一般社団法人　電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
IEICE Technical Report
ICM2018-63 (2019-03)

Webオブジェクトの共起度分析

桜井　洸輝†　　上山　憲昭†　　中尾　彰宏††

† 福岡大学工学部電子情報工学科　〒 814–0180 福岡市城南区七隈 8–19–1
†† 東京大学大学院 情報学環・学際情報学府　〒 113-0033 東京都文京区本郷 7-3-1
E-mail: †tl151225@cis.fukuoka-u.ac.jp, ††kamiyama@fukuoka-u.ac.jp, †††nakao@nakao-lab.org

あらまし　Web ページの表示待ち時間を低減する技術として，Web ページを構成する複数のオブジェクトを同時並列に取得する HTTP/2 が標準化され広く普及している．しかし HTTP/2 の並列配信は同一の配信サーバから取得するオブジェクトに対してのみ可能であり，Web 表示待ち時間の低減効果を高めるには，少数の配信サーバから多数のオブジェクトを取得する必要がある．そこであるオブジェクトの組に対し，その組が出現する Web ページの数を共起度と定義すると，共起度の高いオブジェクト組が優先的にキャッシュに残るよう，キャッシュ置換を行うことが望ましい．しかし共起度に基づくキャッシュ置換制御の効果は，実際の Web ページにおいてどの程度，オブジェクト間の共起現象が生じるかに依存する．そこで本稿では，共起度に基づくキャッシュ置換制御の可能性を明らかにするため，高人気の 7,604 の Web ページを構成する約 70 万個のオブジェクトを対象に，オブジェクトの 2 個組と 3 個組の共起度を計算し，Web ページの共起現象の程度を分析する．その結果，共起度の分布は冪乗則に従い，0.1%程度の 2 個組や 0.01%程度の 3 個組は 100 以上の，0.01%程度の 2 個組や 0.0005%程度の 3 個組は 500 以上の共起度を有することを明らかにする．そのためこれら少数の共起度の高いオブジェクト組を優先的にキャッシュに残すキャッシュ置換制御の有効性が期待される．
キーワード　HTTP/2，Web オブジェクト，共起

Measurement Analysis of Co-occurrence Degree of Web Objects

Kouki SAKURAI†, Noriaki KAMIYAMA†, and Akihiro NAKAO††

† Fukuoka University　8–19–1, Nanakuma, Jounan, Fukuoka 814–0180
†† The University of Tokyo　7–3–1, Hongo, Bunkyo-ku, Tokyo 113–0033
E-mail: †tl151225@cis.fukuoka-u.ac.jp, ††kamiyama@fukuoka-u.ac.jp, †††nakao@nakao-lab.org

Abstract　As a technique to reduce the web response time, HTTP/2 which enables user terminals to simultaneously download multiple web objects consisting one webpage has been standardized and widely used. However, parallel download of HTTP/2 is possible only for web objects which are provided from the same object or cache server, so a large number of objects need to be downloaded from a small number of servers to improve the effect of HTTP/2. Therefore, it is desirable to replace cache servers so that object sets with high co-occurrence degree are remained in the cache, where we define the co-occurrence degree of object set as the number of webpages in which the object set is included. However, the effect of cache-replacement method based on the object co-occurrence degree depends on how the co-occurrence phenomenon of objects appears in the actual webpages. Therefore, to clarify the possibility of cache-replacement method based on the co-occurrence degree, we investigate the degree of co-occurrence phenomenon of objects by calculating the co-occurrence degree of two-objects and three-objects pairs of about 0.68 million objects constructing popular 8,000 webpages in this paper. We confirm that the distribution of co-occurrence degree of web objects obeys the power law. We also clarify that about 0.1% two-object pairs and 0.01% three-object pairs have more than 100 co-occurrence degree, and about 0.01% two-object pairs and 0.0005% three-object pairs have more than 500 co-occurrence degree. Therefore, we can expect the effectiveness of replacing cached objects so that a small number of object sets with extremely high co-occurrence degree are remained with high priority.
Key words　HTTP/2, web object, co-occurrence

1. はじめに

Web 閲覧サービスはインターネット上で最も普及したサービスの 1 つであり，世界中の多くの人々が毎日 Web 閲覧サービスを利用している．しかし毎週 67%のユーザがブラウジング時に長い待ち時間を経験し [5]，17%のユーザが 5 秒を超えても待たないことが報告されている [9]．本稿では，Web 応答時間を Web ページのハイパーリンクをクリックしてから Web ブラウザに Web ページ全体が表示されるまでの待機時間として定義する．ユーザはページが 2 秒以内にロードされることを期待しており、その 40 %が 3 ページ以内にページを開くまで待機すると言われている [17]．また 400 ミリ秒の遅延により Google 検索エンジンでの検索が 0.74%減少することや [22]，Web 応答時間が 0.1 秒だけ削減するごとに Amazon の収益が 1%増加することが報告されている [23]．また高速に表示される Web ページはユーザが購買を完了する回数が 15%も多く，また 1 ページだけ閲覧した後にページから離脱する回数が 9%も少ないことが報告されている [18]．したがって、多くのインターネットサービスプロバイダ (ISP: Internet Service Provider) やコンテンツプロバイダにとって、ウェブ応答時間を短縮することは、ユーザの体感品質およびコンテンツプロバイダの利益を改善するために解決する必要がある重要な課題である．

従来の Web ページは静的なテキストや画像といったオブジェクトがサーバに用意され，Web ブラウザは HTTP を用いてこれら静的オブジェクトを単にダウンロードして表示していた．しかし近年，クライアント端末からの要求受信時に，サーブレットや JSP (Java server pages) のプログラムをサーバ側で実行するか，JavaScript で書かれた Ajax や DOM (document object model) によるプログラムを HTML に埋め込みユーザ端末側で実行することで生成される動的オブジェクトの割合が増加している [5]．また，広告を専用のサーバから取得するなど，各オブジェクトの配信元が多元化している．このように一つの Web ページを構成するオブジェクトは複雑性を増している．

Web オブジェクトの取得には TCP セッション上で HTTP (hypertext transfer protocol) が用いられる．従来，広く用いられていた HTTP/1.1 では，オブジェクトの取得に要する遅延時間を低減するため，同一配信ホストから取得する複数のオブジェクトを 1 つの TCP セッション上で取得する HTTP persistent connection と，さらに同一ホストから複数のオブジェクトを並列に取得する HTTP pipelining が実装されている．しかし同一の TCP コネクション上で転送されるパケットが属するオブジェクトをユーザ端末が識別できないため，配信サーバは HTTP request を受信した順番でオブジェクトを返信する必要があり，送信準備ができたオブジェクトの返信開始が，他のオブジェクトの返信完了まで待たされる Head of Line (HOL) 問題が生じる．図 1(a) に HTTP/1.1 を用いて a,b,c の 3 個オブジェクトを同一の配信ホストからユーザ端末がダウンロードする場合の配信フローを例示する．図では 3 つのオブジェクトに対する配信要求が a,b,c の順で配信ホストに到着し，オブジェクト b の生成に時間を要した場合を示している．オブジェクト c の配信準備は完了しているが，配信サーバは a,b,c の順オブジェクトを送信する必要があるため，オブジェクト b の送信が完了した後で，オブジェクト c の配信が可能となる．

このような HOL 問題を解決するため Google は，パケットに「SPDY stream」と呼ばれる所属 HTTP セッションの識別子を付加することで，ユーザ端末が各パケットの所属オブジェクトを識別可能とし，配信サーバが任意の順番でオブジェクトを配信可能とする SPDY を開発した [13] [23] [25]．図 1(b) に，3 つのオブジェクトを SPDY で配信した場合の例を示す．配信サーバは任意の順番でオブジェクトの配信が可能であり，先に配信準備が完了したオブジェクト c をオブジェクト b に先立ち送信を開始できる．SPDY は HTTP の機能として標準化され，現在，SPDY が組み込まれた HTTP が HTTP/2 として普及しつつある [16]．新バージョンである HTTP/2 にはいくつかの機能が追加され，その中でも特に主要な機能である SPDY による多重化やサーバープッシュ機能により HTTP/1.1 までの問題点であった HOL 問題が回避され，Web 応答時間の低減改善が期待されている．

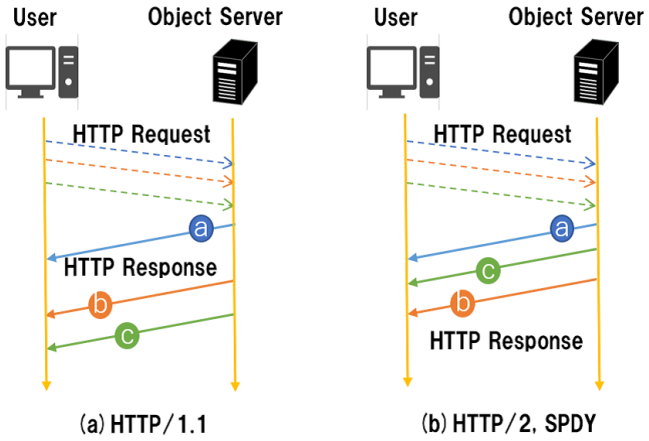

図 1 Example of object delivery sequence in HTTP/1.1 and HTTP/2 (SPDY)

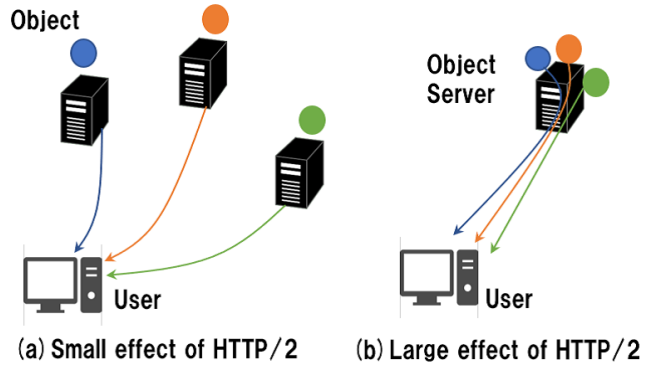

図 2 Comparison of two cases of delivering three objects from three servers or one server

しかし HTTP/2 のオブジェクト並列配信機能は，同一の配信サーバから取得するオブジェクトに対してのみ有効である．すなわち異なる配信サーバから取得するオブジェクトを同一 TCP セッション上で取得することができず，HTTP/2 の並列配信による遅延時間削減効果も期待できない．例えば図 2 に 3 個のオブジェクトを 1 個ごとに別のサーバから取得する場合と，同じサーバから取得する場合の例を示すが，HTTP/2 の効果は同一の配信サーバから取得するオブジェクト数が増えるほど大きくなる．そのため HTTP/2 の効果は期待したほど得られないという報告もなされている [10]．Web 応答時間の低減効果を高めるには，少数の配信サーバから多数のオブジェクトを

取得する必要がある．

そこであるオブジェクトの組に対し，その組が出現する Web ページの数を共起度と定義すると，共起度の高いオブジェクト組が優先的にキャッシュに残るよう，キャッシュ置換を行うことが望ましい．しかし共起度に基づくキャッシュ置換制御の効果は，実際の Web ページにおいてどの程度，オブジェクト間の共起現象が生じるかに依存する．そこで本稿では，共起度に基づくキャッシュ置換制御の可能性を明らかにするため，実際の Web ページを構成するオブジェクトを対象に共起度を計算し，Web ページの平均共起度などの共起度に関する各種特性を分析する．以下 2 節では共起度に基づくキャッシュ制御方式の概要を述べる．そして 3 節では，実際の Web ページにおける共起度の測定分析結果について述べ，最後に 4 節で全体をまとめる．

2. HTTP/2 の効果向上のための共起度に基づくキャッシュ制御

2.1 概　　念

1. 節で述べたように HTTP/2 の並列配信の効果は同一の配信サーバから取得するオブジェクト集合に対してのみ有効である．そのため多数の配信サーバからオブジェクトを，各々からは少数だけ取得する場合には HTTP/2 の効果は低減する．ところで Web 応答時間を低減する技術として CDN（Content Delivery Network）が広く普及している [18] [19] [21]．CDN はネットワーク上の多数の場所に設置されたキャッシュサーバにオブジェクトのコピーをキャッシュし，ユーザの近くに存在するキャッシュサーバからオブジェクトを配信することで Web 応答時間を低減する．HTTP/2 を利用するには配信サーバとユーザ端末の両方が対応する必要があるが，オブジェクトのオリジナルを提供するオリジンサーバが HTTP/2 に未対応であっても，CDN 事業者がキャッシュサーバを HTTP/2 に対応させることでキャッシュサーバとユーザ端末間で HTTP/2 を利用することができる．既に Akamai，Amazon CloudFront，Microsoft Azure，Google Cloud Platform などの主要 CDN サービスが HTTP/2 に対応している．

キャッシュサーバは配信要求に対しキャッシュに要求オブジェクトが存在しない場合，オリジンサーバからオブジェクトを取得してキャッシュしたのちユーザに配信する．キャッシュの空き容量が不足する場合はキャッシュ済みオブジェクトの一部を削除するが，その選択方法（キャッシュ置換法）がキャッシュの Web 応答時間の低減効果に影響する．本稿では任意の個数のオブジェクト組が複数の Web ページ内で出現することを「共起」と呼び，あるオブジェクト組が出現する Web ページの数を「共起度」と呼ぶ．例えばあるオブジェクト組を含む 20 の Web ページが存在する場合，これらオブジェクト組の共起度は 20 となる．図 3 に青，オレンジ，緑の 3 つのオブジェクトから構成される 3 つの Web ページが存在する状況における，オブジェクトの 3 つの組の共起度を例示する．例えば青とオレンジのオブジェクト組は，2 つの Web ページに出現しているため共起度は 2 となる．

従来のキャッシュ制御方式は各オブジェクトを独立に扱う．しかし HTTP/2 の並列配信による Web 応答時間の低減効果は，共起度の高いオブジェクトほど大きくなる．そのため共起度の高いオブジェクトの集合が優先して残るよう，キャッシュ置換を行うことが望ましい．例えば共起度の高い 10 個のオブジェクトの組が存在するとき，これら 10 個のオブジェクトの組を一塊として，キャッシュへの挿入と削除を行うことが考えられる．その結果，1 つの Web ページを閲覧したときに同一のキャッシュサーバから配信されるオブジェクト数が増加し，HTTP/2 の効果の向上が期待される．

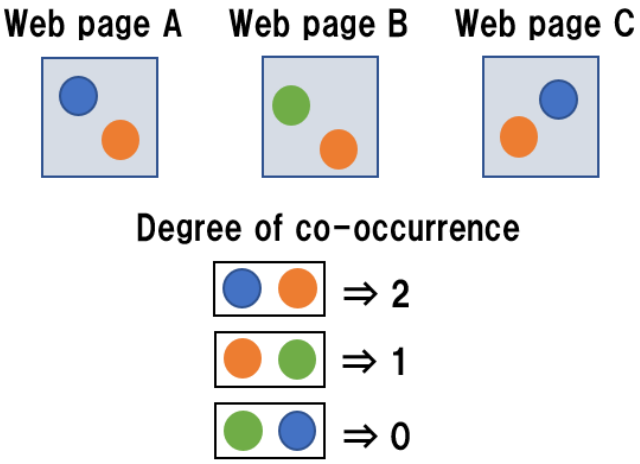

図 3　Degree of co-occurrence of three two-objects pairs among three web pages

2.2 共起度に基づくキャッシュ置換の例

ウェブオブジェクトの共起度に基づくキャッシュ置換方針の簡単な例を示す。1. 節で述べたように，本稿の目的は Web オブジェクトの共起度に基づくキャッシュ置換の概念を提案し，実際の Web オブジェクトの共起度を測定分析することにより共起度に基づくキャッシュ制御の可能性を明らかにすることである．そのため本稿では Web オブジェクトの共起度に基づくキャッシュ置換方針の概略的な例を示すのにとどめ，詳細な検討は今後の課題とする．キャッシュ置換方針の大まかな例を以下に示す．

（1） **共起度の閾値 T の設定**

初めにキャッシュサーバはキャッシュ置換の単位と見なす Web オブジェクト組の共起度の下限値 T を設定する．

（2） **高共起度オブジェクト集合の測定**

キャッシュサーバは世界中の様々な Web ページに定期的にアクセスし，任意のオブジェクト組の共起度を測定することにより，共起度が T 以上のオブジェクト組のリストを更新する．

（3） **組単位でのキャッシュオブジェクトの置換**

新しいオブジェクトをキャッシュに挿入する際に空き容量が不足する場合には，キャッシュサーバは LRU によっていくつかのオブジェクトを削除する．共起度が T 以上のオブジェクト集合に含まれるオブジェクトについては，キャッシュサーバがオブジェクト集合単位で置き換える．これらのオブジェクトセットのそれぞれについて，最後の要求からの経過時間は，オブジェクトセットに含まれるすべてのオブジェクトの最小値により定義する．

3. 共起度の測定分析

2. 節で述べた共起度に基づくキャッシュ置換制御の効果は，現実の Web ページにおいてどの程度，共起現象が見られるかに大きく依存する．そこで本稿では，実際の多数の Web ページに含まれるオブジェクトの共起度を測定分析することで，共

起度に基づくキャッシュ制御の可能性を明らかにする．

3.1 オブジェクトデータの測定法

本節では，Web ページに含まれるオブジェクトデータの測定手順を述べる．アクセス数の多い高人気の Web ページに含まれるオブジェクトを共起度分析の対象とすることが望ましい．そこで adult, arts, business, computers, games, health, home, kids&teens, news, recreation, reference, regional, science, shopping, society, sports の 16 のカテゴリごとに，Web ページのアクセス数のランキングを公開している Alexa の Web ページから [1]，各カテゴリに対して上位 500 の Web ページの URL を測定対象としてリスト化した．次に生成した評価 URL リストの各 URL に対して，測定用 PC から GET の HTTP リクエストを送信した際に発生する通信特性を，HAR (HTTP Archive) ファイルとして取得した [11]．HAR ファイルは，測定用 PC とサーバ間で転送される HTTP データのヘッダ情報から，測定用 PC において，各オブジェクトの URL，サイズ，取得に要した遅延時間等の各種通信特性を算出し，JSON (JavaScript Object Notation) 形式で出力したものである．なお 16 の各カテゴリから 500 ずつの合計で 8,000 の Web ページを測定対象としたが，正常に HAR ファイルを取得できたのは 7,604 ページであったため，実際には 7,604 ページを対象に分析を行う．

3.2 オブジェクトの識別法

共起度分析の前にまずは各オブジェクトの分析を行う．取得した JSON ファイルを解析することで，各 Web ページを構成する各オブジェクトの URL 名を取得し，URL 名で全てのオブジェクトを識別したが，異なる URL でもアクセス結果が同じである事象が多数確認された．そこでオブジェクトの Hash 値，オブジェクトのサイズ，mime タイプの 3 要素が同一のオブジェクトを同一のオブジェクトとして判別する方法についても分析を行った．7,604 ページを構成するオブジェクトの総数は 707,690 で，その異なり数は URL 識別の場合は 489,100，Hash 識別の場合は 373,860 であった．

3.3 オブジェクトの重複度

各オブジェクトに対し，7,604 の Web ページの中で出現する Web ページの数を重複度と定義する．図 4 に，URL 識別，Hash 識別それぞれについての各オブジェクトの重複度の累積補分布 (CCD: complementary cumulative distribution) を示す．重複度 1 のオブジェクトが全体の 9 割程度を占めるが，重複度の分布の裾野は広く，冪乗則が観測される．URL 識別では 0.05%程度のオブジェクトは 100 以上の Web ページに，0.001%程度のオブジェクトは 1,000 以上の Web ページに出現している．また Hash 識別では 0.08%程度ののオブジェクトは 100 以上の Web ページに，0.004%程度のオブジェクトは 1,000 以上の Web ページに出現している．

URL 識別による高重複度オブジェクトの例を表 1 に示す．重複度の高いオブジェクトには google, facebook, doubleclick.net といった名を含む URL が多く存在していた．また高重複度オブジェクトの多くは js の拡張子が付いた javascript オブジェクトであることが確認できる．

また，Hash 識別により同一とみなされたオブジェクトの中には異なる URL を有するものが多数確認された．そのうち重複度が 1,000 を超えているオブジェクトの異なる URL 群の例を表 2 に示す．これらの URL は全て異なるがドメイン名は全て googletagservices.com であり，ファイル名までは全て一致している．動的オブジェクトであるためクエリパラメータは全

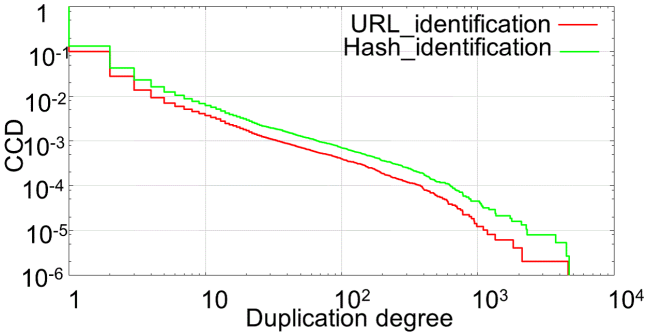

図 4 Complementary cumulative distribution of duplication factor of objects

表 1 Examples of objects with high duplication degree identified by URL

Duplication degree	URL
4861	https://www.google-analytics.com/analytics.js
2194	https://connect.facebook.net/en_US/fbevents.js
2180	https://securepubads.g.doubleclick.net/gpt/pubads_impl_199.js
1864	https://securepubads.g.doubleclick.net/gpt/pubads_impl_rendering_199.js
1708	https://tpc.googlesyndication.com/pagead/js/r20180423/r20110914/activeview/osd_listener.js
1130	https://www.googletagservices.com/tag/js/gpt.js

表 2 Examples of different URLs which were identified as same object by Hash identification

Duplication degree	URL
1728	http://www.googletagservices.com/tag/js/gpt.js
	https://www.googletagservices.com/tag/js/gpt.js?v=106a51d4732159384774994 62fbf1a9072909c13
	https://www.googletagservices.com/tag/js/gpt.js?xhr=1
	https://www.googletagservices.com/tag/js/gpt.js?bust=20180830100103JS_20181018154015344
	https://www.googletagservices.com/tag/js/gpt.js?googfc
	https://www.googletagservices.com/tag/js/gpt.js?bust=1538730195&JS_20170510120322
	https://www.googletagservices.com/tag/js/gpt.js?ver=4.9.8

て異なっているが，Hash 値は全て同一であるため，実際には同じオブジェクトであると考えられる．しかしながら，URL 識別では URL のみで識別するため，これらは異なるオブジェクトとして認識される．このことから Hash 識別の方がより正確にオブジェクトを識別できることが確認できる．

3.4 オブジェクトの 2 個組の共起度

まず，707,690 個のオブジェクトの中で重複度が 2 以上かつ同一 Web ページに出現しているオブジェクトから任意の 2 個組を作成する．1 つ以上の Web ページで出現した組に対し，7,604 の Web ページにおける共起度（出現 Web ページ数）の累積補分布を図 5 に示す．URL 識別では 3,304,528 個の 2 個組を作成し，そのうち 20%程度の 2 個組が 2 つ以上の Web ページに出現していた．Hash 識別では 4,186,537 個の 2 個組を作成し，そのうち 20%程度の 2 個組が 2 つ以上の Web ページに出現していた．オブジェクト 2 個組の共起度の分布にも冪乗則が観測され，裾野は広く，URL 識別では 0.05%程度の 2 個組は 100 以上の，0.005%程度の 2 個組は 500 以上の共起度を有していた．また Hash 識別では 0.1%程度の 2 個組は 100 以上の，0.01%程度の 2 個組は 500 以上の共起度を有していた．このことから少数のオブジェクト 2 個組は極めて高い共起度を有していることが確認できる．

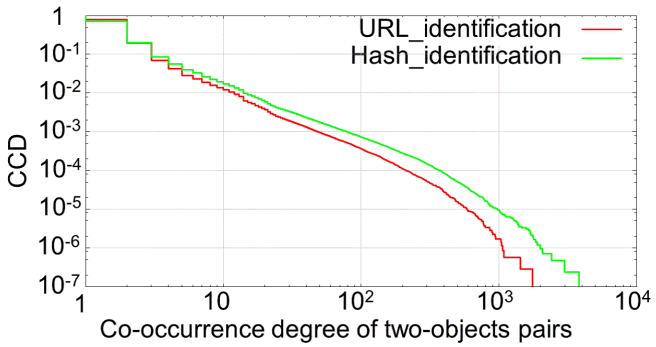

図 5 Complementary cumulative distribution of co-occurrence degree of two-objects pairs

次に各ページを構成する 2 個組の集合の共起度の平均値を求めた。図 6 に各 Web ページに対する平均共起度をプロットする．URL 識別で約 30%，Hash 識別で約 60%程度のページで，平均共起度が 100 を超えていた．少数の高重複度オブジェクトが多くのページで平均値を引き上げているため，このような全体的に高い平均共起度となっていると考えられる．また，少数の Web ページは極めて高い共起度を有していることが確認できる．Hash 識別における高共起度の組の例を表 3 に示す．多くの高共起度組は，google，facebook 等の高重複度オブジェクトによって構成されており，やはり動的オブジェクトが多いことが確認できる．

3.5 オブジェクトの 3 個組の共起度

次に任意の 3 個のオブジェクトを組み合わせた 3 個組について，共起度を分析する．ある N 個のオブジェクトから，任意の n 個のオブジェクトから構成される n 個組の数は n の増加に伴い指数的に増加する．そのため全てのオブジェクト組に対し共起度を計算することは困難である．しかし共起度の高い n 個組を構成する任意の $n-1$ 個のオブジェクト組はやはり共起度が高い．そこで高共起度の $n-1$ 個組のオブジェクトに対し任意の 1 個のオブジェクトを追加して構成される n 個組のみを分析対象とすることで，n の増加に伴う分析時間の増加を線形に抑えることができる．そこで前項で作成した 2 個組に対して，同一ページに存在する重複度 2 以上のオブジェクトを一つ追加し 3 個組を生成し，3 個組の共起度について分析した．

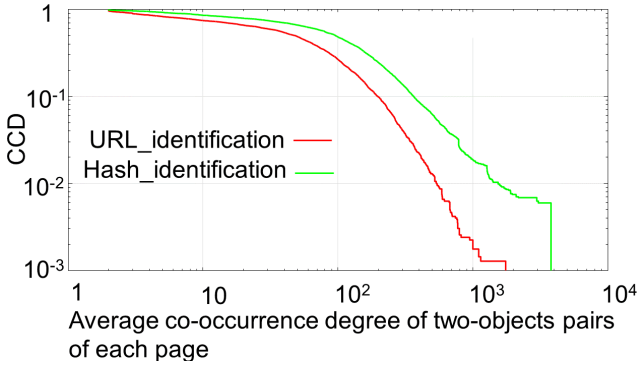

図 6 Complementary cumulative distribution of average of co-occurrence degree of two-objects pairs of each page

表 3 Examples of two-object pairs with high co-occurrence degree identified by Hash

Co-occurrence degree	URL
3805	https://www.google-analytics.com/collect
	https://ssl.google-analytics.com/analytics.js
2402	https://www.google-analytics.com/csi/batch
	http://sync.sharethis.com/ttd?uid=06266238-45aa-4a83-9840-4e9ac17b398a
2067	https://connect.facebook.net/en_US/fbevents.js
	https://www.facebook.com/tr?id=1572139883063794&ev=PageView-nfl

1 つ以上の Web ページで出現した組に対し，7,604 の Web ページにおける共起度（出現 Web ページ数）の累積補分布を図 7 に示す．2 個組の場合と同様，3 個組の共起度の分布の裾野は広く，URL 識別では 139,344,813 個の 3 個組を作成し，そのうち 30%程度の 3 個組が 2 つ以上の Web ページに出現していた．Hash 識別では 184,317,491 個の 3 個組を作成し，そのうち 30%程度の 2 個組が 2 つ以上の Web ページに出現していた．オブジェクト 2 個組の共起度の分布にも冪乗則が観測され，裾野は広く，URL 識別では 0.005%程度の 3 個組は 100 以上の，0.001%程度の 3 個組は 500 以上の共起度を有していた．また Hash 識別では 0.01%程度の 3 個組は 100 以上の，0.0005%程度の 3 個組は 500 以上の共起度を有していることが確認できる．2 個組の共起度に比べて全体的に共起度は低くなっているが，これは共起度は組の構成オブジェクトのうち重複度の低い方に依存するためであると考えられる．例えば，高重複度オブジェクト 2 個組に低重複度オブジェクトを加えても，考えうる共起度の最大値は低重複度オブジェクトの重複度になる．Hash 識別における高共起度の組の例を表 4 に示す．2 個組の高共起度組の例と同じように，高重複度の動的オブジェクトによる組が多く存在していることが確認できる．

4. ま と め

HTTP/2 の並列配信は同一の配信サーバから取得するオブ

ジェクトに対してのみ可能であり，Web 表示待ち時間の低減効果を高めるには，少数の配信サーバから多数のオブジェクトを取得する必要がある．そこであるオブジェクトの組に対し，その組が出現する Web ページの数を共起度と定義すると，共起度の高いオブジェクト組が優先的にキャッシュに残るようキャッシュ置換を行うことが望ましい．しかし共起度に基づくキャッシュ置換制御の効果は，実際の Web ページにおいてどの程度，オブジェクト間の共起現象が生じるかに依存する．そこで本稿では，共起度に基づくキャッシュ置換制御の可能性を明らかにするため，高人気の 8,000 の Web ページを構成する約 70 万個のオブジェクトを対象に，2 個組と 3 個組の共起度を計算し，Web ページの共起現象の程度を分析した．その結果，共起度の分布は冪乗則に従い，0.1%程度の 2 個組や 0.01%程度の 3 個組は 100 以上の共起度を有することを明らかにした．そのためこれら少数の共起度の高いオブジェクト組を優先的にキャッシュに残すキャッシュ置換制御の有効性が期待されることを確認した．今後は共起度の基づくキャッシュ置換制御法を検討する予定である．

謝辞　本研究成果は，SCAT 研究費助成 180047 の援助を受けたものである．ここに記して謝意を表す．

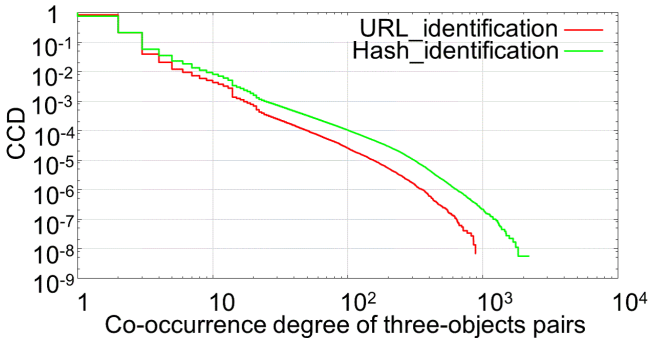

図 7　Complementary cumulative distribution of co-occurrence degree of three-objects pairs

表 4　Examples of three-object pairs with high co-occurrence degree identified by Hash

Co-occurrence degree	URL
2202	https://www.google-analytics.com/collect
	https://ssl.google-analytics.com/analytics.js
	http://sync.sharethis.com/ttd?uid=06266238-45aa-4a83-9840-4e9ac17b398a
1285	https://adservice.google.co.jp/adsid/integrator.js?domain=www.espn.com
	https://tpc.googlesyndication.com/pagead/js/r20181107/r20110914/activeview/osd_listener.js
	https://pagead2.googlesyndication.com/pagead/osd.jsl
1190	https://www.googletagservices.com/tag/js/gpt.js
	https://adservice.google.co.jp/adsid/integrator.js?domain=www.espn.com
	https://securepubads.g.doubleclick.net/gpt/pubads_impl_273.jsl

文　献

[1] Alexa, https://www.alexa.com/siteinfo
[2] R. Baeza-Yates, C. Castillo, and E. N. Efthimiadis, Characterization of national Web domains, ACM Trans. Internet Technology (TOIT), 7(2),Article No.9, 2007.
[3] M. Belshe, R. Peon, and M. Thomson, Hypertext Transfer ProtocolVersion 2 (HTTP/2), IETF RFC 7540, May 2015.
[4] L. Bent, M. Rabinovich, G. M. Voelker, Z. Xiao, Characterization of a Large Web Site Population with Implications for Content Delivery,ACM WWW 2004.
[5] M. Butkiewicz, H. V. Madhyastha, and V. Sekar, Understanding Website Complexity: Measurements, Metrics, and Implications, ACM IMC 2011.
[6] R. Fielding and J. Reschke, Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing, IETF RFC 7230, June 2014.
[7] P. Gill, M. Arlitt, N. Carlsson, and A. Mahanti, Characterizing Organizational Use of Web-based Services: Methodology, Challenges,Observations, and Insights, ACM Trans. The Web, 5(4),
[8] GitHub, Network Monitoring with PhantomJS, http://phantomjs.org/network-monitoring.html
[9] When seconds count. http://www.gomez.com/wp-content/downloads/GomezWebSpeedSurvey.pdf.
[10] Not as SPDY as You Thought, Guy’ s Pod, Thoughts and research on Web Performance & Secirity, Jun. 2012.
[11] Software is hard, http://www.softwareishard.com/blog/harviewer/
[12] S. Ihm and V. Pal, Towards Understanding Modern Web Traffic, ACM IMC 2011.
[13] M. Jiang, X. Luo, T. Miu, S. Hu, and W. Rao, Are HTTP/2 Servers Ready Yet?, IEEE ICDCS 2017.
[14] N. Kamiyama, Y. Nakano, and K. Shiomoto, Cache Replacement Based on Distance to Origin Servers, IEEE Transactions on Network and Service Management, Vol.13, Issue 4, pp. 848-859, Dec. 2016.
[15] D. Kumar, Z. Ma, Z. Durumeric, A. Mirian, J. Mason, J. A. Halderman, and M. Bailey, Security Challenges in an Increasingly Tangled Web, WWW 2017.
[16] J. Manzoor, I. Drago, and R. Sadre, How HTTP/2 is changing Web traffic and how to detect it, TMA 2017.
[17] J. Mickens, Silo: Exploiting JavaScript and DOM Storage for Faster Page Loads, USENIX WebApps 2010.
[18] E. Nygren, R. Sitaraman, and J. Sun, The Akamai Network: A Platform for High-Performance Internet Applications, ACM SIGOPS 2010.
[19] J. Ott, M. Sanchez, J. Rula, and F. Bustamante, Content Delivery and the Natural Evolution of DNS, ACM IMC 2012.
[20] F. Schneider, S. Agarwal, T. Alpcan, and A. Feldmann, The new web: characterizing AJAX traffic, ACM PAM 2008.
[21] A. Su, D. Choffnes, A. Kuzmanovic, and F. Bustamante, Drafting Behind Akamai: Inferring Network Conditions Based on CDN Redirections, ACM Trans. Networking, Vol. 17, No. 6, pp. 1752-1765, Dec. 2009.
[22] S. Sundaresan, N. Feamster, R. Teixeira, and N. Magharei, Characterizing and Mitigating Web Performance Bottlenecks in Broadband Access Networks, ACM IMC 2013.
[23] X. S. Wang, A. Balasubramanianm A. Krishnamurthy, and D. Wetherall, How speedy is SPDY?, NSDI 2013.
[24] Y. Zaki, J. Chen, T. Potsch, and T. Ahmad, Dissecting Web Latency in Ghana, ACM IMC 2014.
[25] T. Zimmermann, J. Ruth, B. Wolters, and O. Hohlfeld, How HTTP/2 Pushes the Web: An Empirical Study of HTTP/2 Server Push, IFIP Networking 2017.