

マルチホップネットワークにおけるリンクごとの送受信の同期によるIoTデバイスの省電力化

河竹 純一¹ 串田 高幸¹

概要: スマートパーキングにおいて車両を検出するために、バッテリーで動作するIoTデバイスがToFセンサの距離データをサーバに送信する。サーバに届かない距離データは近傍のIoTデバイスの中継して送信する。課題は、中継するIoTデバイスは距離データの受信タイミングを予測できないため、ネットワーク機能を動作し続ける必要があり電力を浪費することである。提案方式はマルチホップネットワークのリンクごとに送受信のタイミングを同期し、中継するIoTデバイスの限られた受信時間の中で距離データを送信するスケジューリング手法である。実験として4台のESP32を使用してマルチホップネットワークを構築し、10秒に1つの頻度で16bytesのデータをサーバに送信した。実験は次の2種類の環境で行った。実験1は周囲に遮蔽物が少なく、15m × 16.4mの長方形の四隅に配置する理想的な環境であり、実験2は22.6m × 54.5mの駐車場の四隅に配置しユースケースシナリオに近い環境である。評価としてネットワーク機能の動作時間から消費電力を換算し、常にネットワーク機能を動作させる場合と比較した結果、実験1において約86%削減され、実験2において約81%削減されることがわかった。

1. はじめに

背景

屋外の駐車場で車両の入出庫を検出し、ユーザが駐車場の空き状況を確認したり駐車スペースを予約するシステムを一般的にスマートパーキングと呼ぶ。スマートパーキングにおいてIoTデバイスがToFセンサから取得した距離データをサーバに送信する際の方法の一つとしてマルチホップネットワークがある[1]。マルチホップネットワークとは、IoTデバイスがサーバに距離データを直接送信できないときに近傍のIoTデバイスの中継してバケツリレー方式で距離データをサーバまで届ける手法である。

スマートパーキングにおけるマルチホップネットワークでは、IoTデバイスは二つの主要な機能を担う。一つは距離データを取得してサーバへ送信する機能であり、もう一つは他のIoTデバイスからのデータを中継する機能である。距離データを取得し、他のデバイスのデータも中継する場合、IoTデバイスはこれら二つの機能を担う。例えば、全てのIoTデバイスから距離データを収集する場合、末端のIoTデバイスを除く全てのIoTデバイスは両方の機能を担う。この場合、中継する距離データと自身の距離データを合わせて送信する。

屋外の駐車場に設置されるスマートパーキングのIoTデバイスはバッテリーで動作する[2-4]。バッテリーの電力は限られているため、IoTデバイスの消費電力を削減する研究がある[5]。消費電力の削減は、IoTデバイスの稼働時間の延長やバッテリー交換の頻度を低減することに寄与する[6,7]。IoTデバイスはネットワーク機能の動作時に約2.5倍の電力を消費するため、ネットワーク機能の動作時間を短縮することで消費電力を削減できる。

課題

マルチホップネットワークにおける課題は中継するIoTデバイスがネットワーク機能を動作し続ける必要があり電力を浪費することである。動作し続ける理由は、IoTデバイスの設置位置によってサーバまでのホップ数や子ノードの数が異なり、中継するIoTデバイスにおける距離データの受信タイミングが予測できないためである。課題の図を図1に示す。図1はIoTデバイスA, B, CがIoTデバイスDを経由してサーバに距離データを送信する状況を表す。 t_s と t_c はそれぞれ距離データの取得から送信までの処理時間と通信プロセスにかかる時間を表す。IoTデバイスA, B, CのIoTデバイスDまでの中継するIoTデバイスの数はそれぞれ2, 1, 0であり、ホップ数が異なる。したがってIoTデバイスA, B, Cが同時刻に距離データを取得し、直後に距離データを送信する場合に、IoTデバイスDにお

¹ 東京工科大学大学院バイオ・情報メディア研究科コンピュータサイエンス専攻
〒192-0982 東京都八王子市片倉町1404-1

る受信タイミングが一致しない。

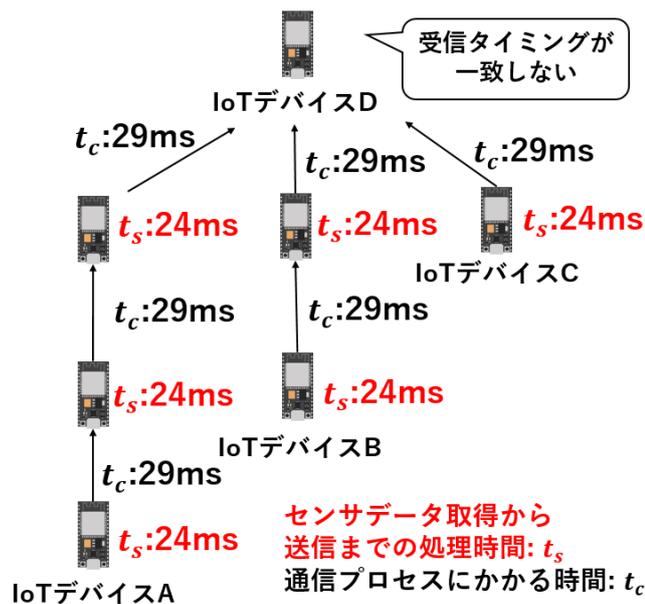


図 1 課題の図

各章の概要

第 2 章ではマルチホップネットワークにおける省電力化の関連研究を説明する。第 3 章では提案方式とユースケース・シナリオについて述べる。第 4 章ではソフトウェアの詳細と実装方法について述べる。第 5 章では提案方式を評価する実験方法とその結果について述べる。第 6 章では提案方式の問題点と解決方法について議論する。第 7 章では本稿のまとめを述べる。

2. 関連研究

マルチホップネットワークのデータ集約手法について、QADA というスキームが提唱されている [8]。この手法は、特定のノードへのデータ集約により、エネルギー消費とネットワークトラフィックを効果的に削減することを目指している。しかし、IoT デバイスにおける送受信の同期問題には触れておらず、今後の課題といえる。

BLE (Bluetooth Low Energy) を基盤とした革新的なハイブリッド型マルチホップネットワークが、スター型とメッシュ型の組み合わせによって構築されている [9]。この研究は、従来の Zigbee ベースのネットワークと比較して、消費電力を大幅に削減することに成功している。ただし、Bluetooth の利用に伴い、定期的なアドバタイズやスキャンが必要となるため、通信回数の増加が問題となる。

省電力化を目指すマルチホップネットワークのための MAC プロトコルとして、WiseMAC が開発されている [10]。このプロトコルは、送受信の効率的な同期により、Zigbee ベースのネットワークと比較して 85% の電力削減を達成

しているが、研究の焦点はダウンリンク通信に限定されている。

無線センサネットワークのアルゴリズム改善に関して、一般的な LEACH の拡張版である E-LEACH が開発されている [11]。E-LEACH は、センサノードのバッテリー残量を考慮し、ネットワーク負荷を均等に分散することに着目しており、クラスタヘッダの役割をクラスタの大きさに応じて動的に変更することで効率的なネットワーク運用を実現している。しかし、このアプローチはシミュレーションベースの研究に限定されており、実際の使用環境での電力消費に関する詳細な分析はまだ行われていない。

3. 提案

提案方式

提案の目的は、マルチホップネットワークの中継する IoT デバイスにおけるネットワーク機能の動作時間を削減し、その消費電力を削減することである。提案方式はマルチホップネットワークのリンクごとに送受信のタイミングを同期し、中継する IoT デバイスの受信時間を集約するスケジューリング手法である。末端の IoT デバイスから送信し、送信タイミングが重ならないように順に送信するようにサーバでスケジュールする。固定されたスケジュールで距離データを送信するために、提案方式は以下の前提条件に従うものとする。

- IoT デバイスは固定された位置に設置され、追加や撤去がされないものとする。
- 遮蔽や干渉によるパケットロスが発生しない理想的なリンクとする。
- ネットワークゲートウェイは一つである。

図 2 に提案の概要図を示す。図 2 は IoT デバイスがサーバに距離データを送信するときの状況を表す。IoT デバイスは A, B, C, D, E, F の 6 台であり、中継するデバイス

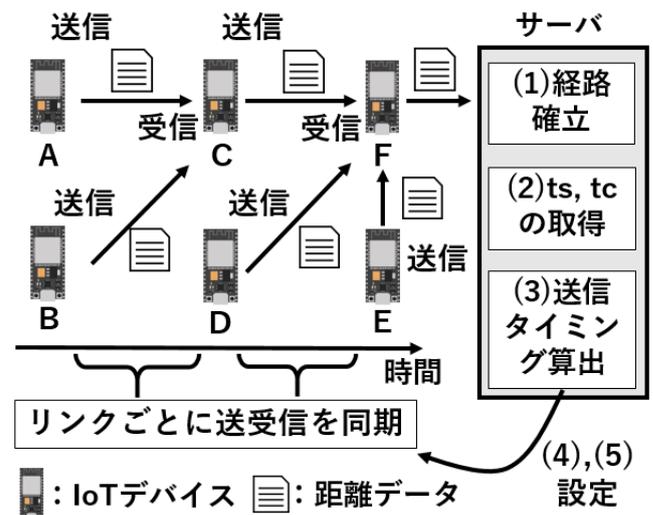


図 2 提案の概要図

はIoTデバイスCとIoTデバイスFである。IoTデバイスAとIoTデバイスBがIoTデバイスCを経由して距離データを送信しており、IoTデバイスCとIoTデバイスDとIoTデバイスEがIoTデバイスFを経由して距離データを送信している。提案方式の詳しい流れについて以下の(1)-(5)で説明する。(1)と(2)は前段処理として運用前に行うものとする。

- (1) 経路の確立 (IoTデバイスからサーバまでの最短経路を事前に決定する。)
- (2) t_s , t_c の取得 (距離データを送信してから中継するIoTデバイスが受信するまでの時間を概算し、サーバに送信する。)
- (3) 送信タイミングのスケジュール (距離データを送信するタイミングをサーバで算出する。)
- (4) 送信タイミングの取得 (サーバで算出した送信タイミングをIoTデバイスが問い合わせる。)
- (5) ネットワーク機能を起動する時間の設定 (送信タイミングと時刻の誤差をもとに、IoTデバイスでネットワーク機能を起動する時間を設定する。)

(1) 経路の確立

IoTデバイスがサーバに距離データを送信するために通信経路を確立する。まず、IoTデバイスがブロードキャスト通信によって経路制御パケットを送出する。経路制御パケットは、距離データの取得元のIoTデバイスのMACアドレス、経由したIoTデバイスのMACアドレスのリスト、サーバに到達するまでに経由したIoTデバイスの数(以下ではホップカウントとする。)をペイロードに含むパケットである。その後サーバに到達した経路制御パケットのホップカウントを参照し、最小のホップカウントとなる経路をそのIoTデバイスの経路として登録する。登録した経路を経路表にしてIoTデバイスにレスポンスとして返す。

(2) t_s , t_c の取得

送信タイミングをサーバで算出するためにIoTデバイスの t_s と t_c を測定してサーバに送信する。測定方法は、32[bytes] のデータをサーバに1000回送信し、1回経由するごとにIoTデバイスがデータを送信した時刻と受信した時刻を記録する。受信してから送信するまでの時間を t_s 、送信してから受信するまでの時間を t_c として、サーバに送信する。(1)経路の確立と(2) t_s , t_c の取得はスマートパーキングを運用する前に事前に行う前段処理とする。

(3) 送信タイミングのスケジュール

提案方式の送信タイミングのスケジューリングアルゴリズムをアルゴリズム1に示す。アルゴリズム1は前段処理の後にIoTデバイスが送信タイミングを取得するためにサーバにリクエストを送信したときに実行される。アルゴリズムの入力は、リクエスト元のIoTデバイスのMACアドレス (iot_mac)、リクエスト元のIoTデバイスからサーバまでのホップ数 (N)、各階層に存在するIoTデバイス

アルゴリズム 1 スケジューリングアルゴリズム

Input: iot_mac : IoTデバイスのMACアドレス, N : サーバまでのホップ数, N_max : ネットワークの最大ホップ数, num_in_level : 各階層のIoTデバイスの数, iot_queue : 階層における送信の順番, ts_list : プログラム実行時間のリスト, tc_list : 通信時間のリスト, σ : 時刻の誤差の標準偏差

Output: $send_time$: 送信タイミング

- 1: **function** SEND_SCHEDULE(iot_name , N , N_max , num_in_level , iot_queue , ts_list , tc_list , σ)
- 2: $num_in_nw \leftarrow \text{sum of } num_in_level$
- 3: Initialize sum_ts_tc as an empty list
- 4: **for** $i \leftarrow 0$ **to** num_in_nw-1 **do**
- 5: Append $ts_list[i] + tc_list[i]$ to sum_ts_tc
- 6: **end for**
- 7: $timeslot \leftarrow \max(sum_ts_tc)$
- 8: $turn \leftarrow 0$
- 9: **for** $i \leftarrow 0$ **to** $N_max - N$ **do**
- 10: $turn \leftarrow turn + num_in_level[i]$
- 11: **end for**
- 12: $turn \leftarrow turn + \text{position of } iot_name \text{ in } iot_queue[N_max - N]$
- 13: $send_time \leftarrow turn \times (timeslot + 3 \times \sigma)$
- 14: **return** $send_time$
- 15: **end function**

の数 (N_max)、リクエスト元のIoTデバイスが存在する階層における送信の順番 (iot_queue)、プログラム実行時間のリスト (ts_list)、通信時間のリスト (tc_list)、時刻の誤差の標準偏差 (σ) とする。アルゴリズムの出力は送信タイミング ($send_time$) である。

最初に(2) t_s , t_c の取得によって作成される t_s と t_c のリストから、1ホップにかかる時間 (t_s と t_c を合計した数値)の最大値を算出し、それをタイムスロットの時間とする。タイムスロットはIoTデバイスが距離データを送信するときに個別に専有する時間であり、その時間内に距離データを送信することで他のIoTデバイスの送信タイミングと衝突しないようにする。次に、リクエスト元のIoTデバイスが距離データを送信するときの順番 ($turn$) を決める。順番はリクエスト元のIoTデバイスが存在する階層よりも低い階層におけるIoTデバイスの数と iot_queue における順番を足し合わせた数とする。最後に、タイムスロットと時刻の標準偏差を3倍した数値 (3σ) を足して、リクエスト元のIoTデバイスの送信する順番の数を乗算した値を送信タイミングとして返す。 3σ とする理由は、IoTデバイスにおいて99%の時刻の誤差が平均値- 3σ ~ 平均値+ 3σ の区域にあるためである [12] 送信タイミングは、同時刻にToFセンサの距離データを取得した時刻から何秒後に距離データの送信をするかを示す時間である。

(4) 送信タイミングの取得

- (2) t_s , t_c の取得後に、IoTデバイスがサーバに対して送

信タイミングを要求する。IoT デバイスからサーバまでの経路は (1) 経路の確立に準拠する。送信タイミングを取得したのち、センサからデータを取得する時刻からタイマーを開始し、送信タイミングの時間が経過した直後に距離データを送信する。

(5) ネットワーク機能を起動する時間の設定

中継する IoT デバイスは送信タイミングを取得した後にネットワーク機能を起動する時間を設定する。図 3 に距離データの送信フローを示す。まず、IoT デバイス A-F において同時刻にセンサから距離データを取得する。その後、IoT デバイス A の送信タイミングよりも 3σ だけ早く IoT デバイス C がネットワーク機能を起動する。中継する IoT デバイスが送信する IoT デバイスよりも早くネットワーク機能を起動することによって、送信される距離データが受け取れない事態を防いでいる。その後 IoT デバイス A が距離データを送信したのちに IoT デバイス B が距離データを送信する。IoT デバイス C が IoT デバイス A と IoT デバイス B の距離データを受け取った場合に IoT デバイス C においてネットワーク機能を停止する。IoT デバイス C-E が IoT デバイス F に対して距離データを送信するときも同様の方法で送信される。IoT デバイス F において IoT デバイス C の送信タイミングよりも 3σ だけ早くネットワーク機能を起動し、IoT デバイス C-E が距離データを送信したのちに IoT デバイス F のネットワーク機能を停止する流れである。

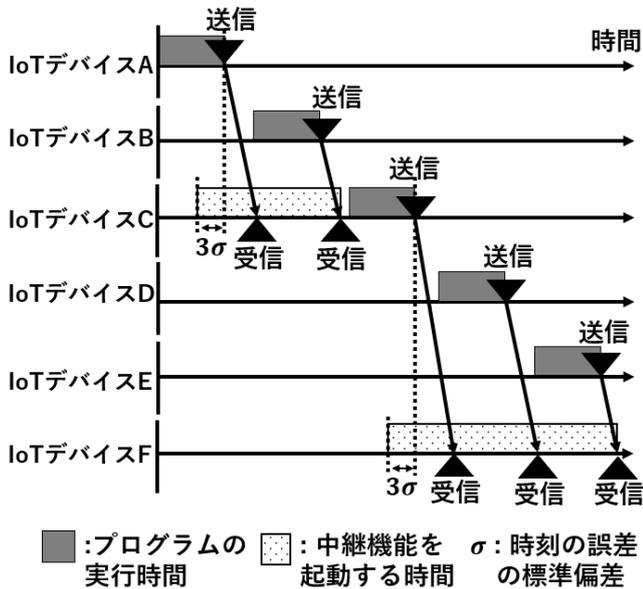


図 3 距離データの送信フロー

ユースケース・シナリオ

ユースケースはスマートパーキングである [1]。ユースケース図を図 4 に示す。スマートパーキングでは、駐車場における車両の入出庫を IoT デバイスが ToF センサを用

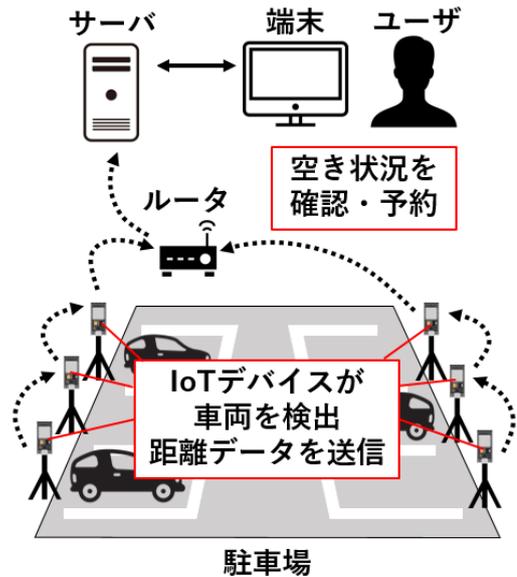


図 4 ユースケース図

いて検出する。それらの距離データをサーバに集約し、管理することによってユーザが端末から空き状況の確認や駐車予約をすることが可能になる。屋外駐車場において IoT デバイスがバッテリーで動作する状況を想定している。バッテリーで動作するスマートパーキングは電気工事が不要で設置が簡易であることがメリットである。一方で、駐車場の管理者がバッテリーの交換をする手間がかかるというデメリットが存在する。そこで提案手法を適用して省電力化を図ることによって、バッテリーの交換頻度が低減する。

4. 実装

IoT デバイスとサーバのソフトウェアを実装する。IoT デバイスの実装を MicroPython で行う。サーバの実装を Python で行う。ソフトウェアの概要図を図 5 に示す。IoT デバイスのソフトウェアは (1) 初期設定プログラムと (4) 中継プログラムの 2 つである。サーバのソフトウェアは (2) 経路算出プログラムと (3) 送信タイミング算出プログラムの 2 つである。

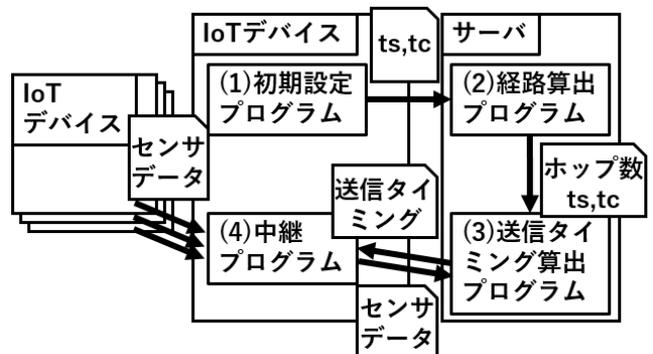


図 5 ソフトウェアの概要図

5. 評価実験

実験環境

東京工科大学八王子キャンパス構内に IoT デバイスを 4 台設置して以下の 2 つの実験（実験 1 と実験 2）を行う。

実験 1

実験 1 は周囲に遮蔽物が存在しない理想的な環境に近い状態で距離データを送信する実験である。図 6 に実験 1 の構成図を示す。IoT デバイスは A-D の 4 台であり、横幅 16.4m、縦幅 15m の長方形の四隅に設置した。実験した場所はキャンパス内の厚生棟と図書館棟の間の広場である。IoT デバイスを街灯の下に設置したため、横幅 16.4m と縦幅 15m はその街灯の間隔となっている。通信経路としては IoT デバイス A-C が IoT デバイス D に距離データを送信する。IoT デバイス D に送信するデータはタイムスタンプ、送信元の MAC アドレス、距離データに見立てた 32bit の浮動小数点であり、データ量は 1 個あたり 16bytes である。実験の流れとしては、IoT デバイス A-D が 10 秒に一回の頻度でサーバに距離データを 1 個送信し、実験を 1 時間行う。このとき、距離データを中継する IoT デバイス D においてネットワーク機能の動作時間を計測する。

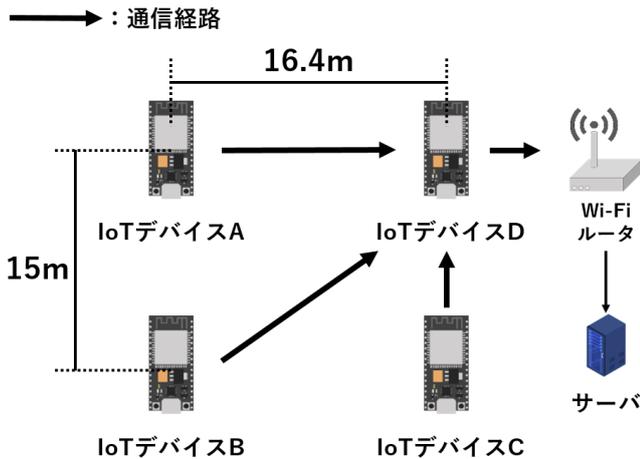


図 6 実験 1 の構成図

実験 2

実験 2 は横幅 54.5m、縦幅 22.6m のキャンパス内の屋外駐車場の四隅に IoT デバイスを設置し、ユースケースシナリオに近い環境で距離データを送信する実験である。図 7 に実験 2 の構成図を示す。IoT デバイスは実験 1 と同じく 4 台であるが、IoT デバイス B が IoT デバイス D に対して直接的に距離データを送信することができないため、IoT デバイス B が IoT デバイス C を経由して距離データを送信しており、IoT デバイス A と IoT デバイス C が IoT デバイス D に距離データを送信している。送信するデータは実験 1 と同様に、タイムスタンプ、送信元の mac アドレス、距離データに見立てた 32bit の浮動小数点を含む 16bytes

のデータである。実験の流れも実験 1 と同様に、IoT デバイス A-D が 10 秒に一回の頻度でサーバに距離データを 1 個送信する実験を 1 時間行い、距離データを中継する IoT デバイス D においてネットワーク機能の動作時間を計測する。

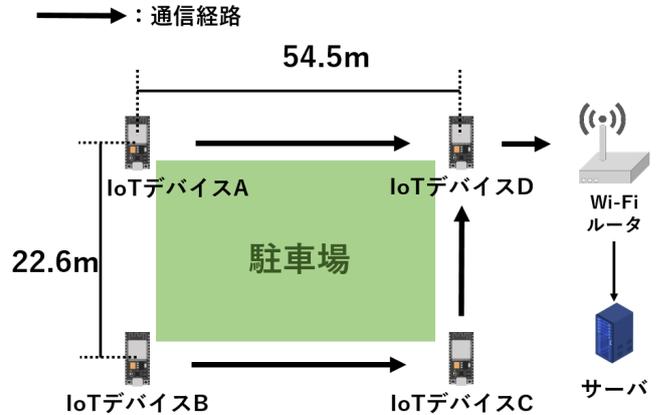


図 7 実験 2 の構成図

実験結果と分析

図 8 に実験 1 と実験 2 における中継する IoT デバイスのネットワーク機能の動作時間のグラフを示す。ネットワーク機能の動作時間は、距離データの送信周期である 10 秒の中でどれだけネットワーク機能が動作していたかを示す時間である。グラフは横軸が実験の経過時間、縦軸がネットワーク機能の動作時間を表している。実験 1 では平均で約 192ms まで削減し、実験 2 では平均で約 818ms まで削減した。一方で実験 2 における結果が実験 1 における結果の約 4.26 倍 ($818ms/192ms \approx 4.26$) の差がある。これは提案方式の t_s と t_c を測定する前段処理において、1 ホップあたりにかかる時間の最大値を使用しているために、理想的な環境の実験 1 とユースケースシナリオに近い実験 2 における t_c の最大値に差があることに加えて、実験 2 のネットワークにおける最大ホップ数が実験 1 に比べて 1 ホップ

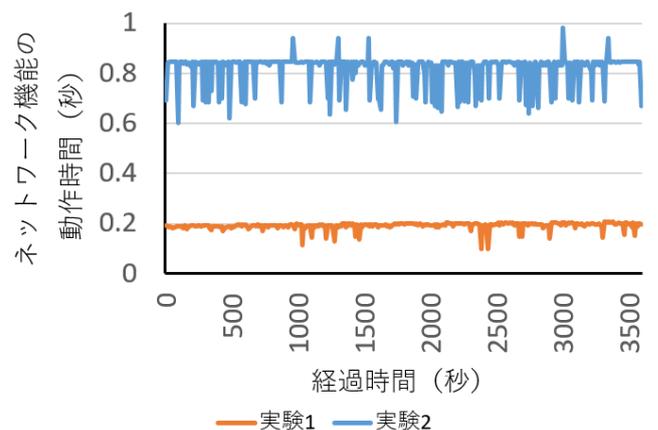


図 8 実験の経過時間とネットワーク機能の動作時間

多いことに起因する。

次に、ネットワーク機能の動作時間を消費電力に換算する。ネットワーク動作中の電力を 400mW ($120\text{mA} \times 3.3\text{V} \approx 400\text{mW}$)、それ以外の時間で Deep Sleep モードにすることで 50mW ($15\text{mA} \times 3.3\text{V} \approx 50\text{mW}$) とすると、積算電力量は図 9 のように表せる。図 9 より、1 時間の積算電力量は実験 1 で約 57mWh であり、実験 2 で約 79mWh であることがわかる。したがって常にネットワーク機能を動作させる場合の 400mWh ($400\text{mW} \times 1\text{h} = 400\text{mWh}$) と比較すると、実験 1 の消費電力の削減量は約 86% ($100\% - ((57/400) \times 100)\% \approx 86\%$)、実験 2 の消費電力の削減量は約 81% ($100\% - ((79/400) \times 100)\% \approx 81\%$) である。

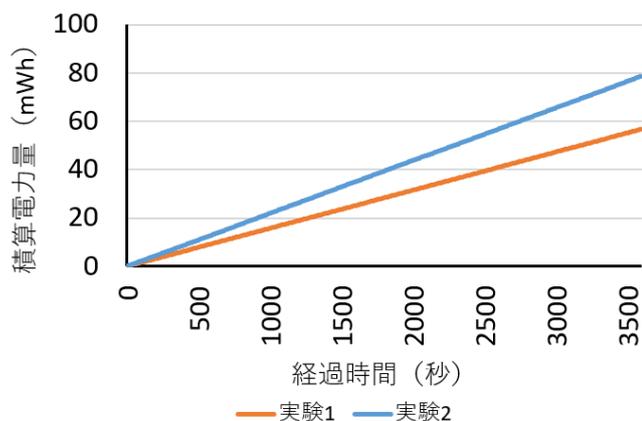


図 9 ネットワーク機能の起動時間から換算した積算電力量

6. 議論

提案方式のアルゴリズムではネットワーク機能の動作時間が 1 ホップあたりにかかる時間の最大値に依存しているために、リンクごとの通信時間にばらつきがある環境においては必要以上に動作時間を要するという問題がある。この問題を解決する方法として、リンクごとに異なる長さのタイムスロットを割り当て、スループットが良好であるリンクにおけるネットワーク機能の動作時間を減らす方法がある。この方法は中継する IoT デバイスにおける消費電力を削減するだけでなく、通信待ち受け時間を短縮させるためネットワーク全体のスループットを向上することにもつながる。

7. おわりに

マルチホップネットワークにおける課題は、中継する IoT デバイスがネットワーク機能を動作し続ける必要があることである。提案はリンクごとに送受信のタイミングをサーバで決定するスケジューリング手法である。実験として、理想的な環境での実験 1 とスマートパーキングの状況に近い環境の実験 2 を行った。常にネットワーク機能を起

動する場合と消費電力を比較した結果、実験 1 では約 86% 削減され、実験 2 では約 81% 削減されることがわかった。本稿は省電力なスマートパーキングシステムの実現に貢献している。

参考文献

- [1] Chinrungrueng, J., Sunantachaikul, U. and Triamlumlerd, S.: Smart Parking: An Application of Optical Wireless Sensor Network, *2007 International Symposium on Applications and the Internet Workshops*, pp. 66–66 (online), DOI: 10.1109/SAINT-W.2007.98 (2007).
- [2] Radunovic, B. and Le Boudec, J.-Y.: Rate performance objectives of multihop wireless networks, *IEEE transactions on Mobile computing*, Vol. 3, No. 4, pp. 334–349 (2004).
- [3] Cai, Z. and Chen, Q.: Latency-and-coverage aware data aggregation scheduling for multihop battery-free wireless networks, *IEEE Transactions on Wireless Communications*, Vol. 20, No. 3, pp. 1770–1784 (2020).
- [4] Iqbal, S., Qureshi, K. N., Kanwal, N. and Jeon, G.: Collaborative energy efficient zone-based routing protocol for multihop Internet of Things, *Transactions on Emerging Telecommunications Technologies*, Vol. 33, No. 2, p. e3885 (online), DOI: <https://doi.org/10.1002/ett.3885> (2022).
- [5] Fourati, L., El-Kaffel, S., Mnaouer, A. B. and Touati, F.: Study of Nature Inspired Power-aware Wake-Up Scheduling Mechanisms in WSN, *2020 International Wireless Communications and Mobile Computing (IWCMC)*, pp. 2154–2159 (online), DOI: 10.1109/IWCMC48107.2020.9148433 (2020).
- [6] Jayakumar, H., Lee, K., Lee, W. S., Raha, A., Kim, Y. and Raghunathan, V.: Powering the internet of things, *Proceedings of the 2014 international symposium on Low power electronics and design*, pp. 375–380 (2014).
- [7] Iannacci, J.: Microsystem based Energy Harvesting (EH-MEMS): Powering pervasivity of the Internet of Things (IoT)—A review with focus on mechanical vibrations, *Journal of King Saud University-Science*, Vol. 31, No. 1, pp. 66–74 (2019).
- [8] Rahman, H., Ahmed, N. and Hussain, M. I.: A hybrid data aggregation scheme for provisioning Quality of Service (QoS) in Internet of Things (IoT), *2016 Cloudification of the Internet of Things (CIoT)*, pp. 1–5 (online), DOI: 10.1109/CIOT.2016.7872917 (2016).
- [9] Nair, K., Kulkarni, J., Warde, M., Dave, Z., Rawalgaonkar, V., Gore, G. and Joshi, J.: Optimizing power consumption in iot based wireless sensor networks using Bluetooth Low Energy, *2015 International Conference on Green Computing and Internet of Things (ICGCIoT)*, pp. 589–593 (online), DOI: 10.1109/ICGCIoT.2015.7380533 (2015).
- [10] El-Hoiydi, A. and Decotignie, J.-D.: WiseMAC: an ultra low power MAC protocol for the down-link of infrastructure wireless sensor networks, *Proceedings. ISCC 2004. Ninth International Symposium on Computers And Communications (IEEE Cat. No.04TH8769)*, Vol. 1, pp. 244–251 Vol.1 (online), DOI: 10.1109/ISCC.2004.1358412 (2004).
- [11] Xu, J., Jin, N., Lou, X., Peng, T., Zhou, Q. and Chen, Y.: Improvement of LEACH protocol for WSN, *2012 9th International Conference on Fuzzy Systems and Knowledge Discovery*, pp. 2174–2177 (online), DOI:

10.1109/FSKD.2012.6233907 (2012).

- [12] 河竹純一, 杉本一彦, 串田高幸: ”無線 LAN のマルチホップネットワークにおける送受信のタイミングの一致による IoT デバイスの省電力化”, クラウド・分散システム研究室, Vol. CDSL-TR-109 (2022).