

2018年度 卒業論文

カッコウ探索を用いた  
アドホックネットワーク上の複製配置

2019年1月23日

ネットワークメディア学科  
学生証番号: 544520

黒川 岳児

指導教員: 林原 尚浩

京都産業大学コンピュータ理工学部

## 概 要

参加ユーザ間での通信・やり取りが行われる，Peer to Peer(P2P) ネットワークは，データ共有や分散ストレージ等，幅広い用途で利用される．今回，P2P ネットワークと性質が似ている，アドホックネットワークを，災害時に携帯端末で構成し，物資補給情報や復旧情報，個人の避難情報等のデータがやり取りされる状況を想定した．個人の避難情報は必要とする人が比較的少なくデータ要求数が少ないが，必要とする人が必ずいるため，必要性が高い．しかし，既存の主な複製配置手法では，データ要求数の少ないデータはネットワークから消失しやすいため，必要性の高いデータであっても，ネットワークに生存し続けることは出来ない．そこで，本研究では，データ要求数の少ないデータのネットワークでの生存を考慮した複製配置を提案する．複製配置先の選出には，メタヒューリスティックアルゴリズムである，カッコウ探索を用いる．そして，本提案手法と，関連研究の提案手法，既存複製配置手法の，データの生存数の推移と，累積ストレージ占有量の推移の比較を行った．また，本提案手法と，関連研究の提案手法で行こなわれる，データ要求数の少ないデータの複製配置で，実際に配置したノードのパラメータの統計の平均値の比較を行った．

# 目次

<b>第 1 章</b>	<b>はじめに</b>	<b>1</b>
1.1	背景	1
1.2	問題点	1
1.3	目的	2
1.4	論文構成	2
<b>第 2 章</b>	<b>既存の検索・複製配置手法</b>	<b>3</b>
2.1	検索手法	3
2.2	複製配置手法	3
2.3	問題点	4
<b>第 3 章</b>	<b>関連研究</b>	<b>5</b>
3.1	提案手法	5
3.1.1	概要	5
3.1.2	スーパーノード型 P2P ネットワーク	5
3.1.3	需要予測	5
3.1.4	複製配置	6
3.2	問題点	7
<b>第 4 章</b>	<b>ネットワークモデル</b>	<b>8</b>
4.1	アドホックネットワーク	8
4.2	Random Geometric Graph	8
<b>第 5 章</b>	<b>提案手法</b>	<b>10</b>
5.1	概要	10
5.2	データの需要	10
5.2.1	データ要求	10
5.2.2	低需要データ	10
5.3	複製配置先, 複製個数の決定	11
5.4	カッコウ探索による複製配置	11
5.4.1	カッコウ探索	12
5.4.2	Levy walk	12
5.4.3	Random Geometric Graph 上の Levy walk	13
5.4.4	カッコウ探索のアルゴリズム	13
5.4.5	提案手法でのカッコウ探索のアルゴリズム	14
<b>第 6 章</b>	<b>シミュレーション</b>	<b>16</b>
6.1	概要	16
6.2	シナリオ	16
6.3	環境	17

6.4	結果と考察 . . . . .	17
6.4.1	データの生存数の推移 . . . . .	17
6.4.2	累積ストレージ占有量の推移 . . . . .	23
6.4.3	カッコウ探索と Flooding の比較 . . . . .	30
第 7 章	まとめと今後の課題	32

## 目 次

4.1	Random Geometric Graph の例 . . . . .	8
6.1	Owner Replication の, データ生存数の推移 . . . . .	18
6.2	Path Replication の, データ生存数の推移 . . . . .	19
6.3	関連研究の提案手法の, データ生存数の推移 . . . . .	20
6.4	本提案手法の, データ生存数の推移 . . . . .	21
6.5	データ 15 の, 複製配置手法ごとの, 生存数の推移 . . . . .	22
6.6	Owner Replication の, 累積ストレージ占有量の推移 . . . . .	24
6.7	Path Replication の, 累積ストレージ占有量の推移 . . . . .	25
6.8	関連研究の提案手法の, 累積ストレージ占有量の推移 . . . . .	26
6.9	本提案手法の, 累積ストレージ占有量の推移 . . . . .	27
6.10	複製配置手法ごとの, 累積ストレージ占有量の推移 . . . . .	28

## 表 目 次

6.1	低需要データの複製配置開始サイクル . . . . .	22
6.2	累積ストレージ占有量が増加しなくなったサイクル . . . . .	28
6.3	複製配置先に選んだノードの各パラメータの平均値 . . . . .	30
6.4	複製配置手法ごとの、複製配置成功・失敗回数 . . . . .	30

# 第1章 はじめに

## 1.1 背景

近年、ネットワークインフラの普及により、多くの端末がインターネットに接続するようになった。そのため、従来の、ユーザとサーバが通信してやり取りを行う、クライアントサーバ方式では、サーバ側の負担が大きくなる。そこで、ユーザ同士で通信しやり取りを行う、Peer to Peer(P2P)ネットワークが注目されている。

P2P ネットワークは、参加ピア間でのデータ共有、耐故障性の向上のため、複数のピアにデータを配置する等、幅広い用途で使用されている。これらの P2P ネットワークで扱われるデータのデータ要求数は一様ではなく、データごとに大きく異なる可能性がある。

P2P ネットワークでは、データ要求が発生すると、データの検索が行われ、検索が成功すると、データの複製が作成され、データ要求者に複製が配置される。現在、主な複製配置手法では、データ要求数の多いデータの複製は多く作成されるが、データ要求数の少ないデータは作成される複製が少ない。複製が少なければ、ユーザによる削除や、ユーザの P2P ネットワークからの離脱により、データが P2P ネットワークから消滅してしまう可能性が高くなる。

ここで、データ要求数の少ないデータの必要性や重要性が高い場合を想定する。この場合、データ要求数が少ないデータであっても、P2P ネットワークに長期間データを生存させ続ける必要がある。しかし、前述の通り、現在の主な複製配置手法では、データ要求数の少ないデータは P2P ネットワークから消滅しやすい。その為、データ要求数の少ないデータの生存を考慮した複製配置手法が必要である。

そこで、今回、災害時に携帯端末で構成されるアドホックネットワークで、物資補給情報や復旧情報、個人の避難情報等のデータがアップロードされる状況を想定する。アドホックネットワークは P2P ネットワークと特徴、ネットワークトポロジが似ており、参加ユーザ間での通信、やり取りが行われる。ネットワークにアップロードされるデータの中で、個人の避難情報は、家族や親戚、友人等、数人～数十人程度の、比較的少数の人から必要とされると考えられ、物資補給情報や復旧情報等に比べ、データ要求数が少ないと考えられる。しかし、避難情報は、家族や親戚、友人等、必ず必要とする人がいると考えられ、必要性が高いと考えられる。このケースを考慮し、データ要求数の少ないデータを長期間、アドホックネットワークに生存させる手法を提案する。

## 1.2 問題点

前述したよう、現在の主な複製配置手法は、必要性や重要度の高いデータであっても、データ要求数が少なければ、作成される複製数は少なく、P2P ネットワークから消滅しやすい。

### 1.3 目的

データ要求数の少ないデータが、P2P ネットワーク上で一定期間生存することができるよう、データ要求数の少ないデータを考慮した複製配置手法を提案し、既存の主な複製配置手法と、関連研究との比較を行う。比較内容は、データの生存数（複製数）の推移と、累積ストレージ占有量の推移である。

### 1.4 論文構成

本論文は次のように構成する。第2章では、既存のP2P ネットワークの検索・複製配置手法の説明と、問題点の提示を行う。第3章では、関連研究の説明と、問題点の提示を行う。第4章では、本研究で用いたアドホックネットワークの説明と、アドホックネットワークのモデルとして用いたRandom Geometric Graphの説明を行う。第5章では、本研究で提案する、カッコウ探索を用いた複製配置手法の説明と、カッコウ探索のアルゴリズムの改変について説明する。第6章では、シミュレーションに用いた「PeerSim」の説明と、シミュレーションシナリオの説明、シミュレーション結果・考察を行う。第7章では、本研究のまとめと、今後の課題について説明する。



## 第2章 既存の検索・複製配置手法

本章では、非構造 P2P ネットワークの既存のデータ検索手法と、複製手法の説明を行い、問題点を提示する。

### 2.1 検索手法

検索手法は、目的のデータを持つピアを発見するため、データを要求するピアが送信する検索要求メッセージの転送手法を指す。以下の検索手法が提案されている [1]。

- Flooding

データを要求するピアは、自身の全ての隣接ピアに検索要求メッセージを転送する。検索要求メッセージを受け取ったピアは、該当データを持っていない場合、同様に検索要求メッセージを自身の全ての隣接ピアに検索要求メッセージを転送する。

検索要求メッセージには TTL(Time To Live) が設けられており、検索要求メッセージが隣接ピアに転送されるごとに値が 1 つ減る。値が 0 になった場合は、データを発見することができなかったと判断され、検索失敗となる。

- Expanding Ring

TTL の値を小さく設定し、Flooding による検索を開始する。検索に失敗した場合、TTL の値を増加させ、再び Flooding による検索を行う。TTL があらかじめ設定した上限値に達した場合は検索失敗となる。

- k-walker random walk

データを要求するピアは、k 個の検索要求メッセージを自身の隣接ピアにランダムに転送する。検索要求メッセージを受け取ったピアは検索要求メッセージごとに隣接ピアの内からランダムに一台選択し、検索要求メッセージを転送する。TTL が設けられており、TTL が 0 になった場合は検索失敗となる。

### 2.2 複製配置手法

複製配置手法は、データを要求するピアが送信する検索要求メッセージが、目的のデータを持つピアに届き、検索が成功した場合に、検索の過程で得られた情報を基に複製を配置する際に用いられる手法を指す。以下の複製配置手法が提案されている [1]。

- Owner Replication

データ要求したピアにのみ複製を配置する。

- Path Replication

検索対象のデータを持つピアから、データ要求したピアに至るまでの経路上にあるすべてのピアに複製を配置する。

- Random Replication

検索対象のデータを持つピアから、データ要求したピアに至るまでの経路上にあるピアと同数のピアをネットワーク上からランダムに選び、複製を配置する。

## 2.3 問題点

既存複製配置手法は、データ要求があったデータの複製を配置する手法であるため、需要の高いデータは多くの複製が作成される。複製が多いと、そのデータへのデータ要求が成功しやすくなり、ネットワークの多くのピアでやり取りされる。

一方、需要の低いデータは、データ要求が少なく、複製が作成されにくい。複製が少ないと、ユーザによる複製の削除や、複製を持っているピアの離脱により、ネットワークからデータが消滅しやすくなる。

このように、既存複製配置手法は、需要の高いデータを効率よくネットワークに広めることを目的としており、需要の低いデータの生存を考慮していない。

## 第3章 関連研究

本章では，影山ら [2] によって提案された手法の説明と，問題点の提示を行う．

### 3.1 提案手法

#### 3.1.1 概要

影山らは，スーパーノード型 P2P ネットワークにおける低需要データの消失を防ぐ複製配置手法を提案した．データ要求数を需要とし，各データの需要予測を行い，データが低需要データか判定する．低需要データではないデータのデータ要求があり，検索が成功した場合，Owner Replication により複製配置する．低需要データであり，データ要求がない場合，提案手法による複製配置数を行う．複製配置個数は需要予測により決定され，複製配置先ノードはノードの信頼度を元に配置する．

#### 3.1.2 スーパーノード型 P2P ネットワーク

スーパーノード型 P2P ネットワークは，ネットワークにサーバを含まないネットワークであり，スーパーノードと一般ノードから構成される．スーパーノードは自身と，その配下の一般ノードを 1 グループとし，スーパーノード同士でネットワークを形成する．

スーパーノードは，ネットワーク上の高性能なノード群が担い，スーパーノード配下の一般ノードに関するデータを持つ．データの探索はスーパーノードが行い，データの取得は一般ノード同士で直接行う．スーパーノード配下の一般ノードに関するデータをスーパーノードで集中管理するため，探索速度が速い，システムの管理・制御が可能といった特徴を持つ．

#### 3.1.3 需要予測

データ要求数を需要とし，需要の変動はポアソン過程に従うと仮定する．ある時刻  $t$  における需要  $d(t)$  は次のように表される． $\lambda$  は単位時間中の平均のデータ要求数を示す．

$$d(t) = \frac{e^{-\lambda} \lambda^t}{t!}$$

単位時間毎にデータへの需要を計測し，指数平滑法に基づく需要予測を行う．時刻  $t$  における需要の実測値を  $Md_t, Fd_t$  と表す．この時，需要予測値  $Fd_{t+1}$  は次のように表される．

$$Fd_{t+1} = \alpha \times Md_t + (1 - \alpha) \times Fd_t$$

$\alpha$  は平滑係数を示す．提案手法では， $Fd_{t+1} < 0.05$  となるデータは低需要データと判断され，複製配置処理が行われる．

### 3.1.4 複製配置

複製配置個数は現在のデータ数，需要予測値，実測値の比から求める．時刻  $t$  でのファイル数を  $N_t$ ，安全数を  $S_t$  とすると，複製配置個数は次のように表される．

$$S_t = \beta \times N_t \times \frac{Fd_{t+1}}{Md_t}$$

$\beta$  は安全係数を示す．複製配置先ノードは，貢献度と生存率という2つの基準を用いて選出する．貢献度は，データのやり取りにどれだけ貢献してきたかを示す度合いである．貢献度はデータのアップロード (UL) 回数，ダウンロード (DL) 回数，平均稼働時間から決定し，次のように表される．

$$\text{貢献度} = \frac{UL \text{ 回数}}{UL \text{ 回数} + DL \text{ 回数}} \times \text{平均稼働時間}$$

生存率はユーザがサービスを利用する頻度を推定する．次の3つの仮定を利用し，生存率を決定する．

- 仮定1：ユーザの生存期間は指数分布に従う

生存期間はユーザがサービスを利用する意思のある期間を示す．ユーザのサービスからの離脱は，過去の生存期間に関わらずランダムに発生するとする．生存期間を  $\tau$  とすると，ユーザの生存期間  $f(\tau)$  の確率分布は次のように表される．

$$f(\tau) = \mu e^{-\mu\tau}$$

- 仮定2：DL，UL はポアソン過程に従う

DL と UL は，過去にいつ起きたかに関わらずランダムに発生すると仮定する． $\tau$  期間以上生存したユーザの  $T$  期間の DL と UL の回数  $x$  の確率は次のように表される．

$$P(x|\nu, \tau > T) = \frac{(\nu T)^x}{x!} \times e^{-\nu T}$$

$\nu$  はポアソン過程のパラメータであり，単位時間あたりの DL と UL の頻度を示す．

- 仮定3：生存期間の分布  $\mu$  と DL，UL の頻度の分布  $\nu$  はユーザごとに異なる

以上の3つの仮定より，生存期間  $\tau$  が  $T$  より大きくなる確率，すなはち，計測終了時点でユーザが生存していることを表す生存率は次のように表される．

$$p(\nu, \mu, \tau, T, t) = \frac{1}{1 + \frac{\mu}{\nu + \mu} [e^{(\nu + \mu)(T-t)} - 1]}$$

$t$  は初回から最後の DL または UL までの期間を示す． $T$  は初回の DL または UL から計測終了までの期間を示す．

## 3.2 問題点

影山ら [2] の研究では、スーパーノード型 P2P ネットワーク上での複製配置を提案している。しかし、今回提案する手法で用いるコンピュータは携帯端末であり、端末ごとの性能に大きな差は生じにくい。そのため、今回の提案手法では、スーパーノード型 P2P ネットワークではなく、アドホックネットワークを用いた複製配置を行う。

また、影山ら [2] の研究では、低需要と判定されたデータの複製をネットワークに生存させ続けるという手法であるため、ネットワークにアップロードされるデータの種類が増えていくと、ストレージの使用量も線型的に増加し続ける。そのため、今回の提案手法では、低需要データと判定され、複製配置を開始後、一定期間経過するまでに、一度もデータ要求がない場合、複製配置を取りやめるという方法を取る。

## 第4章 ネットワークモデル

本章では，本研究で用いるアドホックネットワークと，アドホックネットワークのモデルとして用いた，Random Geometric Graph の説明を行う．

### 4.1 アドホックネットワーク

アドホックネットワークは，PC，PDA，携帯電話といった無線で接続できる端末のみで構成されたネットワーク形態のことで，必要に応じてダイナミックなネットワークシステムの構成が可能である．各端末がデータを中継することが可能であり，端末間をリレー方式で接続することにより，これまで線の繋がりであったネットワークを，面でつなぐことが可能となった．そのため，災害時に，不通となってしまった地域で，アドホックネットワークを構築することにより，柔軟かつ迅速に通信インフラを確保するといったことが可能となる [3]．

### 4.2 Random Geometric Graph

Random Geometric Graph は  $G(n, r)$  に従う． $n$  個の頂点を 2 次元平面上に一様ランダムに配置する．頂点は通信半径  $r$  を持っており，2 つの頂点間の距離が  $r$  以下であれば，相互に辺を貼る [4]．図 4.1 に Random Geometric Graph の例を示す．

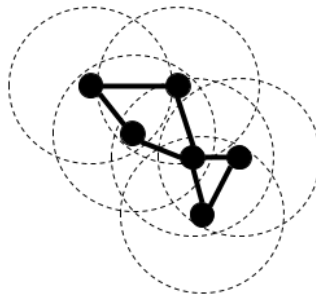


図 4.1: Random Geometric Graph の例

Random Geometric Graph の通信半径  $r$  の値を定めるために，Philips ら [5] の研究で定められた，連結性を保つためのノードの通信半径  $r$  を求める式 4.1 を用いる．

$$r = \sqrt{\frac{(1 + \epsilon) \ln A}{D\pi}}, \epsilon > 0 \quad (4.1)$$

$D$  は単位領域におけるノード数を示しており,  $D = \frac{N}{A}$ .  $N$  はノードの数,  $A$  は2次元領域の大きさを示している.

## 第5章 提案手法

本章では、低需要データを一定期間、アドホックネットワーク上に生存させ続けるための複製配置手法を提案する。提案手法の概要、データの需要について、低需要データの複製配置先・複製個数について、低需要データを複製配置するために用いるカッコウ探索の説明を行う。

### 5.1 概要

本提案手法では、まず、各ノードがデータ要求確率に基づきデータ要求を行う。データ要求を行うノードは、目的のデータを検索し、検索が成功すれば、自身に複製を配置する。全てのノードのデータ要求が完了後、アドホックネットワーク上の、各データごとの複製数を計測する。ここで、低需要データの判定を行い、低需要データと判定されれば、カッコウ探索を用いた複製配置が行われる。

### 5.2 データの需要

#### 5.2.1 データ要求

各ノードは、各データごとに、データ要求をする確率（データ要求確率）を持っている。データ要求確率はポアソン分布に従い変動すると仮定する。データ要求確率は次のように表される。

$$P(X = k) = \frac{\lambda^k e^{-\lambda}}{k!} \quad (5.1)$$

この式 5.1 は、単位時間に  $\lambda$  回発生するデータ要求が、 $k$  回発生する確率を表す。

データ要求確率に従い、データ要求が発生すると、Flooding による検索要求メッセージの転送が行われる。検索要求メッセージが  $TTL > 0$  の間に、該当データを所持するノードに到達すると、検索が成功する。その後、Owner Replication により、データ要求したノードに複製が配置される。複製には TTL が設定されており、時間経過で 1 ずつ減っていく。この値が 0 になると、ユーザによる複製の削除が発生すると仮定する。本稿では、複製数を需要とする。

#### 5.2.2 低需要データ

低需要データの複製配置を行うためには、各データごとに、データが低需要データであるのか判定する必要がある。その為、全てのノードのデータ要求が完了した後、アドホックネットワーク上の、各データごとの需要を計測する。ここで、以下の 2 つの条件を満たすデータは、低需要データとして扱うこととする。



- 条件 1 : あるデータの需要が, 全種類のデータの全需要の 5 % 以下である

$$R_i \leq 0.05 \times \sum_{i=0}^{n-1} R_i \quad (5.2)$$

$R_i$  はデータ  $i$  の複製数,  $n$  はデータの種類の数を示している.

- 条件 2 : あるデータの現在の需要が, そのデータの最大需要と比較し, 減少している

$$R_i < \max(R_i) \quad (5.3)$$

$\max(R_i)$  はデータ  $i$  の最大複製数を示している. この条件は, 需要が減少していることを示している.

低需要データを複製配置するにあたり, 複製配置先の決定と, 複製配置個数の決定を行う,

### 5.3 複製配置先, 複製個数の決定

各ノードは, バッテリー残量とストレージ残量を持っていると仮定する. そこで, 本提案手法の複製配置先の決定には, バッテリー残量とストレージ残量の 2 つの評価を用いる. 評価式は次式を用いる.

$$\text{評価値} = \alpha \times \frac{\text{現在のバッテリー残量}}{\text{最大バッテリー容量}} + \beta \times \frac{\text{現在のストレージ残量}}{\text{最大ストレージ容量}} \quad (5.4)$$

$\alpha, \beta$  は重みを表す. 今回は, 最大バッテリー容量は 100.0, 最大ストレージ容量は 10.0 とする.

バッテリー残量の高いノードは, バッテリー切れでアドホックネットワークから離脱する可能性が低く, ノードの離脱による複製の削除が発生しづらくなる. また, ストレージ残量の高いノードに割り当てることにより, 特定のノードのストレージへ複製が集中する状況を避けられ, 1 つのノード離脱による, 多数の複製の削除が発生しづらくなる.

低需要となったデータの複製は, データがアドホックネットワークに生存し続けられる程度で良い. そこで, 本提案手法では, 低需要データの複製を少なくとも 5 個, アドホックネットワークに生存させ続ける. また, ストレージ節約のため, 複製配置を開始後, 一定期間経過する間に, 一度もデータ要求ない場合, 複製配置を取りやめる, という手法を取る.

低需要データと判定されたデータは, データ要求による複製配置とは別に, カッコウ探索を用いて, 上記の評価指標が優れているノードを選び, 複製配置を行う.

### 5.4 カッコウ探索による複製配置

本提案手法では, 低需要と判定されたデータを, カッコウ探索を用いて複製配置する.

### 5.4.1 カッコウ探索

カッコウ探索 (Cuckoo Search) は, Yang と Deb によって 2009 年に提案された, 連続値最適化問題を対象とした, メタヒューリスティックアルゴリズムである [6][7].

カッコウは, 他種の鳥の巣に卵を産み付け, 育てさせる托卵という生体行動を行う. その際, 産み付ける卵を, 巣の主が生む卵と似せ, 巣の主には気づかれにくくすることができる. しかし, 巣の主に見つかってしまうこともあり, その場合, 巣の主は巣を放棄し, 新たな場所に巣を作るか, カッコウの卵を巣から捨てるという行動をとる.

カッコウ探索のアルゴリズムは, このカッコウの習性から着想を得ており, 以下の 3 つに着目している.

- カッコウは他の鳥の巣に卵を産む
- カッコウが生む卵は, 巣の主の卵に似ている
- 時には巣の主が托卵に気づき, 巣の放棄をする

カッコウの目標は, 托卵を巣の主には気づかれることなく, 自分の卵が孵り, 巣立つまで, 巣の主には育ててもらふことである. そこで, カッコウ探索では, 巣の主には気づかれない卵を「良い卵」, 良い卵のある巣が「良い巣」であると考え, 問題に対する「解」を, 巣に置かれた「卵」として表現し, 巣の卵の更新と, 悪い巣の放棄を繰り返し, 良い卵を作り出す, 世代交代をすることにより, 最適解を導き出す [8].

巣の卵の更新と, 巣の卵の放棄は, 以下の仮定に基づきモデル化する.

- カッコウは Levy walk を用いて, 他の巣の卵に, 似た卵を産む
- 托卵先の卵よりも良い卵を産んだ場合, 巣の主には気づかれない
- 悪い卵を持つ巣は放棄される
- 托卵先の数は変化しない

### 5.4.2 Levy walk

カッコウ探索では, 新しい卵の生成に Levy walk を用いる.

Levy walk は, Paul Levy によって提案された連続空間上のランダムウォークである [9]. このランダムウォークは, 向き  $\theta$  を一様ランダムに  $[0, 2\pi)$  から決定し, step length  $d$  を次式のべき分布に従って決定する.

$$p(d) \propto d^{-\lambda} \quad (5.5)$$

ここで,  $p(d)$  は確率密度で,  $\lambda$  は,  $1.0 < \lambda < 3.0$  を領域とするパラメータである. べき分布に従うため, Levy walk は短距離の移動を続ける中, 稀に長距離移動を行う. パラメータ  $\lambda$  が大きくなるにつれ, 局所的な探索になり,  $\lambda > 3.0$  の時, Levy walk は, ランダムウォークになると報告されている [10].

Levy walk は、アホウドリや蟻などの、動物の採餌行動であると報告されており [11][12]、広大な空間に対し、希少な資源を探すのに有効であると報告されている [13]。

カッコウ探索は、連続値最適化問題を対象としているため、カッコウ探索で扱われる Levy walk もまた、連続空間上での探索を想定されている。しかし、今回の提案手法では、離散空間の Random Geometric Graph 上で Levy walk を用いる。その為、離散空間での探索を想定した Levy walk に改変する必要がある。

### 5.4.3 Random Geometric Graph 上の Levy walk

前述の通り、Levy walk は連続空間上のランダムウォークであるため、離散空間上での探索は想定されておらず、探索手法を改変する必要がある。そこで、ここでは、Random Geometric Graph 上で探索するため、Levy walk の探索方法を改変する。

ここで、西田 [14] によって定義された、グラフ上の Levy walk について記す。

ランダムウォークによって移動する物体をエージェントと呼ぶこととする。

西田 [14] によって定義された、グラフ上の Levy walk では、初めに、向き  $o$  を  $[0, 2\pi)$  から一様ランダムに決定し、次に、step length  $d$  をべき分布に従って決定する。つまり、方向  $o$  へ  $d$  回、隣接頂点に遷移を行う。

続いて、これを Random Geometric Graph に適応する。Random Geometric Graph では、頂点は 2 次元平面上に配置され、各頂点は 2 次元平面上の座標を保持しており、エージェントはそれを知ることが可能であると仮定する。これにより、頂点の接続辺に対し、向きを定義することが可能である。しかし、接続辺の繋がっている方向が各頂点ごとに同じではなく、同じ方向に進み続けることができるとは限らない。この特徴を考慮し、Random Geometric Graph 用に Levy walk を定義する。

エージェントの状態を  $(v, o, d)$  の組で表す。  $v$  はエージェントが現在いる頂点を表す。  $o$  は進む方向を表し、  $[0, 2\pi)$  の実数。  $d$  は今の方向に対し、進む距離を表し、  $d > 0$ 。 エージェントは以下の手順で Random Geometric Graph 上を移動する。

1.  $v$  = Levy walk の出発点とする。
2.  $o$  を  $[0, 2\pi)$  から一様ランダムに決定する。
3.  $d$  をべき分布に従い決定する。  $d$  は式 5.5 の確率分布に従う。
4.  $v$  の接続辺の中に、その方向と  $o$  の成す角が閾値未満である辺が存在しなければ、  $o$  の方向にこれ以上進めないと判断し、 2 に戻る。
5.  $d = 0$  であれば 2 に戻る。 そうでなければ、  $d = d - 1$  とし、  $o$  に最も近い方向にある  $v$  の接続点を通り、移動した先の頂点を  $v$  とし、 4 に戻る

### 5.4.4 カッコウ探索のアルゴリズム

カッコウ探索の基本的なアルゴリズムは、最初に初期の巣をランダムに生成する。次にランダムに選んだ巣を元に、Levy walk を用いて新しい卵を生成する。ランダムに選んだ巣の卵より、新

しい卵の方が評価値が高ければ、卵を置き換える。これが托卵に相当する処理である。その後、ある一定の割合の悪い卵を持つ巣を、新しい巣で置き換える。これが、巣の主が托卵に気づき、巣を放棄することに相当する処理である。それまでに得られた最良卵を記録しつつ、上記の処理を、終了条件が満たされるまで繰り返す。終了条件には、最大世代交代数を超える、目標とする評価値以上の解を得る、最良卵が一定期間変化しない等が用いられる。以下のアルゴリズム 1 に、カッコウ探索のアルゴリズムを記す。

---

**Algorithm 1** カッコウ探索のアルゴリズム

---

```

1: Initialize:
2:   目的関数:  $f(x), x = (x_1, \dots, x_d)^T$ ;
   n 個の巣を生成:  $x_i (i = 1, 2, \dots, n)$ ;
3: while (終了条件); do
4:   巣をランダムに選択, その巣の卵を  $j$  とする;
5:   卵  $j$  を持つ巣を元に, Levy walk を用いて新しい卵  $i$  を生成;
6:   新しい卵  $i$  を評価:  $F_i$ ;
7:   ランダムに選んだ巣の卵  $j$  を評価:  $F_j$ ;
8:   if ( $F_i > F_j$ ), then
9:     ランダムに選んだ巣の卵  $j$  を, 新しい卵  $i$  で置き換え;
10:    巣の並び替え;
11:   end if
12:   一定数の割合の悪い巣を放棄し, 新しい巣を生成;
13:   巣の並び替え
14: end while
15: 最適解が決定

```

---

カッコウ探索で、巣  $N_r$  をもとにして、Levy walk により巣  $N_i$  を生成する際、 $j$  番目の卵の要素  $x_j^i$  は次式で求めることができる。

$$x_j^i = x_j^r + \alpha \times s \quad (5.6)$$

ここで、 $\alpha$  は Levy walk の 1 ステップ長を調整するためのパラメータで、 $\alpha > 0$ 。  $s$  は Levy walk で求めるステップ長を示している。

カッコウ探索は連続値最適化問題を対象としている為、離散値を扱う本提案手法では、アルゴリズムの改変が必要になる。

#### 5.4.5 提案手法でのカッコウ探索のアルゴリズム

本提案手法では、Random Geometric Graph 上に配置されたノードを探索し、評価指標の優れたノードに複製を配置する。その為、Levy walk を用いて新しい巣を生成する式 5.6 を変更する。

本提案手法では、卵は、ノードが保持している「バッテリー残量」と「ストレージ残量」に該当するため、新しい卵を生成することはできない。その為、本提案手法では以下の手順に従い、値の更新（托卵）を行う。

1. 巣  $N_i$  をもとにして, Levy walk により巣  $N_j$  を発見.
2. 巣  $N_i$  の卵  $x_i$  と, 巣  $N_j$  の卵  $x_j$  の評価を比較.
3. Levy walk により発見した巣  $N_j$  の方が良い巣であれば, 巣  $N_i$  を巣  $N_j$  で置き換える.

## 第6章 シミュレーション

本章では、アドホックネットワークのモデルとして Random Geometric Graph を実装し、低需要データを一定期間生存させるシミュレーションの説明を行う。

### 6.1 概要

提案手法を評価するために、P2P ネットワークシミュレータ「PeerSim[15]」を用いた。PeerSim は、高スケーラビリティ、動的システムをサポートしている、オーバーレイネットワークシミュレータである。PeerSim は離散イベントシミュレータであり、サイクルに基づいて実行される。

本提案手法と Owner Replication, Path Replication, 関連研究の提案手法を PeerSim でシミュレートし、比較を行った。比較内容は以下の通りである。

- データの生存数の推移
- 累積ストレージ占有量の推移

関連研究は、スーパーノード型 P2P ネットワークでの提案であった。しかし、今回の比較では、アドホックネットワークを用いる。その為、関連研究では、スーパーノードを通じて評価値の高いノードを探索していたが、今回は、ランダムにノードを選び、そのノードを元に Flooding を用いて探索する手法に変更した。

また、カッコウ探索を用いた本提案手法と、複製配置先の探索を Flooding を用いるよう設定した関連研究の提案手法が、複製配置先として選出したノードの、バッテリー残量、ストレージ残量の比較を行う。

### 6.2 シナリオ

提案手法では、以下の手順で 1 サイクルとする。

1. ノードの参加・離脱確率に従い、ノードの参加・離脱の発生。
2. データ要求確率に従い、データ要求の発生。
3. 需要の計測。
4. 関連研究の提案手法と、本提案手法のみ、低需要データがあれば複製配置。
5. 各データの TTL の減少。

6. 各ノードが保持するバッテリー残量の減少.

サイクルの初めに行われる, ノードの参加・離脱について説明する.

ノードの参加・離脱は, 参加・離脱候補ノードが 10 個あるとし, 共にポアソン分布に従い, 参加・離脱確率が変動すると仮定する. ノードの参加・離脱確率は次のように表される.

$$P(X = k) = \frac{\lambda^k e^{-\lambda}}{k!} \quad (6.1)$$

この式 6.1 は, 単位時間に  $\lambda$  回発生するノードの参加・離脱が,  $k$  回発生する確率を表す.

ノードの参加・離脱確率に従い, ノードの参加が発生すると, 新たにノードが作成され, ネットワークに追加される. 参加候補ノード数は 10 個から変動させないため, 参加したノード数分, 候補ノードを追加する.

ノードの参加・離脱確率に従い, ノードが離脱すると, ランダムに選ばれたノードが, ネットワークから削除される. その際, 保持しているデータと複製は全て削除される. 離脱候補ノード数も 10 個から変動させないため, ノード離脱確率に従い離脱したノード数分, 候補ノードを追加する.

ノードの離脱は, ノードの保持しているバッテリーが 0 以下になった場合にも発生し, バッテリーは毎サイクルごとに  $[0.0, 0.2]$  の範囲でランダムに減っていく.

## 6.3 環境

PeerSim を用いて,  $200 \times 200$  の 2 次元平面に, 2000 個のノードを一様ランダムに配置し, Random Geometric Graph 上にてシミュレーションを行う. ネットワークにアップロードされるデータの種類は全部で 50 種類で, 5 サイクルごとに, 1 種類のデータがアップロードされる. シミュレーションシナリオは, 500 サイクル分実行される.

## 6.4 結果と考察

はじめに, データ生存数の推移の結果を記す. 続いて, 累積ストレージ占有量の推移の結果を記す. 最後に, カッコウ探索と Flooding で複製配置先に選んだノードのパラメータの平均の結果を記す.

### 6.4.1 データの生存数の推移

データは全 50 種類であるが, 結果をわかりやすくするため, 図に記すのは, データ 5, データ 15, データ 25, データ 35, データ 45 の 5 個とする.

図 6.1 に Owner Replication, 図 6.2 に Path Replication, 図 6.3 に関連研究の提案手法, 図 6.4 に本提案手法の, データの生存数の推移の結果をそれぞれ記す. また, 図 6.5 にデータ 15 の, 複製配置手法ごとの, データ生存数の推移を記し, 表 6.1 に本提案手法, 関連研究の提案手法の, 低需要データの複製配置を開始したサイクル数を示す. 図 6.5 は違いをわかりやすくするため, 図

6.1～図 6.4 で用いている複製数の範囲と異なり， $[0, 200]$  の範囲の複製数を示す．また，サイクル数の範囲は図 6.1～図 6.4 と同じである．

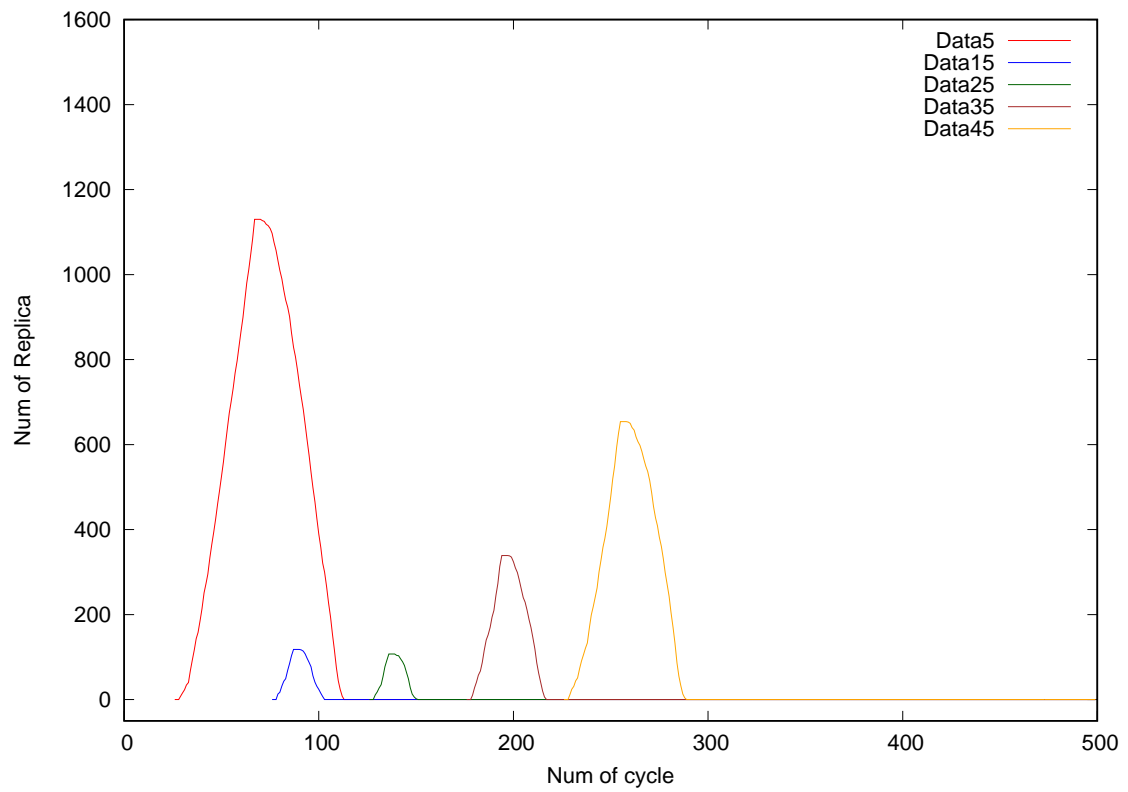


図 6.1: Owner Replication の，データ生存数の推移



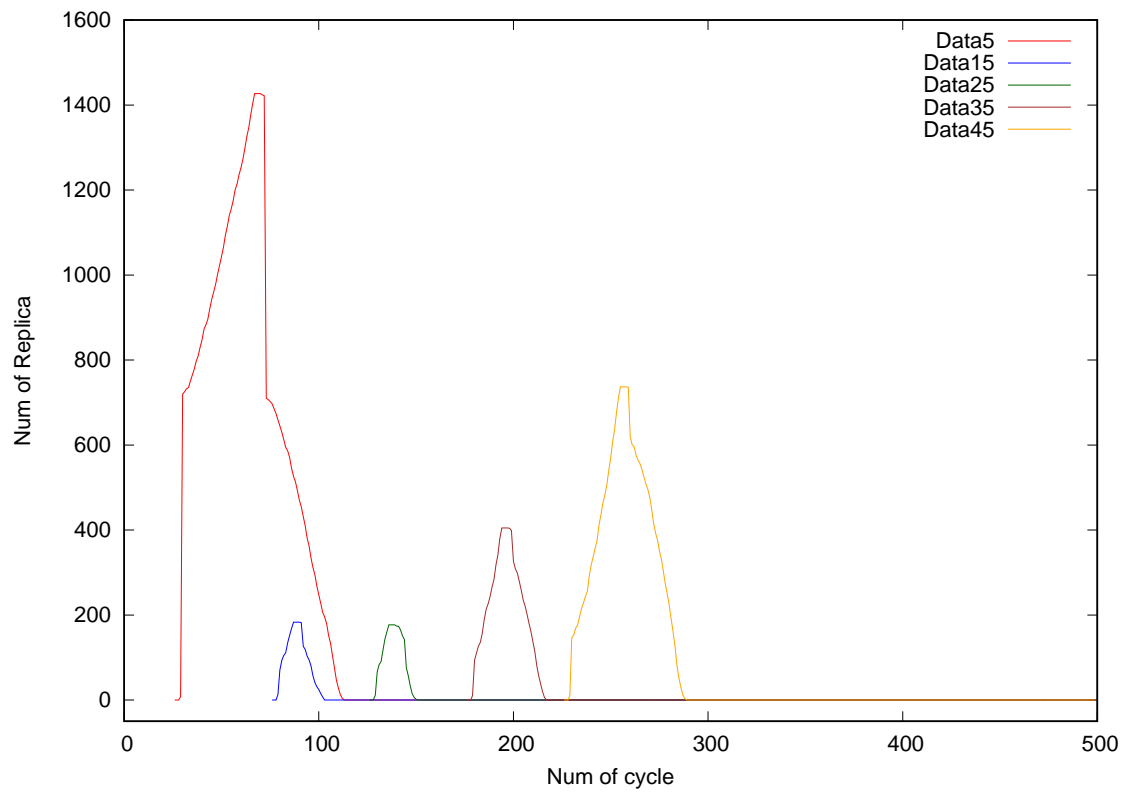


図 6.2: Path Replication の，データ生存数の推移

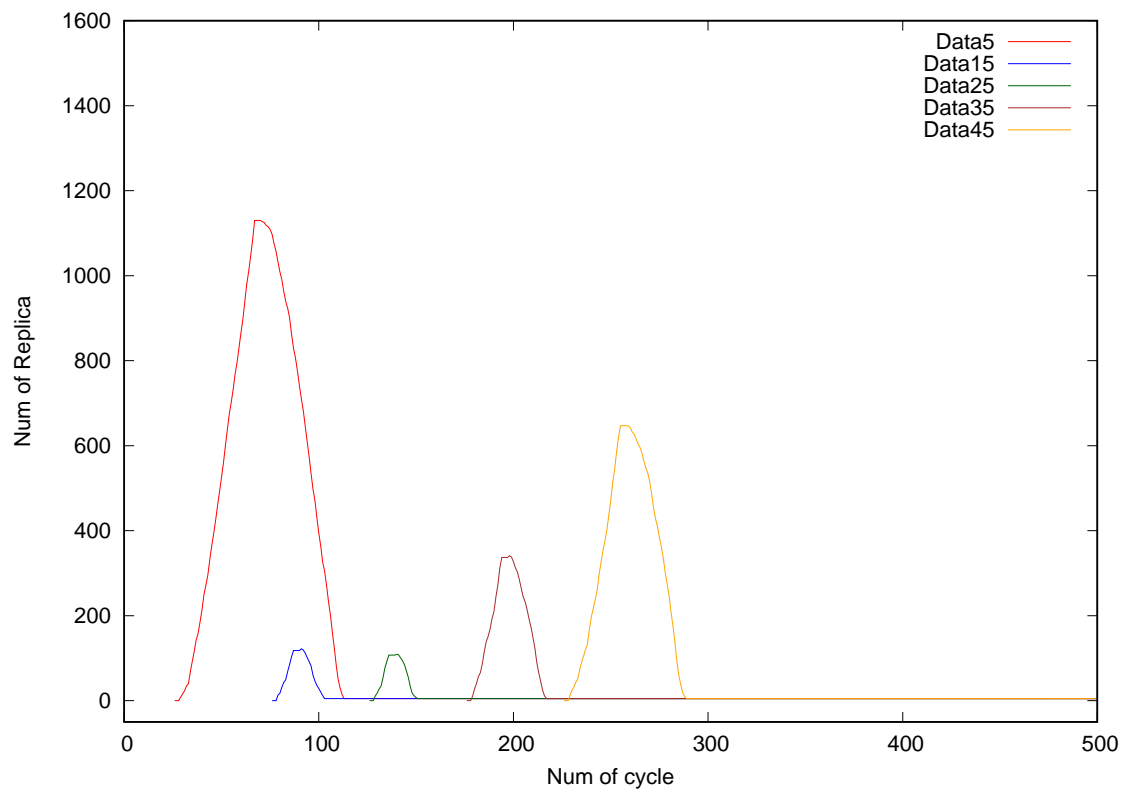


図 6.3: 関連研究の提案手法の，データ生存数の推移

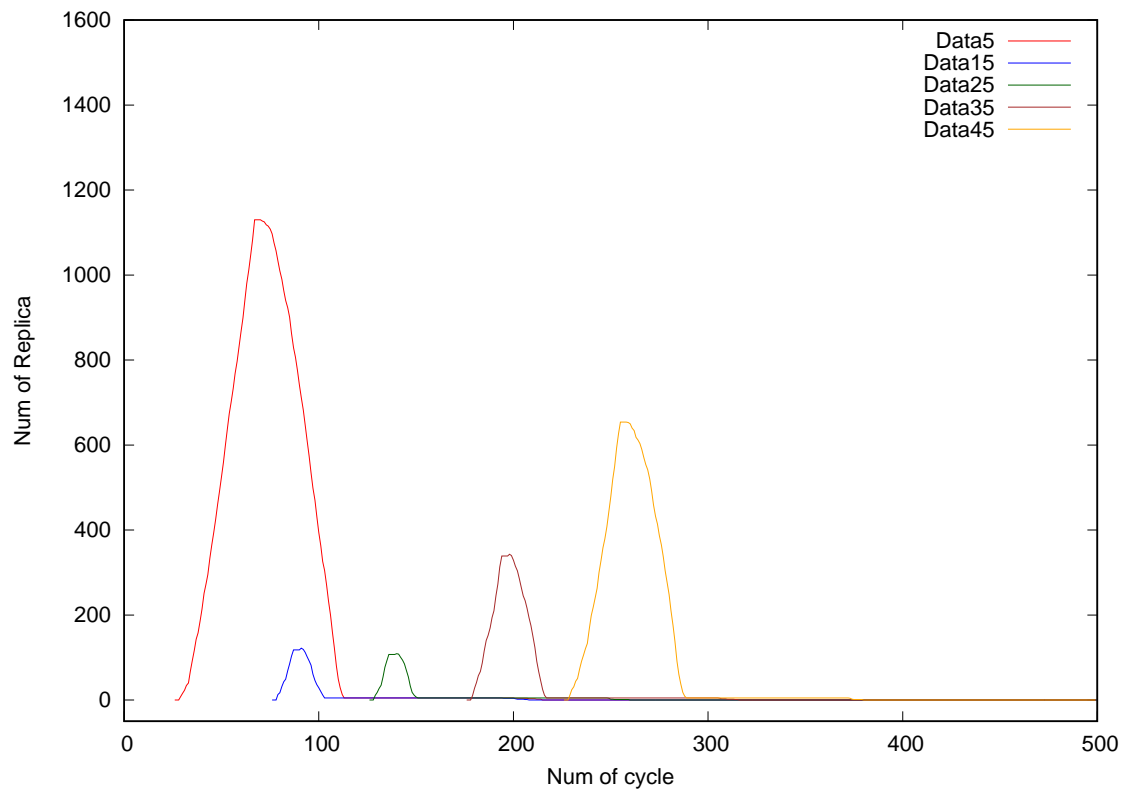


図 6.4: 本提案手法の，データ生存数の推移

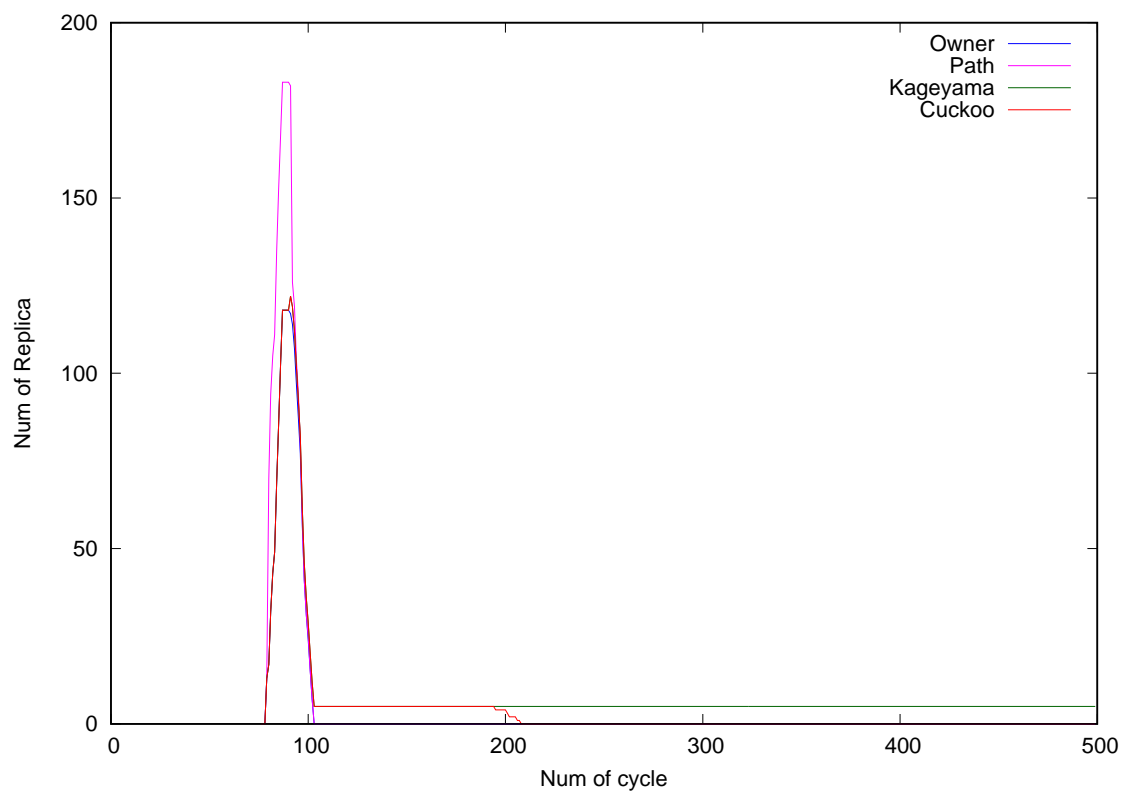


図 6.5: データ 15 の，複製配置手法ごとの，生存数の推移

表 6.1: 低需要データの複製配置開始サイクル

データ番号	複製配置手法	開始サイクル数
データ 5	関連研究	98
	本提案手法	98
データ 15	関連研究	91
	本提案手法	91
データ 25	関連研究	140
	本提案手法	140
データ 35	関連研究	198
	本提案手法	198
データ 45	関連研究	263
	本提案手法	264

全ての複製配置手法で複製数が上昇し、その後、サイクルが経過するとともに減少していく。これは、それぞれのノードがデータ要求確率に従い、データ要求を行った結果、検索が成功し、それぞれの複製配置手法により、複製が配置されたためである。また、複製数の減少は、ユーザによる削除や、ユーザの離脱による削除が発生したためである。

関連研究の提案手法と本提案手法では、データ要求のあったデータに関しては、Owner Replicationを用いて複製配置を行うため、複製数の増加量は、Owner Replication に似たものとなっていることが確認できる。減少量に関しては、関連研究の提案手法と、本提案手法では、低需要データの複製配置を別途行っているため、異なることが確認できる。

全ての複製配置手法で、データ 15 やデータ 25 とデータ 5 の複製数が大きく異なることが確認できる。これは、各ノードのデータ 15 のデータ要求確率が低く、実際にデータ要求を行うノード数が少ない場合や、複製配置されるノードのストレージ残量が、複製配置を行うデータの容量より少ない場合、また、複製配置されるノードが既に、複製配置を行うデータを所持している場合などが考えられる。

図 6.1, 図 6.2, 図 6.5 より、既存の主な複製配置手法の Owner Replication と Path Replication では、全てのデータの複製数が 0 になり、ネットワークから消滅していることが確認できる。これは、既存の主な複製配置手法は、低需要となったデータの生存を考慮していないためである。Owner Replication に比べ、Path Replication の複製数が多くなっていくことが確認できる。これは、Owner Replication は、データ要求を行ったノードにのみ複製配置を行うのに対し、Path Replication は要求対象のデータを保持するノードから、データ要求を行ったノードに至る経路上のノード全てに複製配置を行うためである。

一方、図 6.3, 図 6.5 より、関連研究の提案手法では、全てのデータの複製がネットワークに生存し続けていることが確認できる。これは、関連研究の提案手法が、低需要となったデータの生存を考慮した複製配置が行なっているためである。表 6.1 に記したサイクルから、各データの複製配置が開始され、500 サイクル経過後も複製数が 0 になっていないことが確認できる。

図 6.4, 図 6.5 より、本提案手法では、全てのデータの複製がネットワークに一定期間生存し続けていることが確認できる。これは、本提案手法が、低需要となったデータの生存を考慮した複製配置を行なっているためである。また、関連研究とは違い、複製配置を開始してから、100 サイクル経過する間に、データ要求がなければ複製配置を取りやめるという手法である、そのため、表 6.1 に記したサイクルから、各データの複製配置が開始され、100 サイクルを過ぎた後減少し、複製数が 0 になったことが確認できる。

#### 6.4.2 累積ストレージ占有量の推移

図 6.6 に Owner Replication, 図 6.7 に Path Replication, 図 6.8 に関連研究の提案手法, 図 6.9 に本提案手法の、累積ストレージ占有量の推移の結果をそれぞれ記す。また、図 6.10 に複製配置手法ごとの、累積ストレージ占有量の推移を記す。

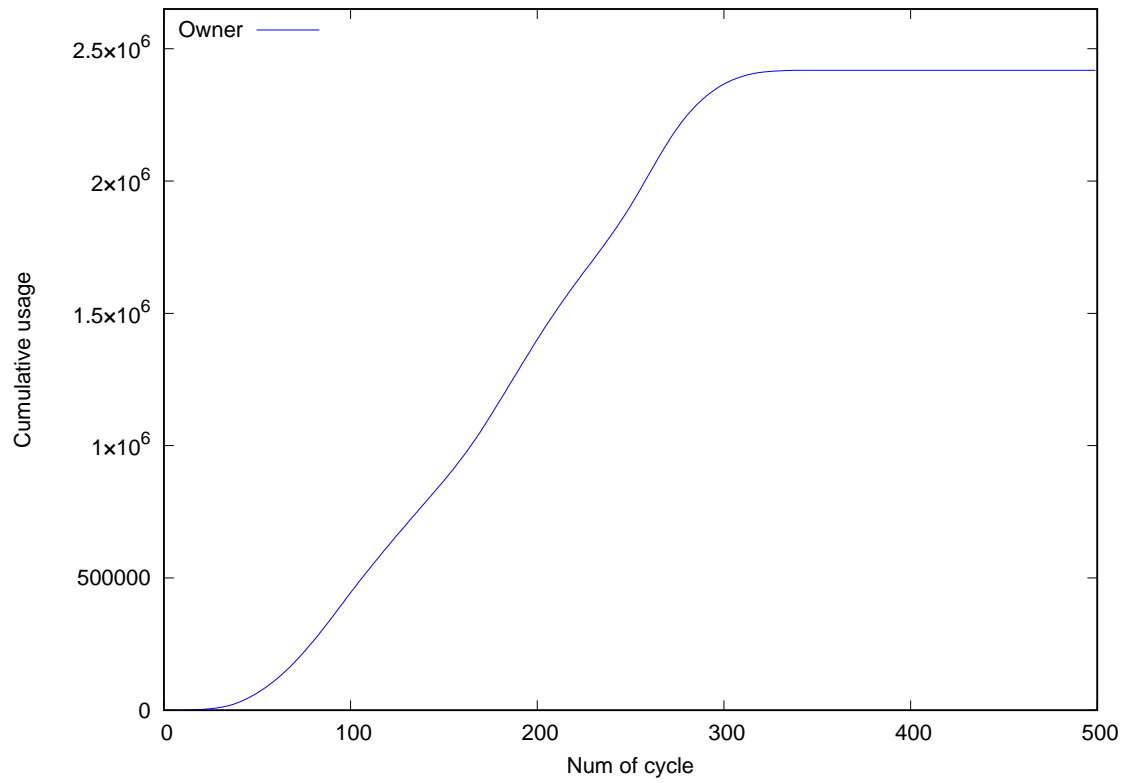


図 6.6: Owner Replication の，累積ストレージ占有量の推移

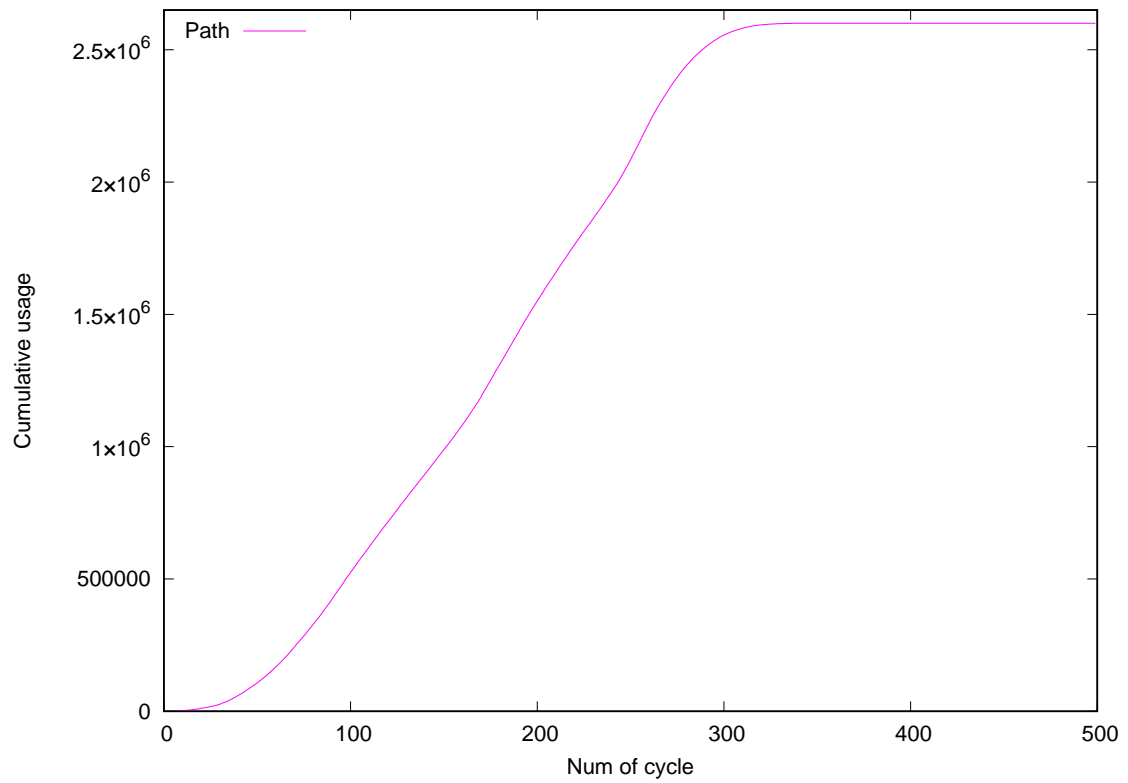


図 6.7: Path Replication の，累積ストレージ占有量の推移

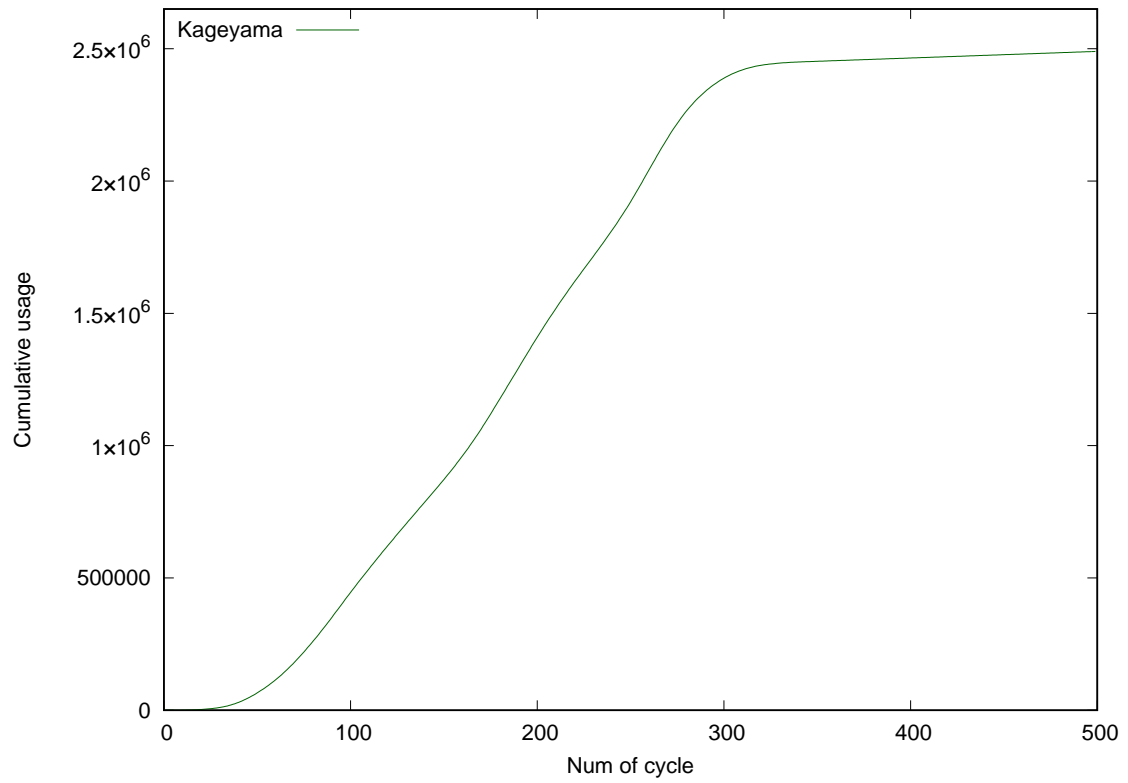


図 6.8: 関連研究の提案手法の, 累積ストレージ占有量の推移

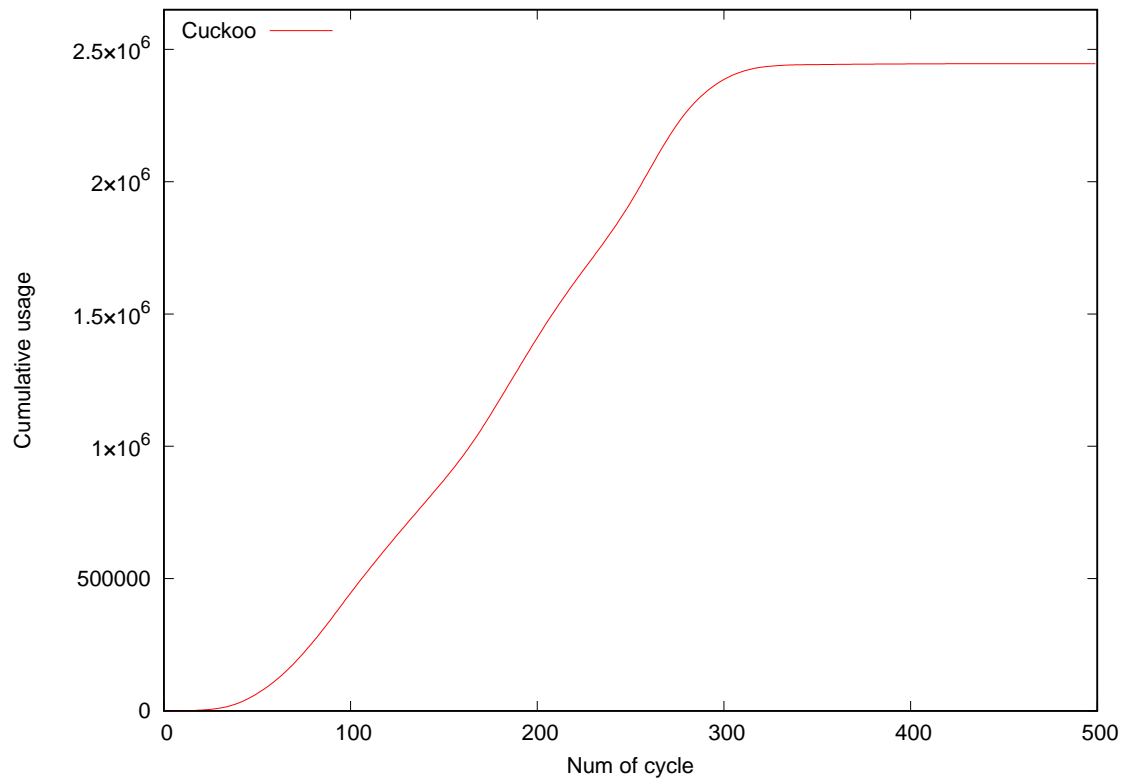


図 6.9: 本提案手法の, 累積ストレージ占有量の推移

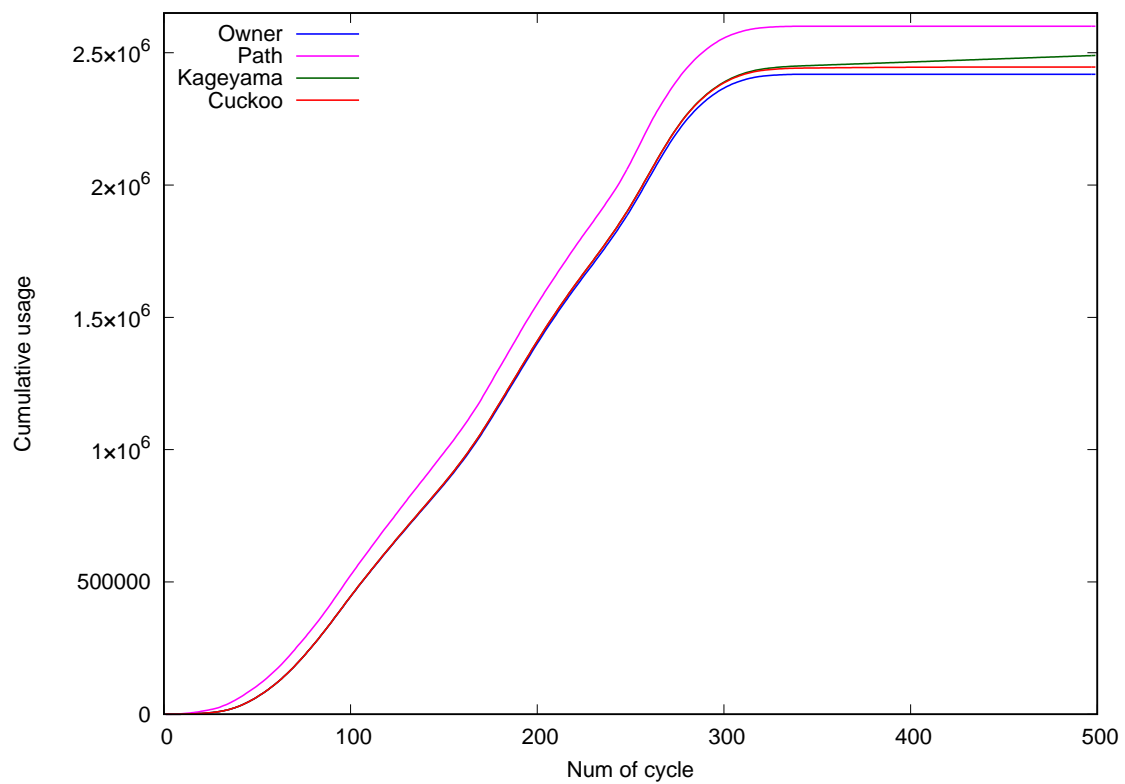


図 6.10: 複製配置手法ごとの，累積ストレージ占有量の推移

表 6.2: 累積ストレージ占有量が増加しなくなったサイクル

複製配置手法	サイクル
Owner Replication	343
Path Replication	343
関連研究の提案手法	N/A
本提案手法	459

表 6.2 のサイクルより、関連研究の提案手法以外の手法ではストレージ占有量が増加しなくなる。これは、データ 50 個が全てアップロードされた後、全種類のデータのデータ要求が無くなり、複製が減り続け、ネットワークからデータが消滅したためである。

関連研究の提案手法と本提案手法では、データ要求のあったデータに関しては、Owner Replication を用いて複製配置を行うため、ストレージ占有量の推移は Owner Replication に似たものとなっていることが確認できる。しかし、関連研究の提案手法と、本提案手法は、低需要データの複製配置を行うため、徐々に Owner Replication に比べストレージ占有量が増加していることが確認できる。

図 6.6, 図 6.7, 図 6.10, 表 6.2 より、Path Replication は、Owner Replication に比べ、多くの複製が作成されるため、ストレージ占有量が多くなっていることが確認できる。2つの複製配置手法とも、低需要データの生存を考慮していないため、表 6.2 のサイクルで、全データとその複製がネットワークから消滅し、ストレージ占有量の増加は止まっている。

一方、図 6.8, 図 6.10, 表 6.2 より、関連研究の提案手法では、ストレージ占有量が常に増え続けていることが確認できる。これは、関連研究の提案手法が、低需要となったデータの生存を考慮した複製配置が行なっており、アップロードされたデータをネットワークに生存し続けている為、断続的にノードのストレージを消費しているからである。具体的には、本提案手法より 18%, Owner Replication より 30%高い。

図 6.9, 図 6.10, 表 6.2 より、本提案手法では、Owner Replication に比べストレージ占有量が 12%多くなっている。これは、本提案手法が、低需要となったデータの生存を考慮した複製配置を行なっているため、一定期間、データがネットワークに生存し、ノードのストレージを消費しているためである。また、関連研究とは違い、複製配置を開始してから、100 サイクル経過する間に、データ要求がなければ複製配置を取りやめるという手法であるため、表 6.2 のサイクルでストレージ占有量の増加が止まっていることが確認でき、関連研究の提案手法よりストレージ占有量は 18%少ない。



### 6.4.3 カッコウ探索と Flooding の比較

表 6.3 に、複製配置先の選出に、カッコウ探索を用いた本提案手法と、Flooding を行うよう変更した関連研究の提案手法が、実際に選出したノードのバッテリー残量、ストレージ残量の統計の平均を記す。

表 6.3: 複製配置先に選んだノードの各パラメータの平均値

選出手法	パラメータ名	平均値
カッコウ探索	バッテリー残量	76.36
	ストレージ残量	8.10
Flooding	バッテリー残量	74.71
	ストレージ残量	3.38

表 6.3 より、両パラメータ共、カッコウ探索を用いた本提案手法が、Flooding を用いるよう変更した関連研究の提案手法より高い値を示していることが確認できる。これは、局所探索を行う Flooding に対し、カッコウ探索は、Levy walk とカッコウ探索のアルゴリズムを用いて広域を探索することができるためであると考えられる。離散値最適化である今回の環境では、良い解を持つノードの、近隣ノードも良い解を持つわけではないため、広域を探索する方が、優れたパラメータを持つノードを発見できると考えられる。この結果から、カッコウ探索は離散値最適化においても有効であると考えられる。

データ要求時の Owner Replication による複製配置とは別に、低需要データの複製配置を行うと、ストレージを追加で消費することになる。そのため、データ要求時の複製配置が失敗する回数が増え、成功回数が減少すると考えられる。表 6.4 に Owner Replication、関連研究の提案手法、本提案手法の、データ要求時に発生する複製配置の、複製配置成功回数と、ストレージ容量不足による複製配置失敗回数を記す。

表 6.4: 複製配置手法ごとの、複製配置成功・失敗回数

複製配置手法	状態	数
Owner Replication	失敗	34007
	成功	47339
関連研究の提案手法	失敗	34147
	成功	47199
本提案手法	失敗	34012
	成功	47334

表 6.4 より、関連研究の提案手法は Owner Replication に比べ、データ要求時の複製配置成功数は 140 回少なく、失敗数は 140 回多いという結果に対し、本提案手法は Owner Replication に比べ、データ要求時の複製配置成功数は 5 回少なく、失敗数は 5 回多いという結果であった。これは、表 6.3 の結果からわかるように、本提案手法の方がストレージ残量の大きいノードに、低需要データの複製を配置してるからであると考えられる。ストレージ残量の大きいノードに、低需要データの複製を配置することにより、そのノードはデータ要求をした際に、ストレージ容量不足

による複製配置の失敗が発生しづらくなると考えられる．これより，本提案手法は，関連研究の提案手法よりストレージを効率よく使用し，データ要求時の複製配置の失敗数を控えながら，低需要データの複製配置を行うことができたと言える．

## 第7章 まとめと今後の課題

本論文では、データ要求数の少ないデータのネットワークへの生存を考慮した複製配置手法の提案を行い、関連研究の提案手法と、既存複製配置手法との比較を行った。

データ生存数の推移の比較では、本提案手法は、一定期間、データをネットワークに生存させていることが確認できた。

累積ストレージ占有量の推移の比較では、本提案手法は、低需要データの複製配置を行った影響で、Owner Replication よりもストレージ占有量が 11% 多くなったが、関連研究の提案手法とは異なり、低需要データの複製配置を開始したサイクルから、100 サイクル経過してもデータ要求がないデータの複製配置を取りやめるという手法により、関連研究の提案手法よりストレージ占有量が 18% 少なくなり、ストレージを節約できている事を確認できた。

離散空間の Random Geometric Graph 上で、連続値最適化を対象とするカッコウ探索を用いて離散値最適化を行ったが、カッコウ探索は広域を探索することができるため、局所探索するよりも、良い結果を得ることができた。この結果より、カッコウ探索は離散値最適化においても有効であると考えられる。また、局所探索とのデータ要求時の複製配置の成功数、失敗数の比較において、共に良い結果となった。

今後の課題として、複製配置の際の評価式の改良。バッテリー減少量の改良、また、充電されたケースの追加。ノードの参加・離脱の、仮定・モデルの改良。ネットワークに生存させる、データ要求数の少ないデータの複製数を定める手法の改良が挙げられる。

## 謝 辞

本研究を進めるにあたり，ご指導を頂いた指導教員の林原教授に感謝致します．また日常の議論を通じて多くの知識や示唆を頂いた同研究室の皆様にも感謝します．

## 参考文献

- [1] 丸田大輔, 山本寛, 尾家祐二. "P2P ネットワークにおけるストレージ負荷分散実現のための複製配置手法". 電気情報通信学会 信学技報. 2004
- [2] 影山潤, 渋谷進. "P2P ネットワークにおけるファイルの消失を防ぐ複製配置手法". DEIM Forum. 2009
- [3] 野村浩司, 岩尾忠重, 細川武司, 山田健二. "アドホックネットワーク技術への取り組み". FUJITSU.57, 3. 2006
- [4] Jesper Dall, Michael Cristensen. "Random Geometric Graphs". Phys. Rev. E 66, 016121. 2002
- [5] Thomas K. Philips, Shivendra S. Panwar, Asser N. Tantawi. "Connectivity Properties of a Packet Radio Network Model". IEEE Transactions on Information Theory, vol.35, No.5. 1989
- [6] Xin-She Yang, Suash Deb. "Cuckoo Search via Levy flights". Proceedings of World Congress on Nature & Biologically Inspired Computing (NaBIC 2009, India), IEEE Publications, USA, pp. 210-214
- [7] Xin-She Yang, Suash Deb. "Engineering Optimisation by Cuckoo Search". Int. J. Mathematical Modelling and Numerical Optimisation, vol.1, No.4, 330-343. 2010
- [8] 大谷紀子 著. "進化計算アルゴリズム入門 -生物の行動科学から導く最適解-". オーム社. 2018
- [9] Paul Levy. "Theorie de L ' addition des Variables Aleatoires". Gauthier-Villars. 1937
- [10] Sergey V. Buldyrev, Ary L. Goldberger, Shlomo Havlin, Chung-Kang Peng, Michael Simons, H. Eugene Stanley. "Generalized levy-walk model for dna nucleotide sequences". Physical Review E, Vol. 47, No. 6, pp. 4514 4523. 1993
- [11] Andrew M. Edwards, Richard A. Phillips, Nicholas W. Watkins, Mervyn P. Freeman, Eugene J. Murphy, Vsevolod Afanasyev, Sergey V. Buldyrev, M. G. E. da Luz, E. P. Raposo, H. Eugene Stanley, Gandhimohan M. Viswanathan. "Revisiting Levy flight search patterns of wandering albatrosses, bumblebees and deer". Nature, Vol. 449, pp. 1044 1048. 2007
- [12] G. M. Viswanathan, V. Afanasyev, S. V. Buldyrev, E. J. Murphy, P. A. Prince, H. E. Stanley. "Levy flight search patterns of wandering albatrosses". Nature, Vol. 381, pp. 413 415. 1996
- [13] 小山英朗, 生天目章. "Random Walk と Levy Flight に基づく探索方法の比較". Technical Report 20, 社団法人情報処理学会. 2008
- [14] 西田昌彦. グラフ上の "Levy Flight に関する研究". Master ' s thesis, 京都産業大学大学院. 2016

[15] PeerSim: A Peer-to-Peer Simulator. "<http://peersim.sourceforge.net>"