers in TopCoder market[2] are the real-world example because they are able to train their skills throughout the participation for the crowdsourcing market. However, the private value is not more than the incurred cost in general cases. In addition, this paper will discuss the one-stop situation of the crowdsourcing, that is, the incurred cost and the private value are usually independent on whether the task is completed or not. Hence, this paper doesn't assume the private value.

The protocol in crowdsourcing market is shown as Figure 3. The following procedure is utilized.

(Step 1) a contractee registers his/her tasks with its detailed description including deadline and an amount of reward.

(Step 2) a market operator announces the registered tasks to contractors.

(Step 3) contractors choose their task to be tackled. Here, we assume that contractors randomly come to the market and choose a task. This paper assumes that a contractor choose a single task to make a discussion simple.

(Step 4) a contactor reports its result of the completed task to the contractee.

(Step 5) a contractee selects the best report and pay reward of r_j to the contractor who reported the best one. If more than one best reports exist, a contractee chooses one among them by using a pre-specified method such as choosing at random, choosing the earliest one.

[1] http://www.istockphoto.com/

[2] http://www.topcoder.com/

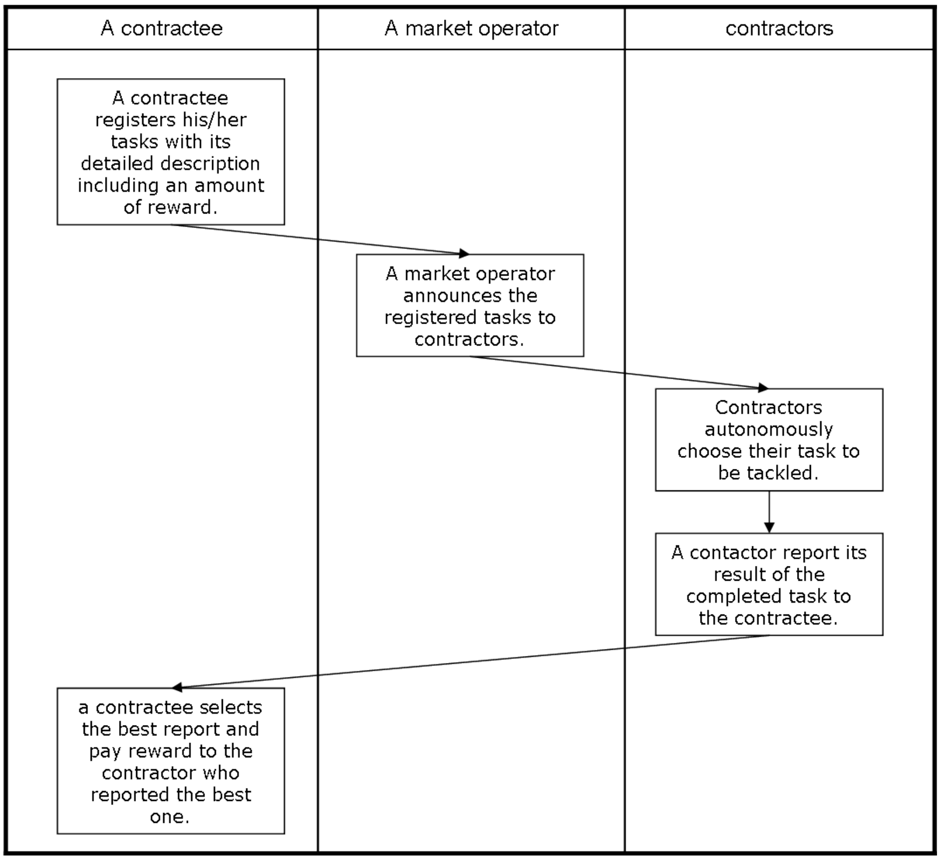


Figure 3: The protocol in crowdsourcing market

I present the above model in strategic form of game, which is called "crowdsourcing game", in order to discuss the Nash equilibrium in later section. The selectable strategy set of each contractor A_k is denoted by $S_k = \{\{t_j\} \cup \emptyset\}$. $\emptyset$ represents that the contractor doesn't choose any tasks. When each contractor chooses $s_k \in S_k$, the tuple of all contractors' strategy is denoted by $s = \{s_k\}$. I define the function $n_{ij}(s)$ as the number of contractors whose type is θ_i and task is t_j in the strategy tuple s. In addition, I utilize the following special notations which are frequently used in game theoretical discussions.

- $s_{-k} = (s_1, ..., s_{k-1}, s_{k+1}, ..., s_n)$

- $(s_k^{'}, s_{-k}) = (s_1, ..., s_{k-1}, s_k^{'}, s_{k+1}, ..., s_n)$

If a contractor A_k of type θ_i chooses task t_j in the strategy tuple s, his/her expected utility $u^{\theta_i}(s)$ is defined as follows.

$$u^{\theta_i}(t_j, s_{-k}) = v_j \sigma_{ij} - c_{ij}$$

where σ_{ij} represents the probability of becoming a winner among contractors choosing task t_j. the value of σ_{ij} depends on the contractors' probability of success and the number of participants for the task. Individual rationality holds if the expected utility of each contractor is larger than zero or equal to zero.

In this research, social surplus is calculated as the sum of all participants' expected utilities. This paper assumes that a contractee pays reward of r_j that is equal to his/her valuation of completing task t_j. Thus, the contractee's utility is equal to zero whether the task is succeeded or failed. The market operator's utility is also equal to be zero. Therefore, social surplus $\gamma(s)$ can be calculated as the sum of the expected utility of all the contractors. That is,

$$\gamma(s) = \sum_{j}^{l} \sum_{A_k : s_k = t_j} u_j^{typeof(A_k)}$$

The social surplus can be calculated as the sum of the expected surplus of task t_j as well. The probability of completing task t_j after the allocation of contractors is determined is denoted by p_j.

Here, I explain the difference of three probabilities $\phi_{ij}, \sigma_{ij}, p_j$, and the relationship. First, ϕ_{ij} represents the ability of contractors, and is given by the constant value in this paper. Next, σ_{ij} represents the winning probability when the type θ_i's contractor carry out the task t_j, depends on the probabilities of success of other types of contractors and the number of contractors who carry out the task, and is denoted by the function such that $\sigma_{ij}(\{\phi_i j\}_i, \{n_i j\}_i)$. And p_j represents the probability of successfully completing the task t_j, denoted by the following expression.

$$p_j = 1 - \prod_{A_k : s_k = t_j} (1 - \phi_{typeof(A_k)j})$$

p_j also represents the sum of the winning probability about task t_j, that is, $p_j = \sum_{A_k : s_k = t_j} \sigma_{typeof(A_k)j}$.

This paper introduces a surplus ratio R to measure social inefficiency. The surplus ratio is calculated by dividing social surplus in an equilibrium allocation by social surplus in an optimal allocation.

Chapter 4 The Pure Nash Equilibrium

This chapter will describe the property of Nash equilibrium in crowdsourcing games. Generally speaking, there are two types of Nash equilibrium; one is the pure Nash equilibrium and the other is the mixed Nash equilibrium. The pure Nash equilibrium represents the equilibrium in which each player cannot improve the utility when the player choose another strategy with probability one. On the other hand, the mixed Nash equilibrium represents the equilibrium of the probability distribution when each player can stochastically choose strategies. It is proved by Nash that any finite, k-person, non-cooperative game has at least one (mixed) Nash equilibrium. However, the existence of the pure Nash equilibrium depends on the definition of games. If the pure Nash equilibrium exists, it is sufficient to evaluate only for each combination of task allocations. To examine the existence of the pure Nash equilibrium, I first introduce the potential game.

Definition 1. *Let $g_i(s)$ be the utility function of the player i. Given the strategy tuple $s = \{s_1, ..., s_n\}$, we call the function $f : S \to \mathbb{R}$ satisfying the following condition the potential of the game.*

$$g_i(s_i, s_{-i}) - g_i(s_i^{'}, s_{-i}) = f(s_i, s_{-i}) - f(s_i^{'}, s_{-i})(s_i, s_i^{'} \in S_i, i \in N)$$

And we call a game which has a potential f a potential game.

The potential game is formulated by Monderer, et al.[16], and its useful property is proved as follows.

Theorem 1. *Any finite k-person, non-cooperative potential game has at lease one pure Nash equilibrium.([16])*

Therefore, the fact that the non-cooperative game has a potential is sufficient condition for the existence of the pure Nash equilibrium.

First, I consider the case of single type contractors.

Theorem 2. *The crowdsourcing game with single type contractors has a potential.*

Proof. When the type of contractors is unique, that is, only θ_1, the expected utility of contractor A_k is calculated as follows.

$$g_k(t_j, s_{-k}) = \frac{v_j(1-(1-\phi_{1j})^{n_{1j}(s)})}{n_{1j}(s)} - c_{1j}$$

In this expression, the term of $(1-(1-\phi_{1j})^{n_{1j}(s)})$ represents the probability that at least one contractor can successfully complete the task. Each contractor has the same probability of receiving the reward, because we assume that all the contractors have the same type. Therefore, $v_j(1-(1-\phi_{1j})^{n_{1j}(s)})/n_{1j}(s)$ represents the contractor's expected reward.

It is sufficient to prove arbitrary potential f exists, so I define a function f as follows.

$$f(s) = \sum_{j}^{l} \sum_{k}^{n_{1j}(s)} \{\frac{v_j(1-(1-\phi_{1j})^{k})}{k} - c_{1j}\}$$

If a contractor A_k change his/her task from t_j to t'_j,

$$\begin{aligned} & f(t_j, a_{-k}) - f(t'_j, a_{-k}) \\ = \; & \{\frac{v_j(1-(1-\phi_{1j})^{n_{1j}(t_j,a_{-k})})}{n_{1j}(t_j, a_{-k})} - c_{1j}\} - \{\frac{v_j(1-(1-\phi_{1j})^{n_{1j}(t'_j,a_{-k})})}{n_{1j}(t'_j, a_{-k})} - c_{1j}\} \\ = \; & g_k(t_j, a_{-k}) - g_k(t'_j, a_{-k}) \end{aligned}$$

the condition holds.

Even if a contracter A_k quits the task t_j,

$$\begin{aligned} & f(t_j, a_{-k}) - f(\emptyset, a_{-k}) \\ & = \; \{\frac{v_j(1-(1-\phi_{1j})^{n_{1j}(t_j,a_{-k})})}{n_{1j}(t_j, a_{-k})} - c_{1j}\} \\ & = \; g_k(t_j, a_{-k}) \\ & = \; g_k(t_j, a_{-k}) - g_k(\emptyset, a_{-k}) \end{aligned}$$

the condition holds as well. Thus, the function f is a potential of the crowdsourcing game with single type contractors. □

Corollary 1. *A crowdsourcing game with single type contractors has at least one pure Nash equilibrium.*

Next, I consider the multiple types contractor case. If multiple types of contractors exist, the potential function defined in the above proof cannot be applicable.

First, I describe some extended class of congestion game in previous researches. In the simple congestion game, it is assumed that the cost function of each facility j depends only on the number of players n_j who choose the same facility, and the function is denoted by $c_j(n_j)$. In (unweighted) player-specific congestion game, the cost function is player-specific, and is denoted by $c_{ij}(n_j)$ of player i. Furthermore, in weighted player-specific congestion games, each player has weight β_i and the number of players who choose the same facility j is calculates as $n_j = \sum_{i:a_i=j} \beta_i$.

Milchtaich characterizes the existence of pure Nash equilibrium about each class of congestion games.

Theorem 3. *A unweighted player-specific congestion game has at least one pure Nash equilibrium.([17])*

Theorem 4. *Any k-player, two-strategy, weighted player-specific congestion game has at least one pure Nash equilibrium. In addition, Any l-strategy, two-player, weighted player-specific congestion game has at least one pure Nash equilibrium.([17])*

Theorem 5. *Any k-player, l-strategy, weighted player-specific congestion game has the possibility of not having pure Nash equilibrium.([17])*

Milchtaich shows the actual three-player, three-strategy case that has no pure Nash equilibrium. And now, I describe the difference between the crowdsourcing model in this paper and existing congestion game models. In the crowdsourcing model, if the tuple of strategies is $s = (t_j, s_{-k})$, the expected utility of type θ_i's contractor A_k is represented as follows.

$$g_k(t_j, s_{-k}) = v_j \sigma_{ij}(\{\phi_{ij}\}_i, \{n_{ij}\}_i) - c_{ij}$$

When the type $\theta_{i'}$'s contractor who chooses the same task t_j at s has the probability of success $\phi_{ij} \neq \phi_{i'j}$, the winning probability $\sigma_{ij}(\cdot)$ is also different, therefore the expected utility is player-specific. Moreover, if other types of contractors denoted by θ_x, θ_y exist, and $\phi_{xj} > \phi_{yj}$, the winning probability of contractor A_k become higher if the type θ_y's contractor newly carry out the task t_j than the type θ_x's contractor newly carry out it. Thus, the impact of each contractor's participation on the expected utility is weighted. Furthermore, if the probabilities of success for another task $t_{j'}$ are $\phi_{xj'}$ and $\phi_{yj'}$ such that $\phi_{xj'} < \phi_{yj'}$, the weight at the same player differs from the choosing tasks. The above discussion shows that crowdsourcing games with multiple types of contractors belong to the more complicated class of congestion games.

Corollary 2. *A (general) crowdsourcing game has the possibility of not having pure Nash equilibrium.*

It is generally said that crowdsourcing games with multiple types of contractors do not always have the pure Nash equilibrium. However, I can generate the algorithm to find the pure Nash equilibrium in the concrete case of two types and two tasks.

Theorem 6. *A two-task, two-type crowdsourcing game always has at least one pure Nash equilibrium.*

Proof. I present the concrete algorithm to find the pure Nash equilibrium. The flow of algorithm is shown as Algorithm 1.

Assume that n_1 number of type θ_1's contractors and n_2 number of type θ_2's contractors exist. Let n_{ij} be the number of type θ_i's contractors for task t_j. and an allocation to the tasks is also denoted by the taple $(n_{11}, n_{12}, n_{21}, n_{22})$. There are constraints such that $n_{11} + n_{12} = n_1$, $n_{21} + n_{22} = n_2$. And let $g_j^{\theta_i}(n_{1j}, n_{2j})$ be the expected utility of type θ_i's contractors to the task t_j.

The algorithm works as follows. First, we set the initial allocation as $(n_1, 0, 0, n_2)$. Next, We find the pair (n_{11}, n_{12}) which satisfies following two inequalities.

$$g_1^{\theta_1}(n_{11}, 0) > g_2^{\theta_1}(n_{12} + 1, n_2)$$
$$g_1^{\theta_1}(n_{11} + 1, 0) < g_2^{\theta_1}(n_{12}, n_2)$$

Following that, We also find the pair (n_{21}, n_{22}) which satisfies following two inequalities.

$$g_1^{\theta_2}(n_{11}, n_{21}) > g_2^{\theta_2}(n_{12}, n_{22} + 1)$$
$$g_1^{\theta_2}(n_{11}, n_{21} + 1) < g_2^{\theta_2}(n_{12}, n_{22})$$

If the pair (n_{21}, n_{22}) does not change, the allocation in this point is obviously the pure Nash equilibrium. If it changes, we find the appropriate pair

(n_{11}, n_{12}) again in the same manner. By repeating this procedure, we can find the convergence point of the number of contractors. The convergence point in this process is always the pure Nash equilibrium, if exists, because contractors always seek to the best response strategy.

Algorithm 1 The process of finding pure Nash equilibrium

Finding the pure Nash equilibrium

$(n_{11}, n_{12}, n_{21}, n_{22}) \leftarrow (n_1, 0, 0, n_2)$;
$(n_{11}^{TMP}, n_{12}^{TMP}, n_{21}^{TMP}, n_{22}^{TMP}) \leftarrow (n_1, 0, 0, n_2)$;
$end \leftarrow false$;
while $\neg end$ **do**
  find $(n_{11}^{TMP}, n_{12}^{TMP})$ which satisfies:
  $g_1^{\theta_1}(n_{11}^{TMP}, 0) > g_2^{\theta_1}(n_{12}^{TMP} + 1, n_2)$ and
  $g_1^{\theta_1}(n_{11}^{TMP} + 1, 0) < g_2^{\theta_1}(n_{12}^{TMP}, n_2)$ and
  $n_{11}^{TMP} + n_{12}^{TMP} = n_1$;
  find $(n_{21}^{TMP}, n_{22}^{TMP})$ which satisfies:
  $g_1^{\theta_2}(n_{11}^{TMP}, n_{21}^{TMP}) > g_2^{\theta_2}(n_{12}^{TMP}, n_{22}^{TMP} + 1)$ and
  $g_1^{\theta_2}(n_{11}^{TMP}, n_{21}^{TMP} + 1) < g_2^{\theta_2}(n_{12}^{TMP}, n_{22}^{TMP})$ and
  $n_{21}^{TMP} + n_{22}^{TMP} = n_2$;
  if $(n_{11}, n_{12}, n_{21}, n_{22}) = (n_{11}^{TMP}, n_{12}^{TMP}, n_{21}^{TMP}, n_{22}^{TMP})$ **then**
    $end \leftarrow true$;
  else
    $(n_{11}, n_{12}, n_{21}, n_{22}) \leftarrow (n_{11}^{TMP}, n_{12}^{TMP}, n_{21}^{TMP}, n_{22}^{TMP})$;
  end if
end while

It is possible to prove the convergence of this process. Type θ_1's contractors can improve their expected utility only if the number of participants for task t_2 increases. At the previous step, the number of type θ_2's contractors for task t_1 increases, so the expected utility of type θ_1's contractors about task t_1 decreases, and that of task t_2 relatively increases. Therefore, the number of type θ_1's contractors who choose the task t_2 always increases in the process.

On the other hand, the number of type θ_2's contractors who choose the task t_1 always increases from the same discussion. Since the permanent increment is impossible, the process converges. □

This algorithm is useful for simulating the actual surplus ratio achieved at pure Nash equilibrium. I utilize it for the efficiency analysis in later section.

Chapter 5 Single Type Contractor Case

This chapter will examine the case that all the contractors have the same type, that is, a single type of contractors' case.

First, I illustrate an example of the evaluation of the surplus ratio. Here, I assume that n contractors of single type θ_1 exist and two tasks of t_1 and t_2 exist. The numbers of contractors who carry out task t_1 and t_2 are n_1 and n_2, respectively ($n_1 + n_2 = n$). When only type θ_1's contractors exist, the expected utility of the contractor who chooses task t_j as follows.

$$u^{\theta_1}(s) = \frac{v_j(1-(1-\phi_j)^{n_j(s)})}{n_j(s)} - c_{1j}$$

In this expression, the term of $(1-(1-\phi_{1j})^{n_j(s)})$ represents the probability that at least one contractor can successfully complete the task. Each contractor has the same probability of receiving the reward, because we assume that all the contractors have the same type. Therefore, $v_j(1-(1-\phi_{1j})^{n_j(s)})/n_j(s)$ represents the contractor's expected reward. The term of c_{1j} represents the difference between the private value and the cost incurred by each contractor.

Table 1: The expected utilities of a contractor.

n_1	t_1	t_2	n_1	t_1	t_2
0	-	0.66196	10	1.61384	0.82927
1	3.00000	0.67772	11	1.48819	0.84724
2	2.82000	0.69362	12	1.36733	0.86546
3	2.64720	0.70987	13	1.25103	0.88392
4	2.48127	0.72626	14	1.13912	0.90262
5	2.32192	0.74287	15	1.03138	0.92158
6	2.16883	0.75970	16	0.92766	0.94079
7	2.02174	0.77675	17	0.82776	0.96026
8	1.88038	0.79402	18	0.73153	0.98000
9	1.74450	0.81153	19	0.63882	1.00000
			20	0.54946	-

Table 2: The social surplus in Nash equilibrium.

	n_1	n_2	n	ss(social surplus)
equilibrium solution	16	4	20	18.61
optimal solution	9	11	20	24.63

Suppose that both of tasks' rewards v_1 and v_2 are equal to 100, and $\phi_{11} = 0.06$, $\phi_{12} = 0.02$, for the probabilities of success, and $W_{11} = -3$, $W_{12} = -1$ for the costs.

Table 1 shows the expected utility of each n_{11} for the number of contractors $n = 20$, which tells that an equilibrium allocation is that $n_1 = 16$ and $n_2 = 4$. On the other hand, as shown in Table 2, an socially efficient allocation is $n_1 = 9$ and $n_2 = 11$. In this case, the surplus ratio is $R = 18.61/24.63 = 0.756$.

5.1 The number of contractors

Here, this section investigates the impact of the number of contractors on the surplus ratio. I assume that completing a task is independent from completing the other tasks, and consider each task separately. First, I examine an optimal contractor allocation. Because contractors have the same type, discussing only the number of contactors is sufficient. It is not necessary to consider which contractors should be included. If n contractors carry out the task t_j, the surplus is expressed as follows.

$$s^{t_j}(n) = v_j\{1 - (1 - \phi_{1j})^{n_j}\} - nc_{1j}$$

The term of v_j is the contractee's valuation of completing the task and the term of $(1 - (1 - \phi_{1j})^{n_j})$ represents the probability that at least one contractor succeeds to complete the task. If a contractor is added from n, the increment of the surplus is expressed as follows.

$$\begin{aligned}\Delta^{t_j}(n) &= s^{t_j}(n+1) - s^{t_j}(n)\\ &= v_j\{1-(1-\phi_{1j})^{n+1}\} + (n+1)w_{1j} - v_j\{1-(1-\phi_{1j})^{n}\} + nw_{1j}\\ &= v_j\{(1-\phi_{1j})^{n} - (1-\phi_{1j})^{n+1}\} + w_{1j}\\ &= v_j(1-\phi_{1j})^{n}\{1-(1-\phi_{1j})\} + w_{1j}\\ &= v_j\phi_{1j}(1-\phi_{1j})^{n} + w_{1j}\end{aligned}$$

If n is sufficiently large, $\Delta^{t_j}(n)$ becomes lower than zero. So there is a optimal number of contractors n_j^{OPT} in terms of task t_j. n_j^{OPT} can be obtained by differentiating the above equation with respect to n_j and letting it zero, which is represented as follows.

$$-v_j(1-\phi_j)^{n_j^{OPT}}\ln(1-\phi_j) - c_{1j} = 0$$

Thus, the following expression is obtained.

$$n_j^{OPT} = \ln(\frac{-c_j}{v_j\ln(1-\phi_j)})/\ln(1-\phi_j)$$

On the other hand, if a contractor is added from n, the expected utility of individual contractors is expressed as follows.

$$U^{t_j}(n) = \frac{v_j\{1-(1-\phi_{1j})^{n+1}\}}{n+1} - c_{1j}$$

Lemma 1. *$U^{t_j}(n)$,$\Delta^{t_j}(n)$ is the monotonically decreasing function in terms of n.*

Since $U^{t_j}(n)$ is the monotonically decreasing function, in the case of sufficient number of contractors, the number of participants at equilibrium, which is denoted by n_j^{NASH}, should satisfy $U^{t_j}(n_j^{NASH}) > 0$ and $U^{t_j}(n_j^{NASH}+1) < 0$.

Next, I discuss the relation between n_j^{NASH} and n_j^{OPT}. For preparation, I prove the following lemma.

Lemma 2. *For any $n \geq 0$, $U^{t_j}(n) \geq \Delta^{t_j}(n)$.*

Proof. If $n = 0$, the condition holds because of the following equations

$$U^{t_j}(0) = v_j\phi_{1j} - c_{1j} = \Delta^{t_j}(0)$$

I assume that the condition holds if $n = k (k > 0)$.

$$
\begin{aligned}
& \frac{v_j\{1-(1-\phi_{1j})^{k+1}\}}{k+1} \geq v_j\phi_{1j}(1-\phi_{1j})^{k} \\
\iff & \frac{v_j\{1-(1-\phi_{1j})^{k+1}\}}{k+1} \geq v_j\phi_{1j}(1-\phi_{1j})^{k+1}
\end{aligned}
\tag{1}
$$

If $n = k+1$ on the above assumption, the following inequalities holds.

$$
\begin{aligned}
U^{t_j}(k+1) - \Delta^{t_j}(k+1) &= \frac{v_j\{1-(1-\phi_{1j})^{k+2}\}}{k+2} - v_j\phi_{1j}(1-\phi_{1j})^{k+1} \\
&\geq \frac{v_j\{1-(1-\phi_{1j})^{k+2}\}}{k+2} - \frac{v_j(1-\phi_{1j})\{1-(1-\phi_{1j})^{k+2}\}}{k+2} \\
&= \frac{v_j}{(k+1)(k+2)}\{(k+2)\phi_{1j} + (1-\phi_{1j})^{k+2} - 1\} \\
&\geq 0
\end{aligned}
$$

Therefore, the condition also holds. □

The above result suggests the following corollary.

Corollary 3. *If sufficient number of contractors exists, the following condition is always satisfied.*

$$n_j^{NASH} \geq n_j^{OPT}$$

That is, in the crowdsourcing game with sufficient number of contractors, the number of participants is equal or more than the optimum.

As long as the expected utility is larger than zero or equal to zero, an additional contractor participates in doing task t_j. Thus, the maximum number of contractors n'_j choosing task t_j is the largest number that satisfies the following inequality.

$$v_j(1-(1-\phi_{1j})^{n_j^{NASH}})/n_j^{NASH} - c_j \geq 0$$

In the above expression if n'_{1j} is a large number, $(1-\phi_{1j})^{n'_{1j}}$ approaches to zero, for example, if the probability of success $\phi_{1j} = 0.5$ and n_{1j} is 10, $(1-\phi_{1j})^{n'_{1j}}$ becomes 0.000977. In this case, n'_{1j} can be approximated to be v_j/c_{1j}. This is the case that the cost incurred by each contractor is small compared to the amount of reward. In addition, social surplus is close to zero. So far, I have not discussed which task a contractor chooses if more than one tasks exist. If v_j/c_{1j} becomes large enough for all the tasks, contractor's expected utility approaches to zero, which means that a contractor is indifferent to a task selection.

From the above discussion, if v_j/c_{1j} is sufficiently large and a sufficiently large number of contractors exist, the surplus ratio approaches to zero, that is, social surplus in an equilibrium allocation approaches to zero.

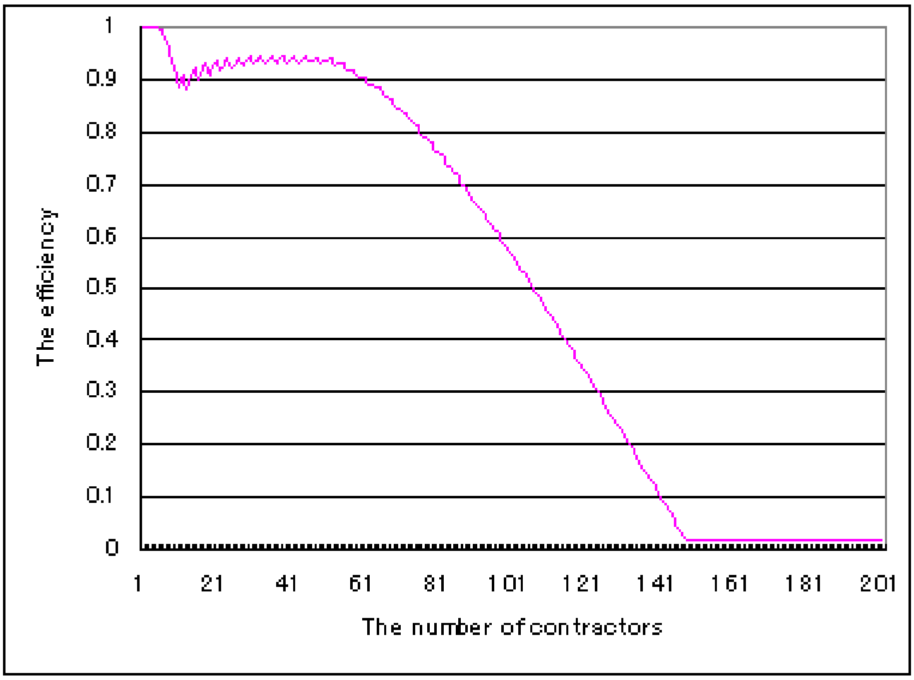


Figure 4: The relationship between the number of contractors and the social efficiency.

Figure 4 shows how the surplus ratio changes if the number of contractors increases. The horizontal axis represents the number of contractors n, while the vertical axis represents the surplus ratio. Here, I set $v_1 = v_2 = 100$, $c_{11} = c_{12} = 1$ and $\phi_{11} = 0.05$, $\phi_{12} = 0.1$. This figure confirms that the above discussion is correct, though the surplus ratio does not monotonically decrease in terms of the number of contractors.

When small number of contractors exist ($1 \leq n \leq 15$), the winning probability at task t_2 is larger than that at task t_2, so all contractors choose the task t_2, though the probability of successfully completing the task t_2 becomes sufficiently large when $n = 7$, the amount of expected utilities if a contractor is allocated task t_1 is larger than if a contractor is allocated task t_2 in $8 \leq n \leq 15$, and the surplus ratio decreases. And in the following interval $15 \leq n \leq 50$, the surplus ratio slightly changes with vibrations. The difference between the two tasks become small if a new contractor added to the market, and the difference between the increment of surplus in the equilibrium and that in optimal allocation could vary in this interval. The reason why the vibrations occur is that the surplus ratio decreases if the increment of surplus in the equilibrium is relatively small and vice versa. In addition, if n is sufficiently large, the surplus ratio monotonically decreases because the optimal allocation does not change even if the number of contractors increases, and contractors continue to participate the task unless the expected utility is less than zero.

5.2 The probability of success

If the success probability of completing task t_1 (t_2) increases, the number of choosing t_1 does not decrease in the optimal allocation, while it does not increase in an equilibrium.

Suppose that ϕ_{11} changes to $\phi_{11}^{'}$ ($> \phi_{11}$). At the optimal allocation, the following expressions should be maximized.

$$\max\{v_1(1-(1-\phi_{11})^{n_{11}}) - n_{11}c_{11} \\ +v_2(1-(1-\phi_{12})^{n_{12}}) - n_{12}c_{12}\} \tag{2}$$

$$\max\{v_1(1-(1-\phi'_{11})^{n'_{11}}) - n'_{11}c_{11} \\ +v_2(1-(1-\phi_{12})^{n'_{12}}) - n'_{12}c_{12}\} \tag{3}$$

Here, assume that $n_1^{'}$ is larger than n_1. The increase of $1-(1-\phi_{11})^{n_{11}}$ is larger than that of $1-(1-\phi'_{11})^{n'_{11}}$, while the decrease of $n_{11}c_{11}$ is equal to that of $n_{11}^{'}c_{11}$ when the number of contractors carrying out task t_1 increases from n_{11} to $n_{11}^{'}$. The changes of terms related to task t_2 are also the same as each other because the values of the parameters are the same at expressions 2 and 3. Thus, if social surplus increases by increasing from n_{11} to $n_{11}^{'}$ for expression 3, social

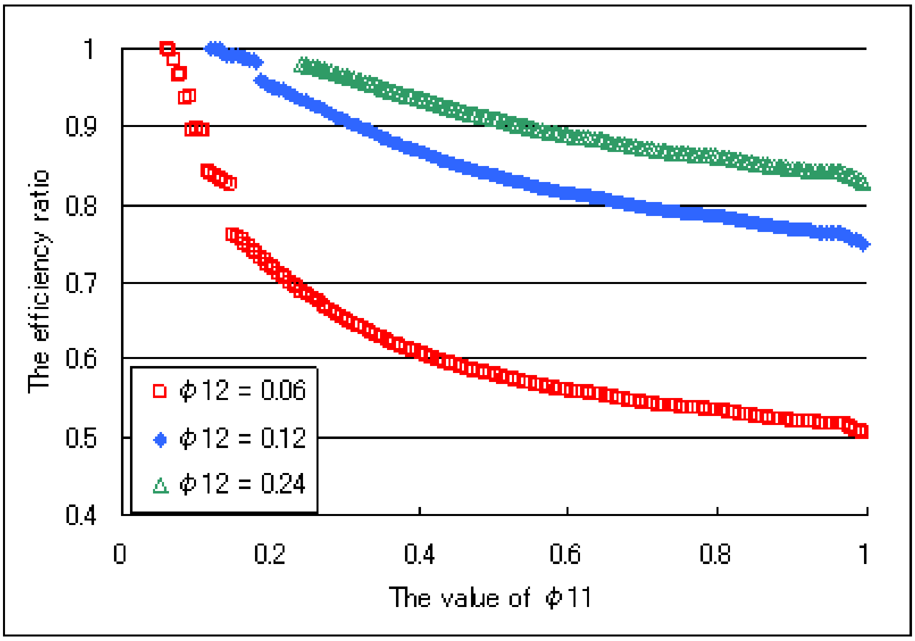


Figure 5: The relationship between the success probability of t_1 and the social efficiency.

surplus also increases by doing the same for expression 2. This contradicts with the assumption that expression 2 is maximized at (n_{11}, n_{12}). Therefore, n'_{11} is not larger than n_{11}.

On the other hand, the following equation is satisfied at an equilibrium.

$$v_1(1-(1-\phi_{11})^{n_1})/n_1 - c_{11} = v_2(1-(1-\phi_{12})^{n_2})/n_2 - c_{12}$$

If the success probability of completing task t_1 increases from ϕ_{11} to ϕ'_{11}, the left-hand side increases. If a contractor change his mind to choose t_2 instead of t_1, the difference between the expected utility by choosing t_1 and the expected utility by choosing t_2 increases, which means it cannot reach a Nash equilibrium. Thus, $n_{11}{}'$ is not smaller than n_{11}.

Figure 5 shows how the surplus ratio changes if the probability of success ϕ_{11} increases. The horizontal axis represents the probability of success ϕ_{11}, while the vertical axis represents the surplus ratio. Here, I set $v_1 = v_2 = 100$,

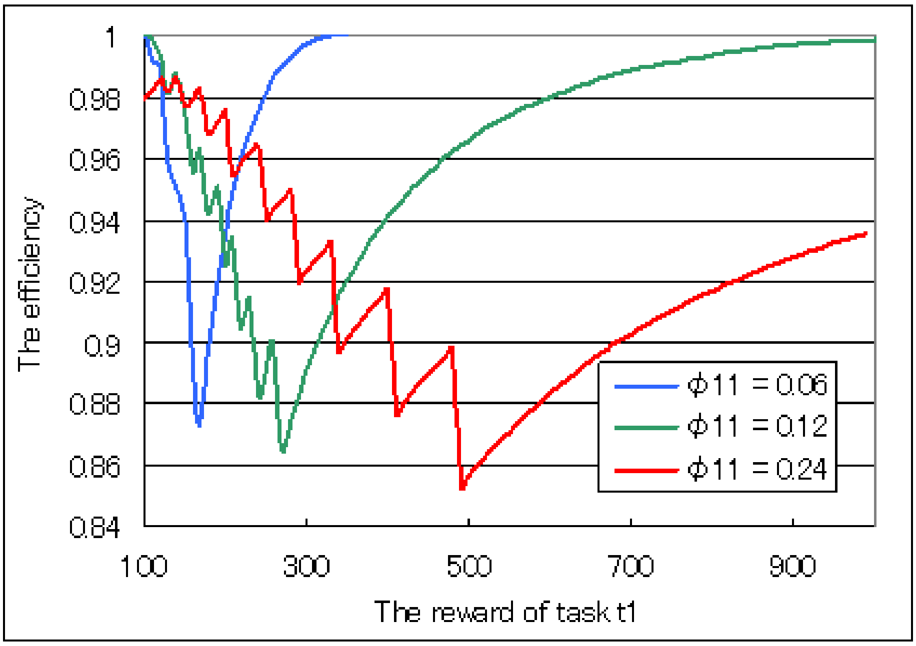


Figure 6: The relationship between the reward of t_1 and the social efficiency.

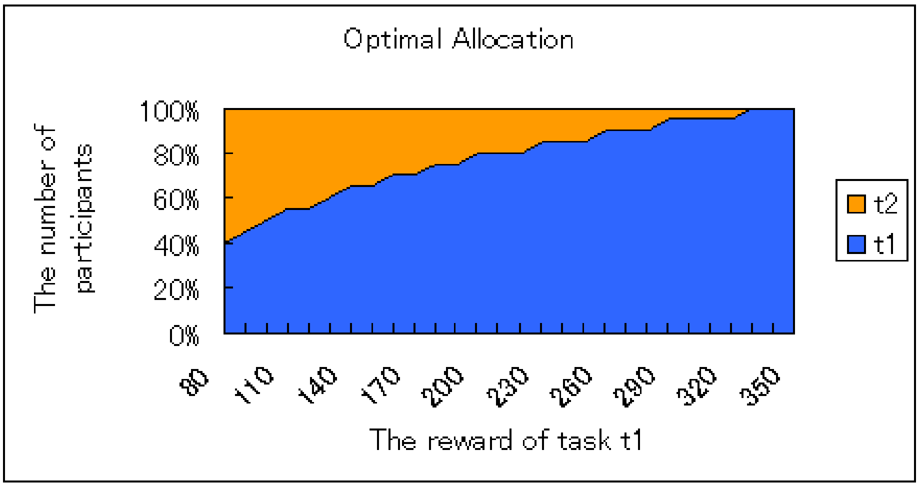
and $c_{11} = c_{12} = 3$. The three lines in the figure corresponds to the cases of $\phi_{12} = 0.06, 0.12, 0.24$. This figure confirms that the above discussion is correct.

5.3 An amount of reward

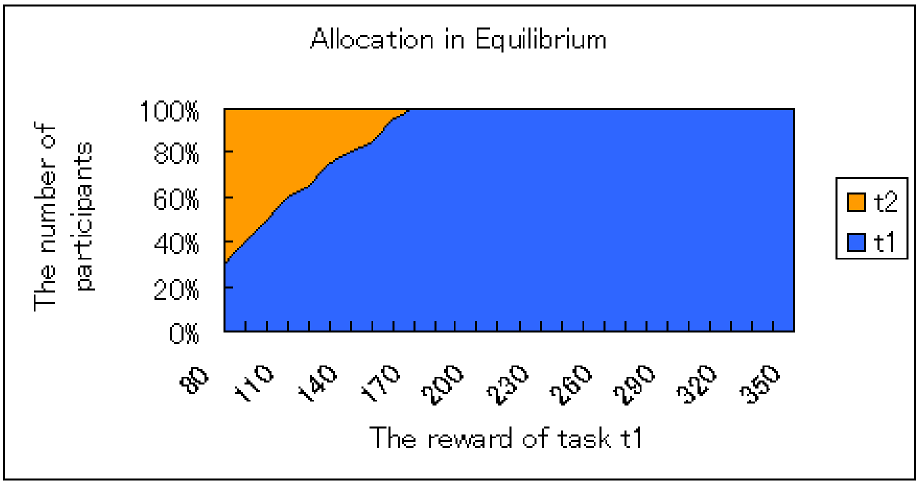
The increase of ϕ_{11} affects the behaviors at the optimal allocation and a Nash equilibrium differently, that is, at the optimal allocation n_{11} decreases, while at a Nash equilibrium n_{11} increases. Thus, as ϕ_{11} increases, the surplus ratio monotonically decreases.

On the other hand, as reward v_1 increases, the surplus ratio does not change monotonically.

Given reward v_1, the optimal allocation $(n_{11}^{opt}, n_{12}^{opt})$ and an allocation in a Nash equilibrium $(n_{11}^{Nash}, n_{12}^{Nash})$ can be obtained. If v_1 increases by Δv and the number of the allocation does not change, the expected social utility about task t_1 increases by $\Delta v(1-(1-\phi_1)^{n_1})$. If $n_1^{OPT} < n_1^{NASH}$, the surplus ratio is improved because the increment of social surplus at Nash equilibrium is larger



(a) the optimal allocation



(b) the allocation in equilibrium

Figure 7: The reward of t_1 and the number of participants for each of tasks.

than that at the optimal allocation. As the reward v_1 increases sufficiently, n_{11}^{opt} reaches n and n_{11}^{Nash} reaches n. If n_{11}^{opt} is smaller than n_{11}^{Nash}, the following sequence of $(n_{11}^{opt}, n_{12}^{opt}, n_{11}^{Nash}, n_{12}^{Nash})$, $(n_{11}^{opt}+1, n_{12}^{opt}-1, n_{11}^{Nash}, n_{12}^{Nash})$, $(n_{11}^{opt}+1, n_{12}^{opt}-1, n_{11}^{Nash}+1, n_{12}^{Nash}-1)$ is found. At a change from $(n_{11}^{opt}, n_{12}^{opt}, n_{11}^{Nash}, n_{12}^{Nash})$ to $(n_{11}^{opt}+1, n_{12}^{opt}-1, n_{11}^{Nash}, n_{12}^{Nash})$, the gap between the contractors' allocation in an optimal state and in a Nash equilibrium becomes large. This means the surplus ratio becomes worse. On the other hand, at a change from $(n_{11}^{opt}+1, n_{12}^{opt}-1, n_{11}^{Nash}, n_{12}^{Nash})$ to $(n_{11}^{opt}+1, n_{12}^{opt}-1, n_{11}^{Nash}+1, n_{12}^{Nash}-1)$,

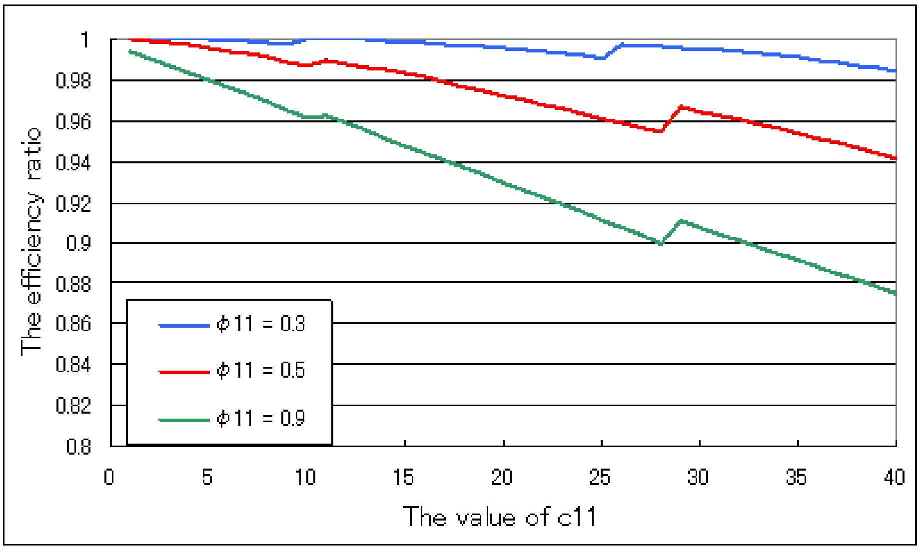


Figure 8: The relationship between the cost of t_1 and the social efficiency.

the gap between the contractors' allocation in an optimal state and in a Nash equilibrium becomes small. This means the surplus ratio is improved. Thus, the change of the surplus ratio is not monotonic.

Figure 6 shows how the surplus ratio changes if the reward increases in an example. The horizontal axis represents the reward t_1, while the vertical axis represents the surplus ratio. Another reward v_2 is fixed to 100. The other parameters are also fixed except t_1, that is, $c_{11} = c_{12} = 3$. The data are plotted for the three cases of $\phi_{11} = 0.06, 0.12, 0.24$. The difference between the rewards of t_1 and t_2 is not large, the surplus ratio decreases first, then the surplus ratio becomes one.

Figure 7 shows the contractors' allocation in the case of $\phi_{11} = 0.06$. The horizontal axis represents the reward for task t_1. The vertical axis represents the ratio of contractors carrying out task t_1. Figure 7 (a) shows the optimal task allocation, while Figure 7 (b) shows the allocation in a Nash equilibrium.

When the difference between the amounts of rewards is not large, the number of contractors choosing task t_1 rapidly increases in the case of a Nash equilibrium as the reward of task t_1 increases. All contractors choose task t_1 at $v_1 = 170$. On the other hand, the number of contractors choosing task t_1 gradually increases in the case of an optimal allocation as the reward of task t_1 increases. The value of reward v_1 where all contractors choose task t_1 is 330. Thus, it is confirmed that the difference of these profiles in task allocations causes social inefficiency.

5.4 Cost for completing a task

This case can be discussed in the same manner as the case of changing the rewards. Figure 8 shows how social efficiency changes as the cost of t_1 changes. The horizontal axis represents the cost of t_1, while the vertical axis represents the surplus ratio. Here, I set $v_1 = v_2 = 1000$, and $c_{12} = 1$. The three lines in the figure corresponds to the cases of $\phi_{11}(= \phi_{12}) = 0.3, 0.5, 0.9$. This figure confirms that the surplus ratio decreases if the cost of completing a task increases. There are two break points in each line. These correspond to the points that the number of contractors choosing t_1 changes.

Chapter 6 Multiple Types Contractor Case

In the previous chapter, I assume the case where all the contractors have the same type. From a practical standpoint, however, some contractors might be specialists, and their success probability or cost might be different from general contractors.

This chapter will consider the case that multiple types of contractors' case. Type sets of contractors are denoted by $\{\theta_i\}$, the success probabilities of task t_j are denoted by $\{\phi_{ij}\}$, and the costs of task t_j are denoted by $\{c_{ij}\}$. When the number of type θ_i contractors who carry out task t_j is equal to n_{ij}, the expected surplus V_j of task t_j is expressed as follows.

$$V_j = v_j(1 - \prod_{i=1}^{l}(1-\phi_{ij})^{n_{ij}}) - \sum_{i=1}^{l} n_{ij}c_{ij}$$

As the above expression shows, expected utilities of contractors depend on the success probability of contractors of other types, and the number of participants of the task. Thus, the expected utility $g_j^{\theta_i}$ of the contractors who have type θ_1 is expressed as follows.

$$g_j^{\theta_i} = v_j\sigma_{ij}(\{\phi_{ij}\}_i, \{n_{ij}\}_i) - c_{ij}$$

Here, let n_{ij} be the number of contractors of the type θ_i who carry out the task t_j. The first term of the g_{ij}'s expression represents the expected reward for attaining task t_j. $\sigma_i(\cdot)$ represents the winning probability of the contractors of type θ_i.

This chapter examines the case of two types and two tasks for the sake of simplicity. As mentioned former chapter, the pure Nash equilibrium always exists in this class of crowdsourcing games, and we can find it by the algorithm mentioned in that chapter. Types of contractors are denoted by θ_1, θ_2, the expected utility is denoted by $g_j^{\theta_i}(n_{1j}, n_{2j})$. And the term of $\sigma_{ij}(\cdot)$ is calculated as follows.

$$\begin{aligned}
&\sigma_{ij}(\phi_{1j}, \phi_{2j}, n_{1j}, n_{2j}) \\
&= \sum_{k=0}^{n_{1j}-1}\sum_{l=0}^{n_{2j}}[\frac{\{\phi_{1j}\cdot\binom{n_{1j}}{k}\phi_{1j}^k(1-\phi_{1j})^{(n_{1j}-1-k)}\cdot\binom{n_{2j}}{l}\phi_{2j}^l(1-\phi_{2j})^{n_{2j}-l}\}}{(k+l+1)}]
\end{aligned}$$

If the number of contractors who have type θ_1 and successfully complete the task, is $k+1$ and the number of contractors who have type θ_2 is l, the probability of attaining the task calculates as follows.

$$\phi_{1j} \cdot \binom{n_{1j}}{k} \phi_{1j}^{k} (1-\phi_{1j})^{(n_{1j}-1-k)} \cdot \binom{n_{2j}}{l} \phi_{2j}^{l} (1-\phi_{2j})^{n_{2j}-l} \tag{4}$$

The winning probability of contractors of type θ_1 is calculated by multiplying the probability of attaining the task by $1/(k+l+1)$ because I assume that the best contractor is elected at random if many contractors succeed the task. In $\sigma()$, the sum from (k,l) equals $(0,0)$ to $(n_{1j}-1, n_{2j})$ of the winning probability is calculated. The similar calculation is possible for type θ_2's contractors.

Next, I will illustrate the conditions of Nash equilibrium. Given the number of contractors of type θ_i is n_i, the conditions of Nash equilibrium where the allocation (n_{11}, n_{12}, n_{21}, n_{22}) which satisfies n_{11}+n_{12}=n_1 and n_{21}+n_{22}=n_2 are defined as follows.

$$v_1\sigma_{11}(\phi_{11}, \phi_{21}, n_{11}, n_{21}) - c_{11} \geq$$
$$v_2\sigma_{12}(\phi_{12}, \phi_{22}, n_{12}+1, n_{22}) - c_{12}$$

and

$$v_2\sigma_{12}(\phi_{12}, \phi_{22}, n_{12}, n_{22}) - c_{12} \geq$$
$$v_1\sigma_{11}(\phi_{11}, \phi_{21}, n_{11}+1, n_{21}) - c_{11}$$

and

$$v_1\sigma_{21}(\phi_{21}, \phi_{11}, n_{21}, n_{11}) - c_{21} \geq$$
$$v_2\sigma_{22}(\phi_{22}, \phi_{12}, n_{22}+1, n_{12}) - c_{22}$$

and

$$v_2\sigma_{22}(\phi_{22}, \phi_{12}, n_{22}, n_{12}) - c_{22} \geq$$
$$v_1\sigma_{21}(\phi_{21}, \phi_{11}, n_{21}+1, n_{11}) - c_{21}$$

It is computationally hard to calculate the expected utility, when n_1 or n_2 become large. Here, I experimentally calculate the Nash equilibrium in some cases, and compare the social surplus in Nash equilibrium with the optimal one.

6.1 Sufficient large number of contractors

In this section, I analyze the case when the number of contractors is sufficiently large. The rational contactors must consider the strategies of other contractors who participates after his/her decision making, so he/she must consider not only his/her own expected utility but also the expected utilities of other types' contractors.

The finding process of the pure Nash equilibrium about type θ_1 is as follows. First, we seek the number of participant of type θ_2's contractors at equilibrium in the case of $n_{1j} = 0$ which is denoted by $(0, n_{2j}^{NASH(n_{1j}=0)})$. If $g_j^{\theta_1}(1, n_{2j}^{NASH(n_{1j}=0)}) < 0$, the allocation $(0, n_{2j}^{NASH(n_{1j}=0)})$ is the pure Nash equilibrium, that is, it is rational for the type θ_1's contractors not to participate the task t_1. By contraries, if $g_j^{\theta_1}(1, n_{2j}^{NASH(n_{1j}=0)}) \geq 0$, then we calculate $n_{2j}^{NASH(n_{1j}=1)}$, and evaluate $g_j^{\theta_1}(2, n_{2j}^{NASH(n_{1j}=1)})$ in a similar way. we can obtain all possible pure Nash equilibria by repeating the process if $n_{2j}^{NASH(n_{1j}=k)} > 0$. The similar process works about type θ_2.

I give an example. Now, I consider the case that $v_1 = 100$, $\phi_{11} = 0.2, \phi_{21} = 0.1$, $c_{11} = 12$, $c_{12} = 6$. Type θ_1's contractors have high probability of success but high incurred cost, while type θ_2's contractors have low probability of success but low incurred cost. In this case, there is not sufficient difference among these two types about the efficiency. Actually, this case has following 7 pure Nash equilibrium, $(6,0)$, $(5,1)$, $(4,3)$, $(3,5)$, $(2,7)$, $(1,9)$, $(0,11)$.

On the other hand, If I give the other example such that $v_1 = 100$, $\phi_{11} = \phi_{21} = 0.01$, $c_{11} = 6$, $c_{12} = 9$, the result differs. It is obviously efficient to allocate the type θ_1's contractors to the task. The allocation $(11, 0)$ is the unique pure Nash equilibrium in this case, indicating that the crowdsourcing mechanism successfully select the efficient type of contractors.

6.2 Specialists of tasks

I will analyze the case where specialists of tasks exist. I assume that specialists have the high success probability of specific tasks, and examine the case where contractors of type θ_i are specialists of task t_i.

Figure 9 shows the case where the settings are $v_1 = v_2 = 100$, $c_{11} = c_{22} = 3$,

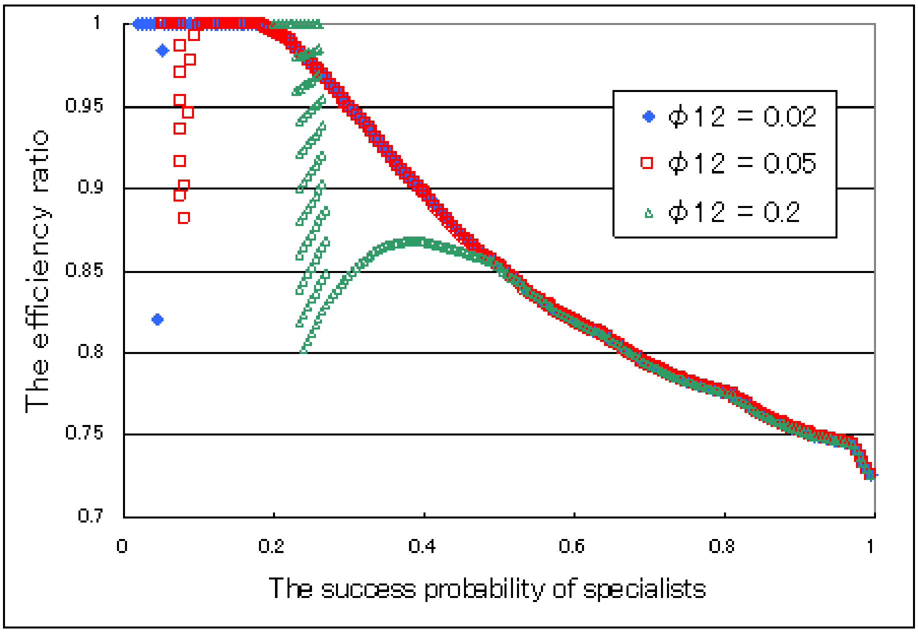


Figure 9: The success probability of specialists and the social efficiency.

$c_{12} = c_{21} = 1$. The data are plotted for the three cases of $\phi_{12}(= \phi_{21}) = 0.02, 0.05, 0.2$.

As mentioned in the previous section, when the success probability of tasks become high, the expected utility of contractors increases. Thus, in this situation, the number of allocation to specialists increases. On the other hand, the expected surplus of the task with high success probability becomes enough large, even if the number of allocation is small. Therefore, in the optimal allocation, the number of allocation to specialists is relatively small. As a result, inefficient allocations arise when the success probability of specialists is enough high. I also found that these problems can arise regardless of the cost of contractors' type.

Meanwhile, I discovered that the surplus ratio decreases rapidly in specific range in the situation that the cost of specialists is larger than the cost of non-specialists, like $c_{11} = c_{22} = 3$, $c_{12} = c_{21} = 1$.

To understand this problem, I analyze this case in detail. Table 3 shows the

Table 3: Inefficiency of equilibrium.

$(\phi_{11}, \phi_{12}) = (0.075, 0.05)$					
	n_{11}	n_{12}	n_{21}	n_{22}	ER
EA	0	10	10	0	1.0
OA	0	10	10	0	-

$(\phi_{11}, \phi_{12}) = (0.08, 0.05)$					
	n_{11}	n_{12}	n_{21}	n_{22}	ER
EA	10	0	0	10	0.88
OA	0	10	10	0	-

$(\phi_{11}, \phi_{12}) = (0.11, 0.05)$					
	n_{11}	n_{12}	n_{21}	n_{22}	R
EA	10	0	0	10	1.00
OA	10	0	0	10	-

allocation of the case, and two cases when ϕ_{11} is changed slightly. EA means the allocation in equilibria and OA means the optimal allocation. In the case where $\phi_{11} = 0.08$, a reversal phenomenon arises. Indeed, although special tasks are assigned to the specialists in Nash equilibrium, in the optimal allocation, they are assigned to the non-specialists.

Chapter 7 The Price of Anarchy

This chapter will examine the social efficiency in the worst case equilibrium. If more than one equilibrium can exist, which equilibrium is realized in the real-world market is unknown. Thus, it is reliable for the market operator to guarantee the efficiency in the worst case equilibrium.

The price of anarchy, first introduced by Koutsoupias, et al.[11], is utilized to evaluate what extent of efficiency is realized at the worst case equilibrium. This paper evaluates the surplus ratio of crowdsourcing as the price of anarchy.

Definition 2. *If the surplus ratio at the worst case equilibrium is denoted by R_-, the price of anarchy of crowdsourcing game is defined as $PoA = \frac{1}{R_-}$.*

Theorem 7. *The price of anarchy of crowdsourcing game approaches infinity.*

Proof. From the past discussion, it is obvious for general crowdsourcing games that the price of anarchy approaches infinity if the number of contractors is sufficiently large. □

This result suggests that it is important for the market operator to restrict the number of contractors adequately for the persistent operation of crowdsourcing. However, if the properties of tasks in the market vary, adequate number of contractors for each task might vary as well. It is desirable to find the efficient method for restricting the number of contractors according to the property of tasks.

First, I assume concrete cases of two tasks and no incurred cost with single type contractors. I can find the strict price of anarchy from the discussion of previous chapter.

Theorem 8. *In crowdsourcing games with single type of contractors, two tasks, and no incurred cost, the price of anarchy is $\frac{2n-1}{n}$, and this bound is tight.*

Proof. In the two tasks case, the surplus ratio decreases as the asymmetry of allocation becomes large. First, I suppose the situation in which all n contractors carry out the task t_2 at equilibrium, but all of them are allocated to the task t_1 in optimum. However, the situation is impossible. When I consider the situation from the viewpoint of equilibrium, the surplus if one contractor carry out the task t_2 is always larger than that if one contractor carry out the task t_1. On the other hand, if I consider the situation from the viewpoint of optimum, the social surplus becomes larger if the number of participants for the task t_1 increases from $n-1$ to n than if the number of participants for the task t_1 increases from zero to one. This situation contradicts the Lemma 2.

Therefore, the worst situation is that all n contractors carry out the task t_2 at equilibrium, but $n-1$ contractors are allocated to the task t_1, and only one contractor is allocated in optimum.

The surplus ratio R of the situation is expressed as follows.

$$R = \frac{v_2(1-(1-\phi_2)^n)}{v_1(1-(1-\phi_1)^{n-1}) + v_2\phi_2}$$

Suppose that $n \geq 2$. And I utilize the expression $v_2(1-(1-\phi_2)^n) = v_2(\phi_2 + \epsilon), \epsilon > 0$.

From the equilibrium condition,

$$v_1\phi_1 < \frac{v_2(\phi_2+\epsilon)}{n} < \frac{v_2\phi_2}{n}$$

So, $v_1(1-(1-\phi_1)^{n-1})$ is deformed as follows.

$$\begin{aligned} v_1(1-(1-\phi_1)^{n-1}) &< v_2\phi_2 \frac{(1-(1-\phi_1)^{n-1})}{n\phi_1} \\ &\leq v_2\phi_2 \lim_{\phi_1 \to +0} \frac{(1-(1-\phi_1)^{n-1})}{n\phi_1} \\ &= v_2\phi_2 \lim_{\phi_1 \to +0} \frac{\frac{d}{d\phi_1}(1-(1-\phi_1)^{n-1})}{\frac{d}{d\phi_1} n\phi_1} \\ &= v_2\phi_2 \frac{n-1}{n} \end{aligned}$$

Consequently,

$$\begin{aligned} R &< \frac{v_2(\phi_2+\epsilon)}{\frac{n-1}{n} v_2\phi_2 + v_2\phi_2} \\ &< \frac{v_2\phi_2}{\frac{(n-1)v_2\phi_2 + nv_2\phi_2}{n}} \\ &= \frac{n}{2n-1} \end{aligned}$$

The result shows the upper bound of the price of anarchy is $\frac{2n-1}{n}$.

In addition, the surplus ratio approaches $\frac{n}{2n-1}$ if $\phi_1 \to 0$ and $\phi_2 \to 1$.

Table 4 shows the change of the efficiency ratio in terms of ϕ_1 and ϕ_2. The number of contractors is fixed at $n = 20$, and the reward of task t_1 is also fixed at $v_1 = 10$. The value v_2 in the table indicates reward of task t_2 at the worst case equilibrium. From the above result, the theoretical lower bound of the surplus ratio is calculated as $20/(40-1) = 0.512821$. Table 4 indicates that the surplus ratio converges the theoretical lower bound when $\phi_1 \to 0$ and $\phi_2 \to 1$.

Table 4: The change of the efficiency ratio

ϕ_1	ϕ_2	v_1	v_2	R
0.01	0.99	10	50	0.537878
0.001	0.999	10	499	0.515832
0.0001	0.9999	10	4999	0.513122
0.00001	0.99999	10	49999	0.512851
0.000001	0.999999	10	499999	0.512824
			lower bound	0.512821

These results show that this bound is tight. □

7.1 The class of valid game

Valid games, introduced by Vetta[18], is a set of non-cooperative games for which Vetta proved that the upper bound of the price of anarchy is two. In this section, I prove that the restricted version of crowdsourcing is included in the class of valid games. This result guarantees that the upper bound of the price of anarchy is two. First, I define some mathematical concepts in order to discuss the result.

Definition 3. *A set function $f : 2^{\mathcal{H}} \to \mathbb{R}$ is submodular function if $f(\emptyset) = 0$ and for any set $A, B \subseteq \mathcal{H}$, the following inequality is satisfied.*

$$f(A) + f(B) \geq f(A \cap B) + f(A \cup B)$$

And, A set function f is non-decreasing function if $f(X) \leq f(Y)$ for any *set* $X \subseteqq Y \subseteqq \mathcal{H}$.

Given the strategy profile $\mathcal{S} = (S_1, ..., S_n)$, the set $\mathcal{H}_\mathcal{S} = \{(k,i) : 1 \geq k \geq n, i \in S_k\}$ is called the pair set of $\mathcal{S}$. In addition, given the function $f : \prod_k S_k \rightarrow \mathbb{R}$, the corresponding set function is $f^\mathcal{S} : 2^{\mathcal{H}_\mathcal{S}} \rightarrow \mathbb{R}$.

Definition 4. $\mathcal{G}(N, \{S_i\}, \{g_i(\prod_k S_k)\}$ *denotes a non-cooperative game which has social utility function* $\gamma : \prod_k S_k \rightarrow \mathbb{R}$. $\mathcal{G}$ *is a valid game if following conditions are satisfied.*

- *The set function* $\gamma^\mathcal{S}$ *corresponding* γ *is submodular and non-decreasing function.*
- *The payoff of a player is at least equal to the difference in the social function when the player participates versus when it does not participate.*
- *The sum of the utility or payoff functions for any strategies should be less than or equal to the social function.*

If the player i's utility is denoted by $g_i(s)$, the second condition of valid games is equivalent to the following condition.

$$\forall s, i, g_i(s) - g_i(\emptyset, s_{-i}) \geq \gamma(s) - \gamma(\emptyset, s_{-i})$$

The expression means the increment of individual utilities is always equal or more then that of the social utility. And the third condition of valid games is equivalent to the following condition.

$$\forall s, \sum_i g_i(s) \leq \gamma(s)$$

This means that the social surplus in the game system is distributable. Note, we do not require that $\sum_i g_i(s) = \gamma(s)$. In fact, we may view $\gamma(s) - \sum_i g_i(s)$ as the utility of some non-agent, say the market operator.

The following property is known and useful to discuss the price of anarchy.

Theorem 9. *Let* $\mathcal{G}$ *be a valid game, then for any mixed strategy Nash equilibrium, the social function at this equilibrium is at least half the optimum social function, i.e., in a Nash equilibrium a,* $\gamma(OPT) \leq 2\gamma(a)$*.([18])*

Here, I introduce a restricted version of crowdsourcing games.

Definition 5. *The crowdsourcing game is additive if the optimal allocation of any task* t_j *is realized when all contractors carry out the task.*

Lemma 3. *A crowdsourcing game with single type of contractors, two tasks, and no incurred cost is a additive crowdsourcing game.*

In additive crowdsourcing games, the social surplus is improved whichever tasks contractors choose compared to the case that contractors do not choose any task. In addition, the individual rationality holds, that is, the expected utility is always more than zero. This means that a task selection is important in maximizing its utility.

Theorem 10. *An additive crowdsourcing game is a valid game.*

Proof.

- The social function in crowdsourcing game is defined as follows.

$$\gamma(s) = \sum_{j=1}^{l}\{v_j(1 - \prod_{k:s_k=t_j}(1-\phi_{typeof(A_k)j})^{n_{typeof(A_k)j}(s)})) - n_{typeof(A_k)j}(s)c_{typeof(A_k)j}\}$$

From the definition of additive crowdsourcing game, the following condition holds for any strategy taple s.

$$\gamma(s_k, s_{-k}) \geq \gamma(\emptyset, s_{-k}), \forall s_k$$

Thus, $\gamma^{\mathcal{S}}$ is obviously non-decreasing function. And I utilize another equivalent definition of submodularity, that is, f is submodular function if $f(A \cap \{i\}) - f(A) \geq f(B \cap \{i\}) - f(B)$ for any $A, B, A \subset B$ and for any $i \notin B$. To prove the submodularity of $\gamma^{\mathcal{A}}$, it is sufficient to show that if any strategy tuples s and s' such that $s_1 = \emptyset$ and either $s_k = s_k^{'}$ or $s_k = \emptyset$ for any k exist, then $\gamma(t_j, s_{-1}) - \gamma(s) \geq \gamma(t_j, s_{-1}^{'}) - \gamma(s'), \forall t_j$. To show that this condition is satisfied, let $n_{ij}(s)$ be the number of type θ_i's contractors who carry out the task t_j. It is apparent that $n_{ij}(s) \geq n_{ij}(s')$ because of the settings of s and s', and the following inequality holds.

$$1 - \prod_{k:s_k=t_j}(1 - \phi_{typeof(A_k)j}) \geq 1 - \prod_{k:s_k^{'}=t_j}(1 - \phi_{typeof(A_k)j})$$

The following two equations

$$\gamma(t_j, s_{-1}) - \gamma(s) = \phi_{1j}\{1 - \prod_{k:s_k=t_j}(1 - \phi_{typeof(A_k)j}\} - c_{1j}$$

$$\gamma(t_j, s_{-1}^{'}) - \gamma(s') = \phi_{1j}\{1 - \prod_{k:s_k^{'}=t_j}(1 - \phi_{typeof(A_k)j}\} - c_{1j}$$

show that the condition is satisfied. Consequently, $\gamma^{\mathcal{S}}$ is a submodular function. Note that $\gamma^{\mathcal{S}}$ is a submodular function even if the crowdsourcing game is not additive. As indicated above, the first condition of valid games is satisfied.

- Next, I prove that the individual expected utility is always more than the increment of social surplus. This paper defines the social surplus as the sum of the expected utilities of contractors, and the following equations obviously hold.

$$\gamma(s) = \sum_{A_k} g_k(s)$$
$$\gamma(\emptyset, s_{-i}) = \sum_{A_k:k\neq i} g_k(\emptyset, s_{-i})$$

  It is sufficent to show that $g_i(s) \geq \sum_{A_k} g_k(s) - \sum_{A_k:k\neq i} g_k(\emptyset, s_{-i})$ because $g_i(\emptyset, s_{-i}) = 0$.

$$\begin{aligned} g_i(s) &\geq \sum_{A_k} g_k(s) - \sum_{A_k:k\neq i} g_k(\emptyset, s_{-i}) \\ \sum_{A_k:k\neq i} g_k(\emptyset, s_{-i}) &\geq \sum_{A_k} g_k(s) - g_i(s) \\ \sum_{A_k:k\neq i} g_k(\emptyset, s_{-i}) &\geq \sum_{A_k:k\neq i} g_k(s) \end{aligned}$$

  It is also sufficient to show that the condition $g_k(\emptyset, s_{-i}) \geq g_k(s), \forall k, k \neq i$ is satisfied. This inequality means that the expected utility of the contractor who already participate in the task t_j always decreases if a new contractor participates in the task.

  Recall that the expected utility is denoted by $g_k(s) = v_j\sigma_{ij}(s) - c_{ij}$. Only $\sigma_{ij}(s)$ changes when a new contractor participates. Recall again that $\sigma_{ij}(s)$ means the winning probability about the task. The probability always decreases by a new contractor because of the competition, that is, $\sigma_{ij}(\emptyset, s_{-k}) > \sigma_{ij}(s), \forall k : s_k = j$. As indicated above, the second condition of valid games is satisfied.

- As indicated previous discussion, it is obvious that the third condition of valid games is also satisfied.

To sum it up, an additive crowdsourcing game is a valid game. □

Corollary 4. *The upper bound of the price of anarchy of additive crowdsourcing games is two.*

By using this additive condition, we can design a efficient restricting mechanism about the number of contractors. The sketch of the algorithm is shown as follows. It is assumed that contractors with single type exist.

1. First, we calculate the individual expected utility $g_i(n_j^{OPT})$ for each task t_j if the number of contractors who carry out the task is optimum n_j^{OPT}.
2. We can restrict the number of contractors for each task to n_j such that n_j satisfies following conditions.

$$g_i(n_j) > \min_j g_i(n_j^{OPT})$$

$$g_i(n_j + 1) < \min_j g_i(n_j^{OPT})$$

In this mechanism, for any pure Nash equilibrium s, $\gamma(t_j, s_{-k}) \geq \gamma(\emptyset, s_{-k})$ is always satisfied for any task t_j and any contractor A_k. Therefore, the crowdsourcing game with above mechanism is considered as one of additive crowdsourcing games, and has the upper bound of the price of anarchy which is two.

Chapter 8 Conclusion

This paper considered the crowdsourcing, which is one of new task allocation systems through the World Wide Web, and analyzed the social efficiency. In the crowdsourcing market, contractees announce their tasks, and then each of contractors chooses a task and reports the result if he/she completed the task. The contractee receiving reports from the contractors selects the best solution, and then pays reward to the contractor who reported the best solution. However, the efficiency of this mechanism was questionable if contractors were interested in their own utility. It might be inefficient from the viewpoint of task allocation mechanisms because many contractors could gather the specific task. In addition, there was not enough accumulation of knowledge about the operation and the management of the crowdsourcing market because of the novelty. This paper focused on the strategic behavior of individual contractors in the crowdsourcing market, and formulated it as a multiagent market model. To analyze the social efficiency of crowdsourcing mechanism, this paper tackled the following two issues.

Modeling of the task allocation in the crowdsourcing market In many researches of resource allocation, it is assumed that each agent strategically reports its information, and discuss the determination mechanism of the allocation calculated by the reported information. Meanwhile in crowdsourcing, there is no top-down allocation because contractors autonomously select the task and carry it out. Therefore, we need to build a new model focusing on the strategic task selection.

Analysis of the social efficiency as a market In crowdsourcing market, the probability of successfully completing tasks becomes high and the social surplus is expected to be improved because tasks are carried out in parallel by contractors. However, the social inefficiency is also conceivable because too many contractors might carry out the specific task. we need to know the impact of the asymmetry of tasks and the asymmetry of contractors on the efficiency as task allocation mechanism.

To solve the above issues, this paper formalized the strategic task selection of contractors in crowdsourcing as the form of the game theory. I assumed that contractors in the market chose the task considering the trade-off between the expected reward for carrying out the task and the incurred cost, and considered that it was an analogical decision making of the road selection in traffic network. This paper built the crowdsourcing game model based on congestion game model which was frequently utilized in equilibrium analyses of the traffic network. To understand what extent of efficiency realized by the task allocation at Nash equilibrium compared to the top-down optimal allocation, this paper defined the surplus ratio as a indicator of the social efficiency.

This paper approached the social efficiency analyses through two methods: one was the analysis of average impact of the asymmetry of tasks or contractors, and the other was that in the worst case of crowdsourcing models.

If the pure Nash equilibrium exists, it is sufficient as the efficiency analysis to evaluate for each combination of task allocations. Thus, this paper first examined the existence of pure Nash equilibrium. I proved that the pure Nash equilibrium of the crowdsourcing game with single type of contractors always exists. However, I also found that crowdsourcing game models belong to the more complicated class of congestion games, and indicated that the pure Nash equilibrium did not always exist. From those results, this paper analyzed the impact of the number of contractors, the number of types of them, the probability of success and the reward about tasks, and so forth, on the surplus ratio about the class having pure Nash equilibrium. As a result, I found that the social surplus at Nash equilibrium approached to zero when sufficient number of contractors exist, and that the difference between the probability of success of two tasks increased, the surplus ratio monotonically decreased because of the congestion in the task with higher probability of success. In the discussion of the reward, I found that the fractional difference between rewards brought inefficiency if two tasks had the same probability of success and incurred cost. Besides, I discovered that, when specialists having higher success probabilities in a particular task are present, a reversal phenomenon might arise. Indeed, in that case, although special tasks are assigned to the specialists in Nash equilibrium, in the optimal allocation, they are assigned to the non-specialists.

On the other hand, this paper also mentioned the price of anarchy of crowdsourcing games, which is determined by the surplus ratio in the worst case. From the previous discussion, it became apparent that the price of anarchy of general crowdsourcing game approached to infinity according to the increment of the number of contractors. This result suggested that it was important for the market operator to restrict the number of contractors adequately for the persistent operation of crowdsourcing. To understand the desirable method for restricting the number of contractors, we studied the price of anarchy in terms of the number of contractors, and discovered that the idea of valid games was applicable. This paper defined the additive class of crowdsourcing games, and proved that the upper bound of the price of anarchy in additive crowdsourcing games was two because of the submodularity and non-decreasing property of the social utility function. From these results, this paper suggested a mechanism restricting the number of contractors for each task.

The above contributions gave us a useful guideline on building a new problem solving market by crowdsourcing, and gave us a basic knowledge for designing an efficient task allocation mechanism in the future.

Acknowledgments

The author would like to express his gratitude to the supervisor, Associate Professor Shigeo Matsubara at Kyoto University, for his consecutive instruction, pertinent advice and earnest argument. The author would like to tender his acknowledgments to Professor Toru Ishida, Assistant Professor Hiromitsu Hattori at Kyoto University, for his technical and constructive advice. The author would like to express his appreciations to the advisers, Professor Tetsuro Sakai at Kyoto University, Associate Professor Keishi Tajima at Kyoto University for his valuable advice. Finally, the author would like to thank all members of Ishida and Matsubara laboratory for their various supports and discussions.

References

[1] Howe, J.: The rise of crowdsourcing, *Wired Magazine*, Vol. 14, No. 6, pp. 1–4 (2006).

[2] Gale, D. and Shapley, L.: College admissions and the stability of marriage, *American Mathematical Monthly*, pp. 9–15 (1962).

[3] Rosenthal, R.: A class of games possessing pure-strategy Nash equilibria, *International Journal of Game Theory*, Vol. 2, No. 1, pp. 65–67 (1973).

[4] Fotakis, D., Kontogiannis, S., Koutsoupias, E., Mavronicolas, M. and Spirakis, P.: The structure and complexity of Nash equilibria for a selfish routing game, *Theoretical Computer Science*, Vol. 410, No. 36, pp. 3305–3326 (2009).

[5] Goemans, M., Li, L., Mirrokni, V. and Thottan, M.: Market sharing games applied to content distribution in ad-hoc networks, *Proceedings of the 5th ACM international symposium on Mobile ad hoc networking and computing*, ACM, p. 66 (2004).

[6] DiPalantino, D. and Vojnovic, M.: Crowdsourcing and all-pay auctions, *Proceedings of the tenth ACM conference on Electronic commerce*, ACM, pp. 119–128 (2009).

[7] Zhao, B., Kubiatowicz, J. and Joseph, A.: Tapestry: An infrastructure for fault-tolerant wide-area location and routing, *Computer*, Vol. 74 (2001).

[8] Porter, R., Ronen, A., Shoham, Y. and Tennenholtz, M.: Fault tolerant mechanism design, *Artificial Intelligence*, Vol. 172, No. 15, pp. 1783–1799 (2008).

[9] Arrow, K.: Economic welfare and the allocation of resources for invention (1962).

[10] Barzel, Y.: Optimal timing of innovations, *The Review of Economics and Statistics*, pp. 348–355 (1968).

[11] Koutsoupias, E. and Papadimitriou, C.: Worst-case equilibria, *Computer Science Review*, Vol. 3, No. 2, pp. 65–69 (1999).

[12] Wardrop, J.: ROAD PAPER. SOME THEORETICAL ASPECTS OF ROAD TRAFFIC RESEARCH., *ICE Proceedings, Engineering Divisions*,

Vol. 1, No. 3, Thomas Telford, pp. 325–362 (1952).

[13] Roughgarden, T. and Tardos, É.: How bad is selfish routing?, *Journal of the ACM (JACM)*, Vol. 49, No. 2, pp. 236–259 (2002).

[14] Gairing, M., Lücking, T., Mavronicolas, M., Monien, B. and Rode, M.: Nash equilibria in discrete routing games with convex latency functions, *Journal of Computer and System Sciences*, Vol. 74, No. 7, pp. 1199–1225 (2008).

[15] Lücking, T., Mavronicolas, M., Monien, B. and Rode, M.: A new model for selfish routing, *Theoretical Computer Science*, Vol. 406, No. 3, pp. 187–206 (2008).

[16] Monderer, D. and Shapley, L.: Potential games, *Games and economic behavior*, Vol. 14, No. 1, pp. 124–143 (1996).

[17] Milchtaich, I.: Congestion games with player-specific payoff functions, *Games and economic behavior*, Vol. 13, No. 1, pp. 111–124 (1996).

[18] Vetta, A.: Nash equilibria in competitive societies, with applications to facility location, traffic routing and auctions.