

地域需要に基づく LEO 衛星のキャッシュ制御法

栗田 悠翔[†] 上山 憲昭[†]

[†] 立命館大学 情報理工学部

〒567-8570 大阪府茨木市岩倉町 2-150

E-mail: †is0668xh@ed.ritsumei.ac.jp, ††kamiaki@fc.ritsumei.ac.jp

あらまし 地上インフラを展開できない地域へネットワークを接続するための方法として LEO (Low Earth Orbit) 衛星の活用が有効である。しかし、LEO 衛星には上りのスループットが低いという欠点がある。これは地上に配置されたサーバから LEO 衛星経由で大容量コンテンツを配信する際のボトルネックとなる。そのため LEO 衛星にキャッシュサーバを設置し、LEO 衛星からキャッシュ配信を行うことが有効である。従来の地上キャッシュではキャッシュサーバが不動であり、コンテンツの人気の空間的局所性を活用し、キャッシュヒット率向上が可能である。しかし、LEO 衛星は常に高速で移動しているため、コンテンツ人気の空間的局所性を活用できずにキャッシュ性能が著しく低下すると予想される。そこで本稿では軌道上を周回している各衛星に対して地表の異なる地域を割り当て、各衛星は自身が割り当てられた地域からのみキャッシュコンテンツを置換するキャッシュ制御方式を提案する。そして計算機シミュレーションを行い、キャッシュヒット率および遅延減少量の向上を確認する。

キーワード 低軌道衛星, キャッシュ, 空間的局所性, 地域需要

Caching Method Based on Region Demand in LEO Satellite Networks

Yuto KUWATA[†] and Noriaki KAMIYAMA[†]

[†] College of Information Science and Engineering, Ritsumeikan University

2-150, Iwakura-cho, Ibaraki, Osaka 567-8570, Japan

E-mail: †is0668xh@ed.ritsumei.ac.jp, ††kamiaki@fc.ritsumei.ac.jp

Abstract Low Earth orbit (LEO) satellite networks are an effective solution for providing connectivity to regions where terrestrial infrastructure is difficult to deploy. However, LEO systems often suffer from limited uplink capacity, which can become a bottleneck when delivering bandwidth-intensive content from ground servers via satellites. To address this issue, deploying caches on LEO satellites and serving content directly from orbit is promising. In terrestrial caching, fixed cache locations can exploit the spatial locality of content demand to improve the cache hit ratio; in contrast, the high mobility of LEO satellites makes such locality harder to leverage, potentially degrading caching performance. In this paper, we propose a cache control scheme that assigns distinct ground regions to individual satellites and restricts cache replacement to requests originating from the assigned region. Simulation results demonstrate that the proposed scheme improves both the cache hit ratio and delay reduction compared with baseline approaches.

Key words LEO Satellite, Cache, Spatial locality, Region demand

1. 研究背景

現在インターネットは生活に欠かせないインフラとなり、それに伴ってネットワークトラフィック需要も増大している。一方で過疎地や海上といった地域ではネットワークインフラを展開しにくく、その地域にいるユーザの需要に十分にこたえることが出来ていない。そこで、従来地上に展開されるネットワークを代替・補完する方法として LEO 衛星ネットワークが挙げられる。LEO 衛星ネットワークは高度 2,000km 以下の低軌道を周回する衛星を中継ノードとして用いたネットワークである。高度 36,000km 前後に位置する静止軌道衛星や 2,000km から 36,000km 程度の中軌道衛星に比べて低い位置を周回しているため、停電力・低遅延・大容量の通信が可能である。近年、日本国内でも LEO 衛星ネットワークを用いた商用通信サービスが本格的に展開されている。2022 年には Starlink [1] が一般向けインターネット接続サービスを日本で開始し、山間部や離島におけるブロードバンドアクセス手段として利用

が広がりつつある。さらに 2025 年 4 月、KDDI は Starlink の Direct to Cell 技術を用いた au Starlink Direct を開始し、国内で初めてスマートフォンが単体で LEO 衛星に直接接続できる商用サービスを実現した [2]。このようにコンテンツ配信にも活用される LEO 衛星だが、アップリンクのスループットが低いという課題がある。図 1(a) に示すように地上に存在するサーバから大容量のコンテンツを地上の端末に配信する際に、アップリンクのスループットがボトルネックとなり、ユーザが要求したコンテンツを配信できない・遅延が大きくなるという問題が起こる。そこで、図 1(b) に示すように LEO 衛星にキャッシュサーバを設置し、LEO 衛星からキャッシュ配信を行うことが有効である。キャッシングとは、ユーザがアクセスした情報を一時的にキャッシュサーバに保存し、次回以降にキャッシュされたコンテンツに素早くアクセスできるようにする技術である。LEO 衛星ネットワークのキャッシュ配信において空間的局所性はキャッシュ性能に大きく影響を及ぼす [3]。しかし、LEO 衛星は常に高速で移動しているため、

時間が経過するとその衛星でキャッシュしているコンテンツが全く要求されない可能性が発生する。

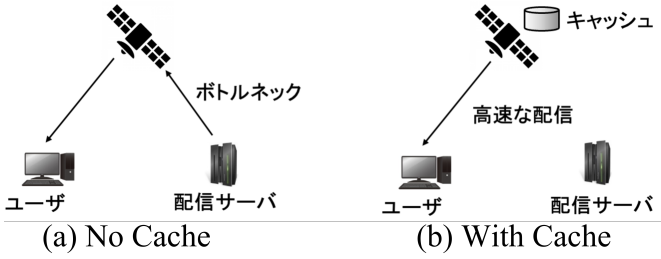


図1 LEO衛星からのキャッシュ配信

2. 関連研究

2.1 固定カバレッジ

Wuらは、衛星と地上ネットワークを統合した環境での効率的なコンテンツ配信のための二層キャッシュモデルを提案している[5]。この研究では衛星が地上ネットワークの補完的な役割を果たすことを前提として、キャッシュコンテンツの管理の最適化をすることで、トラフィック量の低減を実現している。Liらは、統合型衛星/地上無線アクセスネットワークにおけるエネルギー効率とトラフィックオフロードの最適化について議論している[6]。これらの研究では静止軌道衛星や中高度衛星といった静的な衛星を前提としてモデルとしており、LEO衛星の移動する特性を考慮した設計にはなっていない。そのため、LEO衛星に動的なトポロジ変化を反映したキャッシュ管理手法に適用することが困難である。

2.2 衛星間通信

Chaudhryらは、LEO衛星コンステレーションにおけるレーザーISLについて分析を行っている[7]。この研究はStarlinkに焦点を当てており、衛星間通信における接続方式やその性能に着目している。衛星間通信の接続方式は静的接続方式と動的接続方式の2つに分類される。静的接続方式では接続相手が固定されており、トポロジが一定に保たれるため、トポロジ管理が容易である一方、LEO衛星の高速移動や頻繁なトポロジ変化に対応しきれないという課題が存在する。動的接続方式では衛星の移動に応じて接続衛星が動的に変更される。そのため、LEO衛星の動的性に対応することが可能となるが頻繁な接続衛星の変更による制御の複雑化が問題点として挙げられる。また、衛星間通信で接続する衛星間の距離をパラメータとして変更することで、各衛星が接続する台数を変化させてシミュレーションを実行するが、衛星間距離が長くなるほど接続が不安定になる。Zhaoらは、ISLを活用した協調キャッシングを前提として、キャッシュ配置と配信遅延を同時に最適化する方式を提案している[8]。LEO衛星ネットワークにおける協調ビデオキャッシングとしてCACVCを提案し、衛星カバレッジやコンテンツ人気・限られたキャッシュ資源を考慮しつつ、要求動画をアクセス衛星・隣接衛星・地上局のいずれから提供するかに伴う遅延を最小化する問題をPOMDPとして定式化した。さらに、MADDPGを用いたマルチエージェント強化学習により分散実行可能なキャッシュ方を導出し、遅延削減及びヒット率向上を示している。

2.3 要求コンテンツの偏りと地域特性

Tangらは、衛星-地上統合ネットワーク(STIN:satellite-terrestrial integrated network)におけるキャッシングにおいてコンテンツ人気度を事前既知とする点と中央制御で更新する前提が現実的ではないという問題点にアプローチしている[9]。そこでユーザー要求から地域特徴を推定・更新することで地域別の人気度推定を行っている。さらにridge回帰で地域特徴を推定し、類似度により隣接地域を協調エリアへ分割して協調キャッシングを行う手法を提案している。このような地域差を推定してキャッシュに反映する発想は有効だが、地域需要をどのデータからどの粒度で構成するかについては設計余地がある。Zhangらは、要求の時間変化とコンテンツ種別を考慮

してキャッシュ割合を制御するCA-propを提案している[10]。これにより、サービスパス上の協調キャッシュと衛星移動方向を考慮した事前キャッシュにより遅延削減を図った。しかし、地域ごとの需要の偏りを考慮できておらず、実データから作成することが主軸とはなっていない。LEO衛星ネットワークの動的性において時間変化・多様性を考慮することは重要であり、需要推定を固定の仮定から時間変化へ寄せる必要がある。

3. キャッシュ方式の概要

3.1 コンテンツのバリエーション

提案方式では衛星をグルーピングし、グループごとにキャッシュするコンテンツを分散させる。この方式により、任意の場所からの要求が発生した際にその要求を満たせる異なるグループの衛星が存在することで、コンテンツのバリエーションが高まり、キャッシュヒット率が向上する。その結果、オリジンサーバへアクセスする回数が減少し、遅延の削減にも起因する。図2では衛星をグルーピングしていない場合(a)とグルーピングした場合(b)のコンテンツのバリエーションを例示している。グルーピングしていない場合は全ての衛星に類似したコンテンツがキャッシュされることになり、十分に需要の局所性に対応できていないとも言えない。一方で、グルーピングした場合はグループごとに地域を割り当て各エリアの高需要コンテンツを主にキャッシュすると考えられるため、需要の局所性に対応しやすくなる。提案方式ではユーザーがコンテンツを要求した際には通信可能範囲内で同じグループの衛星が優先してユーザーと通信を行うこととする。ユーザーと通信する衛星が要求コンテンツを取得していなかった場合、衛星間通信を利用してオリジンサーバからコンテンツを取得する。また、経路上に同じグループの衛星が存在した場合は、その衛星もコンテンツを取得する。

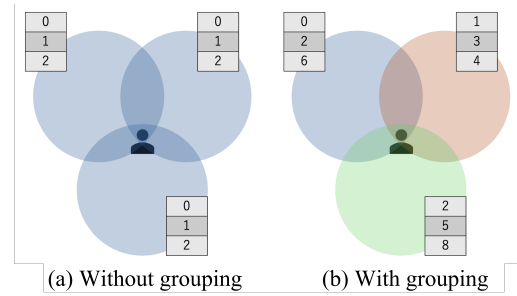


図2 グルーピングの有無

3.2 従来のキャッシュ方式の課題

図3が示すように、従来方式では全ての六角形グリッドから一様にコンテンツをキャッシュする。しかし、LEO衛星ネットワークでは衛星が常に高速で移動しているため、地上キャッシュのように空間的局所性を活用したキャッシュが困難である。そのため各地域の需要の偏りに対応できず、キャッシュしたコンテンツが再要求される可能性が低い。

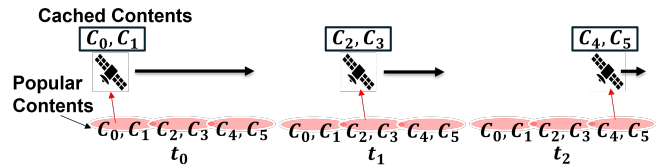


図3 従来方式

3.3 著者らの先行研究

そこで著者らは、各衛星がキャッシュするコンテンツを分散させるために、各衛星と地域をグループ分けし、各衛星は割り当てられたエリアからの要求に対してのみキャッシュコンテンツを置換する方式を提案した[11]。図4が示すように、地表グリッドをランダムにグルーピングし、各衛星は割り当

てられたエリアからの要求に対してのみキャッシュコンテンツを置換する。その結果、LEO 衛星ごとにキャッシュに保持するコンテンツを差別化し、キャッシュコンテンツの多様性の向上が図られる。しかし本方式では地表のエリアのグループをランダムに決定しているため、地表のコンテンツ需要の空間的局所性に対応できているとはいえず、キャッシュ性能の向上は限定的である。

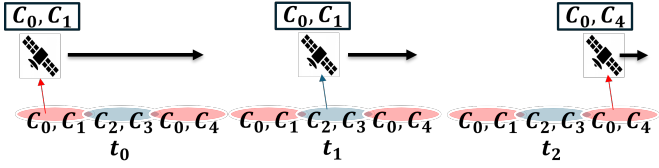


図 4 グループ化 LEO 衛星キャッシュ方式

3.4 提案方式

3.4.1 提案方式の概要

本稿の提案方式では衛星を各国の需要に基づいてグルーピングする。提案方式では地表グリッドをコンテンツ需要の類似性が高い地域同士をグループ化しており、衛星は自身のグループの地域からのみキャッシュ要求を受け付ける。本方式を用いることで、動的性の高い LEO 衛星ネットワークにおいても地上キャッシュ同様に空間的局所性を活用したキャッシュ置換が期待できる。すなわち需要が高い地域に、そのグループで人気のコンテンツを優先的に配置することが可能になるため、キャッシュヒット率向上と遅延削減が期待できる。また、ランダムなグルーピングと比較して需要の類似度が高い地域間でキャッシュを共有することが容易になる。以下に衛星と地表グリッドのグループ数を G としたときの提案アルゴリズムを示す。

- (1) 地表面を一様に六角形グリッドに分割し、それぞれを各国の需要に基づいてグループ分けする。
- (2) 同一軌道面上を周回する衛星に対して、順番に属するグループを $1, 2, \dots, G, 1, \dots$ のように割り当てる。
- (3) ユーザがコンテンツを要求した際に、通信可能範囲内でコンテンツを所持している衛星が優先的にユーザと通信する。

- (4) 直接通信する衛星がユーザの要求するコンテンツを持っていない場合は衛星間通信を利用してオリジンサーバからコンテンツを配信する。なお、経路の途中で同じグループの衛星があった場合は、その衛星もコンテンツをキャッシュする。

図 4 を用いて、衛星がキャッシュコンテンツを置換する際の手順を示す。各衛星は、自身と同じグループに属するユーザからの要求をキャッシュする。例えば赤グループに属している衛星は、自身と同じ赤色の地域のユーザから要求されたコンテンツが自身を経由する際に本コンテンツをキャッシュする。一方で青色の地域のユーザから要求に対しては単に中継衛星としての役割のみを果たし、コンテンツを自身のキャッシュに挿入しない。ただし、非割り当てエリアに関して要求されたコンテンツを保持している場合は配信を行う。

3.4.2 地表エリアのグルーピング法

本稿の提案方式では地表エリアのグルーピングを YouTube Data API を用いて各国におけるコンテンツ需要を収集し、国ごとの需要傾向を特徴ベクトルとして表現したうえで、クラスタリングを行う。クラスタリングには DBSCAN を用いた [12]。得られたクラスタ ID を地表グリッドに割り当てることで、地表全体を需要傾向に基づく複数のグループへ分割する。各国 c に対して複数のコンセプト集合 K を定義し、YouTube Data API により国別・コンセプト別の需要指標を収集する。得られた国別需要を、コンセプト次元 $|K|$ を持つベクトル

$$\mathbf{d}_c = [d_{c,1}, d_{c,2}, \dots, d_{c,|K|}] \quad (1)$$

として表現する。ここで $d_{c,k}$ は国 c におけるコンセプト k の需要量を表す。国による規模差の影響を抑えるため、 d_c は正規化を行い、国 c の需要傾向を特徴量として扱う。次に、国

別需要ベクトル集合 $\{\mathbf{d}_c\}$ に対し、DBSCAN を用いてクラスタリングを行う。DBSCAN はクラスタ数を事前に考慮する必要がないうえに、外れ値を分離できるため、需要傾向が明確に近い国群と、どの群にも属しにくい国を区別できる。距離尺度にはコサイン距離を用い、DBSCAN の主要パラメータ ϵ および min_samples を設定してクラスタを推定する。その結果、国 c にはクラスタ ID $g(c) \in \{1, 2, \dots, G\}$ が割り当てられる。最後に、地表を緯度経度 1 度のグリッドに分割し、各セル i に対して所属国 $c(i)$ を付与する。セル i のグループは

$$g(i) = g(c(i)) \quad (2)$$

として国のクラスタ ID を継承させることで定義する。この処理により、全地表の各セルが需要傾向に基づくグループ ID を持つことになり、提案方式ではこのグループ ID をもいって衛星のキャッシュ置換対象を制限する。また、DBSCAN によりノイズと判定された国が存在する場合、シミュレーションの安定性を確保するために、近傍クラスタへの割当などによりグループ ID を補完し、全セルが有効なグループ ID を持つようにする。図 5 に単一衛星の時間経過に伴う位置の変化を示す。衛星は地球の自転の影響を受けて同じ位置を周回し続けるわけではない。そのため、特定地域に重点を置いたグルーピング方式よりも一様にグルーピングした方が幅広く局所性に適応できる。

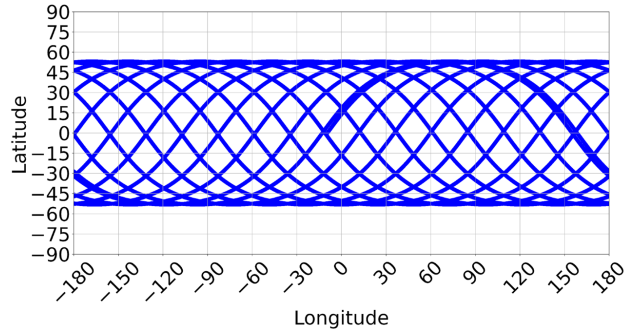


図 5 衛星の移動経路

4. 動画の視聴データの作成法

4.1 YouTube Data API の概要

本稿では各グリッドの需要ベクトルを作成するために YouTube Data API を使用した。YouTube Data API [4] は YouTube の動画・チャンネル・再生リストなどのメタデータにアクセスすることが出来る REST API である。本稿ではキーワード・地域・言語指定の動画検索が可能な `search.list` と動画閲覧数や高評価数などの統計情報が取得可能な `videos.list` の 2 つを使用した。収集した期間は 2025 年 10 月で、対象地域は Youtube Data API が対応している 100 か国とした。また、YouTube Data API は検索ワードを設定する必要があるため 30 個のコンセプトを用意し、それらに対する各国の人気上位 50 動画を収集した。

4.1.1 動画スコアの定義

取得した YouTube Data API のレスポンスは個々の動画に対する全世界の統計情報の集合であり、そのままでは地域差が生じず、地域別需要ベクトルとして扱うことは出来ない。そこで各動画の人気度を表すスコアを定義し、それを地域及びコンセプトごとに集約することで地域別コンテンツ需要ベクトルを構成する。まず、各動画 v に対して表示順位・再生回数・ローカル性を用いて人気度スコアを定義する。地域 r とコンセプト c の組に対して収集された候補動画集合を $V_{r,c}$ とし、その中での表示順位を $\text{rank}_{r,c}(v)$ (1 位が最も上位) とする。このとき、順位に基づくスコア $\text{rank_score}_{r,c}(v)$ を

$$\text{rank_score}_{r,c}(v) = \frac{R_{\max} - \text{rank}_{r,c}(v) + 1}{R_{\max}} \quad (3)$$

と定義する。ここで $R_{\max} = 10$ は考慮する最大順位である。再生回数については、地域ごとにスケールが大きく異なる点を踏まえ、各地域 r における上位動画の再生回数の 95 パーセンタイル値を $p95_r$ とおき、

$$\text{popularity}_{r,c}(v) = \min \left(1, \frac{\log_{10}(\text{views}(v) + 1)}{\log_{10}(p95_r + 1)} \right) \quad (4)$$

とすることで、0~1 の範囲に正規化した人気度指標を用いる。さらに、本稿においてはどの程度その地域固有の動画であるかを表すローカル性指標 $\text{local_hint}_{r,c}(v) \in [0, 1]$ を導入する。これはチャンネルに登録された国コードが地域 r と一致する場合には 1.0 とし、一致しない場合には動画の基底言語及びタイトル・概要欄の言語推定結果が地域の主要言語とどれだけ一致するかに基づいて 0~0.9 の値を付与することで求める。この local_hint を導入した理由は、前述の通り、Youtube Data API のレスポンスが個々の動画に対する全世界の統計情報の集合である点にある。この性質を考慮せず需要データを収集すると、全世界で人気のコンテンツが各国の人気動画ランキングの上位を独占し、各国の需要の違いが十分に表現されない。以上の 3 要素を用いて、まず

$$\begin{aligned} \text{inner}_{r,c}(v) = & w_{\text{rank}} \text{rank_score}_{r,c}(v) + w_{\text{pop}} \text{popularity}_{r,c}(v) \\ & + w_{\text{local}} \text{local_hint}_{r,c}(v) \end{aligned} \quad (5)$$

を計算する。ここで重みは

$$w_{\text{rank}} = 0.30, \quad w_{\text{pop}} = 0.45, \quad w_{\text{local}} = 0.25 \quad (6)$$

と設定し、順位・再生回数・ローカル性がバランスよく反映されるようにした。ただし、世界的にどの地域でも視聴されている動画は「地域ごとの好み」ととらえる上では情報量が小さい。このため、動画 v が出現する地域数を $\text{occ}(v)$ 、全地域数を N_{reg} とおき、TF-IDF に類似した一意性重み

$$\text{uniq}(v) = \max \left(0, \frac{\log \left(\frac{N_{\text{reg}} + 1}{1 + \text{occ}(v)} \right)}{\log(N_{\text{reg}} + 1)} \right) \quad (7)$$

を掛け合わせることで、広く視聴される汎用的な動画ほど重みを小さく、特定地域に偏って視聴される動画ほど重みを大きくする。さらに、YouTube の mostPopular API で得られるトレンド動画については、基礎スコアに 1.1 倍のブーストを掛ける。これらをまとめて、動画 v の最終的な人気度スコア $s_{r,c}(v)$ を

$$s_{r,c}(v) = \text{inner}_{r,c}(v) \times \text{uniq}(v) \times \text{trend_boost}(v) \quad (8)$$

と定義する。ここで $\text{trend_boost}(v)$ はトレンド動画なら 1.1、そうでなければ 1.0 とした。

4.1.2 地域・コンセプト別の集約と正規化
次に、収集対象とした各地域 r とコンセプト c の組 (r, c) について、上記スコアを用いて需要量を集約する。あらかじめ用意したコンセプトごとの多言語クエリを用い、YouTube Data API の `search.list` により各地域の検索上位動画を取得するとともに、`videos.list(chart="mostPopular")` により同地域のトレンド動画も取得する。両者を統合した候補動画列から上位 $R_{\max} = 10$ 本を取り出し、その集合を $V_{r,c}$ とする。地域 r 、コンセプト c に対する需要量 $d_{r,c}$ を、集合 $V_{r,c}$ に属する動画の人気度スコアの総和

$$d_{r,c} = \sum_{v \in V_{r,c}} s_{r,c}(v) \quad (9)$$

として定義する。以上により、全ての地域 r とコンセプト c について需要量 $d_{r,c}$ が求まる。これらを行方向に地域、列方向にコンセプトを並べた行列

$$\mathbf{D} = [d_{r,c}] \quad (10)$$

として整理し、本稿ではこれを地域別コンテンツ需要行列 (*region_concept_matrix*) と呼ぶ。さらに、各地域ごとに列方向の総和で割ることにより、

$$\hat{d}_{r,c} = \frac{d_{r,c}}{\sum_{c'} d_{r,c'}} \quad (11)$$

と正規化し、 $\hat{d}_{r,c}$ を「地域 r においてコンセプト c が占める需要割合」として扱う。この正規化により、地域間で総視聴規模が異なる場合でも、各地域内でのコンテンツ嗜好の違いを比較できるようになる。正規化後の行列 $\hat{\mathbf{D}}$ を、以降のクラスタリングおよび地表グループ割り当ての入力として用いる。なお、本稿で用いた YouTube Data API 由来の需要データにはいくつかの制約とバイアスが存在する。第一に API には 1 日当たりのクォータ制限が設けられており、利用可能なリクエスト数が有限である。そのため、コンセプト数や取得する動画本数を制限しながらデータ収集を行ったため、理論上取得可能な全ての動画を対象にすることは出来ていない。第二に YouTube Data API の `regionCode` は国コードであり、本稿で構築した需要ベクトルも国単位の集計に基づいて作成されたものである。その結果、同一国内の地域間に本来存在する需要の違いや言語・民族分布の差異は表現できていない。つまり、本稿では同一国内では一様な需要構造を持つと近似で扱っており、実際の空間分布に比べて地域性を過大評価している可能性が存在する。

5. 数値評価

5.1 評価条件

5.1.1 想定環境

提案方式の有効性を検証するために計算機シミュレーションを用いた数値評価を行う。具体的には提案方式とグルーピングを行わない場合でキャッシュヒット率・遅延削減量を各々、計算機シミュレーションにより算出する。総コンテンツ > 衛星数 × キャッシュ容量という制約のもと、総コンテンツ数を 50,000 とし各衛星のキャッシュ容量は 10 と設定する。シミュレータごとに重み B を設定し、ユーザ要求は、確率 $1-B$ で自国の主要言語、確率 B で多言語コンテンツを要求する。国ごとにコンセプト分布を与え、要求ごとにコンセプトをサンプリングする。次にそのコンセプトに対応する言語内コンテンツ集合を人気順に並べ、要求されるコンテンツの人気順位 r を Zipf 分布に従ってサンプリングし、その順位に対応するコンテンツ ID を決定する。各グリッドが属する国は事前処理として NOMITATIM API [13] を用いて収集し、シミュレーションの際に読み込んだ。各グリッド内の需要の偏りは、GPW [14] の人口分布と YouTube Data API で取得した需要データを統合して作成した。各グリッドの 1 秒あたりの要求数は自身のエリアの人口密度が全世界の人口密度の総和に占める割合から算出した。これは simplemaps の World Cities Database [15] により主要なネットワークサービスの月間 Web ページ訪問数から 1 秒当たりの訪問数を計算した。また、現時点で全てのネットワークトラフィックが LEO 衛星に集中するとは考えにくい上に LEO 衛星自体にまだそのような性能は搭載されていない。よって、全世界の 100 分の 1 の人口が LEO 衛星ネットワークを通して通信を行うものと仮定し、全世界 1 秒当たりの要求数が合計で 1,500 回になるように設定した。地表面を分割する際には六角形グリッドを採用した。六角形は実際に LEO 衛星が通信を行う円形セルに近い形状でありながら平面を完全に埋めることができるため、地表を均等に分割することに適している。オリジンサーバは人口都市ランキングトップ 1,000 の都市 [16] に配置し、サーバが属する国の言語のオリジナルコンテンツを保持する。オリジナルコンテンツはいずれかのオリジンサーバが 1 つだけ保持するものとする。ユーザが要求したコンテンツはキャッシュミスした際に衛星間通信を通して転送される。その際の経路選択方法は遅延を重みとしたダイクストラ法である。現実世界では接続する衛星との位置関係からリンクが切れやすくなる場合があるが、本シミュ

レーションでは衛星間通信のリンク接続切れは起こらないものとする。YouTube Data API で収集した需要と人口分布から言語ごとのコンテンツ総量を決定し、そのうえで、各オリジンサーバが保持するオリジナルコンテンツ量は、同一言語に属する都市人口の総和に対する当該都市人口の割合に比例させて割り当てる。本来はコンテンツ取得の際に大きな遅延があれば衛星がそのユーザの通信可能範囲外に移動してしまうため、ユーザにコンテンツを配信できない可能性が存在する。その場合、ハンドオーバーによってユーザにコンテンツを配信する処理が必要となるが、本シミュレーションでは要求されたコンテンツは1つの衛星で応答できるものとし、ハンドオーバーは発生しないものと仮定する。また、各衛星の置換方式はLRUである。衛星の位置が変化してもユーザ要求の空間的局所性に対応できることを示すために、シミュレーション時間は軌道周期10周分に相当する60,000秒とし、シミュレーションの開始は全ての衛星のキャッシュが埋まってからとする。

5.1.2 StarPerf

シミュレータを実装するにあたって、LEO 衛星のネットワーク性能をシミュレーション・評価するためのプラットフォームである Starperf [17] を使用した。Starperf は Starlink や OneWeb, Kuiper といった主要な LEO 衛星のシミュレーションを行えるプラットフォームである。シミュレーションでは現在実際に運用されており実用性の高い Starlink を採用した。Starlink は目的に応じて高度・起動傾斜角の異なるシェと呼ばれる層を構築している (表 1)。Starlink は主に人口の多い中緯度地域をターゲットとしてネットワークを提供することを目的としており、中緯度地域を中心に多数の LEO 衛星を展開している。

表 1 衛星モデル

	高度(km)	軌道周期	軌道傾斜角	軌道数	軌道内衛星数	合計衛星数
一層	550	5731	53.0	72	22	1584
二層	570	5755	70	36	20	720
三層	560	5743	97.6	10	52	520
四層	540	5718	53.2	72	22	1584

5.1.3 衛星間通信

衛星ネットワークを構築するにあたって、衛星間通信 (inter-satellite-links) はトポロジを形成するうえで重要な役割を担っている。ユーザからのコンテンツ要求に対して Direct Satellite がキャッシュミスの場合はオリジンサーバからコンテンツを取得する。その際に Direct Satellite からオリジンサーバまでの経路は衛星間通信を通じて確立されるが、衛星の動的性により常にトポロジが変化している。衛星同士を接続する方式は多数提案されているが、本稿においては、相対位置が変わりづらい上下左右の衛星と接続する Grid+ 方式 [17] を採用している。この手法では各衛星は同じ軌道傾斜角の衛星と接続され、メッシュのような構造を成す。同じ軌道傾斜角を持つため、衛星が移動しても相対位置が変化せず、安定した通信が期待できる。

5.1.4 要求発生モデル

LEO 衛星ネットワークのシミュレーションを行う上でユーザの要求パターンは結果に大きな影響が及ぶ。4. 節で述べた方法で作成した各国の需要データと、生成したグリッド内の人口密度を掛け合わせた需要発生パターンを新たに作成した。以下に作成した需要発生パターンの生成方法を示す。

- (1) 地表面を緯度経度 1 度ずつのグリッドに分割する。
- (2) 構成した地域別コンテンツ需要行列 $D = [d_{r,c}]$ と各グリッドの人口密度を組み合わせる。
- (3) 言語別に zipf 分布に従う要求頻度を設定する。
- (4) ユーザは自身が所属しているグリッドが属している国の主要言語に該当するコンテンツからコンテンツを要求する。先行研究 [11] と同様、地表面を緯度 1 度×経度 1 度のグリッドに分割し、各セルに対して人口密度に基づく重みを定義する。これらの総和を正規化することで各セルからユーザ要求

が発生する確率分布を構成する。そして、世界全体の要求発生率を固定したうえで、各セルにおける 1 秒当たりの要求発生率を定義し、このポアソン到着率に従って各セルからユーザ要求を生成する。4. で述べたように、YouTube Data API を用いて各国・各コンセプトに対する視聴需要を収集し、地域別コンテンツ需要行列 $D = [d_{r,c}]$ を構成した。要求発生モデルではまず国 r ごとの総需要

$$D_r = \sum_c d_{r,c} \quad (12)$$

を計算し、これを国レベルでの YouTube 由来の需要量とみなす。次に各グリッド (i,j) をそのセルが属する国を表す $r(i,j)$ に対応付けることで、セルごとの YouTube 需要重み

$$p_{i,j}^{\text{yt}} = D_{r(i,j)} \quad (13)$$

を定義する。すなわち、同一の国に属するセルは同じ YouTube 需要量を持っていると仮定し、後述する人口密度との組み合わせによって、より詳細な空間分布を表現する。上記で定義した人口密度ベースの重み $p_{i,j}^{\text{pop}}$ と YouTube 需要ベースの重み $p_{i,j}^{\text{yt}}$ を組み合わせることで、本稿では新たな需要発生パターンを構成した。具体的には、まず両者を全セルで正規化し、

$$\tilde{p}_{i,j}^{\text{pop}} = \frac{p_{i,j}^{\text{pop}}}{\sum_{i',j'} p_{i',j'}^{\text{pop}}}, \quad \tilde{p}_{i,j}^{\text{yt}} = \frac{p_{i,j}^{\text{yt}}}{\sum_{i',j'} p_{i',j'}^{\text{yt}}} \quad (14)$$

と定めたうえで、両者の積

$$w_{i,j} = \tilde{p}_{i,j}^{\text{pop}} \tilde{p}_{i,j}^{\text{yt}} \quad (15)$$

を計算し、その総和で再度正規化することでハイブリッドな需要重み

$$p_{i,j}^{\text{hyb}} = \frac{w_{i,j}}{\sum_{i',j'} w_{i',j'}} \quad (16)$$

を得る。直感的には、人口が集中しており、かつ YouTube 視聴需要も高いセルほど大きな重みになるように設計した。

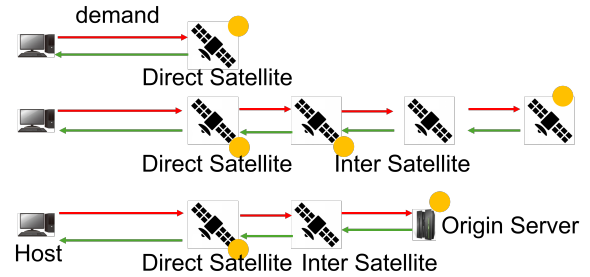


図 6 コンテンツ要求から配信までの流れ

ユーザがコンテンツ要求を発行した場合は図 6 に示す 3 パタンでコンテンツを返送する。このうち上 2 つのパタンはオリジンサーバまでのコンテンツ取得経路上に要求コンテンツをキャッシュしている衛星が存在していたことを示し、コンテンツ取得の際に生じる遅延を減少させることが出来る。以下に各パタンの概要を示す。

- (1) ユーザと直接通信している衛星 (Direct Satellite) がユーザの要求するコンテンツをキャッシュしており、そのまま返送できるパタン
- (2) Direct Satellite でキャッシュミスの場合、オリジンサーバまでの経路上の衛星 (Inter Satellite) でキャッシュヒットし、その衛星から返送できるパタン
- (3) Direct Satellite・Inter Satellite の双方でキャッシュミスし、オリジンサーバからコンテンツを取得して返送する

5.1.5 比較方式

本稿では、提案方式の有効性評価として、以下の3方式を比較対象として用いる。以降、本稿では方式名を All caching, Random clustering, Proposed method とし、以下で各方式を説明する。

All caching: 各衛星は受領したすべてのコンテンツをキャッシュに挿入する。これは従来の地上キャッシュ方式をそのまま LEO 衛星ネットワークに適応したベースラインに相当する。LEO 衛星の動的性により、地上キャッシュで前提となる空間的局所性が活用しづらく、キャッシュヒット率の低下が生じる。

Random clustering: 先行研究 [11] で著者らが提案した、地表グリッドをランダムにクラスタ化したグルーピング法である。すなわち、クラスタリングを導入するものの、需要に基づく空間的局所性を考慮しないため、グルーピング効果のうちグループ化そのものの効果を評価する比較対象となる。

Proposed method: 本稿で提案する方式である。各衛星は、自身に割り当てられたグループから発生した要求に基づいてのみキャッシュコンテンツを置換することで、LEO 衛星の移動環境下でも需要の空間的局所性を擬似的に活用し、キャッシュヒット率および遅延減少量の向上を狙う。

5.2 キャッシュヒット率

キャッシュヒット率は、ユーザが要求したコンテンツのうち、要求を満たすコンテンツが Direct Satellite または Inter Satellite から返送された割合である。図 7 に zipf の値の変化に伴うキャッシュヒット率変化を示す。なお図 7 では Direct Satellite を DS, Inter Satellite を IS と略記する。図 7 から、どのパラメタであっても Proposed method が Random caching および All caching を上回っていることが確認できる。

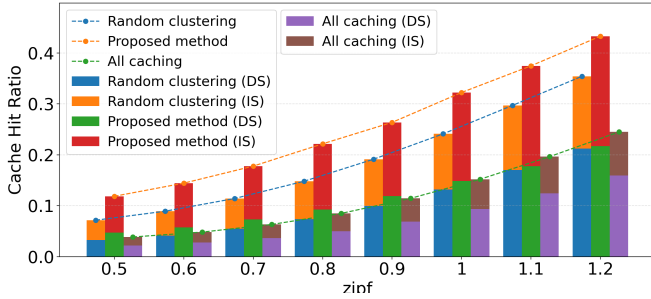


図 7 Zipf 値の変化に伴うキャッシュヒット率の増減

5.3 遅延減少量

遅延減少量はオリジンサーバから返送された場合の伝搬遅延から実際にかかった遅延を引くことで計算される。図 8 に zipf の値の変化に伴う遅延減少量の変化を示す。zipf パラメタの増加に伴って遅延減少量も増加していることが図 8 から確認できる。遅延減少量もキャッシュヒット率と同様に Proposed method が Random clustering および All caching を上回っているが、bias が 0.3 の場合、zipf パラメタが 1.1 以上の場合提案方式よりも先行研究方式の結果がよい結果を示している。

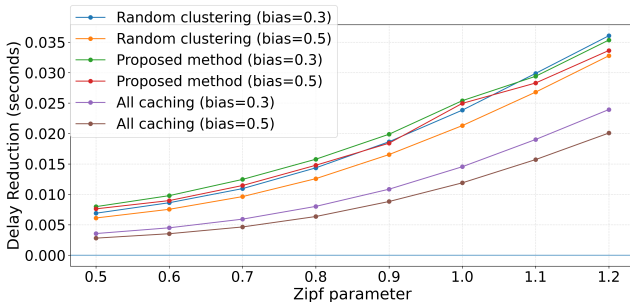


図 8 Zipf 値の変化に伴う遅延減少量の増減

6. ま と め

LEO 衛星ネットワークにおいてアップリンクのスループットが低いという課題に対して現在、LEO 衛星にキャッシュサーバを設置し、衛星からキャッシュ配信が行われている。この対策法における従来の地上キャッシュと LEO 衛星上でのキャッシュの違いに着目し、LEO 衛星ネットワークにおいてもコンテンツ人気度の空間的局所性に対応できるキャッシュ制御法を提案した。シミュレーションにより、提案方式が従来の地上キャッシュ・先行研究と比較してキャッシュヒット率・遅延減少量の双方において優れていることを確認した。今後は、より詳細で現実的なシミュレーション環境の下に評価をする予定である。

謝辞 本稿は JSPS 科研費 (25K03113, 23K28078) の助成を受けたものである。

文 献

- [1] Starlink, <https://www.starlink.com/jp>
- [2] au Starlink Direct, <https://biz.kddi.com/service/starlink-direct/>
- [3] S. Traverso, M. Ahmed, M. Garetto, P. Giaccone, E. Leonardi, and S. Niccolini, "Unravelling the impact of temporal and geographical locality in content caching systems," IEEE Trans. Multimedia, vol.17, no.10, pp.1839 – 1854, Oct. 2015.
- [4] YouTube Data API, <https://developers.google.com/youtube/v3>
- [5] H. Wu, J. Li, "A Two-layer Caching Model for Content Delivery Services in Satellite-terrestrial Networks," IEEE Global Communications Conference (GLOBECOM) 2016
- [6] J. Li, K. Xue, D. S. L. Wei, J. Liu, and Y. Zhang, "Energy efficiency and traffic offloading optimization in integrated satellite/terrestrial radio access networks," IEEE Transactions on Wireless Communications, vol. 19, no. 4, pp. 2367-2381, 2020.
- [7] A. U. Chaudhry and H. Yanikomeroglu, "Laser intersatellite links in a starlink constellation: A classification and analysis," IEEE Veh. Technol. Mag., vol. 16, no. 2, pp. 48-56, Jun. 2021.
- [8] X. Zhao, Y. Zhang, X. Wang, Y. Dong, and Y. Xie, "Towards Coverage-Aware Cooperative Video Caching in LEO Satellite Networks" IEEE Global Communications Conference (GLOBECOM), 2022.
- [9] J. Tang, J. Li, X. Chen, K. Xue, L. Zhang, Q. Sun, and J. Lu, "Cooperative Caching in Satellite-Terrestrial Integrated Networks: A Region Features Aware Approach," IEEE Transactions on Vehicular Technology, vol. 73, no. 7, pp. 10602-10616, Jul. 2024.
- [10] Jiaran Zhang, Yating Yang, Huanyu Sang, Zhuoqun Gao, Tian Song, "Content-Aware Proportional Caching for Efficient Data Delivery over Satellite Network," IEEE GLOBECOM 2023
- [11] 岡崎直人, LEO 衛星ネットワークにおける需要の空間的局所性を活かすキャッシュ制御法, 信学技報, NS2024-258, 2025 年 3 月
- [12] DBSCAN, <https://scikit-learn.org/stable/modules/generated/sklearn.cluster.DBSCAN.html>
- [13] OpenStreetMap contributors. Data © OpenStreetMap contributors. NOMITATIM API, <https://nominatim.org/>
- [14] Gridded Population of the World (GPW). NASA EarthData
- [15] Most Visited Websites Worldwide simplemaps, <https://simplemaps.com/data/world-cities>.
- [16] statista, <https://jp.statista.com/statistics/1357292/most-visited-websites-worldwide>.
- [17] Z. Lai, H. Li, and J. Li, "STARPERF: Characterizing Network Performance for Emerging MegaConstellations," IEEE 28th International Conference on Network Protocols (ICNP), 2020