

無線機器間の直接的な接続指示手法

岩崎 陽平[†] 河口 信夫^{*‡} 稲垣 康善[†]

[†] 名古屋大学大学院工学研究科

[‡] 名古屋大学情報連携基盤センター

^{*} 名古屋大学総合音響情報研究拠点

iwasaki@inagaki.nuie.nagoya-u.ac.jp

Bluetooth や無線 LAN などの短距離無線通信技術の普及により、様々な機器が互いに無線接続可能な環境が整いつつある。しかし、無線機器間の接続の際には、一般にアドレスや名前の指定などの煩雑な設定が必要となる。本稿では、無線機器間の接続を、接続したい両機器のボタンを押すだけで直接的に指示できる手法である Touch-and-Connect を提案する。本手法では、状態を表示可能なボタンを使用し、複数人間が独立に操作を行う状況においても誤接続を防止する。ブロードキャスト通信を用いた本手法のプロトコルは、管理サーバを必要とせず、動的な端末の入退出にも対応するため、必要に応じて一時的に構築されるアドホックネットワークでも利用可能である。また、グループの導入によりセキュリティと使いやすさの向上を行っている。

Intuitive Connecting Method for Wireless Devices

Yohei Iwasaki[†] Nobuo Kawaguchi^{*‡} Yasuyoshi Inagaki[†]

[†] Graduate School of Engineering, Nagoya University

[‡] Information Technology Center, Nagoya University

^{*} Center for Integrated Acoustic Information Research (CIAIR), Nagoya University

The spread of the short distance wireless communication technology such as Bluetooth and wireless Ethernet is making it possible for various information appliances to communicate through the wireless connection. However, to connect and to use them, we are generally required complicated setup operations such as setting addresses or names. In this paper, we propose an intuitive connecting method for wireless devices called “Touch-and-Connect”. With this method, we can connect any wireless appliances by only button pushing operations. This system is designed so that it can work without any server, and even if many persons operate it independently, any incorrect connection does not occur. In order to improve security and usability, a user can make the group that devices can join in.

1 はじめに

Bluetooth[1]や無線 LAN[2] などの短距離無線通信技術の普及により、様々な情報機器が互いに無線接続可能な環境が整いつつある。無線通信を用いれば、ケーブルに制限されずに自由に機器を移動することが可能となる。また、無線通信モジュールの低価格化や小型化により、パソコンの周辺機器をはじめとして、家電製品、AV 機器、文房具などの様々な機器が無線ネットワークに参加し、相互に通信可能となる。

無線ネットワーク上の機器を連携させるためには、機器間を特定のプロトコルにより論理的に接続する必要がある。この際に、ユーザが意図する接続をシステム側に伝えること（接続指示）が必要であるが、一般に、接続指示のためには煩雑な設定をしなければならない。

例えば、Bluetooth に対応した機器間の接続指示には、Digianswer 社の Bluetooth Neighborhood [3] が広く使われている。このソフトウェアは、アドレスや名前によって無線ネットワーク上の機器の一覧を表示する。接続指示を行う際には、この中から接続したい機器を選択する必要があるが、機器のアドレスを調べる、多数の機器に名前をつけて覚える、といった作業

は煩わしく感じられる。

従って、アドレスや名前の指定などの設定をしなくてすむような、接続指示のためのより簡単で分かりやすいユーザインタフェースが求められる。本稿では、接続指示手法が満たすべき性質を示し、それを指針として接続指示手法 Touch-and-Connect を提案する。

2 無線ネットワーク環境における接続指示

本稿では、Bluetooth や無線 LAN などの短距離無線通信で構成される無線アドホックネットワーク環境を想定する。ただし、マルチホップのメッセージは想定せず、端末間は直接通信可能とする。

アドホックネットワークとは、必要に応じて一時的に構築されるネットワークであり、(1)管理サーバや事前の設定を前提としない、(2)端末の入退出によりネットワーク構成が動的に変化する、などの特徴がある。アドホックネットワークでも利用可能とすることにより、いつでも、どこでも、提案する接続指示手法を使うことができる。例えば、外出先でモバイル機器間を接続する場合や、事前に設定がないノート PC 間を接続するような場合にも利用可能となる。

2.1 接続指示に関連する問題点

無線ネットワーク環境における接続指示に関連して、

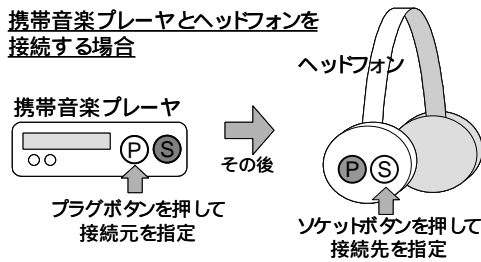


図 1 インタフェースの使用法

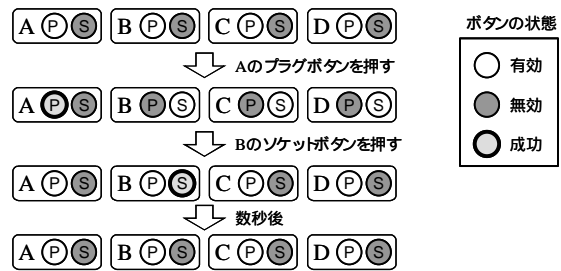


図 2 ボタンの状態遷移例

以下のような問題点が考えられる。

- 無線接続では、有線での接続のように、ケーブルを操作して接続を変更したり、機器間の接続状況を目で見て直観的に理解することができない。
- 一般に、無線機器間の接続指示を行う際には、機器のアドレス・名前の指定や、接続方法（プロトコル）の選択などの煩雑な設定が必要になる。特に、機器の画面表示や入力インタフェースが制限されている場合は、このような設定による接続指示は困難である。
- 多数の無線機器がネットワークに参加するような場合には、他の機器操作との干渉が発生する。
- 無線環境では機器間に障害物（壁など）があっても通信が可能のため、悪意のある他人により、機器所有者の意図しない接続が行われることが考えられる。この結果、非公開の情報が盗まれるなどの問題が発生する。

2.2 接続指示手法が満たすべき性質

2.1 節の問題点を考慮すると、接続指示のためのインタフェースは以下のような性質を満たすことが望ましい。

- (1) 直接的な接続指示：機器そのものを直接指定して接続指示を行えるべきである。アドレスや名前の指定は直観的ではなく分かりにくい。
- (2) 接続の実現が容易：機器間を接続する際に、接続方法の選択などの設定が必要なのは望ましくない。
- (3) 接続状態の可視化：無線機器間の接続状態を可視化し、接続の確認ができることが望ましい。また、これを実現するためのシステムは、以下のような性質を満たすことが望ましい。
- (4) サーバの不利用：アドホックネットワーク環境でも利用可能とするため、管理サーバの存在を前提とするべきではない。
- (5) スケーラビリティ：多数の無線機器や多数の人間が存在する環境を考慮する必要がある。
- (6) セキュリティの考慮：悪意のある他人による意図しない接続を防止できることが望ましい。

3 接続指示手法 Touch-and-Connect

無線機器間の接続指示手法として、Touch-and-Connect を提案する。本手法は、2.2 節に挙げた性質を指針としており、接続したい両機器のボタンを押すだけで無線機器間を容易に接続することができる。機器のボタンを直接押す必要があるため、比

較的近く（手の届く範囲）にある無線機器間の接続に使われることを前提とする。

本手法では、図 1 のように、各機器にプラグボタンとソケットボタンを設ける。なお、図中ではプラグボタンを P、ソケットボタンを S で表す。機器 A（例えば携帯音楽プレーヤ）と機器 B（例えばヘッドホン）を接続したい場合には、まず、機器 A のプラグボタンを押すことによってそれが接続元機器であることを指示し、その後、機器 B のソケットボタンを押すことによってそれが接続先機器であることを指示する。プラグをソケットに差し込むという比喻により、ユーザはこの操作を容易に理解することができる。

また、各ボタンは「有効」、「無効」、「成功」のいずれかの状態を表示できるようにする。ボタンは「有効」の時に押すことができ、押した後、「成功」になったことを確認する必要がある。これは、接続指示の成功を表示すると共に、複数の人間が独立に接続操作を行う場合に発生しうる誤接続を防止するために用いられる。

複数の人間が独立に操作を行う場合、システム側はどの人間がボタンを押したのかを知ることができないため、誤接続が発生する可能性がある。例えば以下のような順番でボタンが押された場合を考える。

- (1) ユーザ 1 が機器 A のプラグボタンを押す
- (2) ユーザ 2 が機器 C のプラグボタンを押す
- (3) ユーザ 2 が機器 D のソケットボタンを押す
- (4) ユーザ 1 が機器 B のソケットボタンを押す

この場合、ユーザ 1 は機器 A と B の、ユーザ 2 は C と D の接続を意図しているにもかかわらず、A と D、C と B が接続されてしまう可能性がある。

本手法では、接続操作中は他の機器のプラグボタンが「無効」となり（ロック）、他人の操作が禁止されるため、誤接続を防止することができる。先ほどの例の場合、ユーザ 1 の操作中は、A 以外の全機器のプラグボタンが「無効」となり、ユーザ 2 は操作を開始できない。この場合のボタンの状態遷移例を図 2 に示す。

3.1 機器情報

接続指示が行われる際に、接続可能性の判断、適切な接続方法の選択、および接続の実現を行うために、各機器は以下の情報を持つ。

- アドレス（Touch-and-Connect プロトコルの通信用）
- 機器の Type
- グループ ID
- 付加情報

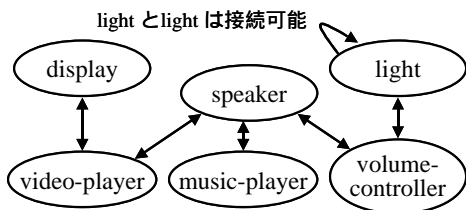


図 3 接続可能性の定義例

bidirectional-link: display, video-player
bidirectional-link: speaker, volume-controller
bidirectional-link: speaker, music-player
bidirectional-link: speaker, video-player
self-link: light

bidirectional-linkは双方向リンク, self-linkは同Type間のリンクを表す

図 4 接続可能性の記述

3.1.1 機器の Type

本手法は、様々な種類の接続に対して一般的に用いることができる。接続要求時に適切な接続方法を自動的に選択するために、各機器に Type(機器の種類)を定義する。ここでは単純な Type として、light (ライト)、video-player (ビデオプレーヤ)、music-player (音楽プレーヤ)、display (表示装置)、speaker (スピーカ)、volume-controller (ボリュームコントローラ)などを考える。

また、2つの Type 間の関係として接続可能性を定義する。機器 A から機器 B への接続要求(Aのプラグボタンを押し、Bのソケットボタンを押すこと)に意味がある場合、AからBへ接続可能とする。接続可能性の定義例を有向グラフに表したものを図3に示す。

接続可能性の定義により、接続操作を行う際に、接続可能な機器のソケットボタンのみを「有効」にし、また、必要最小限の機器のみをロックする。機器Aのプラグボタンが押された場合は、「機器Aから接続可能な機器」へ接続可能な機器全てをロックするべきである。例えば、先ほどの定義の場合、lightのプラグボタンが押された場合は、light, speaker, volume-controllerをロックするべきである。

Type間の接続可能性を判断できるようにするために、接続可能性を記述した図4のような設定ファイルを各機器に持たせる。

3.1.2 グループ ID

本手法では、プラグボタンを押した後、悪意のある他人にソケットボタンを押されることにより、意図しない接続が行われてしまう可能性がある。また、多数の接続が行われる場合には、ロックが解除されるまで長時間待たされる可能性がある。

これらに対応するため、接続可能な機器をグループにより制限することができる。各機器はグループIDを持ち、同じグループIDを持つ機器間のみが接続可能、およびロックが必要となる。機器は初期状態では「デフォルトグループ」に属する。(グループIDの設定のためのユーザインタフェースに関しては、3.5節を参照)

3.1.3 付加情報

付加情報は、機器の Type によって内容が定められる任意の情報であり、接続の実現に使用する。例えば light の場合、明るさを調整するためのプロトコルの通信アドレスである。

3.2 接続の実現方法

接続指示の完了後、接続される両機器に、接続相手の機器情報と、自分が接続元であるかどうか通知される。接続される機器は、この情報を元に、適切な接続方法の選択、および接続の実現を行う。自分が接続元かどうかの情報を通知することにより、接続指示の方向によって、接続方法を変更することが可能である。

接続方法の選択には、主に接続相手の Type を用いる。例えば、video-player は、display と接続する場合は映像伝送を行い、speaker と接続する場合は音声伝送を行う。本手法では具体的な接続方法については特に規定しない。例えば、接続指示の結果、既存のリンクが切断されたり、データを移動したりしてもかまわない。

3.3 その他の特徴

本手法のその他の特徴として、以下のようなものがある。

- プラグボタンを押して接続元を指定した後、再度プラグボタンを押すことにより、接続先の指定をキャンセルすることができる。
- ロックしたまま放置された場合に対応するため、プラグボタンを押してから一定時間経過後、自動的にキャンセルされる。

3.4 プロトコル

ブロードキャスト通信を用いた本手法のプロトコルは、管理サーバを必要とせず、動的な端末の入退出にも対応するため、アドホックネットワークでも利用可能である。

本手法のプロトコルを示す(図5)。時間パラメータとして request-wait, cancel-wait, success-time がある。また、各機器はリクエストリストとロックリスト(初期状態は空)を持つ。

リクエストおよびロック

- プラグボタンが押された機器(リクエスタ)は、Request メッセージをブロードキャストする。接続可能性を判断するために、このメッセージにはリクエスタの機器情報(付加情報を除く)が含まれる。
- Request メッセージを受信した機器は、それが自分へ接続可能な機器からのメッセージであれば、そのメッセージをリクエストリストに追加する(リクエスト)。また、それがロックされるべき機器からのメッセージであれば、そのメッセージをロックリストに追加する(ロック)。

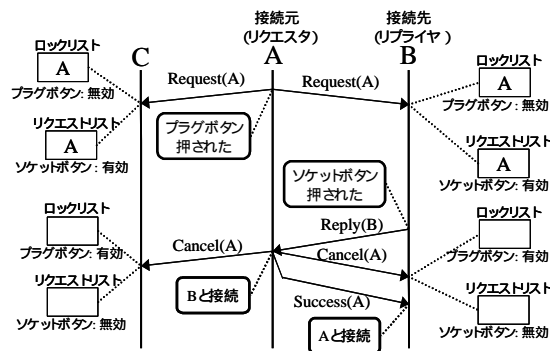


図 5 Touch-and-Connect プロトコル

リクエストリストとロックリストが分かれているのは、リクエストを行う範囲とロックを行う範囲が接続可能性により異なるためである。

リプライ

- ソケットボタンが押された機器(リプライヤ)は、リクエストリストの先頭要素の送信元に対して、Reply メッセージを送信する(リプライ)。

ロックが正しく行われていれば、リクエストリストの要素は1つである。パケットが届かなかった場合や、ネットワーク構成が動的に変化した場合などは、リクエストリストの要素が複数になる可能性がある。要素が複数の場合は、どれにリプライすればよいのかがあいまいとなり、誤接続が発生する可能性がある。

接続操作の完了

- リクエストが Reply メッセージを受信すると、リプライヤと接続することを認識する。リクエストおよびロックをキャンセルするために、Cancel メッセージをブロードキャストする(キャンセル)。また、接続することを通知するためにリプライヤへ Success メッセージを送信する。
- Cancel メッセージを受信した機器は、リクエストリストおよびロックリストから、送信元機器の要素を削除する。

Reply, Success メッセージには、送信元機器の機器情報が含まれている。この機器情報を元に、適切な接続方法を選択し、接続を実現する(3.2 節参照)。

キャンセル

- リクエストは、リクエスト後にプラグボタンが押された場合は、キャンセルする(手動キャンセル)
- ロックしたまま放置された場合に対応するため、リクエストは、リクエスト後 cancel-wait 経過した場合は、自動的にキャンセルする。

プラグボタンの同時押しへの対応

複数の機器のプラグボタンが同時に押された場合への対策として、機器にはアドレスにより優先順位が決められている。

- リクエストは、リクエスト後 request-wait 以内に、より優先する機器からロックされた場合は、キャンセルする。

ボタンの状態表示

リクエストリストが空の場合、ソケットボタンは「無効」となり、要素がある場合「有効」となる。ロックリストが空の場合、プラグボタンは「有効」となり、要素がある場合「無効」となる。ボタンが無効の場合は、押すことはできない(押されても無視する)。

接続元指定の成功を表示するために、リクエストは、リクエスト後 request-wait 経過してからキャンセルするまでの間、プラグボタンが「成功」となる。また、接続先指定の成功を表示するために、リプライヤは、Success メッセージを受信してから success-time の間、ソケットボタンが「成功」となる。

リクエストリストの要素が複数の場合は、そのままソケットボタンを押すと誤接続が発生する可能性があるため、何らかの表示でユーザに警告することが望ましい。

メッセージのリピート

動的な端末の入退やパケットの損失に対応するため、Request メッセージは、有効時間を決め、送信を繰り返す。また、Cancel メッセージも、数回送信する。

有効リクエストのキャッシュ

各機器は、ネットワーク上で現在有効になっている全リクエストの情報を記憶しておく。この情報は、以下の目的で使用する。

- ネットワークトラフィックの増加を抑えるために、ネットワーク上で現在有効なリクエストの数が増加するほど、メッセージのリピート間隔を長くする。
- 動的に機器情報を変更する場合、例えば、一時的にグループ ID を変更する場合(3.5.2 節参照)に、メッセージのリピート間隔を待たずにすばやくボタンを適切な状態にする。

時間パラメータのデフォルト値

時間パラメータのデフォルト値は、request-wait=300ms, cancel-wait=30000ms, success-time=2000ms とする。

3.5 グループ設定のためのインタフェース

機器の属するグループを変更したい場合は、機器のグループ ID(3.1.2 節参照)を変更する必要がある。グループ ID を設定するためのユーザインタフェースとしては、ボタンなどにより直接 ID を入力する方法の他に、より簡単な方法としてグループチェンジャの利用が考えられる。また、一時的なグループ ID の変更も考えられる。

3.5.1 グループチェンジャ(グループ変更器)

グループチェンジャは、グループ ID を変更する目的で使用する特別な機器である。各グループチェンジャは「設定用 ID」を持っている。設定したい機器からグループチェンジャへの接続指示を行うと、グループチェンジャの設定用 ID が接続元機器のグループ ID として設定される。従って、グループ ID を入力することなしにグループ ID の設定を行うことができる。

グループチェンジャの設定用 ID は、初期状態でユニークな値が割り当てられている。何らかの UI を設

けてこの設定用 ID を変更可能としてもよい。

接続指示の方向

グループチェンジャの使用においては、接続指示の方向が重要である。グループチェンジャから設定したい機器への接続指示を行うと、グループチェンジャのプラグボタンを押した後、他人に他の機器のソケットボタンを押されることにより、他人の機器を自分のグループに参加させてしまう可能性がある。逆に、設定したい機器からグループチェンジャへの接続指示を行うと、同様の理由により、自分の機器が他人のグループへ参加させられてしまう可能性がある。

グループ設定は主に機器の新規購入時に行われると考えられるため、新規購入した機器が他人のグループへ参加させられることは、他人の機器が自分のグループへ侵入することに比べれば危険性は低い。従って、任意の機器からグループチェンジャへの接続可能性のみを許可し、逆方向は接続不可能とする。

デフォルトグループ限定

グループチェンジャ自体は、デフォルトグループに属するため、デフォルトグループに属する機器のグループ設定のみを行える。属するグループを変更しなくなった場合のため、グループに参加している機器に何らかの操作（例えば、リセットボタンを押す）を行うことにより、グループの解除（デフォルトグループへ戻る）が行えることが望ましい。

任意の機器からグループチェンジャへの接続を可能にすると、3.1.1 節の定義より、接続指示を行う際に全ての Type の機器をロックしなければならない。この影響を最小限にするために、グループチェンジャへの接続可能性は、デフォルトグループのみで有効とする。すなわち、デフォルトグループとそれ以外のグループで、接続可能性を別々に定義する。

3.5.2 一時的なグループ ID の変更

簡易的な ID 入力手段（例えば、数個のボタンを複数回押す）を設け、一時的なグループ ID の変更を可能とすることができる。元のグループ ID と入力された ID を組み合わせたものを一時的なグループ ID とし、接続指示が完了した後に元のグループ ID に戻す。

これにより、ボタンがロックされている場合でも、接続したい両機器で同じ ID を入力して接続指示を行うことができる。セキュリティを向上させるためにこれを用いることもできる。

3.6 ボタン 1 つ版インタフェース

ボタンの状態表示を工夫することにより、各機器に対してボタン 1 つのみで本手法を実現することができる。従来の 2 つのボタンの状態を、1 つのボタンで表現するために、ボタンに、接続元指定、接続先指定成功、接続先指定、接続先指定成功、無効、の 5 つの状態を設ける。ユーザは、まず、接続元指定状態のボタンを押して接続元機器を指示し、ボタンが接続元指定成功状態になったことを確認する。その後、同様に接続先指定状態のボタンを押して接続先機器を指示する。

ボタンの状態変化と動作は、表 1 のルールに従う。ただし、リプライ可能となった後、ChangeModeWait（デフォルト値は 1000ms）経過するまでの間は、無

表 1 ボタン 1 つ版インタフェースの動作

条件	ボタン状態	押された時の動作
リクエスト成功 (リクエスト中)	接続元指定成功	キャンセル
リプライ成功 (接続確定後一定時間)	接続先指定成功	
リプライ可能	接続先指定	リプライ
ロックされている	無効	
デフォルト (リクエスト可能)	接続元指定	リクエスト

より上のルールが優先される

効状態（操作不可能）とし、その後に接続先指定状態とする。これは、複数のユーザがほぼ同時に操作を開始するような場合に、他人の操作によりボタンが接続先指定状態になったのに気づかずに操作を行ってしまい、誤接続が発生するのを防止するためである。

ボタン 2 つ版と比較すると、ボタン 1 つ版は、インタフェースの部品コストや実装面積を削減できるという利点があるが、誤接続が発生する可能性が高くなるという欠点がある。

3.7 満たしている性質

本手法は、2.2 節に挙げた性質のうち、以下の性質を満たしている。

機器のボタンを押すだけで直接的な接続指示が可能であるため、(1)を満たす。機器の Type により自動的に接続方法が選択されるため、(2)を満たす。本手法のプロトコルは管理サーバを必要としないため、(4)を満たす。(5)に関しては、状態表示可能なボタンによる各ユーザの操作間の排他制御、グループによるロック範囲の制限、リクエスト数に応じたメッセージリピート間隔の調整など、多数の無線機器が存在する環境の考慮を行っている。(6)に関しては、グループによる接続可能性の制限により、セキュリティの向上を行っている。(3)の接続状態の可視化は今後の課題である。

4 プロトタイプシステムの実装

本手法に基づくプロトタイプシステムを Java 上に実装した。端末間の通信には無線 LAN を使用し、通信プロトコルの実装を容易に行うために、簡易メッセージ通信ライブラリを作成した。

インタフェースを内蔵した機器として、図 6 のような機器の、ソフトウェアによるエミュレータ、およびハードウェアを作成した。上下ボタンによる操作が可能なボリュームコントローラを、対象物（ライト、扇風機、またはリリーススイッチ）に接続し、明るさ、風速、電源スイッチなどを制御することができる。また、コントローラの対象物同士を接続することにより、ボリューム値の変化（明るさの変化など）を連動させることができる。また、任意の機器をグループチェンジャへ接続することにより、容易にグループ ID の設定を行うことができる。

Touch-and-Connect の接続指示ボタンの状態は色によって表示した。ボタン 2 つ版のインタフェースで

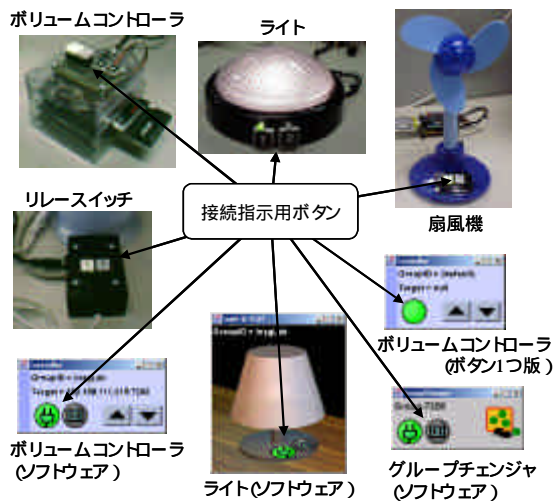


図 6 インタフェースを内蔵した機器

は、緑が有効、赤が成功、黒（消灯）が無効を表す。ボタン1つ版のインタフェースでは、緑が接続元、赤が接続先を表し、点滅により成功を表す。

4.1 ハードウェアの制御

ハードウェアの制御にはワンチップマイコン H8/3664F を使い、ファームウェアとしてテキストベースの対話式モニタを実装した Java で書かれた制御用ソフトウェアを組み込み用小型 PC (LAMB-EM-01) 上で動作させ、シリアル通信を使って対話式モニタと通信を行い、マイコンを制御する。

5 関連研究

本研究に関連する研究・規格としては、以下のようなもの挙げられる。

ネットワーク上の機器間を接続するための技術仕様として、Universal Plug and Play[4]、Jini[5]、HAVi[6]、ECHONET[7]などが提案されている。これらを用いれば、機器の機能をサービスとしてネットワーク上で公開し、相互に機能を提供しあうことができる。しかし、接続指示のためのユーザインタフェースについてはあまり重視されておらず、アドレスや名前の指定などの設定が必要であることが多い。

STONE[10]、AMIDEN アーキテクチャ[8]では、機器の種類等により、ネットワーク上から自動的に適切な機器を検索する。また、Follow-me application [15]は、ユーザの位置を超音波を用いたセンサにより取得し、自動的にユーザに最も近いディスプレイが作業環境として選択される。しかし、多数の機器が存在する環境では自動決定には限界があり、Touch-and-Connectのようにユーザが望んでいる機器を直接指示できる手段も必要となるだろう。

Pick-and-Drop[9]は、端末の画面をペンでタップすることにより、コンピュータ間をまたがるデータの移動を直接的に行うことができる手法である。GUI オブジェクトのコンテキストメニューにボタンインタフェースを表示することによって、Touch-and-Connectを同様の目的に使うことができる。Pick-and-Drop はコ

ニークな ID を持ったペンや管理サーバを必要とするが、Touch-and-Connect はこれらを必要としない。

AirReal[11]は、レーザポインタを内蔵したリモコンで機器を指し示すことにより、様々な指示を行うことができる。特に、複数の機器を指し示すことによって、それらの機器間を接続し連携させることができる。機器に接触する必要がないため、遠く離れた機器を指定するために AirReal は有用である。しかし、事前に管理 PC に機器の位置を入力しておく必要があるため、Touch-and-Connect が対象としているような、事前の設定を前提としないアドホックネットワーク環境での利用や、移動する可能性のある小型機器を対象とした利用は困難である。

ユーザの人影をカメラで上面より撮影し、画像処理を行うことにより、ユーザが腕で指し示している方向を入力する手法[16]が提案されている。これを応用すれば、特別なデバイスを持たなくても、腕で指し示すことにより遠く離れた機器を指定可能なシステムが実現できる。しかし、腕の上下方向（ピッチ角）の角度入力が行えないため、多数の機器が存在する場合には詳細な指定が困難である。

Resurrecting Duckling[12][13]は、無線アドホックネットワーク環境における情報機器のためのセキュリティモデルである。機器を新規購入した際にその機器を所有者のものとして Imprinting(刷り込み)し、以後その機器は他人の機器との通信が制限される。グループチェンジャによるグループ ID の指定(3.5.1 節参照)は Imprinting と類似しており、グループチェンジャに秘密情報を持たせることにより、同様の役割を果たすことができる。

6 おわりに

無線機器間の接続指示手法として、ボタンを押すだけで容易に接続の指示が行える Touch-and-Connect を提案した。またプロトタイプシステムの実装によりその実現可能性を示した。

本手法は、複数の人間が独立に操作を行う状況においても誤接続を防止する。また、管理サーバの不利用や、動的な端末の入退場への対応により、アドホックネットワークでも利用可能である。

6.1 今後の課題

本手法に関連した今後の研究課題としては、以下のようなもの挙げられる。

- 評価実験：インタフェースの使用法の習得は容易か、誤接続は発生しないか、ロック解除までにどの程度待たされるか、ボタン2つ版か1つ版かどちらが使いやすいかなどを、実験により評価する必要がある。
- 拡張性のある機器間連携フレームワーク：本手法では、機器間の接続可能性や通信プロトコルをあらかじめ決めておく必要があり、拡張性が十分ではない。cogma[14]などのモバイルエージェントシステムと連携し、ネットワーク機器間の連携のための拡張性のあるフレームワークを実現する手法に関する研究を進めている。
- 接続状態の可視化：機器間の現在の接続状態を確認できることが望まれる。このために、ネットワ

ーク上に存在する機器と、機器間の連携状態を視覚化する GUI (ネットワークビュー) を実現する方法を検討している。GUI 上のシンボルと実際の機器との対応を分かりやすく表現するために、機器の位置情報の利用を考えている。

- 遠隔機器の指定手法：本手法は機器のボタンを押す必要があるため、遠く離れた機器に対しての利用は困難である。指示デバイスで指し示すことにより、遠く離れた接続先機器を指定することが可能な、実世界指向インタフェースを実現する方法に関する研究を進めている。実現手段として、地磁気センサ、加速度センサ等を組み合わせ、デバイスの方向を測定することが可能な方向センサの使用を検討している。

参考文献

- [1] The Official Bluetooth SIG Website. <http://www.bluetooth.com/>
- [2] Wireless Ethernet Compatibility Alliance (WECA). <http://www.wi-fi.org/>
- [3] Bluetooth Software Suite. <http://www.btsws.com/>
- [4] Microsoft Corporation. Understanding Universal Plug and Play : A White paper. <http://www.upnp.org/resources/default.asp>
- [5] Sun Microsystems Inc. Jini[tm] Technology Architectural Overview. <http://www.sun.com/jini/whitepapers/architecture.html>
- [6] HAVi: Home Audio Video Interoperability. <http://www.havi.org/>
- [7] ECHONET 規格書 Ver.1.0. http://www.echonet.gr.jp/8_kikaku/spec_v10.htm
- [8] Minoh,M and Kamae,T. "Networked Appliance and Their Peer to Peer Architecture AM IDEN". IEEE Communications Magazine, Vol.39, No.10. pp.80-84 (2001).
- [9] 暦本純一. "Pick-and-Drop:複数コンピュータ環境での直接操作技法". WISS'97. pp.141-150 (1997).
- [10] 澤島康仁,杉田馨,南正輝,森川博之,青山友紀. "ネットワークサービスシンセサイザにおける機能接続・管理機構の設計と評価". マルチメディア,分散,協調とモバイル(DICOMO 2001)シンポジウム論文集. pp.103-108 (2001).
- [11] 星野剛史,堀井洋一,丸山幸伸,片山淳詞,柴田吉隆,吉丸卓志. "AirReal: ホームネットワークのユーザーインタフェース". WISS 2001. pp.113-118 (2001).
- [12] Frank Stajano and Ross Anderson. "The Resurrecting Duckling: Security Issues for Ad-hoc Wireless Networks". Proceedings of 3rd AT&T Software Symposium. (1999).
- [13] Frank Stajano. "The Resurrecting Duckling - what next?". <http://www-lce.eng.cam.ac.uk/~fms27/duckling/>
- [14] 河口信夫, 稲垣康善. "cogma: 動的ネットワーク環境における組み込み機器間の連携用ミドルウェア". コンピュータシステム・シンポジウム論文集. pp.1-8 (2001).
- [15] Andy Harter, Andy Hopper, Pete Steggles, Andy Ward and Paul Webster. "The Anatomy of Context-Aware Application". MOBICOM'99. pp.59-68. (1999).
- [16] 木原 民雄. "実写映像の多人数操作による情報ナビゲーションシステム". マルチメディア,分散,協調とモバイル(DICOMO 2002)シンポジウム論文集. pp.9-12. (2002).