

## GPU を活用した高性能トランザクション処理のための依存グラフ構築と競合の解析 Dependency Graph Construction and Conflict Analysis for High-Performance Transaction Processing with GPU Acceleration

千代丸怜央<sup>†</sup>      塩井隆円<sup>‡</sup>      宮崎純<sup>†</sup>  
Reo Chiyomaru      Takamitsu Shioi      Jun Miyazaki

### 1. はじめに

データベースシステムにおいてデータの整合性を保証するシステムは不可欠である。複数のユーザーが複数データにアクセスする際に競合が発生し、これによりデータの異常 (Anomaly) が引き起こされる可能性がある。これを防ぐために最も厳格な手法が直列化可能性 (Serializability) である。Serializability は、複数のトランザクションが並行実行された結果が、あたかもそれらが何らかの順序で直列に実行されたかのような結果と等価であることを保証する。

近年のデータベースシステムの主流である Multi-Version Concurrency Control (MVCC) は、データを上書きする際に新しいバージョンを追加し、古いバージョンを読めるように残すことで、並行実行の効率を高める。MVCC 環境では、トランザクション間で read/write したデータの競合を依存関係 (write-read (wr)、write-write (ww)、read-write (rw)) として構築される依存グラフ (Direct Serialization Graph: DSG) [1] に循環 (サイクル) が存在しない場合に Serializability が保証される。

Serializability を保証するためには、トランザクションが依存グラフ上のサイクルを作らないかを判定する必要がある。しかし、現在の手法には以下の問題が存在する。

1. **グラフ探索の計算負荷**: そもそもサイクルを探すようなグラフ探索が大変である。特に大量のトランザクションが関わる場合、同じデータの複数バージョンに対する read/write 競合を依存グラフとして構築し、サイクルを検出する計算コストが高く、性能上のボトルネックとなる。
2. **待ち時間と Abort による性能低下**: その間、トランザクションが終了できずに待たされる。しかもサイクルが作られた場合は、そのサイクルを形成したトランザクション群のいずれか一つを Abort する。この待ち時間と Abort したときの再実行によって、システム全体の性能が低下する。

本研究では、これらの問題に対して以下の解決策を提案する。

1. **GPU によるグラフ構築・探索の高速化**: GPU の並列処理能力を活用して、各トランザクション間のデータ競合からグラフを構築・探索する処理を高速化する。
2. **RSS による Abort の回避**: Read-Safe Snapshot (RSS) [2] という手法を使って、トランザクションを Abort させることなく Serializability を保証する。

本研究は、GPU を用いた依存グラフ構築・解析の高速化により、RSS のような高度な整合性保証手法を実用的な性能で実現することを目指す。

### 2. 関連研究

グラフ理論におけるサイクルの検出は基本的な問題であり、様々なアルゴリズムが提案されている。Johnson のアルゴリズムは効率的な逐次アルゴリズムの一つだが、大規模グラフではメモリや時間的な限界がある。並列アルゴリズムも研究されており、Tarjan の DFS ベースのアルゴリズムが並列化が難しいとされる一方で、BFS や Coloring を活用した並列 Strongly Connected Components (SCC) 検出手法などが提案されている。また、GPU を用いたグラフ処理やサイクルの検出の高速化に関する研究も行われている [3]。

本研究は、データベーストランザクションの実行履歴から生成される依存グラフという特殊なグラフの構築と解析を、GPU を活用してリアルタイムで処理する点で従来研究と差別化される。

### 3. 提案手法

本研究の目的は、依存グラフ構築・解析処理を GPU により高速化することである。

依存グラフは、トランザクション間の依存関係をエッジとして持つ有向グラフである。RSS は、読み出し専用トランザクションがサイクルを作らずに読むことができる最新バージョンのデータの集合である。RSS の構築に必要な依存関係は、ww や wr のように read/write 時に自トランザクションが新バージョンのデータとして読み書きする際に発生するリアルタイムで追跡しやすい依存関係に加えて、rw のような他のトランザクションが新

<sup>†</sup> 東京科学大学 Institute of Science Tokyo, Tokyo, Japan

<sup>‡</sup> 日本電気株式会社 NEC Corporation, Tokyo, Japan

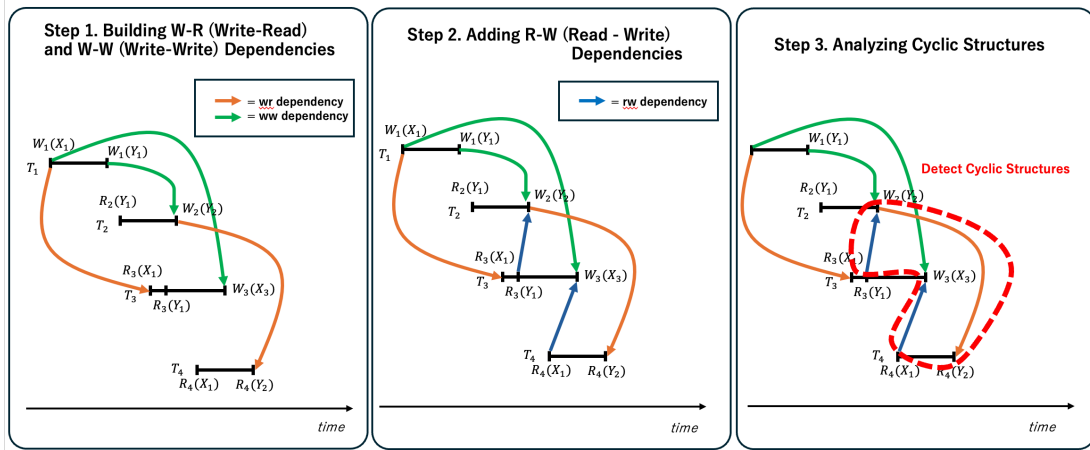


図1 提案手法の概要

バージョンを書いた際に発生するリアルタイムでの把握が難しいものも含まれる。したがって、リアルタイムにデータベースで実行したトランザクションの実行履歴から ww と wr の依存関係情報を収集し、その後から rw 依存関係を構築する必要がある。その後、すべての依存関係を用いて依存グラフを作成し、SCC を見つけることでサイクルを検出する。処理の概要は図1の通りである。

1. **ww/wr 依存関係の構築:** トランザクションの実行に伴う ww、wr 依存関係情報から有向グラフを作成
2. **rw 依存関係の構築:** 1 で得た情報から、オフラインまたは準リアルタイムで rw 依存関係を構築
3. **依存グラフの構築とサイクルの発見:** 1 と 2 の依存関係から依存グラフを構築し、GPU 上でサイクルが存在しないか探索する。サイクルを見つけた場合は、サイクルを回避するように読むことで、どのトランザクションも Abort されずに Serializability が保証される。

GPU を活用することで、上記のステップの処理時間を大幅に短縮することを目指す。これにより、RSS のような依存関係の厳密な解析に基づく整合性保証手法を、低遅延で実行することが可能となる。

また、本手法は分散データベース環境において特に有効である。従来の手法では、分散システム間での Serializability 保証には高い通信コストが伴い、Abort が発生した場合は他ノードとの通信による再調整が必要となる。本手法により高速に RSS を構築することで、読み出し専用トランザクションが安全に読めるデータを待つ時間が短縮され、低遅延なデータ整合性保証と他のトランザクションへの影響の緩和が可能となる。HTAP 環境や分散 OLTP システムにおいても、読み出し専用トランザクションの整合性保証のための通信コストの排除だけでなく、書き込みトランザクションにおいてリアルタ

イムでサイクルを検出し、サイクルを回避するか Abort を実行することで Serializability を保証する手法への適用も可能となる。

#### 4. おわりに

本稿では、データベーストランザクションの Serializability 保証において、依存グラフの構築・解析を GPU で高速化する手法を提案した。GPU の並列処理能力を依存関係構築後のグラフ構築・解析 (サイクルの検出) に適用することで、既存手法である CPU 処理と比較して大幅な性能向上ならびにリアルタイム性を実現することが期待される。

今後は、GPU 上でのアルゴリズム詳細設計・実装、各種ワークロードでの性能評価、データベースシステムとの連携方法などが重要な課題となる。

#### 謝辞

この成果は、国立研究開発法人新エネルギー・産業技術総合開発機構 (NEDO) の助成事業 (JPNP16007) の結果得られたものである。

#### 参考文献

- [1] Adya, A., Liskov, B. and O'Neil, P.: Generalized isolation level definitions, *Proceedings of 16th International Conference on Data Engineering (Cat. No.00CB37073)*, pp. 67–78 (2000).
- [2] Shioi, T., Kambayashi, T., Arakawa, S., Kurosawa, R., Hikida, S. and Yokota, H.: Read-safe snapshots: An abort/wait-free serializable read method for read-only transactions on mixed OLTP/OLAP workloads, *Information Systems*, Vol. 124, p. 102385 (2024).
- [3] Benachour, A., Yahiaoui, S., El Baz, D., Nouali-Taboudjemat, N. and Kheddouci, H.: Fast parallel algorithms for finding elementary circuits of a directed graph: a GPU-based approach, *J. Supercomput.*, Vol. 79, No. 5, p. 4791–4819 (2022).