

HTML5 を用いたインターネットの品質測定

永見 健一¹ 水越 一郎² 白石 雄太³

概要：HTML5 には、コンテンツの取得時間・表示時間を測定する API が定義されている。本論文では、この HTML5 の API を利用したインターネット品質測定方式を提案する。本提案は、ブラウザで表示するコンテンツのダウンロード時間を測定するパッシブ測定である。測定用のトラフィックを必要としないため、ネットワークやサーバへの負荷を増加させない。この提案方式を実際の WEB サーバに設置し、スマートフォンの品質測定を実施した。ネットワーク遅延の中央値は、携帯網経由では 55ms~90ms、Wi-Fi 経由では 20ms となった。

Quality measurement using the HTML5

Kenichi Nagami¹ Ichiro Mizukoshi² Yuta Shiraishi²

1. はじめに

HTML5 には、コンテンツの取得時間・表示時間を測定する API が定義されている。本論文では、この HTML5 の API を利用したインターネット品質測定方式を提案する。本提案は、ブラウザで表示するコンテンツのダウンロード時間を測定するパッシブ測定である。測定用のトラフィックを必要としないため、ネットワークやサーバへの負荷を増加させない。また、測定項目が API で標準化されているため、複数組織のデータ比較が容易であり、ネットワークの品質測定に向いている。

本論文では、ネットワーク品質測定について、提案方式及び既存方式を比較し、それぞれの特徴を整理する。また、提案方式をアクセスの多いウェブページに設置し、ネットワーク品質測定を実施した。この結果を利用して、近年利用が拡大しているスマートフォンのインターネット品質に関して、モバイル網及び Wi-Fi 利用時の遅延、スループットを報告する。また、Google 社が提供しているモバイル向けデータ圧縮プロキシを利用した品質に関して、通常のアクセスとの比較を報告する。

キーワード: HTML5, 品質測定, モバイル, モバイルプロキシ, インターネット, ネットワーク管理, Web サービス

2. 関連研究

国内においては 2014 年から総務省でインターネットのサービス品質計測に関する研究会が開催されており、報告書[1]が公開されている。この内容に基づきモバイル端末上のアプリケーションを用いたアクティブ測定が 2015 年度から国内 1500 地点で実施され、結果が公開されている[2][3][4]。この測定方式には、収集データ数を増やすためのコスト増加が大きい、1 年間で 1 回しか測定されない、アクティブ測定によるネットワークへの負荷が増す、測定装置への最高速度を測定しているのであって実際の利用状況とは異なるといった課題が存在している。

米国 FCC では 2011 年から固定ブロードバンドの品質測

定を行っており 2014 年からはモバイルに対する測定も開始した。測定方法やその結果については FCC のホームページ[5]で公開している。固定回線ではモニター宅に専用のブロードバンドルータを設置し、モバイル回線では専用アプリケーションを無料で配布し、アクティブ測定を行なっている。このため、総務省方式における測定と同様な課題を内在している。

一方、Google 社[6]や Netflix 社[7]は動画の配信品質を測定し、配信品質を満たしているインターネットサービスプロバイダとして結果の公開を始めている。

3. 提案方式

本章では、ブラウザを利用した品質測定方式を説明する。最初に主要なブラウザに実装されており、本提案に利用する Navigation Timing API を説明する。その後、提案方式を説明する。

3.1 W3C Navigation Timing API

HTML の国際標準化団体である W3C(World Wide Web Consortium)では、ブラウザのコンテンツ表示のパフォーマンスを測定するための API を規定している。Navigation Timing API は、2012 年 12 月に勧告され、現在の主要なブラウザに実装されている。また、本 API 以外のパフォーマンスに関連する API も W3C の WG で議論されており、Resource Timing API も主要なブラウザに実装されている。

図 1 は、Navigation Timing API で取得できる情報である。本 API は、ユーザがブラウザをクリックした時点からコンテンツが表示されるまでの各種の時刻を取得できる。ネットワークに関連する時刻としては、以下の時刻をミリ秒単位で取得できる。

- DNS 要求/応答の時刻
(domainLookupStart/domainLookupEnd)
- TCP コネクション確立に関する時刻

1 株式会社インテック Intec Inc.

2 情報セキュリティ大学院大学 Institute of Information Security

3 株式会社イー D IID Inc.

(connectStart/connectEnd)

・ HTTP 要求/応答の時刻

(requestStart/responseStart/responseEnd)

図 2 は、Navigation API で取得できる時刻とブラウザとサーバ間のメッセージシーケンスを示している。

本 API は、メインリソース(html)のパフォーマンスを測定できるが、Resource Timing API を利用することによりメインリソースで指定されたサブリソース(jpeg 等)のパフォーマンスを測定できる。

Navigation Timing API や Resource Timing API は、Internet Explorer, Firefox, Chrome, Opera, Android ブラウザなど主要なブラウザで実装されている。

Resource Timing API や Navigation Timing API の次世代版の Navigation Timing2 API では、転送バイト数が定義されている。転送バイト数と転送時間によりスループットが計算できる。

本稿では、Navigation Timing API 及び Resource Timing API を利用した品質測定を提案する。以下では、Navigation Timing API を説明するが、Resource Timing API も同様に利用可能である。

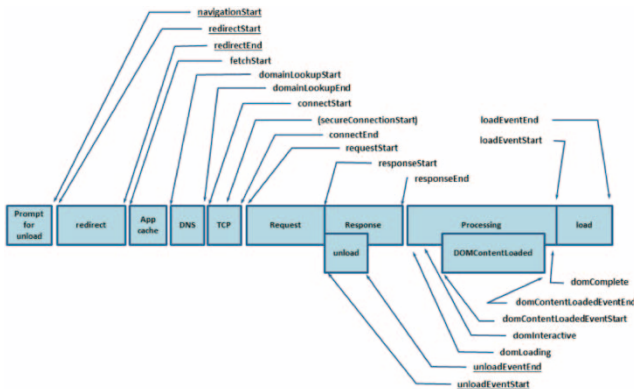


図 1 Navigation Timing API の取得情報

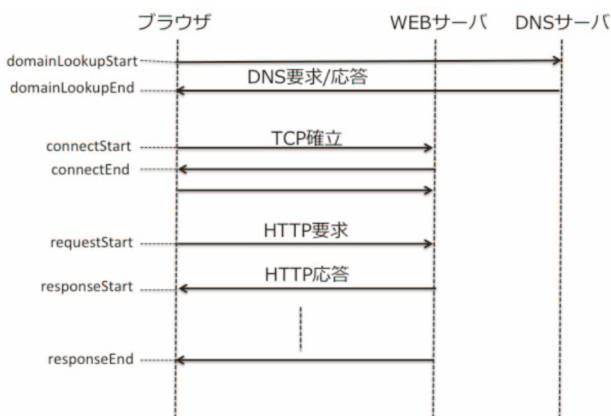


図 2 Navigation Timing API とシーケンス

3.2 提案方式

本節では、Navigation Timing API を利用した提案方式を説明する。図 3 は、現状の WEB サーバと提案する測定方式である。現状の WEB サーバは、PC やスマートフォンのブラウザからのコンテンツ要求によりコンテンツを送信する。この結果として、

ブラウザにコンテンツが表示される。

提案方式では、WEB サーバのコンテンツ(html)に測定用のスクリプトを挿入する。スクリプトを挿入するのみなので、WEB サーバの管理者の開発や運用の負荷は少ない。品質測定用の測定スクリプトは、Navigation Timing API を利用して、DNS 応答時間、TCP 接続時間、コンテンツダウンロード時間を測定し、測定結果をサーバにアップロードする。このスクリプトは、WEB サーバのコンテンツを表示した時に自動的に実行されるので、ユーザが操作をする必要はなく、ユーザに負担をかけない。また、ユーザが閲覧するコンテンツをもとに測定するので、ネットワークの負荷も増えない利点がある。

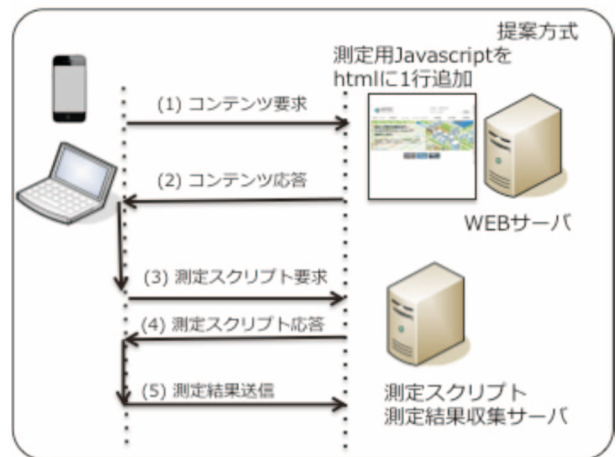
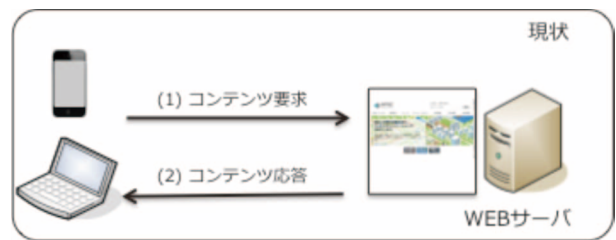


図 3 提案方式

4. 提案手法の特徴

この章では、提案手法の特徴を説明する。まず図 4 で提案方式を含む3つの方式を説明し、表 1 でそれぞれの方式の比較を行う。

方式 1 は、測定用の専用アプリを使って最大帯域を測定する方式である。専用アプリの例としては、スマートフォン用のスピード測定アプリやスピード測定サイトがある。方式 2 は、本論文の提案方式である。方式 3 は、コンテンツ配信サーバの配信ログを用いてコンテンツの配信の測定を行う方式である。

・測定項目

方式 1 は、ネットワークの最大帯域を測定する。方式 2、方式 3 は、ネットワークの最大帯域ではなく、コンテンツが実際に利用した配信帯域を測定する。ネットワークの能力を測定するためには、方式 1 が適しており、配信コンテンツの体感を測定するには、方式 2、方式 3 が適している。

・測定対象

方式 1 は、専用アプリで測定用のパケットを利用できるため、ネットワークの最大帯域を柔軟に測定することができる。方式 2 は、W3C 標準の API を利用している。現状の API では動画を測定することできず写真などの静的コンテンツのみとなる。方式 3 は、配信コンテンツのログの取得項目に依存している。詳細な配信コンテンツログを取得していれば、動画の停止時刻等を取得することはできるが、それぞれのコンテンツの配信ログ次第となる。

・ネットワーク・サーバの負荷

方式 1 は、測定専用のパケットを送出するアクティブ計測のため、ネットワークや測定サーバに測定のための負荷がかかる。方式 2、方式 3 は、利用者が見るコンテンツを用いたパッシブ測定のため、測定のためのコンテンツはサーバから送出不し。このため、測定のためのネットワークやサーバの負荷は増加しない。

・測定結果の比較の容易性

測定項目が同一であれば、複数の測定結果を比較することが容易になる。方式 1 と方式 3 は、専用アプリの測定アルゴリズムや配信コンテンツのログによって測定項目が異なる可能性があり、測定結果を比較するのは容易ではない。方式 2 では、W3C 標準の API を利用しているため、測定項目が統一され、測定結果の比較は容易である。

・サンプル獲得の難易度

測定の精度をあげるためには、測定サンプル数とそのばらつきが重要である。方式 1 は、専用アプリが実行された回数となり、サンプル数、ばらつき共に限られてしまう。方式 2、方式 3 は、人気コンテンツ配信サイトで測定できれば、多くの利用者の品質を測定することができ、大量にばらついたサンプルが獲得できる。

これらの比較により、提案方式(方式 2)は、測定対象は限られてしまうが、ネットワークの負荷を増加させず、多くのサンプル数が取得できるため、ネットワーク品質を測定するのに適していると考えられる。また、測定結果の比較が容易であることから、複数サイトの測定結果を集計ことができ、ネットワーク全体の傾向を把握するのに向いている。

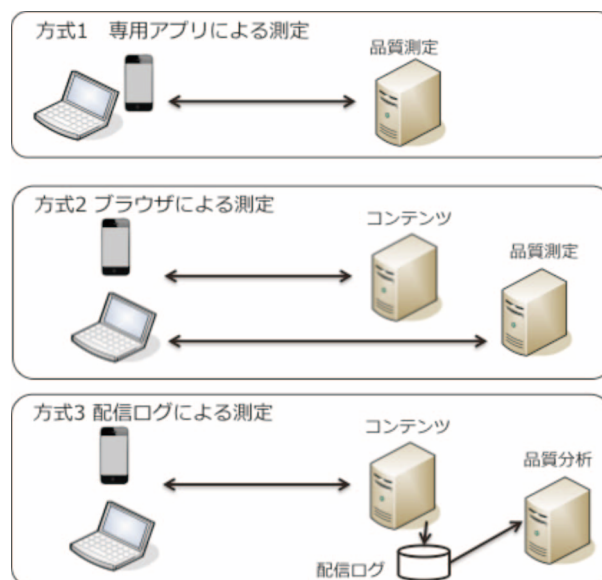


図 4 品質計測方式

表 1 測定方式比較

	方式 1 専用アプリ	方式 2 提案方式 配信コンテンツ をブラウザで測 定	方式 3 配信コンテン ツの配信ログ 分析
測定項目	最大帯域	コンテンツ 配信帯域	コンテンツ 配信帯域
測定対象	○ 測定用パケット	× 静的コンテンツ	○ 配信ログが取得できる全コンテンツ
ネットワーク・サーバの負荷増加	× アクティブ測定	○ パッシブ測定	○ パッシブ測定
測定結果の比較の容易性	× アプリ依存	○ W3C 標準 API	× コンテンツ依存
サンプル数獲得の難易度	× 難	○ 人気コンテンツ配信が利用できれば容易	○ 人気コンテンツ配信が利用できれば容易

5. 評価

提案方式を用いて 2015 年からデータを取得している[10]. 2016 年後半に主要ブラウザに転送バイト数取得 API 等が実装されたため、スループット等の計測項目が増加した。また、設置サーバの増加によりデータ取得数が増加している。本章では、提案方式で測定したスマートフォンの品質測定の結果を最新のデータを用いて示す。

5.1 データセット

提案方式を複数の WEB サーバに設置しデータを収集した。WEB サーバは、PC やスマートフォンなどのさまざまな機器で閲覧されている。今回は、http ヘッダのユーザーエージェントの情報を用いて、スマートフォンのデータを抽

出した。本論文では、2017年4月16日から2017年4月29日の2週間のデータの分析結果を示す。データ数は、2週間で約940万アクセス。1日あたり約67万アクセスである。

5.2 ネットワーク遅延とスループット

本節では、ネットワーク遅延とスループットの分析結果を示す。ネットワーク遅延は、TCPコネクション接続時の接続遅延である。スループットは、コンテンツのダウンロード時間とコンテンツの転送サイズにより計算している。評価で利用しているWEBサーバでは、スマートフォン利用者の表示待ち時間を少なくするため、できる限り小さなファイルサイズの写真等が設置されている。スループット計算には、TCPのスロースタートの影響を少なくするため、ある程度大きなファイルサイズが望ましい。今回は、小さいファイルサイズを取り除き、10kBから100kBのファイルサイズでスループット計算を実施した。

図5は、アクセス回線が携帯網の場合とWi-Fiの場合のネットワーク遅延を示す。縦軸は、1日毎のネットワーク遅延の中央値である。携帯網の場合には、遅延のばらつきがあり、55ms～90msの遅延であった。Wi-Fiの場合は、約20ms程度の遅延であった。ネットワーク遅延は、携帯網よりWi-Fiが少ない結果となった。

図6は、携帯網及びWi-Fiのスループットを示す。縦軸は、1日毎のスループットの中央値である。携帯網の場合には、2Mbps～2.5Mbpsであった。Wi-Fiの場合は、1.5Mbps～2Mbpsであった。Wi-Fiよりも携帯網のスループットが高い結果となった。

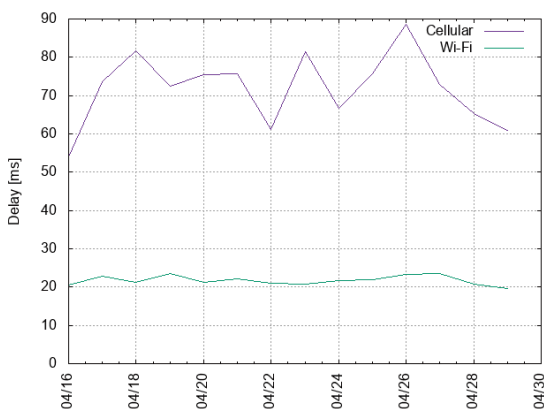


図5 ネットワーク遅延

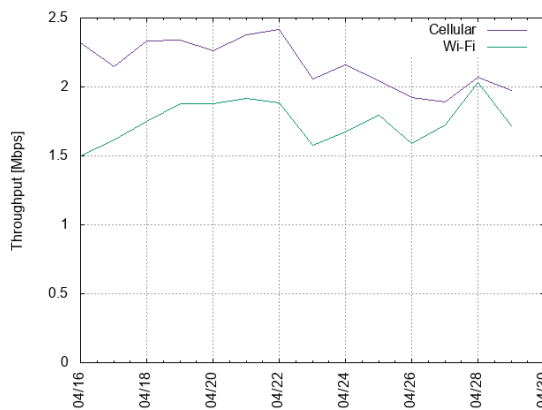


図6 スループット

5.3 アクセス回線比率

スマートフォンでは、インターネットへのアクセスの方法として、携帯網を利用する方法とWi-Fiを利用する方法の2通りがある。スマートフォン利用者は、通常は携帯網を利用している。携帯網は、1ヶ月の利用容量の上限の制限や、接続帯域の制限があるため、スマートフォン利用者は、Wi-Fiも利用する。携帯事業者は、携帯網の利用統計を取得しているが、自身の設備ではないWi-Fiの利用統計を取得するのは困難であり、スマートフォンの利用状況を把握するのは難しいと考えられている。

本提案手法では、アクセス回線が携帯網及びWi-Fiを検出できる。このデータを利用し、スマートフォンでのアクセス回線の利用状況を分析した。図7は、データセットの中から平日5日を抜き出し、Wi-Fiとアクセス回線のアクセス数を表した図である。図8は、時間ごとのWi-Fiアクセス比率を示す。

毎日、時間ごとに傾向があり、朝及び夜にWi-Fiの利用が高くなっている。

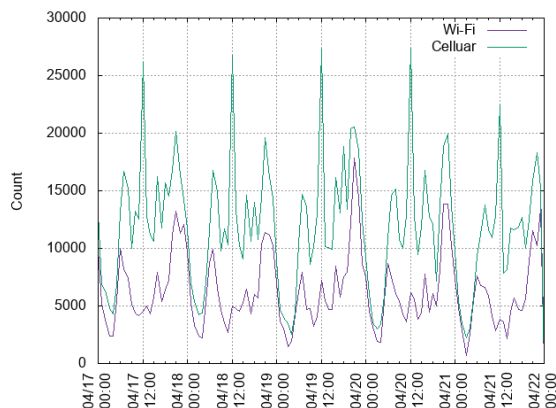


図7 アクセス回線別アクセス数

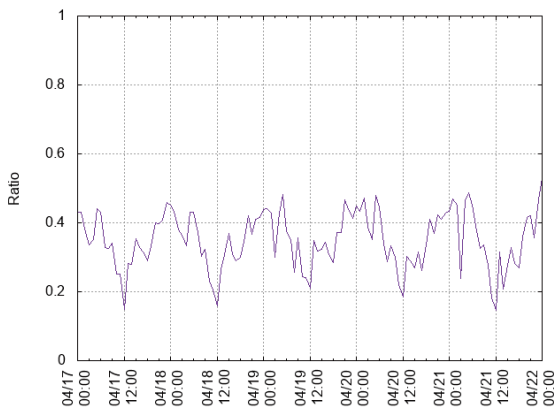


図 8 Wi-Fi アクセス比率

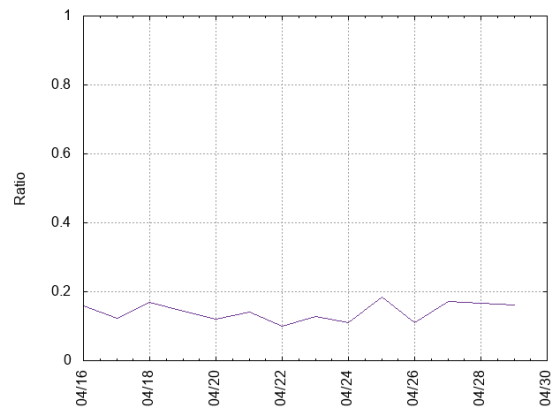


図 10 データ圧縮プロキシの利用率

5.4 Google データ圧縮プロキシ

Android の Chrome を利用しているユーザは、データ圧縮を設定していると、図 9 のように Google 社のデータ圧縮プロキシを経由し、データ転送サイズが小さくなる[8][9]. データ圧縮プロキシを経由する WEB サイトは、http を利用している通信が暗号化されていない WEB サイトである. https を利用するセキュアサイトへの通信は、データ圧縮プロキシを経由しない.

本節では、データ圧縮プロキシの利用率及びデータ圧縮サイズを分析した.

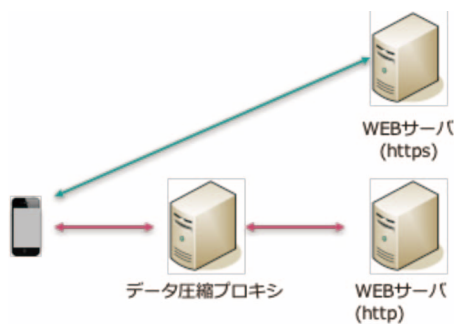


図 9 データ圧縮プロキシ

図 10 は、http を利用しているサイトでのデータ圧縮プロキシの利用率である. データ圧縮プロキシの利用率は、約 20%である. 図 11 は転送データ量を示している. データ圧縮プロキシを経由すると、約 1/3 程度のデータに圧縮されている. 転送データ量が少なくなるため、ユーザの体感も早くなると考えられる.

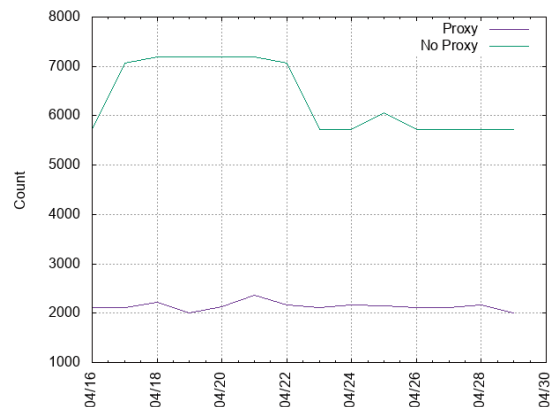


図 11 データ圧縮プロキシの転送データ量

6. まとめ

本稿では、W3C 標準 API を利用した品質測定方法を提案した. 本方式は、ネットワークに測定用の負荷をかけることがなく、ユーザの体感速度を大量のサンプル数を取得することができる手法である.

また、提案方式により実際のデータを収集して分析した. ネットワーク遅延、スループット、プロキシの利用等ネットワークの現状を把握する 1 つの手法と考えられる.

7. 参考文献

- [1] インターネットのサービス品質計測等の在り方に関する研究会, "インターネットのサービス品質計測等の 在り方に関する研究会 第一次報告書," 2014.
- [2] NTT ドコモ社, 実行速度計測結果, [Online]. https://www.nttdocomo.co.jp/support/area/effective_speed/index.html
- [3] KDDI 社, 実行速度について, [Online]. <https://www.au.com/mobile/area/effective-speed/i/>
- [4] ソフトバンク社, 「総務省実効速度調査」測定方法/測定結果, [Online].

<http://www.softbank.jp/mobile/network/explanation/speed-survey/>

[5] FCC. Measuring Broadband America. [Online].
<http://www.fcc.gov/measuring-broadband-america>

[6] Google. Google 動画品質レポート. [Online].
https://www.google.com/get/videoqualityreport/#how_video_gets_to_you

[7] Netflix ISP Speed Index. [Online].
<https://ispspeedindex.netflix.com/>

[8] Google, Data Compression, [Online].
<https://developer.chrome.com/multidevice/data-compression>

[9] V. Agababov, et al., Flywheel: Google's Data Compression Proxy for the Mobile Web, USENIX NSDI, 2015.

[10] 永見, 水越, HTML5 を用いた品質測定の提案, WIT, 2015.