

Winny ネットワークにおけるキーの生存時間とノード数を考慮したインデックスポイズニングの評価

山本 碧¹ 油田 健太郎¹ 山場 久昭² 久保田 真一郎² 朴 美娘³ 岡崎 直宣²

概要: 近年, P2P 技術を応用して作成されたファイル共有を目的としたソフトウェアが社会的に注目を浴びている. しかし, これらは効率の良いファイル共有ができる反面, その多くがファイルの流通制御機能をもたないため, 情報漏洩や違法なファイルの流通が問題となっている. この解決策としてインデックスポイズニングと呼ばれるファイル流通制御方式が研究されている. インデックスポイズニングを P2P ファイル共有ネットワークへ適用する際に, トラフィックの増大やインデックスの汚染などの問題が発生し, 通信インフラへの影響があることが確認されている. これらの問題を解決する手法として, ファイルの流通制御を低下させることのない, 動的なクラスタリングを用いた手法が提案されている. 本論文ではその手法において, 現実的な実験環境にてその有効性を評価する.

キーワード: P2P, Winny, インデックスポイズニング

1. はじめに

近年, P2P(Peer to Peer) 技術 [1] を用いたシステムである, 動画配信, IP 電話, データ通信などが世界中で利用されている. その中でもファイル共有を目的としたソフトウェアが効率的なファイル共有が可能であることから, 社会的に注目を浴びている. 世界では Gnutella, BitTorrent, FastTrack, Overnet が, 日本国内としては, ピュア P2P 型である Winny[2] や Share が利用されている. ピュア P2P 型は, 中央サーバを必要とせず, 各ノードが通信しあうことによってネットワークを構成する. クライアントサーバ型と異なり, サーバ機能を各ノードが分散して維持しているため, 特定部分に負荷が集中しない. また, 1つのノードに障害が起きた場合でも, 分散化されているため, 他ノードに及ぼす影響が少なく, 耐障害性に優れている. このような利点がある反面, その多くがファイルの流通制御機能をもたないため, 情報漏洩や, 権利者の許諾を得ていない著作物ファイルの流通, インターネットのトラフィック量の増加の要因であることなどが問題となっている. インターネットのトラフィック量は年々増加傾向にあり, 日本国内のインターネットトラフィックは平成 26 年時点で

約 2.9Tbps に達している [3]. 2010 年初頭に改正著作権法が施行されたのち, 国内の P2P トラフィックが激減していることが指摘されていることから, 著作物の違法な流通が P2P トラフィックの増加の要因として考えられる. これらの問題を解決するためのファイル流通制御方式として, ポイズニングと呼ばれる手法が研究されている. ポイズニングには, 本物のファイルに扮した偽のファイルを流通させるコンテンツポイズニング, 偽のファイル情報を拡散させるインデックスポイズニング, ファイルのパラメータ部分を偽装するパラメータポイズニングがある. なかでもインデックスポイズニングは, 流通制御に要するトラフィック量が比較的少ないため特に注目されている. しかし, インデックスポイズニングを P2P ネットワークに適用させる際, 前述のトラフィック量の増加や, インデックスの汚染などの問題が発生し, 通信インフラへの影響があることが確認されている. その問題を解決する手法として, ファイルの流通制御を低下させることのない, 動的なクラスタリングを用いた手法が提案・実装されている [4][5]. 本論文ではその手法のダミーキーの生存時間を考慮したシミュレーションと, より大規模な環境において, その有効性を評価する.

2. Winny

2.1 Winny の特徴

P2P 技術を利用したファイル共有ソフトである Winny

¹ 大分工業高等専門学校
Oita National College of Technology

² 宮崎大学
University of Miyazaki

³ 神奈川工科大学
Kanagawa Institute of Technology

や Shere はピア P2P 型である。Winny は、中継機能によりファイル転送者の区別ができないことから、高い匿名性を持ち、キャッシュの利用により効率の良いファイル共有が可能である。さらに階層化・クラスタリングの機能がある。階層化により高速回線をもつピアほどネットワークの上流に配置され、クラスタリングにより、検索範囲が限られるというフラッディング型検索方式の弱点をカバーしている。そういった利点がある反面、違法なファイルの流通や、通信トラフィックの増加による通信インフラへの影響がある。さらに、情報漏洩事件が問題となっている。ユーザを標的としたコンピュータウイルスによりコンピュータ上の情報が Winny ネットワーク上に公開され、公開された情報は、Winny ネットワークから回収することが不可能である。

Winny 起動時、上流ノード 2 つに接続を要求し、Winny ネットワークに接続する。Winny ネットワークに参加したあと、下流ノード 5 つまでの接続要求を受け付ける。このノード間の接続は、検索リンクと転送リンクの 2 種類がある。検索リンクは他のノードの情報やファイルのキーを交換、ファイル検索を行う。転送リンクはファイルの転送を行う。Winny ネットワークには、検索や転送の効率を高めるため、高速な回線に接続するノードを上流、低速な回線に接続するノードを下流とする概念がある。上流ノードになるにつれ、キーやファイルのやり取りが頻繁に行われる。高速な回線に接続するノードに多くのキーやファイルが集まるため、Winny ネットワークは全体として多数のダウンロード要求に応えることができる。回線速度は Winny の起動時に、ユーザが回線速度を指定できるようになっているため、必ずしも実際の回線速度であるとは限らない。

2.2 ファイル流通のプロトコル

ファイル保持ピアがファイルをネットワーク上に公開するとき、隣接ピアに対してファイルのメタデータを拡散する。Winny では、ファイルのメタデータをキーと呼ぶ。キーには、保持ファイルのファイル名、ファイルサイズ、ファイル ID、ファイル位置情報 (IP アドレスとポート番号)、キーの寿命などが記載されている。キーは定期的に隣接ピア間で交換される。Winny では、ネットワークに参加する各ピアがキーをインデックスとして保持することで流通ファイルの情報を管理している。これにより全てのピアがインデックスサーバの役割を果たし、中央サーバが存在しない。インデックスは、ユーザがファイルをダウンロードする際に、ファイルの位置を特定するためのファイルのポインタとして利用される。各ピアのインデックスは、隣接ピア間で定期的にキーをランダムに交換することによって更新される。ユーザが公開されているファイルを取得するためには、そのファイルのキーが必要になる。検索リンクを通してキーワードを設定した検索クエリを送信

する。この検索クエリは 1 方向にのみユニキャストで最大 6 ホップ先のノードまで転送される。キーを持つノードはクエリにキーを付けて、同じ経路を通して検索元のノードにクエリを返送する。こうして得たキーの中にある位置情報をもとに、所有者に転送リンクで直接接続し、目的のファイルのキャッシュを得る。しかし、キーの位置情報は一定の確率で書き換わるように設計されている。書き換えられた位置情報のファイル転送要求を受けたノードは、ファイルを保持していない場合、自分の持っているキーを照合し、真のファイル保持ノードからファイルをダウンロードする。Winny にはダウンロード中のファイルを同時にアップロードする仕組みがあるため、元々そのファイルを要求していたノードへファイル転送の中継を行う。中継をしたノードは、このファイルを消すことなく、そのままキャッシュとしてハードディスクに蓄積する。Winny では、人気の高いファイルほどその位置情報が書き換えられるように設計されており、中継が頻繁に行われることで、ファイルの第一発信者の区別ができなくなり、匿名性が確保される仕組みになっている。

3. インデックスポイズニング

3.1 ポイズニングの種類

P2P ファイル共有ソフトのネットワーク上で違法なファイルの流通を阻止するためのポイズニング法として、コンテンツポイズニング、インデックスポイズニング、パラメータポイズニングの 3 種類の手法がある。コンテンツポイズニングは、偽ファイルのブロックを拡散する手法で、ネットワーク負荷が大きいことが特徴である。インデックスポイズニングは、偽のファイル情報を拡散する手法で、情報量が少なく済む。パラメータポイズニングは、ファイルのパラメータ部分を偽造する手法である。これらのポイズニングは、加工した偽のデータをネットワークに拡散することで、本物のファイルをダウンロードさせないようにするファイルの流通制御方式である。なかでもインデックスポイズニングは、流通制御に要するトラフィック量が比較的少ないため特に注目されている。

3.2 インデックスポイズニングの仕組み

Winny はファイルのメタ情報をキーと呼ばれるもので管理していて、キーにはそのファイルを所有するノードの IP アドレス (ファイルの位置情報) が書き込まれている。このファイルの位置情報を書き換えたキーを周囲のノードに拡散することで、他のノードは正常にファイルを取得できなくなる。これが Winny における基本的なインデックスポイズニングである。ファイルを Winny ネットワーク上に公開するとき、ファイルからキャッシュとそのファイルのキーを生成する。ファイルをダウンロードするにはキーが必要である。インデックスポイズニングは本物のキーか

表 1 ダミーキーの例

キー情報	内容
ファイル名	本物のキーと同様
ファイルサイズ	本物のキーと同様
ファイル ID	ランダム
ファイル本体の位置情報	存在しない IP アドレス

らダミーキーを生成してそれを拡散することでファイルの流通を防ぐ。ダミーキーでは、存在しない IP アドレスに対してキャッシュを要求するため、ダウンロードは必ず失敗する [6]。Winny では、検索リンクで接続されている隣接ノードとの間でキーの交換を定期的に行うため、ファイルのキーは次第に Winny ネットワークに拡散していく。ここで、インデックスポイズニングで拡散するダミーキーの例を表 1 に示す。

3.3 インデックスポイズニングの問題点

インデックスポイズニングを P2P ネットワークに適用した場合には、以下の問題点が発生することが報告されている。

(1) 制御対象ファイル以外のファイルに対する誤った制御

インデックスポイズニングは、制御対象ファイルと同じファイル名とファイルサイズを持つダミーキーを、ネットワークに大量に拡散する手法である。インデックスポイズニングは、検索クエリのキーワードがダミーキーに記載されたファイル名の一部と一致することでクエリヒットが起これ、この結果として大量のダミーキーを入手するため、ファイル検索ピアがファイル名をキーワードとして検索クエリを発行する場合に効果的である。しかし、インデックスポイズニングでは制御対象ファイルでないファイルの検索にも影響を与えてしまう。たとえば、ファイル名に「単語 1 単語 2」の 2 つを持つ違法ファイルと、「単語 2 単語 3」の 2 つを持つ違法でないファイルを「単語 2」をキーワードに検索した際に、その検索クエリに大量のダミーキーがヒットしてしまう。そのため、違法でないファイルの検索結果に大量のダミーキーの情報が表示されるなどの影響を及ぼす可能性がある。

(2) ファイル ID を用いたファイル検索への対応

Winny をはじめ、現在普及しているほとんどの P2P ファイル共有ネットワークでは、ネットワーク上でユニークなファイル ID を用いたファイル検索が可能となっている。Winny におけるファイル ID には、ファイルのデータから求める MD5 ハッシュ値が使用されており、データが少しでも異なるファイルか

ら生成されるファイル ID は、それぞれ異なるようになっている。そのため、ファイル名やファイルサイズなどは本物のキーと同じであるが、ファイル ID は異なるダミーキーを拡散することで制御を行うインデックスポイズニングは、ファイル ID 検索を行うユーザに対しては効果がなくなってしまう。特に近年では、P2P ファイル共有ネットワーク上に大量の偽ファイルが流通するようになったため、掲示板や SNS(Social Networking Service) などにおいて本物のファイルのファイル ID が公開され、ファイル ID 検索に利用されるようになっている。

(3) ダミーキーによるインデックスの汚染

ファイル流通制御のために、Winny ネットワーク上に拡散したダミーキーがネットワーク上に長時間残る場合には、Winny ネットワークのインデックスをダミーキーが汚染する可能性がある。ここで述べたインデックスの汚染とは、P2P ファイル共有ネットワークのインデックスの大部分を制御用のダミーキーが占めることにより、制御対象でないファイルの流通が阻害されることを指す。特に Winny ネットワークにおいては、各ピア 1 台あたりがインデックスとして保持できるキーの数には限りがある(最大 35,000 個)。大量のダミーキーが長時間 Winny ネットワークに残っていると、他のファイルの流通に使用可能なインデックスの領域がダミーキーによって圧迫されるため、他のファイルの流通を長時間阻害してしまう可能性がある。Winny プロトコルはインデックスが汚染されていないことを前提に設計されているため、ダミーキーにより汚染された Winny ネットワークでは、制御対象でないファイルの検索効率などが低下してしまう。

(4) DDoS 攻撃の誘発

Winny プロトコルにはキーの認証機構が存在しないため、ファイル検索ピアがキーを入手した場合に、そのキーがダミーキーなのか本物のキーなのかを確認することができない。ダミーキーを入手したファイル検索ピアは、ダミーキーに記載されたファイルの位置のピアに対してファイル転送を要求することになる。非常に人気のあるファイルを制御している場合は、ダミーキーに記載されたファイル位置のピアに対して多数のピアからのファイル転送要求が来るため、そのピアの通信回線や PC に大きな負荷がかかってしまう。近年、このようなインデックスポイズニングが DDoS(Distributed Denial of Service) 攻撃を引き起こすという危険性が、複数の P2P ファイル共有ネットワークにおいて指摘されている。

表 2 ファイル ID 検索に対応したダミーキー

キー情報	内容
ファイル名	本物のキーと同様
ファイルサイズ	本物のキーと同様
ファイル ID	本物のキーと同様
ファイル本体の位置情報	存在しない IP アドレス

表 3 寿命が 0 秒のダミーキー

キー情報	内容
ファイル名	本物のキーと同様
ファイルサイズ	本物のキーと同様
ファイル ID	本物のキーと同様
ファイル本体の位置情報	存在しない IP アドレス
生存時間	0 [sec]

表 4 評価条件

ノード	初期クラスタワード
制御対象ノード	“Target”, “bbb”, “ccc”
制御ノード	“Target”

3.4 関連手法

3.4.1 ファイル ID 検索に対応したインデックスポイズニング

3.3 節で述べたように、インデックスポイズニングの既存手法ではファイル ID 検索に対しては制御することができなかった。そこで、表 1 のダミーキーを表 2 のダミーキーに変えることでファイル ID 検索に対しても制御を行うことができる。

3.4.2 キーの寿命を利用したインデックスポイズニング

3.3 節で述べた Winny ネットワークにインデックスポイズニングを適用する際に発生する問題を、キーの生存時間を利用することで解決する手法が提案されている [6]。キーには生存時間が設定されているため、ファイル保持ピアがネットワークから離脱すると自動的にそのキーもネットワーク上から消えていくことになる。この生存時間の規定値は 1500 秒前後であるが、任意に設定することもできる。また、Winny ネットワークでは既に保持しているキーと同じファイル ID を持つキーを受け取った場合に、それまで保持していたキーを上書きされるという特徴を持つ。これらの機能を用いて、Winny ネットワーク上の制御対象ファイルのキーをダミーキーで上書きし、キーの生存時間を 0 秒に設定することで、ネットワーク上に長時間ダミーキーを残すことなくインデックスポイズニングを適用する方法である。ダミーキーの例を表 3 に示す。拡散したダミーキーは、即座にネットワーク上から削除されるため、ダミーキーがネットワーク上で起こす問題の発生を防ぐことができる。しかし、制御対象ファイルを保持しているピアが Winny アプリケーションを再起動すると再びキーの拡散が開始されるため、継続してダミーキーの拡散を行わなければならない。そのため、通常のインデックスポイズニングよりも制御トラフィックが増加する可能性がある。

4. 提案手法

本研究では、一昨年度に提案され [4]、昨年度に実装され

た [5]、クラスタリングを用いたインデックスポイズニング手法を現実的な実験環境にて評価する。従来のインデックスポイズニング手法の 3.3 節で述べたような問題点に対し、ダミーキーのファイル ID を制御対象のファイルのファイル ID と同一のものとする。提案手法ではファイル ID を用いたファイル検索に対応させることができる。また、Winny のクラスタリングの仕組みを応用して、制御対象ファイルに関心のあるノードに対して効率的に制御を行えるようにする。この手法で拡散するダミーキーの生存時間を 1500 秒、3600 秒、5700 秒と変更させ、インデックス汚染率の推移、ダウンロード所要時間の比較を行う。また、ノード数を 23 ノードから 47 ノードに増やし、それらの比較について考察、評価を行う。

5. 評価

昨年度の実装におけるシミュレーションについて、提案手法のダミーキーの生存時間は 3600 秒であった。しかし、実際に Winny ネットワークに流通しているキーの生存時間は 1500 秒前後である。昨年度は生存時間が 1500 秒である場合と 3600 秒である場合の比較ができていないため、生存時間を変更してシミュレーションを行い、ファイルのダウンロード所要時間、インデックス汚染率を評価する。また、一般的に Winny ネットワークでは上流、中流、下流の順にノード数が増加していく構成になっている。しかし、昨年度の環境は各クラスタで上流ノード 2、中流ノード 2、下流ノード 1 という構成になっていた。そのため、本研究では 47 ノードを増やし、より大規模な環境にて評価を行った。

5.1 評価環境

VirtualBox を用いて仮想マシンを作成し、昨年度実装された流通制御システムを Winny 上で適用させる。ポイズニングはファイル ID を本物のキーと同一のものとする提案手法を採用し、最もインデックス汚染率の平均値が高く、安定していた表 4 の条件で行う。測定ノードから全ての制御対象ファイルをダウンロードに要する時間を比較することで手法の有効性を再評価する。昨年度のダミーキーの生存時間は 3600 秒であったが、本研究では生存時間を変更させてそれぞれの評価を行う。また、Winny ネットワークに参加するノード数を増やして、より大規模な環境でシミュレーションを行う。

本来のキーの生存時間である 1500 秒、昨年度評価が行われた 3600 秒、それらの差分である 2100 秒を 3600 秒に

足した 5700 秒の 3 つの生存時間のインデックス汚染率を比較する。Winny ネットワークに参加するノードは昨年度の手法では 23 ノードであり、そのうち 20 ノードが一般ノードである。一般ノードは 3 種類のクラスタに編成されており、それぞれ上流ノードが 2、中流ノードが 2、下流ノードが 1 で構成されている。また、どのクラスタにも参加していない無所属のノードが 5 ノード存在する。残りの 3 ノードは制御対象ノード、測定ノード、制御ノードである。

ここで、評価環境の概略図を図 1 に示す。制御対象ノードは 20 個の制御対象ファイルを保持し、送信可能な状態にしてある。測定ノードは制御対象ファイルをダウンロードする。制御ノードは制御対象ノードが保持している制御対象ファイルに対してポイズニングを行う。一般ノードはそれぞれ 20~21 個の異なる制御対象外ファイルと、28~29 個の制御対象外の著作物ファイルを所有している。ネットワーク上に存在するファイルの総数は 1000 個であり、そのうちの 572 個が制御対象外ファイル、408 個が制御対象外の著作物ファイル、20 個が制御対象の著作物ファイルである。ここで、制御対象外ファイルと著作物ファイルの比率は、文献 [6] を参考にし、著作物ファイルが全体のおよそ 42.75 % を占めるように設定し、制御対象の著作物ファイルの名前は “CopyrightedA[1-20 の連番]”，制御対象外の著作物ファイルの名前は “CopyrightedB[1-408 の連番]”，制御対象外ファイルの名前は “Public[1-572 の連番]” とする。

ノード数を増やした環境の詳細は、一般ノード 3 種類のクラスタを上流ノード 2、中流ノード 4、下流ノード 7 で構成し、その他がクラスタ無所属ノードが 5 ノード、制御対象ノード、測定ノード、制御ノードである。ここで、評価環境の概略図を図 2 に示す。制御対象ノードは 40 個の制御対象ファイルを保持し、送信可能な状態にしてある。測定ノードは制御対象ファイルをダウンロードする。制御ノードは制御対象ノードが保持している制御対象ファイルに対してポイズニングを行う。一般ノードはそれぞれ 18~19 個の異なる制御対象外ファイルと、26 個の制御対象外の著作物ファイルを所有している。ネットワーク上に存在するファイルの総数は 2000 個であり、上記と同じく著作物ファイルが全体のおよそ 42.75 % を占めるように設定し、制御対象の著作物ファイルの名前は “CopyrightedA[1-40 の連番]”，制御対象外の著作物ファイルの名前は “CopyrightedB[1-816 の連番]”，制御対象外ファイルの名前は “Public[1-1144 の連番]” とする。

5.2 評価結果

まず、ノード数 23 の場合のファイルダウンロード所要時間の平均値を図 3 に、ノード数 47 の場合のファイルダウンロード所要時間の平均値を図 4 に示す。ノード数 23 の場合は、生存時間が 3600 秒で最もダウンロードに時間がかかっていることが分かる。生存時間 5700 秒では最もダ

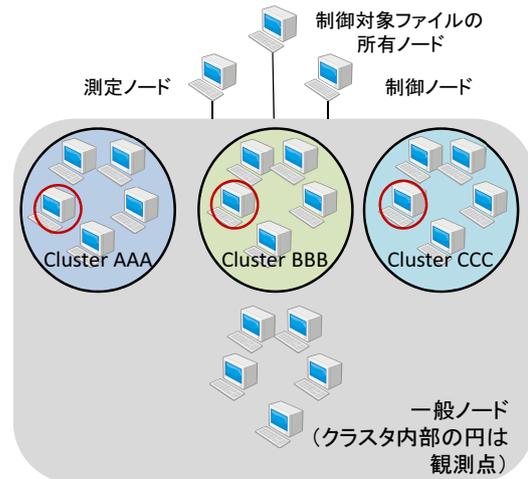


図 1 評価環境の概略図

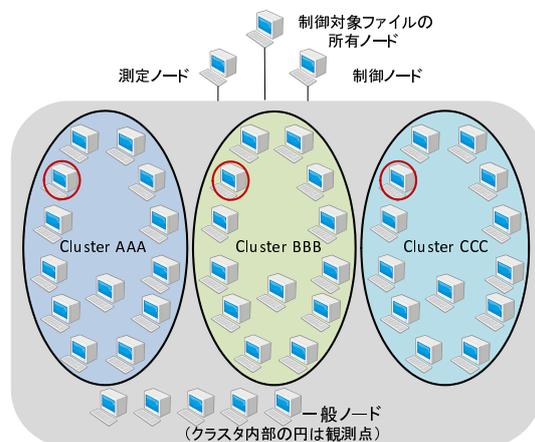


図 2 新しい評価環境の概略図

ウンロード所要時間が短い。それに比べ、ノード数 47 の場合は、生存時間が 5700 秒で最もダウンロード所要時間が長い。

ダミーキーの生存時間 1500 秒の結果を図 5、3600 秒の結果を図 6、5700 秒の結果を図 7 に示す。3 つの図を比較してみると、図 5 の汚染率が 0.6 % から 1.2 % と低く推移している。図 6 と図 7 の汚染率は 1.2 % から 2.4 % を推移しており、さほど変化がないことが分かる。

ノード数 47、生存時間 1500 秒の結果を図 8 に、3600 秒の結果を図 9 に、5700 秒の結果を図 10 に示す。ノード数が 23 の場合と比較して、全体的に汚染率が高くなっていることが分かる。

6. まとめ

本研究では、P2P ファイル共有ネットワークにおけるファイル流通制御方式であるインデックスポイズニングにおいて、制御効率を低下させることなく制御トラフィック量を削減するクラスタリングを用いたポイズニング手法の

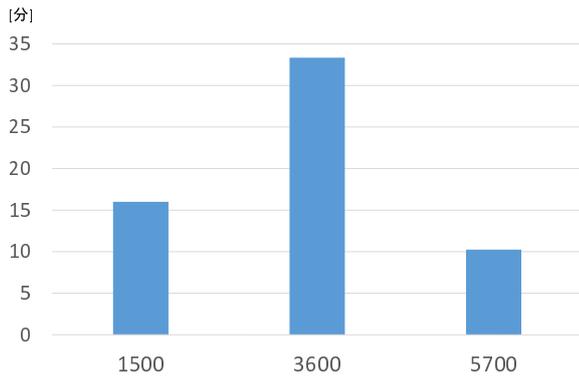


図 3 ノード数 23 のダウンロード所要時間

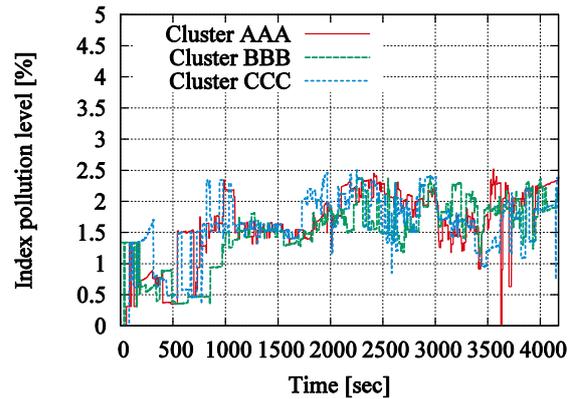


図 6 生存時間 3600 のインデックス汚染率【ノード数 23】

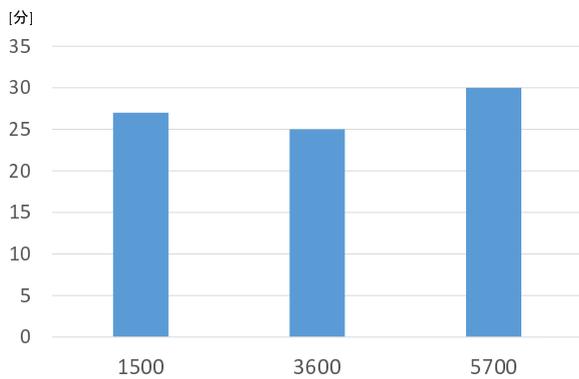


図 4 ノード数 47 のダウンロード所要時間

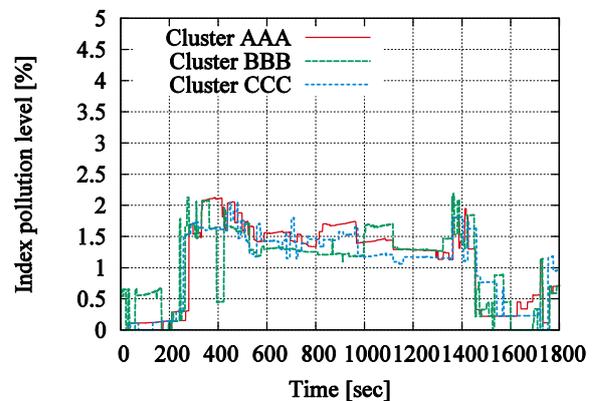


図 7 生存時間 5700 のインデックス汚染率【ノード数 23】

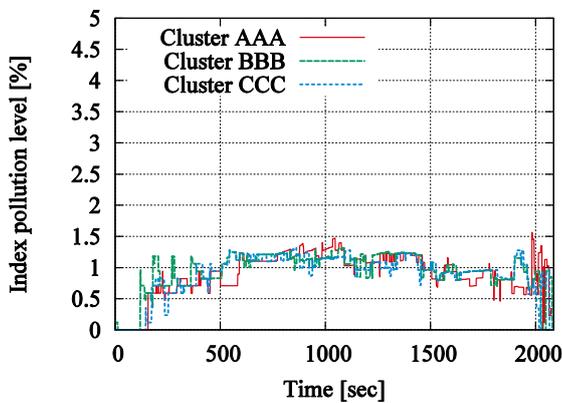


図 5 生存時間 1500 のインデックス汚染率【ノード数 23】

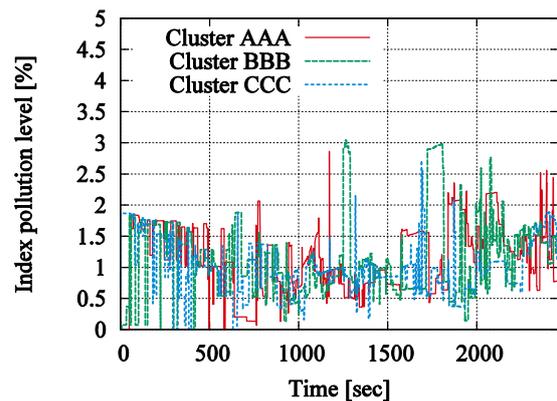


図 8 生存時間 1500 のインデックス汚染率【ノード数 47】

実装について、提案手法の再評価を行った。

評価条件として、全てのダウンロードに要した時間をダウンロード所要時間としている。

図 5 をみると、平均してインデックス汚染率が低いことから、ダミーキーの拡散が広範囲に及ばなかったのではないかと考える。図 7 では、400~1400 秒の間は安定してポイズニングが行われているが、その後インデックス汚染率が激減していることから、この間にダウンロードが急速に行われ、ダウンロードにかかる時間が短くなったと考える。インデックス汚染率が激減した原因としては、Winny で

は定期的に隣接ノードとキーの交換を行うが、その際に制御対象ファイルを持ったノードが上流ノードを隣接ノードとして、本物のキーを拡散し、上位ノードによってさらにネットワーク上に拡散されたからではないかと考えられる。

ノード数が 23 の場合では、生存時間によってインデックス汚染率の推移が異なっていたが、ノード数を 47 に増加させたインデックス汚染率は、それぞれ 0.5~2 % を推移

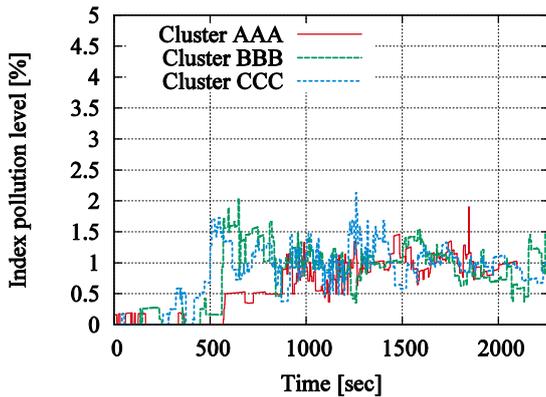


図 9 生存時間 3600 のインデックス汚染率【ノード数 47】

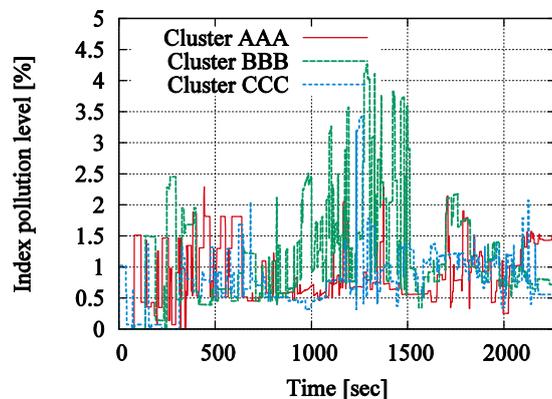


図 10 生存時間 5700 のインデックス汚染率【ノード数 47】

している。ノード数を増やしたことで Winny ネットワークのノード構成に近づけることができ、より安定してポイズニングが行えたのではないかと考えた。

6.1 今後の課題

実際の Winny ネットワークは動的であり、多くのネットワークから接続と離脱が繰り返されている。しかし、本実験の評価環境は、各ノードが同じネットワークでのシミュレーションを行っている。したがって今後、業務用のネットワーク機器を用い、ルータやスイッチの設計、および設定を行い、より現実的な環境での再評価を行う。各クラスターを異なるネットワークに置き、ダウンロード所要時間やインデックス汚染率に起こる変化についての考察が必要である。

参考文献

- [1] 江崎浩, “P2P 教科書”, インプレス R&D, 2008.
- [2] 金子勇, “Winny の技術”, ASCII, 2005.
- [3] 財務省:<http://www.soumu.go.jp/> (参照, 我が国のインターネットにおけるトラフィック総量の把握).
- [4] 後藤香穂, “P2P アプリケーションにおけるインデックスポイズニングの提案”, 大分工業高等専門学校卒業論文, 2012.

- [5] 下村龍志, “Winny ネットワークにおけるクラスタリングを用いたインデックスポイズニングの実装”, 大分工業高等専門学校卒業論文, 2013.
- [6] 吉田雅裕, 大坐畠智, 中尾彰宏, 川島幸之助: “Winny ネットワークに対するインデックスポイズニングを用いたファイル流通制御方式”, 情報処理学会論文誌, Vol.50, No.9, pp.2008-2022, (2009).