

漸増的に端末を認識するアドホックネットワーク構築手法の評価

戸崎貴資[†] 河口信夫^{†‡} 外山勝彦^{†‡} 稲垣康善[†]

近年の携帯情報端末の急速な普及により、携帯端末間で他の端末との情報の共有・交換を手軽に行いたいという要求が増えている。我々はすでに、携帯情報端末が出会ったその場で手軽にアドホックネットワークを構築することを目的とした、漸増的に端末を認識するアドホックネットワーク構築手法を提案している。本稿ではコンピュータシミュレーションによる本手法の性能評価について報告する。様々なネットワーク状況をモデル化してシミュレーションを行い、本手法の動作を検証した。また、本手法中で用いている各パラメータの値の変化に伴うネットワーク構築状況の変化をシミュレーションにより調査した。物理デバイスとして用いている赤外線通信(IrDA)の通信コネクション確立の時間的コストの高さを考慮した通信回数削減方法の有効性を確認した。

Performance Evaluation of Incremental Construction of Ad-Hoc Network

Takashi Tosaki[†] Nobuo Kawaguchi^{†‡} Katsuhiko Toyama^{†‡} Yasuyoshi Inagaki[†]

Recently, it becomes popular to use small size computers in a mobile environment and there are many demands to make a direct communication among mobile computers for exchanging or sharing their information. We have proposed an incremental construction method for ad-hoc networks based on infrared communication (IrDA). In this paper, we show a performance evaluation of this method by computer simulations. By using various simulation models, we simulate the construction of ad-hoc networks and evaluate the process. We also evaluate the influence of the algorithm parameters in several simulation models. By the evaluation we exemplify the validity of the proposed methods which reduce the traffic of the node information packets.

1 はじめに

近年、携帯電話やノートパソコン・PDAに代表される小型で高性能な携帯端末が普及し、いつでもどこでも必要な時に様々な情報の利用が可能になりつつある。より多くの人々が携帯端末を持つようになると、会議など多くの人々が集まる場合は、同時に多数の携帯端末が集まる場となる。そのような場で、携帯端末間でネットワークを構築し、端末に保存されている情報の交換・共有を直接行うことが可能となれば、これまで人が行ってきた共同作業の支援が可能になることが期待される。

このように、必要に応じて一時的に構築するネットワークはアドホックネットワークと呼ばれ、近年盛んに研究が行われている[5][6][7]。

我々は既に、アドホックネットワークを手軽に構築する手法として、漸増的に端末を認識するアドホックネットワーク構築手法を提案し、Java言語を用いて実装した[1]。本手法において、各端末は探索により近傍の端末を発見し、既知の端末とノード情報を交換することによりネットワークを時間の経過と共に漸増的に把握する。また、認識した端末とは即座に通信が可能である。

本手法は、具体的な通信デバイスとして赤外線通信(IrDA[2])を用いている。赤外線通信は、安価で小型なため、多くの携帯端末上に実装されており、これを用いることにより、特別な通信機器を必要とすることなく、各端末は手軽にアドホックネットワークを構築可能である。赤外線通信は通信範囲が狭く、帯域幅もそれほど大きくないため、本手法ではあまり大規模なネットワークの構築

を目的としておらず、会議等で構築する小規模なネットワークを対象とする。

本稿では、提案した手法のコンピュータシミュレーションによる評価実験について報告する。ネットワークポロジの動的変化の頻度とトラフィック量の異なるさまざまなシミュレーションモデルについてネットワーク構築のシミュレーションを行い、ネットワークを把握していく過程を調べ、どの程度の時間でネットワーク内の端末を把握できるかを求めた。本手法では赤外線通信のコネクション確立の時間的コストの高さを考慮し通信回数を削減する方式を提案していたが、今回のシミュレーションにより提案方式の有効性を確認した。また、数種類のネットワークモデルについて、本手法で用いているパラメータの値の変更がネットワーク構築に及ぼす影響についても調査した。

以下、提案した漸増的に端末を認識するアドホックネットワーク構築手法について2章で説明する。また、本手法のシミュレーションの方法について3章で述べ、実験結果と考察について4章で述べる。5章では関連研究について述べる。

2 漸増的に端末を認識するアドホックネットワーク構築手法

本章では、漸増的に端末を認識するアドホックネットワーク構築手法について説明する。以下、各端末をノードと呼び、ノード間の直接通信経路をリンクと呼ぶ。また、前提として、各ノードは固有のIDを持つとする。本手法において、ネットワーク構築に必要な情報をノード情報、アプリケーションからの送信データをメッセージと呼び区別する。

本稿では提案した手法概要のみ説明するが、本手法の詳細な内容は、文献[1]に記述してあるので参照していただきたい。また、本手法は既に、Java言語を用いて実装

[†]名古屋大学大学院工学研究科

Graduate School of Engineering, Nagoya University

[‡]名古屋大学統合音響情報研究拠点

Center for Integrated Acoustic Information Research (CIAIR),
Nagoya University

済みである。

各ノードは探索により、直接通信可能なノードを発見し、保持しているノード情報を相互に交換することにより、時間の経過と共にネットワーク全体を漸増的に把握する。各ノードは、把握したノードと即座にメッセージ通信することができる。

本手法は具体的な通信手段として、赤外線通信を対象にしているが、赤外線通信は、通信速度に比べ、コネクション確立の時間的コストが大きいという短所を持っている。すなわち、赤外線通信の標準規格 IrDA[2] では、最大 4M bps の通信速度が可能であるが、コネクション確立には最低で 500ms 必要である。また、過去 500ms 以内に先行する通信が存在する場合、通信できないという制限が存在する。そのため一回の通信で、できる限り多くの情報を送信し、通信回数をできる限り少なくする方法が必要となる。

以下、§2.1, §2.2 でアドホックネットワーク構築の手順を示す。§2.3 でルーティングについて §2.4 でメッセージ形式と送信方法について説明する。

2.1 準備

本手法では、各ノードは次の 2 種類のタイマーを保持する。これらのタイマーは、探索とノード情報送信のタイミングの決定に用いられる。

• DTimer

探索のタイミングの決定に用いる。各ノードはタイマーを初期状態である値 α に設定し、タイマーが 0 になったら探索を行い、再び α にセットする。

• NTimer(n_j)

探索により発見した隣接ノードごとに用意されるタイマー。引数の n_j は隣接ノードである。

探索によりトポロジ変化を認識するとタイマーは β にセットされる。また、ノード情報を受信しトポロジ変化を認識したときは、タイマーは γ にセットされる。ただし、タイマー作動中にトポロジ変化が起きても β や γ にリセットされない。

タイマーが 0 になると引数の隣接ノードへノード情報を送信する。ただし、NTimer(n_j) が作動中に、ノード n_j へメッセージを送信・中継する際は、タイマーは 0 にセットし、メッセージにノード情報を付加して送信する。

2.2 ネットワーク構築の手順

本手法における各ノードの動作の概要を以下に示す。ただし、動作の順序は特に決められていない。

- DTimer がタイムアウトすると、探索を行い DTimer を α にセットする。探索の結果が以前と異なっていれば全ての隣接ノード n_j の NTimer(n_j) を β にリセットする。
- NTimer(n_j) がタイムアウトすると、隣接ノード n_j にノード情報を送信する。
- ノード情報をノード n_k から受信すると、受信した情報が新しければ $n_j \neq n_k$ である全ての隣接ノード n_j の NTimer(n_j) を γ にセットする。

- 隣接ノード n_j へメッセージ送信・中継をするとき、NTimer(n_j) が作動していればタイマーをリセットし、ノード情報とメッセージを同時に送信する。

以上により、各ノードは直接通信可能なノードは α 時間で認識し、他のノードは近傍のノードから、時間の経過と共に漸増的に認識する。

本手法では、赤外線通信の制限について考慮し、以下の 2 つの点で通信回数の削減を期待している。

1. ノードが周囲のネットワークの変動を認識した場合、同様に近傍のノードもトポロジの変化を認識していると考えられる。トポロジ変化の認識後、即座にノード情報を送信する動作を考えると、あるノードがノード情報を即座に送信した直後、変化を把握している近傍のノードからノード情報を受信した時、もう一度ノード情報を隣接ノードへ送信する必要が生じる。このような動作では、通信回数が多くなり、ノード情報の送信のみでネットワークが混雑する可能性がある。

そこで、ノード情報を送信するまでに待機時間 β を設け、自分のノード情報と他ノードから受信したノード情報をまとめて送信することで、通信回数を削減する。

2. NTimer(n_j) が作動しているときに、メッセージを n_j に送信・中継する場合は、ノード情報とメッセージを同時に送信する。ノード情報とメッセージをまとめて送信することで、2 回の通信回数を 1 回に削減する。

なお、 β と γ の関係は、 $\beta \leq \gamma$ である。探索によるノード情報は自らが作成したノード情報であるので、他ノードから受信したノード情報よりも信頼できる。そのため、探索の結果による情報をより積極的に早く送信すべきである。ただし、今回のシミュレーションではすべて $\gamma = \beta$ としたため、以下では γ については記述しない。

2.3 ルーティングテーブル

各ノードは、探索結果や受信したノード情報を基に全域木の最小パスを求めるアルゴリズムによりルーティングテーブルを作成する。ルーティングテーブルには既知の宛先ノードとその宛先ノードへの最短経路上の次の中継先が示されている。

2.4 メッセージ

メッセージは以下の形式である。

$M(n_i, F, data)$

ここで n_i は送信元ノード ID、 F は送信先ノードの ID のリスト、 $data$ はアプリケーションからの送信要求データである。

メッセージの配送は、ルーティングテーブルを参照し、最短経路となる中継先にメッセージを送信し、目的のノードへメッセージが到達するまで中継を繰り返す。

3 シミュレーションの概要

本節では評価実験のために行ったコンピュータシミュレーションの概要を説明し、次節で実験結果と考察についての報告を行う。

表 1: シミュレーションの条件

ネットワークトポロジ ノード数	3x3 行列 (固定) 9 (固定)
リンク状態変化間隔 リンク切断率 (<i>lbr</i>)	5 s (固定) 変動
メッセージ送信間隔 <i>mi</i> メッセージ送信率 (<i>msr</i>)	1 s (固定) 変動
1 シナリオの長さ	300 s (固定)

シミュレーションによる実験の目的は、様々なネットワーク環境下における本手法の振る舞いと、ノード情報送信の待機時間の設定とノード情報とメッセージの同時送信による通信回数の削減効果、及び、パラメータ α, β の値がネットワーク構築に及ぼす影響を調べることである。また、赤外線通信環境では通信リンクの競合による通信エラーが生じやすいために、パケット送信の際、再送処理を施す必要がある。そこで、再送処理について数種類の方式についてシミュレーションによりパケット送信のエラーの頻度を測定し、適切な再送処理方法を求めた。

3.1 シミュレーションモデル

シミュレーションでは、ネットワークの変化と、メッセージ送信を記述したシナリオを作成し、そのシナリオ上で各ノードは発見したノードとメッセージを送信しながら、アドホックネットワークを構築する。1 シナリオの長さは 300 秒とする。

なお、今回の実験では、同一条件のモデルに対して 10 種類のシナリオを作成し、各シナリオ毎にシミュレーションを行った。各種測定値は 10 回分のシミュレーションから得られた値の平均をとることにした。

3.1.1 ネットワークモデル

シミュレーションでは、9 個のノードを 3x3 行列上に配置し、行列上で隣接するノードを、直接通信可能な隣接ノードとする (図 1)。本手法は赤外線通信を用いてアドホックネットワークを構築するが、赤外線は通信時のコネクション設定の時間的コストが高く、1 回の通信に最低 500ms 必要とするため、宛先ノードへの通信に数十ホップ必要となるような大規模なネットワークの構築は考えられない。本手法では、会議等で、持ち寄った端末間でその場で構築するような小規模なアドホックネットワークの構築を目的としている。適用できるのは、10 台程度の規模を考えており、ネットワークの端末間での通信ホップ数は長くても 4,5 ホップ程度を考えている。そのため、シミュレーションで対象にするネットワークの規模はこの程度で十分であるといえる。

アドホックネットワーク特有のネットワークトポロジの動的変化は以下のように実現する。実験シナリオでは、各リンクは、それぞれ独立に、5 秒毎に、リンク切断率 $\frac{lbr}{100}$ の確率でその後 5 秒間、切断された状態になる。この処理は 5 秒毎に行われる。*lbr* の値を大きくすればするほど、リンクが切断されやすく、不安定なネットワークをシミュレートできる。

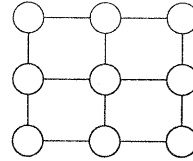


図 1: ネットワークトポロジ

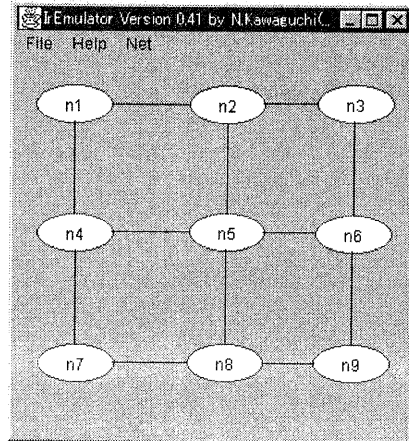


図 2: 赤外線通信エミュレータ上動作画面

ネットワークトポロジが変化しながらも、各ノードが通信可能であるノードを時間の経過と共に漸増的に把握してゆく動作を確認するには、このようなトポロジの動的変化モデルで妥当である。

3.1.2 トラフィックモデル

各ノードは、1 秒ごとにメッセージ送信率 $\frac{msr}{100}$ の確率でメッセージを送信する。すなわち、各ノードは平均して $\frac{100}{msr}$ 秒に一回メッセージを送信する。メッセージの送信先は、その時点で送信経路を把握しているノードのうち、ランダムでどれか 1 つのノードが選ばれる。*msr* の値が大きければするほど、トラフィック量が大きい状況をシミュレートできる。

3.2 赤外線通信のシミュレーション

本手法は通信媒体として赤外線通信 (IrDA) を対象としているため、コンピュータシミュレーションでは、各端末間の通信は IrDA をシミュレートすることが必要である。本手法の赤外線通信上での実装コードを用いるために、赤外線通信エミュレーションサーバを利用し、赤外線通信層をソフトウェアによりシミュレートし、評価実験を行った。図 2 にエミュレーションサーバ上での動作画面を示す。画面上の円が各ノードを表し、ノードとノードを結ぶリンクが直接通信リンクを表す。各ノードはエミュレーションサーバとはソケットを用いて接続する。他のノードへの通信はエミュレーションサーバを介して接続する。各ノードが探索を行う時は、エミュレーションサーバから隣接ノードのリストを取得する。

IrDA 規格では、各端末間の通信は以下のことを規定している。

1. 物理層のモニタで、過去 500ms 以内に先行する通信を検出した場合、割り込んで通信を行うことができない。
2. コネクション確立の際、最低 500ms 必要とする。

そのため、エミュレーションサーバはノード間の通信時に、500ms 以内に先行する通信が行われた場合、ノードに送信エラーを返す。また、コネクション確立時に 500ms 待機した後、送信先ノードにソケットにより、通信パケットを送信する。

3.3 ネットワーク認識率と誤認ノード数

ネットワーク構築状況の時間的な推移を測定するために、次式で定義されるネットワーク認識率 nr (network recognition rate) を導入する。

$$nr = \frac{1}{N} \sum_{i=1}^N \frac{Recognized(n_i)}{Reachable(n_i)}$$

ここで、 N はネットワークにおける全ノード数であり、ここでは 9 である。 $Reachable(n_i)$ は、ノード n_i から到達可能なノード数を示している。また $Recognized(n_i)$ は、ノード n_i が認識しているノード数を表す。共に自ノード n_i を含む。ここで、探索又はノード情報の受信により、ノード n_i がネットワーク内のノード n_j までの実際に到達可能な経路を検出したとき、ノード n_i はノード n_j を認識した。ネットワーク認識率とは、各ノードが到達可能なノードのうちの認識しているノードの割合の、全ノードの平均をとったものである。ネットワーク認識率が 1 であるとき、各ノードはネットワーク内の全ての到達可能なノードを認識している。

認識率はネットワークの構築が進む毎に上昇し、ある程度時間がたった後に、定常的な値に収束すると思われる。このときの定常的な値を以下、定常認識率と呼ぶ。

もう一つの指標として誤認ノード数 er (error nodes) を定義する。

$$er = \frac{1}{N} \sum_{i=1}^N errRecognized(n_i)$$

ここで、 $errRecognized(n_i)$ はノード n_i が到達不可能なノードを到達可能であると誤って把握しているノードの数を示す。誤認ノード数 er は、全ノードの $errRecognized(n_i)$ の値の平均をとったものであり、この値が大きいほど、各ノードはネットワークポロジを誤って把握しているときみることができる。

4 実験結果と考察

4.1 パケットの再送処理について

ノード情報とメッセージのいずれの場合のパケットの送信の際、送信先ノードの帯域占有によるエラーや、通信リンクの消失により、エラーが生じた際、そのパケットを再送する。再送処理は以下に示すアルゴリズムに基づいて行われる。

パケット送信に失敗した時、待機時間 w だけ待機し、パケットを再送信する。送信が成功したら終了し、エラー

表 2: 各種再送処理のエラー率 (%)

待機時間 w (ms)	再送回数 l	エラー率 (%)	
		$msr=5$	$msr=20$
(再送なし) 0	0	35.5	50.4
300 ± 150	2	39.6	41.7
300 ± 150	4	29.3	31.1
300 ± 150	6	6.9	9.3
500 ± 250	2	9.7	18.0
500 ± 250	4	1.9	2.5
500 ± 250	6	2.0	2.4
1000 ± 500	2	33.4	38.2
1000 ± 500	4	12.5	13.5
1000 ± 500	6	11.9	12.3

が生じた場合は w 時間待機し、再び送信する。これを最大で l 回繰り返す。 l 回繰り返しても送信が成功しなかった場合は、パケットを破棄する。

上記の w と l の値について、様々な値をとり、ネットワーク構築のシミュレーションを行い、パケット送信エラー率を測定した。リンクの切断による送信エラーを考慮しないために、 $lbr=0$ 、すなわちネットワークの変動がない状況で、 $msr=5, 20$ の 2 つのトラフィックモデルを用いた。また、 $\alpha=3, \beta=6$ とする。

表 2 に、 w と l の組み合わせと、それぞれのトラフィックモデルにおけるパケット送信エラー率の測定結果を示す。また、再送処理を行わない場合のエラー率も測定した。

表から再送処理を全く行わない場合は、再送処理を行ったときよりも、ほとんどの場合でエラー率が高くなるのが分かる。このことから、赤外線通信環境下では送信エラーを少なくするために、再送処理を行うことが重要であるといえる。

待機時間 w については 500 ± 250 (ms) のときが最もエラー率が低くなっていることが分かる。赤外線通信では、過去 500ms 以内に通信が行われていると、割り込み通信が出来ないという性質をもっているが、このときの w の値は 500 から 250ms 前後した値であるため、ちょうどこれだけの時間だけ待機することで、過去の通信が終了しているため、再送信が成功しやすかったと思われる。

また、表から再送回数 l の値を大きくするほどエラー率が少なくなることが分かる。しかし、 l の値を大きくすると、最初に送信したときから相当な時間経過後も再送信繰り返すことになる。ノード情報などは、送信開始から長時間経過するとネットワークの変動により、送信する意味がない場合が起こり得る。また、 $w = 500 \pm 250$ の場合は、 $l=4$ と 6 のときでエラー率にほとんど差が生じない。そこで、 $w = 500 \pm 250, l=4$ の組み合わせを今回の最適な組み合わせとし、以降のシミュレーションにおいては、この w と l の組み合わせを用いることにする。

4.2 種々のネットワーク条件下でのシミュレーション

本手法の性能を評価するために、パラメータを $\alpha = 3, \beta = 6$ とし、 $msr=5, 20$ の 2 通りのトラフィックモデルと、リンク切断率 lbr の値を 0 から 50 まで 10 ずつ増加させたネットワークモデルを組み合わせた条件設定でネットワークを構築し、認識率を測定した。

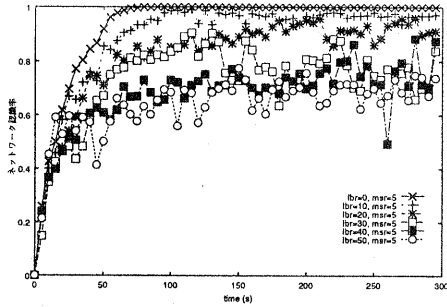


図 3: $msr=5$ に対するネットワーク認識率の推移

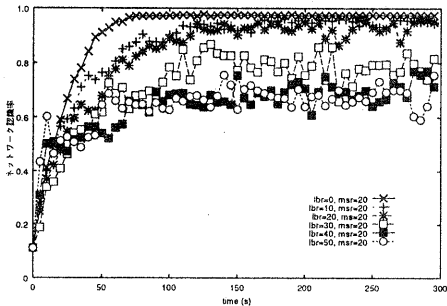


図 4: $msr=20$ に対するネットワーク認識率の推移

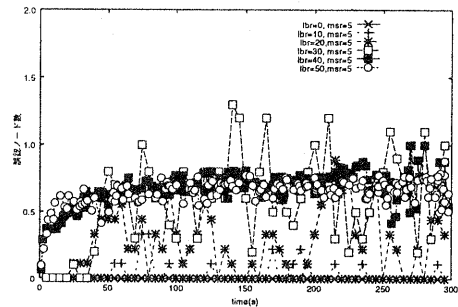


図 5: $msr=5$ に対する誤認ノード数の推移

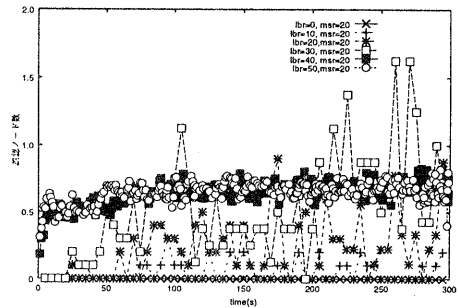


図 6: $msr=20$ に対する誤認ノード数の推移

$msr=5$ としたときの各リンク切断率毎のネットワーク認識率の時間的推移を図 3 に、 $msr=20$ の場合のネットワーク認識率の推移を図 4 に示す。また、各トラフィックモデル毎の誤認ノード数の時間的推移を、 $msr=5$ について図 5 に、 $msr=20$ については図 6 に示す。また、図 7 に各シミュレーションモデルにおける定常認識率を示す。図 8 には各シミュレーションモデルにおけるメッセージ到達率を示す。メッセージ到達率は、送信したメッセージ中の宛先ノードに到達したものの割合を示す。

図 3, 4, 7 からわかるように、 lbr の値の増加に伴いネットワーク認識率は低下している。また、図 7 から $msr=5$ の場合と $msr=20$ の場合とを比較すると、同じ lbr のネットワークモデルに対して、 $msr=20$ のトラフィックモデルのときの方が、 $msr=5$ のトラフィックモデルの時よりも認識率が低下している。これは、トラフィックの増加に伴う通信エラーの上昇によるものと考えられるが、図 7 をみると、トラフィックの増加に伴う認識率の低下よりも、 lbr の値の増加によるネットワークの不安定化の方が認識率に与える影響は大きい。

また誤認ノード数については、図 5 と図 6 からネットワーク構築の初期段階では、ノード情報の交換があまり行われていないので誤認ノード数が少ないが、ノード情報の交換により、ネットワーク構築が進むにつれ、時間の経過と共に上昇し、ある程度の値に収束しているのが分かる。

lbr の値との関係については、 $lbr=0$ のときは、通信不可能なノードが存在しないため、誤認ノード数は 0 である。 lbr の値が大きくなっていくほど、誤認ノード数の

収束する値も大きくなっている。また、 lbr の値が 20 と 30 のとき誤認ノード数の変動幅が大きく、それ以外の lbr の値については変動幅が小さくなっている。これについては、 lbr の値が 10 である場合は、リンクの切断が起こりにくいため、通信不可能になるノード自体が多くないので、誤認ノード数が少ない状況が多いと考えられる。 lbr が 20 や 30 のときは、図 7 から通信可能なノードの多くを認識している状況が多いが、認識しているノードがリンクの切断によりある通信不可能になると、それらのノードを誤認している状態になってしまう。更新されたノード情報を受け取るまでその状態が続いてしまうが、ネットワークの変化により誤認したノードが通信可能になったり、ノード情報の受信により、誤認したノードが通信不可能であることを検出した場合、誤認ノード数は即座に減少するため、変動幅が大きくなると考えられる。 lbr の値が小さい時は通信不可能なノードが存在すること自体が少ないため、誤認ノード数は少ない。 lbr の値が 40, 50 と大きい場合は、ネットワークの変動が大きいので誤認しているノードも多いが、認識しているノード数が少ないため、ネットワークの変化により、認識しているノードと通信不可能となることによる、誤認ノード数の急激な変化はあまりみられないため、誤認ノード数の変動幅が小さいと思われる。

メッセージ到達率についても認識率と同様に、 lbr と msr の値が大きい場合の方が値が低くなっている。本手法では、リンク間では再送処理を行っているが、end-end 間での再送処理は行っていないが、しかし、 $lbr=20$ の

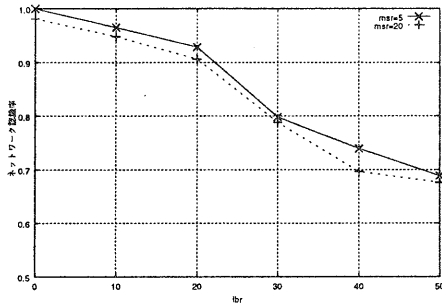


図 7: 各条件下の定常認識率

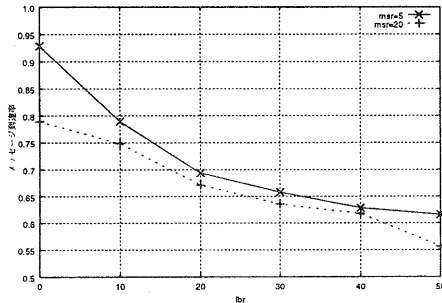


図 8: 各条件下のメッセージ到達率

ときは約7割の確率でメッセージを宛先ノードに送信できている。

図3,4から、本手法では、約30秒ほどで定常認識率の半分に到達し100秒ほど経過するとほぼ定常認識率に到達するのが分かる。すなわち、ノード数9個のアドホックネットワークでは、約30秒ほど経過すれば、4,5個のノードを認識できている。すなわち、2ホップ程度先のノードは30秒程度で認識できる。さらに遠くのノードに関しては、既に認識している近いノードと通信しながら、約70秒ほど経過すれば、3ホップ、4ホップ先の遠いノードも認識でき、ほぼネットワーク全体のノードを認識できることになる。また、 $lbr = 50$ のようなネットワークのリンクの半分が5秒に1度切断されるような極めて不安定なネットワークにおいても、7割近い認識率を保つことが出来る。

4.3 待機時間の設定の有効性の評価

本手法では、トポロジ変化の検出後即座にノード情報送信を送信せず、待機時間を設定することで、待機中にメッセージの送信や中継をする際に、ノード情報とメッセージを同時に送信すること(以下この処理をpiggybackと呼ぶ)の2点により、通信回数の削減を狙っている。そこで、上記の通信回数の削減のための処理の有効性を調べるために、 β の値を様々な値に設定し、piggybackありの場合となしの場合の2通りのネットワーク構築のシミュレーションを行い、送信パケット数を測定した。また比較のため、待機時間を設定しない場合として、 $\beta = 0$ のときに

ついてもシミュレーションを行った。

シミュレーションにおいては、 α の値は3(s)とし、 $lbr=20$, $msr=20$ のシミュレーションモデルを対象とした。

図9,10にシミュレーションの結果を示す。図9の横軸は β の値を示す。縦軸は、全送信パケット数を示す。また、図10はそれぞれのシミュレーション毎の定常認識率を示している。

図9からわかるように、明らかに、待機時間を設けることにより、送信パケット数を削減できている。また、 β の値が大きくなるほど、パケット送信数が減少しているのが分かる。また、待機時間を設けた上で、piggybackを行わない場合よりも、piggybackを行った場合の方が多くの送信回数を削減できている。

図9から、 β の値を大きくすればするほど、パケット送信数が減ることによって、ネットワークトラフィックの混雑を回避できる。しかし、図10から分かるように、待機時間が長くなるほどノード情報を送信するのが遅れるため、ネットワーク認識率が低下してしまう。図10では $\beta = 6$ (s)の時の定常認識率が最大となり、それ以上またはそれ以下では認識率は低下する。

故に、 β の値を大きくすることによりトラフィック量を削減することができるが、ノード情報の送信が遅れるため認識率が低下するといえる。逆に小さくすることで早くノード情報を送信しようとするが、トラフィック量の増大による通信エラーなどにより、かえって認識率が低下する。そのちょうど中間の、トラフィック量が削減でき、かつ、ノード情報をできるだけ早く送信するように、 β の値を決定することが重要である。

4.4 α, β の値と認識率の関係

4.3節では、 β の値を変更することにより、トラフィック量とネットワーク認識率が変化することを示した。トラフィック量を削減し、かつ認識率を最大にするような最適な β の値は常に一定でなく、ネットワークの状況により、変化すると考えられる。

ある条件下において最適な β の値を調べるために、4組のシミュレーションモデルで β の値を様々な値に変更し、ネットワーク構築のシミュレーションを行った。また、 α の値を各種変更することで、 α がネットワーク認識率に与える影響についても調べた。

図11,12,13,14にそれぞれのシミュレーションモデルにおけるシミュレーションの結果を示す。図の縦軸は各モデル毎の定常認識率を示し、横軸は β の値を示す。 α は、3,5,7(s)の3種類の値に設定した。

まず、 α の値がネットワーク認識率に与える影響について述べる。図13と図14に着目する。この2つのシミュレーションモデルは、 $lbr=40$ であるので、ネットワークの変動が激しいモデルである。このときは、図から分かるように、 α の値が小さいほど認識率が高くなっている。ネットワークの変動が大きい状況では、 α の値を小さくして探索間隔を短くし頻りに探索を行うことにより、近傍のトポロジの変化をできる限り早く検出することで、ネットワーク認識率を高くすることができると考えられる。逆に、図11と図12から、ネットワークの変動が少ない状況では、 α の値を変更しても、認識率の大幅な変化

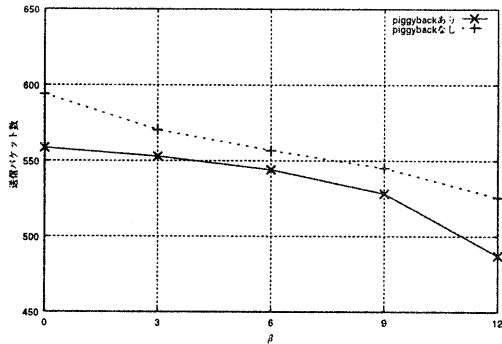


図 9: 各 β 値における総送信パケット数 ($lbr = 20, msr = 20$)

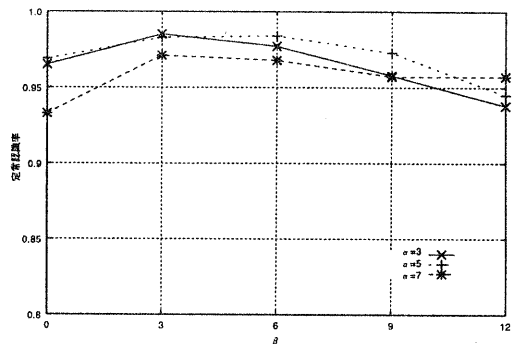


図 11: $lbr=10, msr=5$ のときの定常認識率

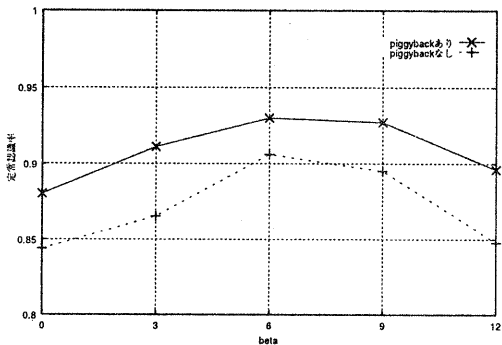


図 10: 各 β 値における定常認識率 ($lbr = 20, msr = 20$)

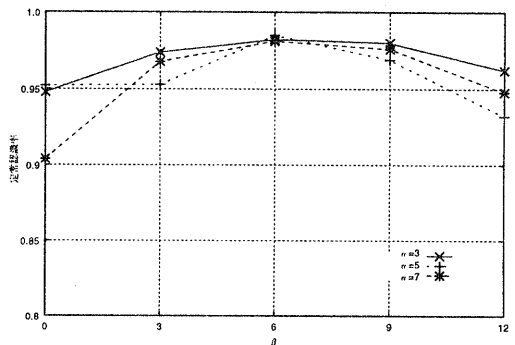


図 12: $lbr=10, msr=20$ のときの定常認識率

は見られない。このことからトポロジの変動が少ないネットワークでは、 α の値が与える影響は小さくなるといえる。

次に、 β の値がネットワーク認識率に与える影響について考える。まず、 lbr が等しく msr が異なるシミュレーションモデルとして図 11 と図 12 を比較する。図 11 では $\beta = 3$ で最も認識率が高いのに対し図 12 では $\beta = 6$ で認識率が最も高くなっている。同様に図 13 と図 14 について比較すると、図 13 では $\beta = 6$ で最も認識率が高くなっているが、図 14 ではそれよりも大きな値、 $\beta = 9$ 前後で最も認識率が高くなっている。この結果から、トラフィック量が大きい状況下では、 β の値を大きくした方が認識率が高くなるといえる。理由は次のように考えられる。トラフィック量が増えると、通信エラーも増えるため、ノード情報の送信もエラーが起りやすくなる。しかし、 β の値を大きくすることにより、送信パケット数を削減することで通信エラーが押えられるため、ノード情報が伝達しやすくなると考えられる。また、メッセージの送信回数が多いため、 β 時間待機する途中で piggyback が行われることも多いため、トラフィック量が小さい状況より β の値を大きくとつても、ノード情報送信の遅延が大きくなりにくいと思われる。

次に msr が等しく、 lbr の値が異なる状況を考える。

図 11 と図 13 を比較すると、図 11 では認識率が最も高いのは $\beta = 3$ のときであるが、図 13 では $\beta = 6$ のときである。 $lbr = 40$ のときの方が頻りにネットワークが変化するので、送信するノード情報数も多いと考えられる。そのため、ノード情報を送信する時間が遅くなる可能性はあっても、 $lbr = 10$ のときよりも長い時間待機した方がパケット送信数を削減でき、通信エラーが減少するため、認識率が上がると考えられる。同様のことが図 12 と 14 を比較した場合についてもいえる。

この実験により、 $\alpha = 3, \beta = 6$ の組み合わせが一般的に認識率が高くなっていることが分かる。

5 関連研究

アドホックネットワーク上でのルーティングプロトコルについては IETF[3] 内で MANET というアドホックネットワークのワーキンググループにより研究が行われている。これは AODV[5] や DSR[6] に代表されるルーティングやデータの転送に関する研究である。主に既存の IP を用いたアプリケーションをモバイルコンピューティングでも利用することを目的としている。本研究は、IP をモバイル環境に使うのではなく、新たなモバイル用のネットワーク構築プロトコルを考えている点で異なる。また、これらの研究は大規模なネットワークを対象としているが、本

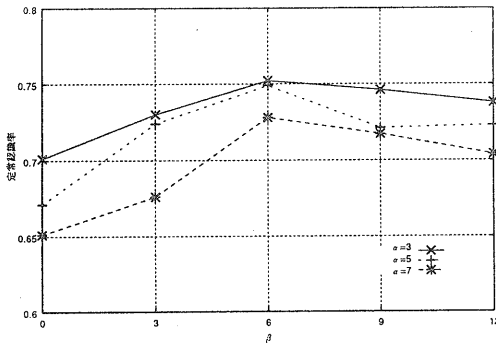


図 13: $lbr=40, msr=5$ のときの定常認識率

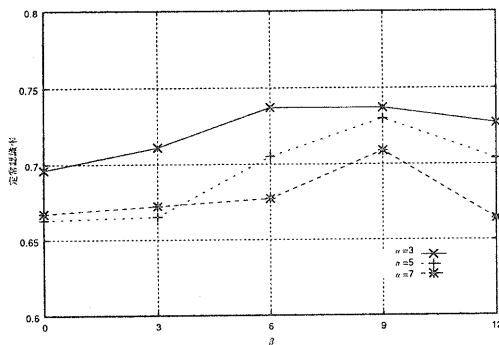


図 14: $lbr=40, msr=20$ のときの定常認識率

研究は比較的小規模なアドホックネットワークの構築を目的としている。文献 [7] では、DSR や DSDV [4] など MANET で提案されたルーティング手法の比較・評価が詳細に述べられている。しかし、この文献には本稿のように、各プロトコルにおける各種パラメータの設定と、パラメータ値の変更が与える影響については詳しく述べられていない。

最近では、Bluetooth [8] や HomeRF [10], IEEE802.11 [9] などの近距離無線通信技術の研究・開発が盛んに行われている。本手法では赤外線通信を無線通信の手段として考えており、現在、携帯端末上で最も手軽に利用できるものは赤外線通信しかないが、赤外線通信でなく、Bluetooth などが携帯端末上で標準搭載される可能性は十分ある。そのため、これらの近距離無線通信デバイスを本手法の通信手段とすることも今後の視野に入れたい。

6 まとめと今後の課題

今回の評価実験により、様々なネットワークの状況に対する本手法の振る舞いを求めることが出来た。シミュレーション結果から、各ノードは、30 秒程度で 2 ホップ程度先のノードを認識でき、100 秒程度で 4 ホップ程度先のノードを認識できることが分かった。また、シミュレーションには、ネットワークの変化が少ない状況では、各

ノードは、通信可能なほぼ全てのノードを認識でき、リンクが 10 秒間で 5 秒切れているような極めて不安定なネットワークでも約 7 割近くの通信可能ノードを認識できた。

また、赤外線通信環境下では通信時に再送処理を行うことが重要であることを示し、数種類の再送処理方法についてシミュレーションを行い送信エラー率を求めた。

本手法では、赤外線通信の通信コネクション確立の際の時間的コストを考慮してノード情報送信の待機時間の設定と piggyback を行い、通信回数を削減しているが、その有効性をシミュレーションにより示すことができた。 β の値を大きくすることでパケットの送信回数を削減可能であり、パケット送信のエラーを減らすことができる。パケット送信エラーが少なくとネットワーク認識率も認識率も増加するが、 β の値を大きくとりすぎると、ノード情報を送信するまでに遅延が生じ、認識率がかえって低下した。これより、 β の値を大きくすると認識率は上昇し、 β がある値をこえると認識率は低下することがわかった。本稿では、いくつかのシミュレーションモデルについて、認識率を最大にする β の値を求めた。

また、 α の値がネットワークの構築に与える影響について調査し、その結果として、ネットワークの動的変化が小さい状況では α の変化は大きな影響を与えないが、ネットワークの動的変化が大きい状況では、 α の値を小さくし、探索間隔を短くすることによりネットワークの変化への対応が早くなり、認識率が上昇するという知見を得た。

実際のネットワークでは、トポロジ変化やトラフィック量は一定でなく、常に変化する。トポロジやトラフィックの動的な変化に対し、状況に応じてパラメータ α, β を最適な値に動的に変化する仕組みを作ることが今後の課題である。

参考文献

- [1] 戸崎貴資, 河口信夫, 外山勝彦, 稲垣康善: 漸増的に端末を認識するアドホックネットワーク構築手法, DICO 1999, (1999).
- [2] Infrared Data Association: Serial Infrared Link Access Protocol, version 1.1 (1996).
- [3] Internet Engineering Task Force : (<http://www.ietf.org>)
- [4] Charles E. Perkins, Pravin Bhagwat: Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers, SIGCOMM '94 (1994).
- [5] Charles E. Perkins: Ad Hoc On Demand Distance Vector (AODV) Routing, Internet Draft, draft-ietf-manet-aodv-02.txt (1997) work in progress.
- [6] David B. Johnson, David A. Maltz: Dynamic Source Routing in Ad Hoc Wireless Networks, In Mobile Computing, edited by Tomasz Imielinski and Hank Korth, Chapter 5, Kluwer Academic Publishers, pp.153-181 (1996).
- [7] Josh Broch, David A. Maltz, David B. Johnson, Yuh-Chun Hu, Jorjeta Jetcheva: A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols, Mobi-com 98 (1998).
- [8] Bluetooth : (<http://www.bluetooth.com>)
- [9] IEEE802.11 : (<http://grouper.ieee.org/groups/802/11/>)
- [10] HomeRF : (<http://homeRF.org/>)