CM-003

Probe Request フレームの到着時間間隔および周期性に着目したランダム MAC アドレスにも対応可能な Wi-Fi 端末の間接推定法の提案

A Proposal for Indirect Estimation of Wi-Fi Device Utilizing Random MAC Address by Focusing on the Inter-arrival Time and Periodicity of Probe Request Frames

齋 進† 杉浦 彰彦†
Jin Qi Akihiko Sugiura

1. はじめに

近年、Wi-Fi 設備がますます普及し、公共ネットワーク以外に、家庭用ネットワークにおいても Wi-Fi ルーターを使用するようになってきた。しかし、地震や津波など災害発生時に通信インフラが破壊される場合、被災者は Wi-Fi ネットワークに接続できない可能性がある。このような情報難民を見つけるため、Wi-Fi アクセスポイントに接続していない端末を追跡する必要がある。

Wi-Fi 端末はアクセスポイントを発見するため、周期的に一連の Probe Request フレームを送信する［1］。これらのフレームに MAC アドレス、シーケンス番号、ベンダー情報などの端末情報を含めるため、研究者たちは Probe Request フレームの観察による端末追跡を行ってきた。例えば、Freudiger らはシーケンス番号を分析することにより端末の数量を判断する方法を提案した［2］。しかし、近年ではセキュリティの観点から多くの端末はシーケンス番号の乱数化を実施した［3］。Vanhoef らはベンダー情報などフレームボディに含める端末情報を利用することにより端末を特定する方法を提案した［4］。しかし、一部の端末はベンダー情報を隠蔽するようになり、またフレームボディに含める端末情報を利用するだけで同機種の複数台端末は識別できないことが明らかである。齋らは MAC アドレスを観察することにより災害発生時 Wi-Fi に接続できない情報難民の状況を把握する方法を提案した［5］。しかし、ここ数年ではより多くのメーカーはランダム MAC アドレスを採用した［6］。従って、端末の個別追跡がより難しくなっている。端末は、短時間で複数の Wi-Fi チャンネルに Probe Request フレームを送信し、その後一定時間が経過してからもう一度 Probe Request を送信する［7］。Franklin らは Probe Request の時間特性を特徴量として利用し、これに基づいて特定の端末を識別する方法を提案した［8］。しかし、複数の端末が存在する場合、各端末からの Probe Request フレームは混在している。そのため、特徴量を通じて個別端末に対する識別、追跡を行うことは困難である。

本研究では、端末が送信する一連の Probe Request フレームの到着時間間隔および周期性に着目し、時系列の Probe Request データを分析することにより端末追跡を行う。まず、Raspberry Pi 4 に基づくスニファーを製作し、空間中の Probe Request フレームを収集する。このスニファーはモニターモードで動作するため、Wi-Fi ネットワークに干渉を与えない状態で情報を収集することが可能である。Raspberry Pi の寸法が小さく消費電力も低いため、災害発生時にこのスニファーをドローンに搭載し、被災地域を巡回しながら Wi-Fi フレームを収集することができる。また、隣接する Probe Request フレームとの到着時間間隔に基づき、端末が各時間帯に出すフレームを分離することを試みた。さらに、電波強度データの周期性を分析し、分析結果により個別端末に対する追跡を行った。実機実験では、連続する四つの Wi-Fi チャンネルをスニッフィングすることで、市販品の 10 台のスマートフォンに対して個別追跡を行う。従来の研究では一つの Wi-Fi チャンネルだけに対してスニッフィングを行ったが［9］、連続する四つのチャンネルをスニッフィングする場合、より多くのフレームを記録できるため、端末を識別、追跡する精度を向上することができる。実験結果から、本提案手法を用いることにより個別端末を高精度で識別、追跡できることが証明された。

†静岡大学創造科学技術大学院
Graduate School of Science and Technology,
Shizuoka University

2. Probe Request の原理

2.1 Probe Request 概要

端末が Wi-Fi ネットワークに接続するため、アクセスポイントからネットワーク名、利用できるチャンネルなどの情報を受信する必要はある。端末がアクセスポイントを発見する方法は、パッシブスキャン（Passive Scan）およびアクティブスキャン（Active Scan）という二種類ある。パッシブスキャン（Passive Scan）では、端末はアクセスポイントからブロードキャストされる Beacon フレームを受信し、その中に含まれる必要な情報を獲得する。アクティブスキャン（Active Scan）では、端末が Probe Request フレームをブロードキャストし、それを受信した後アクセスポイントは Probe Response で応答する。端末はアクセスポイントの応答から必要な情報を獲得することができる。

2.2 Probe Request フレームの到着時間間隔

移動端末 Probe Request フレームの到着時間間隔および周期を図 1 にて示す。端末は、短時間で複数の Wi-Fi チャンネルに Probe Request のグループを送信し、一定時間が経過してからもう一度 Probe Request のグループを送信する。本稿では、到着時間間隔は隣接する 2 フレーム間の時間間隔を指し、総到着時間間隔は連続する四つのチャンネルにあるフレームの到着時間間隔の総和を指す。また、周期とは、端末が一つのチャンネルにフレームを出してから再びこのチャンネルにフレームを出すまでの時間間隔である。

本研究で実際に使用した一台の移動端末が出した Probe Request フレームの実測値を図 2 に示す。図 1 と同じように、オレンジ色のフレームは一つの周期内にあり、青色のフレームは次の周期内にある。端末の Wi-Fi 設定が“On”になっている状態で、36 チャンネル 80MHz 帯域において 8

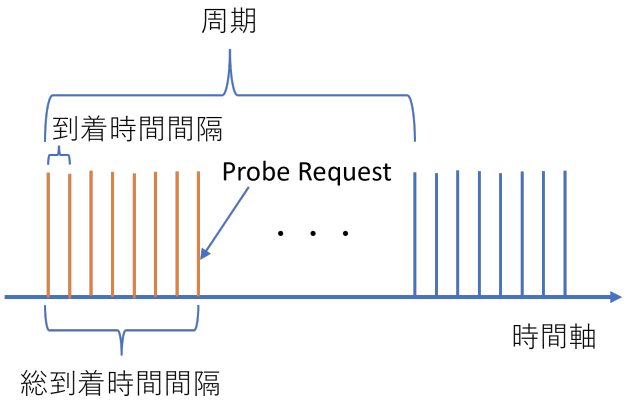

図 1　Probe Request フレーム
到着時間間隔、周期

Time	Signal strength (dBm)
10.025903	-12 dBm
10.046146	-16 dBm
10.071552	-12 dBm
10.09177	-10 dBm
10.117212	-14 dBm
10.137639	-10 dBm
10.160543	-12 dBm
10.180707	-12 dBm
・・・・・・	
20.017384	-12 dBm
20.037708	-16 dBm
20.062077	-10 dBm
20.082288	-14 dBm
20.106845	-14 dBm
20.126899	-10 dBm
20.150138	-14 dBm
20.170331	-12 dBm

図 2　到着時間間隔、周期の実測

個の Probe Request フレームが確認された。隣接フレーム間の到着時間間隔は約 0.02 秒であり、同じグループ内の 8 フレーム間の総時間間隔は約 0.15 秒である。また、端末が周期的に Probe Request フレームをブロードキャストするが、この周期は明らかに同一グループ内にあるフレームの到着時間間隔より長いのである。例えば、図 2 に示しているように、今回の例では、グループの周期は約 10 秒である。また、図中の信号強度データは 3.3 にて示す周期分析で使用する。

Probe Request フレーム到着時間間隔の特徴に基づき、私たちは受信したフレームを複数のグループに分け、同一時間帯にあるフレームを同じグループに追加する。本研究では、フレーム分類およびステージ分類という二種類の方法を提案した。

フレーム分類では、隣接フレームとの時間間隔が基準時間 T より小さいかまたは等しい場合は、これらのフレームをグループ化する。一方で、時間間隔が基準時間 T より大きい場合は、新しいグループを作成し、この隣接フレームを新グループに追加する。同一端末にとって、周期内にあるフレームの到着時間間隔が周期の間隔より遥かに小さいため、適切な基準時間 T を設定することにより、同じ周期内のフレームを同一グループ内に分けることができる。具体的な計算方法について、3.1 にて紹介する。

ステージ分類では、基準時間間隔 T（ステージ 1）、T の 2 倍値（ステージ 2）、T の 4 倍値（ステージ 3）という三種類の時間間隔を設定し、各時間間隔内にあるフレームの数量を計算する。これらのフレーム数により、適用するステージを決定する。例えば、ステージ 2 のフレーム数がステージ 1 のフレーム数のx倍以上である場合は、ステージ 2 内のフレームをグループ化する。一方で、ステージ 3 のフレーム数がステージ 2 のフレーム数のx倍以上である場合は、ステージ 3 内のフレームをグループ化する。ステージ分類を用いて、より高精度でグループ化を行うことができると想定する。端末の機種が非常に多く、異なる機種の Probe Request フレーム到着時間間隔は同じではない可能性がある。フレームを三つのステージに分類することにより、異なる到着時間間隔に対応することが可能になる。具体的な計算方法について、3.2 にて紹介する。

上記二種類の方法はいずれもフレームに対してグループ化を行うことができる。ただし、複数の端末は同じ時間帯に Probe Request フレームを送信する可能性があるので、グループの中に含める端末の台数を推定することが必要である。もし端末の台数を推定しなければ、一つのグループが一台の端末だけになり、ほかの端末を見逃す可能性はある。私たちは各グループのフレーム数を計算し、並びにこのフレーム数により端末の台数を推定する。この方法を使用することにより、複数の端末が同時に存在する時にも、個別の端末を識別することが可能である。

2.3 Probe Request フレームの周期性

Probe Request フレームの周期性に基づき、私たちは個別端末を追跡する。

まず、Probe Request フレームの信号強度データに対して周期性分析を行う。端末の送信電力や位置が異なるため、スニファーは各端末から受信したフレームの電波強度も同じではない。短時間にスニッフィングを行う場合、同一端末の位置が変更しなく信号強度も同じであると想定する。そのため、信号強度データの周期性を分析することにより、各端末の Probe Request 周期を推定することができる。本研究では、信号強度データの相関関数により周期性を分析する。先頭から一部のデータを抽出し、ウィンドウ関数として使用する。次に、このウィンドウ関数と全データの相関関数を計算する。相関関数から一連の極大値を取り出し、これらの極大値の横軸（時間軸）の座標値が信号強度データの周期であると推定する。さらに、信号強度データの周

期が各端末の Probe Request フレームの送信周期であると想定する。

また、得られた周期情報により個別端末に対して追跡を行う。具体的には、識別された端末に対し、算出した周期が経過した後、私たちは端末がもう一度出現するかどうか確認する。確認された場合は、この端末に対する追跡が成功したと判定する。確認されていない場合は、追跡が失敗したと判定する。具体的な計算方法について、3.3 にて紹介する。

3. Probe Request フレームの到着時間間隔および周期の分析

環境中に複数台の端末が存在する場合、各端末からの Probe Request フレームは混在している。また、同一端末からのフレームも周期的に時間軸上に出現する。本研究では、フレームの到着時間間隔により各端末が異なる周期に送信した Probe Request を分離する。また、周期性分析の結果により各端末に対して個別追跡を行う。

3.1 到着時間間隔によるフレーム分類

本方法では、到着時間間隔により、フレームを異なるグループに分類する。スニファーを使用して環境中の Probe Request を記録した後、最初に受信した Probe Request から隣接するフレームの到着時間間隔を計算する。また、基準時間を 0.165 秒に設定し（T = 0.165 秒）、算出した時間間隔が 0.165 秒より小さいかまたは等しい場合、これらのフレームを同一グループに統合する。逆に、時間間隔が 0.165 秒より大きい場合は、この隣接フレームを最初の要素として新しく生成されてグループに追加する。図 2 の端末は、同じ周期内にある 8 フレームの総時間間隔が約 0.15 秒であるが、本研究では、確実にフレームをグループ化できるようにマージンを設けたため、基準時間を 0.165 秒に設定した。

仮に、デルタ関数を使用して Probe Request フレームを記述する。例えば、時間t_nにフレームを受信した場合、このフレームを以下の数式で表す。

$$f_n(t) = \delta(t - t_n) \tag{1}$$

従って、すべてのスニッフィングしたフレームは以下の数式で表示することができる。

$$F(t) = \sum_{n=1}^{M} \delta(t - t_n), \{t_1, t_2, \cdots, t_M\}\text{は到着時間} \tag{2}$$

仮に各グループ中のフレーム数はN_iであり、iはグループの番号である。まず、最初に受信したフレーム $f_1(t) = \delta(t - t_1)$から、0.165 秒以内のフレーム数を求める。計算式は以下の通りである。

$$N_1(1) = \int_{t_1}^{t_1+0.165} \sum_{n=1}^{M} \delta(t - t_n) \tag{3}$$

$N_1(1) = 1$の場合は、このグループの中に$\delta(t - t_1)$しかないため、次に新しいグループを作成し、$f_2(t) = \delta(t - t_2)$から新グループ中のフレーム数を計算する。

$N_1(1) > 1$の場合は、引き続き以下の区間内のフレーム数を計算する：

$$N_1(2) = \int_{t_1}^{t_{N_1(1)}+0.165} \sum_{n=1}^{M} \delta(t - t_n) \tag{4}$$

このように、$N_1(1), N_1(2), \cdots, N_1(x)$まで計算し、$N_1(x) = N_1(x-1)$の場合は、このグループが終了する。グループ中のフレーム数N_1は$N_1(x)$である。次に、新しいグループを作成し、以下のフレームから新グループ中のフレーム数を計算する：

$$f_{N_1+1}(t) = \delta\left(t - t_{N_1+1}\right) \tag{5}$$

第Y番のグループを作成したときに、仮にその先頭にあるフレームの時間は t_y であり、y の計算式は以下の通りである。

$$y = \sum_{i=1}^{Y-1} N_i + 1, Y > 1 \tag{6}$$

この方法を通じ、混在している Probe Request を複数のグループに分けることができる。

3.2 到着時間間隔によるステージ分類

本研究では、三種類の時間間隔 0.165 秒、0.33 秒、0.66 秒を設定する（ステージ 1 = 0.165 秒、ステージ 2 = 0.33 秒、ステージ 3 = 0.66 秒）。最初に受信したフレームから計算し、0.33 秒内にあるフレームの数量が 0.165 秒内の数量の 1.5 倍より小さい場合は、ステージ 1 を適用し、0.165 秒内のフレームをグループ化する。逆に、0.33 秒内の数量が 0.165 秒内の数量の 1.5 倍より大きいまたは等しい場合は、私たちは 0.33 秒内の数量を 0.66 秒内の数量と比較させる。0.66 秒内の数量が 0.33 秒内の数量の 1.5 倍より小さい場合にステージ 2 を適用し、逆にステージ 3 を適用する。

最初に受信したフレーム$\delta(t - t_1)$から計算し、各ステージにおけるフレーム数は以下の通りである。

$$N = \int_{t1}^{t1+stage} \sum_{n=1}^{M} \delta(t - t_n), \tag{7}$$

$stage = 0.165$ 秒, ステージ 1 の場合
$stage = 2 \times 0.165$ 秒, ステージ 2 の場合
$stage = 4 \times 0.165$ 秒, ステージ 3 の場合

ステージ 2 のフレーム数がステージ 1 のフレーム数の 1.5 倍より小さい場合は、 ステージ 1 を決定し、逆にステージ 2 を昇格する。同じように、ステージ 2 に昇格後、ステージ 3 のフレームがステージ 2 のフレーム数の 1.5 倍より小さい場合は、ステージ 2 を決定し、逆にステージ 3 を決定する。次に、決定したステージ内のフレームをグループ化し、以下のフレームから新しいグループを作成する：

$$\delta(t - t_{N+1}), \begin{cases} t_{N+1} - t_1 > 0.165, \text{ステージ 1 の場合} \\ t_{N+1} - t_1 > 0.33, \text{ステージ 2 の場合} \\ t_{N+1} - t_1 > 0.66, \text{ステージ 3 の場合} \end{cases} \tag{8}$$

3.1 或いは 3.2 のグループ化を行った後、私たちは各グループ内のフレーム数により、端末の台数を推定する。まず、閾値 Th を設定し、フレーム数が Th を下回るグループを廃棄する。ここで、閾値 Th は端末を認定する際に必要最低限の受信フレーム数を示し、全体の受信状況に応じて一意に設定される。また、残ったグループに対し、フレーム数を 8 で割り、仮に商は Q であり、余りは R である。端末の台数を推定する計算式は、以下の通りである。

$$台数 = \begin{cases} Q, & R < Th \\ Q+1, & R \geq Th \end{cases} \tag{9}$$

3.3 周期性分析を用いた端末推定

Wi-Fi スニファーは、Probe Request フレームの MAC 層情報の他に、信号強度など物理層情報も記録可能である。本研究では、Probe Request フレームの信号強度データを利用し、周期性分析を行う。また、データの周期性に基づき、各端末を追跡する。本手法は二つの部分により構成される。

第一に、信号強度データに基づき、周期情報を推定する。まず、信号強度データに対し、0.01 秒間隔でゼロパディングを行う。また、最初に受信したフレームからT_w以内のデータを抽出し、このデータと全データの相関関数を計算する。本研究では、使用されているすべての端末が一つのチャンネルに Probe Request フレームを出してから再びこのチャンネルにフレームを出すまでの最大時間間隔は 15 秒以内である。そのため、T_wを 15 秒に設定する（$T_w = 15$秒）。さらに、相関関数の極大値から周期性を推定する。

ゼロパディング後の電波強度データを表す数式は以下の通りである。

$$S = \sum_{i=1}^{N} A_i, \quad i\text{は離散時間}, A_n\text{は電波強度} \tag{10}$$

仮に 15 秒以内にあるデータの離散時間の最大値が L であり、L は以下の条件を満足する：

$$t_L \leq 15, t_{L+1} > 15 \tag{11}$$

相関関数の計算式は以下の通りである。

$$C(j) = \sum_{i=1}^{L} A_i \cdot A_{i+j}, 0 \leq j \leq N - L \tag{12}$$

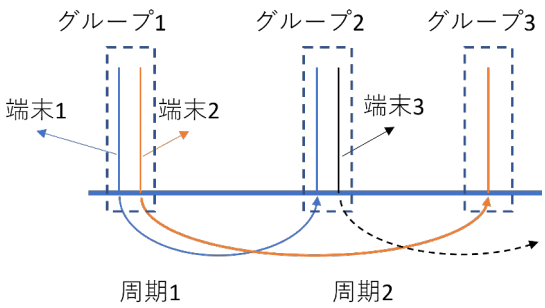

図 3　端末追跡

相関関数$C(j)$の極大値の位置を検出し、それに対応する時間が Probe Request フレームの周期であると推定する。抽出する極大値の数量について、最初の 15 秒で認定された端末の台数と同等にする。

第二に、図 3 に示したように、第 3 章で得られた結果および推定した周期により個別端末を追跡する。まず、最初の 15 秒内にあるグループを点数の大きい順に並べ替え、点数が大きいグループから端末追跡を始める。推定した周期が経過した後、新しい時間帯に端末が出現するかどうか確認する。一定周期経過後に端末が再度認定された場合、私たちはこの端末に対する識別、追跡が成功したと判断する。

4. 実機実験

4.1 実験環境

本研究では、市販のスマートフォンから 10 台を選定し、教室の中で実機実験を行う。Raspberry Pi4 および外付け Wi-Fi アダプターを使用してスニファーを製作し、モニターモードで 36 チャンネル 80MHz 帯域幅内の Probe Request 信号を収集する。

10 台の端末のうち、5 台は iPhone であり、残りの 5 台は Android である。そのため、iPhone および Android おいて本提案の有効性を検証することができる。また、この二種類の OS はほとんどの市場シェアを占めているので、本提案の汎用性を確保することもできる。

5 台の Android 端末のうち、2 台は同じ機種である。それにより、本提案を使用して同機種の複数台端末を識別できるかどうか検証することができる。私たちはこれらの Android 端末に A、B、C、D、E の端末名を付ける。

5 台の iPhone のうち、異なる年代の機種は混在している。従って、本提案が新旧バージョンの iOS および iPhone 端末に対する有効性を検証することができる。私たちはこれらの iPhone に F、G、H、I、J の端末名を付ける。

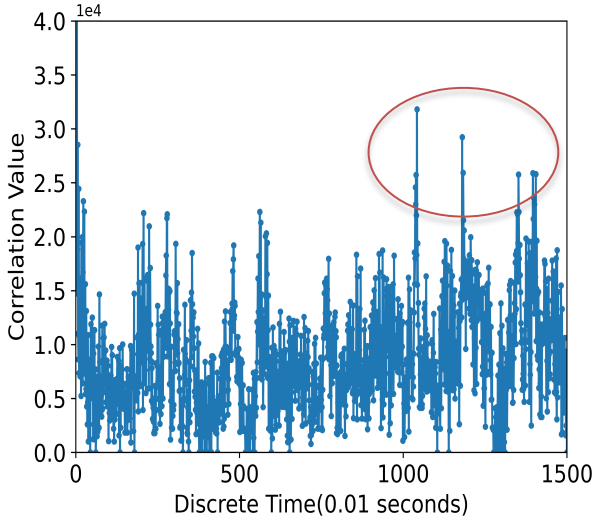

図 4　相関関数による周期分析

4.2 周期性分析

実験では、スニファーが有線 LAN を通じてサーバーに接続する。私たちは 30 秒間の実験を行い、Probe Request フレームだけをスニッフィングする。記録した データから電波強度情報（dBm）を取り出し、ゼロパディングを行う。その後、0〜15 秒のデータをウィンドウとして抽出し、0〜30 秒データとの相関関数を計算する。計算結果を図 4 に示す。ウィンドウの末尾が 30 秒を超えた時に計算を停止するため、横軸（時間軸）の最大値は 15 秒である。関数列の各点の座標を記録し、横軸の値は時間であり、縦軸の値は相関係数である。

この関数列の極大値が Probe Request フレームの周期であると推定する。例えば、図 4 に示しているように、相関係数の大きい順に 10 個の極大値を抽出する場合、それに対応する時間値は以下の通りである：10.42 秒、11.8 秒、11.83 秒、13.96 秒、14.05 秒、13.52 秒、10.39 秒、10.37 秒、13.98 秒、10.38 秒。今回の実機実験で使用されている端末について、Android 端末 Probe Request の周期は約 10 秒であり、iPhone 端末 Probe Request の周期は約 13 秒である。これらの周期は、相関関数の極大値から得られた時間値に一致している。

4.3 フレーム分類法による端末追跡

受信した Probe Request データに対し、フレーム分類手法を使用してグループ分けを行い、結果を図 5 に示す。周期性が検出され推定が成功した端末に対しては、機種ごとにそれぞれ異なる色を付ける。また、周期性が検出されなかった端末については黒色で示す。縦棒横軸の座標は各グループの先頭にあるフレームの受信時間であり、縦軸の座標は 3.1 の数式（式（3）、式（4））で算出したグループ中のフレーム数N_iである。この結果は Probe Request の特徴に

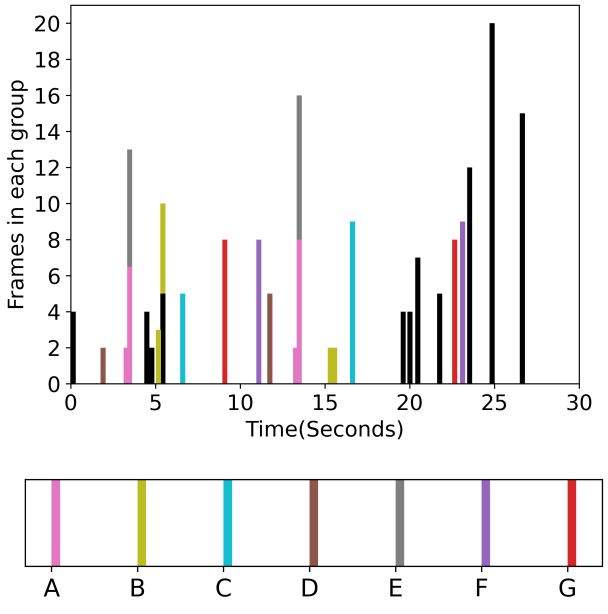

図 5　フレーム分類後のフレーム数

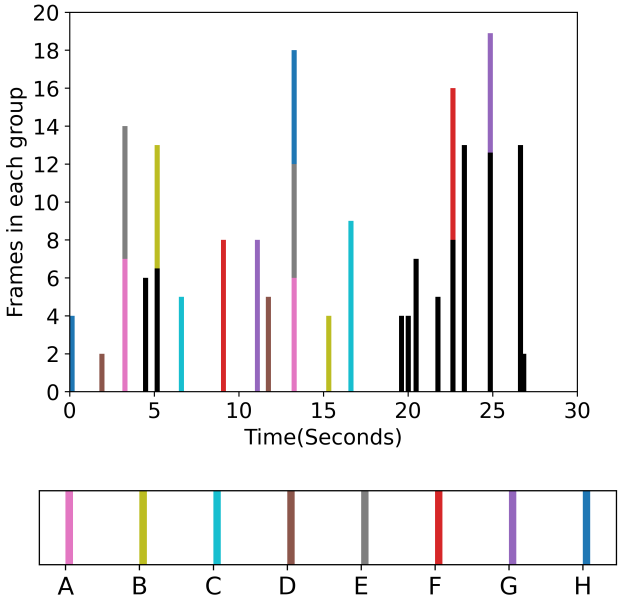

図 6　ステージ分類後のフレーム数

一致する。すなわち、端末は短時間に複数のフレームを送信し、より長い時間が経過してからもう一度フレームを連続的に送信する。実験では、最初に受信フレームの時間を 0 秒にした。つまり、3.1 の式（2）の中は、t_1の値は 0 である。従って、次の縦棒の横軸の座標は t_{N1+1} - t_1 であり、第 Y 番縦棒の横軸の座標は t_y - t_1 である。

図 5 から、0〜15 秒内に初めて検出される端末は 13 台であった。4.2 で得られた関数列の座標を相関係数の大きい順に並べ、先頭から時間値を抽出する。これらの時間値が端末 Probe Request の周期であり、一台の端末は一つの時間値に対応すると推定する。検出過程の中で、これらの時間値付近で実際に 2 周期が検出されたのは次の 7 カ所であった。それぞれの周期から Android/iPhone を推定すると、10.42 秒は端末 A、11.8 秒は端末 F、13.52 秒は端末 G、10.39 秒は端末 B、10.39 秒は端末 B、10.37 秒は端末 C、10.38 秒は端末 D、10.40 秒は端末 E の 7 機種であると推定される。

上記の推定周期情報に基づいて端末追跡を行い、追跡結果と実際の端末情報を比較した結果、フレーム分類手法による端末追跡を行う場合は、追跡の成功率が 70%である。

4.4 ステージ分類法による端末追跡

受信した Probe Request データに対し、ステージ分類手法を使用してグループ分けを行い、結果を図 6 に示す。図 5 と同じように、周期性が検出され推定が成功した端末に対しては、機種ごとにそれぞれ異なる色を付ける。また、周期性が検出されなかった端末については黒色で示す。縦棒横軸の座標は各グループの先頭にあるフレームの受信時間であり、縦軸の座標は 3.2（式（7））の数式で算出したグループ中のフレーム数Nである。この結果も Probe Request の特徴に一致する。4.3 と同じように、最初に受信したフレームの時間を 0 秒に設定した。

図 6 から、0〜15 秒内に初めて検出される端末は 10 台であった。4.2 で得られた関数列の座標を相関係数の大きい順に並べ、先頭から時間値を抽出する。これらの時間値が端末 Probe Request の周期であり、一台の端末は一つの時間値に対応すると推定する。検出過程の中で、これらの時間値付近で実際に 2 周期が検出されたのは次の 8 カ所であった。それぞれの周期から Android/iPhone を推定すると、10.42 秒は端末 A、13.96 秒は端末 F、14.05 秒は端末 G、13.52 秒は端末 H、10.39 秒は端末 B、10.39 秒は端末 B、10.37 秒は端末 C、10.38 秒は端末 D、10.40 秒は端末 E の 8 機種であると推定される。

上記の周期情報に基づいて端末追跡を行い、追跡結果と実際の端末情報を比較した結果、ステージ分類手法による端末推定を行う場合は、追跡の成功率が 80%である。

5. おわりに

本研究では、Wi-Fi アクセスポイントに接続していない移動端末を検出する方法を提案した。本提案では、Probe Request フレームの到着時間間隔および周期性に着目し、端末の個別追跡を行った。

まず、二種類の方法を使用し、スニッフィングした Probe Request フレームのグループ化を行った。第一に、フレーム分類による方法である。隣接フレーム間の到着時間間隔と基準時間を比較し、比較結果により Probe Request フレームを複数のグループに分類した。第二に、ステージ分類による方法である。設定した基準時間およびこの時間の 2 倍、4 倍値に基づき、Probe Request フレームを三つのステージに分類した。その後、各ステージ内のフレーム数により適用するステージを選定し、その中のフレームをグループ化した。また、フレーム数により、各グループの中に含まれる端末の台数も推定した。

次に、周波数分析に基づき、端末対する追跡を行った。0〜15 秒の信号強度データを抽出し、全データとの相関関数を計算した。関数の極大値が Probe Request の周期情報に関連すると推定し、また得られた周期により個別端末を追跡した。

実験の結果、フレーム分類法を使用する場合は、端末追跡の成功率が 70%であり、ステージ分類法を使用する場合は、端末追跡の成功率が 80%である。ステージ分類法の成功率が高い理由は、異なる到着時間間隔の端末に対応できると推測する。また、30 秒間の実験で上記の高い成功率を達成したため、本提案の時間効率も証明された。

本提案手法を使用することにより、ランダム MAC アドレスを使用したり、端末情報を隠蔽したりする機種においても、Probe Request フレームの到着時間間隔および周期のみを用いることで、高精度で Wi-Fi 端末の間接推定ができることを確認できた。

将来的に、体育館や室外など多種多様な環境で実験を行う予定である。また、端末の台数を増やし、更なる実験を行うことも考えている。

参考文献

[1] 村井大地, 浦野健太, 望月祐洋, 米澤 拓郎, 西田純二, 河口信夫, “大規模屋外施設における Wi-Fi パケットセンサへの影響と利活用の検証”, 情報処理学会論文誌デジタルプラクティス, Vol.3, No.2 (April 2022)

[2] Julien Freudiger, “How Talkative is your Mobile Device? An Experimental Study of Wi-Fi Probe Requests”, WiSec’15: 8th ACM Conference on Security & Privacy in Wireless and Mobile Networks, No.8 (2015).

[3] Taylor C. Artunian, “Analysis of Wi-Fi Probe Request Bursts for Device Counting”, Masters Thesis, San Francisco State University, California (2023).

[4] Mathy Vanhoef, Célestin Matte, Mathieu Cunche, Leonardo S. Cardoso and Frank Piessens, “Why MAC Address Randomization is not Enough: An Analysis of Wi-Fi Network Discovery Mechanisms”, Proceedings of the 11th ACM on Asia Conference on Computer and Communications Security (May 2016)

[5] 齋進, 杉浦彰彦, “知的環境認識を用いた災害状況推定における Wi-Fi データ通信中の RTS/CTS 分析”, 情報処理学会全国大会論文集, pp. 41–42 (2023).

[6] Ellis Fenske, Dane Brown, Jeremy Martin, Travis Mayberry, Peter Ryan and Erik Rye, “Three Years Later: A Study of MAC Address Randomization in Mobile Devices and When It Succeeds”, Proceedings on Privacy Enhancing Technologies, Vol.2021, No.3 (2021).

[7] Célestin Matte, Mathieu Cunche, Franck Rousseau and Mathy Vanhoef, “Defeating MAC Address Randomization Through Timing Attacks”, Proceedings of the 9th ACM Conference on Security & Privacy in Wireless and Mobile Networks (July 2016)

[8] Jason Franklin, Damon McCoy, Parisa Tabriz, Vicentiu Neagoe, Jamie Van Randwyk and Douglas Sicker, “Passive Data Link Layer 802.11 Wireless Device Driver Fingerprinting”, Proceedings of the 15th conference on USENIX Security Symposium, Vol.15, No.12 (July 2006)

[9] 岡村健太, 沼尾 雅之, “無線 LAN アクセスポイントを用いた店舗待ち時間予測”, マルチメディア、分散、協調とモバイル (DICOMO2017)シンポジウム (June 2017)