

# 投機的実行におけるコンテキスト予測共有による Ethereum ノード高速化

大場 雄太郎<sup>†</sup>      山崎 憲一<sup>†</sup>  
Yutaro Oba      Kenichi Yamazaki  
<sup>†</sup> 芝浦工業大学 大学院 理工学研究科

## 1. はじめに

Ethereum は、大規模分散型のコンピューティングプラットフォームである。Ethereum における取引はスマートコントラクトによって記述され、ブロードキャストによって各ノードに伝播される。その後、ブロックチェーンに追加され、各ノードにおいて順次実行される。伝達・合意・実行という三段階で構成される新たな処理パラダイムにおいて、トランザクションが伝達されてから実際に合意・実行されるまでの時間差は、投機的実行を利用してトランザクション処理を高速化する機会となる [1]。

そこで、Forerunner [2] では、複数の未来に基づく制約ベースの投機的実行手法を提案している。本論文では、Forerunner の手法を拡張し、投機的実行結果を得る過程で生成される未来コンテキスト予測を活用することで、処理能力の低いノードにおける実行性能の向上を提案する。具体的には、先行して高速ノードが生成した未来コンテキスト情報を低速ノードと共有し、それを基に低速ノードでも高速化された実行パスを事前に構築・適用可能にすることで、ネットワーク全体における処理のばらつきを抑え、分散システムとしてのスループットの向上を図る。

## 2. 関連研究

### 2.1 Ethereum におけるトランザクション処理手法

Ethereum では各ノードは、外部からブロードキャストされたトランザクションをペンディングプールに蓄積し、マイナーがそれらを選択してブロックを構成する。ブロックはマイナーによって作成され、Proof-of-Stake により合意を形成した後、ネットワーク全体に伝達される。

Ethereum における各ブロックは、トランザクションリストだけでなく、ブロック終了時点の状態ルートも格納している。ブロック作成時には、各トランザクションが順番に Ethereum Virtual Machine (EVM) 上で実行され、状態遷移が適用された後の最終的な状態ルートが計算される。ノードは新たなブロックを受信した際、ブロック内のトランザクションを順に適用し、状態遷移を再現することで状態ルートの検証を行う。

この方法では、トランザクション実行が逐次的に行われる各トランザクションは、前のトランザクションの状

態に依存するため、基本的に並列処理が困難であり、トランザクションスループットの向上には新たな工夫が求められている。

### 2.2 Forerunner

Forerunner [2] は、分散ネットワーク環境における非決定性に対応するため、トランザクション実行時に複数の可能な未来状態を仮定するアプローチを採用している。各未来状態に対して仮想的な EVM 実行を行い、実行中に得られる制御フローやデータ依存性に基づいて「制約」を抽出する。制約とは、将来的な状態においてトランザクション実行が決定的に成立するための前提条件である。この制約に基づき、本番実行時に実際の状態が条件と一致していれば高速な専用パスを適用し、一致しない場合には通常の EVM 実行へフォールバックすることで、Ethereum の正確な動作を維持する。

この手法では、ブロックが確定する前に仮想的な実行を行うため、その実行環境を事前に構築する必要がある。このときに用いられるのが「未来コンテキスト」である。未来コンテキストとは、将来のブロック確定時に想定される状態の集合であり、具体的にはトランザクションの順序、ブロックのタイムスタンプ、アカウント残高、ストレージ値など、状態遷移に関連する要素を含む。これらは仮想的に構成され、制御フローやデータ依存性に基づく制約を導出し、高速実行パスの適用条件として利用される。

そのため、Forerunner における未来コンテキストは投機的実行結果を導くための中間的な情報にすぎない。仮に誤ったコンテキストが伝播された場合でも、制約を満たさなければ高速パスは適用されず、通常の処理に戻ることで Ethereum の正確な動作は維持される。

Ethereum ネットワークでは、各ノードが同一のトランザクションセットを独立して実行する。この方式では、計算能力の低いノードが処理を終えるまで合意形成が遅延し、ネットワーク全体のスループットが低下するという課題がある。この課題に対して本研究では、遅延が発生しがちなノードに対して事前に未来コンテキストを共有することで、計算開始時間を早める方法を提案する。

### 3. 提案手法

本研究では、計算を先行して完了したノードが生成した未来コンテキストを他ノードに共有することで、冗長な計算を回避し、全体の処理効率を向上させる「非対話型ブロードキャスト方式」を提案する。

本方式は、計算の速いノードが未来コンテキストの生成を完了した時点で、ネットワーク内の任意のノードに対して一方的にその結果を送信する。送信先ノードの状態（処理の進捗や必要性）を考慮せず、非同期・非対話的に情報を展開する点が特徴である。これにより、構成の単純さと柔軟な受信処理が可能となる。

本方式における未来コンテキストの生成と共有の全体的な流れを図1に示す。各ノードは、トランザクションプールに新たなトランザクションが追加されると、未来コンテキストの生成を開始する。その間に他ノードから未来コンテキストを受信した場合は、生成中のコンテキストを破棄し、受信したものを採用して自身の計算を進める。一方、受信がなかった場合は、未来コンテキストの生成を完了し、任意の数のノードに対して非対話型で未来コンテキストを送信する。いずれの場合も、最終的には生成された未来コンテキストに基づきブロックが生成される。なお、受信した未来コンテキストが必ずしも自身で計算中のものより良質とは限らず、採用可否の判断基準を設ける必要があることも今後の課題である。

この手法の主な利点として、以下が挙げられる。第一に、通信相手との事前調整や応答を必要としないため、実装が極めてシンプルである。第二に、送信された未来コンテキストを用いるかどうかは各ノードの裁量に任されており、自身の状況に応じて計算負荷を軽減できる柔軟性がある。第三に、未来コンテキストは、先行ノードが仮想的に生成した中間的な実行条件を他ノードでも活用可能であり、それにより各ノードが一部の計算ステップを省略することができる。

一方で、帯域幅が増加するなどの問題もありうるため、処理の遅れたノードが能動的に未来コンテキストを要求する「対話型共有方式」の導入も検討する。この方式では、ノードが自身の遅延を検知した場合に限り、高速ノードに対して明示的にコンテキストの提供を求めることで、通信の最適化が期待される。

### 4. 評価方法

本研究では、提案手法の有効性を検証するため、ブロック生成と伝播のシミュレーションが可能な SimBlock を用いて評価を行う。

主に以下の2点について評価を行う。ブロック処理時間がどれだけ短縮されるかと、未来コンテキストの伝播

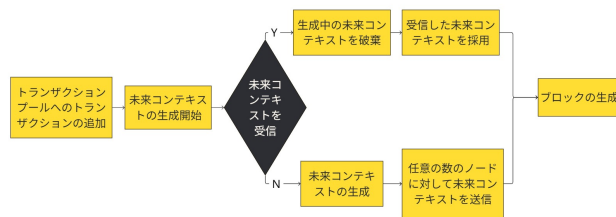


図1. 提案手法の概要

によって生じる通信量の増加である。

### 5. 今後の課題

シミュレーションベースにより提案の有効性を評価することに加え、以下について検討が必要である。

まずセキュリティ面で、未来コンテキストの正当性確認を行わない設計により、虚偽の情報が伝播するリスクが挙げられる。フォールバック処理により整合性は維持されるが、処理頻度が増えれば計算資源の浪費につながるため、抑制手法の検討が必要である。

また、非対話型でのコンテキスト共有において、共有対象の選定や送信データの軽量化が今後の検討課題である。

### 6. おわりに

本研究では、Ethereum ネットワークにおける全体スループットを向上させるため、未来コンテキストの共有に基づく「非対話型ブロードキャスト方式」を提案した。本手法により、計算能力の低いノードでも先行ノードの投機的実行結果を活用することで、処理の高速化が期待できる。

今後は、信頼性・帯域負荷・導入性といった観点からさらに改良を重ね、より実用的な分散実行基盤の実現を目指す。

### 謝辞

本研究にあたり、技術的なご支援を賜りました NTT DOCOMO, Inc. / NTT Digital, Inc. の大月 魁 様に深く感謝申し上げます。

### 参考文献

- [1] Vitalik Buterin, "Ethereum: A Next-Generation Smart Contract and Decentralized Application Platform," 2014.
- [2] Yang Chen, Zhongxin Guo, Runhuai Li, Shuo Chen, Lidong Zhou, Yajin Zhou, and Xian Zhang, "Forerunner: Constraint-based Speculative Transaction Execution for Ethereum," In \*Proceedings of the ACM SIGOPS 28th Symposium on Operating Systems Principles (SOSP '21)\*, October 26–29, 2021, Virtual Event, Germany. ACM, New York, NY, USA, 18 pages.