

Web 閲覧の対価



研究ノート

榎本 真佑*

Compensation for Web Browsing

Key Words : Web, advertisement, distributed computing, MapReduce

1. はじめに

Web ブラウザ利用時に表示される広告に対して、一定の煩わしさを感じたことはないだろうか？典型的な例としては、動画を用いた点滅する広告や、スマホのスクロール操作に追従する広告などが挙げられる。これらはとにかく目立てばよい、クリックさせられれば良いといった短絡的な戦略に基づいた広告であり、Web ページ閲覧時の快適さや利便性を大幅に損なうものも少なくない。

このような煩わしい広告の増加に伴い、Web 広告を手軽に遮断できる技術が普及しつつある。この技術は主に Web ブラウザのプラグインとして動作し、定義されたルールにマッチした HTML 要素を非表示、あるいは通信そのものを遮断する。HTML や JavaScript 等の Web を支える各種技術は基本的にオープンなため、このようなクライアント側での HTML コンテンツの操作が容易に実現可能である。PageFair の調査 [1] によると、2015 年時点での米国での広告遮断技術の普及率は 15% に達していると報告されている。

この広告遮断技術は Web のエコシステム（生態系）を崩壊させる要因となりうる。多くの Web サイトは、

無償での情報提供と引き替えに広告掲示スペースを提供しており、ユーザ側による一方的な広告遮断は不公平かつ身勝手な行為であろう。ドイツでは大手出版グループがこの遮断技術の提供元に対して 6 度に渡って訴訟を起こす [2] など、Web の将来を揺るがす大きな問題となっている。

我々の研究の長期的な目標は、現在のような無償でオープン、かつ自由な Web を存続させることにある。本稿では、広告閲覧を対価とする現在の Web エコシステムの代替手段として、クライアントの持つ計算能力を対価とする研究テーマについて紹介する。

2. Web 上での分散処理基盤

提案手法の基本的なアイデアは、Web ページ内に広告を埋め込む代わりに計算タスクを埋め込み実行させることで、クライアントの持つ計算能力をサーバ側で回収するという方法である。かつてはグリッド・コンピューティングと呼ばれていた、複数のマシンで巨大な計算タスクを処理する分散処理コンピューティングの仕組みを、Web サーバとブラウザの間で実現する方法だと見なすことができる。

処理の流れを図 1 に示す。まず、クライアントはブラウザを用いて (1) 特定の Web ページに対し HTTP リクエストを発行し、(2) その結果を受け取る。さらに (3) ブラウザは受け取った HTTP の結果 (HTML データ) を処理し、画面上に表示する。ここまでは一般的な Web 閲覧と同じ処理の流れである。さらに、提案手法では (3) で受け取った HTML データ内に計算タスクを取得する処理が埋め込まれており、(4) 計算集約サーバに問い合わせることで計算タスクを取



* Shinsuke MATSUMOTO

1983年7月生
 奈良先端科学技術大学院大学 情報科学研究科 博士後期課程修了 (2010年)
 現在、大阪大学大学院情報科学研究科 助教 博士(工学) ソフトウェア工学
 TEL : 06-6879-4112
 FAX : 06-6879-4112
 E-mail : shinsuke@ist.osaka-u.ac.jp

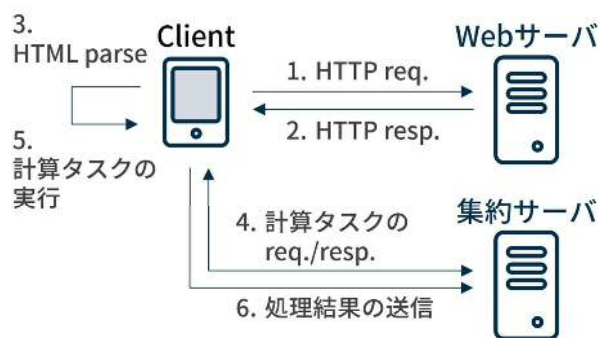


図1 提案手法の流れ

得する。ブラウザは (5) 取得した計算タスクを実行し、(6) その結果を集約サーバに転送する。

Web はクライアントのリクエストに対して HTML や JavaScript のソースコードを送りつけ、クライアント側で解釈・実行するコードオンデマンドと呼ばれるアーキテクチャを取る。これにより、クライアントに任意のプログラムを実行させることが可能であるため、上記の流れは比較的容易に実現できる。しかしながら、巨大な計算タスクをどのように分解し、どのように複数マシンに配布するかという分散処理における定番の問題を解決しなくてはならない。

この問題に対しては、MapReduce[3] と呼ばれる分散処理のためのプログラミングモデルを採り入れている。これはある巨大な計算処理を、Map と Reduce の 2 つの抽象的な関数の組み合わせで実現する分割統治に基づいた考え方である。計算タスクの依頼者は、自身の抱える巨大な計算タスクを Map 関数と Reduce 関数の形に設計し、実装するだけでよい。

3. 特徴と課題

提案手法と Web 広告との大きな違いは、ユーザに対する視覚的な阻害とならないことにある。計算タスクはブラウザのバックグラウンドで処理されるため、Web ページの見た目的一切影響がない。ディスプレイ上に表示されるのは Web ページ本来が持つ情報のみとなる。一方で、クライアントの計算機リソース (CPU やメモリ) を消費するため、端末

の発熱やバッテリー消費の問題が少なからず発生する。特にスマホ等のバッテリーで駆動するモバイル環境では致命的な問題だ。

このように、計算能力の貸与という手段は広告の問題全てを解決するものではない。そもそも広告産業は数多くのメディアで採用されている巨大な産業の一つであり、この市場がなくなることは考えにくい。また、近年ではビッグデータや IoT のような巨大なデータから価値を生むようなコンセプトも広く浸透しており、計算能力に対する需要も高まりつつある。よって、提案手法はあくまで広告の代替手段であり、Web におけるユーザからの報酬を回収する一つの選択肢であると捉えるべきだろう。

また重要な課題として、1 節で述べたような広告遮断技術はそもそも回避できない点が挙げられる。既存の遮断技術のルールを少し工夫するだけで、提案手法の実行を妨げることも容易だ。現在では、遮断技術の実行を検知してページそのものを非表示にする方法や、遮断フィルタのルールにマッチしないように HTML 要素の名前に乱数を付与するなどの回避策も生まれているが、基本的にはイタチごっこであり根本的な解決には至っていない。Web がオープンな思想で生まれた以上、クライアント側での自由な改変を防ぐことは技術的に不可能である。研究テーマとして取り組むにはいささか壮大ではあるものの、本質的には Web 全体の文化や利用者の考え方そのものを変えることが肝要である。読者の中には遮断技術を利用している方もいらっしゃると思うが、無償で情報を得る「対価」として自身が何を差し出せるかをもう一度見直して頂ければ幸いである。

- [1] PageFair. "Adobe (2015): The cost of ad blocking – PageFair and Adobe 2015 Ad Blocking Report." 2016.
- [2] R. J. Walls, et al. "Measuring the Impact and Perception of Acceptable Advertisements." Conf. Internet Measurement Conference, 2015.
- [3] J. Dean, and S. Ghemawat. "MapReduce: Simplified Data Processing on Large Clusters." Commun. ACM 51.1, 2008.