

TCP 並列接続を用いたプログレッシブダウンロード における順序制御方式の実装

Implementation of Sequence Control Method in Progressive Download Using Parallel TCP Connections

1420180 平城 光雄 Mitsuo Heijo

指導教員 舟阪 淳一

1 はじめに

近年、効率的なコンテンツ配信を目的とした CDN でのコンテンツの分散配置が既に運用されている。このような同一のファイルが分散配置されている状況を利用して複数のサーバと同時に通信を行うことで、より高速なファイルの取得を実現する方式が提案されている [1][2]。しかし、既存の提案では動画再生等を考慮した順序制御は目的とされていない。本研究では、複数の性能が異なるサーバから TCP 接続を通して同一のファイルを分割取得する場合を想定し、要求送信時に各接続の性能を比較することで到着順序逆転を抑制する順序制御方式を提案する。

2 関連研究

既存研究では、複数のサーバからファイルを分割して並列ダウンロードし、順序通りにクライアントに提供する代理サーバが提案されている [2]。既存の提案である重複再要求方式では、非有効ブロックの個数がある閾値を超えた場合は別の接続に要求を再送信することで事後的に順序逆転に対処していた。この提案では、要求送信時における予防的な順序逆転への対策は行われていない。また、動画等の再生を考慮した評価も行われていない。

3 提案方式

確立した TCP 接続群の中で最高性能の接続には、最若番のブロックを要求する。ここでの最高性能の定義は使用回数が最多の接続とする。低性能の接続には前回の要求送信時からブロック到着までのブロック総受信回数の差分 D を計測し、 D に基づいて最若番でないブロックを要求する。 D は式 (1) から求める。 T はその時刻でのブロックの総受信回数、 $P[X]$ は接続 X の前回のブロック到着時におけるブロックの総受信回数、 N は接続の個数である。図 1 に 2 つの接続間に 2 倍の性能差がある場合の模式図を示す。 $t=0$ では遅延要求を行っていないので $t=1$ にブロック 1 が到着しても、ブロック 0 が到着していないのでブロック 1 は再生できない。 $t=2$ では $T=3$, $P[B]=0$, $N=2$ であるので接続 B に $D=2$ としてブロック要求を送信することで、ブロック 3-5 が順序通りに到着している様子を示している。

$$D = T - P[X] - N + 1 \quad (1)$$

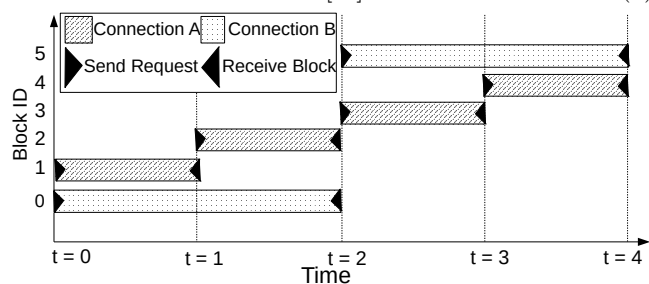


図 1: 遅延要求の模式図

4 実験と評価

従来方式と提案方式を実装した HTTP クライアントを用いて、テストベッドと公開ネットワークで評価を行なった。本稿では紙面の都合上、公開ネットワークでの結果のみを掲載する。公開ネットワークでの実験には利用のしやすさから Ubuntu のイメージファイルのパブリックミラーを利用した。サーバは国内 2 台、国外 7 台の合計 9 台利用した。事前に調査したミラーサーバにおける帯域性能より、国内のサーバと国外のサーバの帯域性能差はおよそ 6 倍から 46 倍程度であることがわかっていて、また、取得するファイルは、映像と音声合わせてビットレートが 40Mbps で長さが 150 秒の動画の取得を仮定し、754MByte のイメージファイルを選択した。実験では各方式 10 回ずつファイル取得を行い、比

較する。評価対象は、先行研究で提案されている重複再要求を実装した (NORMAL) と、重複再要求に加えて提案方式を実装した (DIFF) の 2 種類である。

評価項目は、初期バッファリング時間、バッファ内の非有効ブロックの数、平均遅延時間である。初期バッファリング時間とは遅延時間の最大値である。遅延時間とは各ブロックが円滑に再生できるタイミングからの取得の遅れとする。初期バッファリング時間が短くなれば再生開始時の待ち時間が短縮される。非有効ブロック数は取得したにもかかわらず、再生が不可能なブロックの数である。平均遅延時間は各ブロックが理想的な到着時刻からの正の遅延時間の平均値である。非有効ブロック数、平均遅延時間の値が小さいほど複数の TCP 接続を確立し得られた帯域をより有効に利用できているといえる。実験結果を表 1 に示す。値は平均値、標準偏差の組で表す。10 回試行の各ブロック ID における非有効ブロック数と遅延時間をプロットしたものを図 2, 3 に示す。

表 1: 各評価項目の平均値、標準偏差

方式	初期バッファリング時間 [s]	平均非有効ブロック数	平均遅延時間 [s]
NORMAL	6.203, 0.717	6.435, 0.657	1.550, 0.322
DIFF	3.130, 0.592	3.730, 0.390	0.866, 0.261

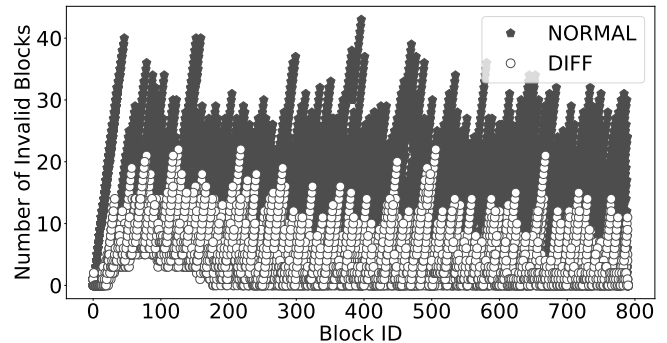


図 2: 非有効ブロック数

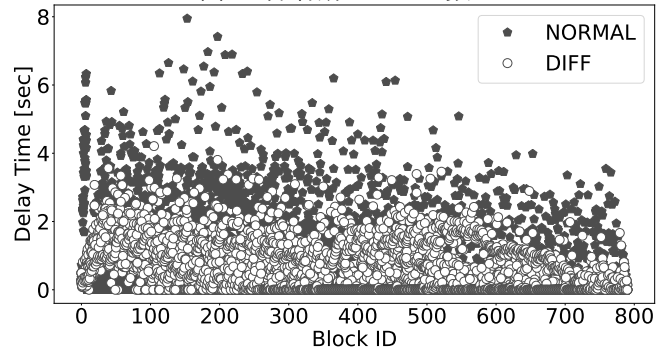


図 3: 遅延時間

表 1 より、公開ネットワークでの評価において提案方式は従来方式と比較して、約 50% の初期バッファリング時間の短縮、約 30% の非有効ブロック数の削減、約 30% の平均遅延時間の短縮ができたことが確認できる。また、図 2, 3 から、DIFF 方式のほうが非有効ブロック数は少なく、遅延時間も短い傾向にあることがわかる。

5 まとめ

複数のサーバから同一のファイルを並列に分割ダウンロードする際の順序逆転の発生を抑制する要求方式を提案し、実装した HTTP クライアントをテストベッドと公開ネットワークで評価し優位性を確認した。今後の課題としては、プロキシとして組み込んだシステムでの実際の動画の視聴体験まで含めた評価などが考えられる。

参考文献

- [1] J. Kim, Y. Chen, R. Khalili, D. Towsley, A. Feldmann, Multi-source Multipath HTTP (mHTTP): A Proposal, ACM SIGMETRICS Performance Evaluation Review, vol.42, no.1, pp.583-584, 2014.
- [2] J. Funasaka, A. Kawano, K. Ishida, Adaptive Parallel Downloading Method for Proxy Systems, IEICE Trans., vol.E90-B, no.4, pp.720-727, 2007.