

MaCC: 通信状況に応じた WPAN コンフィグレーションシステムの提案

滝澤 允[†] 間 博人[‡] 斉藤 匡人[‡] 戸辺 義人[§] 徳田 英幸^{†‡}

近年, 新たな無線ネットワークである WPAN(Wireless Personal Area Networks) のネットワーク・トポロジ構築に関するプロトコル/アルゴリズムの研究がなされている。しかし, 従来の研究はネットワーク・トポロジに重点を置いたネットワーク構築手順を中心としており, WPAN における通信品質を考慮していない。そこで, 本稿では WPAN における通信品質を考慮した動的にネットワーク編成を行う MaCC(Master-driven Connection Control) システムを提案する。MaCC は, ノード数やノードの位置が頻繁に変動する WPAN での通信に対して, 最少ホップ数の経路を提供する。本稿では, MaCC の設計と実装, 今後の展望と課題を述べる。

MaCC: A WPAN Configuration System based on Communication Quality

Makoto Takizawa[†] Hiroto Aida[‡] Masato Saito[‡]
Yoshito Tobe[§] Hideyuki Tokuda^{†‡}

In Wireless Personal Area Networks(WPAN), there are many research results on the WPAN network formation protocols and algorithms. However, these work concentrate on establishment procedures, not taking into account the communication quality in WPAN. In this paper, we present Master-driven Connection Control(MaCC) system, which provides the minimum hop-count routes for each communication dynamically when communications occur between nodes. We describe the design and implementation of MaCC, the problems to solve, and future work.

1 はじめに

無線通信技術の発展により, WPAN(Wireless Personal Area Network) という無線ネットワークが登場し注目を集めている。WPAN はノート PC や携帯電話, PDA に加え, 通信機能を有した家電や各種センサ等が形成する半径十数メートル程の小規模なネットワークである。有線ネットワークにおける物理的な制約が少なく容易に構成できる。

現在 WPAN の通信媒体として近距離, 低消費電力の無線通信規格である Bluetooth [10] が有力である。しかし现阶段では, Bluetooth は WPAN に対して有効なネットワーク構成機能やルーティング管理機能を提供していない。場合によっては, ネットワークが効率良く構成されず, 通信効率が著しく悪化する可能性がある。

そこで Bluetooth による WPAN ネットワーク構成機能に関する研究が多く行われている。例えば, TSF(Tree Scatternet Formation) [9] と呼ばれる木構造ネットワークを構成するもの, またスキヤットネット構成を 1 台のノードが決定する BTCP(Bluetooth Topology Construction Protocol) [8], スキヤットネットを構成するピコネット数を

最小限に抑えるフォーメーション・プロトコル [3] [4] 等がある。

従来の研究は構成するネットワーク・トポロジに焦点を置き, トポロジ構築手順を研究内容の中心としている。これらはネットワーク内の通信品質を考慮したネットワーク・トポロジではない。従来の研究に見られる簡潔なネットワーク・トポロジは必ずしも通信品質を向上させるものではない。

そこで, 本稿ではネットワーク構成機能に加え, WPAN における通信品質を考慮した MaCC(Master-driven Connection Control) を提案する。MaCC は, 各ノードが効率的にネットワーク全体の状況を把握し, その情報に基づいた最適な判断を可能とする。ネットワーク全体の状況の情報を効率良く集中管理できるためのネットワーク構成を実現し, 同時に, 動的に通信ノード間への最短ホップ数経路の提供を実現する。通信経路にある中継ノードは利用可能な通信帯域が減少する。MaCC は各通信の他ノードへの影響を最小限に抑えるために, 各通信に対する中継ノード数を最小限にする。

本稿では, 第 2 節で MaCC の背景を述べ, 第 3 節で MaCC が有する機能と設計を詳説する。第 4 節では MaCC の実装に関して述べる。第 5 節において MaCC に対する考察と課題を述べ, 最後に第 6 節で本稿のまとめを述べる。

2 背景

本節で MaCC において背景となる Bluetooth 仕様について述べる。MaCC が用いる Bluetooth ネットワークとその特徴を述べ, 本稿が定義するリレーションという状態の

[†] 慶應義塾大学 環境情報学部

[‡] 慶應義塾大学大学院 政策・メディア研究科

[§] 東京電機大学 工学部 情報メディア学科

[†] Faculty of Environmental Information, Keio University

[‡] Graduate School of Media and Governance, Keio University

[§] Department of Information Systems and Multimedia Design, Tokyo Denki University

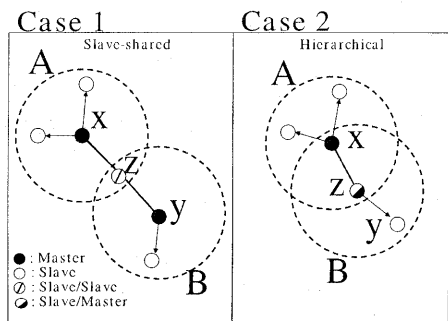


図 1: Bluetooth スキャットネット (Case 1:スレーブ共有型 Case 2:階層型)

背景知識として、Bluetooth 搭載端末同士がコンネクションを確立し通信状態へと至るリンク管理層の仕様に関して述べる。

2.1 Bluetooth スキャットネット

Bluetooth は 1 台のマスターに最大 7 台のスレーブからなるピコネットと呼ばれるネットワークを形成し、また複数ピコネットから成るスキャットネットを形成する。形成されたスキャットネットはピコネット間接続形態により 2 種類に分けることができる。それはスレーブ共有型と階層型である。それぞれのスキャットネットを図 1 に示す。Case 1 がスレーブ共有型スキャットネットである。ノード x と y が各々のピコネット A, B のマスターであり、ピコネット A, B 両方のスレーブとしてノード z を共有してピコネット A, B が相互接続する。Case 2 は階層型スキャットネットである。ここでは、ノード z がピコネット A のスレーブであると同時にピコネット B のマスターである。Case 2 では、あるピコネットのマスターが他のピコネットのスレーブとなつてスキャットネットが構成されている。

本稿で提案する MaCC はスレーブ共有型スキャットネットおよび階層型スキャットネットを状況に応じて使い分ける。簡単にスレーブ共有型および階層型スキャットネットの特徴を示す。スレーブ共有型はスキャットネットを構成するピコネット数が比較的少ない。一方、階層型スキャットネットはネットワーク全体でデータを集中させる場合に有効である。

2.2 Bluetooth リンク管理層

Bluetooth はノード間で通信状態へと遷移するまでに「問い合わせ (Inquiry)」および「呼び出し (Page)」と呼ばれる手続きを行い、ノード間で互いに情報交換を行う。「問い合わせ」のシーケンスを図 2 に示す。「問い合わせ」手続きでは、ノードが「問い合わせ」要求をブロードキャストし、自身の通信範囲内に存在するデバイスを検知する。一方、「問い合わせ」要求を受信したノードは自身の存在を通知するパケットを送る。要求を行う側がマスターとなり、IQ パケットという「問い合わせ」要求パケットをブロードキャストする。それを受信したスレーブは FHS(Frequency Hop Synchronization) パケットと呼ばれる応答パケットを、IQ パケット送信元であるマスターに返信する。FHS パケットは内部情報の 1 つとして Bluetooth アドレスを含む。よってマスターノードは「問い合わせ」により通信範囲内に存在するスレーブの Bluetooth アドレスを知ることができる。

「呼び出し」のシーケンスを図 3 に示す。「呼び出し」には 3 つのステップがある。(1) マスターが ID パケットを特定のスレーブへ送信し、受信したスレーブはそれと同一の ID パケットをマスターへ返信する。(2) マスターは (1) の過程でスレーブが送信した ID パケットを受信し、FHS パケットをスレーブに送信する。(3) FHS パケットを受信したスレーブは受信確認として ID パケットを返信する。「呼び出し」手続き完了後、接続手続きを行い通信状態へ遷移する。

FHS パケットは Bluetooth アドレスを通知する。よって「問い合わせ」および「呼び出し」手続きが完了した時点でマスターとスレーブは互いの Bluetooth アドレスを得る。本稿では新たにリレーションという状態を定義する。リレーションは FHS パケットにより、マスターがスレーブの、かつスレーブがマスターの Bluetooth アドレスを得た状態であると定義する。リレーションもマスター/スレーブ関係にあるが、接続手続きは完了しておらず通信はできない。

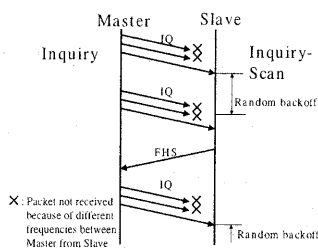


図 2: 問い合わせ (Inquiry) 手続きのシーケンス

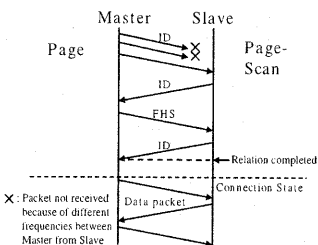


図 3: 呼び出し (Page) 手続きのシーケンス

3 MaCC

本節は MaCC について詳述する。MaCC は家庭内で構築される WPAN を想定している。デスクトップ PC, プリンタ, ノート PC, 携帯, 家庭用電話器など従来の通信機器に加え、テレビ, ビデオデッキ, 各種オーディオ機器がネットワークを構成するノードである。そこで MaCC の前提条件を、現在のインターネットのような莫大なノード数から成る大規模ネットワークではなく、30 台前後の小規模ネットワークとする。

また、家庭内ネットワークを想定することで、頻繁にコンネクションが切り替わる移動は考えない。比較的狭い小規模ネットワークであり、各ノードが他ノードの通信範囲を頻繁に出入りするような移動は起こらないとする。しかし、通信ノード数の変動は想定する。従来のような通信機器の種類の制限がなくなり今後も通信機能を有した機器の増加が予想される。よって家庭内ネットワークにおいても通信

ノード数は従来のインターネットに比べ頻繁に変動すると想定される。また各機器の電源オン/オフによるネットワーク内ノード数の変動も十分考えられる。

以上のような前提条件を満たす WPAN に MaCC を適用する。MaCC は以下の 3 機能を有する。以下の小節において、下記 3 つの機能設計を詳説する。

1. リレーションネットワーク構成機能: ネットワークを木構造となるネットワーク・トポロジで構築する。リレーションネットワークは本稿で導入した用語であり、3.1 節で詳述する。
2. 最短経路検索機能: ある 2 点間で通信要求が発生した時点で、その 2 点を結ぶ最短ホップ数経路を検索し、検索結果である最短経路を与える
3. コネクション確立機能: 検索結果である最短経路を繋ぐコネクションを確立する。

3.1 リレーションネットワーク構成機能

MaCC はネットワーク・トポロジを図 1 の Case 1 の階層型を利用し、木構造として構築する。木構造はグラフの 1 種であり、以降、構築されたネットワーク・トポロジにグラフという用語を用いる。構築されたグラフ例を図 4 に示す。図 4 のグラフは 9 つのノードが構築したネットワーク・トポロジを表し、各頂点がノードを表す。またノードを結ぶ実線と破線が辺を表す。

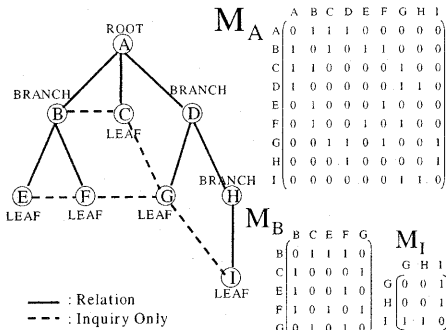


図 4: ネットワーク・トポロジのグラフによる表現と行列による表現例

図 4 のグラフで、実線で結ばれたグラフは木構造である。本稿では、MaCC により構成された木構造ネットワーク・トポロジを有するネットワークをリレーションネットワークと定義する。リレーションネットワークはコネクションではなく、リレーションの確立で構成されたネットワークである。すなわち、図 4 の実線によるグラフの辺はリレーションを表す。例えば A, B 間では、A は B の Bluetooth アドレスを保持し、その逆もまた成り立つ。破線で描かれた辺は「問い合わせ」手続きにおいて互いに通信範囲内に存在していることを示す。

MaCC は TSF のアルゴリズムを利用し、木構造を構築する。表 1 は TSF の接続規則を表す。FREE は他のノードとリレーションを確立していない状態、ROOT は木構造の「根」にあたる状態、Non-ROOT は、木構造の末端である LEAF とそのいずれにも該当しない BRANCH の状態であ

る。図 4 のグラフ例では、(A) が ROOT であり、(C, E, F, G, I) が LEAF、(B, D, H) が BRANCH である。

図 5 が各ノードの状態遷移を表し、図 6 は状態遷移を示す疑似コードを表す。FREE から始まり、他のノードとリレーションを接続規則に従って確立し、そのリレーションにおけるマスタ/スレーブによって次の状態が決定する。すべてのノードが表 1 の規則に従って状態遷移することで閉路を含まないリレーションネットワークを構築する。

証明

木 T 内で閉路が存在しないことを示す。表 1 から、Non-ROOT 間はリレーション確立不可である。また単一の木には ROOT は 1 つである。よって T は閉路を含まない。

MaCC によるリレーションネットワークにおいて、頂点の数が $n_1 (\geq 1)$ の木を T_1 、また頂点の数が $n_2 (\geq 1)$ の木を T_2 とする。 T_1, T_2 は木構造であるから、各々の辺の数は $n_1 - 1, n_2 - 1$ である。この時、 T_1 と T_2 がリレーションを確立する場合を考える。 $n_1 = n_2 = 1$ の時、閉路が存在しないことは自明である。 $n_1 \geq 2$ か $n_2 \geq 2$ の場合、図 1 の規則により ROOT 間でリレーションを確立する。よって、 T_1 と T_2 がリレーションにより繋がった木を T_3 とすると、 T_3 の辺の数が、

$$T_3 \text{ の辺の数} = (n_1 - 1) + (n_2 - 1) + 1 = (n_1 + n_2) - 1$$

となる。よって、MaCC により構築されるリレーションネットワークは木構造である。(証明終り)

表 1 の規則を満たすために、TSF では「問い合わせ」手続きにおいて、Non-ROOT は要求パケットの送信は行わず、受信は行わない設定で、Non-ROOT 同士が接続することを防ぐ。これを MaCC では「呼び出し」手続きにおいて実現する。また、ROOT かそれ以外かを判別するために MaCC は TSF と同様の方法を用いる。Bluetooth 仕様で「問い合わせ」は 2 種類 (GIAC¹, DIAC²) 存在し、ROOT は DIAC の「問い合わせ」を行い、ROOT 同士のみのリレーション確立を保証する。

図 6 の 13, 19, 22 行目が、LEAF および BRANCH が「問い合わせ」要求パケットを受信するが、「呼び出し」要求パケットは受信しない設定を表す。10 行目は、ROOT が DIAC の「問い合わせ」を行う設定を示している。

MaCC はリレーションネットワーク構成以降も常に「問い合わせ」により通信範囲内のノードの検知、自身の存在の通知を行う。MaCC が「問い合わせ」によって得た情報については次小節で述べる。

表 1: TSF 接続規則: 1 は接続可, 0 は接続不可

State	ROOT	Non-ROOT	FREE
ROOT	1	0	0
Non-ROOT	0	0	1
FREE	0	1	1

¹ General Inquiry Access Code: すべてのノードで共有するアクセスコード

² Dedicated Inquiry Access Code: ノードをグループ化するために使用するアクセスコード

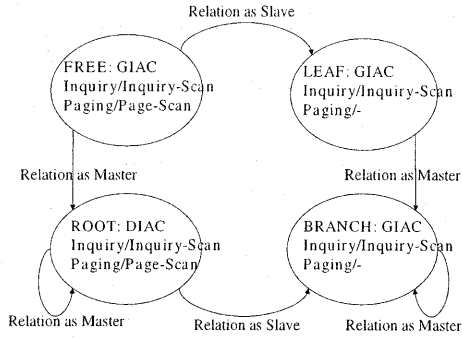


図 5: リレーション・フォーメーション時におけるノードの状態遷移図

3.2 最短経路検索

MaCCは最短経路検索機能を有し、構築したリレーションネットワーク内において、通信要求時に2点の通信ノード間を繋ぐ最少ホップ数経路を与える。

MaCCは経路検索に、ネットワーク構成を無向グラフとして行列で表現した情報を用いる。グラフを表現した行列では行および列は頂点を表し、各頂点が隣接³している場合、対応する要素は「1」、隣接していない場合には「0」である。

ここで、あるグラフを表した行列 M を k 乗した行列 M^k の各要素は2頂点間を繋ぐ長さ⁴ k の経路数を与える、という性質を利用する。頂点の数が n のグラフにおいて、閉路を含まない経路の最大長は $n-1$ となる。よって、以下の式。

$$C = M^1 + M^2 + M^3 + \dots + M^{n-1}$$

で与えられる行列 C は到達可能な2点間を示す。したがって、行列 M を最大(頂点1)乗することで、すべての到達可能な任意の2点を得る。

行列の情報はすべてのノードが保持する。しかし、すべてのノードが保持する行列は同一ではなく、階層化されて保持される。階層構造であるリレーションネットワークの ROOT を最上位層とし、LEAF を最下位層とする。各ノードは自身が属する階層以下の行列を保持する。具体的に図4では、ノードAがネットワーク全体を表す行列 M_A を、ノードBはノードB, E, Fによる行列 M_B 、ノードIはノードIのみの M_I を有する。単一ノードが有する行列は「問い合わせ」による情報である。すなわち行列は実線および破線の辺が示す隣接状況を表している。階層化された行列を保持するために、各ノードが有する行列を上位層へ、すなわち自身に対するマスタへ送信する。下位層から行列情報を受信したノードは自身の行列とマージする(図7)。

経路検索は、はじめに通信元が有する行列の情報から行い、検索失敗が起これると上位層に経路検索要求を行う。検索失敗が起これた後の上位層へ検索要求は、検索結果を得るまで再帰的に繰り返す。ROOT および FREE で検索失敗が起これるとネットワーク内で経路が存在しないと判断し、通信不能が通知される(図8)。経路検索の関連研究として、経路検索に MANET(Mobile Ad-hoc Networks) [5] で用い

³ グラフにおいて頂点 v_1, v_2 に対し、辺 $\{v_1, v_2\}$ が存在する時、頂点 v_1, v_2 は隣接している、という。

```

1: enable_scan_mode |=
    INQUIRY_SCAN | PAGE_SCAN;
2: inquiry_type = GIAC;
3: pseudo_function state_transition(role)
4:   role; # Master/Slave in Relation.
5: {
6:   switch (current_state) {
7:     FREE:
8:       if (role == MASTER)
9:         current_state = ROOT;
10:        inquiry_type = DIAC;
11:       else # role is SLAVE
12:         current_state = LEAF;
13:        enable_scan_mode |=
14:          (INQUIRY_SCAN & ~PAGE_SCAN);
15:     ROOT:
16:       if (role == MASTER)
17:         current_state = ROOT;
18:       else # role is SLAVE
19:         current_state = BRANCH;
20:        enable_scan_mode |=
21:          (INQUIRY_SCAN & ~PAGE_SCAN);
22:     LEAF:
23:       current_state = BRANCH;
24:       enable_scan_mode |=
25:         (INQUIRY_SCAN & ~PAGE_SCAN);
26:     default:
27:       ;
28:   }
29: }

```

図 6: 状態遷移を示す疑似コード

られるルーティングプロトコル(DSR [2], AODV [6])と同様にフラッディングにより経路検索を行うRVM(Routing Vector Method) [1]がある。しかし、フラッディングによる経路検索はネットワーク全体へ負荷を与えるという問題点がある。MaCCは経路検索要求を階層構造に従って行い、ネットワークへの負荷を抑えることができる。

3.3 コネクションセットアップ

第3の機能として、MaCCは経路検索結果である通信経路を確立する。図4のコネクションを確立していない実線および破線による無向グラフに対して通信経路に必要なコネクションのみ確立する。

図4のグラフにおいて、ノードB, H間で通信要求に対する経路検索結果が図9に示すように[B-C-G-I-H]である場合を例として以下説明する。

MaCCは経路検索結果を経路要求元のみへ送信する。この例では、要求元および通信元であるノードBへ通信経路が通知され、中継ノードC, G, Iへは経路情報を通知せずにソースルーティングを行う。送信元Bが経路情報(B-C-G-I-H)を含めたパケットを送信し、中継ノードC, G, Iはそれを元に転送先を決定する。送信先ノードHもパケット内の経路情報から判断し、受信処理を行う。

通信経路のコネクション確立とは、コネクション状態の

⁴ 長さとは到達可能な2頂点間を繋ぐ辺の本数である。

```

1: pseude_function hierarchize_matrix (matrix)
2:   matrix; # from lower node
3: {
4:   if (matrix != NULL)
5:     my_matrix = merge_matrix(matrix);
6:
7:   if (current_state == LEAF | BRANCH)
8:     send_to_master(my_matrix, master_addr);
9:   else
10:    ; # this node is ROOT or FREE
11: }

```

図 7: 行列を階層化させる部分の疑似コード

```

1: pseude_function search_route (src_dst)
2:   src_dst; #source and destination address
3: {
4:   route = search_path
5:     (route_request, my_matrix);
6:   if (route != NULL)
7:     return route;
8:   else {
9:     if (current_state == ROOT | FREE)
10:      return ERR_DISABLE;
11:     route = send_request
12:       (route_request, master_addr, ROUTE);
13:   }
14:   return route;
15: }

```

図 8: 再帰的経路検索を示す疑似コード

スキヤットネットワーク(ピコネット)を経路に適した構成にすることで、データ通信時の最初の1パケットがスキヤットネットワーク(ピコネット)構成機能を兼ねる。データ通信時のスキヤットネットワークは、まず送信元がスレーブとなり、以降マスタとスレーブを交互に割り当て、スレーブ共有型を構成する。図9では、ノードGがピコネット1, 2間を中継する共通スレーブとなる。背景の節で述べたように、スレーブ共有型と階層型ではスレーブ共有型の方が経路に含まれるピコネット数を少なく抑える。よって、ピコネット間転送のオーバーヘッドを最小限に抑えることができる。MaCCは通信終了後、コネクションを解除し各ノードをFREE状態へと初期化する。

以上がMaCCの有する機能である。MaCCはこの3つの機能に関してスキヤットネットワークの規模を限定しない。すなわち、単一ピコネットでも機能する。MaCCは、行列(隣接情報)を上位層へ送信し、リレーションネットに属するすべてのノードに関する隣接(行列)情報をROOTへ集め、集中管理を行う。しかし一方で各ノードがネットワーク状態に依存せず自律的に動作決定を行う。そのため、ネットワーク内のどのノードがダウンしたとしても、MaCCは機能停止しない。

4 Bluetooth を使用した MaCC の実装

本節では、MaCCの実装について述べる。実装環境は、Redhat 7.3J カーネル 2.4.18 に、BlueZ project [7] による bluez モジュールを加えたものを用いる。bluez は、C 言語

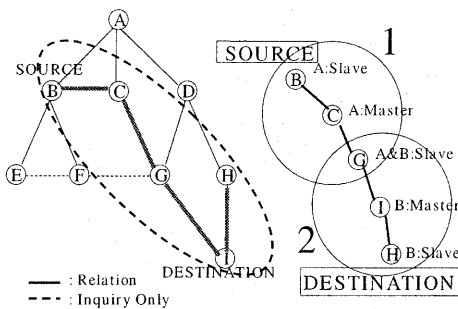


図 9: ある 2 点のノード間データ通信例

で書かれた公式の Linux 用 Bluetooth プロトコルスタックである。

Bluetooth 標準仕様は HCI(Host Controller Interface)を提供している。Bluetooth ベースバンド層へのインターフェイスである HCI が提供するコマンドによってベースバンド層を制御することができる。MaCC は HCI を利用し、ベースバンド層の制御を行う。

以下に、リレーションネット構成部の実装に関する内容、次に構成されたリレーションネット内で起こる通信の最短経路の検索部と検索結果である経路のコネクション確立部の実装について述べる。

4.1 リレーションネット構成部

MaCC では、すべてのノードが「問い合わせ」状態において、IQ パケットを送信しノードの存在を検知する一方、IQ パケットをスキャンして受信した IQ パケットに対して FHS パケットを返信する。よって IQ パケットの送信と IQ パケットのスキャンを一定の間隔で交互に行う。

各ノードが有する変数で主要なものを図 10 に示す。

```

/* Parameters */
# enable_inquiry_scan/enable_page_scan
enable_mode;
# Master address of the Piconet I belong to
master_addr;
# which two nodes are adjacent
matrix;

```

図 10: 状態遷移に用いる主要な変数

「問い合わせ」および「呼び出し」スキャンの有無を示す変数 enable_mode と自身のマスタのアドレスを示す変数 master_addr から、現在のノードの状態 (FREE, ROOT, LEAF, BRANCH) が判断でき、各ノードの状態を示す特定の変数は用意しない。

変数 enable_mode の値は、Bluetooth 仕様準じた、0x03 (「問い合わせ」、「呼び出し」要求に回答する) と 0x01 (「問い合わせ」要求にのみ応える) いずれかの値である。

ノードの状態遷移は「呼び出し」手続き完了直後に行う。しかし、HCI では「呼び出し」手続きが単独でなく、コネクション要求コマンドに含まれている。よってコネクシ

ン要求を表す HCI_Create_Connection コマンドを発行し、「呼び出し」手続き終了後、コネクション手続き時にコネクションを拒否して、その後状態遷移を行う。

4.2 最短経路検索部

MaCC の最短経路検索の実装は、経路を得る部分のみを述べる。行列の積では経路数が求まり、経路自体は求まらない。よって、行列の積の計算を必要な行と列に対して 1 ホップごとに一時的に保存しながら行い、最終的な経路を得る。

4.3 コネクションセットアップ

MaCC では、データ通信における最初の packets が 1 コネクションを確立する。パケットのデータ部であるペイロードボディの先頭に、SRC/DST/SR⁵ フィールドを追加する。パケットを受信したノードはまず DST フィールドを調べ、自ノード宛は上位層プロトコルへ送り、それ以外は SR フィールドに従って転送する。パケット処理の簡単な疑似コードを図 11 に示す。

```
1: pseudofunction receive (packet)
2: {
3:   if (DST == MY_ADDR) {
4:     raise_to_upper_proto(packet);
5:     if (connected(nextHop)) {
6:       # connected to nextHop
7:       forward(packet, nextHop);
8:     }
9:     else {
10:      # not connected yet
11:      connect(nextHop);
12:      forward(packet, nextHop);
13:    }
14:   return;
15: }
```

図 11: パケット処理表す疑似コード

5 考察と課題

本節では MaCC に対する考察と今後解決すべき課題を述べる。MaCC は WPAN において必須となる自律分散機能を有しながら、ネットワークでは集中管理機能を実現する。しかし、本稿では MaCC が対象とするネットワーク規模を限定している。確かに、現在家庭内で通信の対象となる機器数は最大でも 50 程度であると考えられる。しかし、今後も現在と同様である保証はない。センサネットワークも最近一般に認知されるに至り、家庭内で構築される可能性もある。WPAN においてノード数が増加する可能性があり、ノード数が莫大になった場合は MaCC はリレーションネットワークの階層が深くなり、経路検索要求に対する応答遅延の増加が予想される。本稿ではノード数を上限 30 台程度と定めたが、今後、ノード数が非常に多いネットワークへの対応が必要である。

また、MaCC は通信時間がファイル転送などある程度持続する通信を前提としている。例えば、2~3 パケットのみの通信に対しては最短経路検索は効率が悪い。通信時間の異なる通信への対応も今後の課題である。1 つの解決策として、プロトコル・サービス多重識別子 (PSM:Protocol

⁵ 送信元アドレス/送信先アドレス/通信経路

Service Multiplexer) により、Bluetooth プロトコルの上位層プロトコルを識別する方法が考えられる。

MaCC に限らず、Bluetooth スキャットネットはスケジューリングの問題を抱えている。スキャットネット内で複数ピコネットに属すノードは、ピコネット構成変換の処理が必要となる。ピコネット変換処理はオーバーヘッドとなるため、ピコネットの切り替えのスケジューリングが通信に大きく影響する。特に、マルチホップ通信への影響が大きいので、MaCC もスケジューリングに対して検討する必要がある。

6 まとめ

本稿では、Bluetooth による WPAN の通信に対し、最短ホップ数経路を提供し動的にスキャットネットを編成することで、ネットワーク全体で見た通信品質を向上させる MaCC を提案した。また MaCC が有するリレーションネットワーク構成機能、最短経路検索機能、コネクションセットアップ機能について述べた。MaCC は各ノードが自律的に動作する一方、ネットワーク内では集中管理機能を実現する。しかし、Bluetooth スキャットネットには解決すべき問題が多く、MaCC もまた十分ではない。本稿の考察において、依然として解決すべき問題点を提示し、今後対処すべき課題として示した。

参考文献

- [1] P. Bhagwat and A. Segall. A Routing Vector Method (RVM) for Routing in Bluetooth Scatternets. In *Bhagwat, P.; Segall, A. A routing vector method (RVM) for routing in Bluetooth scatternets 1999 IEEE International Workshop on Mobile Multimedia Communications (MoMuC'99)*, 1999.
- [2] Josh Broch, David B. Johnson, and David A. Maltz. The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks. IETF Internet-Draft [Work in Progress], March 2001.
- [3] Ching Law, Amar K. Mehta, and Kai-Yeung Siu. Performance of a new Bluetooth Scatternet Formation Protocol. In *Proceedings of the ACM Symposium on Mobile Ad Hoc Networking and Computing*, October 2001.
- [4] Ching Law and Kai-Yeung Siu. A Bluetooth Scatternet Formation Algorithm. In *Proceedings of the IEEE Symposium on Ad Hoc Wireless Networks*, November 2001.
- [5] Mobile Ad-hoc Networks Working Group (MANET). <http://www.ietf.org/html.charters/manet-charter.html>. Charter [Work in Progress], 2002-09-16.
- [6] Charles E. Perkins, Elizabeth M. Royer, and Samir R. Das. Ad Hoc On Demand Distance Vector (AODV) Routing. IETF Internet-Draft [Work in Progress], June 2002.
- [7] BlueZ project Official Linux Bluetooth protocol stack. <http://bluez.sourceforge.net/>.
- [8] Theodoros Salonidis, Pravin Bhagwat, Leandros Tassiulas, and Richard LaMaire. Distributed Topology Construction of Bluetooth Personal Area Networks. In *Proceedings of IEEE INFOCOM*, April 2001.
- [9] Godfrey Tan, Allen Miu John Guttag, and Hari Balakrishnan. Forming Scatternets from Bluetooth Personal Area Networks. Technical Report MIT-LCS-TR-826, MIT Laboratory for Computer Science, October 2001.
- [10] The Bluetooth Special Interest Group. <http://www.bluetooth.com>.