

Claude: ロケーションベース P2P アプリケーション開発フレームワーク

川部 勝也 福田 浩章 三好 匠

近年、車の相乗りや AR ゲームなど、地理情報を利用した様々なサービスやアプリケーションが開発されている。それらサービスの多くは近隣のノードと通信するために P2P 通信を用いている。P2P アプリケーションを実行するデバイスは、セキュリティや IPv4 アドレスの枯渇問題などの理由から、NAT を介したプライベートネットワークに接続している。そのため、インターネットを介した P2P を実現する場合、STUN や TURN といった通信プロトコルを利用し、異なるプライベートネットワークに接続しているピア同士を接続する必要がある。また、地理情報の管理やピアの発見も行う必要がある。P2P アプリケーション開発者は、アプリケーション本来のロジックに加え、P2P を実現するこれらの技術を理解し、自身で実装する必要がある。

そこで本論文では、地理情報を利用した P2P アプリケーションを開発するために必要な通信、地理情報管理、ピアの探索をアプリケーションから分離するフレームワーク、Claude の提案と実装を行う。Claude ではプライベートネットワークごとにプロキシサーバを用意し、既存のネットワークトポロジ、ネットワーク機器に大幅な変更を施すことなく地理情報を利用した P2P アプリケーション開発を容易にする。開発者は、Claude が提供する API を使用することにより、プロキシサーバの存在や、通信するピアが異なるプライベートネットワークに存在することを意識することなく P2P アプリケーションを開発することができる。そして、Claude を利用して異なるプライベートネットワークに接続したピア同士のスループットを測定し、同地点でグローバル IP を保持するピア同士のスループットと比較することで、Claude の性能を示す。

Recently, many applications using geolocation information are developing, for example, Uber and Grab. These applications use P2P to communicate with the neighbor nodes. The devices executing P2P application connect to the Internet via private-network to avoid exhausting the addressing space of IPv4. In this case, a peer in private-network needs to use some protocols such as STUN and TURN to directory communicate with other peers in different private-network. Additionally, it is necessary to do managing geolocation information and looking up other peers. As a result, the application developer must comprehend and implement P2P communication logic on their application.

Claude is a location-based framework that separates the implementation of P2P communication. That is including managing geolocation information, looking up other peers, and communicating with them. Claude provides the proxy server for each private-network. The proxy server plays managing geolocation information, looking up peers, and communication with other peers without large modifying the network devices and network topology. And the application developer uses the client library provided by Claude to connect proxy from an application on the peer. Therefore, the proxy server and the client library helps to isolate those from the application developer.

In this paper, we propose the location-based P2P framework: Claude. And it indicates that the network throughput between peers each in private-network is similar to network throughput between the same peers which have global IP.

1 はじめに

過去 10 年間でスマートフォンを始めとしたポータブルなコンピュータは大きく普及した。スマートフォンの利用者数は 2014 年に 1.57 億人で、2020 年には 2.87 億人にも昇ると見られている [1]。スマートフォ

ンの機能の主要な一つとして GPS がある。GPS は特別なデータのやり取りを必要とせず、衛星からの信号を受信することで自身の地理情報を取得することができる技術である。GPS を用いたサービスとして Uber が存在する。Uber はライドシェアリングサービスと呼ばれる Uber の提供するアプリを通じて、車を手配して欲しい利用者と時間の空いている運転手をマッチングさせるサービスである。この時、利用者は自身がどこで待っているかを運転手に伝えるために GPS を用いてなるべく正確な地理情報を伝えることができる。また、Tinder というマッチングサービスでは、利用者の GPS を用いて数 km 圏内に存在する他利用者と優先的にマッチング可能となる仕組みも存在する。上記のサービスは利用者同士の通信に一部 P2P 通信を用いている [2]。P2P 通信は従来のクライアント/サーバ間通信のように他利用者と通信を行う場合はサーバを介さずに利用者同士が直接通信を行う。利用者のノード (ピア) 同士が直接通信を行うにはお互いの IP アドレスとポート番号を指定する必要があるが、セキュリティや IPv4 アドレスの枯渇問題などの理由から全てのピアが一意的な IP アドレスを割り振られているわけではなく、NAT を介したプライベートネットワークに接続している。そのため、インターネットを介した P2P 通信を行う場合は STUN [3] や TURN [4] といった通信プロトコルを利用することで、異なるプライベートネットワークに接続しているピア同士の接続を確立する必要がある。地理情報を用いたサービスを実現するには、これらのプロトコルの理解やピアの発見、管理、アプリケーションロジックへの組み込みを行わなくてはならない。

本論文では、地理情報を利用した P2P アプリケーションを開発するために必要な通信、地理情報管理、ピアの探索をアプリケーションから分離するフレームワーク、Claude の提案と実装を行う。Claude ではプライベートネットワークごとにパケットの中継、ピアの探索を担うプロキシサーバを用意し、既存のネットワークトポロジ、ネットワーク機器に大幅な変更を加えることなく、地理情報を用いた P2P アプリケーションの開発を容易にする。アプリケーション開発者は Claude が提供する API やクライアントライブラ

リを用いることでプロキシサーバとの通信、ピアの探索ロジックなどを意識することなくアプリケーションの実装が可能となる。2 節でロケーションベースの関連研究との差異を示し、3 節で Claude の基本的な構成要素と役割について触れる。また、具体的なピアの探索や地理情報の管理、通信のアルゴリズムについても同節で言及する。最後に 4 節で Claude を利用してプライベートネットワークに接続したピア同士のスループットと同地点でグローバル IP を保持するピア同士のスループットを比較し、Claude の性能について検証する。

2 関連研究

本節では Claude と同様にロケーションベースな P2P プロトコルについての関連研究について述べる。

2.1 DGT

DGT [5] は位置情報に基づいてネットワーク構築を行うピア P2P な手法として提案された。DGT では、ネットワークに参加したピアの情報を自身からの距離に応じてグループ分けした Geo-Buckets (GB) と呼ばれるリストを用いる。ピアがネットワークに参加する時に、自身の位置情報と共に参加要求を全てのピアの情報管理するブートストラップノードに送信する。ブートストラップノードは要求のあったピアの近くに存在するピアのリストを返信することでネットワークへの参加を実現している。Claude では DGT の Geo-Buckets と類似した手法を用いているがそれをプロキシが管理することでピアへの負担を軽減している。

2.2 Globase.KOM

Globase.KOM [6] はスーパーノード型の P2P ネットワークを位置情報に基づいて構築する手法である。Globase.KOM では物理的な領域を格子状に分割し、各格子領域内に存在するピアから CPU 性能が高く、ネットワーク状況の良いピアを選択する。選択されたピアはスーパーノードとして同一領域内に存在するピアからの探索を中継する。これによって各領域間のピアの探索、通信を実現している。しかし、スー

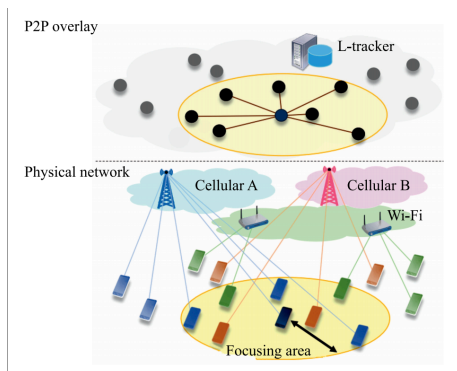


図 1 G-LocON によるオーバーレイネットワーク

パーノードはネットワークに参加する一般的なピアから選出されるため、ピアが不意にダウンする可能性や移動により、通信遅延の増加が懸念される。Claude におけるプロキシは Globase.KOM のスーパーノードと役割が類似しているが、プロキシはピアから選出されることなく、プロキシ専用のマシンを予めアプリケーションを運用者が適したものをプライベートネットワークに配置することで、ピアへの負担を減らす。

2.3 G-LocON

G-LocON は図 2.3 に示すとおり、LTE や Wi-Fi を用いたロケーションベースの P2P オーバーレイネットワークを構築するためのフレームワークである [7]。例えば、ユーザの 1km 圏内にいるピアと通信したい場合は G-LocON を用いることでサーバを介さずにピア間の P2P 通信を実現可能となる。ピアはプライベートネットワークに接続している場合、自身のグローバルな IP とポート番号を知らない。そのため、まずピアは自身の情報をグローバルに観測する STUN サーバにアクセスすることでピア自身のグローバル IP とポート番号を確認する。次に、ピアはシグナリングサーバにアクセスすることで一意な ID を付与される。ピアは受け取った ID を L-Tracker というネットワークに参加しているピアの地理情報を管理するサーバに自身の経度、緯度と共に送信することでグローバルな P2P ネットワークに参加する。ピア同士で通信を行う時にはピアは必ずシグナリングサーバに 1km 圏内のピアのリストのようなクエリを送信する。シグ

ナリングサーバはピア間の通信を実現するためにピアの IP、ポート番号と位置情報に一意な ID を紐付けて管理するサーバである。次に、シグナリングサーバは条件にマッチしたピアの情報を返信する。そして、アプリケーションは G-LocON の提供するクライアントライブラリを介して、受信したリストに記載されているグローバル IP とポート番号を元に通信を開始する。

G-LocON はフレームワークとして実装されているが、Android 専用のフレームワークとなっている。また、ピア検索のクエリも対応しているのは近隣のピアをリストアップすることのみのため、ユーザが任意の地域のピアとコネクションを確立することはできない。

3 Claude

本節では Claude の概要と機能を説明する。Claude ではピアの代わりにプライベートネットワークに予め用意したプロキシがピアの探索や他のピアとの通信を行う。これによってピアに提供するライブラリとユーザーのアプリケーションはプロキシに地理情報の管理、検索、通信を一任することができる。加えて、ピアはグローバルなネットワークに接続することなくハイブリッドな P2P 通信を実現することができる。以下にプロキシを介してピアが異なるプライベートネットワークのピアと通信を行うまでの手順を説明する。

3.1 情報の登録

ピアが異なるプライベートネットワークのピアと通信を行うには現在の IP とポート番号をベースにしたネットワーク通信の仕様上、相手のピアやプロキシの IP、ポート番号や検索を行うための地理情報を取得する必要がある。そのためにプロキシ、ピアは通信を行う前にお互いの情報が取得可能な場所に自身の情報を登録しなくてはならない。まずプロキシの情報の登録手順について述べる。

3.1.1 プロキシの登録手順

プロキシは起動後に GPS や外部サービスを用いて自身の経度、緯度 (地理情報) を取得する。その後、外部のタブレットサーバに地理情報を付与して登録のリ

クエストを送信する。タブレットサーバは予めプロキシが存在を認知しているプロキシの情報を保存するグローバルなサーバである。リクエストを受け取ったタブレットサーバは送信元の IP とポート番号、付与された地理情報に加え、一意な名前を生成し保存する。この時タブレットサーバに保存される IP とポート番号はプライベートネットワークの IP とポート番号ではなく、グローバルネットワークに接続する時に経由する NAT で変換された後の IP とポート番号である。これを他のプロキシが取得することによってプロキシ間の通信が可能となる。そして、タブレットサーバは情報を保存後、生成した名前をプロキシに返信する。生成された名前は実際に通信を行う際に宛先、送信元のプロキシを判別するために用いられる。以上でプロキシの情報登録が完了となる。

3.1.2 ピアの登録手順

3.1.1 節の手順によってプロキシはタブレットサーバに自身の情報を登録した。その後にプロキシはピアからの情報の登録を受付可能となる。ピアはプロキシに情報を登録するために最初にプライベートネットワークに存在するプロキシをブロードキャストを用いて検索する。ブロードキャストを受け取ったプロキシはピアへ自身の IP とポート番号を返信する。ピアはその IP とポート番号に対して自身の地理情報を付与して登録リクエストを送信する。プロキシはピアに一意な名前を割り振り、地理情報と送信元の IP とポート番号を保存し、ピアに割り振った名前を返信する。ピアは割り振られた名前を保存し、登録は完了となる。割り振られた名前はプロキシと同様に通信の際にデータがどのピアから送られてきたかを判別するために用いる。以降、登録を行ったピアはプロキシを介してピアの検索、通信を行うことができる。

3.2 ピアの検索

ピアの検索はまず、検索元のピアが経度、緯度、距離 (D_{req}) を指定し、検索リクエストをプロキシへ送信する。プロキシは受け取ったリクエストをタブレットサーバへ中継する。タブレットサーバはリクエストに含まれる経度と緯度とタブレットサーバが保持しているプロキシの経度と緯度の 2 点間の距離 D をヒュベ

ニの公式 (1) を用いて計算する。タブレットサーバが保持しているプロキシの情報それぞれに対して D を求めた後に $D_{req} > D$ を満たしたプロキシの IP とポート番号のリストを検索元のプロキシに返信する。リストを受け取ったプロキシはリストに含まれるプロキシに対して保持しているピアのリストをリクエストする。リクエストを受け取った他のプロキシは自身が保持しているピアの名前のリストとプロキシ自身の名前を返信する。ピアのリストを受け取った検索元のプロキシはその結果をピアへ中継する。これによってピアは指定した経度、緯度からの一定距離以内に属するプロキシが保持するピアのリストを取得した。以降、検索元のピアは受け取ったリストに含まれるピアの名前とそのピアが登録されているプロキシの名前を元に通信を開始する。

$$D = \sqrt{(Dy * M)^2 + (Dx * N * \cos P)^2}$$

Dy : 2 点の緯度の差

Dx : 2 点の経度の差

P : 2 点の緯度の平均

$$M = \frac{Rx(1 - E^2)}{W^3} : \text{子午線曲率半径}$$

$$W = \sqrt{1 - E^2 * \sin^2 P}$$

$$N = \frac{Rx}{W} : \text{卯酉線曲率半径}$$

$$E = \sqrt{\frac{Rx^2 - Ry^2}{Rx^2}} : \text{離心率}$$

$Rx = 6378137.000$: 赤道半径

$Ry = 6356752.314$: 極半径

(1)

3.3 ピア間の通信

3.1, 3.2 節の手順によって、ピアは経度、緯度、距離を指定することで通信したいピアの情報を取得可能となった。次に、ピアが取得したリストに含まれるピアと通信を中継する時の手順を示す。

3.3.1 通信の疎通確認

通信を行うピアとプロキシの例を図 2 に示す。プロキシとピアの組が 2 つ存在し、それぞれプライベートネットワークに属し、IP とポート番号が割り振られている。プロキシとピアにはそれぞれ一意な名前が割り振られており、図中では Proxy-A, Peer-A, Proxy-B, Peer-B としている。プライベートネットワークのプ

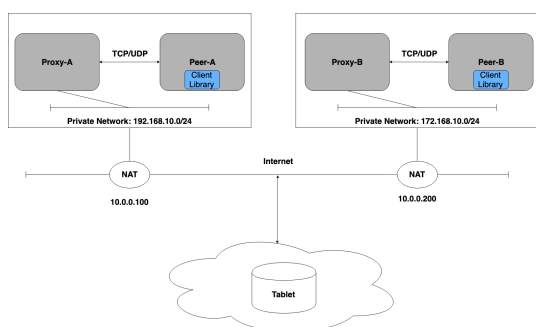


図 2 ピアとプロキシの構成例

ロキシがプライベートネットワーク外のピアやプロキシ、タブレットと通信するには NAT を経由して通信する。また、本節では通信の開始を Peer-A から行い、それぞれのプロキシを経由して宛先の Peer-B に対してデータを中継する。まず、Peer-A は 3.2 節での手順に則りピアを検索した結果、Proxy-B の名前と Peer-B の名前を取得した。次に Peer-A は Proxy-A に対し、Proxy-B に属する Peer-B と通信を行う旨のリクエストを送信する。Proxy-A は Proxy-B の名前を鍵としてタブレットサーバに問い合わせる。タブレットサーバは Proxy-B の IP とポート番号番号を返信する。そして、Proxy-B に対して Peer-B と通信を確立するリクエストを送信する。同時に Proxy-B の IP とポート番号を保存しておく。Proxy-B はリクエストを受け取り、自身が管理しているピアのリストから Peer-B を鍵として対応する IP とポート番号を取得する。この時、先程と同様に Proxy-A の名前と IP とポート番号の情報を保存しておく。Proxy-B は取得した Peer-B の IP とポート番号に通信可能かを問い合わせる。Peer-B は通信可能な状態であればその旨を Proxy-B に返信する。Peer-B から返信を受け取った Proxy-B は Proxy-A へ通信可能な旨を返信する。最後に Proxy-A が Peer-A へ Proxy-B, Peer-B が通信可能な状態であることが伝わることで Peer-A は通信を開始することができる。

3.3.2 通信の中継

Peer-A は Peer-B が通信可能な状態であることを確認した後に実際に Proxy-A, Proxy-B を経由して Peer-B へのデータの送信を開始する。実際にデータ

を送信する際にはネットワークの仕組みとして送信するデータに宛先となる IP とポート番号を指定する必要がある。しかし、現状の仕組みでは 1 つのデータに対して IP とポート番号は 1 つしか指定できない。その場合、Peer-A が Peer-B の IP とポート番号を指定した場合、NAT は宛先がプライベートネットワークのマシンだと思い、データは存在しない IP とポート番号に転送される。同様に Proxy-A の IP とポート番号を指定した場合でも、Proxy-A に対してデータは届くが、その後 Proxy-B に対してデータを転送するための情報が欠如している。正しくデータを中継し、別のプライベートネットワークのピアに転送するために Claude では送信するデータに以下の情報を付与する。

- 送信元のピアの名前
- 送信元のピアが属するプロキシの名前
- 宛先のピアの名前
- 宛先のピアが属するプロキシの名前

Peer-A は上記の情報が付与されたデータを Proxy-A に送信する。Proxy-A は受け取ったデータから宛先のプロキシの名前を元に先程保存した Proxy-B の IP とポート番号に対してデータを中継する。もし、この時 Proxy-B の情報が存在しなければ名前を元にタブレットサーバに問い合わせ Proxy-B の IP とポート番号を取得し解決する。Proxy-B は受け取ったデータから宛先のピアの名前を取得する。Peer-B の名前を鍵に自身が管理しているピアのリストから対応する IP とポート番号を取得する。そして、Proxy-B は取得した IP とポート番号に対してデータを送信することで Peer-B にデータが届く。

以上の手順によってピアの検索、通信の中継をプロキシを介して行えることを示した。

4 スループットの評価

Claude の性能評価のために、プロキシを経由したピア間の通信とそうでない場合のスループットを検証した。2 つのピアを準備し、一方のピアからもう一方のピアへ 1GB のデータを送信した場合にかかった時間を評価する。検証には Google Cloud Platform を用いて仮想マシンを東京と大阪、オレゴンに用意し、東京-大阪間と東京-オレゴン間のスループットをそれ

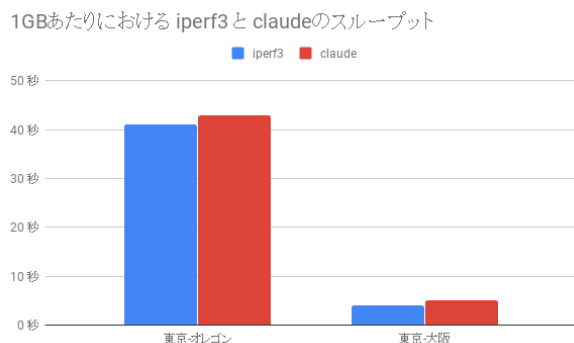


図 3 評価: 各リージョン間におけるスループット

ぞれの手法で計測した。ピア間のプロキシを経由しない場合のデータの転送には一般的なスループット検証ツールである iperf3 を用いた。その結果を図 3 に示す。

図にある通り、プロキシを経由した場合とそうでない場合の転送時間にはほぼ差がないことがわかる。これはヘッダをプロキシで検証するオーバーヘッドはあるが、同一プライベートネットワーク内でのデータ転送におけるスループットが 2.0Gbps[8] と非常に高速なことから、iperf3 によるリージョン間の通信コストとほぼ差がなかったと考える。

5 まとめと今後の課題

プライベートネットワーク内に存在する様々なプラットフォームのピアが内部、外部問わずに通信可能となるフレームワークを提案、実装した。また、ピア

間のパケットのスループットを検証し、提案手法でのオーバーヘッドが許容範囲内であることを示した。しかし、常に近隣のピアと検索、通信を行うアプリケーションを実装する場合はタブレットサーバに検索クエリが集中してしまう。これを解決するためにタブレットサーバ、プロキシが保持するデータ構造と手順を改善し、検索にかかる負荷の分散を図らなくてはならない。

参考文献

- [1] Statista, "Number of smartphone users worldwide from 2014 to 2020", 2018 [Online]. Available: <https://www.statista.com/statistics/330695/number-of-smartphone-users-worldwide>
- [2] investopedia "Peer-to-Peer (P2P) Service" <https://www.investopedia.com/terms/p/peertopeer-p2p-service.asp>
- [3] Rosenberg, Jonathan, et al. "STUN-simple traversal of user datagram protocol (UDP) through network address translators (NATs)." (2003).
- [4] <https://tools.ietf.org/html/rfc5766>
- [5] Picone, Marco, Michele Amoretti, and Francesco Zanichelli. "Proactive neighbor localization based on distributed geographic table." International Journal of Pervasive Computing and Communications 7.3 (2011): 240-263.
- [6] Kova, Aleksandra, Nicolas Liebau, and Ralf Steinmetz. "Globase. kom-a p2p overlay for fully retrievable location-based search." Seventh IEEE International Conference on Peer-to-Peer Computing (P2P 2007). IEEE, 2007.
- [7] Miyoshi Takumi, Shimomura Yusuke, Fourmaux Olivier. (2019). A P2P-based Communication Framework for Geo-Location Oriented Networks. Journal of Telecommunications and Information Technology. 1. 58-66. 10.26636/jtit.2019.131119.
- [8] <https://www.apps-gcp.com/gce-performance-survey/>