エクスターナルグリッドにおける 各種先行処理手法の定量的比較

大西 伊吹[1] 遠藤 慶一[1] 小林 真也[1]

概要：エクスターナルグリッドは，インターネット上に存在する不特定多数の計算機でグリッドを構成し，分散処理を行う技術である．しかし，グリッドを構成する計算機に悪意を持った人間の計算機が紛れ込めば，処理内容の漏洩や処理結果の改ざんに繋がってしまう．それらの問題を解決するため，セキュアプロセッシングの研究が行われてきた．また，セキュアプロセッシングに伴う処理時間の増加という問題について，先行処理を行うことで解決を図ってきた．先行研究にて，3 種類の先行処理手法が，これまでに提案されているが，それらは定量的に比較されてはいない．そこで本研究では，グリッドの管理者が処理目的に応じて適切な先行処理手法を選択できるようにするため，信頼性，高速性，機密性の 3 つの観点から各手法を定量的に評価し，比較する．

Quantitative comparison of various advanced processing methods in the external grid

IBUKI ONISHI[1] KEIICHI ENDO[1] SHINYA KOBAYASHI[1]

1. はじめに

グリッドコンピューティングとはネットワーク上に存在する計算資源を統合し，分散処理することで，高性能な処理能力を得ることのできる技術である．グリッドコンピューティングの一種であるエクスターナルグリッドは，インターネット上に存在する不特定多数の計算機でグリッドを構成するため，悪意を持った人間の所有する計算機 (以下悪人という) であれば，不正行為を働く可能性がある．

不正行為には，処理内容が不正に取得されることによる「処理内容の漏洩」と，処理の委託先が正しい結果を返さないことによる「処理結果の改ざん」が考えられる．これらの問題を解決するため，セキュアプロセッシングの研究が行われてきた．しかし，セキュアプロセッシングは，処理速度の低下を伴う．この問題に対しては，先行処理という手法の研究が行われている．

先行処理には，網羅法，閾値網羅法，閾値暫定法の三種類の手法があるが，それぞれ，処理内容の漏洩への耐性 (機密性)，処理結果の改竄に対する耐性 (信頼性)，処理速度性能 (高速性) が異なる．しかし，現状ではそれらの定量的な比較は行われていない．

本研究では，先行処理の三種の手法を機密性，信頼性，高速性の観点から定量的に比較し，グリッド管理者が処理目的に合わせた先行処理手法を選択できるようにすることを目指す．

2. セキュアプロセッシング

セキュアプロセッシングとは，エクスターナルグリッドにおける悪人の不正行為対策の総称である．セキュアプロセッシングには，処理内容の不正な取得への対策である「プログラム分割」や処理結果の改ざんへの対策である「処理の多重化」，その両方に効果のある「チェックコードの挿入」などが存在する．

本稿では，複数の処理ノードでの並列処理を行うためには必須となるプログラム分割と先行処理を行うためには必須となる処理の多重化のみを採用したグリッドコンピューティングを取り扱う．以下にこの 2 点に関する説明を記述する．

[1] 愛媛大学大学院理工学研究科

2.1 プログラム分割

プログラム分割は，処理内容の漏洩への対策である．依頼するプログラムを複数のプログラム (以下，プログラム断片と呼ぶ) に分割し，異なる計算機に処理を依頼する．これにより，委託するプログラム一つあたりの情報量を減らし，不正な解析を抑制することができる．

2.2 処理の多重化

処理の多重化は，処理結果の改竄への対策である．処理を一つのノードに依頼するのではなく，複数のノードへ同一の処理を依頼し，全体の過半数以上の等しい結果のみを最終的な結果として確定することで，信頼性を確保することができる．過全ての処理結果が過半数に満たない場合は，新たな処理ノードに再度処理を依頼する．

処理の多重化を行った場合，処理ノードの結果を改ざんには，過半数以上の「等しく誤った結果」を集める必要がある．そのため，悪人が処理結果の改ざんを行うことが難しくなる．また，マシントラブルなどにより意図せず誤った処理結果を返してしまった場合の影響を抑える効果も期待できる．

3. 先行処理

処理の多重化は，計算機を複数使用するため，処理ノードの性能差により投票待ちが発生する．これは，信頼性を向上させるというメリットの一方で，処理時間が増加してしまうというデメリットが生じる．このデメリットを解決するため，多重化において過半数の処理ノードから処理結果が帰ってくる前に，得られた処理結果を基に暫定的に次の処理を開始する先行処理という手法が提案されている．

先行処理には，網羅法，閾値暫定法，閾値網羅法の 3 つが存在する．

3.1 網羅法

この手法では，処理を依頼したノードが返した結果のうち，先行処理をまだ開始していない結果全てに対して並列に先行処理を行う方法である．

高速な処理が期待できる反面，参加する処理ノードの数が非常に多くなるため，プログラム断片を不正に取得される可能性が高くなる．

3.2 閾値網羅法

この手法では，処理を依頼したノードが返した結果のうち，一定数同じ結果が返ってきたもの全てに対して並列に先行処理を行う方法である．この一定数のことを暫定閾値と呼ぶ．

網羅法での過剰な処理ノードの参加を抑える狙いがある．

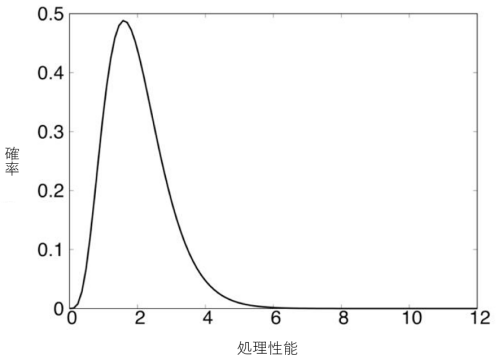

図 1 処理に参加する計算機の処理性能の確率分布図

3.3 閾値暫定法

この手法では，処理を依頼したノードが返した結果のうち，最も早く暫定閾値と等しい数の結果が返ってきた 1 つのみを先行処理する方法である．

閾値網羅法以上に参加するノード数を抑えることができるが，先行処理の結果が最終的な結果とは違っていた場合は，処理速度を向上させる効果が無い．

4. 各指標の概念

本研究で定量的な評価を行う対象は，処理の多重化，プログラム分割，先行処理を取り入れたエクスターナルグリッドである．また，グリッドに参加した悪人は，全員が不正確な同一の結果を返し，処理結果を改ざんしようとするものとする．

シミュレーションを行う際の計算機の処理性能は，形状尺度 $k = 5$，尺度分母 $\theta = 2/5$，期待値 2 のガンマ分布に基づくものとする．計算機の処理性能の確率分布図を図 1 に示す．処理性能 X の計算機は，プログラム全体を $1/X$ の時間で処理することができる計算性能を持つとして取り扱う．

4.1 信頼性

プログラム全体の結果が正しくなる確率は，全ての部分プログラムが正しい結果を返す確率と等しい．

ある部分プログラム a が正しい結果を返す確率 P_a は，式 (1) で表せる．ただし，式中の p は処理ノードが正しい結果を返す確率 (真正処理率)，m は多重度，V_k は多数決が決定する値 (確定閾値) とする．

$$P_a = \sum_{n=0}^{m-V_k} {}_mC_{m-n} p^{m-n} (1-p)^n \tag{1}$$

プログラム分割数を n とするとき，プログラム全体の結果が正しくなる確率 P は，

$$P = P_a^n \tag{2}$$

と表せる．

この P を，信頼性の定量的評価の指標として用いる．

4.2 高速性

グリッドが処理を開始してから，プログラム全体の処理の終了までにかかった時間を，定量的評価の指標に用いる．このとき，通信によるオーバーヘッドは，先行処理手法やパラメータの違いによって変化せず，環境への依存度も高いことから考慮しないこととする．

4.3 機密性

プログラム全体のうち悪人に取得されたプログラム断片の割合 (以後，被取得割合と呼ぶ）を定量的評価の指標に用いる．

5. 結果と考察

先行処理を行わない場合と，網羅法，閾値網羅法，閾値暫定法を定量的に比較した結果を，信頼性，高速性，機密性のそれぞれの観点から示す．比較に当たって，閾値網羅法と閾値暫定法についてはそれぞれ暫定閾値が小さい場合（全体の 1 割）と大きい場合（全体の 4 割）の 2 種類を含め，6 種類を比較する (以後，閾値網羅法 (1 割)，閾値網羅法（4 割）のように記述する)．また，比較結果に対して，ウィルコクソンの符号付き順位和検定を用いて，それぞれの有意差の有無を検証する．

5.1 信頼性の結果

信頼性は，4.1 の式 (2) を指標として用いる．これは 4.1 の式 (1) に依存する．式 (1) は真正処理率 p，多重度 m，確定閾値 V_k で表される式である．しかし，p，m，V_K は先行処理手法の違いによって変化する値ではない．そのため，式 (2) の値も，先行処理手法の違いによっては変化しない．

このことから，信頼性は，先行処理手法の違いによっては変化しないことが分かる．

5.2 高速性の結果

高速性は，処理を開始してからプログラム全体の処理の終了までに必要とした時間を用いて評価を行う．

図 2 は，悪人の存在確率 5%，多重度 20 で，分割数を変化させた際の処理時間の変化を表したグラフである．また，図 3 は，悪人の存在確率 5%，分割数 20 で，多重度を変化させた際の処理時間の変化を表したグラフである．図 4 は，分割数 20，多重度 20 で，悪人の存在確率を変化させた際の処理時間の変化を表したグラフである．

いずれの値を変化させた場合であっても，先行処理を行わない場合は，いずれの先行処理手法を用いた場合よりも処理時間が長くなっており，処理時間の長さは以下のような順となっている．

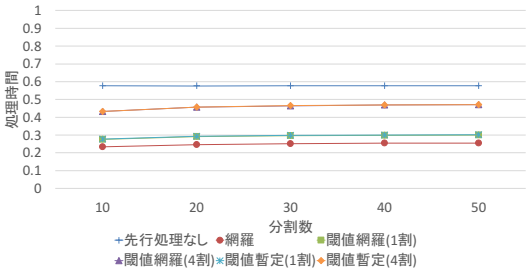

図 2 分割数を変化させた際の処理時間の変化

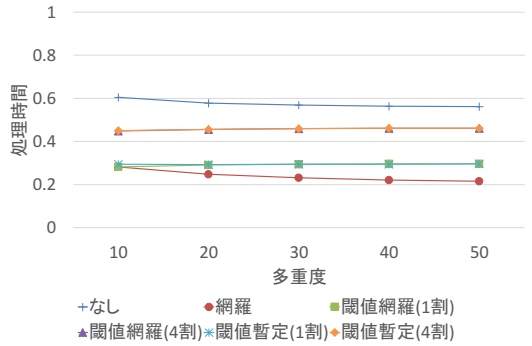

図 3 多重度を変化させた際の処理時間の変化

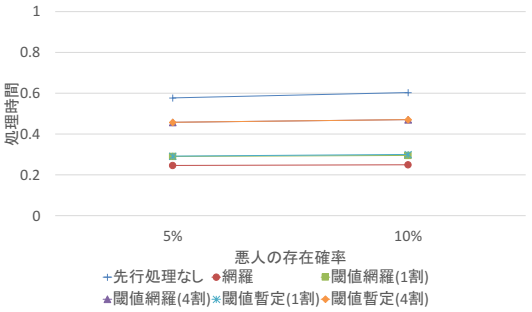

図 4 悪人の存在確率を変化させた際の処理時間の変化

$$T_{mo} < T_{sm1} = T_{sz1} < T_{sm4} = T_{sz4} < T_{no}$$

T_{mo} : 網羅法

T_{sm1} : 閾値網羅法 (1 割)

T_{sz1} : 閾値暫定法 (1 割)

T_{sm4} : 閾値網羅法 (4 割)

T_{sz4} : 閾値暫定法 (4 割)

T_{no} : 先行処理なし

また，どの先行処理手法も，分割数の変化によって処理時間は変化していないことが分かる．図 3 において，網羅法，閾値網羅法（ 1 割）が同様の値を示しているが，これは，多重度 10 の場合において網羅法と閾値網羅法の処理内容が全く同じとなるためである．

それぞれの先行処理手法について，分割数 30，多重度 30，悪人の存在確率 5%の場合について，ウィルコクソンの符号付き順位検定を用いて p 値を算出した．$p < 0.05$ であれば，2 つの検定結果について有意差が存在すると言え

表 1　高速性に関する検定結果

	mo	sm1	sm4	sz1	sz4
no	3.33E-165	0.0113	0.0293	3.32E-165	3.32E-165
mo	-	0.842	0.106	2.28E-165	2.28E-165
sm1	-	-	1.54E-16	0.262	2.28E-164
sm4	-	-	-	2.29E-165	0.808
sz1	-	-	-	-	2.98E-08

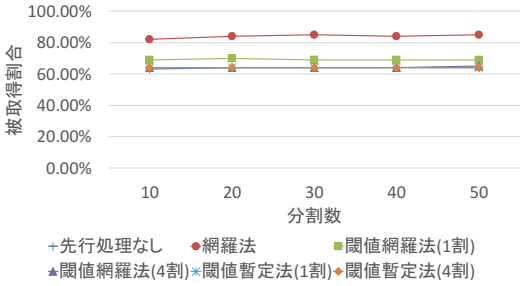

図 5　分割数を変化させた際の被取得割合の変化

る．なお，p 値が 0.05 より大きい場合であっても，有意差が存在しないとは限らない．

それぞれの先行処理手法に関してウィルコクソン符号付き順位和検定を用いて p 値を算出した結果を表 1 に示す．ただし，先行処理なしを no，網羅法を mo，閾値網羅法を暫定閾値に合わせて sm1,sm4，閾値暫定法を暫定閾値に合わせて sz1,sz4 と表記する．

表 1 より，有意差が存在すると示されたのは，以下の組である．

- 「先行処理なし」と「網羅法」
- 「先行処理なし」と「閾値暫定法 (1,4 割)」
- 「網羅法」と「閾値暫定法 (1,4 割)」
- 「閾値網羅法 (1 割)」と「閾値網羅法 (4 割)」
- 「閾値網羅法 (1 割)」と「閾値暫定法 (4 割)」
- 「閾値網羅法 (4 割)」と「閾値暫定法 (1 割)」
- 「閾値暫定法 (1 割)」と「閾値暫定法 (4 割)」

5.3　機密性の結果

機密性は，非取得割合を用いて評価を行う．

図 5 は悪人の存在確率 5%，多重度 20 で分割数を変化させた際の被取得割合の変化を表したグラフである．また，図 6 は悪人の存在確率 5%，分割数 20 で多重度を変化させた際の被取得割合の変化を表したグラフである．図 7 は，分割数 20，多重度 20 で，悪人の存在確率を変化させた際の被取得割合の変化を表したグラフである．

図 5 より，分割数を変化させても，被取得割合に大きな変化はない．また，被取得割合の大きさは，小さい方から順に以下のようになっている．

$$C_{no} = C_{sz1} = C_{sm1} = C_{sz4} < C_{sm4} < C_{mo}$$

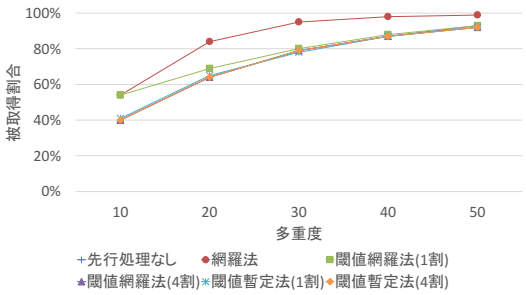

図 6　多重度を変化させた際の被取得割合の変化

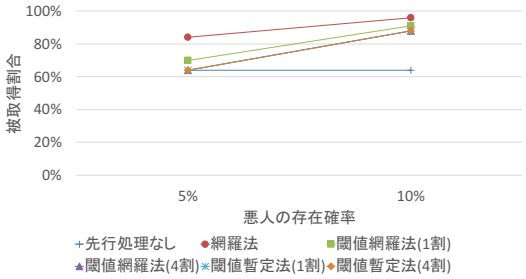

図 7　悪人の存在確率を変化させた際の被取得割合の変化

C_{mo} : 網羅法

C_{sm1} : 閾値網羅法 (1 割)

C_{sz1} : 閾値暫定法 (1 割)

C_{sm4} : 閾値網羅法 (4 割)

C_{sz4} : 閾値暫定法 (4 割)

C_{no} : 先行処理なし

また，図 6 より，多重度を大きくすると，被取得割合が大きくなり，機密性が低下することが分かる．これは，同じプログラム断片を処理するノードの数が増えたことにより，処理ノードの中に悪人が存在する可能性が高まったためであり，閾値暫定法においては先行研究 [1] でも同様の結果が報告されている．

高速性の際と同様，多重度 10 においては，網羅法と閾値網羅法の処理は等しくなるため，同じ被取得割合を示している．また，多重度 30 以上の場合においては，網羅法を除いてほぼ同様の被取得割合を示している．

図 7 より，悪人の存在確率が大きくなると，先行処理を行わない場合を除いて被取得割合が大きくなる．一方，先行処理を行わない場合は，被取得割合は変化しない．

それぞれの先行処理手法について，分割数 30，多重度 30，悪人の存在確率 5%の場合について，ウィルコクソンの符号付き順位検定を用いて p 値を算出した．$p < 0.05$ であれば，2 つの検定結果について有意差が存在すると言える．なお，p 値が 0.05 より大きい場合であっても，有意差が存在しないとは限らない．

それぞれの先行処理手法に関してウィルコクソン符号付き順位和検定を用いて p 値を算出した結果を表 2 に示す．

表 2　機密性に関する検定結果

	mo	sm1	sm4	sz1	sz4
no	1.67E-148	1.29E-25	0.367	0.548	0.68
mo	-	1.35E-26	1.35E-26	2.38E-165	2.38E-165
sm1	-	-	0.637	0.457	0.234
sm4	-	-	-	0.658	0.721
sz1	-	-	-	-	0.865

ただし，先行処理なしを no，網羅法を mo，閾値網羅法を暫定閾値に合わせて sm1,sm4，閾値暫定法を暫定閾値に合わせて sz1,sz4 と表記する．

表 2 より，有意差が存在すると示されたのは，以下の組である．

- 「先行処理なし」と「網羅法」
- 「先行処理なし」と「閾値網羅法 (1 割)」
- 「網羅法」と「閾値網羅法 (1,4 割)」
- 「網羅法」と「閾値暫定法 (1,4 割)」

5.4　考察

信頼性，高速性，機密性に関する比較結果よりエクスターナルグリッドの処理内容や処理目的によって適する先行処理手法が異なることが分かる．

高速性が重視される処理を行う場合は，網羅法を利用することで高速に処理を行うことができる．一方で，機密性が他の先行処理手法よりも低い点を考慮する必要がある．すでに広く知られている情報を用いて大規模な計算を行うような場合に網羅法は有効であると考えられる．

機密性を重視する場合は，被取得割合を低く抑える必要がある．何らかの先行処理を行った場合，悪人の存在確率によって，被取得割合は大きく変化する．一方，先行処理を利用しない場合は，悪人の存在確率による被取得割合は変化しない．通常，グリッド管理者は悪人がどの程度存在するかを特定することはできない．そのため，悪人が確実に少ないということが断定できない場合においては，機密性を重視する場合は，現在提案されている先行処理手法は利用すべきではないと言える．

悪人が確実に少ないと断定できる場合は，他の手法よりも機密性の低い網羅法を除いた 4 手法が候補である．この内，最も高速性に優れた手法は閾値暫定法 (1 割) であった．このことから，低い暫定閾値での閾値暫定法を用いるべきであると言える．

6.　まとめ

本稿では，グリッドコンピューティングにおいて，グリッド管理者が処理目的に合わせた先行処理手法を選択することができないという問題点を解決するため，先行処理手法を行わない場合，網羅法，閾値暫定法，閾値網羅法を使った場合を，高速性，機密性，信頼性の 3 つの観点から定量的に比較した．信頼性に関しては先行処理手法によって変化しないことがわかった．高速性を重視する処理内容の場合では，網羅法を用いることが有効である．しかし，網羅法は機密性が低いので，一般に公開されている情報を用いた大規模計算など，情報や処理内容が知られても構わない場合に利用するのが良い．機密性に関しては，いずれかの先行処理を利用した場合, 悪人の存在確率の違いによって，機密性が大きく変化する．しかし，グリッドの管理者が悪人がどの程度存在するかを知るすべはないため，先行処理を利用した場合に機密性が意図せず低下してしまう可能性がある．そのため，確実に悪人が少ないとわかっている場合以外は先行処理を利用すべきではないと言える．

今後の予定として，グラフ上では差があるにもかかわらず，検定では有意差が確認できなかった手法の組み合わせについての詳細な検証と，グラフ上で同様の値を示した手法の組み合わせについて，差がないことの検証が挙げられる．

参考文献

[1] 田中祐生・遠藤慶一・樋上喜信・小林真也 (2017)，“閾値暫定法を用いたエクスターナルグリッドにおける高速性・機密性・信頼性のトレードオフ関係の定量的考察”，情報処理学会第 79 回全国大会講演論文集

[2] 山口晃右・藤橋卓也・遠藤慶一・小林真也 (2017)，“エクスターナルグリッドにおける網羅法の処理ノード数増加に対する抑制手法の提案”，マルチメディア，分散，協調とモバイルシンポジウム論文集 (DICOMO 2017)

[3] 富田航平・藤橋卓也・遠藤慶一・小林真也 (2018)，“閾値網羅法における機密性・高速性を考慮した閾値決定法”，情報処理学会第 80 回全国大会公演論文集