

早稲田大学大学院国際情報通信研究科

博士論文審査報告書

論 文 題 目

Checkpoint Systems
in P2P-Grid Environment

P2P グリッド環境における
チェックポイントシステム

申 請 者

Huan WANG

国際情報通信学専攻
分散コンピューティングシステム研究Ⅱ

2016年 2月

大きな計算能力を必要とする情報処理や負荷が変動する情報処理に対して、遠隔の計算資源を活用するグリッドコンピューティング（以下、グリッド）やクラウドコンピューティング（以下、クラウド）といったコンピュータ資源の利用形態がある。一般ユーザのコンピュータ資源（以下、計算資源）をインターネットにより相互接続することにより、グリッドやクラウドを実現することが可能である。このような構成は P2P グリッドと呼ばれる。

P2P グリッドを構成する計算資源は、一般のパーソナルコンピュータであり、P2P グリッドへの参加／離脱は、その利用者の意思による。したがって、任意のタイミングで、システムからの計算資源の離脱が発生する。この事象に対応するために、一般に P2P グリッドでは、同一の処理（以下、各計算資源で実行される処理をタスクと呼ぶ）を複数の計算資源に同時に割り当て、並行してタスクを実行する。これにより一部の計算資源がシステムから離脱しても、同じタスクを実行する計算資源が残ることによりタスクを完了させる仕組みとなっている。

このような空間的冗長性により計算資源の離脱に対処する仕組みでは、一定の計算資源量のもとで処理可能な仕事量が減少する。多くの資源を必要とする情報処理で利用され、また多くの利用者により供用される P2P グリッドでは、同じ計算資源量で提供できる仕事量が限定されるのは好ましくない。

計算資源の離脱に対処するもう一つの方法として、時間的冗長性を利用し、離脱発生後に同一タスクを他の計算資源に割り当て、再実行することで離脱により消失した処理を回復する方法がある。この手法では、離脱の発生によりタスク処理完了が遅延することになる。割り当てた処理量が大きく、処理完了までの時間が長い場合、処理中に離脱が発生する確率が高まり、タスク処理完了の遅延が大きくなり好ましくない。

この遅延を軽減するために、処理の途中経過を記憶装置に保存することにより、離脱が発生しても、その途中経過から処理を再開できるチェックポイントシステムが有効である。チェックポイントシステムを利用した場合、処理資源の離脱により再実行しなくてはならない情報処理量は、チェックポイントを実施した時点から離脱の起こった時点までである。したがって、タスクの最初からの再実行を必要とせず、遅延を削減することができる。ただし、チェックポイントを実施すること自体が、タスク処理に対するオーバーヘッドとなるため、その影響を最低限に留めるためには適切な設定が必要となる。

本論文は、P2P グリッドにおけるチェックポイントシステムについて、申請者が博士後期課程の 5 年間およびその後助手としての 2 年間に行った研究成果をまとめたものである。なお、本文は英語で書かれている。

以下、各章毎に概要を述べ、評価を加える。

第 1 章「Introduction」では、本研究の背景、目的、および論文の概要について述べている。

第 2 章「Checkpointing Interval Calculation」では、最適なチェックポイ

ント実施周期を求めている。タスク実行の途中経過をチェックポイントとして保存する際に、保存する情報の一貫性確保のため、タスクの実行を一時的に停止する。よって、頻繁なチェックポイントの実施は、タスクの実行完了を遅延させることになる。一方、チェックポイント実施周期が長くなると、離脱が発生した際に損失し、再度実行しなければならない処理が増加し、結果的にタスク完了が遅延する。以上から、タスク完了の遅延を最小にするチェックポイント実施周期が存在することがわかる。本章では、離脱発生確率、チェックポイントによる遅延、消失した処理回復のための時間などからタスク実行完了時間の期待値を定式化し、最適なチェックポイント実施周期を解析的に求めている。また、シミュレーションにより、解析の妥当性も検証している。チェックポイントの最適な実施周期を解析的に求めることに成功したことは、今後チェックポイントシステムを実用する上でも有用である。

第3章「A Checkpointing System on P2P Grid Environment」では、P2Pグリッドにおける、チェックポイントシステムの構成法を提案している。まず、本論文で提案するチェックポイントシステムの3つの構成要素、Working Peer (WP), Working Candidate Peer (WCP), Checkpointing Peer (CPP)とその役割を定義している。さらに、P2Pグリッド内に存在する計算資源の中から、各構成要素を選定する基準を定義している。チェックポイントシステムは、第2章で求めたチェックポイント実施周期を基に、構成要素の個数を決定し、上記選定基準にしたがって各構成要素の役割を果たす計算資源を選定して構成される。離脱が発生する環境下では、提案の構成法により、先行研究と比べてタスク実行完了が早くなることが示された。チェックポイントシステムの構成として考えられる様々なバリエーションの中から有効な構成法が示され、システム構成の道筋が示された意義は評価できる。

第4章「Checkpointing Implementation in Virtual Environment」では、仮想マシン環境でのチェックポイントシステム実装方法を提案し、それを実装し、評価している。データセンタやクラウドは、その管理の容易性から、計算資源を仮想マシンとして実装する。P2Pグリッドの計算資源も、同様な理由により仮想マシンとして実装される可能性が高い。仮想マシン環境では、仮想マシン全体をデータとして記憶装置に保存することが可能であり、その仕組みによってチェックポイントを実現することができる。しかし、この方法ではチェックポイントとして保存するデータ量が多く、P2Pグリッドのように、チェックポイントデータをネットワーク経由で移動させるには不向きである。本章では、仮想マシンの管理機構であるハイパーバイザによってチェックポイントを実施することにより、P2Pグリッドで実行するタスクに変更を加えることなく、対象となるタスクのみについてチェックポイントを実施する方法を提案している。さらに、第3章で述べたチェックポイントシステム構成法を含めた提案手法を実装することにより、その実現性を示している。仮想マシン全体としてチェックポイントを実施するのに比べ、大幅なデ

一夕量の削減が可能であることが示されている。本論文で提案されたチェックポイントシステムが実装可能であり、その有効性が示されたことにより、提案手法の実利用が可能になったことの意義は大きい。

第5章「Failure Probe in P2P-Grid System」では、計算資源離脱の検出方法を提案している。チェックポイントシステムによって計算資源の離脱から回復するためには、まず離脱を検出しなければならない。離脱の発生からできるだけ短時間でそれを検出することが、タスク完了を遅延させないためにも重要である。本論文で提案の三つの構成要素である WP, WCP, CPP のそれぞれで離脱検出を行う仕組みを提案し、それらを第4章で実装したチェックポイントシステムに組み込み、先行研究に提案された方式と比較して評価を行った。その結果、提案手法の WP による検出が最も短時間で離脱を検出できることが示された。本章で述べられている計算資源の離脱検出機構を併せることにより、本論文でここまで述べられてきたチェックポイントシステムが完結したものとなり、実用上意義がある。

第6章「Conclusion」は、本論文での議論をまとめ、今後の研究方向について述べている。

以上要するに本論文は、一般ユーザの計算資源をインターネット接続することによって大きな情報処理能力を実現する P2P グリッドにおいて、そこで頻繁に発生する計算資源のシステムからの離脱に対処する機構の提案を行ったものである。P2P グリッドが保有する資源を有効に活用するため、時間的冗長性を利用したチェックポイントシステムによって離脱に対処しながら、タスク完了遅延を最小に抑えることができる機構を提案し、それを仮想マシン環境で実装し、その実現可能性と有効性を示した。この成果は、P2P グリッドばかりでなく、現在仮想マシンで実現されているデータセンタや世界的な広がりを見せるクラウドコンピューティングにも応用することが可能であり、世界的な規模で実社会に貢献する成果である。よって本論文は、博士（国際情報通信学）の学位を授与するに値するものと認める。

2016年2月18日

審査員

主任 早稲田大学教授（専門：情報通信ネットワーク）

Ph.D.（イリノイ大学）

中里秀則

中里 秀則

早稲田大学教授（専門：情報通信工学）

工学博士（早稲田大学）

亀山 渉

亀山 渉

早稲田大学大学院客員教授（専門：情報通信ネットワーク）

工学博士（東京大学）

津田俊隆

津田 俊隆

早稲田大学教授（専門：情報通信工学）

工学博士（北海道大学）

渡辺 裕

渡辺 裕