

バッファリング遅延を考慮した低消費電力ネットワーク制御の性能評価

Performance Evaluation of Low Power Network Control Considering Buffering Delay Time

高瀬 優人[†] 中山 明[‡] ザビル・サラウディン・ムハマド・サリム[§] 内海 哲史[‡]
Yuto Takase Akira Nakayama Salahuddin Muhammad Salim Zabir Satoshi Utsumi

1. はじめに

ICT技術、特にインターネットを含むネットワークは近年の情報社会の中核である。2017年時点でのインターネットの利用人口は約36億人と見積もられ、これは世界人口の約48%に相当する[1][2]。ネットワークの普及に伴い、それに接続されるICT機器による消費電力の増大が指摘されている。サーバファームや移動体通信システムにおいては、4、5年ごとに消費電力が倍増する傾向が見られ[3][4]、2007年の時点で世界全体の電力消費の1%以上がサーバファームにより消費されている[5]。この問題は日本も例外ではなく、データセンタ、ネットワークによる消費電力は2025年に約1,500億kwh/年に達すると予想されており、これは2010年比で約2倍である[6]。

消費電力の増加は高度に情報化された昨今の社会において、様々な問題を引き起こす可能性がある。ここでは日本における環境負荷、災害への備え、IoTの普及の三点から問題提起を行う。

まず環境負荷であるが、日本における発電手段の中で、依存度が最も高いLNG(液化天然ガス)火力発電[7]を用いて、データセンタ、ネットワークの消費する電力をすべて賄うと仮定する。LNG火力発電が排出する二酸化炭素は599g/kwhであるから[8]、2025年には約8,985万t/年の二酸化炭素がデータセンタ、ネットワークの電力消費により排出されると予想される。2017年度の日本における二酸化炭素総排出量である14億800万t/年[9]と比較しても、これは約6%強を占めるほどの割合である。日本においては、二酸化炭素総排出量は2013年度以降、年々減少傾向にあり[10]、今後データセンタ、ネットワークなどICT機器の省電力化が図られなければ、実際の2025年にはICT機器由来の二酸化炭素排出量の割合がさらに増えると考えられる。

続いて災害への備えの観点である。災害時には電力供給が限定的になる、あるいは喪失する状況がしばしば起きうる。具体的には2011年の東日本大震災後に発生した計画停電[11]や、2018年の北海道胆振東部地震で発生したブラックアウト[12]などである。このような場合には、残された発電・送電施設やデバイスに搭載されたバッテリーなどの電力資源を効率的に用いるこ

とがネットワークを維持する上で不可欠となる。

最後にIoT(Internet of Things)の普及である。従来のインターネットはサーバやネットワーク機器、パソコンやスマートフォンなどの各種端末が接続されていた。しかし近年、スマート家電などの普及に伴い、IoTの考え方が急速に普及している[13]。IoTを構成する機器にはセンサなどの単機能のデバイスも含まれる。そういったデバイスの中には小容量のバッテリーで動作するものや、環境中から熱・電磁波・振動などのエネルギーを収集して電力源とするエネルギーハーベスティングを行うものも存在する[14]。このようなデバイスは効率的に電力資源を利用しなければならないため、ネットワークの省電力化が特に重要であると考えられる。

これらの課題を解決するためにも、ネットワークの省電力化は喫緊の課題である。そのためには複数個の packets を集約して送信するバースト送信が有用であるが、遅延時間とのトレードオフの関係であることが先行研究により知られている[15]。遅延時間の増加はネットワークを利用するサービスの品質を低下させる。特に近年はネットワーク機器に搭載されるメモリの低価格化、大容量化に伴い、Bufferbloat[16]と呼ばれるバッファリング遅延の増加が問題視されている。遅延時間は通信性能における重要な評価指標のひとつである。したがって省電力化だけではなく、遅延時間の抑制についても考慮する必要がある。

本稿では、packetsのバースト送信を用いてネットワーク機器のスリープ機能を効果的に利用する、遅延時間を考慮した低消費電力ネットワーク制御の提案と性能評価を目的とする。

本稿の構成は以下の通りである。2章で本研究で提案する低消費電力ネットワーク制御の概要を示す。3章では、シミュレーション実験によって提案方式の性能評価を行う。最後に、4章で本稿をまとめる。

2. 提案方式：低消費電力ネットワーク制御

本章では、ネットワークの省電力化を実現するために回線のスリープ機能を効果的に利用する低消費電力ネットワーク制御を提案する。

2.1 概要

提案方式のネットワーク内の各ノードはコンテンツと呼ばれる単位でpacketsのバースト送信を行う。コンテンツとは、提案方式のネットワークにおけるデータ転送の単位であり、1個以上のpacketsから構成される。コンテンツサイズ M [packets] は可変であり、 $M \geq 1$ の範囲で調整される。図1にpacketsごとに送信する従来方式と提案方式の比較図を示す。斜線は回線にお

[†]福島大学大学院 共生システム理工学研究科, Graduate School of Symbiotic Systems Science and Technology, Fukushima University

[‡]福島大学 理工学群 共生システム理工学類, Faculty of Symbiotic Systems Science, Cluster of Science and Technology, Fukushima University

[§]国立高等専門学校機構 鶴岡工業高等専門学校 創造工学科, Department of Creative Engineering, Tsuruoka College, National Institute of Technology

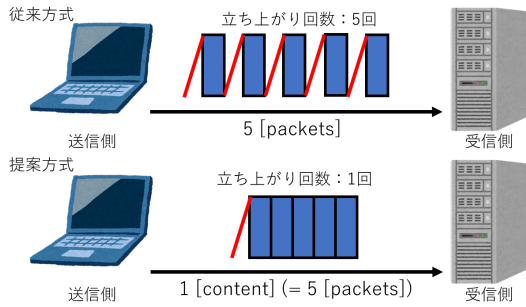


図 1: 従来方式と提案方式の比較

けるスリープ状態からの立ち上がりを示す。スリープ状態からの立ち上がりに係る電力は比較的大きい。この立ち上がり回数が提案方式では削減されていることがわかる。

提案方式のネットワークは、エッジノードとコアノードから構成される。送信側エッジノードはパケットを集約してコンテンツにする。受信側エッジノードはコンテンツを分離してパケットに戻し、コンテンツに対する確認応答を送信側エッジノードに送信する。コアノードはコンテンツに集約されたパケットの送受信を行う。図 2 に提案方式のネットワークの構成を示す。

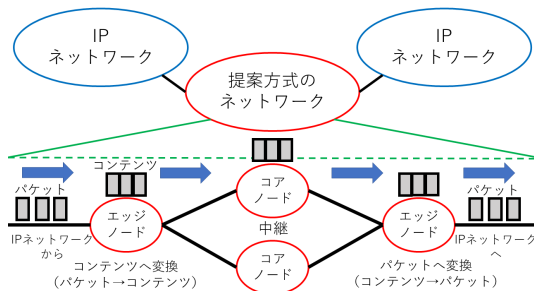


図 2: 提案方式のネットワーク構成

提案方式のネットワークは、上記のようにエッジノード、コアノードで役割分担をする。送信側エッジノードはコンテンツサイズ分のパケットが到着するまで、先に到着したパケットの送信を待ち、コンテンツサイズ分のパケットがそろい次第、直ちにパケットをバースト送信する。コアノードはコンテンツを維持したバースト送信を行うため、コンテンツを構成するパケットが全てそろいまで、先に到着したパケットの送信を待ち、そろい次第、直ちにパケットをバースト送信する。受信側エッジノードはコンテンツを構成する全てのパケットを受信すると、コンテンツに対する確認応答を送信側エッジノードに送信する。

2.2 エッジノードにおけるコンテンツサイズの調整アルゴリズム

提案方式では、遅延時間に基づいてコンテンツサイズを調整する。コンテンツサイズが大きいほどネット

Algorithm 1 コンテンツサイズの調整アルゴリズム

```

1:  $Packets\_Counter$  [packets] : これまでにいくつのパケットを送信したかを記録するカウンタ
2:  $Controlled\_delay$  [sec] : 現在の controlled delay
3:  $M$  [packets] : 現在のコンテンツサイズ
4:  $Target\_delay \leftarrow 0.02$  [sec] : target delay
5:  $\alpha \leftarrow 1 \times 10^3$  [packets] // コンテンツサイズを  $\beta$  増加させるまでに送信するパケット数
6:  $\beta \leftarrow 1$  [packet] // コンテンツサイズ  $M$  の増加・減少幅
7: On sending each Contents :
8: if  $Controlled\_delay > Target\_delay$  then  $\triangleright$  (A)
9:    $M \leftarrow M - \beta$ 
10: else
11:    $Packets\_Counter \leftarrow Packets\_Counter + M$ 
12:   if  $Packets\_Counter \geq \alpha$  then  $\triangleright$  (B)
13:      $M \leftarrow M + \beta$ 
14:      $Packets\_Counter \leftarrow 0$ 
15:   end if
16: end if

```

ワークの省電力効果が高くなる一方、遅延時間が増加する [15]。不要に遅延時間が増加すればネットワークを利用するサービスの品質は低下するため、遅延時間にしきい値 (target delay) を設ける。(I) エッジノードでのコンテンツのためのパケット待ち時間と (II) 提案方式のネットワーク全体におけるバッファリング遅延について、その合計を controlled delay と定義する。提案方式の送信側エッジノードは、controlled delay が target delay と等しくなることを目標とする、コンテンツサイズ M の調整アルゴリズムを持つ。ただし、 M は自然数とする。

controlled delay は観測値であり、以下の式にて導出する。

$$\text{controlled delay} = \delta + \omega \quad (1)$$

ここで、 δ [sec] とは観測されるネットワーク全体における平均バッファリング遅延を示す。また、 ω [sec] とは送信側エッジノードにコンテンツサイズ M のパケットが揃うまでの時間であり、 M を平均パケット到着率で除した値となる。target delay は controlled delay の目標値であり、定数として設定する。

ボトルネックリンクにおけるパケット到着率の変化に対応できるよう、エッジノードは Algorithm1 のアルゴリズムに従いコンテンツサイズを調整する。controlled delay が target delay 以下であれば、 α [packets] 分のコンテンツを送信するごとに M を β [packets] 増加する。controlled delay が target delay を超えたことを検出すると、すぐに M を β 減少する。すなわち M を AIAD (Additive-Increase/Additive-Decrease) で調整するアルゴリズムである。本研究では $\alpha = 1 \times 10^3$ [packets]、 $\beta = 1$ [packet] とする。

以下では、各変数の初期値確定後の Algorithm1 について説明する。

Algorithm1 はコンテンツ送信時に呼び出され、コンテンツサイズ M を調整する。

(A) は、コンテンツサイズ M を減少させる基準

を満たしているかを判断する。 $Controlled_delay$ が $Target_delay$ より大きければ、 M を β 減少させる。

(B) は、コンテンツサイズ M を減少させない場合に、 M を増加させる基準を満たしているかを判断する。 $Packets_Counter$ が α [packets] 以上ならば、 M を β 増加させる。

3. 性能評価

本章では、図3のようなネットワークシナリオを想定し、2章で説明した提案方式を簡易的に評価する。

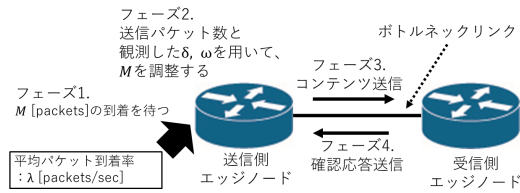


図3: シミュレータのネットワークシナリオ(注1)

3.1 シミュレータの簡易実装

図3のネットワークシナリオにおいて、パケットの到着はポアソン分布に従い、単位時間あたりの平均パケット到着率を λ [packets/sec] とする。

送受信側エッジノードについて、4つのフェーズがある。

フェーズ1は、送信側エッジノードがコンテンツを構成するパケットの到着を待つ段階である。送信側エッジノードにパケットが到着すると、パケットをバッファに格納する。バッファに格納されたパケット数が M を超えると、フェーズ2に移行する。

フェーズ2は、コンテンツ送信前に、コンテンツサイズ M を調整する段階である。観測した平均バッファリング遅延 δ と平均パケット待ち時間 ω から controlled delay を計算し、それを用いて Algorithm1 の処理を行い、 M を調整する。

フェーズ3は、送信側エッジノードがバッファ内のパケットをコンテンツとして受信側エッジノードにバースト送信する段階である。コンテンツの送信完了後、フェーズ4に移行する。

フェーズ4は、受信側エッジノードがコンテンツを構成する全てのパケットを受信し、送信側エッジノードへ確認応答を送信する段階である。

3.2 シミュレーション実験の概要

シミュレーション実験により、図3に示したネットワークシナリオにおいて、2章で提案したネットワー

ク制御の性能を評価する。シミュレーション実験に用いた各パラメタの記号とその意味及び、具体的な数値を表1にまとめる[17]。

提案方式と従来方式のそれぞれについて、図3のネットワークシナリオにおいてシミュレーション実験を行う。シミュレーション実験において、提案方式は2章で示したコンテンツサイズの調整アルゴリズムによって動的に M を調整する。従来方式はコンテンツサイズを常に $M = 1$ に固定して動作する。従来方式には、図3におけるフェーズ2はない。シミュレーション時間は100 [sec] で、エラーによるパケットロスは発生しない。target delay は0.02 [sec] に定める。

その結果について、シミュレーション時間中に生じる瞬時の消費電力 P [W] の平均値として平均消費電力 $E(P)$ [W] を算出し、従来方式の $E(P)$ に対する提案方式の $E(P)$ の比を $Power_Ratio$ として、評価指標に用いる。 $E(P)$ は、シミュレーション実験におけるポトルネックリンクの合計消費電力量(単位は[J])をシミュレーション時間(単位は[sec])で除して算出する。 $E(P)$ は小さい方がよい。

さらにシミュレーション実験における各コンテンツの送受信ごとに発生した controlled delay R [sec] の平均値として $E(R)$ [sec] を評価指標とする。 $E(R)$ は小さい方がよい。

また、各コンテンツのコンテンツサイズ M [packets] の平均値として平均コンテンツサイズ $E(M)$ [packets] を算出する。

3.3 シミュレーション実験の結果

実験結果について、図4に平均コンテンツサイズ $E(M)$ の変化、図5に従来方式に対する提案方式の平均消費電力 $E(P)$ の比 $Power_Ratio$ 、図6に平均の controlled delay $E(R)$ の変化のグラフをそれぞれ示す。

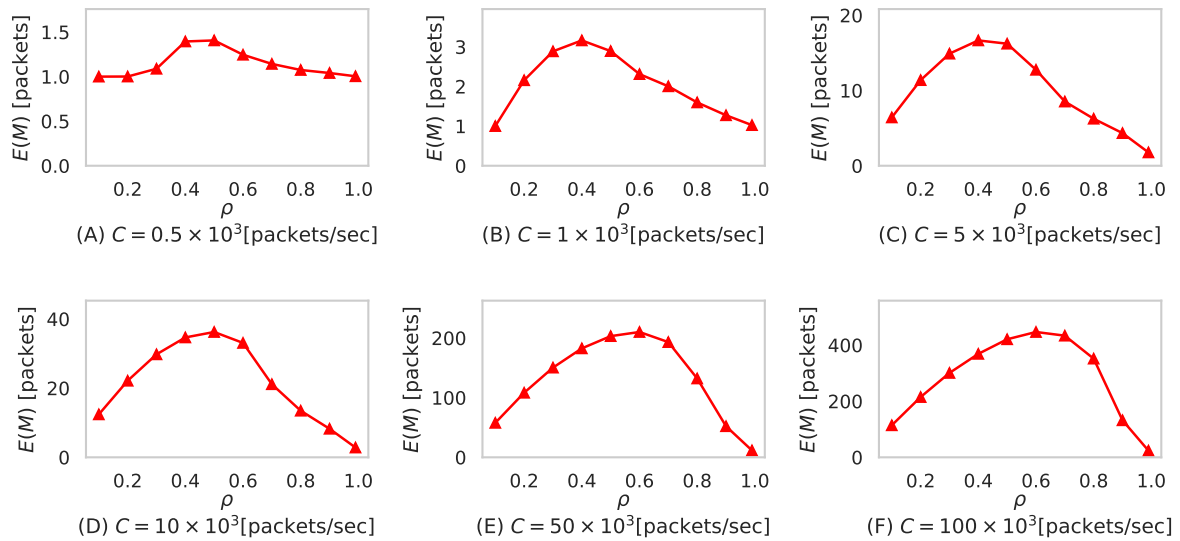
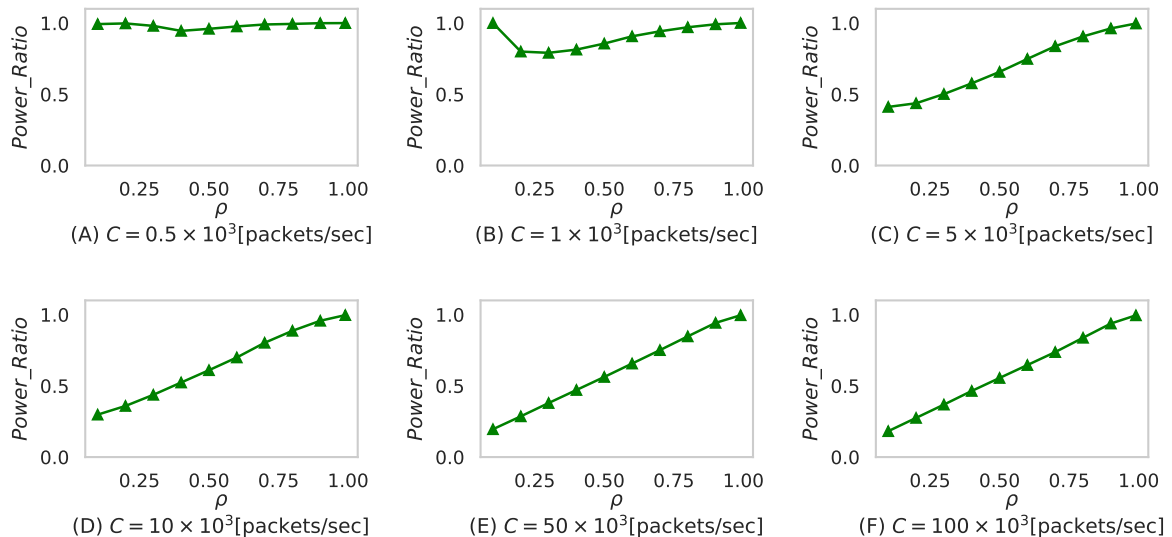
図4について、各リンク容量 C における、リンク利用率 ρ に対する平均コンテンツサイズ $E(M)$ のグラフを示している。 $E(M)$ は最大で 446.8 [packets] ($C=100 \times 10^3$ [packets/sec], $\rho=0.6$ の場合)、最小で 1.0 [packet] ($C=0.5 \times 10^3$ [packets/sec], $\rho=0.1$ の場合)であった。リンク利用率 ρ が小さいうちは平均コンテンツサイズ $E(M)$ を小さく抑えられている。これは、この状況で M をさらに大きくした場合、パケットの到着間隔が長くなるためパケットが M 個到着するまでの待ち時間 ω が延び、 $E(R)$ が大きくなってしまったためである。 ρ が $0.4 \leq \rho \leq 0.6$ 付近まで $E(M)$ は増加し続け、それを超えると $E(M)$ の減少が見られる。これは M が大きくなるとバッファリング遅延 δ が延びるので、それを抑制するためである。またリンク容量 C の増加による $E(M)$ の増加が、 ρ の増加による $E(M)$ の減少より大きくなる傾向が見られた。

図5について、各リンク容量 C における、リンク利用率 ρ に対する $Power_Ratio$ のグラフを示している。 $Power_Ratio$ は最大で1.00倍 ($C=1 \times 10^3$ [packets/sec], $\rho=0.99$ の場合)、最小で0.18倍 ($C=100 \times 10^3$ [packets/sec], $\rho=0.1$ の場合)であった。実験に用いた

(注1)提案方式のネットワークがマルチホップの場合、ネットワークのどこがポトルネックリンクになり、バッファリング遅延を発生させるかわからない。そのためバッファリング遅延の観測のため、確認応答によるバッファリング遅延の通知などの機能が必要である。しかし図3のシナリオでは、単一ホップであるため送信側エッジノードでバッファリング遅延が発生するので、その平均値 δ と平均パケット待ち時間 ω を送信側エッジノードで観測することで、確認応答によるバッファリング遅延の通知機能を実装せずに controlled delay を得られる。

表 1: シミュレーション実験に用いた各パラメタ [17]

記号	意味	数値
C	ボトルネックリンク容量	$0.5 \times 10^3 \sim 100 \times 10^3$ [packets/sec]
λ	平均パケット到着率	$0.05 \times 10^3 \sim 99 \times 10^3$ [packets/sec]
ρ	ボトルネックリンク利用率 ($\rho = \frac{\lambda}{C}$)	0.1~0.99
P_R	パケット送信に係る消費電力	80 [W]
P_T	スリープ状態からの立ち上がりに係る消費電力	80 [W]
δ_w	スリープ状態からパケット送信状態へ移行するまでの時間	0.001 [sec]
δ_p	ボトルネックリンクにおける片方向伝搬遅延	0.02 [sec]

図 4: シミュレーション実験における平均コンテンツサイズ $E(M)$ の変化図 5: シミュレーション実験における提案方式と従来方式の消費電力の比 $Power_Ratio$ の変化

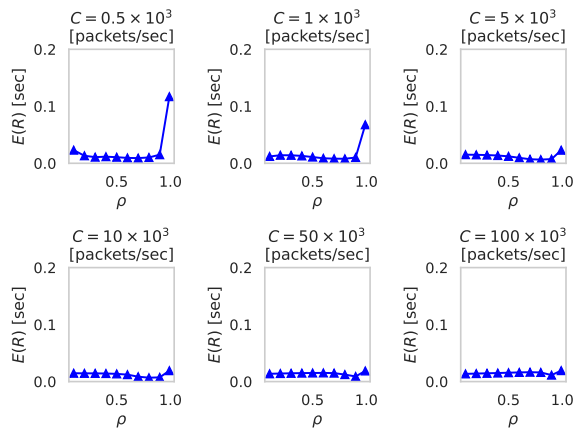


図 6: シミュレーション実験における提案方式の平均の controlled delay $E(R)$ の変化

条件のうち、ほとんどの場合において提案方式により消費電力が削減されていることがわかる。また、 C が増加するにつれて $Power_Ratio$ が小さくなるのがわかる。これはリンク容量の増加に伴い M を大きく取ることが可能になるため、それにより効果的なバースト送信とスリープが実現されるためであると考えられる。 $Power_Ratio$ は、 ρ が小さいほど小さくなり、 ρ が大きいほど大きくなる傾向が見られる。

図 6 について、各リンク容量 C における、リンク利用率 ρ に対する平均の controlled delay $E(R)$ のグラフを示している。 $E(R)$ は最大で 0.117 [sec] ($C=0.5 \times 10^3$ [packets/sec], $\rho=0.99$ の場合)、最小で 0.006 [sec] ($C=5 \times 10^3$ [packets/sec], $\rho=0.8$ の場合) であった。提案方式では、 $E(R)$ は target delay である 0.02 [sec] を基準として制御される。しかし、図 6 では、特にリンク容量 C が $0.5 \times 10^3 \leq C \leq 1 \times 10^3$ [packets/sec]、リンク利用率 ρ が $\rho=0.99$ の範囲で $E(R)$ が 0.02 [sec] を超えている。これは、 M を自然数とし、 $M=1$ をコンテンツサイズの下限としていることに起因する。

ここで、図 6 のうち、平均の controlled delay $E(R)$ が target delay である 0.02 [sec] を下回っている部分に着目する。図 7 に図 6 のグラフの縦軸の上限を 0.02 [sec] として拡大したグラフを示す。

図 7 より、特にリンク容量 C が $1 \times 10^3 \leq C \leq 100 \times 10^3$ [packets/sec]、リンク利用率 ρ が $0.7 \leq \rho \leq 0.9$ の範囲で $E(R)$ が減少する傾向を確認できる。この理由として、リンク容量 C 、リンク利用率 ρ がともに高い場合に、コンテンツサイズ M が収束していないことに起因する。図 8 に $C=100 \times 10^3$ [packets/sec], $\rho=0.1$ の場合のコンテンツサイズ M と controlled delay R の時間変化を、図 9 に $C=100 \times 10^3$ [packets/sec], $\rho=0.9$ の場合のコンテンツサイズ M と controlled delay R の時間変化をそれぞれ示す。

図 8 からは時間の経過に伴い、コンテンツサイズ M と controlled delay R が収束する様子がわかる。一方

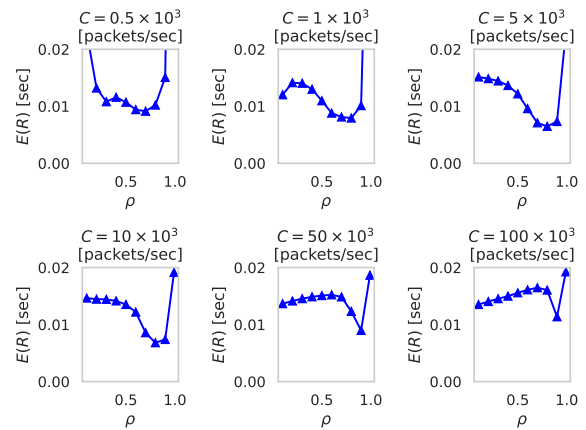


図 7: シミュレーション実験における提案方式の平均の controlled delay $E(R)$ の変化 (図 6 を拡大したグラフ)

で図 9 からは M と R が収束することなく、増減を繰り返し振動している様子が伺える。これはバッファリング遅延 δ の変動が大きいことに起因する。つまりコンテンツサイズ M は controlled delay を用いて計算されるが、controlled delay にはバッファリング遅延 δ が含まれる。リンク利用率 ρ が大きいとき、 δ は変動が大きくなる。その結果、コンテンツサイズ M の振動が現れたと考えられる。

4. まとめ

本稿では、バッファリング遅延を考慮した低消費電力ネットワーク制御を提案し、シミュレーション実験により従来方式のネットワークとの性能評価を行った。

本研究の提案方式は、パケットを集約したコンテンツをバースト送信することによって、スリープ状態からの立ち上がりに係る電力を抑制する。バースト送信による省電力化は遅延時間の増加とトレードオフの関係にあるため、遅延時間がしきい値の範囲で収まるようにコンテンツサイズを調整するアルゴリズムを提案した。

簡易的なシミュレーション実験により、提案方式は従来方式と比較し、リンク容量が大きいとき、消費電力を大きく削減できることを示した。

今後の課題は、コンテンツサイズの調整アルゴリズムの改良や、マルチホップのネットワークにおける性能評価を実施することである。

参考文献

- [1] International Programs Center for Demographic and Economic Studies, U.S. Census Bureau, "Total Midyear Population for the World: 1950-2050". Archived on 2017-04-17. Retrieved 20120-01-07. (<https://web.archive.org/web/20170417134744/https://www.census.gov/population/international/dataslashidb/worldpoptotal.php>)
- [2] Telecommunication Development Bureau, International Telecommunication Union (ITU), "ICT Facts

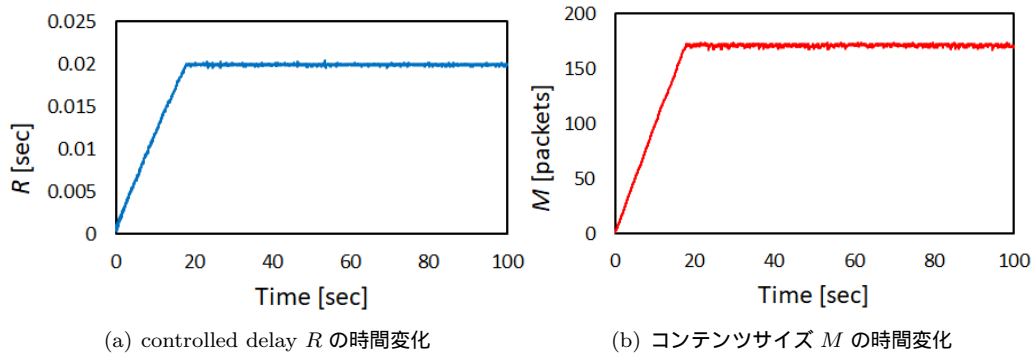


図 8: controlled delay R とコンテンツサイズ M の時間変化 ($C=100 \times 10^3$ [packets/sec], $\rho=0.1$)

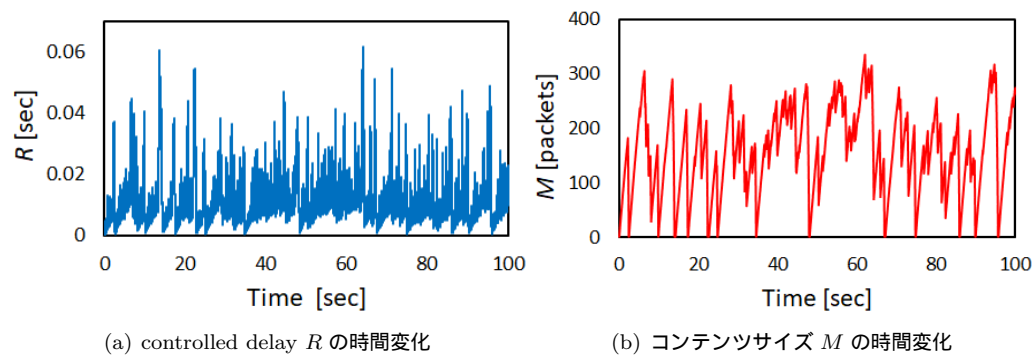


図 9: controlled delay R とコンテンツサイズ M の時間変化 ($C=100 \times 10^3$ [packets/sec], $\rho=0.9$)

and Figures 2017". Retrieved 2020-01-07. (<https://www.itu.int/en/ITU-D/Statistics/Documents/facts/ICTFactsFigures2017.pdf>).

- [3] Jonathan G. Koomey, "Estimating Total Power Consumption by Servers in the U.S. and the World", Feb. 2007.
- [4] Vodafone Corporate Social Responsibility Reports 2001/02 to 2005/06.
- [5] Gerhard Fettweis, Ernesto Zimmermann, "ICT ENERGY CONSUMPTION? TRENDS AND CHALLENGES", WPMC 2008.
- [6] 株式会社 NTT データ研究所, 我が国際情報経済社会における基盤整備 (IT 機器のエネルギー消費量に係る調査事業) 報告書, 2013 年.
- [7] コスト等検証委員会, "コスト等検証委員会報告書", 2011 年 12 月 19 日, 参照 2021 年 4 月 28 日 (<http://www.cas.go.jp/jp/seisaku/npu/policy09/pdf/20111221/hokoku.pdf>).
- [8] 日本原子力文化財団, "原子力・エネルギー図面集 2017", 2017 年 9 月 1 日, 参照 2021 年 4 月 28 日 (<https://www.ene100.jp/zumen/2-1-9>).
- [9] 環境省, "2017 年度 (平成 29 年度) の温室効果ガス排出量 (確報値) について", 2019 年 4 月 16 日, 参照 2021 年 4 月 28 日 (<http://www.env.go.jp/press/111337.pdf>).
- [10] 国立環境研究所, "日本国温室効果ガスインベントリ報告書", 2019 年 4 月 16 日.
- [11] 東京電力, "需給逼迫による計画停電の実施と一層の節電のお願いについて", 2011 年 3 月 13 日, 参照 2021 年 4 月 28 日 (<http://www.tepco.co.jp/cc/press/11031315-j.html>).
- [12] 経済産業省資源エネルギー庁, "日本初の "ブラックアウト"、その時一体何が起きたのか", 2018 年 11 月 2 日, 参照 2021 年 4 月 28 日 (<https://www.enecho.meti.go.jp/about/special/johoteikyoku/blackout.html>).
- [13] NTT コミュニケーションズ, "意外と知らない? IT トrend用語 IoT とは", 参照 2021 年 4 月 28 日 (<https://www.ntt.com/bizon/glossary/e-i/iot.html>).
- [14] 富岡理, "JASA 発 IoT 通信 (7) IoT の電力を担うエネルギーハーベスティングの可能性と課題", EE Times Japan, 2018 年 1 月 25 日, 参照 2021 年 4 月 28 日 (<https://eetimes.jp/ee/articles/1801/25/news016.html>).
- [15] Mehrgan Mostowfi, Ken Christensen, "An Energy-Delay Model for a Packet Coalescer", 2012 Proceedings of IEEE Southeastcon, 2012.
- [16] Jim Gettys, "Bufferbloat: Dark buffers in the internet", IEEE Internet Computing, 15(3):96-96, 2011.
- [17] Hiroyuki Okamura, Tadashi Dohi, Shunji Ozaki, "On the Effect of Power Saving by Auto-sleep Function of a Computer System II- Queueing Model", Transactions of Information Processing Society of Japan Vol.40, No.3, pp1027-1040, 1999 in Japanese.