

協調問題解決手法の並列化

炭田 昌人 進藤 静一
三菱電機(株)中央研究所

協調問題解決の一手法であるコントラクトネットを並列マシン上に実現するための方法と、その方法を具体的な問題に適用した実験と評価について報告する。並列化の方法として、同期式協調方式と非同期式協調方式の2つの方法を提案する。両方式による問題解決は、タスク公示-入札-落札という交渉ステップを繰り返すことにより進展する。この交渉ステップの同期・非同期により方式が別れる。両方式を配送故障復旧問題に適用し、並列化の適合性を処理時間とメッセージ数の観点から評価した。両方式とも並列マシンとの親和性はあるが、同期式はメッセージ数で劣り、非同期式は解の再現性に問題があることが分かった。

Two Parallel Implementation methods For Cooperative Problem Solving

Masato Sumida Seiichi Shindo
Central Research Laboratory
Mitsubishi Electric Corporation

Two parallel implementation methods for *Contract Net*, which is one of the fundamental cooperative problem solving methods, are proposed. One works in a synchronized way and the other works in an asynchronized way. Our methods solve a problem through the iterated step which is a contract sequence of "task announcement", "bid" and "award". To evaluate our methods, we applied them to the trouble recovery problem in the area of transportation. The experiments showed that both of our methods are suitable to parallel computing environment. But the synchronized method needs much more messages for contract sequence, when the problem becomes larger.

1 はじめに

最近、分散人工知能の研究が盛んに行なわれている。分散人工知能を端的に表現すると、複数の人間（集団）による知的活動のモデル化であると言える。集団による知的活動の中で、複数の人間が協調しながら問題を解決する過程に着目した分野が、（分散）協調問題解決の分野である。その中でコントラクトネット（以下C-ネット）は、集団の交渉（negotiation）過程をメタファとして開発された手法である [1]。C-ネットではエージェント（知識源）を問題解決の自律的行動主体として、エージェント間のプロトコルを規定している。ただし、C-ネットはエージェント間のプロトコルを定めているが、その実現メカニズムについては触れていない。

一方、コンピュータハードウェアの進歩は著しく、近年、数々の並列マシンが開発されている。しかし、並列マシンの能力を十分に活かすソフトウェア技術は、まだまだ発展途上にある。

そこで、本報告では並列マシン上でC-ネットを実現するための手法を提案し、手法を具体的な問題に適用することにより、解析・評価を行なう。なお、実現に用いた並列マシンは、疎結合分散メモリ型の並列マシン（マルチPSI [2]）である。

2 コントラクトネットの概要

並列化の対象となるC-ネットについて簡単にその概要を説明する。C-ネットは、人間の交渉過程をメタファとした、計算機上での協調問題解決手法を定めている。個々の人間と対応した、計算機上の自律的な行動主体としてエージェントが存在する。複数のエージェントがタスク公示－入札－落札という交渉ステップを繰り返しながら、問題解決が進められる。

C-ネットでは、エージェント間の交渉プロトコルとして以下が規定されている。これらのプロトコルに沿って、エージェント間でメッセージが通信される。

[C-ネットの交渉プロトコル]

(1) タスク公示 (task announcement)

他エージェントに依頼したい仕事（タスク）をもつエージェントがマネージャ（manager）となり、そのタスクを他のエージェントに公示するときのプロトコルである。他のエージェントがタス

クを引受るための条件、入札すべき値の種類、期限などが記述される。

(2) 入札 (bid)

マネージャが公示したタスクに対して、他のエージェントが入札値を返すためのプロトコルである。タスク公示に対して入札する可能性のあるエージェントは、そのタスクのマネージャに対する潜在的コントラクタ（potential contractor）と呼ばれる。

(3) 落札 (award)

入札期限になると、マネージャはそれまでに入札してきた潜在的コントラクタの中から、最適な相手を選択し、そのエージェントを相手として契約（contract）を結ぶ。このときのプロトコルが落札である。

このC-ネットの特徴は、

(a) 複雑な情報をタスク公示－入札－落札という交渉シーケンスに埋め込めること、

(b) 個々のエージェントに関連した特有の状況で、エージェントが情報をローカルに評価できること、

(c) マネージャ・潜在的コントラクタの選択が双方向にできること

である。注意すべき点は、複数のマネージャが存在し得ることであるが、C-ネットはマネージャ間の関係を明確に規定していない。そこでC-ネットの実現に際しては、各マネージャが管理する交渉グループ間の関係を定める必要がある。交渉グループは、マネージャとそのタスクに入札した潜在的コントラクタの集団を指す。

3 並列化の手法

C-ネットの並列化に際して、我々は個々のマネージャが管理する交渉グループ間の関係に着目して、次の2種類の並列化の方式を提案する [3]。

(1) 同期式協調方式

すべてのマネージャが入札情報を共有し、全体で一つの交渉グループを形成して、各マネージャのタスクの契約相手を決定する。

(2) 非同期式協調方式

各マネージャが、各々自分の交渉グループを管理し、お互い独立に交渉を進める。

3.1 同期式協調方式

同期式協調方式では、全マネージャが一つの交

渉グループを形成し、交渉ステップを繰り返して行く。このとき各マネージャは同期しており、一つの交渉ステップ内でのタスクの割当（公示タスクとその契約相手）が決定してから、次の交渉ステップに進む。図1に同期式協調方式の一交渉ステップ内のアルゴリズムを示す。

同期式のアルゴリズムではプロトコルとして以下のメッセージを用いる。

- ・公示メッセージ：マネージャがタスクを公示する。
- ・非公示メッセージ：非マネージャがタスクを公示しないことを他に知らせる。
- ・入札メッセージ：潜在的コントラクターが公示タスクに入札する。
- ・共有メッセージ：マネージャが他のマネージャに自分への入札値を通知する。
- ・落札メッセージ：入札した潜在的コントラクターにマネージャが契約相手を通知する。

C-ネットに対して、非公示メッセージと共有メッセージを追加している。非公示メッセージは、すべてのマネージャが公示済みになったことを各エ

ージェントが認識するために用いられる。これによりタスク公示の終了を認識するための無駄な待ち時間を省くことができる。また、共有メッセージはマネージャ間のタスク割当の衝突を防ぎ、ステップ内での最適なタスク割当を実現するために用いられる。

以下、図1に沿って処理の流れを説明する。

(i) 終了条件の判定〔図中(1)〕

すべてのマネージャは公示候補のタスク集合を保持する。そのタスク集合が全マネージャで空になった時点で、全体の処理が終了する。公示タスク候補は公示条件を満たすタスクの集合である。一度公示に出して契約相手が見つからなかった場合にそのタスクは候補から除かれる。ただし、契約が成立した（どのエージェント間でも良い）時点で、除かれていたタスクは再び候補に戻される。従って、一度も契約が成立せずに、全マネージャがタスクを公示し終わったとき、全体の処理も終了する。

(ii) 態度表明〔図中(2)(3)(4)(13)〕

エージェントが公示タスク候補をもつエージェ

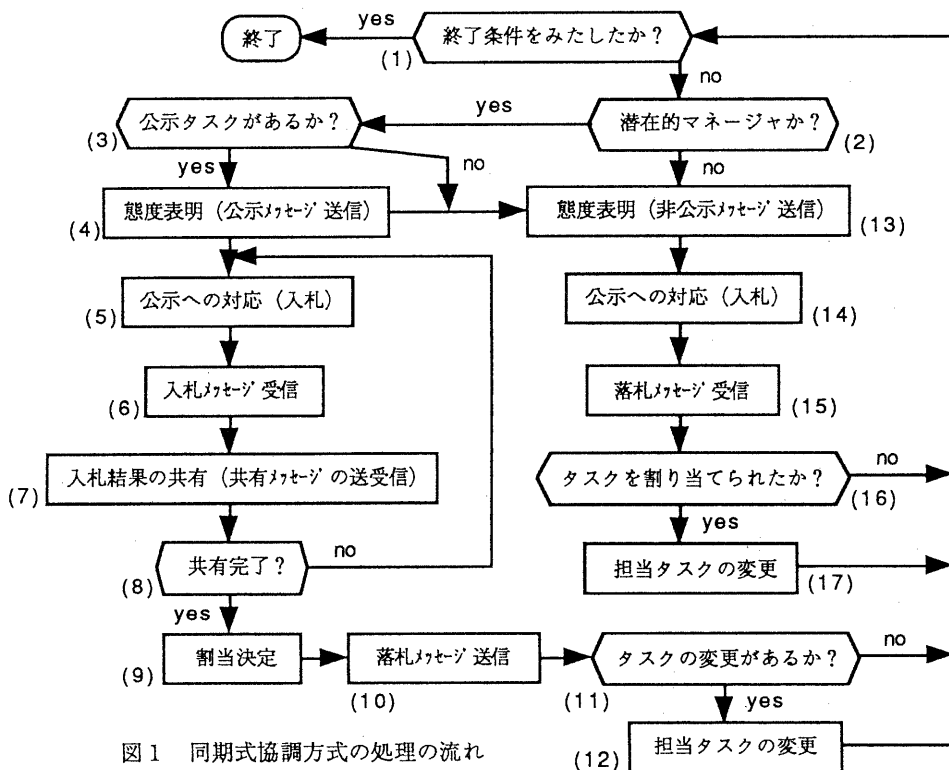


図1 同期式協調方式の処理の流れ

ント（潜在的管理者）であれば、公示タスク候補を調べ、公示に出せるタスクを選択し、公示メッセージを使ってそのタスクを他のエージェントにブロードキャストする。もし公示に出せるタスクが見つからなければ、非公示メッセージでその旨を他エージェントに知らせる。エージェントが潜在的管理者でない場合にも、非公示メッセージを他へ送信する。

(iii) 公示への対応 [図中(5)(14)]

全エージェントは公示メッセージに対して、メッセージ中で指定された入札値を求め、公示タスクの管理者に入札メッセージを送る。非公示メッセージに対しては、その送信エージェントを記録しておく。各エージェントは他のエージェントから公示・非公示メッセージのどちらかを受信することになる。

(iv) 情報共有 [図中(6)(7)(8)]

全管理者は公示タスクとその入札値のマトリックス（評価マトリックス）を保持する。自分以外の管理者は (iii) で認識できる。評価マトリックスを共有するために、管理者は自分への入札値を他管理者へ共有メッセージで通知する。評価マトリックスが完成した時点で、タスクの割当を各管理者が決定できる。

(v) タスク依頼 [図中(9)(10)]

各管理者は評価マトリックスを元に、最速で衝突のない割当を決定し、潜在的コントラクターに契約相手を落札メッセージを用いて通知する。

(vi) タスクの変更認識 [図中(11)(12)(15)(16)(17)]

管理者の場合、(v) で契約が成立していれば、落札メッセージ送信終了後、ただちに自分のタスクを変更する。潜在的コントラクターの場合には、落札メッセージを受信して自分が契約相手であれば、契約内容に従い自分のタスクを変更する。落札メッセージをすべて受信した後、次の交渉ステップへ移行する。このときタスクを引受ることにより新たに潜在的コントラクターが管理者に変化する可能性が存在する。

3. 2 非同期式協調方式

非同期式協調方式では、各管理者が独立に交渉グループを形成し、非同期に交渉を進めていく。すなわち管理者は他の管理者の交渉を気にすることなく、自分の交渉ステップを順次進めることができる。ただし、潜在的コントラク

タは、あるタスク公示に入札している間、他のタスク公示に入札することはできない。図2に非同期式協調方式の一交渉ステップ内のアルゴリズムを示す。

非同期式のアルゴリズムではプロトコルとして公示・入札・落札メッセージ以外に、以下のメッセージを用いる。

- ・入札拒否メッセージ：潜在的コントラクターが公示タスクに対して入札条件を満たさないと、入札を拒否することをそのタスクの管理者に通知する。
- ・多忙メッセージ1：入札中なので別タスクに入札できない旨を潜在的コントラクターが管理者に返答する。
- ・多忙メッセージ2：管理者が他の管理者の公示タスクに入札できない旨を知らせる。
- ・状態変化メッセージ：タスクの契約により、自分のタスク状態が変化したことを他に知らせる。

これらのメッセージは、管理者がタスク公示を送る相手を決める際に用いる情報（通信履歴）を管理者に伝えるために導入された。管理者は盲目的にタスクをブロードキャストするのではなく、通信履歴により公示すべき相手を選び出す。管理者は自分のタスク毎の通信履歴に未公示であるか、公示に出した場合には、公示に対する他エージェントの返答メッセージの種類を送信エージェントと共に記録しておく。

以下、図2に沿って管理者と潜在的コントラクターとに分けて処理を説明する。

(i) 潜在的コントラクターの振舞い [図中(1)-(8)]

まず到着している最初の公示メッセージを取り出す(1)。公示メッセージのタスクを評価し(2)、入札条件を満たしていればタスクの管理者に入札メッセージを送り、条件を満たさなければ入札拒否メッセージを管理者に送る(3)(4)(5)。入札拒否の場合には、拒否した相手の管理者を記録しておき（入札拒否記録）、次の公示メッセージを処理する。入札した場合には、その管理者からの落札メッセージを待つ(6)。入札してから落札の処理が終わるまでに到着した公示メッセージに対しては、多忙メッセージ1を送り返しておく。落札の契約相手が自分か否か判断し(7)、自分であればタスクの変更処理を行なう(8)。このとき入札拒否記録中のエージェントに対しては状態

変化メッセージを送り、入札拒否記録をリセットする。自分が契約相手でなければ次の公示メッセージを処理する。

(ii) マネージャの振舞い [図中 (11)-(19)]

到着した公示メッセージに対しては、多忙メッセージ2を送り返し、状態変化メッセージが到着した場合には、それを通信履歴に反映する(11)。通信履歴中に現われる状態変化メッセージの送信エージェントからの返答の種類を多忙状態に変更する。次に、公示に出すタスクを選択する(12)。このとき、公示条件を満たすタスクで、かつそのタスクの通信履歴が未公示であるか、履歴に多忙メッセージ1が含まれるものを選択する。公示タスクが無くなった場合には、その旨を管理部(次の終了判定の項を参照)に知らせる。公示タスクが決まると、そのタスクの通信履歴を調べ、未公示であればタスク公示をブロードキャストし、公示済みであれば公示に対する返答が多忙メッセー

ジ1のエージェントに再度公示する(13)。公示に出したエージェントからの返答メッセージがすべて揃うまで待ち(14)、入札値を返してきたエージェントがあるか否か判定する(15)。入札エージェントが無かった場合には次のメッセージの処理に移り、有った場合にはその入札値を評価し(16)、最も評価の高いエージェントを契約相手とする落札メッセージを、入札した全エージェントに通知する(17)。契約が成立した場合には、自分のタスクを契約内容に従って変更し、契約が不成立の場合には次のメッセージの処理に移る(18)(19)。

(iii) 終了判定 [図中 (9)(10)と関連]

全体の処理は、すべてのマネージャが公示に出すタスクの無くなったとき終了する。この状態を検出するために、エージェントとは別に管理部を設ける。各エージェントは新たに自分がマネージャになった場合に管理部にその旨を通知する(10)。逆にマネージャでなくなった場合にも管理部に知

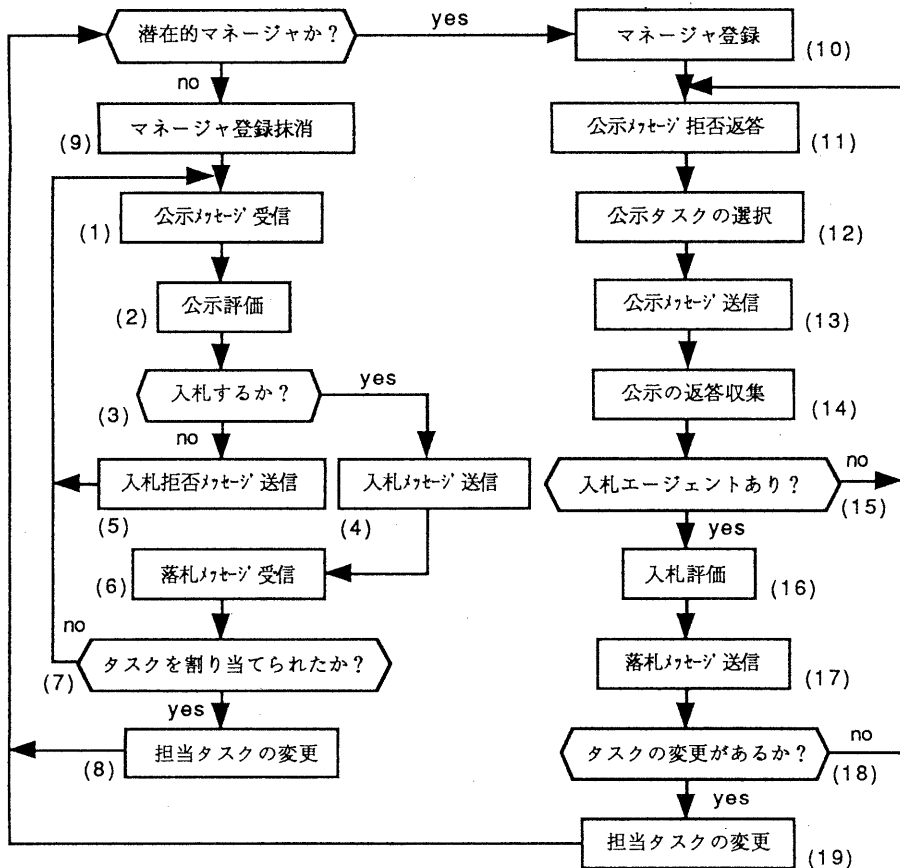


図2 非同期式協調方式の処理の流れ

らせる(9)。管理部ではマネージャの状態を把握して、すべてのマネージャの公示タスクがなくなったことを検知したとき、全体の終了処理を行なう。

4 適用実験

4.1 対象問題

3節で提案した二つの方式を評価するために、配送故障復旧問題を対象として両方式を適用した。配送故障復旧問題は、配送中のトラックが故障したとき、その故障トラックが配送予定であった荷物を、配送中の他のトラックが代理配送する問題である。今、問題解決の目標を配送遅延時間の総和をなるべく小さくすることに設定する。一台のトラックの配送遅延時間は次式で与えられ、すべてのトラックの遅延時間の和を最小とすることが目標となる。

$$\text{配送遅延時間} = \sum_{PT > OT}^{\text{Task}} (PT - OT)$$

Task: 配送荷物

OT: 荷物の配送希望時刻

PT: 荷物の配送予定時刻

このとき、協調方式のエージェントがトラックに対応し、方式の各プロトコルとの対応は以下のようになる。

- ・公示タスク: 他のトラックに配送させたい荷物の情報(配送先店舗の位置、配送希望時刻等)
- ・入札値: 公示タスクで指定された荷物を引受ることにより生じる配送遅延時間
- ・落札相手:
 - (同期式) 最も遅延時間の総和が減少する組合せの相手
 - (非同同期式) マネージャとコントラクターとの遅延時間の和の減少が最大となる相手

両方式とも遅延時間の発生したトラックがマネージャとなり、タスク(荷物)を公示する。各トラックは公示の荷物を組み込んだ場合の配送ルートを計画することにより公示タスクを評価する。契約が成立すると、契約内容に従って担当タスクが変化し、配送ルートが変更される。配送ルートの変更例を図3に示す。各方式に従って、交渉を

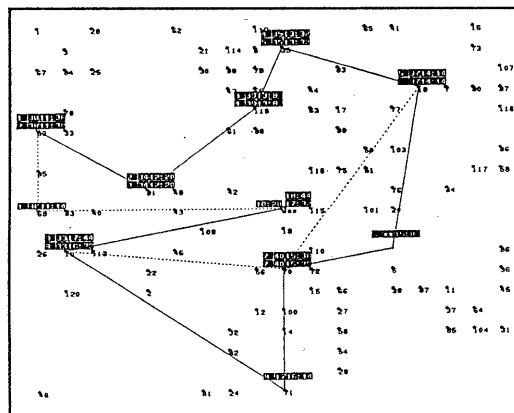
繰り返すことにより、徐々に配送遅延時間の総和が減少していく。

4.2 実験結果

エージェント数の異なる4つの例題(エージェント数: 15・30・60・120)に両方式を適用し、遅延時間総和の変化、処理時間、総メッセージ数、落札数を計測した。4つの例題ともに、初期計画(遅延のない状態)実行中に故障トラックが発生し、その荷物を適当に他トラックに割り付けた状態からアルゴリズムを開始している。実験には要素プロセッサ(PE)数が16PE・64PE構成のマルチPSIを使用し、エージェントを割り付けるPE数を1・2・4・8・15・30・60と変えて計測を行なった。エージェントの割り付け方は、使用できるPEにサイクリックに割り付ける方法をとった。ただし、全体を管理する(初期処理・終了処理等)ために、別に1PEを使用している。

(1) 同期式協調方式

同期式協調方式で4つの例題を実行した計測結果を表1に示す。遅延時間総和の変化は、測定開始時からアルゴリズム終了時までの遅延時間の総和の減少を表している。総メッセージ数はその間全体でやり取りされたメッセージの総和である。落札数は契約が成立した数である。同期式協調方式は処理が決定的に進むので、これらの結果は何度計測しても変化しない。このときの2回計測した平均の処理時間と台数効果を表2に、台数効果をグラフにした図を図4に示す。台数効果は例題



● 店舗 黒枠内: 配送順, タスク番号, 予定時刻
変更前点線ルート(下段黒枠) 変更後実線ルート(上段黒枠)

図3 配送ルートの変更例

毎に、1PEの処理時間を基準とし、それを各処理時間で割った値である。

(2) 非同期式協調方式

同期式と同様に計測した結果をまとめて、表3に示す。非同期式協調方式では、処理が非決定的に進むので、計測する毎に結果が変わってくる。そこで、各例題を15PEで実行し、5回ずつ計測を行なった。

5 解析・評価

提案した両方式ともエージェントを単位としてプロセッサへの割付が可能であり、エージェント数の増減に柔軟に対応できるので、並列マシンと親和性の高い方式となっている。処理時間の観点から、さらに両方式の解析・評価を試みる。

同期式協調方式では、エージェント数=PE数の場合に、エージェント間のメッセージ通信時間とGCの影響を無視した理想的な台数効果は次式で与えられる。

$$\frac{C_{ann}T_{mng} + C_{step}N_aT_{cntr}}{C_{step}(T_{mng} + T_{cntr})}$$

- C_{ann} : 全交渉中の公示タスクの総数
- C_{step} : 終了までの全ステップ数
- N_a : 全エージェント数-1
- T_{mng} : 1ステップのマネージャとしての処理時間
- T_{cntr} : 1ステップの潜在的コントラクタとしての処理時間

この式に15エージェントの例題の15PEのときの台数効果をあてはめると、

$$T_{mng} : T_{cntr} = 1 : 0.12$$

となる。従って、潜在的コントラクタに比べて、マネージャの計算量が大きいと考えられる。このことから、PE数<エージェント数の場合には、なるべくマネージャが均等になるようプロセッサに割り付ければ良いことがわかる。しかし、マネージャは動的に変化するので、全交渉過程を通じて、

表1 同期式協調方式・計測結果

エージェント数	遅延時間総和	総メッセージ数	落札数
15	1,642 → 479	9,077	21
30	823 → 276	22,745	16
60	464 → 168	68,877	12
120	248 → 84	118,440	5

台数効果

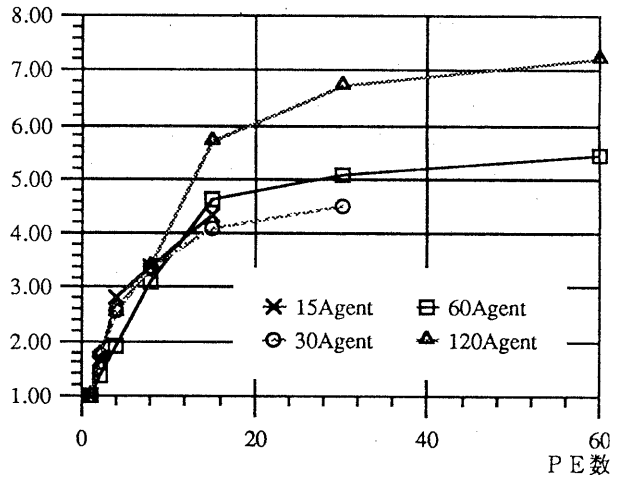


図4 同期式協調方式・台数効果

表2 同期式協調方式・処理時間 (処理時間単位: 秒)

		PE数							
		1	2	4	8	15	30	60	
エ ー ジ ェ ン ト 数	15	処理時間	2,070	1,237	738	614	479		
		台数効果	1	1.67	2.80	3.37	4.32		
	30	処理時間	2,885	1,656	1,121	855	703	641	
		台数効果	1	1.74	2.57	3.37	4.10	4.50	
	60	処理時間	5,292	3,899	2,792	1,704	1,147	1,042	975
		台数効果	1	1.36	1.89	3.10	4.61	5.08	5.43
	120	処理時間	4,473	2,597	1,757	1,324	787	667	620
		台数効果	1	1.72	2.54	3.37	5.68	6.71	7.21

表3 非同期式協調方式・結果集計

エージェント数	遅延時間総和	総メッセージ数	落札数	時間(s)
15	1,642 → 713	5,465	28	101
	1,642 → 789	4,318	25	102
	1,642 → 763	4,637	25	130
	1,642 → 789	4,526	25	101
	1,642 → 956	4,670	25	168
30	823 → 172	9,766	28	150
	823 → 126	10,897	26	168
	823 → 160	7,549	24	94
	823 → 160	5,854	26	121
	823 → 160	5,594	25	89
60	464 → 174	5,288	11	120
	464 → 224	4,345	9	101
	464 → 224	4,244	9	101
	464 → 224	4,304	9	102
	464 → 224	4,282	9	95
120	248 → 84	1,050	2	44
	248 → 84	1,453	3	54
	248 → 84	1,551	3	53
	248 → 85	1,749	3	79
	248 → 84	1,555	3	52

そのような割付を維持することは困難である。また、ステップ単位で交渉が同期しているため、潜在的コントラクタの割り付けられたPEでは、アイドル期間が発生する。そこで、この期間にマネージャの処理を代替させれば良いと考えられるが、1エージェント1PEというモジュール性を損なうことになる。

一方、非同期式協調方式では、上で述べたようなエージェントの計算量の違いとステップ間の同期から生じる無駄な待ち時間は、同期式ほど無いと思われるが、表3からも分かるように処理が非決定的であるため、台数効果を測定できない。そこで、処理時間÷落札数の値を同期式と比較してみた。表1・2・3でPE数15のとき、同期式の場合、この値は25.3(15エージェント)・51.0(30エージェント)・110.9(60エージェント)・192.6(120エージェント)であり、非同期式の場合、3.6~6.7(15エージェント)・3.6~5.4(30エージェント)・10.6~11.2(60エージェント)・17.3~22.0(120エージェント)である。このことから、1回の落札に関する効率は非同期式の方が良いことがわかる。しかし、1回の落札による解の改善度(遅延時間総和の減少)は同期式の方が良い。従って、解を出すまでに許される時間に依存して、二つの方式の有効性が交錯する。

メッセージ数の観点から両方式を比較してみる。同期式の場合、交渉に要するメッセージの数は、エージェント数の2乗に比例する。このため図4の台数効果グラフにおいて、PE数が増加した場合の効果が鈍っていると考えられる。一方、非同期式の場合には、交渉の初期にはメッセージ数はエージェント数の2乗に比例するが、その後減少する。このためエージェント数の増加に伴うメッセージ通信のオーバーヘッドに関しては、同期式よりも有利である。

6 おわりに

並列処理に向けたアルゴリズムとして、C-ネットを並列化する二つの方式を提案した。両方式とも並列マシンとの親和性は高いが、同期式協調方式は、メッセージ数の点で非同期式協調方式に劣る。しかし、非同期式協調方式は非決定的処理のため解の再現性が低い。今後、このような非決定的な処理を含む並列処理を評価するための基準を明確にしなければならない。

最後に、有益な助言を頂いた、ICOT第7研究室の新田室長、市吉室長代理を始め、研究員の皆様に感謝致します。

【参考文献】

- [1] Davis, R. and Smith, R.G. "Negotiation as a metaphor for distributed problem solving", Artificial Intelligence 20, North-Holland, pp.63-109 (1983)
- [2] 第7回第五世代コンピュータに関するシンポジウム, "第五世代コンピュータ研究開発の概要", (財)新世代コンピュータ技術開発機構 (1989.6)
- [3] Shindo, S. and Sumida, M. "Group problem solving for task allocation problem", Proceedings of the IASTED International Conference Modelling and Simulation, ACTA PRESS, p.59-63, (1991.7)