Vol.2023-DPS-194 No.36
Vol.2023-CSEC-100 No.36
2023/3/7

動的なレート制御を持つアプリケーションの識別に関する考察

青木寛[1]　小津喬[1]　横山浩之[1]

概要：多様な通信が混在するトラヒックでは各アプリの通信フローに対して品質要求を適切に設定するのは困難であり，もし一律に品質要求を設定した場合，低遅延や高信頼性を要求する通信フローが他の通信フローと同じ扱いにされ輻輳の影響を受けやすくなりサービス継続に支障が出る恐れがある．そこで筆者らは，多様な通信が混在するトラヒックから通信フローを識別し，それぞれに適切な品質要求を設定することにより，サービス継続を保つ技術を研究している．本稿では，動的なレート制御機能を持つアプリケーションにおいて通信フローの識別を行う際，リンク容量の変化による挙動の変化によって識別精度が低下することを実測によって示し，その原因について考察する．

キーワード：QoS，アプリケーションカテゴリ識別，機械学習

A Study for Identification Applications with Dynamic Rate Control

HIROSHI AOKI[1]　TAKASHI OZU[1]　HIROYUKI YOKOYAMA[1]

Abstract: It is difficult to appropriately set the quality requirements for the communication flow of each application in the traffic mixed with various types of communication. If quality requirements are set uniformly, communication flows that require low delay and high reliability are treated the same as other communication flows, and are more likely to be affected by congestion, which may hinder service continuity. Therefore, we are researching a technology to maintain service continuity by identifying communication flows from the mixed traffic of various types of communication and setting appropriate quality requirements for each. In this paper, we show that the behavior of the application varies with the change of the link capacity, and the accuracy of application category identification decreases in actual traffic measurement data.

Keywords: QoS, Application category identification, Machine learning

1. はじめに

インターネット上のトラヒックは常に増大傾向にあるほか，低遅延や高信頼性を要求する各種アプリのサービス継続のため品質要求が多様化している．一般的に品質要求が高ければ高いほど，それを実現するにはネットワークリソースを多く使用する必要があり，ネットワークリソースを有効に利用するためには，各アプリの通信フローに対して品質要求を適切に設定する必要がある．しかし近年，セキュリティのために通信が暗号化されており，多様な通信が混在するトラヒックでは各アプリの通信フローに対して品質要求を適切に設定するのは困難であり，もし一律に品質要求を設定した場合，低遅延や高信頼性を要求する通信フローが他の通信フローと同じ扱いにされ輻輳の影響を受けやすくなりサービス継続に支障が出る恐れがある．

そこで筆者らは，多様な通信が混在するトラヒックから通信フローを識別し，それぞれに適切な品質要求を設定することにより，サービス継続を保つために適切な QoS を設定する技術を研究している．

本稿では，アプリケーションカテゴリを推定し通信フロー毎に品質要求 QoS flow ID を設定することを「フロー識別」とし，そのフロー識別方式が動的なレート制御を持つアプリケーションにおいて識別精度が低下する問題について考察する．

2. 関連研究

インターネットでは QoS 制御方法が検討され続けており，パケットヘッダーにより QoS 要求を指定する手法のほか，ネットワーク側で QoS を制御する手法が存在する．Differentiated Services (DiffServ)[1]は，パケットヘッダーにより QoS 要求を指定する手法であるが，エンドユーザーが常に適切な QoS 要求をパケットに与えるとは限らないため問題が生じることがある．例えばエンドユーザーが実際のアプリケーションよりも高い QoS を設定することを防ぐことは困難である．ネットワーク事業者の観点からは QoS 要求の検証方法が必要になる．

一方，パケットヘッダーの QoS 要求を使わずに，インターネットトラヒックのアプリケーション推定について多くの研究が行われている．過去の研究[2]では，IP アドレス，ポート番号，パケット長，ジッター，TCP ヘッダーなどのフロー特性を組み合わせたランダム フォレストのような単純な機械学習手法により，約 80%の精度でアプリケーションを識別している．また，別の研究[3]では，パケットサイズがパターン認識で使用される画像データに似ているという事実に焦点を当て，ディープラーニングを使用して特徴学習，プロトコル識別，異常プロトコル検出アプリケーションを実現している．これらの手法は主に異常検出に焦点を当てており，アプリケーションの識別精度自体には改善の余地がある．他のいくつかの研究[4][5]では，トラヒックパターンのバースト特性に焦点を当てて，アプリケーシ

1 株式会社国際電気通信基礎技術研究所
Adaptive Communications Research Laboratories

Vol.2023-DPS-194 No.36
Vol.2023-CSEC-100 No.36
2023/3/7

ョンの正確な推定を実現している．これらの手法では比較的大きな静的な閾値によってバーストを分離し，バーストの時間長は数秒程度となっている．これは人間の行動によって引き起こされるアプリケーションの長期的な動作を分析することを意味しているが，アプリケーションの種類を識別するのに数十秒かかることになる．

筆者らの手法[6]では，パケット間隔に応じてバーストの終了を判断する閾値を調整したうえでバースト特徴に基づいたアプリケーション推定を行っているため，バースト特性を 1 秒未満の時間スケールで分析することでき，様々なアプリケーションを 10 秒以内に識別できる．そのため，リアルタイムの QoS 制御に非常に適しているといえるが，短い時間長のバーストを扱うため．動的なレート制御を行うアプリケーションにおいて通信速度の影響による識別精度低下が少なくない．これはアプリケーションが通信速度に応じて動作を変化させた結果バーストの特徴が変化するのも原因の一つである．

本稿では動的なレート制御を行うアプリケーションに対するフロー識別において，リンク容量の変化による精度低下を報告し，その内容について考察する．

3. アプリケーションカテゴリ識別手法

本章では，筆者らが提案しているアプリケーションカテゴリ識別手法[6]の概要を示す．筆者らは，ネットワークに流入するトラフィックをフローに分類し，フロー毎のバースト性に基づいてアプリの識別を行う方法を研究している．ここでフローとは，送受信 IP アドレス及びポート番号が同じパケットの集合を指す．アプリの種類はサーバー側の IP アドレス，ポート番号からある程度は特定できるため，サーバーと QoS flow ID を紐づける「変換表」を基にフロー識別を行うことを想定している．その「変換表」の作成方法を図 1 に示す．

サーバーが既知のアプリの通信フローであればアドレスフィルタを設定することにより「変換表」に反映できるが，そうでない通信フローはトラヒックの特徴から QoS Flow ID に振り分ける．本方式では，アプリケーションが既知であるトラヒックを教師データとして用い，トラヒックの特徴によりフロー識別する．そのため，トラヒックの特徴を定量化し，その定量化された特徴を基に機械学習による識別を行っている．

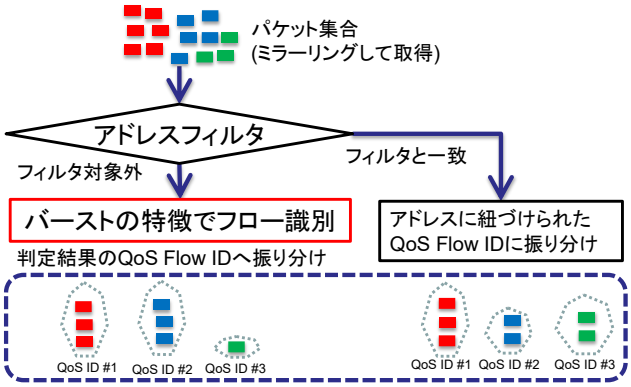

図 1 フロー識別の「変換表」作成方法[7]

アプリはその動作に応じて通信を行うため，パケット毎の特徴量やトラヒック全体の特徴ではアプリの使用状況による偏りが大きく判定困難である．トラヒックは多くの場合，図 2 に示すようにバースト状に発生し，そのバーストの間隔はアプリの操作や環境に応じて大きく変わるからである．そこで，筆者らはトラヒックがバースト状に発生するのはアプリの動作によるものであり，アプリを特定するのにそのバースト自体の特徴が有用であると考えて，トラヒックの特徴をそのバーストの特徴量により決定する．

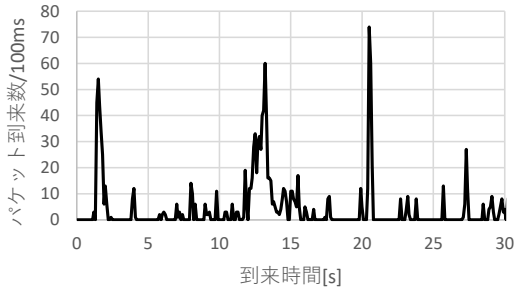

図 2 ニュースサイト閲覧でのパケット到来の時間分布

トラヒックのバーストの特徴はそれを流すアプリによって異なる．例えば図 3 に示すように，HTTPS によるファイルダウンロードであれば MTU に近いサイズのパケットの連続で構成され，音声通話であれば比較的小さいパケットが数 ms 程度の間隔を持った形で構成される．

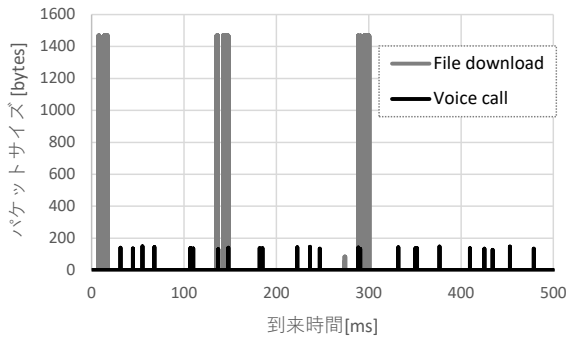

図 3 ファイルダウンロードと音声通話におけるパケット到来時間とパケットサイズの関係

Vol.2023-DPS-194 No.36
Vol.2023-CSEC-100 No.36
2023/3/7

あるフローから時間的に連続する IP パケットで構成されるバーストを切り出すには，パケット到来間隔を用いてバーストの切れ目を設定する必要がある．そこで図 4 に示すようにパケットの到来間隔が閾値 Th より大きい場合，その部分がバーストの切れ目と判定する．

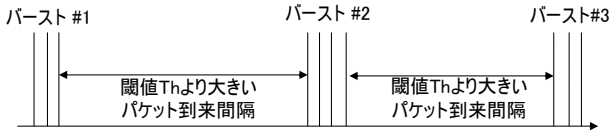

図 4 バーストの定義

この閾値 Th によってバーストがどの部分で切れるかは図 5 に示すように変化する．トラヒックのバースト状態はアプリやその使用状況に応じて大きく変わるため，閾値 Th を固定値とすると適切な特徴量を持つバーストの切れ目を設定することができない恐れがあり，忘却係数を用いた式(1)でパケット毎に更新する閾値 Th によりバーストの切れ目の判定を行った．ただし，タイムアウト値として Th の最大値を 300ms，パケット間隔の最小値として Th の最小値を 4ms とし，バーストの時間長が 2000ms を超える状態になる場合は，その時点でバーストを切り出す．

$$Th=\alpha\times\Delta t+(1-\alpha)\times Th\_OLD \quad (1)$$

α：忘却係数(今回は α=0.3 を使用)
Δt：パケットの到達間隔
Th_OLD：更新前のバースト閾値

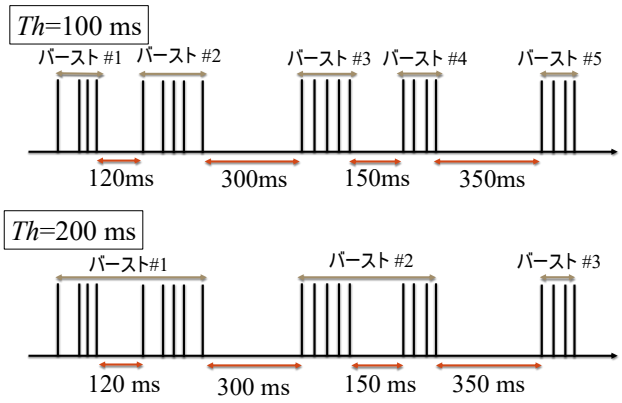

図 5 バースト閾値に応じて変わるバーストの切れ目

具体的には同一のサーバーとの通信フローの時間的に連続して通信される複数パケットを時間間隔がこの閾値 Th 以上空いたときに区切って 1 つの「バースト」として扱い，これをフロー識別の識別単位とする．

そのバースト毎に，バーストの時間長，パケット数，平均パケットサイズを計算し，これを「バースト」の特徴とする．

バーストの特徴から QoS Flow ID に振り分ける処理は自動で調整できることが望ましいため，機械学習を用いてフロー識別を行う．

機械学習のアルゴリズムとして，各特徴量の重要度(importance)を確認することが可能である random forest を使用する．本稿の評価では機械学習のオープンソースライブラリ scikit-learn[8]を用い，random forest のハイパーパラメータは乱数シード以外についてライブラリのデフォルト値[9]を使用する．説明変数は「バースト」の特徴として定量化する，バーストの時間長，パケット数，平均パケットサイズの 3 つの値を使用する．

アプリによっては一時的に通常と異なる特徴を持つトラヒックを流すことがあり，そのようなトラヒックのバーストは他のバーストと特徴が異なるため，本来のアプリ種別(Application Category)ではないものに誤判定されやすくなる．また，特徴が似ているアプリ種別同士では誤判定が生じやすい．そのような状況でも正しく判定されるようにするため，複数のバーストの判定結果を基にして，多数決で採取的な判定結果を決定する．

具体的には時間的に連続である複数のバースト N 個分の識別結果に対して多数決をとって，これを最終的なフロー識別の識別結果とする．

4. リンク容量の変化による識別精度の低下

本章では，リンク容量の変化によりバーストの特徴が変化すると，アプリケーションカテゴリの識別精度が低下することを実測データによって示す．テレビ会議／音声通話／動画配信等の各種アプリのトラヒックを測定して学習データを収集したうえで，帯域を制限した場合のアプリケーションカテゴリ識別を行い，その判定結果を確認する．

4.1 トラヒックのキャプチャ

3GPP における 5QI と QoS 目標値との対応表[10]を参考に，通話，動画閲覧等のアプリの暗号化されたトラヒックを 7 種類のアプリケーションカテゴリに分類し，表 1 に示す動作をさせてトラヒックを実測した．

表 1 評価対象のアプリケーションカテゴリ

カテゴリ	測定時の操作内容
音声通話	音声通話アプリ(facetime, Line, skype, ZOOM)で電話
ビデオ通話	Web 会議(ZOOM, WebEX)を行う
リアルタイムゲーム	リアルタイム応答が要求される FPS のオンラインゲーム(GTA online)で対戦
Push To Talk voice	Push To Talk voice ができるアプリで何回かやり取り

機械制御	模擬的に周期的に一定間隔でパケットの送受信を行うソフトで測定
バッファビデオ	動的なレート制御を行うアプリケーションの例として YouTube で動画を閲覧
その他	コマンド選択制等でラグが比較的影響しない双方向ゲームで対戦 ニュースサイト，ブログ等の閲覧 画像ベースのサイト閲覧 HTTPS で大容量ファイルのダウンロード，アップロード メールの送信，受信 チャットアプリでテキスト，画像，その他データ等を送受信

測定では，図 6 に示すような構成において表 1 の動作を行うアプリをそれぞれ評価用端末にインストールし，それを動作させた際に送受信されるパケットを収集装置で稼働するネットワークプロトコルアナライザーソフト(Wireshark[11])によってキャプチャした．

動的なレート制御を行うアプリケーションの例としてバッファビデオにおいては帯域制限を行うことが可能なスイッチングハブを使用してインターネットとの最大帯域幅を設定した条件でのキャプチャを行った．更に比較対象として音声通話，ビデオ通話，リアルタイムゲーム，ファイルダウンロードにおいても最大帯域幅を設定した評価を行った．ただし，いずれのアプリケーションカテゴリにおいても帯域制限を付加してないトラヒックを学習データに使用した．

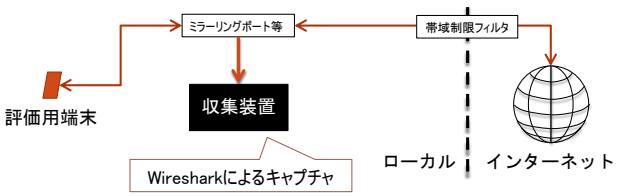

図 6 トラヒックのキャプチャ方法

複数のポート番号，サーバーと送受信するアプリの場合，それぞれのポート番号，サーバーとの通信毎にフローとして扱う．そのフロー毎に前述の方法でバーストを切り出し，このバースト毎に特徴量を算出した．

今回のフロー識別対象は DL のみとした．測定の環境上，目的外のトラヒックも少量であるがキャプチャされるため，そのような通信フローはノイズとしてフロー識別の対象外とすることにし，バースト数が少ないフローを今回の評価から除去した．

4.2 フロー識別結果

表 2～表 6 に，バッファビデオ，音声通話，ビデオ通話，リアルタイムゲーム，ファイルダウンロードのカテゴリ毎に帯域制限を付加した場合の識別精度と最も誤判定された先のカテゴリとその判定割合を示す．なお，ここでの識別精度はバースト単体での識別精度である．

学習データのトラヒックとは別に取得した帯域制限を付加していないトラヒックと 8, 4, 2, 1Mbps の帯域制限をそれぞれ付加したトラヒックをテストデータとしてアプリケーションカテゴリを推定した．ただし，リアルタイムゲームでは通信量が 0.3Mbps 未満であったため帯域制限を 0.5, 0.3Mbps とした．

今回のフロー識別において，帯域制限を付加してないトラヒックを学習データとしているため，帯域制限を付加したトラヒックでは付加していないトラヒックに比べて識別精度が低下すると予想され，リアルタイムゲームとファイルダウンロードを除いて帯域制限を掛けるほど識別精度が低下していることが確認できる．特に動的なレート制御を行うアプリケーションであるバッファビデオにおいては帯域制限を付加したトラヒックのほとんどがビデオ通話やその他に誤判定されている．このように動的なレート制御を持つアプリケーションにおいては帯域制限により精度が大きく低下する．

5. 考察

バッファビデオ(表 2)やビデオ通話(表 4)では帯域制限を付加したトラヒックにおいて識別精度が低下する一方，ファイルダウンロード(表 6)のように精度がほとんど低下しないアプリケーションカテゴリもある．

アプリケーションカテゴリの識別ではバーストの時間長，パケット数，平均パケットサイズといった 3 つの値のバースト特徴を使用しており，このバーストの特徴は学習データに使用した帯域制限を付加していないトラヒックとテストデータに使用した帯域制限を付加したトラヒックで異なることが識別精度の低下原因だと考えられる．

図 7~図 9 に帯域制限によるバースト特徴の変化を示す．平均パケットサイズは図 7 に示すように帯域制限を付加しても大きな変化が見られないが，平均パケット到来間隔は図 8 に示すように帯域制限の帯域幅が狭いほど大きくなる傾向が確認できる．平均バースト内パケット数においては図 9 に示すようにバッファビデオやファイルダウンロードで変化する一方，ビデオ通話では変化が小さい．

Vol.2023-DPS-194 No.36
Vol.2023-CSEC-100 No.36
2023/3/7

表 2 バッファビデオのアプリケーションカテゴリ識別結果の割合[%]

帯域幅	音声通話	ビデオ通話	リアルタイムゲーム	その他	Push To Talk	機械制御	バッファビデオ
無制限	0.0	10.5	0.0	10.5	0.0	0.0	78.9
8Mbps	0.1	25.9	0.0	73.9	0.0	0.0	0.1
4Mbps	0.0	93.4	0.0	6.6	0.0	0.0	0.0
2Mbps	0.0	2.9	0.0	97.1	0.0	0.0	0.0
1Mbps	0.3	57.8	0.2	41.7	0.0	0.0	0.0

表 3 音声通話のアプリケーションカテゴリ識別結果の割合[%]

帯域幅	音声通話	ビデオ通話	リアルタイムゲーム	その他	Push To Talk	機械制御	バッファビデオ
無制限	66.1	20.4	10.0	1.7	1.8	0.0	0.0
8Mbps	60.2	8.2	28.7	1.8	1.0	0.0	0.0
4Mbps	53.5	8.7	35.1	2.0	0.7	0.0	0.0
2Mbps	51.2	9.4	36.5	2.1	0.8	0.0	0.0
1Mbps	60.7	10.2	25.7	2.7	0.7	0.0	0.0

表 4 ビデオ通話のアプリケーションカテゴリ識別結果の割合[%]

帯域幅	音声通話	ビデオ通話	リアルタイムゲーム	その他	Push To Talk	機械制御	バッファビデオ
無制限	0.1	99.6	0.1	0.2	0.0	0.0	0.0
8Mbps	5.5	90.3	3.3	0.8	0.1	0.0	0.0
4Mbps	10.1	84.2	4.9	0.8	0.0	0.0	0.0
2Mbps	11.1	81.9	5.9	1.0	0.0	0.0	0.0
1Mbps	17.5	75.4	5.6	1.5	0.0	0.0	0.0

表 5 リアルタイムゲームのアプリケーションカテゴリ識別結果の割合[%]

帯域幅	音声通話	ビデオ通話	リアルタイムゲーム	その他	Push To Talk	機械制御	バッファビデオ
無制限	9.8	3.3	81.0	4.6	1.3	0.0	0.0
0.5Mbps	20.5	1.7	76.4	0.1	1.3	0.0	0.0
0.3Mbps	9.4	3.1	86.3	0.1	1.1	0.0	0.0

表 6 ファイルダウンロード(その他)のアプリケーションカテゴリ識別結果の割合[%]

帯域幅	音声通話	ビデオ通話	リアルタイムゲーム	その他	Push To Talk	機械制御	バッファビデオ
無制限	0.0	0.0	0.0	100.0	0.0	0.0	0.0
8Mbps	0.0	0.1	0.0	99.9	0.0	0.0	0.1
4Mbps	0.0	0.1	0.0	99.8	0.0	0.0	0.1
2Mbps	0.0	0.0	0.0	100.0	0.0	0.0	0.0
1Mbps	0.0	0.0	0.0	100.0	0.0	0.0	0.0

Vol.2023-DPS-194 No.36
Vol.2023-CSEC-100 No.36
2023/3/7

このようにアプリケーションカテゴリ毎に帯域制限によるバースト特徴の変化が異なるものの，ファイルダウンロードのように精度がほとんど低下しない場合でもバーストの特徴が変化している．帯域制限を付加した場合のバースト特徴が帯域制限を付加していない場合のバースト特徴から外れることになるが，本来とは異なるアプリケーションカテゴリのバースト特徴に近くなれば識別精度が低下し，本来のアプリケーションカテゴリのバースト特性のほうが近かった場合は識別精度が低下しないと考えられる．バッファビデオのように本来のアプリケーションカテゴリとは異なるカテゴリにバースト特徴が近くなるのは，動的なレート制御による動作変化が影響している可能性がある．

ここで誤判定最多のカテゴリを見てみると，バッファビデオでは表 2 に示すように「その他」や「ビデオ通話」が主な誤判定先になっている一方，ビデオ通話では「音声通話」が誤判定先になっている．これらのアプリケーションカテゴリのバースト特徴のほうが本来のアプリケーションカテゴリでのバースト特徴より近い値だったためと思われる．

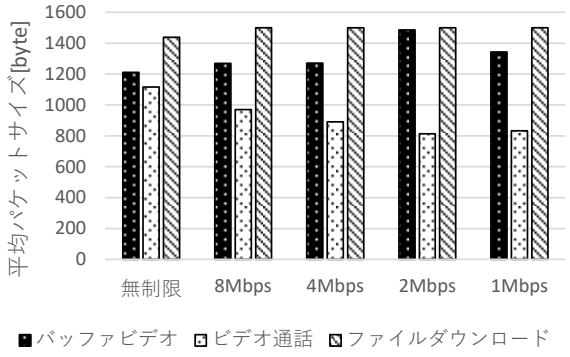

図 7 帯域制限による平均パケットサイズの変化

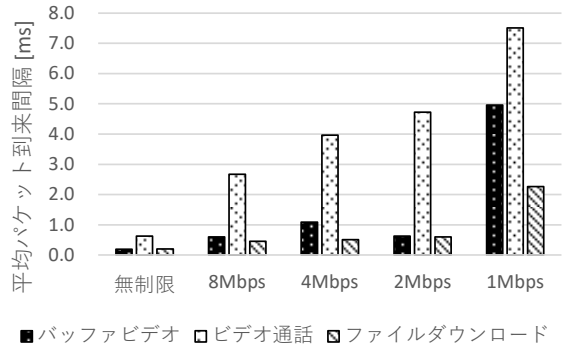

図 8 帯域制限による平均パケット到来間隔

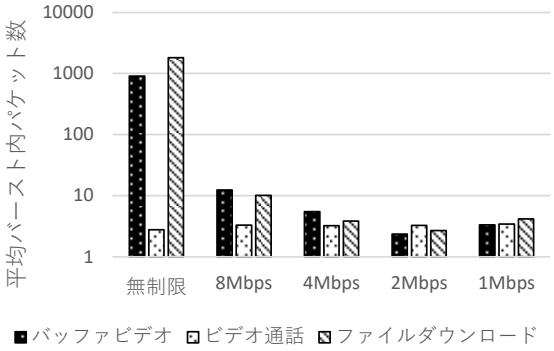

図 9 帯域制限による平均バースト内パケット数

6. まとめ

本稿では，動的なレート制御を行うアプリケーションにおけるフロー識別の精度が帯域制限により低下することを示し，その原因であるバースト特徴の変化を確認した． 同一のアプリケーション，同じ内容の通信であっても通信速度等でバーストの特徴が変化するため，より多様なネットワーク環境に対応する手法を今後，検討する必要がある．

謝辞 本研究は総務省委託研究「第 5 世代移動通信システムの更なる高度化に向けた研究開発(JPJ000254)」の成果の一部である．

参考文献

[1] K. Nichols, S. Blake, F. Baker, and D. Black, “Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers,” RFC 2474, December 1998.
[2] A.Nakao, P. Du, and T. Iwai, “Application Specific Slicing for MVNO through Software-Defined Data Plane Enhancing SDN,” IEICE Trans. Communications, Vol.98, No.11, pp.2111-2120 (2015)
[3] Zhanyi Wang. The applications of deep learning on traffic identification. BlackHat USA, 2015.
[4] V. F. Taylor, R. Spolaor, M. Conti and I. Martinovic, “Robust Smartphone App Identification via Encrypted Network Traffic Analysis,” in IEEE Transactions on Information Forensics and Security, vol. 13, no. 1, pp. 63-78, Jan. 2018.
[5] G. Aceto, D. Ciuonzo, A. Montieri and A. Pescapè, “Multi-classification approaches for classifying mobile app traffic,” J. Netw. Comput. Appl., vol. 103, pp. 131-145, Feb. 2018.
[6] H. Aoki, T. Ozu, A. Hasegawa and H. Yokoyama, "A Study on Application Category Estimation Method Based on Burst Characteristics of Communication," 2022 IEEE 12th Annual Computing and Communication Workshop and Conference (CCWC), 2022, pp. 0184-0190.
[7] 青木他，“適応型通信フロー識別方式の検討”， ソサイエティ大会 B-5-21，2021 年 9 月．
[8] https://scikit-learn.org
[9] https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html
[10] 3GPP TS 23.501 V16.4.0 (2020-03)
[11] https://www.wireshark.org