

Wireless Powered-MEC における 自律分散的なオフロード確率決定法の検討

小林 拓弥[†] 安達 宏一[†]

[†] 電気通信大学 先端ワイヤレス・コミュニケーション研究センター
〒182-8585 東京都調布市調布ヶ丘 1-5-1
E-mail: †{t.kobayashi,adachi}@awcc.uec.ac.jp

あらまし モノのインターネット (IoT: Internet-of-Things) 端末の小型化に伴う計算能力の制約ならびに充電管理が重要な検討事項となっている。これらの2つの問題を解決するために、無線端末 (WD: Wireless Device) のタスクを代替計算処理するモバイルエッジコンピューティング (MEC: Mobile Edge Computing) と、アクセスポイント (AP: Access Point) に具備されたパワービーコンから WD へ無線高周波 (RF: Radio Frequency) 信号を送り、エナジーハーベスティング (EH: Energy Harvesting) を行う無線電力伝送 (WPT: Wireless Power Transfer) を組み合わせた Wireless Powered-MEC (WP-MEC) システムが期待されている。筆者らは、自律分散型確率的制御によってオフロードとローカル計算の2つのモードを選択する確率的バイナリーオフロード (PBO: Probabilistic Binary Offloading) 方式を提案した。本稿では、バッテリー蓄積量やパケット配信率 (PDR: Packet Delivery Rate) に依存する新しいオフロード確率式を提案する。計算機シミュレーションにより提案手法が AP での集中制御を用いた手法と比較して PDR, 処理時間, エネルギー消費量の間に存在するトレードオフのバランスを保てることを示す。

キーワード センサネットワーク, IoT, MEC, WPT, 確率的制御

Autonomous Distributed Offloading Probability Decision for Wireless Powered-MEC

Takuya KOBAYASHI[†] and Koichi ADACHI[†]

[†] Advanced Wireless & Communication Research Center, The University of Electro-Communications,
1-5-1 Chofugaoka, Chofu, Tokyo, 182-8585 Japan
E-mail: †{t.kobayashi,adachi}@awcc.uec.ac.jp

Abstract The Internet-of-Things (IoT) devices face critical issues of limited computing power and charge management due to miniaturization and increasing device density. Mobile edge computing (MEC) that computes heavy tasks of wireless devices (WDs) alternatively has been expected as a solution for the computing power problem. Wireless power transfer (WPT) that charges batteries of WDs from a power beacon at an access point (AP) is a solution for battery management. Recently, wireless powered-mobile edge computing (WP-MEC) system combining WPT and MEC has been expected to solve those problems jointly. We proposed the probabilistic binary offloading (PBO) strategy that selects proceeding methods based on the offloading probability. In this paper, we propose new offloading probability that depends on battery storage and packet delivery rate (PDR) in the environment charged by WPT. Comparing the proposed strategy and the centralized control method, we evaluate that the trade-off between PDR, processing time, and energy consumption can be balanced by computer simulation.

Key words Sensor Network, IoT, MEC, WPT, Probabilistic Control

1. はじめに

近年、センサとネットワークを接続する通信規格の標準化に伴

い、モノのインターネット (IoT: Internet-of-Things) が発展している [1]。IoT ネットワークでは小型のセンサを多数配置するような環境が想定されている。低コストかつ小型の無線センサ

等の IoT 機器は携帯端末等に比べ計算能力が低いため、端末内での計算処理はエネルギー消費量や処理遅延が増大してしまう問題がある。この問題を解決する技術として、モバイルエッジコンピューティング (MEC: Mobile Edge Computing) が挙げられる [2]。MEC は無線端末 (WD: Wireless Device) で生じた計算タスクをアクセスポイント (AP: Access Point) にオフロードし、AP に具備された高性能なサーバ上で代替処理を行う技術である。従来のクラウドコンピューティング [3] では、WD とクラウドサーバ間の物理的距離が長いために伝搬遅延が増大することが問題であったのに対し、MEC は WD とサーバ間の物理的距離を近づけることで、低遅延の対話型応答を可能にしている。IoT ネットワークでは多数の IoT 機器やセンサに対してどのように充電管理を行うかが問題となっている。この問題を解決する技術として、無線電力伝送 (WPT: Wireless Power Transfer) が挙げられる [4]。WPT は AP に具備されたパワービーコンから WD へ無線高周波 (RF: Radio Frequency) 信号を送り、WD が受信した RF 信号を直流電流に変換するエナジーハーベスティング (EH: Energy Harvesting) を行うことでバッテリーを充電する技術である。得られたエネルギーを WD の通信回路や計算回路に充てることで、有線充電無しで IoT 機器を稼働することが可能になる。これら 2 つの技術を組み合わせた Wireless Powered-MEC (WP-MEC) システムによって、以上の問題を同時に解決することが期待されている [5]。

また、MEC のオフロード方式には 2 つのオフロード方式が存在する [2]。パーシャルオフロードは WD で生じた 1 つのタスクをオフロード部分とローカル計算部分に分割する方式である。時間効率の良い処理が可能であるが、センサが生起するような単純なタスクには適用が難しく、分割には複雑な処理が必要になる。バイナリーオフロードは WD で生じたタスクを分割せず、オフロードまたはローカル計算のバイナリーモードのどちらかで処理をする方式である。時間効率は低いものの、分割の処理を省略しシステムを単純化することができる。

多くの既存研究では、AP における集中制御型のリソース割当手法が検討されている [6] [7] [8]。これらの集中制御型システムでは制御情報のやり取りによって処理遅延やエネルギー消費に関するオーバーヘッドが発生してしまう問題がある。文献 [8] では、処理時間とエネルギー消費量の負荷からオフロードの有効性を得るユーザ効用関数を用いてバイナリーモードを決定する手法を提案している。しかし、当手法では全 WD が同時にタスクを生起するモデルを使用しており、各 WD がランダムにタスクをオフロードするモデルには適用することができない。それに加えて、処理優先度を固定にしているためバッテリー蓄積量に応じた変化を考えられていない。

筆者らは、自律分散型の確率的制御によってバイナリーモードを選択する確率的バイナリーオフロード (PBO: Probabilistic Binary Offloading) 方式を提案した [9]。この手法は各 WD が自律分散的にオフロード確率を設定して処理を行うことで、パケット衝突を避けつつ、処理時間やエネルギー消費を抑えることを目標としている。文献 [10] では、各 WD の AP からの距離に応じてオフロード確率を決定する方法を提案し、パケット配

信率 (PDR: Packet Delivery Rate) の向上と WD のバッテリー駆動時間の伸長を計算機シミュレーションにより示した。

本稿では、タスクの処理時間と WD のエネルギー消費量と PDR のトレードオフのバランスを保つことを目的としている。これを達成するために、ユーザ効用関数を用いてバッテリー蓄積量に応じて動的に処理の優先度を変化させるオフロード確率式を提案する。当手法に WPT を組み合わせた際に、バッテリー蓄積量や通信状況の変化に応じて各 WD が効果的に計算タスクを処理できているかを計算機シミュレーションにより評価する。本稿の構成は、以下の通りである。第 2 節で本稿で使用したシステムモデルの説明と評価を行うパラメータの定義をし、第 3 節で提案方式である新たなオフロード確率の決定法を説明する。第 4 節で計算機シミュレーションの結果について説明し、第 5 節で本論文のまとめを行う。

2. システムモデル

MEC サーバを具備した 1 台の AP を中心として、 K 台の WD (集合 K) がランダムに固定配置された環境を想定する。この節では各種システムモデルについて説明する。

2.1 タスク生起モデル

各 WD の計算タスクは、ポアソン過程 [12] に従ってランダムに生起するものとする。WD $k \in K$ の i 番目のタスクが生起する時間 $t_{k,i}$ [sec] は次式で表される。

$$t_{k,i} = \begin{cases} -\frac{\log X}{\lambda}, & (i = 0) \\ t_{k,i-1} - \frac{\log X}{\lambda}, & (i \geq 1) \end{cases} \quad (1)$$

ここで、 λ はポアソン過程の到着率 [/sec]、 X は $(0, 1)$ の一様分布に従って生起される乱数である。以降はポアソン過程の到着率 λ をタスク生起率と表記する。

2.2 チャンネルモデル

オフロード時の上りリンクと下りリンク及び WPT で使用するチャンネルモデルには自由空間伝搬損失を仮定する。

WD k のチャンネル電力利得 g_k^t は次式で与えられる [11]。

$$g_k^t = \left[\frac{\lambda^t}{4\pi(d_k + 1)} \right]^2 G \quad (2)$$

ここで、 λ^t は搬送波の波長 [m]、 d_k は AP と WD 間の距離 [m]、 G は送受信アンテナの利得の積を表す。 $t = \{\text{off}, \text{wpt}\}$ とし、それぞれオフロード搬送波、WPT 搬送波に関するパラメータとする。

2.3 バイナリーオフロード

バイナリーオフロードは、WD で生じたタスクを MEC サーバへオフロードするかローカル計算を行うかの 2 つのモードをタスクごとに選択するオフロード方式である [2]。ここで、各タスクをローカル計算する場合をモード 0、オフロードする場合をモード 1 とし、WD k の i 番目タスクのモードを $m_{k,i} \in \{0, 1\}$ と定義する。バイナリーオフロードを用いた時の WP-MEC の概略図は図 1 のようになっている。

2.3.1 ローカル計算 ($m_{k,i} = 0$)

モード 0 が選択された時、各 WD は内部の計算回路を用い

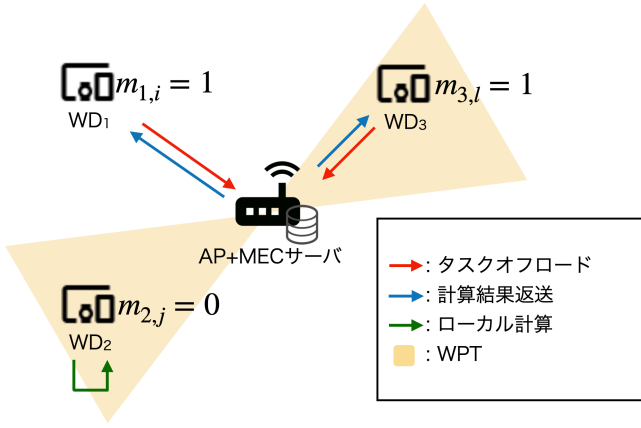


図1 バイナリーオフロードを用いた WP-MEC システムの概略図

て処理をする, ローカル計算を行う. WD k の i 番目タスクのローカル計算時間 $\tau_{k,i}^{\text{loc}}$ [sec] は次式で与えられる.

$$\tau_{k,i}^{\text{loc}} = \frac{A_k L_{k,i}}{f_k} \quad (3)$$

ここで, f_k は CPU 周波数 [Hz] ([CPU cycle/sec]), A_k は 1bit の処理に必要な CPU サイクル数 [CPU cycle/bit], $L_{k,i}$ はオフロードタスクサイズ [bit] を表す. 以降, A_k をタスクタイプと表記する. ローカル計算の消費電力は CPU 実効キャパシタンス係数 ζ_k を用いると, $\zeta_k f_k^3$ [W] で表される [15]. ローカル計算に必要なエネルギー $E_{k,i}^{\text{loc}}$ [J] は次式で表される.

$$E_{k,i}^{\text{loc}} = \zeta_k f_k^3 \tau_{k,i}^{\text{loc}} = \zeta_k f_k^2 A_k L_{k,i} \quad (4)$$

2.3.2 タスクオフロード ($m_{k,i} = 1$)

モード 1 が選択された時, 各 WD は AP へのタスクオフロードを行う. 本研究では, タスクオフロード時のランダムアクセス方式として ALOHA 方式 [13] を採用する. 全 WD はオフロード時に同一のチャンネルを使用するものとし, 複数の WD が同時にタスク送信をした場合, パケット衝突が発生するものとする. パケット衝突が発生した場合には, キャプチャ効果は考慮せず, 衝突パケットは受信失敗とする. また, AP が WD に計算結果を返送しているときにオフロードを行った場合, AP はタスクを受信できないものとする. この時, AP から WD への計算結果の返送は必ず成功するものとする.

WD はオフロード送信電力を理想的に制御可能であると仮定し, 変調方式によって与えられる最大容量を超えないように電力制御を行う. シャノンの通信路容量定理 [14] より, 送信電力 P_k^{off} [W] は次式で与えられる.

$$P_k^{\text{off}} = \begin{cases} \frac{(2^{C^{\text{max}}} - 1)\sigma^2}{g_k^{\text{off}}} & (\text{if } C_k \geq C^{\text{max}}) \\ P_{\text{WD}}^{\text{max}} & (\text{otherwise}) \end{cases} \quad (5)$$

ここで, σ^2 は雑音電力 [W], C^{max} は最大容量 [bit/sec/Hz], $P_{\text{WD}}^{\text{max}}$ は WD の最大送信電力 [W] を表す.

WD $k \in \mathcal{K}$ から AP への i 番目タスクの送信時間 $\tau_{k,i}^{\text{off}}$ [sec] は次式で与えられる.

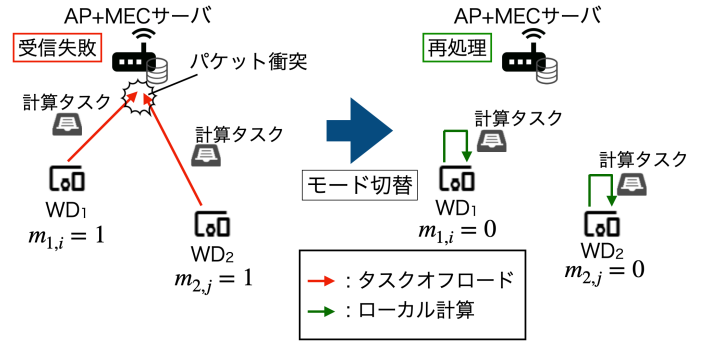


図2 モード切替の概略図

$$\tau_{k,i}^{\text{off}} = \frac{L_{k,i}}{R_k} = \frac{L_{k,i}}{B \log_2 \left(1 + \frac{g_k^{\text{off}} P_k^{\text{off}}}{\sigma^2} \right)} \quad (6)$$

ここで, R_k は送信レート [bit/sec], B は帯域幅 [Hz] を表す. オフロードに必要なエネルギー $E_{k,i}^{\text{off}}$ [J] は次式で表される.

$$E_{k,i}^{\text{off}} = \tau_{k,i}^{\text{off}} P_k^{\text{off}} = \frac{L_{k,i}}{B \log_2 \left(1 + \frac{g_k^{\text{off}} P_k^{\text{off}}}{\sigma^2} \right)} P_k^{\text{off}} \quad (7)$$

また, WD k からオフロードされた i 番目計算タスクの MEC での計算時間 τ_k^{M} [sec] は次式で与えられる.

$$\tau_k^{\text{M}} = \frac{A_k L_{k,i}}{f^{\text{M}}} \quad (8)$$

ここで, f^{M} は MEC の CPU 周波数 [Hz] ([CPU cycle/sec]) を表す. オフロードされたタスクの計算結果の返送時間 $\tau_{k,i}^{\text{re}}$ [sec] は次式で与えられる.

$$\tau_{k,i}^{\text{re}} = \frac{H_{k,i}}{R^{\text{M}}} = \frac{H_{k,i}}{B \log_2 \left(1 + \frac{g_k^{\text{off}} P^{\text{M}}}{\sigma^2} \right)} \quad (9)$$

ここで, $H_{k,i}$ は計算結果のタスクサイズ [bit], R^{M} は MEC の送信レート [bit/sec], P^{M} は計算結果の返送電力 [W] を表す. MEC の送信電力は WD の送信電力と同様に最大容量に従って電力制御を行うものとする.

2.3.3 オフロード失敗時のモード切替

通常のランダムアクセス方式では AP で WD の送信パケットを正しく受信成功した場合, 受信成功を通知する ACK(acknowledgement) 信号を WD に送信する. ACK 信号を受信できなかった WD はバックオフを用いた再送処理を行う. 一方で, MEC システムでは WD がオフロードに失敗した場合, AP からのタスク返送の有無によりオフロードの成功・失敗を判定できる. AP からのタスクの返送を受信しなかった場合, WD は再送処理を行わず, オフロードを中止してローカル計算に切り替える, モード切替を行う. この方式によって, 繰り返しの再送処理による通信環境の悪化を防ぎつつ, 全てのタスクを処理することが可能となる. 図 2 にパケット衝突時のモード切替の概略図を示す.

2.4 WPT

WPT はビームフォーミングを用いて実行する. AP は全 WD の位置を把握し, WPT は所定の角度方向に理想的にビームを

向けることができると仮定する。さらに, WPT の RF 信号の送信には N 本のアレイアンテナを用いて送信するとし, ビーム角度幅 $2\pi/N$ [rad] ずつ N 本のビームを送信できるとする。ここで, $n \in \{1, \dots, N\}$ 番目ビームの WPT 実行時間 τ_n^{wpt} [sec] を次式のように設定する。

$$\tau_n^{\text{wpt}} = \frac{K_n}{K} T \quad (10)$$

ここで, K_n は n 番目ビームにカバーされている WD の台数を表す。

n 番目のビームにカバーされている WD k の EH で得られるエネルギー量 E_k^{wpt} [J] は次式で表される。

$$E_k^{\text{wpt}} = \tau_n^{\text{wpt}} \eta_k g_k^{\text{wpt}} P^{\text{wpt}} \quad (11)$$

ここで, η_k は WD の EH 効率, P^{wpt} は AP の WPT 送信電力 [W] を表す。

2.5 タスク処理時間及びエネルギー蓄積量

本研究では処理遅延及び WD のエネルギー蓄積量を評価する。タスクが生起してからオフロードタスクの計算結果を受信完了するまで, もしくはローカル計算が完了するまでをタスク処理時間と定義する。WD k の i 番目タスク処理時間 $T_{k,i}$ [sec] は次式で表される。

$$T_{k,i} = \begin{cases} \tau_{k,i}^{\text{loc}} & (m_{k,i} = 0) \\ \tau_{k,i}^{\text{off}} + \tau_{k,i}^{\text{M}} + \tau_{k,i}^{\text{re}} & (m_{k,i} = 1, \text{if success}) \\ \tau_{k,i}^{\text{off}} + \tau^{\text{wait}} + \tau_{k,i}^{\text{loc}} & (m_{k,i} = 1, \text{if failure}) \end{cases} \quad (12)$$

ここで, τ^{wait} は計算結果返送の待機時間 [sec] を表す。

また, WD のエネルギー蓄積量はオフロード送信, ローカル計算, EH のみに依存すると仮定する。 i 番目タスクの処理終了時に WD のバッテリーに充電されているエネルギー蓄積量 $\varepsilon_{k,i}$ [J] は次式で表される。

$$\varepsilon_{k,i} = \begin{cases} \min \left(\varepsilon_{\max}, [\varepsilon_{k,i-1} + E_k^{\text{wpt}} - E_{k,i}^{\text{loc}}]^+ \right) & (m_{k,i} = 0) \\ \min \left(\varepsilon_{\max}, [\varepsilon_{k,i-1} + E_k^{\text{wpt}} - E_{k,i}^{\text{off}}]^+ \right) & (m_{k,i} = 1) \end{cases} \quad (13)$$

ここで, ε_{\max} は最大バッテリー容量 [J], $[x]^+$ は $\max(x, 0)$ の関数を表す。また, $\varepsilon_{k,0}$ は初期バッテリー蓄積量 [J] を表す。

2.6 確率的バイナリオフロード

筆者らは, 確率的制御を用いてタスク処理方法を選択する確率的バイナリオフロード (PBO: Probabilistic Binary Offloading) 方式を提案した [9]。この手法では WD $k \in \mathcal{K}$ がオフロード確率 $p_k \in [0, 1]$ を自律分散的に決定する。タスク生起の度にオフロードとローカル計算のどちらのモードで処理するかをオフロード確率に基づいて決定することで, 通信環境の悪化や処理時間とエネルギー消費量の落ち込みを避けることができる。例えば, $p_k = 1.0$ の時はオフロードを行うモード 1, $p_k = 0.0$ の時ローカル計算を行うモード 0 を確実に選択し, $p_k = 0.5$ の時は 2 つのモードを 1/2 の確率で選択し, 処理を行う。

2.7 ユーザ効用関数

文献 [8] ではユーザ効用関数を用いた, MEC における処理時間とエネルギー消費量双方のバランスを保つような AP での集中制御による最適化法を検討している。WD k の i 番目タスク処理時のユーザ効用関数 $v_{k,i}$ は次式のように与えられている。

$$v_{k,i} = m_{k,i} \left[\beta_{k,i} \frac{\tau_{k,i}^{\text{loc}} - \tau_{k,i}^{\text{off}}}{\tau_{k,i}^{\text{loc}}} + (1 - \beta_{k,i}) \frac{E_{k,i}^{\text{loc}} - E_{k,i}^{\text{off}}}{E_{k,i}^{\text{loc}}} \right] \quad (14)$$

ここで, $\beta_{k,i}$ は処理時間とエネルギー消費量の処理優先度を表し, $\beta_{k,i} \in [0, 1]$ を満たす。優先度を変化させることで WD ごとに処理時間とエネルギー消費量のトレードオフのバランスを操作することができる。式中の各分数はオフロードとローカル計算の処理時間およびエネルギー消費量の比を表し, オフロードでの負荷が小さいほど v_k が大きな値を取る。

3. 提案手法

3.1 ユーザ効用関数の優先度

文献 [10] では WD-AP 間距離にのみ依存した時不変の簡易なオフロード確率式を提案した。しかし, この確率式はエリア端に位置する WD の処理時間の増加や PDR の低下などの問題があった。本稿では, 時間とともに変化するバッテリー蓄積量および通信環境に応じて動的にオフロード確率を更新する確率式を提案する。WD のバッテリー蓄積量が多い際は処理時間を優先し, 少ない際はエネルギーを優先するような処理にするために, 2 つの処理の優先度を決められるユーザ効用関数を利用する。ここで, ユーザ効用関数の優先度 $\beta_{k,i}$ を次式のように定義する。

$$\beta_{k,i} \triangleq \frac{\varepsilon_{k,i-1}}{\varepsilon_{\max}} \quad (15)$$

3.2 オフロード負荷の期待値

文献 [8] で定義されているユーザ効用関数は集中制御を前提とするため, オフロードの失敗が考慮されていない。そこで, 各 WD は自身のオフロードの成功数を把握していることからオフロード処理時の処理時間およびエネルギー消費量の期待値を計算し, 利用する。オフロードの負荷を PDR から計算することが考えられるが, 各 WD の PDR は時間に応じて累積されていくため長時間経過すると収束してしまい, 通信状況に応じたオフロード確率の変化ができなくなってしまう。これを避けるために, 過去 γ 回のオフロードの PDR $s_{k,i}$ を用いた期待値計算式を考える。各期待値は次式のように求められる。

$$\begin{aligned} \mathbb{E}[\tau_{k,i}^{\text{off}}] &= s_{k,i} \cdot \tau_{k,i}^{\text{off}} + (1 - s_{k,i}) \cdot (\tau_{k,i}^{\text{off}} + \tau^{\text{wait}} + \tau_{k,i}^{\text{loc}}) \\ &= \tau_{k,i}^{\text{off}} + (1 - s_{k,i}) \cdot (\tau^{\text{wait}} + \tau_{k,i}^{\text{loc}}) \end{aligned} \quad (16)$$

$$\mathbb{E}[E_{k,i}^{\text{off}}] = E_{k,i}^{\text{off}} + (1 - s_{k,i}) \cdot E_{k,i}^{\text{loc}} \quad (17)$$

3.3 オフロード確率式

オフロード確率式は $0 \leq p_{k,i} \leq 1$ の制約を満たす必要がある。ユーザ効用関数は $v_{k,i} \leq 1$ の値を取るため, 関数 $[x]^+ = \max(x, 0)$ を用いることで確率の制約を満たすようにする。そこで, 次のような 2 つのオフロード確率式を提案する。

- 式全体に制限をかける場合 (Form 1)

表 1 シミュレーション諸元

パラメータ	値
最大通信距離 d^{\max}	50 [m]
WD 数 K	50
タスクタイプ A_k	10^3 [CPU cycles/bit]
ACK 信号待機時間 T^{ack}	0.5 [sec]
生起タスクサイズ L_k	[10, 100] [kbits]
計算結果のタスクサイズ H_k	$[L_k/100, L_k/10]$ [bits]
CPU キャパシタンス係数 ζ_k	10^{-28}
EH 効率 η_k	0.3
タスク生起率 λ	0.1 [1/sec]
WD の CPU 周波数 f_k	[10, 50] [MHz]
MEC の CPU 周波数 f^{M}	10 [GHz]
WPT 実行時間 τ_k^{wpt}	1.0 [sec]
雑音電力 σ^2	10^{-11} [W]
AP の WPT 送信電力 P^{wpt}	3 [W]
WPT 搬送波周波数 f_c^{wpt}	915 [MHz]
オフロード搬送波周波数 f_c^{off}	2.4 [GHz]
送受信アンテナ利得の積 G	4.11
WPT 送信アンテナ数 N	4
WPT 受信アンテナ数 M	1
オフロード PDR の参照回数 γ	10
最大バッテリー容量 ε_{\max}	10^{-1} [J]

$$p_{a,k,i} = \left[\beta_{k,i} \frac{\tau_{k,i}^{\text{loc}} - \mathbb{E}[\tau_{k,i}^{\text{off}}]}{\tau_{k,i}^{\text{loc}}} + (1 - \beta_{k,i}) \frac{E_{k,i}^{\text{loc}} - \mathbb{E}[E_{k,i}^{\text{off}}]}{E_{k,i}^{\text{loc}}} \right]^+ \quad (18)$$

当提案式は式全体に関数 $[x]^+$ をかけることで確率の制約を満たしている。ユーザ効用関数を元の形で使用しているが、処理時間とエネルギー消費量どちらかの比が負値をとり、オフロード確率が 0 になる可能性がある。

- モードによる負荷の比に制限をかける場合 (Form 2)

$$p_{b,k,i} = \beta_{k,i} \left[\frac{\tau_{k,i}^{\text{loc}} - \mathbb{E}[\tau_{k,i}^{\text{off}}]}{\tau_{k,i}^{\text{loc}}} \right]^+ + (1 - \beta_{k,i}) \left[\frac{E_{k,i}^{\text{loc}} - \mathbb{E}[E_{k,i}^{\text{off}}]}{E_{k,i}^{\text{loc}}} \right]^+ \quad (19)$$

当提案式はローカル計算とオフロードの処理時間とエネルギー消費量のそれぞれの比に関数 $[x]^+$ をかけることで確率の制約を満たしている。一方の比が負値をとった場合でもオフロード確率が 0 になる可能性が Form 1 よりも低くなる。

これら 2 式をそれぞれ用いて計算機シミュレーションを行い、評価する。

4. 計算機シミュレーション

4.1 シミュレーション諸元

シミュレーション諸元を表 1 に示す [6] [7]。ここで、WD $k \in \mathcal{K}$ の位置は距離 $d_k \sim \mathcal{U}(0, 50)$ 、角度 $\theta_k \sim \mathcal{U}(0, 2\pi)$ [rad] によって決定され、固定配置されているものとする。

また、WD のオフロード送信パケットの最大変調方式は QPSK (Quadrature Phase Shift Keying) とするため、最大チャネル容量は $C^{\max} = 2$ [bit/sec/Hz] となる。帯域幅を $B = 2$ [MHz] と設定し、最大送信レートは $R^{\max} = 4$ [Mbits/sec] とする。WD のオフロード最大送信電力は $P_{\text{WD}}^{\max} = 20$ [mW]、AP の計算結果の最大返送電力は $P_{\text{M}}^{\max} = 100$ [mW] とする。

4.2 シミュレーション結果

4.2.1 PDR の比較

自律分散型制御による通信品質の低下を確認するため、オフロード確率式 $p_{a,k,i}$ (Form 1) を用いた場合と $p_{b,k,i}$ (Form 2) を用いた場合の各オフロード確率式を用いたときの PDR を比較を行う。図 3 に時間に対する全 WD のオフロード送信の PDR を示す。ここで、初期処理優先度 $\beta_{k,0} = \{0.1, 0.5, 0.9\}$ は初期バッテリー蓄積量の最大バッテリー容量との比を表す。結果より、初期バッテリー蓄積量小さいほど PDR が大きくなること示された。検討したモデルではオフロード時間がローカル計算時間よりも短くなるため、バッテリー蓄積量が多く処理時間の優先度が高いほどオフロード確率が高くなり、PDR が低くなると考えられる。しかし、提案手法では端末ごとの PDR に合わせてオフロード確率を変化させることで、大幅な PDR の低下を引き起こさずに処理できている。また、システムの開始直後から 500 秒程度で PDR が高い状態に安定することから、システム全体の堅牢性も保たれていると考えられる。

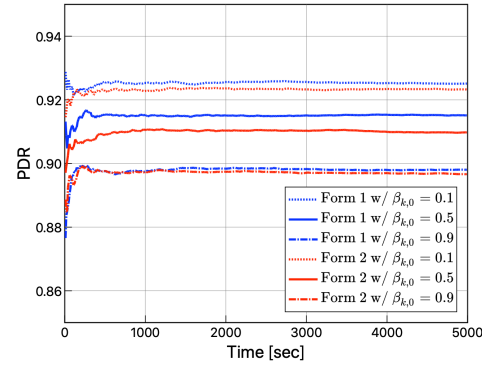


図 3 時間に対するオフロード PDR

4.2.2 1bit あたりの処理時間とエネルギー消費量の比較

オフロード確率式 Form 1 を用いた場合と Form 2 を用いた場合のタスク 1bit を処理するために必要な処理時間およびエネルギー消費量を比較する。比較手法として、各 WD が他の WD のオフロードタイミングを把握し、オフロードを待機して衝突回避 (CA: Collision Avoidance) を行うことができる理想的な集中制御モデルを用いる。この手法では、本来制御情報のやり取りやオフロードの待機で余計な遅延やエネルギー消費が発生してしまう。しかし、本評価においては簡単のため、理想的に情報を把握しこれらのオーバーヘッドは発生しないものとする。

図 4 に初期処理優先度 $\beta_{k,0} = \{0.1, 0.5, 0.9\}$ のときの 1bit あたりの処理時間の比較を示す。結果より、初期処理優先度 $\beta_{k,0} = 0.5$ のとき、CA なしの場合には理想的な CA ありの場合に比べて、Form 1 を用いた場合は約 1.56 倍、Form 2 を用いた場合は約 1.72 倍の処理時間に抑えられていることがわかる。オ

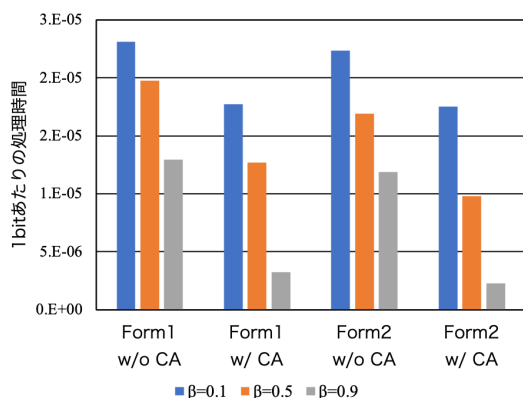


図4 1bitあたりの処理時間

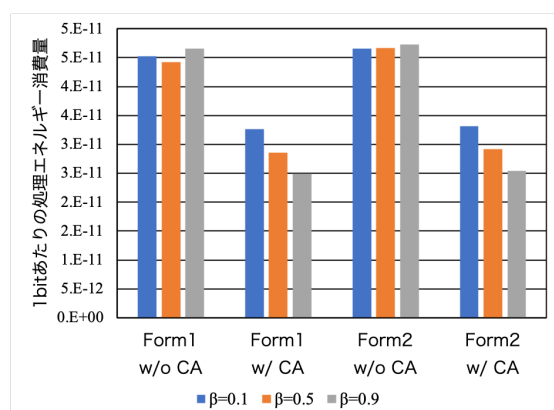


図5 1bitあたりのエネルギー消費量

フロードが失敗した場合モード切替によって遅延が発生してしまうため、提案手法ではPDRを90%近くに保っているものの、CAありの場合より処理時間が長くなっている。しかし、実際にCAを行うためにはAPや他WDとの制御情報のやり取りなどのオーバーヘッドがあるため、提案手法は集中制御を用いた場合の処理時間に近づくことができていると考えられる。また、バッテリー蓄積量が大きく処理時間を優先するようになるにつれて、1bitあたりの処理時間を大きく低減することができる。

図5に1bitあたりのエネルギー消費量の比較を示す。結果より、初期処理優先度 $\beta_{k,0} = 0.5$ のとき、CAなしの場合は理想的なCAありの場合に比べて、Form1を用いた場合は約1.55倍、Form2を用いた場合は約1.60倍のエネルギー消費量に抑えられている。また、PDRを低く保つことができているため、初期処理優先度が高い場合でもエネルギー消費量を大きく増大させることなく処理できている。

5. まとめ

本稿では、確率的バイナリーオフロードを用いたWP-MECシステムにおいて、WDの状況に応じてオフロード確率を変化させるモデルを提案した。各WDのバッテリー蓄積量とPDRを参照することで、処理時間とエネルギー消費量の優先度を変化させ、動的にオフロード確率を変化することを可能にした。これにより処理時間、エネルギー消費量、PDRのトレードオフのバランスを保つことを評価するために計算機シミュレーション

を行った。シミュレーション結果により、PDRを初期バッテリー蓄積量に関わらず、90%程度に安定させることができることを示した。また、1bitあたりの処理時間とエネルギー消費量より、理想的に衝突回避を行うことができるモデルに比べて、大幅な遅延やエネルギー消費を引き起こさずに処理が可能であることが確認された。

謝辞 This work was supported by the European Commission in the framework of the H2020-EUJ-02-2018 project 5G-Enhance (Grant agreement no.815056) and the Ministry of Internal Affairs and Communications (MIC) of Japan.

文 献

- [1] A. A. Fuqaha, et al., "Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 4, pp. 2347–2376, 4th Quart., 2015.
- [2] Y. Mao, C. You, J. Zhang, K. Huang, K. B. Letaief, "A survey on mobile edge computing: The communication perspective," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 4, pp. 2322–2358, Aug. 2017.
- [3] Q. Zhang, L. Cheng, and R. Boutaba, "Cloud Computing: State-of-the-Art and Research Challenges," *J. Internet Services Appl.*, vol. 1, no. 1, pp. 7–18, May 2010.
- [4] Y. Zeng, B. Clerckx, R. Zhang, "Communications and signals design for wireless power transmission," *IEEE Trans. Commun.*, vol. 65, no. 5, pp. 2264–2290, May 2017.
- [5] W. Zhang, et. al., "Energy-Optimal Mobile Cloud Computing Under Stochastic Wireless Channel," *IEEE Trans. Wireless Commun.*, vol. 12, no. 9, pp. 4569–4581, Sep. 2013.
- [6] F. Wang, J. Xu, X. Wang, S. Cui, "Joint Offloading and Computing Optimization in Wireless Powered Mobile-Edge Computing Systems," *IEEE Trans. Wireless Commun.*, vol. 17, no. 3, pp. 1784–1797, Mar. 2018.
- [7] S. Bi, Y. J. Zhang, "Computation Rate Maximization for Wireless Powered Mobile-Edge Computing With Binary Computation Offloading," *IEEE Trans. Wireless Commun.*, vol. 17, no. 6, pp. 4177–4190, Jun. 2018.
- [8] X. Lyu, H. Tian, C. Sengul, and P. Zhang, "Multiuser Joint Task Offloading and Resource Optimization in Proximate Clouds," *IEEE Trans. Vehicular Tech.*, vol. 66, no. 4, Apr. 2017.
- [9] 小林 拓弥, 安達 宏一, "確率的バイナリーオフロードを用いた Wireless Powered MEC の特性評価," 信学技報, vol. 120, no. 74, RCS2020-24, pp. 7-12, 2020年6月.
- [10] 小林 拓弥, 安達 宏一, "ireless Powered MEC における確率的バイナリーオフロード及びモード切替方式の評価," 信学ソ大, B-5-66, 2020年9月.
- [11] H. Kawabata, et al., "Robust Relay Selection for Large-Scale Energy-Harvesting IoT Networks," *IEEE Internet of Things Journal*, vol. 4, no. 2, Apr. 2017.
- [12] A. F. Molisch, *Wireless Communications Second Edition*, United States of America: Wiley-IEEE Press, 2011.
- [13] B. Sklar, and F. J. Harris, *Digital Communications: Fundamentals and Applications*, Prentice-hall Englewood Cliffs, NJ, 1988.
- [14] C. E. SHANNON, "A Mathematical Theory of Communication," *The Bell System Technical Journal*, vol. 27, pp. 379–423, Jul., 1948.
- [15] Y. Wang, et al., "Mobile-Edge Computing: Partial Computation Offloading Using Dynamic Voltage Scaling," *IEEE Trans. Commun.*, vol. 64, no. 10, pp. 4268–4282, Oct. 2016.