DEIM Forum 2022 J43-4

TEEを備えた軌跡ベースのプライバシ保護型接触追跡システムにおける付加情報の利用

Cao Ruixuan† 加藤 郁之†† 曹 洋†† 吉川 正俊††

† 京都大学工学部情報学科 〒606-8501 京都市左京区吉田本町
†† 京都大学大学院情報学研究科 〒606-8501 京都市左京区吉田本町

あらまし 2019年末より新型コロナウイルス感染症が蔓延しており，感染拡大防止のためにプライバシ保護型接触追跡システムが多く提案されている．しかし，既存のBluetoothベースの接触追跡システムには精度と柔軟性に欠けるという問題が存在している．さらに，新型コロナウイルスだけでなく，今後未知のウイルスが蔓延したとしても感染拡大防止に貢献できるような接触追跡システムが望ましい．そこで本研究は，プライバシ保護をしながら，軌跡データと付加情報を利用して，より柔軟で精度が高く，未知のウイルスにも対応できる接触追跡システムを実現するための拡張手法を提案している．また，実際に提案した拡張手法を実装し，実行時間を測った結果，実用可能であることを示した．

キーワード Private Contact Tracing，新型コロナウイルス感染症，プライバシ保護，Trusted Execution Environment，Intel SGX

1 はじめに

2019年末から流行し始めた新型コロナウイルス感染症は，驚異的な感染力で瞬く間に蔓延し，人々にとって予想していない事態をもたらした．

新型コロナウイルス感染拡大防止のため，一般に保健所では感染者が出た際，該当の感染者に電話で聞き込み調査をし，過去二週間の濃厚接触者に該当する人に関しての情報を得る．この電話での聞き込み調査を全ての感染者に対して1人ずつ行っている．これは非常に膨大な作業量であり，多くの保健所は感染拡大期にはコールセンター状態になっている．もし，ある地域の住民全員の移動軌跡データを取得して，感染者が発覚した場合，住民全員とその陽性者の軌跡データを元に接触を追跡するシステムがあれば，保健所の人が1人1人電話をしなくて済むだろう．だが，市民の軌跡データ[1]というのは機微なパーソナルデータであり，それを扱うにはプライバシ保護を考慮する[2]ことが前提となる．

プライバシ保護型接触追跡システムの既存手法として，信頼できる実行環境のもと，軌跡データから空間的かつ時間的にどのくらい近いかを計算し，時間的近接性，空間的近接性の閾値を満たしていたら接したと判定するPCT-TEE[3]がある．しかしながら，このシステムには2つの不足点がある．(1) 距離が近くても遮りがある場合 (隣り合わせの車とバスなど) は接触していなくても接触したと判定してしまう．(2) 実際は接触したから必ず感染するとは限らず，感染確率の計算をする機能がない．この2点を改善することで，将来現れうる未知のウイルスにより柔軟に対応できるシステムを作ることができる．

PCT-TEEの問題を改善するために，軌跡データを扱うだけでなく，ユーザがマスクをしていたか否か，他者との会話が生じたか否か，密閉空間にいたか否かなどの付加情報も扱うことができれば，より柔軟で精度の良い接触追跡システムができると考えられる．

そこで本研究では，プライバシを保護しつつ，接触したか否かの判定条件を自由に決めることができるPCT-TEEで提案されたシステムを付加情報を使ってどのように拡張できるかを検討する．最終目標としては，将来未知のウイルスが発見されても (例えば，新型コロナウイルスとは違う経路で感染する新型のウイルスが蔓延しても)，プライバシを保護しながら，人々の軌跡データによる接触追跡ができるシステムを提案したい．

この論文における貢献を以下に示す：(1) 軌跡データに付加情報を追加することにより，PCT-TEEを拡張する方法を提案した．(2) 実際に提案した拡張手法を実装して，PCT-TEEの論文で検証に使われていた軌跡データを付加情報付きのものに拡張し，実行時間に関する実験を行った．

本稿の構成は以下の通りである．2章では，プライバシを保護しながら接触を追跡するシステムに関する関連研究について述べる．3章では，本研究における問題設定と提案手法について，4章では本研究にて行った実験について述べ，5章で本論文をまとめる．

2 関連研究

新型コロナウイルス感染症拡大防止のため，Bluetoothを使った接触追跡システム[4] [5] [6] [7] [8] [9]は既に存在しているが，2つの問題がある．1つ目は，レストランなどで感染者が使ったシートを使用することで感染する場合などの間接的な接触を検知することができないことである．2つ目は，直接的な接触の場合でも宅配を受け取るくらいの短時間の接触であれば感染の確率は低いが，Bluetoothベースの接触追跡システムで

は接触したと判定され，感染者との接触時間や距離の判定基準を柔軟に変更できないことである．

本章では，プライバシを保護しながら接触を追跡するシステムに関する関連研究としてと信頼できる実行環境のもとで軌跡データを利用したプライバシ保護型接触追跡システム PCT-TEE [3] について述べる．

2.1 PCT-TEE

PCT-TEE は既存の Bluetooth ベースの接触追跡システムの問題を解決し，セキュアなハードウェア Intel SGX を使用することで構築可能な，信頼できる実行環境 TEE [10] のもとで，ユーザの軌跡データを利用して接触を検知する，軌跡データベースの Private Contact Tracing (以下 PCT) という手法を使い，より安全で，効率の良い接触追跡システムを提案した．

PCT-TEE の接触追跡システムでは，医療機関が感染が確認された患者の軌跡データ (このデータは暗号化されているか，患者の同意の下で公開されている) をクライアントから管理者が信頼されていないサーバに登録する．サーバは，クライアントからの問い合わせとして暗号化されたクライアントの軌跡データを受け取り，サーバ内の軌跡データとクライアントの軌跡の交点を秘密裏に計算することで，感染リスクのある接触があったかどうかを示すブール値を返す．

保健所の問題にこれを適用すると，保健所が住民の暗号化された軌跡データをサーバに登録し，感染者が発覚した場合に，感染者の軌跡データを入力として問い合わせをすると，保健所は住民の軌跡データの生データを知ることなく濃厚接触者を追跡することができる．

3 付加情報を利用したプライバシ保護型接触追跡手法

本章では，まず接触追跡タスクの問題設定について述べ，次に，具体的に考えうる付加情報やシステムの拡張方針について説明を行う．

3.1 問題設定

まず，本研究において，時刻データと位置データの対をポイントレコードと呼び，一つ以上のポイントレコードから成る時系列データを軌跡データと呼ぶ．

次に，本研究において想定してるシステムは，サーバ側には大量の住民の軌跡データが保存されており，感染者が発覚した場合，サーバ側に保存されているユーザ u となる住民一人一人の軌跡データ X_u と，ユーザ v となる感染者の軌跡データ X_v を地理的時間的空間にマッピングし，交わりを計算することで，接触を検知する．ただ，接触の際の環境やユーザの状態によって感染の確率が変わるので，付加情報 a を用いてより正確な感染確率 p を計算できるようにする．例を挙げて言うと，これまでの手法では，ユーザ u と感染者 v が空間的近接性と時間的近接性を満たせば，この 2 人は接触したと判定していた．しかし，感染者 v が新型コロナウイルス感染症ワクチンを既に二回接種していて (ワクチンを接種していれば人に移す可能性は低いと言われている [11])，かつユーザ u はマスクをしており 2 人の間には会話がなかったというケースでは，ユーザ u の感染確率 p は極めて低いはずだ．従って，この場合は低い数値で接触確率を出力することでより正確な接触追跡ができるだろう．本研究では，このようなケースをはじめとした，あらゆるケースを付加情報を使って正確に判定できるようにシステムを拡張したい．

$$\begin{cases} 1\left(\exists x_i^{(u)} \in X_u, x_j^{(v)} \in X_v \ \text{ s.t.} \right. \\ \quad \left. \left\| l_i^{(u)} - l_j^{(v)} \right\| \leqq \Theta_{\text{geo}} \ \text{ and } \ \left\| t_i^{(u)} - t_j^{(v)} \right\| \leqq \Theta_{\text{time}} \right) \\ 0(\ \text{others}\ ) \end{cases} \tag{1}$$

上記の式 (1) の変数の定義は以下の通りである．

- $l_i^{(u)}$: ユーザ u の i 番目の軌跡データの位置 (緯度，経度)
- $t_i^{(u)}$: ユーザ u の i 番目の軌跡データの時間
- Θ_{geo} : 空間的近接性の閾値
- Θ_{time} : 時間的近接性の閾値
- $X_i = \left(x_1^{(i)}, \ldots, x_n^{(i)}\right)$: X_iはユーザ i の軌跡データ集合，$x_n^{(i)}$はユーザ i の n 番目の軌跡データ

まず，基本的な時空間近接性の判定について式 (1) で説明する．例えば，ユーザ m の軌跡データを

$$X_m = \begin{pmatrix} x_1^{(m)} = (2022/1/11\ 13:30:00, 35.67, -65.34), \\ x_2^{(m)} = (2022/1/11\ 13:31:00, 35.34, -65.69), \\ x_3^{(m)} = (2022/1/11\ 13:32:00, 35.67, -65.34) \end{pmatrix}$$

とし，ユーザ n の軌跡データを

$$X_n = \begin{pmatrix} x_1^{(n)} = (2022/1/11\ 13:27:00, 22.67, -77.67), \\ x_2^{(n)} = (2022/1/11\ 13:28:00, 25.21, -72.43), \\ x_3^{(n)} = (2022/1/11\ 13:29:00, 28.88, -67.22) \end{pmatrix}$$

として，$\Theta_{\text{geo}} = 10$，$\Theta_{\text{time}} = 3$ とすると，$x_2^{(m)}$ と $x_3^{(n)}$ などが式 (1) の上段の条件を満たしており，ユーザ m と n は時間的近接性と空間的近接性を満たしていると判定される．

一方で，上記の判定方法は厳密すぎて効率が悪いので，PCT-TEE では軌跡データを xyz 軸をそれぞれ経度，時間，経度とした三次元の地理的時間的空間 $\mathbb{A}$ にマッピングし，計算効率を高めている．図 1 を使って説明すると，ユーザ m の軌跡データを $X_m = (x_1, x_2, x_3)$，ユーザ n の軌跡データを $X_n = (x_4, x_5)$ とすると，x_1 は部分空間 A_1 に含まれており，x_2, x_3, x_4, x_5 は部分空間 A_2 に含まれている．これにより，軌跡データ X_m は部分空間 A_1, A_2 にマッピングされ，X_n は A_2 にマッピングされる．両者は部分空間 A_2 で時間的近接性と空間的近接性を満たしているので，接触したと判定される．

上記を踏まえて，PCT-TEE で定式化された式をベースに，軌跡データに付加情報を加えて問題を定式化すると，以下のような式になる．

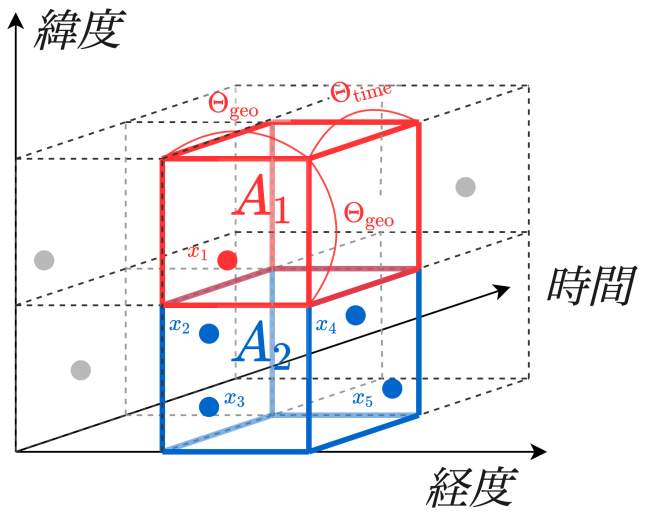

図 1　地理的時間的空間における接触判定 (この図は PCT-TEE [3] の図 2 を参考に作成した)

$$\begin{cases} p\left(\exists \mathbf{A}^{(u)} \cap \mathbf{A}^{(v)} \neq \emptyset \ \text{ s.t.} \right. \\ \quad \mathbf{A}^{(u)} = \left\{ f_\Theta\left(x_i^{(u)}\right) \mid x_i^{(u)} \in X_u \right\} \ \text{and} \\ \quad \left. \mathbf{A}^{(v)} = \left\{ f_\Theta\left(x_j^{(v)}\right) \mid x_j^{(v)} \in X_v \right\} \right) \\ 0 \ \text{(others)} \end{cases} \tag{2}$$

上記の式 (2) の変数の定義は以下の通りである．

- p : クライアントがウイルスに感染した確率
- $a_i^{(u)} = \left(a_{i_1}^{(u)}, a_{i_2}^{(u)}, \ldots, a_{i_n}^{(u)}\right)$: ユーザ u の i 番目の軌跡データの付加情報，添字 1，2，...，n は付加情報の種類を表す
- $A_i \in \mathbb{A}$: A_i は i 番目の部分空間,$\mathbb{A}$ は全部分空間の集合
- $f_\Theta : x \to A$: 軌跡データ x を部分空間 A にマッピングする関数

これらを用いて以下のように入力と出力を設定する．

入力

- 付加情報を含んだユーザ i の軌跡データ

$$X_i = \left(x_1^{(i)} = \left(t_1^{(i)}, l_1^{(i)}, a_1^{(i)}\right), \ldots, x_n^{(i)} = \left(t_n^{(i)}, l_n^{(i)}, a_n^{(i)}\right)\right)$$

- 空間的近接性の閾値 Θ_{geo}
- 時間的近接性の閾値 Θ_{time}

出力

- ユーザの感染確率 p

このように接触追跡問題を定式化し，正確な感染確率を出力することを目標とする．

表 1　システム拡張の分類

拡張の種類	付加情報の具体例
暴露時間に関する拡張	付加情報を使わない
感染確率に関する拡張	年齢，性別，過去の感染歴，基礎疾患の有無，ワクチンの接種回数など
	マスク着用の有無など
接触判定に影響する拡張	自家用車の中にいた，バスにいた，教室にいたなど

3.2　システムの拡張手法

まず，想定するシステムの構造について説明する．システムは軌跡データ，空間的近接性の閾値，時間的近接性の閾値をクエリとして受け取り，回答として感染確率を返す．回答を構成するアルゴリズムは，接触判定パートと感染確率計算パートの 2 部分に分かれている．

接触判定パートでは，軌跡データの位置情報と時間情報が空間的近接性と時間的近接性を満足しているか否かで接触判定を行う．接触判定の結果，接触したと判定された場合，付加情報を使って感染確率計算パートで感染確率を計算し，それを出力する．接触していない場合は，0 という確率を出力する．つまり，システムの拡張を考える際，接触判定パートと感染確率計算パートを独立として考えても問題はない．上記の内容を踏まえた上で，より多くの種類のクエリに対応するためには，以下のような 3 通りの拡張が考えられる．

暴露時間に関する拡張

接触判定パートで，軌跡データから接触の暴露時間を計算し，空間的近接性，時間的近接性だけでなく，暴露時間の閾値も満たしていたら接触と判定するように拡張する．この拡張により，暴露時間も考慮したより柔軟な接触判定条件を指定できる．

感染確率に影響を与える付加情報に関する拡張

この拡張では，感染確率計算パートで付加情報を扱い，システムが感染確率を計算できるようにする．付加情報は，(1) ユーザ自身にのみ関係する付加情報と (2) ユーザ自身だけでなく，他者の感染確率にも影響を与えうる付加情報の 2 種類あり，これによって感染確率計算に使う関数の種類が異なる．

接触判定に影響を与える付加情報に関する拡張

この拡張では，システムの接触判定結果に影響を与えうる付加情報を扱い，接触判定の精度を高める．

拡張の分類と対応するクエリの種類を表にまとめると，表 1 のようになる．

また，各々の拡張に関して，具体的にどのような付加情報を使うか，詳しい拡張の方針について以下に示す．

3.2.1　暴露時間に関する拡張

この拡張は PCT-TEE でも説明されており，式 (2) を少し拡張し，以下の式 (3) のようにすると実現できる．

$$
\begin{cases}
max(p_i)\ \Big(\exists\tau \text{ s.t.} \quad \forall i = (\tau, \tau+1, \ldots, \tau+\Theta_{\text{doe}}\ ) \\
\quad (\exists \mathbf{A}^{(u)} \cap \mathbf{A}^{(v)} \neq \emptyset \ \text{ s.t.} \\
\quad \mathbf{A}^{(u)} = \left\{ f_\Theta\left(x_i^{(u)}\right) \mid x_i^{(u)} \in X_u \right\} \text{ and} \\
\quad \mathbf{A}^{(v)} = \left\{ f_\Theta\left(x_j^{(v)}\right) \mid x_j^{(v)} \in X_v \right\}\Big)\Big) \\
0(\text{ others }\ )
\end{cases}
\tag{3}
$$

Θ_{doe} は暴露時間 (duration of exposure) の閾値を表し，入力と出力は以下のようになる．

入力

- 付加情報を含んだユーザ i の軌跡データ

$$X_i = \left(x_1^{(i)} = \left(t_1^{(i)}, l_1^{(i)}, a_1^{(i)}\right), \ldots, x_n^{(i)} = \left(t_n^{(i)}, l_n^{(i)}, a_n^{(i)}\right)\right)$$

- 空間的近接性の閾値 Θ_{geo}
- 時間的近接性の閾値 Θ_{time}
- 暴露時間の閾値 Θ_{doe}

出力

- ユーザの感染確率 p

ここで，暴露期間は接触したと判定されたケースの数をユーザごとに連続してカウントしている．出力の感染確率は，全ての接触判定されたケースにおいて，つまり接触とカウントされている軌跡データの一点ごとに感染確率を計算し，最大の数値を最終的な感染確率とする．例えば，1 分ごとに軌跡データをとり，暴露期間の閾値が 3 だったとすると，5 個目から 10 個目の軌跡データで接触したと判定されていたら，5 個目から 10 個目の軌跡データを一個ずつ感染確率を計算し，最大値をこのユーザの感染確率とする．これを上記の式 (3) に当てはめると，$\Theta_{\text{doe}} = 3$ で，$\tau = 5$， $i = (5,6,7,8)$ (または $\tau = 6$, $i = (6,7,8,9)$ か $\tau = 7$，$i = (7,8,9,10)$) になるので，p_5，p_6，p_7，p_8，p_9，p_{10} の中の最大値 $max(p_i)$ が感染確率となる．

3. 2. 2 感染確率に影響を与える付加情報に関する拡張

この拡張では，まず PCT-TEE を使って接触判定を行い，その結果接触したと判定された場合，付加情報を引数とした関数によって確率を計算する．

a） ユーザ自身にのみ関係する付加情報

ここで扱う付加情報は，年齢，性別，過去の感染したことがあるか否か，基礎疾患の有無，ワクチンの接種回数など自分の属性に当たるものだ．ここで感染確率計算に使う関数は，各々のユーザの付加情報により決まり，他のユーザの付加情報に影響されない．

具体的には，ユーザ m の各々の属性を引数として，ウイルスに感染する確率を関数

$$f(a^{(m)}_{i_{\text{年齢}}}, a^{(m)}_{i_{\text{性別}}}, a^{(m)}_{i_{\text{過去の感染歴}}}, a^{(m)}_{i_{\text{基礎疾患の有無}}}, a^{(m)}_{i_{\text{ワクチンの接種回数}}}, \ldots)$$

で表し，感染者であるユーザ n がウイルスを感染させる確率を関数

$$g(a^{(n)}_{i_{\text{年齢}}}, a^{(n)}_{i_{\text{性別}}}, a^{(n)}_{i_{\text{過去の感染歴}}}, a^{(n)}_{i_{\text{基礎疾患の有無}}}, a^{(n)}_{i_{\text{ワクチンの接種回数}}}, \ldots)$$

表 2　仮定となる感染確率

年齢	感染確率
10 代未満	0.7
10 代	0.5
20 代	0.3
30 代	0.8
40 代とそれ以上	0.6

過去の感染歴	感染確率
有	0.1
無	0.9

基礎疾患	感染確率
有	0.7
無	0.3

性別	感染確率
女性	0.4
男性	0.6

とすると，ユーザ m とユーザ n が接触して，実際にユーザ m が感染する確率は，

$$p = f(a_i^{(m)}) \cdot g(a_i^{(n)})$$

となる．

より具体的な例で述べる．これらの付加情報は互いに独立しているもので，関数 f と g は各々の確率を掛け合わせることで導出できると仮定すると，以下のように式が得られる．

$$p = p_{\text{年齢}} \times p_{\text{性別}} \times p_{\text{過去の感染歴}} \times p_{\text{基礎疾患}} \times \ldots$$

ユーザ m が持つ付加情報を以下のように定義し，

$$
\begin{aligned}
a_i^{(m)} \quad = (\quad & a_{i_1}^{(m)} = (\text{年齢})20, a_{i_2}^{(m)} = (\text{性別})\ \text{女}, \\
& a_{i_3}^{(m)} = (\text{過去の感染歴})\ \text{有}, a_{i_4}^{(m)} = (\text{基礎疾患})\ \text{無} \ldots),
\end{aligned}
$$

ユーザ n がもつ付加情報を以下のように定義すると，

$$
\begin{aligned}
a_i^{(n)} \quad = (\quad & a_{i_1}^{(n)} = (\text{年齢})40, a_{i_2}^{(n)} = (\text{性別})\ \text{女}, \\
& a_{i_3}^{(n)} = (\text{過去の感染歴})\ \text{無}, a_{i_4}^{(n)} = (\text{基礎疾患})\ \text{有} \ldots)
\end{aligned}
$$

となる．

付加情報別の感染確率を表 2 のようにすると [1]，ユーザ m の感染確率は，

$$
\begin{aligned}
p &= f \cdot g \\
&= p_{\text{年齢}=20} \times p_{\text{性別}=\text{女}} \times \ldots \times p_{\text{過去の感染歴}=\text{無}} \times p_{\text{基礎疾患}=\text{有}} \\
&= 0.3 \times 0.4 \times 0.1 \times 0.3 \times 0.6 \times 0.4 \times 0.9 \times 0.7 \\
&= 0.36\%
\end{aligned}
$$

となる．

b） 他者の感染確率にも影響を与えうる付加情報

これに該当する付加情報は，例えばマスク着用の有無である．新型コロナウイルス感染症対策では，「マスクは相手のウイルス吸入量を減少させる効果より，自分からのウイルス拡散を防ぐ効果がより高く，仮に 50 センチの近距離に近づかざるを得なかった場合でも，相手だけがマスクを着用するより，自分だけがマスクを着用する方が，より効果が高い」[12]．一方，未知のウイルスでは，マスクの効果は相手だけが着用する方が効果が高いかもしれない．未知のウイルスも対応できるようなシステ

1：ここで使っている確率の値は医学的根拠のない架空の数値である．

ムを開発するには，このような想定もする必要がある．この場合，感染確率計算に使う関数は自分の付加情報だけでなく相手の付加情報も引数としたものでなければならない．

具体的には，ユーザ m とユーザ n が接触したとすると，ユーザ m が感染する確率は，

$$p = h(a^{(m)}_{i_{\text{マスク着用の有無}}}, \ldots, a^{(n)}_{i_{\text{マスク着用の有無}}}, \ldots)$$

となる．ここで，関数 h の定義を以下のように仮定する．

$$h_i = \begin{cases} 0.05, & a^{(m)}_{i_{\text{マスク着用の有無}}} = \text{有 and } a^{(n)}_{i_{\text{マスク着用の有無}}} = \text{有} \\ 0.25, & a^{(m)}_{i_{\text{マスク着用の有無}}} = \text{無 and } a^{(n)}_{i_{\text{マスク着用の有無}}} = \text{有} \\ 0.3, & a^{(m)}_{i_{\text{マスク着用の有無}}} = \text{有 and } a^{(n)}_{i_{\text{マスク着用の有無}}} = \text{無} \\ 0.4, & a^{(m)}_{i_{\text{マスク着用の有無}}} = \text{無 and } a^{(n)}_{i_{\text{マスク着用の有無}}} = \text{無} \end{cases}$$

上記の仮定のもと，ユーザ m がマスクをしており $(a^{(m)}_i = \Big(\ldots, a^{(m)}_i =$ (マスク着用の有無) 有,$\ldots$)$)$，ユーザ n もマスクをしている $(a^{(n)}_i = (\ldots,$ $a^{(n)}_i =$ (マスク着用の有無) 有,$\ldots$$\Big)$$)$ 場合，ユーザ m の感染確率は，

$$p = h(a^{(m)}_i, a^{(n)}_i) = 0.05$$

である．

3.2.3 接触判定に影響を与える付加情報に関する拡張

ここで扱う付加情報は，自家用車の中にいた，バスにいた，教室にいたなどの密閉空間に関連するものである．例えば，現在のシステムでは，自家用車を運転していて，バスの隣に停車した際，位置情報が近い距離にあるので接触したとされるが，実際は車とバスによる遮りがあるので，接触していないと判定しなければならない．

上記のシステムを実現するために，付加情報を利用した拡張としては，車やバス，教室に番号をつけ，同じ番号の空間にいた人同士のみで，接触判定を行うなどが考えられる．具体的には，番号 1 の車を運転しているユーザ A とその車の隣の番号 2 のバスに乗っているユーザ B とユーザ C がいたとして，密閉空間の番号を 3 つ目の付加情報とすると，ユーザ A の軌跡データ X_A には付加情報として車の番号 1 が入っており，

$$x^{(A)}_n = \Big(t^{(A)}_n, l^{(A)}_n = (77, 88), a^{(A)}_n = (a^{(A)}_{n_1} = \ldots, a^{(A)}_{n_3} = 1, \ldots)\Big)$$

となり，ユーザ B の軌跡データ X_B は付加情報としてバスの番号 2 が入っており，

$$x^{(B)}_n = \Big(t^{(B)}_n, l^{(B)}_n = (77, 88), a^{(B)}_n = (a^{(B)}_{n_1} = \ldots, a^{(B)}_{n_3} = 2, \ldots)\Big)$$

となり，ユーザ C の軌跡データ X_C は，

$$x^{(C)}_n = \Big(t^{(C)}_n, l^{(C)}_n = (77, 88), a^{(C)}_n = (a^{(C)}_{n_1} = \ldots, a^{(C)}_{n_3} = 2, \ldots)\Big)$$

となり，このとき，軌跡データの位置情報 l は 3 人とも同じになっているが，付加情報の値をみると，ユーザ A はユーザ B やユーザ C とは別の番号になっているので，システムは付加情報により，ユーザ A がユーザ B やユーザ C とは接触していないと判定する．また，ユーザ B とユーザ C は付加情報の値が同じつまり同じバスに乗っているため，与えられた閾値を満足しているか否かによって接触判定を行う．

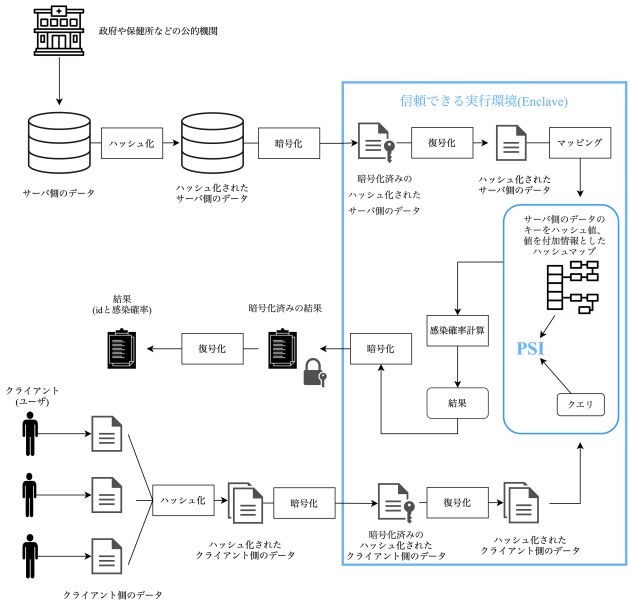

図 2　実装したシステムのアーキテクチャ(この図は PCT-TEE [3] の図 3 を参考に作成した)

4　実　　験

本章では，第 3 章で提案した拡張手法を実際に実装し，その実装をもとに実験を行う．今回の実験の目的は，軌跡データに付加情報を追加したことによって，PCT-TEE と比べて，どのくらい実行時間が増加するのか，またその実行時間は実用上許容範囲であるかを検証することだ．

4.1　実装について

PCT-TEE の論文で実装されていたシステム [2] をもとに，第 3 章で述べた付加情報の追加を必要とする拡張である感染確率に影響を与える付加情報に関する拡張と接触判定に影響を与える付加情報に関する拡張を実装 [3] した．ハッシュ化された軌跡データに，年齢，過去の感染歴の有無，ワクチンの接種回数，マスク着用の有無，密閉空間の番号，合計 5 種類の付加情報を追加したデータに対して接触判定を行い，接触したと判定された場合，ユーザの id とそのユーザの最大の感染確率 [4] を出力するように拡張した．また，PCT-TEE の実装では，軌跡データの格納に使うデータ構造は Hashtable の他に，FST というハッシュ化された軌跡データを高速に扱えるデータ構造も使用可能であったが，今回の実装では，簡単のため，データの格納には Hashtable を用いた．

実装したシステムのアーキテクチャを図 2 に示す．ここでは，

2 : https://github.com/ylab-public/PCT

3 : https://github.com/Hisokalalala/PCT

4 : ここで最大としている理由は，接触が複数回あった場合，各々の感染確率のうち最大のものを出力するためである．

表 3　データセットのサイズ (この表は PCT-TEE [3] の表 8 を参考に作成した)

	クライアント側のデータ	サーバ側のデータ
ニューヨーク	20160 × 100	20160 × 1000
近畿	1440 × 100	1440 × 14000
東京	1440 × 100	1440 × 14000

サーバ側に感染者のデータが格納され，クライアントが住民である場合を想定している [5]．サーバ側のデータは政府や保健所などの公的機関から感染者の軌跡データと付加情報を入手し，それをハッシュ化したのち，共通鍵を使って暗号化し，信頼できる実行環境 (Enclave) へ送る．信頼できる実行環境の中でサーバ側のデータを復号化し，キーを軌跡データのハッシュ値，値を付加情報としたハッシュマップを使ってサーバ側のデータをマッピングする．クライアント側のデータも同様に，ハッシュ化したしたのちに共通鍵で暗号化し，これを信頼できる実行環境に送る．信頼できる実行環境の中で復号化し，データをクエリとして保持し，先ほどハッシュマップに格納したサーバ側のデータと Private Set Intersect (PSI) による比較をする．比較の結果，軌跡データのハッシュ値が一致した場合，その時点での両者の付加情報を使って感染確率を計算し，それを結果に格納し，結果を共通鍵で暗号化する．暗号化された結果を信頼できる実行環境から外部 [6]に送り，復号化すれば，接触があると判定されたクライアントの id と感染確率が出力される．

4.2　実験内容

まず，実験に使うデータセットの軌跡データは，PCT-TEE の論文で使われていた軌跡データと同じものを使う．具体的には，scikit-mobility [7]に実装された密度 EPR モデル [13] によって生成される合成データセットと東京大学空間情報科学研究センターの共同研究利用システム [8]で入手可能な，日本の特定の地域における人々の軌跡のデータを使った実データセット [9]である．合成データセットは，ニューヨーク市における 14 日間の 1 分ごとの個人の軌跡データであり，実データセットは，日本の近畿 [10]と東京 [11]の人の流れデータセットを使って作成してある．

データセットのサイズを表 3 に示す．具体的に述べると，ニューヨークのクライアント側のクエリデータはクライアント

5：PCT-TEE がそのような前提で実装されていたので，こちらでもその前提で図を書いた．逆の場合でも，つまり，サーバ側に住民のデータを入れ，クライアントを感染者のデータを使うことは可能であり，データを変更すれば良いだけである．

6：ここでいう外部は信頼できる実行環境の外という意味であり，システムの内部である．

7：https://github.com/scikit-mobility/scikit-mobility

8：https://www.csis.u-tokyo.ac.jp

9：正確には，これらの人の流れのデータセットは，実際の軌跡から精巧に作られた合成データセットであるが，本研究ではこれらのデータセットを実際のデータセットとみなす．データ作成の具体的なプロセスの詳細についてはこちらのサイトに記載されている http://pflow.csis.u-tokyo.ac.jp/data-service/pflow-data/.

10：https://joras.csis.u-tokyo.ac.jp/dataset/show/id/3038201000

11：https://joras.csis.u-tokyo.ac.jp/dataset/show/id/3000200800

```
positive result queryIds: [2, 3, 4, 5, 6, 7, 8, 9]
```

図 3　PCT-TEE の結果

```
positive result queryIds: [(2, 0.0038400006), (3, 0.0038400006)
, (4, 0.0038400006), (5, 0.0038400006), (6, 0.0038400006), (7,
0.0038400006), (8, 0.0038400006), (9, 0.0038400006)]
```

図 4　本研究のシステムの結果

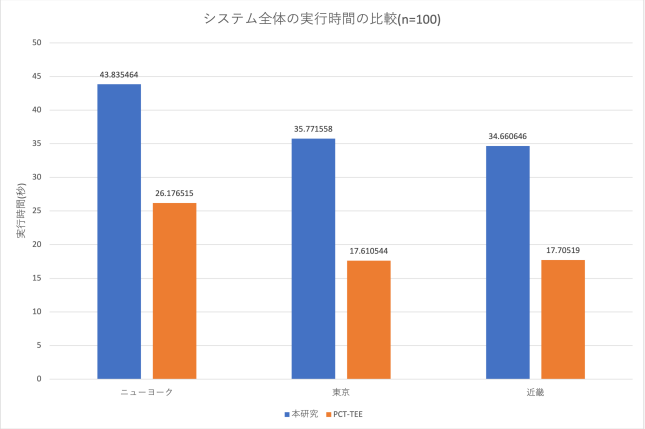

図 5　クライアントサイズ=100 での両手法の実行時間

が 100 人おり，この 100 人のクライアントを 2 週間にわたって 1 分ごとに軌跡データを記録した場合，クライアント側のクエリデータには，

$$14_{(日)} \times 24_{(時間)} \times 60_{(分)} = 20160$$

より，それぞれのクライアントが長さ 20160 個の軌跡データを持っている．従って，この 100 人のクライアントは合計 2016000 個のポイントレコードがある．

これらのデータセットを使い，本研究の実装でサーバ側のデータを格納する際のデータ構造に Hashtable を用いて確率の計算を可能にした実装と PCT-TEE の実装でデータ構造に FST を使った際の実行時間を比較する．

4.3　実験結果

まず，PCT-TEE と本研究で実装したシステムの接触判定の結果 [12]を図 3 と図 4 に示す．本研究のシステムの結果は，PCT-TEE で接触したと判定されたクライアントの id と一致しており，その id ごとに感染確率を出力している [13]．

図 5 では，表 3 で述べたデータセットを使って実行した際の実行時間を比較した結果を示す．この図から，PCT-TEE では FST をデータ構造に使っているため，Hashtable を使っている本研究より PCT-TEE の方が実行速度は速いことが分かる．しかし，その差は 2 倍程度であり，付加情報を追加したデータで感染確率を計算する実行時間としては十分許容範囲であると思われる．

また，図 6 と図 7 でより大きいクライアントサイズを使って

12：ここでは，クライアントサイズが 10 の時の結果を示している．

13：実験用のデータを作成する際付加情報を全て同じ値にしたので，ここでの感染確率は全て同じ値が出力されている．

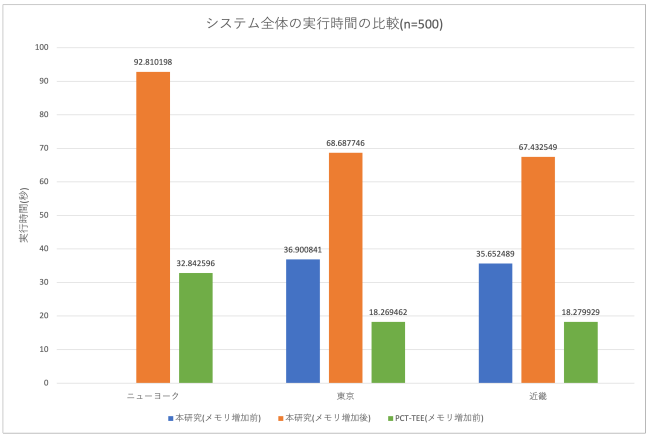

図 6　クライアントサイズ=500 での実行時間 (本研究のメモリ増加前と増加後を含む)

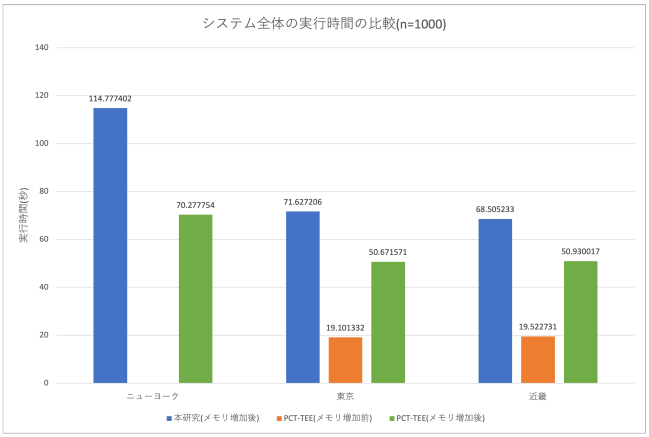

図 7　クライアントサイズ=1000 での実行時間 (PCT-TEE のメモリ増加前と増加後を含む)

実行した時の両手法の比較結果を示す [14]．これらの図で使われているメモリという用語について説明する．Enclave を起動する際，config ファイルに書かれている設定をもとに初期化する．初期化した際のヒープ領域のメモリサイズを超えた場合，私が実行した限りだとメモリ割り当てエラーが発生し，実行が失敗する．これに対処するために，config ファイルでヒープ領域のサイズを増やす必要があるが，Enclave を起動する際の初期化に時間がかかるようになる．つまり，クライアントサイズをより大きくした場合，メモリ増加前では実行に必要なメモリを確保することができず，実行が失敗してしまうため，config ファイルの設定をすることにより使えるメモリ領域を増やして，大きいクライアントサイズでも実行可能とした．そのトレードオフとして，Enclave を起動する際，メモリ増加前より時間がかかるようになる [15]．

図 6 の本研究の手法 (メモリ増加前) のグラフと PCT-TEE(メモリ増加前) のグラフに注目すると，先ほどと同じように，本研究の手法 (メモリ増加前) は PCT-TEE(メモリ増加前) より

14：ここで使ったデータセットは，サーバ側のデータは表 3 と同じであるが，クライアント側のデータはクライアントのサイズが異なる．

15：ただし，Enclave の初期化にかかる時間はどのシステムでもほぼ等しい

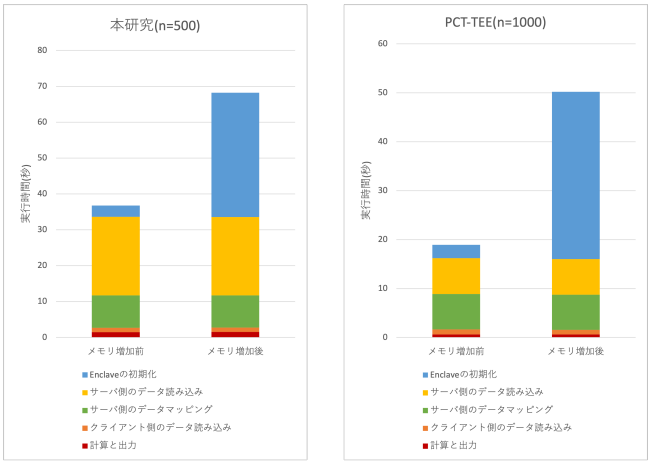

図 8　フェーズごとの実行時間

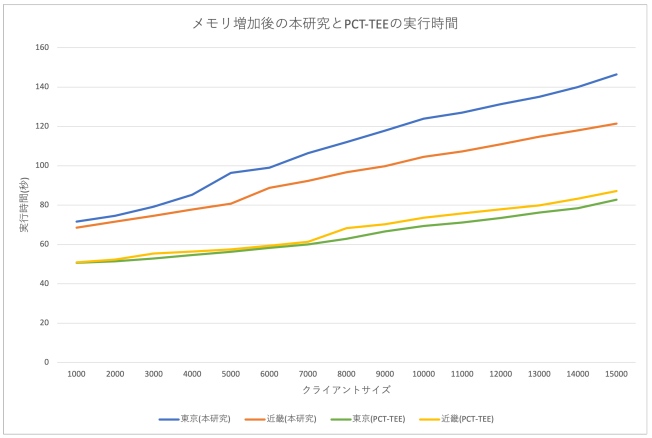

図 9　手法ごとのクライアントサイズと実行時間の関係

実行時間が長いが，二倍程度の差である [16]．また，本研究の手法のメモリ増加前とメモリ増加後のグラフからわかるようにメモリを増加した場合は，Enclave を起動する際の初期化に時間がかかり，実行時間がメモリ増加前より圧倒的に増えていることが分かる．

同様に，図 7 の本研究の手法 (メモリ増加後) のグラフと PCT-TEE(メモリ増加後) のグラフに注目すると，両方メモリ増加後でもやはり実行時間は PCT-TEE の方が短いが，その差は高々二倍であり，許容範囲である．また，PCT-TEE のメモリ増加前のグラフ [17] とメモリ増加後のグラフを比較してみると，やはりメモリ増加後の実行時間が圧倒的に長いことより，Enclave の初期化に時間がかかっていることがわかる．

ここで，本研究の手法と PCT-TEE のメモリ増加前とメモリ増加後のフェーズごとの実行時間を図 8 に示す．図 8 より，メモリ増加前とメモリ増加後の実行時間の差分はほぼ Enclave の初期化の部分だけであり，実際に接触判定をする計算処理の実行時間は許容範囲の速度で実行している．

図 9 に，手法ごとの全体の実行時間をクライアントサイズご

16：ニューヨークのデータを本研究の手法で実行するにはメモリが足りず，メモリ増加前の実行時間のデータは存在しない

17：ニューヨークのデータをメモリ増加前の PCT-TEE で実行するにはメモリが足りず，メモリ増加前の実行時間のデータは存在しない

とに示している．図からわかるように，クライアントサイズが大きくなるにつれ，実行時間はほぼ線形的な増加をしている．このことより，クライアントサイズが非常に大きくなっても，ほぼ線形的な増加なので，許容範囲の実行速度で実行可能であると思われる．

つまり，軌跡データに付加情報を追加しても，プライバシ保護型接触追跡システムとして実用化可能である．

5 まとめ

本研究では，プライバシ保護をしながら，軌跡データを利用した接触追跡システムをベースに，付加情報を利用して，より柔軟で精度が高く，未知のウイルスにも対応できる接触追跡システムを実現するための拡張手法を提案している．そして，実際に軌跡データに付加情報を追加したデータを使って接触判定を行い，接触したクライアントの感染確率も計算し，それを出力するシステムを実装した．実験では，このシステムの実行時間を測り，それが PCT-TEE の実行時間とどれくらいの差があるかを調べ，実用上では許容範囲内であることを検証した．今後の課題については以下にまとめる．

(1) 付加情報の最大許容サイズと最適サイズを探す．本研究の実験では，一つの付加情報が 1 バイトで，追加した付加情報は 5 個なので，合計 5 バイト分の付加情報を追加した軌跡データを使った．ここで，付加情報が大きくなった場合，Intel SGX のリソースがメモリ制約の関係でページングに時間がかかり，実行時間が非常に長くなる恐れがあるので，許容できる最大の付加情報のサイズと，最適なサイズを探索する余地がある．

(2) 他のデータ構造を使った実装を試す．本研究では，簡単のためサーバ側のデータを Hashtable を使って格納したが，FST を使ってデータを格納することによって，ハッシュ化された軌跡データをより高速に扱うことができ，実行時間を縮めることが可能だと思うわれる．また，年代や性別などの個人の属性を表す付加情報は変化がないため，軌跡データの中で，全ての行にこれらの属性データを持つ必要はなく，データ構造を工夫して作れば，より高速なシステムになる可能性がある．

最後に，本研究がプライバシ保護型接触追跡システムの中でも未知のウイルスにも対応できるようなあらゆる判定条件を設定できる高精度なシステムとして，新型コロナウイルスだけでなく今後の疫病対策として貢献できることを期待する．

謝　辞

本研究は，JST CREST JPMJCR21M2，JST SICORP JPMJSC2107， 科学研究費 21K19767， 19K20269， KDDI 財団調査研究助成の支援を受けたものである．

文　献

[1] HUMAN RIGHTS WATCH. Mobile location data and covid-19: Q&A, 2020. https://www.hrw.org/news/2020/05/13/mobile-location-data-and-covid-19-qa.

[2] Hyunghoon Cho, Daphne Ippolito, and Yun William Yu. Contact tracing mobile apps for covid-19: Privacy considerations and related trade-offs. *arXiv preprint arXiv:2003.11511*, 2020.

[3] Fumiyuki Kato, Yang Cao, and Masatoshi Yoshikawa. PCT-TEE: Trajectory-based Private Contact Tracing System with Trusted Execution Environment. *ACM Transactions on Spatial Algorithms and Systems (TSAS)*, Vol. 8, No. 2, pp. 1–35, 2022.

[4] Yanan Da, Ritesh Ahuja, Li Xiong, and Cyrus Shahabi. React: Real-time contact tracing and risk monitoring via privacy-enhanced mobile tracking. In *2021 IEEE 37th International Conference on Data Engineering (ICDE)*, pp. 2729–2732. IEEE, 2021.

[5] Johannes K Becker, David Li, and David Starobinski. Tracking anonymized bluetooth devices. *Proc. Priv. Enhancing Technol.*, Vol. 2019, No. 3, pp. 50–65, 2019.

[6] Yaron Gvili. Security analysis of the covid-19 contact tracing specifications by apple inc. and google inc. *IACR Cryptol. ePrint Arch.*, Vol. 2020, p. 428, 2020.

[7] Ronald L Rivest, Jon Callas, Ran Canetti, Kevin Esvelt, Daniel Kahn Gillmor, Yael Tauman Kalai, Anna Lysyanskaya, Adam Norige, Ramesh Raskar, Adi Shamir, et al. The pact protocol specification. *Private Automated Contact Tracing Team, MIT, Cambridge, MA, USA, Tech. Rep. 0.1*, 2020.

[8] Ni Trieu, Kareem Shehata, Prateek Saxena, Reza Shokri, and Dawn Song. Epione: Lightweight contact tracing with strong privacy. *arXiv preprint arXiv:2004.13293*, 2020.

[9] Carmela Troncoso, Mathias Payer, Jean-Pierre Hubaux, Marcel Salathé, James Larus, Edouard Bugnion, Wouter Lueks, Theresa Stadler, Apostolos Pyrgelis, Daniele Antonioli, et al. Decentralized privacy-preserving proximity tracing. *arXiv preprint arXiv:2005.12273*, 2020.

[10] Mohamed Sabt, Mohammed Achemlal, and Abdelmadjid Bouabdallah. Trusted execution environment: what it is, and what it is not. In *2015 IEEE Trustcom/BigDataSE/ISPA*, Vol. 1, pp. 57–64. IEEE, 2015.

[11] 厚生労働省. 感染症専門医が解説！ 分かってきたワクチンの効果と副反応, 2021. https://www.cov19-vaccine.mhlw.go.jp/qa/column/0001.html.

[12] 厚生労働省. 新型コロナウイルスに関する Q&A（一般の方向け）, 2021. https://www.mhlw.go.jp/stf/seisakunitsuite/bunya/kenkou_iryou/dengue_fever_qa_00001.html#Q4-1.

[13] Luca Pappalardo, Filippo Simini, Salvatore Rinzivillo, Dino Pedreschi, Fosca Giannotti, and Albert-László Barabási. Returners and explorers dichotomy in human mobility. *Nature communications*, Vol. 6, No. 1, pp. 1–8, 2015.