クライアント/サーバ関係に着目したピュアP2Pアプリケーショントラヒック特定方式と評価

大坐畠 智[†1] 川島 幸之助[†1]

近年，ファイル交換ネットワークにP2P技術が広く用いられている．P2Pオーバレイネットワーク上では主として音楽，動画ファイルが交換され，大きなトラヒックの源となっており，これまでのクライアント/サーバ型のアプリケーションと比較して非常に大きなトラヒックを生成している．しかし，匿名性の高い通信方式を用いているP2Pアプリケーションによるトラヒックの実態はあまりよく知られていない．それは，P2Pアプリケーションがデフォルトのサービスポートを持たず，通信が暗号化され，さらに中継ピアを経由させるファイル転送方式を用いているためである．この問題を解決するため，日本で最も人気のあるP2Pファイル共有アプリケーションであるWinnyに対するトラヒック特定方式を開発した．提案する特定方式はピア間のトランスポート層でのクライアント/サーバ関係に着目している．よって，アプリケーション層レベルのパケットの解析は必要としない．評価実験を行うことで提案方式の有効性を示す．

Evaluations of a Traffic Identification Method for a Pure P2P Application Traffic by using Client/Server Relations

Satoshi Ohzahata[†1] and Konosuke Kawashima[†1]

Peer-to-Peer (P2P) file sharing networks are widely used nowadays because of their scalability and flexibility. In overlay networks, it is mainly music and video files that are transferred, and it is known that the traffic volume is much larger than that of classical Client/Server applications. However, the nature of current P2P application traffic is not well known because of the anonymous communication architectures used. Since the application does not use the default service port and the communication is encrypted, the identification of traffic has not been feasible. To solve this problem, we have developed an identification method for pure P2P application traffic, especially for Winny, the most popular pure P2P file sharing application in Japan. Our proposed method relies only on Client/Server relationships among the peers, without recourse to application header information. In addition to describing the method, we also give an evaluation of the identification method by experiments.

1. はじめに

現在，ファイル交換システムとしてピュアP2Pネットワークが広く用いられている．このオーバレイネットワークでは音楽，ビデオファイルが主として取り扱われ，そのトラヒック量はこれまでのクライアント/サーバアプリケーションに比べ，非常に大きくなっている．この影響を調べるためにP2Pトラヒックに関する研究がさかんに行われている[1)–3)]．現在のP2Pアプリケーションは匿名性の高い通信方式を用いており，その現状はよく知られていない．それに加え，ピュアP2Pネットワークは自律分散ネットワークであり，管理することが非常に難しい．しかしながら，その実態を把握することはネットワークの設計，運用，管理において重要である．

インターネットトラヒックに対する研究をするためにはまず，アプリケーションごとにトラヒックを特定する必要がある．そのために多くのアプリケーショントラヒック特定方式が考案されてきた．

初期のP2Pアプリケーションもデフォルトサービスポートを用いて通信を行っていた：Gnutella[4)]（6346, 6347），Kazaa[5)]（1214），BitTorrent[6)]（6881–6889）（() 内はサービスポート番号）．しかし，これらのP2Pアプリケーションではデフォルトのサービスポートが必ずしも用いられているわけではない．つまり，オーバレイネットワークが同一のアプリケーションで構成されていても，それぞれのピアが用いるサービスポー

†1 東京農工大学
Tokyo University of Agriculture and Technology

トが異なり，サービスポート番号によるアプリケーションの特定が難しくなっている．

シグニチャマッチングによるアプリケーションの特定はパケットペイロードにアプリケーション特有の文字列が含まれている場合有効である[7),8)]．しかし，この方法はアプリケーションの用いるプロトコルが変更されるたびに新しくする必要があり，特定のためにはすべてのパケットを解析することになる．Winny[9)]，Share[10)]，BitTorrent のような最近の P2P ファイル共有アプリケーションでは，パケットペイロードが暗号化され，シグニチャマッチングを簡単に用いることができない．たとえ暗号化されたパケットを復号化でき，シグニチャマッチングを用いてアプリケーションを特定可能になったとしても，通信の秘密の侵害になる可能性があり，その利用は研究目的でも利用が大きく制限される．

そのため，アプリケーションレベルの情報を用いずにトラヒックのアプリケーションを特定する方法として，トランスポート層以下のみの情報を用いる方式が考案されてきた[11)–13)]．これらの方式では，P2P の通信パターンを発見的アルゴリズムを用いて解析を行い，定めた項目が基準を超えた場合に P2P として，その他の通信との区別を行っている．Karagiannis ら[11)] は，あるノードが TCP と UDP 通信を併用し，かつ，通信があった 2 つのノードにおいて，他のノードから同程度の頻度のアクセスがあった場合にその 2 つのノードが P2P の通信を行っているとしている．Perenyi ら[12)] は，前述の方式に加え，同時コネクション数，デフォルトサービスポート，大きなフローサイズ等を用いて方式の改良を行っている．Constantinou ら[13)] は，ノード間のアクセス関係をグラフ化し，直径が大きなもの，かつ，ノードがサーバ/クライアントの役目をしている場合に P2P の通信としている．上記発見的アルゴリズムは海外でよく利用されている P2P アプリケーションのために考案されているため，日本でよく利用される Winny 等にはそのまま適用しても必ずしも有効とは限らない．さらに解析をピアごと（アプリケーション）ではなく，ノードごと（IP アドレス）に行っており，同一ノードで複数の P2P アプリケーションを実行していることを想定していないため，個別の P2P アプリケーションごとに通信を特定することが難しい．

本論文では，アプリケーション層のヘッダ情報や，ユーザペイロードの情報は用いず，トランスポート層と IP 層のヘッダ情報のみで，Winny トラヒックを特定する方式を提案する．まず，囮（おとり）の Winny ピアが Winny ネットワークに参加し，他のピアの IP アドレスとサービスポート番号を囮ピアとの TCP 層以下のアクセス関係から直接特定する．次にその特定された IP アドレスとサービスポート番号をもとにして，Winny ピア間では双方向でクライアント/サーバ関係が構築されることに着目し，この関係をたどることにより，囮ピアが直接アクセスしなかったピアの IP アドレスとサービスポート番号を次々に特定する．さらに，同一ノードに Winny と他の P2P アプリケーションが実行されている場合と，Winny だけが実行されているノードでのクライアント/サーバ関係の違いを考察する．これにより，同一ノードに Winny アプリケーションのほかに P2P アプリケーション等が実行されていた場合でも Winny のサービスポートの特定を可能とする．実際の Winny ネットワークとの接続実験により，提案方式の特定精度の評価を行う．

本論文は次のように構成される．2 章でピュア P2P ファイル共有アプリケーション Winny の概略を述べる．3 章で提案する特定方式を説明するために P2P ネットワークを構成するピア間のアクセスモデルについて説明し，4 章で提案する Winny トラヒック特定方式について述べる．5 章で提案方式の評価を行い，6 章で本論文をまとめる．

2. ピュア P2P ファイル共有アプリケーション Winny

Winny は日本で開発されたピュア P2P ファイル共有アプリケーションである．Freenet[16)] のような匿名性のあるネットワーク構造を持ち，かつ，ファイル交換効率を目指して開発された．ファイル検索ネットワークやファイル共有ネットワークは，ピュア P2P ネットワークで構成されており，中央サーバ，スーパピア等は存在しない．Winny はファイル交換アプリケーションというよりも，ファイル共有アプリケーションと呼ばれることが多い．それは，Winny ユーザは欲しいファイルのキーワード等を入力するだけでよく，ファイルが見つかったときにそのあと何らかの操作を求められるということはなく，ファイルが自動的にダウンロードされるのを待つのみである．

Winny ネットワークは匿名性のあるアーキテクチャを持つ．通信，ネットワーク上で共有されるファイルが暗号化され，それぞれのピアで用いられるサービスポート番号が異なり，用いられるプロトコルは Winny の開発者が一部公開するまで非公開であった．さらにファイルを転送する方式が匿名性を高めている．1 つのファイルが 2 つのピア間で共有される際に中間ピア

を経由して，そのファイルを持っているピアが，ファイルを要求しているピアに対して送信を行う場合がある．この場合，ファイルを持っているピアと要求したピアが直接通信をすることはなく，お互いを知ることがなくなる．これらのアーキテクチャにより，Winny ネットワークの匿名性が保たれている．

Winny ネットワークは次の 3 つのネットワーク（リンク）で構成される．

(1) 隣接ピアネットワーク：新規に Winny ピアが Winny ネットワークに参加する際には，まず，すでに Winny ネットワークに参加しているピアを，IP アドレスとサービスポートの組としていくつか知る必要がある．Winny ネットワークに参加すると最大 600 の隣接 Winny ピアとリンクを維持し，隣接ピアネットワークが構成される．

(2) ファイル検索ネットワーク：その隣接ピアとのリンクのうちのいくつか（最大 7）がファイル検索用のリンクとして選ばれる．この非構造化ネットワークでどのピアがファイルを公開し，探しているかの情報が交換される．

(3) ファイル共有ネットワーク：ファイル共有の条件が満たされたとき，そのピア間でファイル共有のためのルートが中間ピアを通して確立される．1 つのピアは初期値で 2 つのダウンロードリンクを持つことができるが，Winny ネットワークに対してファイルをアップロードすると同時ダウンロードリンク数が増加される．これが Winny ネットワークでのインセンティブとなっている．

これらのネットワークはピア間のトランスポート層でのクライアント/サーバ関係により構築される．さらにピュア P2P ネットワークではネットワーク（特にファイル検索ネットワーク）を維持するために多数のピアどうしでつねにアクセスをしあうことになる．これらの特徴に着目することにより，本提案方式は Winny ピアの IP アドレスとサービスポート番号を特定していく．

3. P2P ネットワークにおけるピア間のアクセスモデル

本章で P2P ネットワークにおけるピア間に存在するクライアント/サーバ通信モデルについて述べる．このモデルをもとにして 4 章で Winny ピアの IP アドレスとサービスポート番号を特定する方式を提案する．

インターネットを用いたすべてのアプリケーションはクライアント/サーバ通信モデルが基本となっている．この通信モデルでは，通信をするために片方がサーバ，もう片方がクライアントの役割をする必要がある．サーバはクライアントのためにサービスポートを用意し，クライアントは一時ポートを用いてそのサービスポートに対して接続要求をすることによって通信が開始される．したがって，その通信は送信元 IP アドレスとポート番号，送信先 IP アドレスとポート番号，プロトコル番号によって識別される．ピュア P2P ネットワークでは，それぞれのピアがクライアントとサーバの役割を果たすが，これはアプリケーション層での役割であり，トランスポート層レベルでの 2 つのピア間の通信においては，どちらかが必ず，クライアントもしくはサーバの役割を果たす．提案するアプリケーショントラヒック特定方式では，トランスポート層でのクライアント/サーバ関係に着目をしていく．

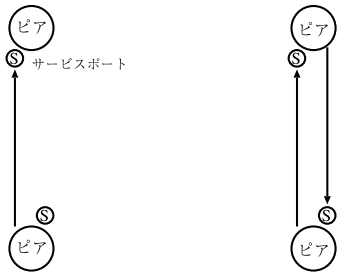


図 1 トランスポート層での通信関係；(a) 1 方向クライアント/サーバ通信モデル（左）．(b) 双方向クライアント/サーバ通信モデル（右）

Fig. 1 Communication relations in the Transport layer; (a) One-way Client/Server model (on left). (b) Two-way Client/Server model (on right).

図 1 は，P2P ネットワークにおけるクライアント/サーバ間のアクセスモデルを示す．サービスポートをピアのそばに記述し，アクセス方向である一時ポートからサービスポートへ矢印を用いて示した．

図 1 (a) に，ピア間で，1 つのコネクションが確立される 1 方向クライアント/サーバ通信モデルを示す．このモデルでは，両方のピアがサービスポートを用意していたとしても，トランスポート層では，片方のピアがサーバ，もう片方がクライアントの役割を果たすことになる．P2P アプリケーションが TCP をトランスポート層のプロトコルとして用いた場合，TCP が双方向通信をサポートするためピア間でお互いにデータの送受信が可能となる．よって，この通信モデルだけでも P2P ネットワークを構築することは可能である．アプリケーション層レベルではトランスポート層で 1 度コネクションが張られれば，トランスポート層でのクライアント/サーバ関係に関係なく，アプリケーション層でのクライアント/サーバ関係を構築することが可能である．

図 1 (b) に双方向クライアント/サーバ通信モデルを示す．このモデルでは 2 つのピア間で双方向にトランスポート層でのクライアント/サーバ関係が確立される．一般にピュア P2P ネットワークではこれら 2 つのモデルが用いられている．いくつかのピュア P2P ネットワークでは，ファイル検索ネットワークは 1 方向クライアント/サーバモデルで構築され，このネットワークの隣接ピア間でファイル交換が行われるとき，双方向クライアント/サーバ通信モデルが構築される．また，双方向クライアント/サーバ通信モデルでは，ピア間でお互いのサービスポートの存在を直接確認することで，堅牢性の高いネットワークを構築することが可能である．それは，NAT 内のピア等が正しくサービスポートを設定しているかを確かめた後にネットワークに参加させるためである．Winny では双方向クライアント/サーバ通信モデルが用いられており，4 章でアプリケーショントラヒックを特定する際に着目をしていく．

4. 提案する Winny トラヒック特定方式

提案するアプリケーショントラヒック特定方式はピアの IP アドレスとサービスポート番号を特定することにより実現する．つまり，パケットのアプリケーションレベルの情報は必要としない．まず，囮の Winny ピアが Winny ネットワークに参加することにより，Winny ネットワークに参加している Winny ピアの IP アドレスとサービスポート番号を特定する（**図 2**）．Winny ネットワークに参加しているピアの IP アドレスは，囮ピアのサービスポートにアクセスすることで判明し，サービスポート番号は囮ピアがアクセスしにいくことで判明する．つまり，囮の Winny ピアとの間で双方向クライアント/サーバ通信モデルが確立することで Winny ネットワーク上のピアの IP アドレスとサービスポート番号が判明する．しかし，囮ピアの処理能力の関係で Winny ネットワークに参加するすべてのピアの IP アドレスとサービスポート番号を特定することは難しい（図 2 ではピア A と B だけを特定したとする）．よって，まだ特定できていないピア C–F をピア間の双方向クライアント/サーバ関係を用いて特定する手順は，3 つのネットワークモデルを用いて次節から述べる．

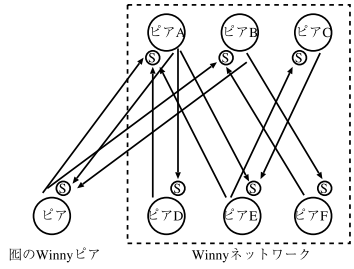


図 2 Winny ピアの IP アドレスとサービスポートの収集
Fig. 2 Collection of IP and service port number of Winny peers by the Decoy peer.

4.1 特定方式 1

図 3 に 2 つの P2P アプリケーションが存在する単純な P2P ネットワークモデルを示す．このモデルではすべての 1 つの IP アドレス（ノード）が 1 つの P2P アプリケーションを実行しているとする．つまり，それぞれのノードが 1 つだけサービスポートを持っているとする（いくつかの P2P アプリケーションは複数のサービスポートを持つが，この後述べる議論の基本的な原理には影響はなく，その場合でも拡張できる）．図 3 のようにそれぞれの P2P ネットワークに参加しているピアどうしでアクセスをしあっている．つまり，一方の P2P ネットワークに参加しているピアが他の P2P ネットワークにアクセスすることはない．

ここで，ノード A と B のサービスポート番号と IP アドレスが囮ピアと双方向クライアント/サーバ通信モデルを確立することにより特定されているとする．この特定されているノードは囮ピアと同じ P2P アプリケーションを実行していることが分かる．図 3 のように 1 つのノードで 1 つの P2P アプリケーションを実行している場合，双方向クライアント/サーバ通信モデルが確立されているならば，ノード A，B にアクセスをしているノード D，E，F も同じ P2P アプリケーションを実行していることが判明する．

よって，この手順を繰り返すことにより，次々に Winny に参加しているピアを特定することができる（最終的にはピア C も）．つまり，1 つの Winny のサービスポートを見つけることができれば，そこからの双方向クライアント/サーバアクセスモデルをたどることで，他の Winny ピアのサービスポートと IP アドレスを次々に特定することが可能である．

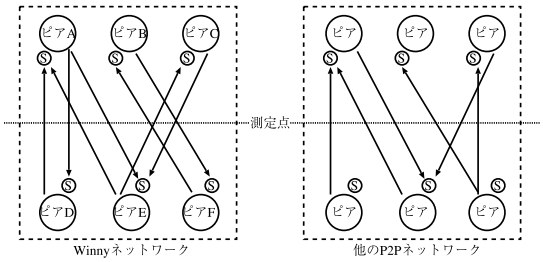


図 3 単純な P2P ネットワークモデル
Fig. 3 Simple P2P network model.

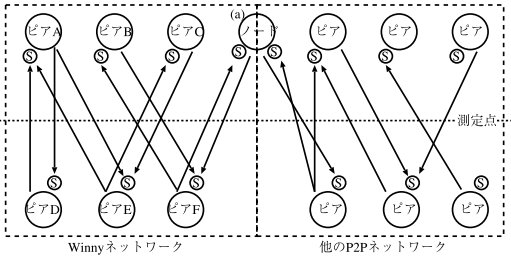


図 4 複雑なネットワークモデル 1
Fig. 4 Complex network model 1.

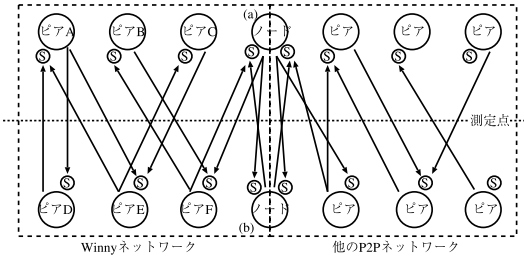


図 5 複雑なネットワークモデル 2
Fig. 5 Complex network model 2.

図 3 にトラヒック測定点を示してある．一般に，トラヒック測定点はバックボーンネットワークとスタブネットワークの間に設定される．測定点では，それらネットワーク間でのピア間のアクセスを測定し，特定に用いることが可能である．

4.2 特定方式 2

図 4 に複雑なネットワークモデル 1 を示す．一般に P2P ユーザの中には，2 つのネットワーク間に存在するノード (a) のように 1 つのノードで複数の P2P アプリケーションを実行している場合がある．複雑なネットワークモデル 1 では，測定点を境に上側もしくは下側のみのネットワークにそのようなノードが存在するとする．それぞれのアプリケーションのピアは同一のアプリケーションのネットワークに参加しているピアとのみアクセスがある．しかし，ノード (a) を適切に扱わない場合，ノード (a) 上で実行しているピアとアクセスしている他の P2P ネットワークに所属しているピアも Winny ピアと判断してしまう誤検知の原因となる．よって，ノード (a) を下記の手順では誤検知が起きないように取り扱う．

最も簡単な方法はノード (a) との通信のすべてをないもの（実際にはログから削除）とし，P2P ネットワークを図 3 のように 2 つに分割する方法がある．この操作により特定方式 1 が使えるようになる．しかし，複数のサービスポートを持つノードの Winny のサービスポートを特定することができない．

そこで特定方式 2 では，ノード (a) のピアとその他のピアとのアクセス関係をさらに検討する．ノード (a) の Winny のサービスポートは Winny ピアだけからアクセスされ，ノード (a) の Winny はそのアクセスしてきたピア F にアクセスし返す．つまり，双方向クライアント/サーバ通信モデルに着目すると，図 4 のような，上側もしくは下側だけに複数のサービスポートを持つノードがあったとしても，通信する相手のピアのサービスポートは 1 つであり，間違って他の P2P のピアを Winny と誤検知することはないことが分かる．つまり，ノード (a) を境に 2 つの P2P ネットワークを分けることが可能となる．

4.3 特定方式 3

図 5 に複雑なネットワークモデル 2 を示す．このモデルは複雑なネットワークモデル 1 より現実的なモデルで，1 つのノードに複数の P2P アプリケーションを実行している場合があり，測定点をはさんで上下どちらにも存在する．

複雑なネットワークモデル 1 では，双方向クライアント/サーバ通信の関係はそれぞれのネットワーク内で閉じていた．しかし，2 つ以上の P2P アプリケーションを実行しているノードどうしが通信した場合，双方向クライアント/サーバ関係を用いても，ノード (a) とノード (b) との間では，サービスポートがアクセスされたら，アクセスし返すという関係がそれぞれの P2P ネットワーク内で閉じておらず，誤検知の原因となる．具体的には，図 5 のピア F とノード (a) の Winny のサービスポートの間で双方向クライアント/サーバ通信モデルが確立されることで，ノード (a) の Winny のサービスポートが特定される．しかし，ノード (a) の Winny のサービスポートに対してアクセスしたノード (b) に対しては，ノード (b) の Winny のサービスポートだけではなく，“その他の P2P アプリケーションのサービスポート” に対しても双方向クライアント/サーバ通信モデルが確立する．この時点で，ノード (b) ではどちらのサービスポートが Winny のものかが分からなくなってしまう．つまり，双方向クライアント/サーバ関係が確立するピア間をたどることにより，“Winny を含めたその他の P2P アプリケーションを実行しているノード” を特定することができるが，サービスポートがどちらが Winny のものかどうか区別することができなくなるという問題が起きる（ノード (a) でも同様に区別がつかなくなる）．さらに，この Winny と他の P2P アプリケーションを実行しているノードを介することにより，双方向クライアント/サーバ通信モデルをたどることで，Winny 以外の

アプリケーションのサービスポートを Winny としてしまう可能性がある．よって，複数の P2P アプリケーションを実行しているノードどうしが通信し合う状況を適切に取り扱い，誤検知をなくす必要がある．

図 4 の複雑なネットワークモデル 1 のように，測定地点の上側もしくは下側のみに複数の P2P アプリケーションを実行しているノードが存在する状況を作り出し，双方向クライアント/サーバ通信モデルに着目することで，Winny トラヒックが特定可能になる．この状況を作るために上側（ノード (a)）もしくは下側（ノード (b)）のネットワークに所属する複数のサービスポートを持つすべてのノードと他のピアとの通信を解析の際に取り除き，1 つのノードで複数の P2P アプリケーションを実行していたとしても，その通信する相手のノードでは 1 つの P2P アプリケーションしか実行していない状況にする．よって，双方向クライアント/サーバ通信モデルを用いた特定方法は，複雑なネットワークモデル 2 のような状況でも有効といえる．

4.4 提案する特定方式の手順

4.1〜4.3 節で議論した Winny トラヒック特定方式をまとめると以下のようになる．

ステップ 1：まず，Winny の囮ピアを用意する．囮ピアは Winny ネットワークに参加し，他の Winny ピアの IP アドレスとサービスポート番号を集める．よって，この段階で図 3〜5 のうち，少なくとも 1 つの Winny ピアのサービスポートと IP アドレスを特定する必要がある．Winny ネットワークに参加しているすべてのピアのサービスポート番号と IP アドレスを特定することは難しいので，この段階で特定した Winny ピアのサービスポート番号と IP アドレスをもとにして，次のステップからピアを特定していく．

ステップ 2：図 5 の場合を考え，1 つのノードに複数のサービスポート番号を持つノードを特定する．特定の際には，低い誤検知確率を目指すため，双方向クライアント/サーバ通信モデル（図 1 (b)）を用いて特定する．このステップでは図 5 のピア (a) とピア (b) の間のような状況を介して Winny を含めた複数の P2P ネットワークを特定することになる．

ステップ 3：ステップ 2 の段階で，複数の P2P アプリケーションを実行しているノードを特定することができたので，特定方式 3 における測定地点をはさんで片側だけ複数のサービスポートを持つノードの通信が特定可能となる．つまり，図 4 のような測定地点の上側もしくは下側の複数のサービスポートを持つノードとの通信をすべて取り除いた状況を作り，下側もしくは上側の複数のサービスポートを持つノードで実行されている Winny ピアを特定する．この手順は上側，下側で並列に行いその結果を合わせることによって，複数のサービスポートを持つノード上で実行されるピアのサービスポートを含めてすべての Winny ピアのサービスポートが特定される．

4.5 提案方式の適用範囲に関する検討

提案方式を適用する際には以下のような条件がある．

(1) 測定は Winny の囮ピアの地点とトラヒック測定点（図 3〜5 での測定点）の 2 カ所が必要になる．さらに囮ピアの通信がトラヒック測定点でのログに直接記録される場合か，Winny の囮ピアが直接特定したピアとの通信がトラヒック測定点でのログに含まれなければならない．つまり，同一時間帯において 2 つの測定点で測定が行われている必要がある．

(2) 特定手順の中でもととなるピア，つまり，Winny の囮ピアに直接特定されたピアのうちの少なくとも 1 つのピアは，同一ノードでは他の P2P アプリケーション等が実行されていない必要がある．直接特定されたすべてのピアと同一ノード上で他の P2P アプリケーションが実行されていた場合，双方向クライアント/サーバ通信モデルをたどることにより誤検知が発生してしまい，Winny のサービスポートを正確に特定することができなくなる．

(3) 特定できるのは，トラヒック測定点を通過する Winny ピアの IP アドレスとサービスポートであり，その他の地点については双方向クライアント/サーバ通信モデルが確立されているか不明のため，適用することができない．同じ理由により，トラヒック測定点では IP と TCP 層の双方向のログが必要となる．

(4) Winny では必ず，双方向クライアント/サーバ通信モデルが確立されるように設計されているが，何かの原因で確立される割合が低くなると特定精度が低下する場合がある．あるピアと双方向クライアント/サーバ通信モデルが確立されなかったピアは，その他のピアと双方向クライアント/サーバモデルが確立されなければ特定できない．

(5) 特定できる割合はサービスポートが 1 つのピアの割合に依存する．それは，サービスポートを持つアプリケーションを 2 つ以上実行させているピアを特定するためには，必ず，サービスポートが 1 つのピアと双方向クライアント/サーバ通信モデルが確立されなければならない

ためである．具体的には，図 5 において，測定地点でノード (b) との通信がノード (a) としか行われない場合，どちらのノードもサービスポートを 2 つ持っており，ノード (b) の Winny のサービスポートを特定することができない．ノード (b) の Winny のサービスポートを特定するためには，1 つのノードで Winny のサービスポートしか持たないピア A–C のうちどれかと双方向クライアント/サーバ通信モデルが確立される必要がある．

(6) ピア間のアクセス頻度が著しく少ない場合，あるピアのサービスポートを特定することができたとしても，双方向クライアント/サーバ通信モデルをたどることでピアを次々と特定することができない．Winny のピアが短い時間で多くのピアとの通信がある必要がある．そして，測定地点では多くの Winny ピア間の双方向クライアント/サーバ通信モデルを含まなければならない．

これらの提案方式の適用範囲により，提案方式が有効であるためには，ピア間で多くのアクセスがあり，さらに 100%に近い確率で双方向クライアント/サーバ通信モデルが確立され，1 つのノードで Winny のほかに P2P アプリケーション等のサービスポートを用意するアプリケーションを実行しているノードの数の割合が少ないことが特定精度の大きな低下につながらないための条件となる．条件 (1)–(3) は測定する際の環境設定により解決が可能であるので，条件 (4)–(6) が特定精度の大きな低下につながっていないことを次章の評価実験にて示す．

5. 提案する Winny トラヒック特定方式の評価

4 章までの議論を実際にインターネット上の P2P ピアとの通信を解析することにより，確認する．まず，実際に双方向クライアント/サーバ通信モデルが確立されていることを確かめた後，提案するトラヒック特定方式の精度を評価する．

5.1 トラヒック測定環境 1

トラヒック測定環境 1（**図 6**）を設定し，2007 年 5 月に採取した 48 時間のログを解析した．研究室内に 5 台の PC を用意し，PC A（TCP：サービスポート番号 15001），PC B–D（TCP：サービスポート番号 10001）に Winny を実行させ，PC B–E（TCP：サービスポート番号 20001）に Share を実行させた．PC B，C，D の 3 台では 2 つの P2P アプリケーション

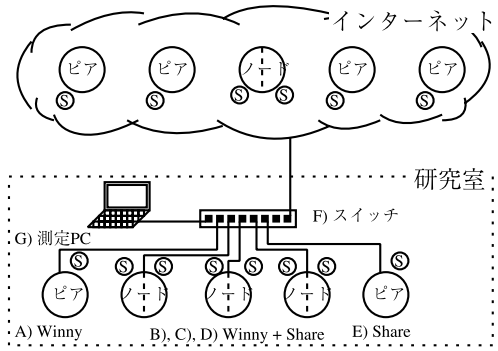


図 6 トラヒック測定環境 1
Fig. 6 Traffic measurement setup 1.

が実行されており，2 つの P2P ネットワークにアクセスすることになる．Winny トラヒックを特定する際，PC A のピアのサービスポート番号（15001）が囮ピアによってわかっているとして，Winny トラヒックを特定していく．(4.5 節 条件 (2)) しかし，インターネット上の Winny ピアのサービスポートへの接続トラヒック，特に PC B，C，D ではどのトラヒックが Winny トラヒックなのかを特定することは難しい．研究室側で用意した実験モデルは小さいが，前節で述べたモデルのすべてを含み，かつ，それぞれのピアは非常に多くのインターネット上のピアと通信を行うため，インターネット側のピアでより現実的な状況が実現される．

提案方式でのアプリケーションの特定に用いるのはインターネットと研究室間の通信であり，双方向の通信をスイッチ F でポートミラーリングを用いて測定し，測定 PC で保存した（4.5 節 条件 (1)，(3)）．提案特定方式の精度をシグニチャマッチング等の他の方式と比較評価するのは難しい．そこで，アプリケーションがどのポートを使ったかを PC A–E で記録する．このログにより，誤検知率，未検知率の正確な値を評価することが可能となる．実際，アプリケーションレベルのパケット情報をリバースエンジニアリングを用いて解析し，アプリケーションを特定する方式でも，誤検知，未検知が存在する[7]．具体的な設定を下記に示す．

測定 PC の設定：TCP，IP ヘッダ情報のみ Snort 2.4[17] を用いて測定し，アプリケーション情報は収集しなかった．

PC の設定：Winny のバージョンは Winny 2β7.1 を用い，クラスタワード，ファイル検索キーワードを設定せずに実行した．Share は version EX2 を用い，クラスタワードを半角スペース，“ ”，とし，ファイル検索キーワードを設定せずに実行した．これらファ

表 1 研究室内の PC からフローが確立されたユニークなサービスポートの数（48 時間）
Table 1 Number of service ports in the Internet accessed over 48 hours by PCs.

	Winny	Share
From PC A	21,855	-
From PC B	22,353	10,719
From PC C	22,017	11,218
From PC D	22,554	10,686
From PC E	-	10,272

イル共有が起こらない設定をしてもそれぞれの P2P ネットワークを維持するためにピア間のアクセスは頻繁に発生する．すべてのコネクション状況（どのポートがアプリケーションに割り当てられたか）は，Port Reporter [18] で記録した．これにより，どのコネクションがどのアプリケーションによって用いられたかを把握することができる．

違法なファイル交換を防ぐために，Winny，Share ともにファイルをキャッシュ，ダウンロード，アップロードするフォルダを Winny もしくは Share を実行しているプロセスがアクセス権限を持たない場所に設定することで，ファイルのやりとりがいっさい起こらない設定とした．実際にはファイルの転送要求等が他のピアからあるが，すべてその試みは失敗することになる．Winny，Share ともにファイルの交換および，ファイルの検索に用いるサービスポートは同一のものを用いるため，そのサービスポート番号を特定するときにこの設定でも問題は生じない．

5.2 双方向クライアント/サーバ通信モデルの検証

本節では双方向クライアント/サーバ通信モデルの検証をする．まず，**表 1** に研究室内の PC からフローが確立されたインターネット上のユニークなサービスポートの数を PC 別に示す．フローは TCP の SYN と FIN にはさまれ，同一の送信元 IP アドレス，ポート番号と送信先 IP アドレス，ポート番号（TCP），および，プロトコル番号を持つものとした．フローは測定 PC のログにより特定され，フローのアプリケーションは測定 PC の Snort のログとそれぞれの PC の PortReporter のログを組み合わせることにより，確認した．どちらのアプリケーションもピュア P2P ネットワークを構築し，ネットワークを維持するために 1 つのピアが数多くのピアにアクセスすることが分かる．この多くのアクセスにより，提案する方式では容易に次々ピアを見つけることができる（4.5 節 条件 (6)）．

図 7 に TCP レベルでの PC A の Winny ピアがインターネット上の Winny ピアのサービスポートにアクセスして，PC A の Winny ピアのサービスポートにアクセスし返されるまでの時間の累積分布を示す．Share については，PC E のものを示す．ログの解析期間は 48 時間の内の前半の 24 時間である．本章での “アクセス” はサービスポートから SYN–ACK が返された場合とする．それは SYN–ACK が返されたということは，TCP コネクションが確立されたことを意味するからである．

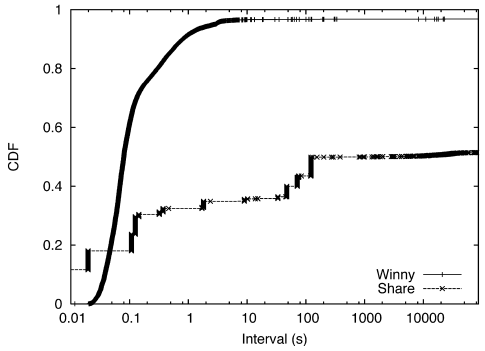


図 7 双方向クライアント/サーバ関係が確立されるまでの時間（累積分布）
Fig. 7 A time period of establishment of Two-way Client/Server communication relations (CDF).

双方向クライアント/サーバ通信モデルが確立されるのは，PC A のピアのサービスポートにアクセスがあった後に PC A の Winny ピアがアクセスし返す場合とその逆の場合がある．よって，PC A のピアがインターネット上のピアにアクセスした前後に PC A にアクセスがあるので，この期間の絶対値で “サービスポートにアクセスし返されるまでの時間” を定義とした．つまり，実際に双方向クライアント/サーバ通信モデルが確立されるまでどの程度の時間がかかっているかを示している．同様に，Share の場合は PC E のアクセスを用いた．

Winny では双方向クライアント/サーバ通信モデルが確立されるように設計されている，つまり，Winny のサービスポートにアクセスしてきた場合，必ずアクセスしてきた Winny ピアのサービスポートに対してアクセスをし返す．よって，100%アクセスし返されるはずであるが，TCP レベルでは 3.2%のアクセスが 24 時間たってもアクセスし返されない．これは，Winny アプリケーションが多くのコネクションを保持し，処理をしきれない場合にアクセスを返さないことがあるためと考えられる．しかし，10 秒以内に 96.3%がアクセスし返されている．提案する特定方式では，この割合は重要であり，Winny ピアのサービスポートを特定できた場合，97%程度までの確率でアクセスしてきたピアのサービスポートを特定できる可能性がある

表 2 解析単位別平均誤検知率と平均未検知率（PC A–E の Winny）
Table 2 Average False negative ratio and Average False positive ratio (Winny in PC A–E).

解析単位	0.5 時間	1 時間	3 時間
平均未検知率	0.055	0.054	0.053
平均誤検知率	0	0	0

ことを意味する（4.5 節 条件 (4)）．

一方，Share では 48%のピアが 24 時間以内にアクセスし返されず，そのままでは本提案方式は有効ではないことが分かる．

5.3 提案特定方式の評価

次に 4.4 節で提案したトラヒック特定方式を評価する．PC A の Winny ピアのサービスポートが囮ピアによって，まず，与えられているとして，Winny のサービスポートの特定を開始する（4.4 節の手順 1）．

まず，4.4 節の手順 2 を用いて 1 つのノードに 2 つ以上のサービスポート（Winny と他のアプリケーション）を持つピアを，48 時間のログを 24 時間ずつに分けて特定する．この手順により，インターネット上に最初の 24 時間に 904，次の 24 時間に 814，48 時間に延べ 1,573 ノードが 2 つ以上のサービスポートを持つことが分かった．提案方式でこれらのノードの扱いがうまくいっていない場合は，誤検知が大きくなることになる．

4.4 節の手順 3 では，24 時間のログの期間を 0.5，1，3 時間ごとに区切り，それぞれの期間で特定方式を実行し，平均を求めた．**表 2** に誤検知率と未検知率のそれぞれの解析単位での平均を示す．提案方式で 48 時間の延べ数で 59,098–59,116 のユニークな Winny ピアのサービスポートを誤検知なしで特定することができた．よって，1 つのノードに複数の P2P アプリケーションが実行されているような場合でも誤検知は 1/59,116 以下であったことがいえる．それぞれのログ解析単位でそれほど大きな差がない．これは双方向クライアント/ サーバ通信モデルがログ解析単位と比較して短い時間で確立されるためである．

5.2 節で述べたように双方向クライアント/サーバ通信モデルが最大で 97%の割合で確立し，最も良い場合では，100% − 97% = 3%程度の平均未検知率が期待される．しかし，平均未検知率が 5.3–5.5%であり，期待された 3%よりも 2%大きな値となっている．これは，インターネット上に 2 つ以上サービスポートを持つノードが 1,573 あったためで，これらのピアが 2 つの P2P アプリケーションを実行している PC B–D だけと通信を行った場合，たとえ，双方向クライアント/サーバ通信モデルが確立されていても，特定することができない．これらのノード上で実行されている Winny ピアのサービスポートを特定するためには，1 つのノードで Winny アプリケーションが 1 つ実行されているピア（研究室内の PC A）にアクセスされなければならないためである．PC A だけでは解析単位内にそれらのピアすべてに対してアクセスすることは難しく，未検知が予想された値よりも上昇している．つまり，本提案方式の特定精度は，1 つのノードに 1 つの Winny ピアが実行されている割合に依存しているが，複数の P2P アプリケーションを実行しているユーザがそれほどいないため（2.5%程度），特定精度の大きな低下は起きなかった（4.5 節 条件 (5)）．

表 3 特定したポート番号の有効期間の評価．(PC A–E の Winny における平均誤検知率と平均未検知率)
Table 3 Average False negative ratio and average False positive ratio with extension of the identification period (Winny in PC A–E).

解析単位	6 時間	12 時間	24 時間	48 時間
平均未検知率	0.047	0.050	0.049	0.041
平均誤検知率	0.000089	0.00029	0.00029	0.00059
平均誤検知ピア数	1.5	5.75	10	35

5.4 特定したポート番号の有効期間の評価

5.3 節では，Winny ピアのサービスポートを特定するために解析単位を区切り，それぞれの解析単位内で Winny ピアを特定して精度を評価した．本節ではある解析単位で特定された Winny ピアが他の期間でどの程度有効かを評価する．まず，1 時間ごとに Winny ピアのサービスポートを特定する．どの程度の間，特定された Winny ピアのサービスポートが有効であるかを示すために 1 時間ごとに特定された Winny ピアのサービスポートを 6，12，24，48 時間ごとにまとめ，まとめた 6，12，24，48 時間ごとで平均未検知率，平均誤検知率を求めた．

解析結果を**表 3** に示す．まとめた期間によって平均未検知率に変化はないが，誤検知が発生している．これは，Winny ユーザがサービスポートを変えずに，Share を実行したため，異なるアプリケーションでも同一のサービスポートが用いられているためである．それは，多くのユーザの PC には，パーソナルファイアウォールが実行されており，2 つの P2P アプリケーションを使い分けたいユーザが，ファイアウォールの設定を変更せずに，異なる時間帯で異なるアプリケーションを同一サービスポート番号で実行したためであると考えられる．

5.5 NAT 内のピアの影響の評価

NAT の中に複数の Winny ピアがおり，1 つのノー

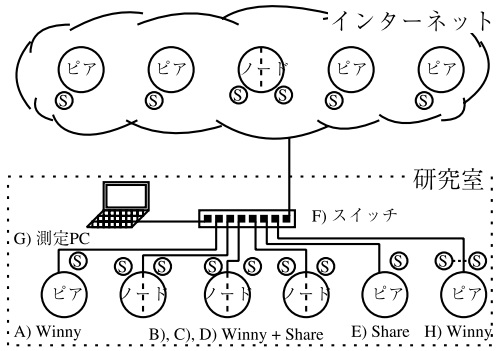


図 8 トラヒック測定環境 2
Fig. 8 Traffic measurement setup.

表 4 NAT 内のピアの影響を含めた平均誤検知率と平均未検知率（PC A–E, H の Winny）
Table 4 Average False negative ratio and Average False positive ratio with NATed Peer (Winny in PC A–E, H).

	ピア A–E, H	ピア H
未検知率	0.099	0.116
誤検知率	0	0

ドに多数の Winny のサービスポートが場合がある．NAT 内のピアが提案する特定方式に影響があるか評価をする．トラヒック測定環境 2 を**図 8** に示す．トラヒック測定環境 1 にノード H を加え，NAT はピア H とスイッチの間に設置した．ノード H に Winny ピアを 10 台実行し，インターネット上のピアと通信をさせた．5.3 節同様，ログの解析期間は 48 時間，解析の単位は 1 時間で行った．解析結果を**表 4** に示す．誤検知は起きないが，表 2 と比較すると未検知が増加している．これは，NAT 内のピアは，インターネット上のサービスポートが 1 つの Winny ピアと双方向クライアントサーバ通信モデルを確立しなければならず，さらに通常のプロトコルの処理である，アクセスされたら，アクセスし返すという処理が短時間に行えず，双方向クライアント/サーバ通信モデルが確立される割合が低下したためと考えられる．しかし，NAT 内に複数の Winny ピアが存在した場合でも，誤検知なしに 88.4%の NAT 内の Winny ピアがインターネットにアクセスしにいく Winny ピアの IP アドレスとサービスポートの組を特定することが可能であった．

6. ま と め

ピュア P2P アプリケーションである Winny のトラヒック特定方式を提案し，特定精度の検討を行った．Winny ピアのサービスポートを特定するとピア間のアクセス関係，つまり，双方向クライアント/サーバ関係をたどることによって，次々に Winny ピアのサービスポートを特定可能であることを示した．本提案方式により，アプリケーションレベルの通信が暗号化されていたとしても，トラヒックの状況，ユーザ動向を把握する手がかりを，トランスポート層以下のログだけでつかむことが可能である．インターネット上の Winny ピアとの接続による評価実験により，提案する特定方式の未検知率は 0.053–0.116 である一方，誤検知率を 10^{-4} 程度におさえることが可能であることを示した．そして，双方向クライアント/サーバ通信モデルが確立されるまでの時間は短く，このことを本特定方式に特定条件として加えることにより，さらなる特定精度の向上が考えられる．

今後，本提案方式を用いて，ISP の Winny トラヒックを特定し解析することで，これまで明らかにされていなかった Winny トラヒックの実態を明らかにし，他の P2P アプリケーショントラヒックについても解析を行う予定である．

謝辞 本研究の一部は科研費基盤 C 課題番号 18500047，電気通信普及財団の援助を受けており，ここに記して謝意を表します．最後に多数の有益なコメントをくださった査読者の方々に感謝いたします．

参 考 文 献

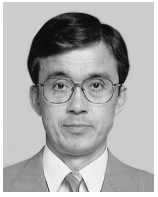
1) Sen, S. and Wang, J.: Analyzing Peer-To-Peer Traffic Across Large Networks, *IEEE/ACM Trans. Networking*, Vol.12, No.2, pp.219–232 (2004).
2) Plissonneau, L., Costeux, J. and Brown, P.: Analysis of Peer-to-Peer Traffic on ADSL, *Proc. PAM 2005*, pp.69–82 (2005).
3) Stutzbach, D. and Rejaie, R.: Understanding Churn in Peer-to-Peer Networks, *Proc. ACM IMC'06*, pp.189–201 (2006).
4) Gnutella（オンライン）(参照 2007-06-17). http://www.gnutella.com
5) Kazaa（オンライン）(参照 2007-06-17). http://www.kazaa.com
6) BitTorrent Protocol（オンライン）(参照 2007-06-17). http://www.bittorrent.com
7) Sen, S., Spatscheck, O. and Wang, D.: Accurate, Scalable In-Network Identification of P2P Traffic Using Application Signatures, *Proc. ACM WWW '04* (2004).
8) Karagiannis, T., Broido, A., Brownlee, N., Claffy, K. and Faloutsos, M.: Is P2P dying or just hiding?, *Proc. IEEE Globecom 2004*, pp.1532–1538 (2004).

9) 金子　勇：Winny の技術，*ASCII* (2005).
10) Share（オンライン）(参照 2007-06-17). http://en.wikipedia.org/wiki/Share_%28p2p%29
11) Karagiannis, T., Broido, A., Faloutsos, M. and Claffy, K.: Transport Layer Identification of P2P Traffic, *Proc. ACM IMC '04*, pp.121–134 (2004).
12) Perenyi, M., Dang, T.D., Gefferth, A. and Molnar, S.: Identification and Analysis of Peer-to-Peer Traffic, *Journal of Communications*, Vol.1, No.7, pp.36–46 (2006).
13) Constantinou, F. and Mavrommatis, P.: Identifying Known and Unknown Peer-to-Peer Traffic, *Proc. 5th IEEE International Symposium on Network Computing and Applications*, pp.93–102 (2006).
14) eMule（オンライン）(参照 2007-06-17). http://www.emule-project.net
15) WinMX（オンライン）(参照 2007-06-17). http://www.winmxworld.com
16) Clarke, I., et al.: Freenet: A Distributed Anonymous Information Storage and Retrieval System, *Proc. ICSI Workshop on Design Issues in Anonymity and Unobservability*, LNCS 2009, pp.46–66, Springer-Verlag (2001).
17) Snort（オンライン）(参照 2007-06-17). http://www.snort.org
18) Port Reporter（オンライン）(参照 2007-06-17). http://support.microsoft.com/kb/837243/

(平成 19 年 6 月 18 日受付)
(平成 19 年 11 月 6 日採録)

大坐畠　智（正会員）

昭和 49 年生．平成 15 年筑波大学大学院博士課程工学研究科電子・情報工学専攻修了．同年東京農工大学工学部情報コミュニケーション工学科助手．平成 19 年同大学助教．ネットワーク性能評価の研究に従事．平成 17 年度電子情報通信学会学術奨励賞受賞．IEEE，ACM，IEICE 各会員．博士（工学）．



川島幸之助（正会員）

昭和 21 年生．昭和 44 年東京大学工学部計数工学科卒業．同年日本電信電話公社入社，研究所所属．通信トラヒック研究部長等を歴任．平成 9 年 NTT アドバンステクノロジ株式会社入社．平成 14 年東京農工大学工学部教授，現在に至る．主に，通信トラヒック工学，通信網，システム性能評価の研究・開発・教育に従事．主な著書として，『最新 コンピュータネットワーク技術の基礎』（共著，電気通信協会），『情報通信トラヒック—基礎と応用』（共著，電気通信協会）．昭和 56 年電子通信学会論文賞（共著），昭和 61 年日本オペレーションズ・リサーチ学会文献賞，平成 19 年同学会業績賞等受賞．日本オペレーションズ・リサーチ学会，電子情報通信学会各フェロー．IEEE，ACM SIGCOMM，IFIP WG6.3 各会員．博士（工学）．