

クラウドコンピューティング基礎論最終課題

工学系研究科航空宇宙専攻

荒居 秀尚

2018 年 7 月 26 日

1 キャッシングによる通信の遅延低減の手法に関して

本稿では、数あるインターネットの遅延低減手法の中でも、接続に関するキャッシュを保存しておくことによって、通信の立ち上げのタイムロス低減を図る取り組み 2 つに関してレビューを行う。2 つの手法はそれぞれ、DNS のプリフェッチと TCP コネクションのキャッシングを用いるもの [1] と、データアクセスのキャッシングと関連データのプリフェッチを用いるもの [2] である。これらに関して順番にレビューを行う。

2 Accelerating Last-Mile Web Performance with Popularity-Based Prefetching

2.1 掲げている課題と提案手法

この論文で掲げられている課題は、クライアント側に最も近い部分で発生する通信の遅延である。筆者らは、この問題を解決するために、DNS のプリフェッチとユーザが頻繁に訪れる Web サイトの TCP コネクションのキャッシングを提案している。DNS のプリフェッチや TCP コネクションの継続は既に一般的に行われているものであるが、これらは接続予測に基づいたものであり、実際のユーザの行動とは乖離している可能性がある。筆者らは、過去のユーザの接続履歴に基づいた DNS プリフェッチや TCP コネクションの継続を提案しそれをシミュレーションにより評価した。

2.2 提案手法の詳細

筆者らが提案した手法は以下の 2 つから構成される。

dnsmasq の増強 ホームネットワークの名前解決履歴に基づいたドメインリストの保持

polipo ユーザが頻繁に訪れるコネクションを保持する HTTP プロキシ

そして保持された DNS リストの中で失効したドメインを更新し、TCP コネクションを維持するために間欠的にダミーの HTTP GET メソッドを送信するヘルパースクリプトを用いて高速化を図る。これらの手法の性能に関わるパラメータは

- ドメインとコネクションのリスト長

- ドメイン保持とコネクション保持のタイムアウト時間

である。これらの最適化をルータに実装することで、ネットワーク内の全てのデバイスに実装を行うことなく最適化をする事ができるほか、ホームネットワーク内の参照履歴に基づいた最適化を図ることができると筆者らは主張する。

2.3 提案手法の評価

筆者らはシミュレータを用いてこれらの提案手法の評価を行った。DNS のプリフェッチに関しては、提案手法を用いない場合に比べ、DNS キャッシュにアクセスする頻度が 15%-35% 程度増加し、名前解決にかかるオーバーヘッドを大きく低減した一方で、世帯ごとのネットワーク利用状況に依存してばらつきが出るとしている。

また、TCP コネクションのキャッシングは、コネクション維持時間を少し取るだけでも大きく TCP コネクションのオーバーヘッドを低減できたとしている。

2.4 提案手法の利点と欠点

提案手法はいくつか利点と欠点を抱えていると思われる。まず、利点としてはユーザの利用履歴を用いた最適化を行うことにより、一般的な予測よりもより個人の行動にフィットした最適化を行うことができると考えられる。また、実装の手間としても大きくないと考えられ、最適化の仕組みを組み込んだルータを利用するだけでユーザはその最適化の恩恵を得ることができると考えられる。

一方で欠点としては、TCP コネクションを維持することでサーバー側にはより負荷をかけることになる。また、大学など大きな組織内での利用を行う場合には、多人数が様々な Web サイトへのアクセスをする状況も考えられ、利用頻度が高いドメインや TCP コネクションが分散する可能性が考えられ、通常の接続予測に基づいたプリフェッチとほとんど差がなくなるか、性能を悪化させる可能性も考えられる。また、TCP のコネクション維持をすることはセキュリティホールに繋がる可能性も考えられる。

3 Data Access History Cache and Associated Data Prefetching Mechanisms

3.1 掲げられている課題と提案手法

この論文で掲げられている課題は、ネットワークアクセスに関するものではなくメモリアクセスに関するものである。通常メモリアクセスの高速化には、利用されることが予測されるデータに関してより高速にアクセスできるような、すなわちプロセッサに近い側のメモリに配置することで対応する。このキャッシングにおいては通常、データの時間的空間的局所参照性を用いる。すなわち、最近アクセスされたデータや、現在アクセスされているデータ近傍のデータなどを事前にキャッシングすることで高速化を図る。しかし、このような局所参照性はアプリケーションによっては存在しないこともある。

筆者らはこの問題に対応するために、過去のデータアクセス履歴を保存しその解析に役立てることができる

データ構造を提案している。

3.2 提案手法の詳細

筆者らが提案した手法は、データにアクセスしてくる PC のインデックスを保存するテーブルと、アクセスされるデータのインデックスを保存するテーブル、そしてデータアクセス履歴を保存するテーブルからなる。PC のテーブルには、PC にユニークなインデックスを振ったものと、一番新しいデータアクセスのデータアクセス履歴テーブル上でのインデックスを保存し、データのインデックスを保存するテーブルにも同様の値を保存する。データアクセス履歴を保存するテーブルには、各レコードごとに PC のインデックス、PC のポインタ、データアドレスのインデックス、そのポインタ、そしてそのアクセスの履歴を保存する。このテーブルを用いることで、各 PC からの各データのアクセスパターンを解析できる、というのが筆者らの主張である。

3.3 提案手法の有効性

提案手法を、ネットワークアクセスに関連して利用する場合に関して述べる。このテーブルに関しては、本来メモリアクセスがボトルネックになるようなアプリケーションで威力を発揮すると考えられ、ネットワークアクセスのようなメモリアクセスが大きな問題とはならないようなアプリケーションではあまり大きな効果はないと考えられる。一方で、同様のデータ構造を用いてユーザのアクセス履歴を元に DNS ルックアップと TCP コネクションを維持する手法 [1] のためのアクセス履歴保持を行うことはできると考えられる。

一方で、そのような用途の場合にはアクセス回数を直接カウントするようなデータ構造のほうがより直接的に効果を発揮できると考えられるため、今回提案されているデータ構造をネットワークアクセスにおいて利用する必要性は高くないと考えられる。

参考文献

- [1] Srikanth Sundaresan, Nazanin Magharei, Nick Feamster, and Renata Teixeira. Accelerating last-mile web performance with popularity-based prefetching. *ACM SIGCOMM Computer Communication Review*, Vol. 42, No. 4, pp. 303–304, 2012.
- [2] Yong Chen, Surendra Byna, and Xian-He Sun. Data access history cache and associated data prefetching mechanisms. In *Proceedings of the 2007 ACM/IEEE conference on Supercomputing*, p. 21. ACM, 2007.