

## 論文の内容の要旨

論文題目 追い出し不可属性を持つキャッシュアルゴリズムの研究  
(Stubborn Cache: Cache with Non-Replacing Lines)

氏 名 野村 隼人

プロセッサへのデータ供給能力は性能上最も重要な要素の一つであり、これを支えるため近年のプロセッサは大容量のLast Level Cache (LLC)を備えている。キャッシュ容量が大きくなることで、プログラマが意識せずとも得られるプログラムのメモリアクセス性能が良くなっている。しかしながら、大容量化したLLCには多くのデッドブロックが含まれており、その一方で追い出しによるミスが発生していることが知られている。これは、直近の履歴やその学習によって得られるキャッシュラインの保持の選択を行うような従来のアプローチが短期の再参照のヒット率を高めることを指向しているため、特にmemory-intensiveなアプリケーションでは、メガバイトオーダーの容量をもつLLCでも救えないミスが生じている。大容量化したLLCをさらに活用するためには、従来の手法がターゲットとしていたものより再参照までの距離間隔（時間）がより長いキャッシュラインへのアクセスや、より大きいワーキングセットをターゲットとしたキャッシュマネジメントが必要となる。

本研究による調査では、まず、LLCで生じる残されたキャッシュミスの多くを占めるものが再参照ミスであり、また10～100M（メガ）の長期の命令数間隔に渡る再参照であることを明らかにした。これは周期の長いスラッシング、ストリーミングアクセスによってもたらされている。

本研究ではそのようなミスに対処するためのキャッシュマネジメント手法として、Stubborn戦略を提案した。これは、キャッシュラインの追い出しを一時的に凍結し、キャッシュラインの入れ替えを起こさない領域を設けることで、長期保持による統計的なヒット数向上を可能とするような戦略である。このStubborn戦略の最も単純な実装であるNormal Stubbornでは、従来の手法では的確な追い出し選択が困難であった10～100M命令間隔の長期再参照が連続するワークロードや、激しいスラッシングによりキャッシュ容量が全く活用されないワークロードなどに有効であることを確認した。さらに、ワークロードのメモリアクセス傾向を詳細に調べ、モデルと共に例示することで

Stubborn戦略が有効であったワークロードにおいて、キャッシュラインの凍結という一見非効率なStubborn戦略が何故LLCにおいて有効に機能するかを明らかにした。

Stubborn戦略は、従来の置き換えアルゴリズムとハイブリッドに実装され、キャッシュの一部の置き換えをアクセス履歴に関係なく凍結する。この性質から、もし不要なキャッシュラインをStubborn戦略で取り込んでしまった場合、キャッシュの容量効率を純減させることとなる。Normal Stubbornのようなナイーブな実装ではこの問題による副作用が確認された。このため、プログラムの動作傾向、ワークロードに合わせて、どのキャッシュラインを、いつからいつまでStubborn戦略で運用するかを動的に判断する機構の実現がStubborn戦略の実用上の課題となる。

そこで本論文では、Stubborn戦略が多様なワークロードに適用するための条件について調査し、さらなる性能向上と副作用の抑制を実現する方法を明らかにする。この調査では、Stubborn戦略のキャッシュ容量への依存性、フェーズに適応させたときのStubborn戦略の効果、Stubborn領域の静的な増減による効果、全域をStubborn領域としたときの効果について調べ、フェーズ適応による副作用の抑制と、Stubborn領域の増減による性能向上が得られることを確認した。また、それに基づいて実行時ワークロード適応型のStubbornキャッシュマネジメント手法を提案した。

シミュレータ上にこの提案手法Stubborn-DYN, BRRIP triggered Stubborn, Stubborn-HL, Stubborn-HL-Half, Stubborn-HL-Resetを実装した評価では、LRUに対して最大42.3%、幾何平均で3.8%の性能向上を確認した。また、Normal Stubbornと比較した場合、今回提案した適応制御により2.8%の性能向上となり、特に、7.4%性能低下していたワークロードでは性能低下を抑えて0.1%の性能向上が得られるまで改善が見られるなど、性能を安定させる効果が確認できた。