社団法人　情報処理学会　研究報告
IPSJ SIG Technical Reports

2005－ＩＣＳ－139（16）
2005／3／15

参加者の匿名性を考慮した CPUリソースオークションシステムの提案

松本　崇志†　　川村　秀憲†　　大内　　東†

† 北海道大学大学院情報科学研究科　〒060-0814 北海道札幌市北区北14条西9丁目
E-mail: †{matumo,kawamura,ohuchi}@complex.eng.hokudai.ac.jp

あらまし　企業間でCPUリソースの売買を行うためのオークションシステムの提案を行った．本提案手法では，公開鍵暗号，ハッシュ関数，秘密分散法などの暗号技術を組み合わせることにより入札を匿名で行い，マッチングが成立した入札に関しては，オークショナーのみがその入札者を特定することができ，オークショナーと参加者の過半数が共謀しない限り不正に入札者を特定することができないシステムを実現した．また，入札データの送信法，及び取引方法においても参加者の匿名性を考慮した．最後に提案手法を構築し，それを用いた実機実験により提案手法が実システムの中で動作すること，及び参加者が売買を行うことで全体のCPUリソースを無駄なく使用していることを示した．

キーワード　匿名オークション，リソース売買，秘密分散法，暗号技術

Proposal of CPU Resource Auction System with an Anonymous Protocol

Takashi MATSUMOTO†, Hidenori KAWAMURA†, and Azuma OHUCHI†

† Graduate School of Information Science and Technology, Hokkaido University　Kita 14 Nishi 9, Kita-ku, Sapporo, 060-0814 Japan
E-mail: †{matumo,kawamura,ohuchi}@complex.eng.hokudai.ac.jp

Abstract　In this paper, we propose the auction system protocol which trading of CPU usage cycles among companies. This proposal system realize that anyone cannot identify bidder illegally unless the auctioneer and participants majority act in collusion, by using cryptographic technologies such like public key encryption, hash function and Secret Sharing Scheme. Moreover, this system is considered anonymous of bidder regarding the trading method and transmission method of bid. In addition, we developed an auction system with the proposal technique, and verified the proposal technique runs as a system and the effect of CPU resource trading.

Key words　anonymous auction, trading of CPU usage cycles, Secret Sharing Scheme, cryptographic technology

1. はじめに

近年，企業がCPUリソースを顧客の必要に応じて有料で貸し出すサービスや，株式や商品取引と同様にCPUリソースをオンラインで取引するサービスが発表されている[1]．企業間でCPUリソースの売買ができれば，参加企業が所有しているCPUリソースを用いるため設備コストが削減できること，参加者はCPUリソースの使用頻度が低い時間を用いて販売することにより利益を得ることができ，またCPUリソースを購入することにより安価で大きな処理能力を得ることができるといった利点があげられる．しかし問題点として，必要なCPUリソース量や余っているCPUリソース量を他社に知られることや取引相手に処理内容を推測される可能性が企業秘密に繋がる恐れがあること，また取引を行うとき，取引相手がライバル企業であることから生じる取引の弊害などがあげられる．企業間でのCPUリソースの売買を考えた場合，これらの問題を解決する必要がある．

そこで本研究では，企業間でCPUリソースの売買をオンラインで行うためのオークションシステムの提案，及び構築を行う．また，複数のマシンを用い実機実験を行うことにより提案手法が実システムの中で動作するか及び，売買を行うことによる参加者のCPUリソースの使用状態の考察を行う．

CPUリソースのように，各企業によってその時々の価値が変化する財に対する売買法は一般的に，株などの売買に用い

表 1　入札に必要な情報

$type_i$	入札の種類
$price_i$	入札価格
$block_i$	CPU リソース量
ID_i	参加者識別情報

られるコンティニアスダブルオークション方式が多く用いられる．電子オークションは現在広く普及しているが，オークション主催者（以下，オークショナー）が常に信頼できるとは限らず，オークショナーによる個人情報の流出などが後を絶たない．通常のオークションでは，落札者は発表されるが，落札されなかった入札者は発表されない．しかし，オークショナーは落札できなかった入札を含め，全ての入札がわかる．そのためオークショナーは，参加者がいつ，何に対し，どれだけの価値観で欲しがっているということがわかる．そこでこの問題を解決するため，匿名入札を用いたオークションシステムがいくつか提案されている [2] [3]．しかし，そのほとんどがシングルオークションに対するものであり，ダブルオークション方式のものではオークショナーとしてふたつの異なる組織を必要とし，そのふたつのオークショナーが共謀を行うと匿名性が壊れるなどの欠点を抱えたものが多い [4]．本稿では，ひとつのオークショナーと複数の異なる参加企業からなる，匿名性の高い CPU リソースコンティニアスダブルオークションシステムを提案する．

2. システム要件

本稿で提案するオークションシステムは，ひとつのオークショナーと複数の異なる参加企業からなる参加者 C_i $(i:1,2,\cdots,n)$ で構成される，オンラインで行うコンティニアスダブルオークションである．また，オークションの入札を行う際に必要な情報を，表 1 に示す．参加者は入札を行う際，入札の種類（売り入札，買い入札，売り入札の取り消し，買い入札の取り消し）を表す $type_i$，入札価格を表す $price_i$，CPU リソース量を表す $block_i$，そして参加者を識別する ID_i を必要とする．ここで $block_i$ は，売り買いに出される CPU リソースがある決められた処理を一定時間で実行することができる回数で表されるものとする．また，ID_i はオークショナーから与えられた参加者を識別するための情報とする．

このような CPU リソースオークションを想定した場合，参加者・オークショナーの要求として以下のものがあげられる．

- 参加者
  - 入札の秘匿性
  - 入札値の公開
- オークショナー
  - 取引が成立した入札者の情報秘
  - 参加者間での匿名性の維持

また，オークショナー及び参加者の不正の防止もシステムで必要になる．

本稿で提案するシステムの概要は，参加者はオークショナーに対し，いつでも CPU リソースの売り・買いの入札，及び入札の取り消しを行うことができる．オークショナーは，参加者からの入札に対しリアルタイムにマッチングを行うが，入札は匿名で行われているため送られてきた入札はどの参加者が行ったものかを知ることはできない．しかし，マッチングが成立した入札に関しては，オークショナーのみその入札者を特定することができる．

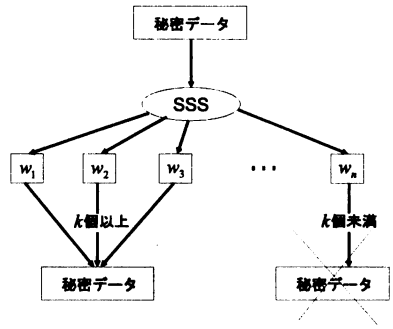

図 1　(k,n) しきい値秘密分散法 (SSS)

ここで，オークショナーに送信する入札情報に ID_i が含まれていると，オークショナーは全ての入札者を特定することができてしまう．そこで本提案手法では入札法に，公開鍵暗号技術，ハッシュ関数，及び秘密分散法を用い参加者の匿名性を考慮したオークションシステムを実現する．本提案システムが実現する項目を以下に記す．

・マッチングが成立していない入札に対し，不正に入札者を特定することが極めて困難

・出されている入札の $type_i$，$price_i$，$block_i$ の公開

・マッチング後の取引は，取引相手を非公開

3. 要素技術：(k,n) しきい値秘密分散法 (SSS)

(k,n) しきい値秘密分散法 $(k \leq n)$ は暗号プロトコルのひとつであり，図 1 に示すように，秘密データから n 個の分散データを作成し，そのうち k 個以上集めると元の秘密データを復元することができるが，k 個未満の場合は秘密データの部分データすら復元することができない特徴をもつ [5]．以下にその構成法を示す．

保管する秘密データを S，n 個の分散データを w_j $(j=1,\cdots,n)$ と表す．S は十分大きな素体 p $(p \leq n+1)$ を法とする有限体 $GF(p)$ の元とし，同様に w_j も $GF(p)$ の元とする．演算は $GF(p)$ 上で行われ，暗号化と復号化は以下の手順で行われる．

分散暗号化

（1）　$GF(p)$ の元から 0 でない別々のものを n 個選びだし，x_j $(j=1,2,\cdots,n)$ とする

（2）　$GF(p)$ の元から $k-1$ 個の乱数 r_l $(l=1,2,\cdots,k-1)$ を選択する

（3）　w_j を

$$w_i = S + \sum_{l=1}^{k-1} r_l x_j^l \quad (mod\, p)$$

表 2　入札時に作成する情報集合

	構成情報	送信先
Bid_i	$type_i$, $price_i$, $block_i$	オークショナー
Id_{ij}	SSS_ID_{ij}	他の参加者

により計算する

（4）　x_j と w_j を保管する（x_j のみ公開することは問題ない）

復　号　化

復号化には，多項式の公式である Lagrange の補完公式を用いる．S は，暗号化の際に作成した多項式の切片を表しているため，Lagrange の補間公式において $x=0$ とすることより求めることができる．

x_j と w_j を k 個以上集め，以下の式に代入することで S を求める．

$$S=\sum_{j=1}^{k} w_j \prod_{1\le l\le k, l\neq j}^{k} \frac{x_l}{x_l-x_j}$$

4. 提 案 手 法

本稿では以下，$H(m)$ は文字列 m のハッシュ値を，$(m_1||m_2)$ は文字列 m_1 と m_2 の連結を，$Sig_x(m)$ を x の公開鍵で m を暗号化したデータをそれぞれ示す．

4.1　売り・買い入札方法

売り・買い入札を行う際，入札者は表 2 に示す二種類の情報の集合を作成する．ひとつは入札情報 Bid_i と呼び，このデータは $type_i$，$price_i$，$block_i$ からなりオークショナーに送信され，マッチングと入札値の公開に用いられる．もうひとつは入札者情報 Id_{ij} と呼ばれ，ID_i に対し (k,n) しきい値秘密分散法を用いた結果の分散データ $SSS\,ID_{ij}$ からなり他の全ての参加者に送信され，マッチング後の入札者の特定に用いられる．オークショナーは，Bid_i からマッチングを行い，取引が成立した入札に対し，参加者から Id_{ij} を回収して ID_i を復元することにより入札者を特定する．このとき，Bid_i と Id_{ij} を結びつける情報として，$one\_time\,ID_i$ と呼ばれる情報が用いられる．参加者が行う売り・買い入札の概要を，図 2 に示す．また，以下に入札の手順を示す．

Step 1: Bid_i の作成

$type_i$，$price_i$，$block_i$ をそれぞれ入力し，$one\_time\,ID_i$ を計算し，Bid_i を作成する

$$one\_time\,ID_i = H(time||ID_i)$$

ここで，$time$ は入札時の時刻情報を表す．

次に，Bid_i を作成し，後述する匿名送信法によりオークショナーに送信する．

$$Bid_i = (one\_time\,ID_i,\ type_i,\ price_i,\ block_i)$$

オークショナーは，送られてきた Bid_i から $type_i$ と $price_i$ と $block_i$ を全ての参加者に公開し，またマッチングを行う．

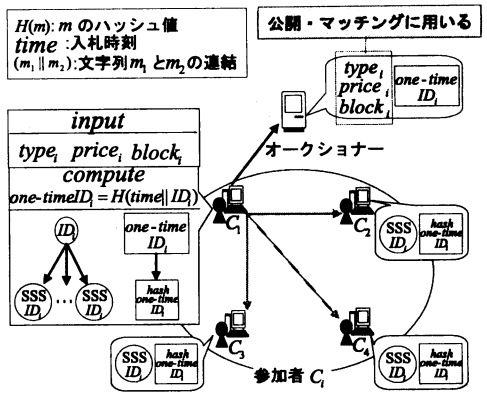

図 2　売り・買い入札法

Step 2: Id_{ij} の作成

ID_i に対し (k,n) しきい値秘密分散法を用いて，分散データ $SSS\,ID_{ij}$ を作成．(n:参加者の総数)

$$w_{ij} = ID_i + \sum_{l}^{k-1} r_l x_{ij}^l \quad (mod\ p)$$

$$SSS\,ID_{ij} = (x_{ij},\ w_{ij})$$

次に，Bid_i に添付した $one\_time\,ID_i$ のハッシュ値を計算し，$hash\ one\_time\,ID_i$ とする．

$$hash\ one\_time\,ID_i = H(one\_time\,ID_i)$$

参加者数分 (n) の Id_{ij} を作成し，そのうちのひとつを自分で保持し，残りを後述する匿名送信法で他の全ての参加者に送信する．

$$Id_{ij} = (SSS\,ID_{ij},\ hash\ one\_time\,ID_i)$$

以上が売り・買い入札の手順である．

オークショナーは，送られてきた Bid_i に含まれる $one\_time\,ID_i$ から ID_i を求めることは，ハッシュ関数が一方向関数であるという性質上困難になる．また，入札の度に入力値の時間情報 $time$ の値が異なるため，同一入札者の異なる入札の $one\_time\,ID_i$ は相関を持たない．このため，オークショナーは $one\_time\,ID_i$ から入札者を特定することは非常と困難になる．

また，Id_{ij} に関しても秘密分散法を適用するときに用いる乱数 r_l，及び x_{ij} の値は入札の度に新たに生成されるため，異なる入札での値は相関を持たなくなる．また，少数 (k 未満) の $SSS\,ID_{ij}$ から ID_i を特定することも秘密分散法の特徴より不可能である．

また，Bid_i では $one\_time\ ID_i$ を用い，Id_{ij} では $hash\ one\_time\,ID_i$ を用いるのは，後述するが $one\_time\,ID_i$ はその入札の取り消しを行う際に必要となるためである．

4.2　匿名送信法

Bid_i と Id_{ij} に含まれている情報からは入札者を特定することはできない．しかし，これらのデータを直接送信先に送ると，

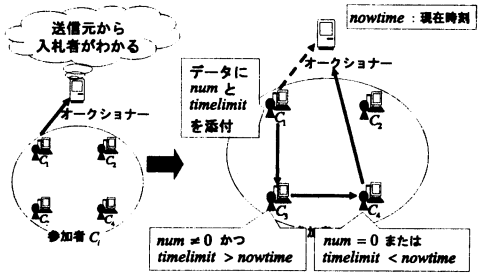

図 3　各データの送信法

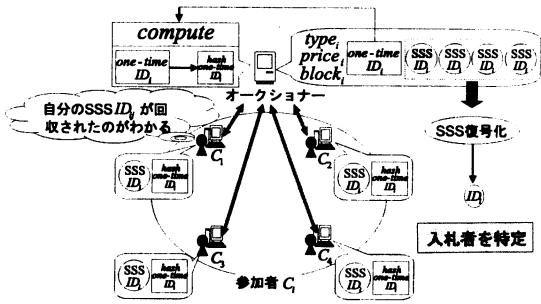

図 4　取引成立後の入札者の特定法

受信者は送信元を調べることにより入札者を特定することができる．そこで，本提案手法では，これらふたつのデータを送信する際，図 3 に示すように，参加者間で複数回ランダムにデータを送信した後，最終送信先に送る手法を用いる．

以下に各データの送信法を示す．

入札者 (送信者) の処理

- Bid_i

入札者は，Bid_i をオークショナー (A) の公開鍵で暗号化する．

$$Sig_A(Bid_i)$$

次に，乱数 $num\ (mod\ n)$ を生成し，データ送信のタイムリミット $timelimit$ を入力し，このふたつのデータとオークショナーの IP アドレス IP_A と暗号化した Bid_i をひとつのデータ $Send\_data$ とし，ランダムに選んだ他の参加者に送信する．

$$Send\_data = \left(timelimit,\ num,\ IP_A,\ Sig_A(Bid_i)\right)$$

- Id_{ij}

Id_{ij} の場合は，まずデータ中の $SSS\ ID_{ij}$ をオークショナーの公開鍵で暗号化し，次にこの暗号化した $SSS\ ID_{ij}$ と $hash\ one\_time\ ID_i$ に対して最終送信先参加者 (C_j) の公開鍵で暗号化を行う．

$$Sig_{C_j}\left(Sig_A(SSS\ ID_{ij},\ hash\ one\_time\ ID_i)\right)$$

この暗号化したデータに Bid_i 同様，num，$timelimit$ 及び最終送信先参加者の IP アドレス IP_{C_j} を添付して $Send\_data$ を作成し，ランダムに選んだ他の参加者に送信する．

$$Send\_data = \left(timelimit,\ num,\ IP_{C_j},\right.$$
$$\left.Sig_{C_j}\left(Sig_A(SSS\ ID_{ij},\ hash\ one\_time\ ID_i)\right)\right)$$

受信者の処理

$Send\_data$ を受け取った参加者は，$timelimit$ と num をチェックし，その値によって送信先を選択する．

以下，$nowtime$ は現在時刻を表すものとする．

- $num \neq 0$ かつ $timelimit < nowtime$ のとき，

num の値を更新し，ランダムに選んだ他の参加者に送信する．

$$num \leftarrow num + random\ \ (mod\ n)$$

ここで，$random$ は乱数を表す．

- 上記以外のとき

$Send\_data$ 中の IP_x $(x : A\ or\ C_j)$ にデータを送信する．

以上が各データの送信方法である．

num は常に $GF(n)$ 上の数値をとり，この数値が 0 である $Send\_data$ を受け取った参加者が最終送信先に送る．$GF(n)$ 上の数値を用いることにより，$Send\_data$ を受け取った参加者から見て，ひとつ前の参加者が入札者である可能性が 1/n になる．

しかし，この num を用いた送信方法だけでは，$num = 0$ に達するまで時間がかかる可能性がある．そのため，この num を用いた手法に加え，入札者がデータ送信のタイムリミットを設定する方法を用いている．入札者は $timelimit$ と表される，最終送信先に遅くても届けたい時刻をデータ送信時に添付する．

これらふたつの手法を組合わせることにより，Bid_i 及び Id_{ij} を受け取ったオークショナー及び参加者は，送信元を調べても入札者を特定することができない．

また (k, n) しきい値秘密分散法の性質上，同一入札の Id_{ij} を k 種類以上中継した参加者は，ID_i を復元することができてしまう．さらに，Bid_i を中継した参加者は，$one\_time\ ID_i$ の値がわかってしまうため，悪意のある参加者はその値を用いて入札の取り消しを行うことができてしまう．このため，各データは最終送信先相手の公開鍵で暗号化を行ってから送信する．これにより，参加者はデータの中継を行うが，自分宛以外のデータの中身を見ることができない．

4.3　取引成立後の入札者の特定法

オークショナーがマッチングを行い，取引が成立した Bid_i の入札者を特定するとき図 4 に示す手順に従う．

まずオークショナーは，取引を成立させた Bid_i の $one\_time\ ID_i$ から $hash\ one\_time\ ID_i$ を計算して，その $hash\ one\_time\ ID_i$ をもつ Id_{ij} の回収命令を全ての参加者に出す．

参加者は回収命令がくると，送られてきた $hash\,one\_time\,ID_i$ の Id_{ij} をもっているならば，オークショナーに送信する．このとき，入札者は Id_{ij} のひとつをもっていることから，自分のものが回収されたことがわかる．

次にオークショナーは，回収した Id_{ij} から ID_i を復元することにより入札者を特定する．

以上が取引が成立した入札者を特定する手順である．

入札者は自分の Id_{ij} が回収されたことがわかることより，オークショナーは不正に ID_i を復元し入札者を特定することは困難になる．ID_i を復元するには，k 個以上の Id_{ij} を必要とする．このためオークショナーが入札者に知られることなく不正に ID_i を復元するには，入札者以外の k 人の参加者から Id_{ij} を回収するか，k 人の参加者と共謀していない限り不可能である．また，Id_{ij} 中の $SSS\,ID_{ij}$ はオークショナーの公開鍵で暗号化しているため，単純に k 人以上の参加者だけが共謀したところで，ID_i を復元することは不可能になる．

これらのことから，k は小さすぎる値であってはいけない．少なくとも n の 8 割以上の値をとることが望まれる．

4.4 入札の取り消し法

参加者は，まだ取引が成立していない入札の取り消しを行うことができる．この手順を以下に示す．

キャンセル入札データの作成

入札者は，入札時に用いたものと同値の $one\_time\ ID_i$ を用い，$type_i$ に売り，または買い入札のキャンセルを示す値を代入し，入札のときと同じ手順で Bid_i を作成して，オークショナー宛でランダムに選んだ他の参加者に送信する．

取り消し処理

オークショナーは受け取ったキャンセル入札の $one\_time\ ID_i$ と同値の入札を削除し，その $one\_time\ ID_i$ から $hash\ one\_time\ ID_i$ を計算して，その $hash\ one\_time\ ID_i$ をもつ Id_{ij} の削除命令を全ての参加者に出す．

参加者は削除命令がくると，送られてきた $hash\ one\_time\ ID_i$ の Id_{ij} をもっているならば，削除する．

以上が入札の取り消しの手順である．

入札の取り消しをする際に，取り消したい入札に用いた $one\_time\ ID_i$ を必要としているのは，$one\_time\ ID_i$ が入札者とオークショナーのみが知りうる値だからである．他の参加者は，$hash\ one\_time\ ID_i$ の値を持っているが，ハッシュ関数が一方向である特徴より $hash\ one\_time\ ID_i$ から $one\_time\ ID_i$ を求めることは困難である．このため，正式な入札者以外が入札の取り消しを行うことは困難になる．

4.5 取引方法

取引成立後，参加者間（売り手・買い手）が直接取引を行うと，参加者は取引相手がわかってしまう．取引相手がどの企業かわかると，買い手が依頼した処理内容を推測される恐れがあり，これは企業秘密に繋がることになる．また，取引相手がライバル企業であった場合，悪意のある参加者により取引の弊害が生じる可能性もある．取引を参加者間で直接行うことは，入札を匿名で行っても企業秘密及びオークションの障害を完全に回避できない結果に繋がる．

そこで本提案システムでは取引を行う際，図 5 に示すように買い手・売り手のやり取りは，それぞれオークショナーを中継して送信する．ここでのやり取りの暗号化などは，各参加者の自己責任として本提案手法では扱わない．この手法により，参加者は取引相手を知らされることなく取引を行うことになり，企業秘密の流出とオークションの障害を抑えることができる．

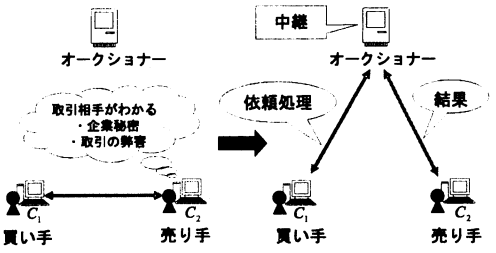

図 5 取引方法

5. 実　　験

5.1 システム化

提案手法のシステム化を行った．開発環境は，ネットワークプログラミングに適しているため，Java 言語（J2SDK 1.4.2_06）を用いた．また，公開鍵暗号による暗号化及び復号化を用いるため，暗号プロバイダーとして JCSI 2.3 を用いた．

5.2 設　　定

提案手法の試作システムを構築し，CPU リソースの売買を行うことによる参加者のリソースの使用状態の考察を行う．

本実験では，オークショナー用に 1 台のマシン（Windows マシン），参加者用に 8 台のマシン（Windows マシン 2 台，Linux マシン 6 台）を用い，参加者用のマシンは 1 台が 1 参加者として振舞う．

参加者にはそれぞれ，タスクのスケジュールをランダムに与え，そのタスクの開始時間及び，ひとり（1 台のマシン）で行った場合の終了時間をあらかじめ全て把握しているものとする．図 6 に各参加者のタスクのスケジュールを示す．このグラフにおいて横軸は全体で 12 時間の時間の流れを示し，縦軸はタスクを実行しているか否かを示している．

5.3 入札規則

本実験においては，入札の際にリソースの規模を表す $block_i$ は考慮しないものとし，各参加者以下に示す入札規則に従う．

- 売り入札

参加者は，何も処理を行っていない（アイドル状態）場合以下の規則に従い売り入札を行う．

・ アイドル状態が 60 分以上続く場合

$price_i = 1$ で売り入札を出す

・ アイドル状態が 30〜59 分続く場合

$price_i = 2$ で売り入札を出す

・ アイドル状態が 20〜29 分続く場合

$price_i = 3$ で売り入札を出す

また，次の自分のタスクが 20 分以内にある場合は入札を行わず，取引が成立していない売り入札がある場合は，入札の取り消しを行う．

- 買い入札

参加者は，自分のタスクを行っている場合に以下の規則に従い買い入札を行う．

・ タスクの残り時間が 30 分未満の場合

表 3 実験結果

	売買なしでの総実行時間 (分)	自分のタスク実行時間 (分)	取引タスクの実行時間 (分)	総実行時間 (分)	売り総額	買い総額	差し引き額
C_1	180	50	227	257	278	126	152
C_2	210	56	275	331	302	236	66
C_3	240	99	203	302	213	117	96
C_4	240	102	194	296	199	174	25
C_5	300	69	239	308	251	219	32
C_6	360	93	211	304	250	250	0
C_7	540	98	199	297	227	425	-198
C_8	540	134	174	308	197	370	-173

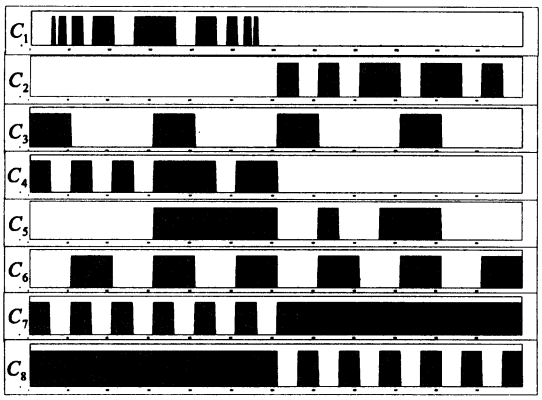

図 6 各参加者に振り分けられたタスク

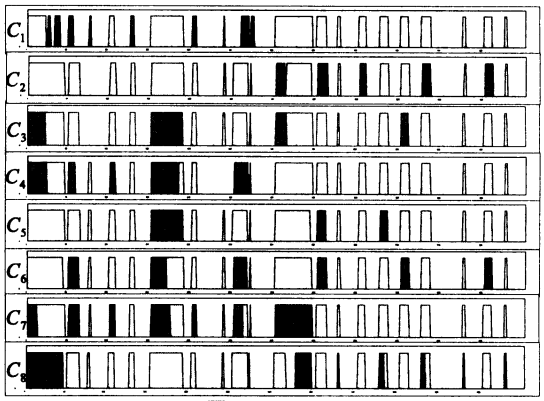

図 7 提案システムを用いた結果

$price = 1$ で買い入札を出す

・ タスクの残り時間が 30〜59 分の場合

$price = 2$ で買い入札を出す

・ タスクの残り時間が 60 分以上の場合

$price = 3$ で買い入札を出す

入札は一度にひとつしか出すことができず，また一度にひとつの参加者にしか売ることはできないが，複数の参加者から買うことはできるものとする．

また取引の終了条件は，買い入札での取引の場合，自分の残りのタスク時間が 10 分以内になるとそれまでに依頼した処理の結果が返ってきたときに取引を終了させる．売り入札での取引の場合は，次の自分のタスクの開始時間まで 10 分未満になったときに取引を終了させる．

オークショナーは，最安値の売り入札と最高値の買い入札を優先して，5 秒おきに行うものとする．

5.4 実験結果

提案システムを用いた場合の結果のタスクの実行状態を図 7 に示す．また，結果をまとめたものを表 3 に示す．

これらの結果から，売買を行うことでそれぞれのタスクの終了時間が大幅に早くなり，その時間を用いて他の参加者の処理を行っているのがわかる．

これらのことから，本研究で提案した手法を構築した試作システムを用いて CPU リソースの売買を行うことにより，全体として参加者は CPU リソースを無駄なく使用することができることが確認された．

6. 結　論

既存の暗号技術を組み合わせ，企業間で CPU リソースの売買を行う匿名性の高いオークションシステムを提案した．また，プロトタイプシステムを実装した実機実験から，CPU リソースの売買を行うことにより参加者は CPU リソースを無駄なく使用できることを確認した．

文　献

[1] William E. Johnston, et al, "Grids as Production Computing Environments:The Engineering Aspects of NASA's Information Power Grid", Proceedings of 8th IEEE Symposium on High Performance Distributed Computing, IEEE Press, 1999.

[2] H. Kikuchi et al, "Multi-round Anonymous Auction Protocols", Proceedings of the First IEEE Workshop on Dependable and Real-Time E-Commerce Systems, June, pp.62-69, 1998.

[3] D. Chaum et al, "Multi-party unconditionally secure protocols", Proceedings of ACM STOC '88, pp.11-19, 1988.

[4] Changjie Wang, Ho-fung Leung, "Anonymity and Security in Continuous Double Auctions for Internet Retails Market", Proceeding of the 37th Hawaii International Conference on System Sciences, 2004.

[5] A. Shamir, "How to share a secret", Communication of the ACM, Vol.22, No.11, pp.612-613, 1979.