

Title	レヴィ飛行的な探索構造が埋め込まれたネットワークを用いたメッセージフェリールーティング
Author(s)	小牧, 嵩征
Citation	
Issue Date	2013-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/11274
Rights	
Description	Supervisor : 林幸雄, 知識科学研究科, 修士

修士論文

レヴィ 飛行的な探索構造が埋め込まれたネットワークを用いた
メッセージフェリールーティング

指導教員：林 幸雄 准教授

北陸先端科学技術大学院大学
知識科学研究科知識科学専攻

1150018 小牧 嵩征

審査委員：林 幸雄 准教授 (主査)

吉田 武稔 教授

中森 義輝 教授

金井 秀明 准教授

2013 年 2 月

Copyright ©2013 by Takayuki Komaki

目次

第1章 序論	13
1.1 研究の背景と目的	13
1.2 関連研究	15
1.2.1 正方格子上におけるレヴィ探索	15
1.2.2 マルチメッセージフェリー	17
第2章 ネットワーク構造	19
2.1 GMSQ ネットワーク生成アルゴリズム	19
2.2 GMSQ ネットワークのリンク長分布	24
2.3 正方格子上のレヴィ飛行	26
2.4 ネットワークでのパケット発生	26
第3章 経路探索アルゴリズム	28
3.1 メッセージフェリーの基本動作	29
3.2 経路要求 $REQ_{st}(s, d)$ の拡散方法	29
3.3 メッセージフェリーの経路確立方法	30
3.3.1 協調的にリンクバッファを利用した宛先ノード d の探索	32
3.4 経路要求 $REQ_{st}(s, d)$ と $fn_A(t-1)$ の削除	33
第4章 シミュレーション結果	37
4.1 GMSQ ネットワークでのシミュレーションの	
1 ステップの流れ	37

4.2	正方格子でのレヴィ飛行シミュレーションの	
	1ステップの流れ	38
4.3	シミュレーション条件	40
4.4	パケット発生ノードの発見時間	41
4.5	宛先ノード d の発見効率	48
4.6	パケットの経路長	58
第5章 結論		68
付録A 周期的境界条件を用いた		
	正方格子上でのレヴィ飛行	72

目 次

1.1	正方格子上でのレヴィ飛行	18
1.2	正方格子上でのレヴィ飛行での探索効率 . 縦軸が探索効率で横軸がベ き乗分布の傾きとなっている . 菱形 , 三角 , 丸○のマークが順に 餌の密度が高 , 中 , 低となっている	18
2.1	GMSQ ネットワーク	21
2.2	関東エリアの人口データ可視化図 . 色が赤の箇所ほど人口が多くなっ ている	21
2.3	一様ランダムな分割での GSMQ ネットワークの可視化図 . 左上 : ノー ド数 500 , 右上 : ノード数 1000 , 左下 : ノード数 3000 , 右下 : ノード 数 5000	22
2.4	関東エリアの人口データを用いて生成した GSMQ ネットワークの可 視化図 . 左上 : ノード数 500 , 右上 : ノード数 1000 , 左下 : ノード数 3000 , 右下 : ノード数 5000	23
2.5	関東エリアの人口データを用いて生成した GSMQ ネットワークのリ ンク長分布 $P(l_{ij})$. ネットワークサンプルは 100 とした	25
2.6	GMSQ ネットワークノードのテリトリーサイズ累積頻度分布 . ノー ドのテリトリー面積は一樣ではなく , 偏りがあることがわかる . ネット ワークサンプル数は 100 とした	27

- 3.1 メッセージフェリーのリンク履歴の保存方法．リンクの重複を避けるために，それをスタックから破棄し，リンク情報を新たにスタックに追加する処理を行なう．追加するリンク情報と同じリンク情報が，スタックの任意の位置に無い場合は，FILO のスタック構造となる (左) . 34
- 3.2 メッセージフェリーの状態による挙動の違い．点線はメッセージフェリーが経路要求を保持しない[フリーな状態]で，経路要求 $REQ_{st}(s, d)$ を探索していることを表す．実線はメッセージフェリーが経路要求 $REQ_{st}(s, d)$ を複数持ち，複数の宛先を探索している[探索状態]を表す．また，メッセージフェリー A は保持するすべての経路要求 $REQ_{st}(s, d)$ を訪れたノードに渡しながらか，複数の宛先と，メッセージフェリー A が持っていない経路要求を探索している 34
- 3.3 メッセージフェリー同士での，ノードを介した経路要求 $REQ_{st}(s, d)$ 交換の時系列の図 35
- 3.4 パケットの経路確立方法．1．経路要求 $REQ_{st}(s, d)$ を保持したメッセージフェリーがノード d を発見．2．ノード d から $fn(t-1)$ をたどり，ネットワークを作成する．3．作成したネットワークから s, d 間の経路を Dijkstra 法で求める 35

- 3.5 リンクバッファを宛先探索に使用したパケットの経路確立方法．黒色の線種は図 3.2 と同じ意味を表している．1．経路要求 $REQ_{st}(s, d)$ を保持したメッセージフェリー A がノード i を通過．その後，宛先ノード d を通過してリンクバッファに i, d 間のリンクを保存したメッセージフェリー B がノード i に到着．2．ノード i から図 3.4 の方法で $fn(t-1)$ をたどり，ノード i, s の経路を含んだサブグラフ (赤線) を作成する．3．2 で作成したサブグラフにメッセージフェリー B のリンクバッファを組み合わせる．以上の処理により，ノード s, d 間の経路を含んだサブグラフ (赤線) が作成でき，そこから s, d 間の経路を Dijkstra 法で求めることにより経路が確立出来る． 36
- 4.1 一様ランダム分割 (左図) と人口を用いた GMSQ ネットワーク (右図) でのメッセージフェリーがパケットを発見するまでのステップ数．縦軸はメッセージフェリーがパケットが作成した $REQ_{st}(s, d)$ を見つけるまでのステップ数で，横軸はメッセージフェリーの数 m となっている．バッファサイズの最大値は全リンク数となっている． 42
- 4.2 一様ランダム分割 (左図) と人口を用いた GMSQ ネットワーク (右図) でのメッセージフェリーがパケットを発見するまでに歩いた距離．縦軸は縦軸はパケットが作成した $REQ_{st}(s, d)$ を見つけるまでに 1 つのメッセージフェリーが歩いた平均距離．横軸はメッセージフェリーの数 m となっている．バッファサイズの最大値は全リンク数となっている． 43

- 4.3 一様ランダム分割 (左図) と人口を用いた GMSQ ネットワーク (右図) でのメッセージフェリーがパケットを発見するまでのステップ数．縦軸はメッセージフェリーがパケットが作成した $REQ_{st}(s, d)$ を見つけるまでのステップ数で，横軸はネットワークのノード数 N となっている． 44
- 4.4 一様ランダム分割 (左図) と人口を用いた GMSQ ネットワーク (右図) でのメッセージフェリーがパケットを発見するまでに歩いた距離．縦軸はパケットが作成した $REQ_{st}(s, d)$ を見つけるまでに1つのメッセージフェリーが歩いた平均距離．横軸はネットワークのノード数 N となっている． 45
- 4.5 関東エリアの人口データを用いた正方格子上でのレヴィ飛行探索 (左図) と人口を用いた GMSQ ネットワーク (右図) での，メッセージフェリーがパケットが作成した $REQ_{st}(s, d)$ を見つけるまでのステップ数．横軸はべき乗分布の傾き μ となっている．平均パケット発生数 R は 0.01 46
- 4.6 関東エリアの人口データを用いた正方格子上でのレヴィ飛行探索 (左図) と人口を用いた GMSQ ネットワーク (右図) での，パケットが作成した $REQ_{st}(s, d)$ を見つけるまでに1つのメッセージフェリーが歩いた平均距離．横軸はべき乗分布の傾き μ となっている．平均パケット発生数 R は 0.01 47
- 4.7 一様ランダムな分割 (左図) と人口を用いた GMSQ ネットワーク (右図) での，メッセージフェリーが宛先ノード d を発見するまでのステップ数．横軸はフェリーの数 m となっている．バッファサイズの最大値は全リンク数となっている． 49

- 4.8 一様ランダムな分割 (左図) と人口を用いた GMSQ ネットワーク (右図) での , 宛先ノード d を発見するまでに最初に $REQ_{st}(s, d)$ を発見したメッセージフェリーが歩いた距離 . 横軸はフェリーの数 m となっている . バッファサイズの最大値は全リンク数となっている 50
- 4.9 一様ランダムな分割 (左図) と人口を用いた GMSQ ネットワーク (右図) での , 最初に $REQ_{st}(s, d)$ を発見したメッセージフェリーがばら撒いた $REQ_{st}(s, d)$ を受け取った他のメッセージフェリーが , 宛先ノード d を発見するまでに歩いた平均距離 . 横軸はフェリーの数 m となっている . バッファサイズの最大値は全リンク数となっている 51
- 4.10 一様ランダムな分割 (左図) と人口を用いた GMSQ ネットワーク (右図) での , メッセージフェリーが宛先ノード d を発見するまでのステップ数で , 横軸はネットワークのノード数 N となっている . 平均パケット発生数 R は 0.01 52
- 4.11 一様ランダムな分割 (左図) と人口を用いた GMSQ ネットワーク (右図) での , 宛先ノード d を発見するまでに最初に $REQ_{st}(s, d)$ を発見したメッセージフェリーが歩いた距離 . 横軸はネットワークのノード数 N となっている . 平均パケット発生数 R は 0.01 53
- 4.12 一様ランダムな分割 (左図) と人口を用いた GMSQ ネットワーク (右図) での , 最初に $REQ_{st}(s, d)$ を発見したメッセージフェリーがばら撒いた $REQ_{st}(s, d)$ を受け取った他のメッセージフェリーが , 宛先ノード d を発見するまでに歩いた平均距離 . 横軸はネットワークのノード数 N となっている . 平均パケット発生数 R は 0.01 54

- 4.13 関東エリアの人口データを用いた正方格子上でのレヴィ飛行探索 (左図) と人口を用いた GMSQ ネットワーク (右図) での, メッセージフェリーが経路要求 $REQ_{st}(s, d)$ を見つけてから宛先ノード d を発見するまでのステップ数. 横軸はべき乗分布の傾き μ となっている. 平均パケット発生数 R は 0.01 55
- 4.14 関東エリアの人口データを用いた正方格子上でのレヴィ飛行探索 (左図) と人口を用いた GMSQ ネットワーク (右図) での, 宛先ノード d を発見するまでに最初に $REQ_{st}(s, d)$ を発見したメッセージフェリーが歩いた距離. 横軸はべき乗分布の傾き μ となっている. 平均パケット発生数 R は 0.01 56
- 4.15 関東エリアの人口データを用いた正方格子上でのレヴィ飛行探索 (左図) と人口を用いた GMSQ ネットワーク (右図) での, 最初に $REQ_{st}(s, d)$ を発見したメッセージフェリーがばら撒いた $REQ_{st}(s, d)$ を受け取った他のメッセージフェリーが, 宛先ノード d を発見するまでに歩いた平均距離. 横軸はべき乗分布の傾き μ となっている. 平均パケット発生数 R は 0.01 57
- 4.16 一様ランダムな分割 (左図) と人口を用いた GMSQ ネットワーク (右図) での, 到達したパケットの経路上のリンク長の和 L_s . 横軸はフェリーの数 m となっている. バッファサイズの最大値は全リンク数となっている. 59
- 4.17 一様ランダムな分割 (左図) と人口を用いた GMSQ ネットワーク (右図) での, 到達したパケットの経路上のリンク長の和 L_s と, パケットの送信ノード s と宛先ノード d 間のネットワークの最短距離長 L_t との比. 横軸はフェリーの数 m となっている. バッファサイズの最大値は全リンク数となっている. 60

- 4.18 一様ランダムな分割 (左図) と人口を用いた GMSQ ネットワーク (右図) での, 到達したパケットの経路上のリンク長の和 L_s . 横軸はネットワークのノード数 N となっている. 平均パケット発生数 R は 0.01 61
- 4.19 一様ランダムな分割 (左図) と人口を用いた GMSQ ネットワーク (右図) での, 到達したパケットの経路上のリンク長の和 L_s と, パケットの送信ノード s と宛先ノード d 間のネットワークの最短距離長 L_t との比. 横軸はネットワークのノード数 N となっている. 平均パケット発生数 R は 0.01 62
- 4.20 関東エリアの人口データを用いた正方格子上でのレヴィ飛行探索 (左図) と, 人口を用いた GMSQ ネットワーク (右図) での, 到達したパケットの経路上のリンク長の和 L_s . 横軸はべき乗分布の傾き μ となっている. 平均パケット発生数 R は 0.01 63
- 4.21 関東エリアの人口データを用いた正方格子上でのレヴィ飛行探索 (左図) と, 人口を用いた GMSQ ネットワーク (右図) での, 到達したパケットの経路上のリンク長の和 L_s と, パケットの送信ノード s と宛先ノード d 間のネットワークの最短距離長 L_t との比. 横軸はべき乗分布の傾き μ となっている. 平均パケット発生数 R は 0.01 64
- 4.22 一様ランダムな分割 (左図) と人口を用いた GMSQ ネットワーク (右図) での, Dijkstra 法でパケットの経路を求める時に使用したリンクの数と, ネットワークの全リンク数との情報量の比. 横軸はフェリーの数 m となっている. 65
- 4.23 一様ランダムな分割 (左図) と人口を用いた GMSQ ネットワーク (右図) での, Dijkstra 法でパケットの経路を求める時に使用したリンクの数とネットワークの全リンク数との情報量の比. 横軸はネットワークのノード数 N となっている. 平均パケット発生数 R は 0.01 66

4.24	関東エリアの人口データを用いた正方格子上でのレヴィ飛行探索 (左図) と, 人口を用いた GMSQ ネットワーク (右図) での, Dijkstra 法でパケットの経路を求める時に使用したリンクの数とネットワークの全リンク数との情報量の比. 横軸はべき乗分布の傾き μ となっている. 平均パケット発生数 R は 0.01	67
A.1	$R=0.01$, $N=1000$, GMSQ	73
A.2	$R=0.01$, Lattice の $BR=0.2$ は $N=500$ のときの $BR=0.2$ に合わせた数となっている. また, Lattice のパケットの発生位置は格子の全てのノード	74
A.3	$R=0.01$, Lattice の $BR=0.2$ は $N=3000$ のときの $BR=0.2$ に合わせた数となっている. また, Lattice のパケットの発生位置は格子の全てのノード	75
A.4	$R=0.01$, Lattice の $BR=0.2$ は $N=500$ のときの $BR=0.2$ に合わせた数となっている. また, Lattice のパケットの発生位置は $N=500$ の GMSQ ネットワークと同じとしている	76
A.5	$R=0.01$, Lattice の $BR=0.2$ は GMSQ が $N=3000$ 時の $BR=0.2$ に合わせた数となっている. また, Lattice のパケットの発生位置は $N=500$ の GMSQ ネットワークと同じとしている	77
A.6	$R=0.01$, Lattice の $BR=0.2$ は GMSQ が $N=500$ 時の $BR=0.2$ に合わせた数となっている. また, Lattice のパケットの発生位置は $N=3000$ の GMSQ ネットワークと同じとしている	78
A.7	$R=0.01$, Lattice の $BR=0.2$ は GMSQ が $N=3000$ 時の $BR=0.2$ に合わせた数となっている. また, Lattice のパケットの発生位置は $N=3000$ の GMSQ ネットワークと同じとしている	79

表 目 次

2.1	GMSQ ネットワークのリンク長分布 $P(l_{ij})$ を最小二乗法でのフィッ ティングした値	25
-----	---	----

第1章 序論

1.1 研究の背景と目的

近年, 移動体間など回線結合が変化する動的な通信環境を想定した DTN(delay/disruption-tolerant networking) において, メッセージフェリーを用いた種々のルーティング手法が提案されている [1]. メッセージフェリーは情報を蓄積, 運搬する役割を果たす. すなわち, 蓄積した情報の宛先に到達するために, 2次元空間を移動しながら宛先を探索する. その際, 2次元空間上の探索をどのような経路で行うのかで, 探索効率が決まる. そのため, メッセージフェリーの経路の決定方法は重要な課題となる. 経路の決定方法として, あらかじめ経路をメッセージフェリーに与えておき, その経路を逐次的に移動する方法 [2] や, 方向をランダムに決定し, 生物の最適な餌探索戦略として知られるレヴィ飛行などで移動距離を制御する方法 [3] が提案されている. レヴィ飛行とは移動距離 l_j の頻度分布がべき乗分布 $P(l_j) \sim l_j^{-\mu}, 1 < \mu < 3$ に従い, 鳥や蜂, 人など様々な動物がレヴィ飛行を行っていることが報告されている [10]. また, 餌の密度が低い場合 $\mu \approx 2$ での探索の効率が良い事が連続空間と三角, 正方格子空間の双方で報告されている [11,12].

メッセージフェリーに関する既存の研究は, 空間上の一様な位置にノードが存在すると仮定したものが多い [1,2,3]. しかしながら, 現実の通信要求の発生位置は, 空間的な偏りがある場合が多く, 例えば, ルーターの空間的な分布は人口の密度に対応していることが報告されている [4]. また, 2次元空間上で空間的に偏った配置を持つ餌 (位置が不明なオブジェクト) 探索の手法として, 餌の密度に対応したノード配置をもつネットワークを自己組織的に作り, その上での餌の探索を行うことが提

案されている [5] . この手法は , 餌の探索効率が最適なレヴィ飛行よりも , 探索効率が良くなることが報告されている .

さらに , 複数のメッセージフェリーを用いて運搬効率を高める , マルチメッセージフェリーという手法も提案されている [6] . マルチメッセージフェリーでは , メッセージフェリーが直接出会った時に同期的に情報を交換し合い宛先を探索する方法と , 各メッセージフェリーが独立に仕事を行う方法などがある . 同期的な方法の場合 , 探す空間が広いほどメッセージフェリー同士が出会う頻度は極端に小さくなってしまう問題がある . 一方 , 独立に仕事を行う方法では , メッセージフェリーが協調的に働かないため , 一般的に情報を交換する方法よりも宛先の発見等の時間が掛かることが問題となる .

そこで本稿では , 複数のメッセージフェリーがノードを介して非同期に交換する方法に着目して , 人口分布のように偏ったノードの空間配置を持つネットワーク構造に適した経路探索アルゴリズムを提案し , その効率性を正方格子でのレヴィ飛行と比較し , 手法の有効性を調べた . 通信端末の移動を考慮した場合ネットワークが非連結性になった場合の影響と , ネットワークの連結性との影響が混合され傾向が分かり難くなることから , 本稿では通信端末の移動については議論せず移動についての議論は今後の課題とする .

1.2 関連研究

レヴィ飛行は2次元空間に配置された餌探索の有効な手法として知られている [11,12] . このことは, 連続空間と格子 (三角, 正方) の双方で確認されている [11,12] . この章では後の議論で必要となる正方格子におけるレヴィ探索と探索効率について紹介する .

また, 多数のメッセージフェリーを使用し伝搬効率を高めるマルチメッセージフェリーは様々な方法が提案されていることから, マルチメッセージフェリーについての既存研究 [6] も紹介する .

1.2.1 正方格子上におけるレヴィ探索

正方格子におけるレヴィ探索は, 図 1.1 の様に正方格子ネットワークの上で探索者が方向をランダムに決定し, べき乗分布に従った移動距離 l_j を歩き餌を探索する . 探索者が餌を発見できる範囲 r_v は予め決まっており, その範囲を探索しながら探索者は餌を探索する . 詳しくは, 以下のルールに従う .

1. もし, 餌を発見できる範囲 r_v に餌が近接する場合, 探索者は餌に向かって直線移動 (途中で曲がってはいけない) し餌を取得することができる .
2. もし, r_v の範囲の中に餌が存在しなかったら, 進む方向をランダムに決め, 移動距離 l_j (最小単位は l_0) を $P(l_j)$ の確率分布 (下記参照) に基づいて選んで移動する . 移動している間も探索者は r_v の範囲で餌を探しながら移動していく . 移動している間に餌を見つけたら 1 に移る . 距離 l_j に達した時点で発見出来なかった場合は, また新たな進行方向と新しい移動距離 l_{j+1} を選択する .

また, 移動距離を決める確率分布は以下の式に従う

$$P(l_j) \sim l_j^{-\mu}, \quad l_j > l_0, \quad (1.1)$$

l_0 は小さい距離をカットオフするためのパラメータ．このべき乗分布の傾きが 1 に近いほど直線的な動きになり，3 に近いほどブラウン運動的な動きとなる．

探索効率 $\eta(\mu)$ は捕まえた餌の数 N_m と歩いた距離 L_m の割合に，餌の密度を掛け合わせたものとする．つまり，餌がどの程度存在する場所かに応じて，どの程度歩いてどのくらいの数の餌を発見したかが指標となる．1 個の餌を探し出すのに要した平均距離の逆数 $\eta(\mu)$ は，M 回シミュレーションを試行して平均をとる．

$$\eta(\mu) = \frac{1}{M} \sum_{m=1}^M \frac{N_m}{L_m(\mu)} \quad (1.2)$$

ここで， L_m は m 番目のシミュレーションで N_m 個の餌を発見するまでに移動した総距離を表す．また，探索者が餌を発見できる餌の 2 点間の平均距離を

$$\lambda = \frac{(L+1)^2}{(N_t 2r_v/s)} \quad (1.3)$$

と定め，餌の密度と探索効率を掛け合わせた指標 $\lambda\eta(\mu)$ を探索効率と定義する．

問題設定としては，餌を 1 つ発見したならばその餌を消して別の場所に 1 つ餌を再配置する方法 (destructive) と，餌を発見しても餌の位置は変えない方法 (nondestructive) がある．

以上の方法で探索効率を図ると図 1.2 の様に餌の密度が少ない時べき指数 $\mu \approx 2$ の箇所がもっとも効率が良くなっていることがわかる．

1.2.2 マルチメッセージフェリー

マルチメッセージフェリー方法のルーティングでは、多数のメッセージフェリーを使用する。そのため、下記のような方法が提案されている [6]。

方法 1. 各メッセージフェリーが単独で仕事をこなす方法 (No interaction)。

方法 2. メッセージフェリー同士で探索しているノードの情報を交換し合い、協調して仕事をこなす方法。

方法 2 に関しては、メッセージフェリー同士が直接出会った時に情報を交換する方法 (Ferry relaying) と、他のノードを介して情報を交換する方法 (Node relaying) に分けられる。これらのどの情報交換方法でも、メッセージフェリー同士が情報を交換し合い協調して探索するため、各メッセージフェリーが単独に仕事をこなす No interaction よりも基本的に効率は高くなる [6]。しかし、Ferry relaying の欠点として、メッセージフェリー同士が同じ時刻に同じ場所にいないといけないという同期化の制約がある。それに対して、Node relaying の場合はノードに情報を渡し、他のメッセージフェリーが訪れた時にその情報をノードが渡すという方法がとれるため、非同期にメッセージフェリーは情報のやり取りが行える利点がある。そこで本稿ではノードを介して非同期な通信が行える Node relaying に焦点を当てた。

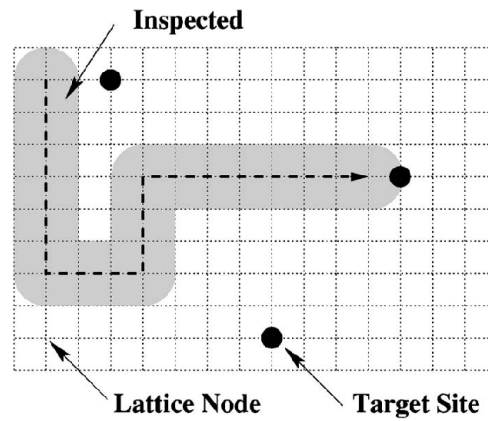


図 1.1: 正方格子上でのレヴィ飛行

出典 : M. C. Santos et al, Optimization of random searches on regular lattices, 2005[12].

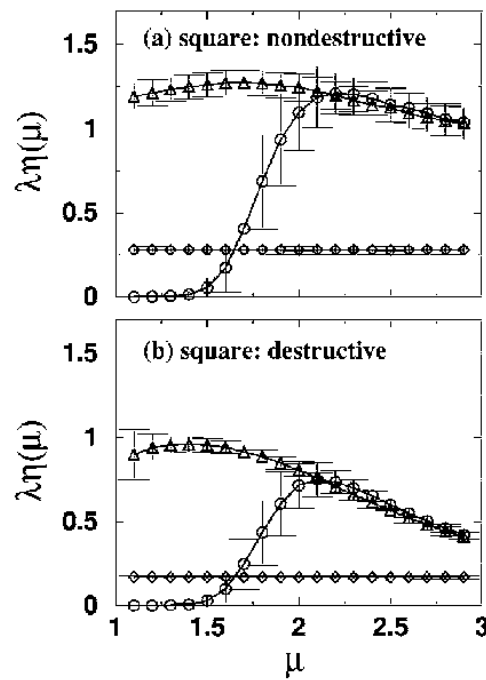


図 1.2: 正方格子上でのレヴィ飛行での探索効率．縦軸が探索効率で横軸がべき乗分布の傾きとなっている．菱形，三角，丸○のマークが順に餌の密度が高，中，低となっている．

出典 : M. C. Santos et al, Optimization of random searches on regular lattices, 2005[12].

第2章 ネットワーク構造

2.1 GMSQ ネットワーク生成アルゴリズム

GMSQ(Generalized Multi-Scale Quarted) ネットワークは以下の手順で生成される。このモデルは、自己相似な正方形分割における Multi-Scale Quarted ネットワーク [7] を長方形分割に一般化したもの [8] である。後述するように、GMSQ ネットワークのリンク長の頻度分布はべき乗分布に近似できるため、歩いた距離の頻度分布がべき乗分布の Levy 飛行と関連している。

1. 一辺の長さが L の正方形を用意し、その四隅のそれぞれにノードを配置。
2. 毎時、長方形の面の集合から 1 つの面を一様ランダムに選択する。ただし、 $L \times L$ の正方格子で分割することから長方形の縦または横の一辺の長さが 1 の場合は分割不可能とし、選択から除外する。
3. $L \times L$ の正方格子の交点のみを分割点とし、交点を一様ランダムに選択する。
4. 上記の交点における縦横軸で、(2) で選ばれた面を 4 つの長方形にする。
5. 全ての長方形の縦か横の一辺の長さが 1 になる時が限界で、選択可能な面が無くない範囲で、決められたノード数 N に達するまで 2~4 を繰り返す。

GMSQ ネットワークは一様ランダムな確率で分割を行っているにも関わらず、図 2.3 のようにノード配置が空間的に密な箇所と疎な箇所に自己組織化され、空間的に偏った配置となる特徴を持つ。

より現実的な設定として，ノード配置の空間的偏りを人口の偏りに対応させるために，以下の手順に従った人口を用いた構成方法も考えられる．

1. 一辺の長さが L の正方形を用意し，その四隅のそれぞれにノードを配置．
2. 毎時，長方形の面の集合から 1 つの面を，各面内に存在する人口に比例した確率で選択する．ただし， $L \times L$ の正方格子で分割することから長方形の縦または横の一辺の長さが 1 の場合は分割不可能とし，選択から除外する．また，面内に存在する人口が 0 の場合も，選択から除外する．
3. $L \times L$ の正方格子の交点のみを分割点とし，選択した面での人口重心から最も近い交点を選択する．
4. 上記の交点における縦横軸で，(2) で選ばれた面を 4 つの長方形にする．
5. 全ての長方形の縦か横の一辺の長さが 1 になる時が限界で，選択可能な面が無くない範囲で，決められたノード数 N に達するまで 2～4 を繰り返す．

以上の手順により，図 2.4 のように人口に対応したノード配置を持つネットワークが自己組織的に生成される．人口統計データに合わせて，空間サイズ $L = 160$ とした．

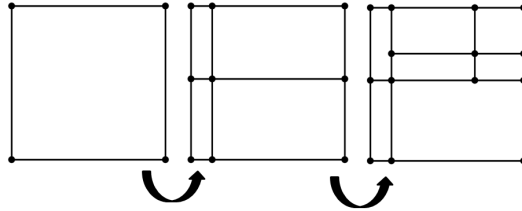


図 2.1: GMSQ ネットワーク

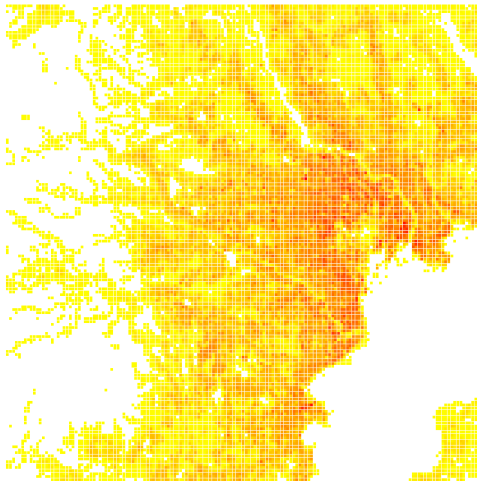


図 2.2: 関東エリアの人口データ可視化図．色が赤の箇所ほど人口が多くなっている．

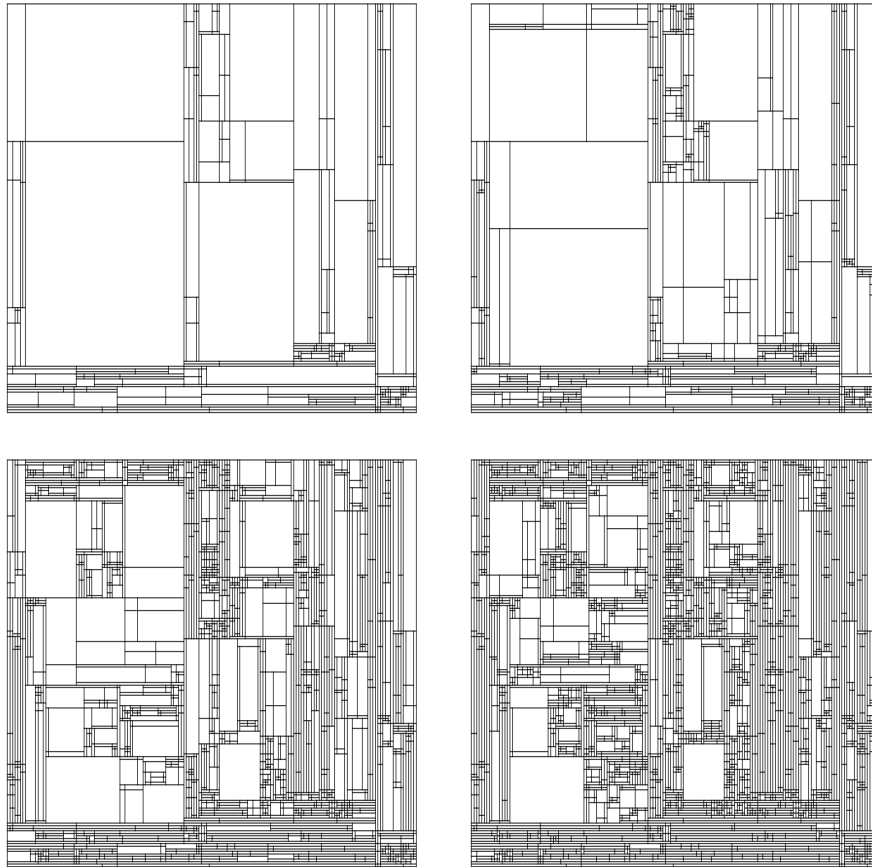


図 2.3: 一様ランダムな分割での GMSQ ネットワークの可視化図．左上：ノード数 500，右上：ノード数 1000，左下：ノード数 3000，右下：ノード数 5000．

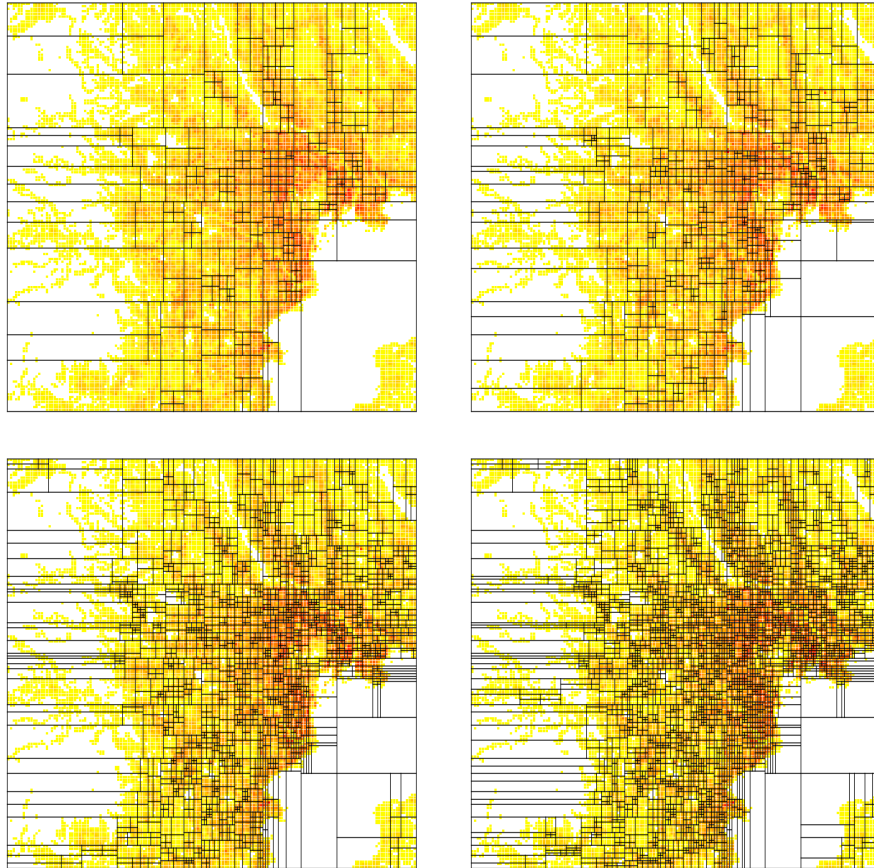


図 2.4: 関東エリアの人口データを用いて生成した GMSQ ネットワークの可視化図 .
左上 : ノード数 500 , 右上 : ノード数 1000 , 左下 : ノード数 3000 , 右下 : ノード数
5000 .

2.2 GMSQ ネットワークのリンク長分布

人口を用いた GMSQ ネットワークのリンク長分布は，べき乗則に従うことネットワークのノード数 N が大きくなるにつれてべき指数が大きくなることが報告されている [5]．そこで，関東の人口データを用いて生成した GMSQ ネットワークのリンク長分布 $P(l_{ij})$ を調べ，べき乗分布

$$f(x) = ax^{-\mu} \quad (2.1)$$

で最小二乗法でフィッティングを行った．その結果，図 2.5 の様に既存研究と同じく，べき乗分布に良くフィットすることが確認できた．表 2.1 には最小二乗法でのフィッティング値で，ノード数 N が大きくなるに連れてべき指数も，既存の研究と同じく大きくなっていることが確認できる．

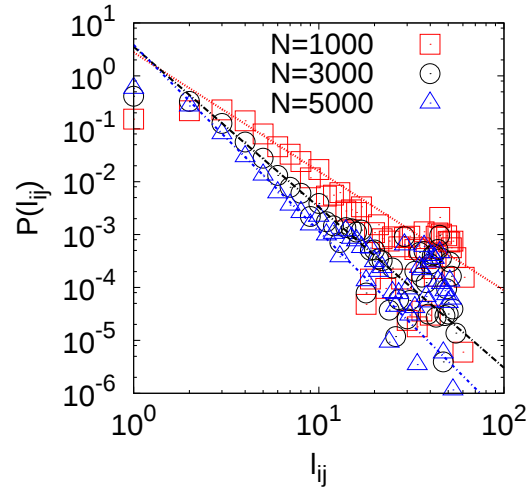


図 2.5: 関東エリアの人口データを用いて生成した GMSQ ネットワークのリンク長分布 $P(l_{ij})$. ネットワークサンプルは 100 とした .

	$N = 500$	$N = 1000$	$N = 2000$	$N = 3000$	$N = 4000$	$N = 5000$
a	1.446	2.779	3.299	3.563	3.89	3.843
μ	1.714	2.248	2.692	3.034	3.335	3.518

表 2.1: GMSQ ネットワークのリンク長分布 $P(l_{ij})$ を最小二乗法でのフィッティングした値 .

2.3 正方格子上的レヴィ飛行

比較のために，正方格子上的レヴィ飛行を考える．GMSQ ネットワークでは周期的境界条件はないので，正方格子でも同様に周期的境界条件なしとした．

2.4 ネットワークでのパケット発生

ネットワークのノードにテリトリーと呼ぶ担当エリアを定め，ノードは自分のテリトリーで発生したパケットを受信し，パケットを蓄積するものとする．ノードのテリトリーの決め方は2次元空間を $L \times L$ の正方格子に分割し，各正方形の面の中心点から最近接のノードがその面を担当するものとした．ネットワークのノードは，自分のテリトリー内のパケットを発生する端末をこの最近接での割り当てにより識別できるものとする．また，人口データを使用したパケット発生では担当エリア内の人口に応じた確率でパケット発生するものとした．また，パケットの宛先ノード d もノード d の担当エリア内の人口に応じた確率で決定する．1 ステップで発生するパケットの数をパラメータ R として与えシミュレーションを行った．

テリトリー面積の頻度を確認したところ，図 2.6 のように各ノードのテリトリー面積は非常に偏りがあることがわかった．

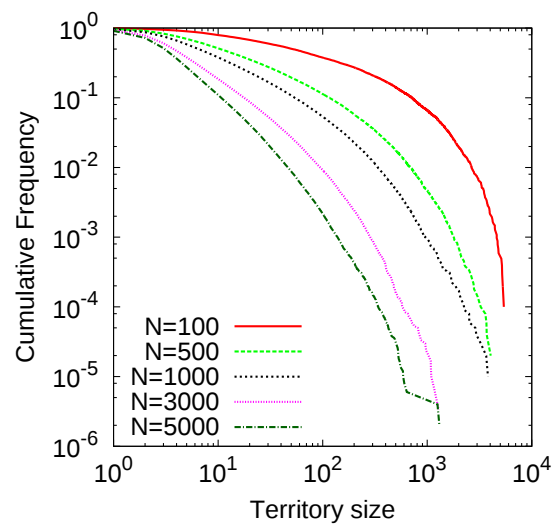


図 2.6: GMSQ ネットワークノードのテリトリーサイズ累積頻度分布．ノードのテリトリー面積は一樣ではなく，偏りがあることがわかる．ネットワークサンプル数は 100 とした．

第3章 経路探索アルゴリズム

本論文で議論する経路探索には複数のメッセージフェリーを利用する。GMSQ ネットワークを使用したシミュレーションでは、メッセージフェリーはネットワーク上をランダムウォークしながら、お互いに情報を交換し、パケットの宛先ノード d を探索するものとする。ランダムウォークを用いるのは、空間的なノード配置が偏っている為に、GMSQ ネットワーク上のリンク長分布がべき乗則に従い、をレヴィ飛行と対応付けが出来るためである [3]。一方、正方格子でのレヴィ飛行では、メッセージフェリーは動く方向をランダムに決定し、移動するホップ数をべき乗分布 $P(l_j) \sim l_j^{-\mu}, 1 < \mu < 3$ で定める。また、GMSQ ネットワークと比較するために、レヴィ飛行ではメッセージフェリーの移動距離は正方格子の境界までで、制限されるものとした。

本研究では経路探索に焦点を当て、メッセージフェリーはパケットを運搬するのではなく、ノードが作成する経路要求 $REQ_{st}(s, d)$ をノードから受け取るものとした。パケットの運搬については経路探索後に扱うものとした。経路要求 $REQ_{st}(s, d)$ を運搬することにより、パケットの大きさに依存しないルーティングが実現できると考えられる。 st は要求が生成されたステップ数、 s は送信ノード、 d は宛先ノードを表す。

経路探索は大まかに以下のものに分けられる。

1. メッセージフェリーはパケット発生しているノード s を見つけ次第、そのパケットの宛先ノード d を他のメッセージフェリーに教え合いながらノード s を探索。

2. メッセージフェリーは宛先ノード d を探索する際，ノードに通過経路を残す．
3. 宛先ノード d を発見したならば，ノードに残っている軌跡を辿りノード s, d 間の経路が含まれたサブグラフを作成する．
4. 作成したサブグラフから Dijkstra 法でノード s, d 間の経路を確立する．

メッセージフェリー，ノードは複数 $REQ_{st}(s, d)$ を保持することができるものとする．

メッセージフェリーは通過したリンク l_{ij} をバッファに保存しながら，ネットワーク上をランダムウォークする．また，通過経路としてメッセージフェリーの1ステップ前のノード情報 $fn(t-1)$ を，各メッセージフェリーの通過ノードに渡しながら移動を行う．これは，宛先を発見した時経路を探索するために行う．

以下，2.1 ではメッセージフェリーの基本動作について，2.2 ではメッセージフェリーが持つ経路要求 $REQ_{st}(s, d)$ を拡散方法する方法について，2.3 では宛先を発見した時の経路を確立する方法について，2.4 では経路確立後にメッセージフェリーとノード上から経路要求 $REQ_{st}(s, d)$ ， $fn(t-1)$ を削除する方法を述べる．

3.1 メッセージフェリーの基本動作

メッセージフェリーはネットワーク上を移動する際に，宛先ノード d を発見する確率を高めるために，図 3.1 のようなバッファを持ち，時刻順に訪問したリンク l_{ij} を重複が無いようにリンクを記録しながら移動する．

3.2 経路要求 $REQ_{st}(s, d)$ の拡散方法

メッセージフェリー A は自分が探している宛先ノード d を他のメッセージフェリー B に教えて，他のノード i にも宛先ノード d の探索を協力してもらう．そのために，

メッセージフェリー A は通過したノード i に経路要求 $REQ_{st}(s, d)$ 渡ししながら移動を探索を行う。これを拡散と呼ぶ。

メッセージフェリーには [探索状態] 経路要求を保持している状態と [フリー状態] 保持としていない状態のどちらかを持つ。図 3.2 のように各状態によってメッセージフェリーはそれぞれに、以下の挙動を行う。

[探索状態]: 複数の経路要求 $REQ_{st}(s, d)$ をメッセージフェリーが持ち、それぞれの宛先ノード d をネットワーク上をランダムウォークし探索する。また、各メッセージフェリーは持っていない経路要求 $REQ_{st}(s, d)$ も同時に探索する。その際、訪れたノードにメッセージフェリーが保持している経路要求 $REQ_{st}(s, d)$ をすべて置きながら探索を行う。

[フリー状態]: 経路要求 $REQ_{st}(s, d)$ を保持しているノードをネットワーク上でランダムウォークしながら探索する。

探索状態のメッセージフェリー A が、図 3.2 や図 3.3 のようにメッセージフェリーが経路要求 $REQ_{st}(s, d)$ をパケット発生ノード以外のノード i にばら撒くことで、他のメッセージフェリー B が経路要求 $REQ_{st}(s, d)$ を発見し易くなるため、メッセージフェリーは持っているすべての経路要求を $REQ_{st}(s, d)$ を通過したノード i に渡ししながら探索を行う。

3.3 メッセージフェリーの経路確立方法

探索状態のメッセージフェリー A が宛先ノード d を発見した時、ノード s, d 間のパケットを届ける経路を確立することができる。その際、できるだけ少ない情報量で経路をもとめられるよう、探索状態のメッセージフェリーはどのノードから来たのか

を表す必要最小限の情報を通過ノードに渡ししながら移動を行うものとする．ノード d からその通過ノードを順に辿ることにより，ノード s, d 間の経路を含んだサブグラフが作成でき，そこから Dijkstra 法などで経路を計算することにより，ネットワークの部分情報のみで経路を知ることが出来る．以下，その処理の流れを説明する．

1. 探索状態のメッセージフェリー A は通過したノードに，前ステップのノード $fn_A(t-1)$ を記憶させながらランダムウォークを行う． $fn_A(t-1)$ はメッセージフェリー A が時刻 $t-1$ にいるノードを表す．また，メッセージフェリー A が前ステップのノード $fn_A(t-1)$ を読み出す際は，メッセージフェリー A の図 3.1 のリンクバッファから読み取るものとする．宛先ノード d を発見するまで (1) の処理を繰り返す．もし，宛先ノード d を発見したならば (2) の処理へ移る．
2. メッセージフェリー A が宛先ノード d を発見した際には，宛先ノード d から $fn(t-1)$ をたどることにより，ノード s, d 間の経路を含んだサブグラフを作成し (3) の処理へ移る．
3. 作成したサブグラフから，ユークリッド距離の意味でリンク長の和が最小となる Dijkstra 法で経路を探索する．

以上の処理により，ノード s, d 間の経路を確立することができる．パケットは経路の確立次第すぐ届けられるものとする．

次に，パケットの経路を確立する際に使用するネットワークの情報をなるべく少なくするために，経路を確立する際に辿る軌跡の情報を制限することを考えていく．辿る軌跡情報は，宛先ノード d を探索していたメッセージフェリーのみの軌跡を辿るものとする．そのために，軌跡情報と経路要求を対応付ける必要がある．そこで， $fn_A(t-1)$ は経路要求 $REQ_{st}(s, d)$ ごとにノードに保存されるものとする．また，経路要求 $REQ_{st}(s, d)$ の宛先ノード d を発見したメッセージフェリーがパケットの経路を確立する際には $REQ_{st}(s, d)$ の合内の要素のみをたどるものとする．以上のよ

うにたどるリンクを制限することにより，経路を確立する際に使用するネットワークの情報が制限され，ネットワークの部分情報のみでの経路探索が可能になると考えられる．

また，経路要求 $REQ_{st}(s, d)$ ごとにノードに保存される各 $f_{n_A}(t-1)$ は，同じメッセージフェリーかつ，同じノードから訪れて重複していた場合はノードに保存しないものとする．

3.3.1 協調的にリンクバッファを利用した宛先ノード d の探索

リンクバッファの利用方法として，リンクバッファからメッセージフェリー A の前ステップのノード情報 $f_{n_A}(t-1)$ を作成するだけでなく，宛先ノード d を発見する際に利用することを考える．もし，経路要求 $REQ_{st}(s, d)$ を持ったメッセージフェリー A が，宛先ノード d 探索を行っている最中に，ノード i を通過したことを考える．図 3.5 のように，メッセージフェリー A がノード i 通過後，他のメッセージフェリー B が i に訪れ経路要求 $REQ_{st}(s, d)$ をノード i から渡された時に，メッセージフェリー B のリンクバッファに宛先ノード d が含まれていた時に以下の処理を行う．

- (1) ノード i, s 間の経路を確立するために，図 3.4 の $f_{n_A}(t+1)$ をたどる処理をノード d からではなく，ノード i から行う．メッセージフェリー A が通過したノードから，ノード i, d 間の経路を含んだサブグラフが作成できる．
- (2) メッセージフェリーのバッファには訪問したリンクが時刻順となって保存されているため，メッセージフェリー B のバッファにノード d が含まれていた場合は，ノード i, d 間の経路は必ず存在する．よって，メッセージフェリー B のリンクバッファを (1) で作成したサブグラフに結合する．これにより，ノード s, d 間の経路を含んだサブグラフが作成できる．
- (3) (2) で作成したサブグラフを，ユークリッド距離の意味でリンク長の和が最小となる Dijkstra 法で経路を求める．

以上の処理を順に行い，複数のメッセージフェリーのリンクバッファを連結して利用することで，ノード s, d 間の経路が求まる．シミュレーションでは，リンクバッファをこの協調的な連携方法を用いて宛先探索に使用した場合と，使用しない場合で比較した．

3.4 経路要求 $REQ_{st}(s, d)$ と $fn_A(t-1)$ の削除

パケットを届け終わった際に，届け終わったパケットが生成した経路要求 $REQ_{st}(s, d)$ と， $REQ_{st}(s, d)$ ごとにノードに保存されている $fn_A(t-1)$ を削除しなければ，ノード，メッセージフェリーに情報が貯まり続けて負荷がかかる．よって，届け終わったパケットに関わるこれら 2 つの情報をメッセージフェリー，ノードから削除する．削除するために，ネットワーク全てに削除情報を流す方法をとった場合，ネットワークにかかる負荷が大きく好ましくない．よって，メッセージフェリーが通過したノード i に $fn_A(t-1)$ を渡した後で，次の時刻にどのノードに移動したかの情報 $fn_A(t+1)$ をノード i に渡すものとした． $fn_A(t+1)$ は $fn_A(t-1)$ と同じく，経路要求 $REQ_{st}(s, d)$ ごとにノード i に保存される．これをノード s から辿ることにより，届け終わったパケットの宛先を探索する際に使用した情報をメッセージフェリー，ノードからすべて削除できる．

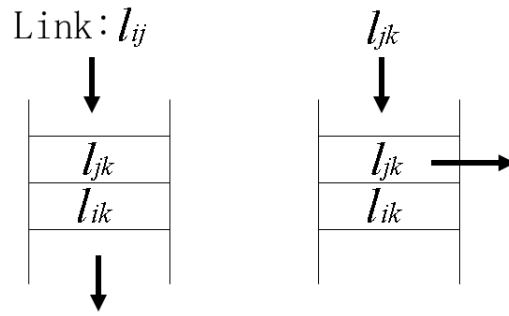


図 3.1: メッセージフェリーのリンク履歴の保存方法．リンクの重複を避るために，それをスタックから破棄し，リンク情報を新たにスタックに追加する処理を行なう．追加するリンク情報と同じリンク情報が，スタックの任意の位置に無い場合は，FILOのスタック構造となる（左）．

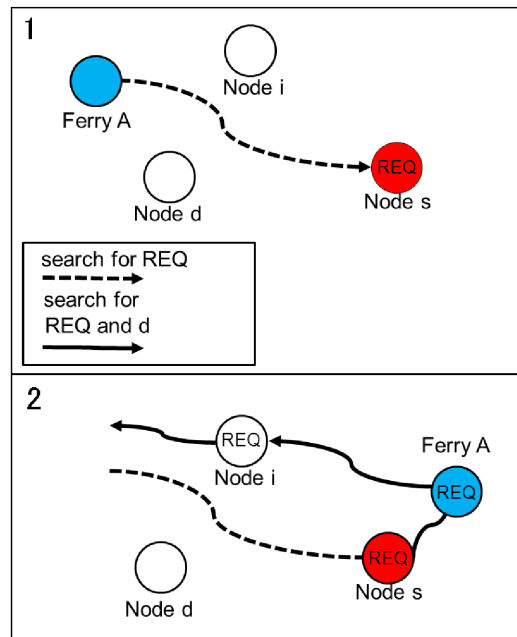


図 3.2: メッセージフェリーの状態による挙動の違い．点線はメッセージフェリーが経路要求を保持しない〔フリーな状態〕で，経路要求 $REQ_{st}(s, d)$ を探索していることを表す．実線はメッセージフェリーが経路要求 $REQ_{st}(s, d)$ を複数持ち，複数の宛先を探索している〔探索状態〕を表す．また，メッセージフェリー A は保持するすべての経路要求 $REQ_{st}(s, d)$ を訪れたノードに渡しなが，複数の宛先と，メッセージフェリー A が持っていない経路要求を探索している．

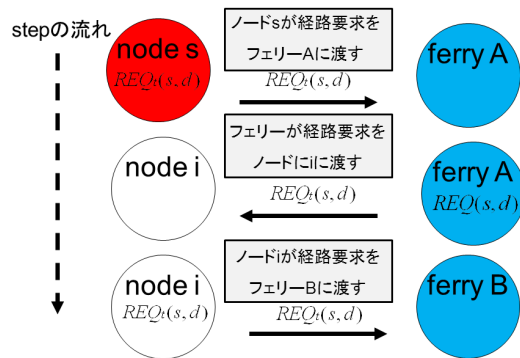


図 3.3: メッセージフェリー同士での , ノードを介した経路要求 $REQ_{st}(s, d)$ 交換の時系列の図 .

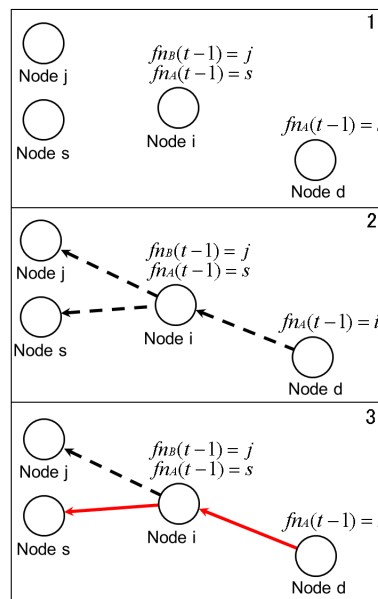


図 3.4: パケットの経路確立方法 . 1 . 経路要求 $REQ_{st}(s, d)$ を保持したメッセージフェリーがノード d を発見 . 2 . ノード d から $fn(t-1)$ をたどり , ネットワークを作成する . 3 . 作成したネットワークから s, d 間の経路を Dijkstra 法で求める .

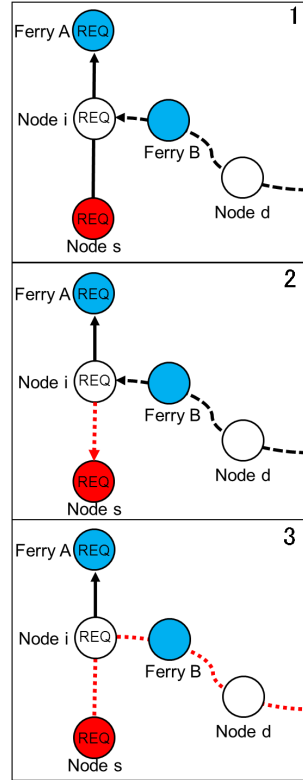


図 3.5: リンクバッファを宛先探索に使用したパケットの経路確立方法．黒色の線種は図 3.2 と同じ意味を表している．1．経路要求 $REQ_{st}(s, d)$ を保持したメッセージフェリー A がノード i を通過．その後，宛先ノード d を通過してリンクバッファに i, d 間のリンクを保存したメッセージフェリー B がノード i に到着．2．ノード i から図 3.4 の方法で $f_n(t-1)$ をたどり，ノード i, s の経路を含んだサブグラフ (赤線) を作成する．3．2 で作成したサブグラフにメッセージフェリー B のリンクバッファを組み合わせる．以上の処理により，ノード s, d 間の経路を含んだサブグラフ (赤線) が作成でき，そこから s, d 間の経路を Dijkstra 法で求めることにより経路が確立出来る．

第4章 シミュレーション結果

本稿では，メッセージフェリーのリンクバッファサイズに制限をつけて，一様な確率でパケットを発生させるシミュレーションを行い，経路探索の探索時間，保存情報量，経路長について述べる．GMSQ ネットワークではノード位置に空間的偏りがあるため，各ノードに一様な確率でパケットを発生しても，空間的な発生位置は非一様になる．また，3.3.1 で述べた協調動作に従ったリンクバッファを宛先探索に使用した場合において，効率性を測るために宛先探索に使用した場合と，使用しない場合との比較も行った．各図のバッファサイズ 0.0 が宛先探索にリンクバッファを使用しない結果を示す．

4.1 GMSQ ネットワークでのシミュレーションの

1 ステップの流れ

GMSQ ネットワークでのシミュレーションの 1 ステップの流れを以下に示す．

- Step 1: パケットを各ノードに一様ランダムまたは，人口に応じた確率 ρ で発生させる．パケットが発生したノード s は，パケットの経路要求 $REQ_{st}(s, d)$ を作成し自身に蓄積する．
- Step 2: m 個の各メッセージフェリーのリンクバッファに，現在のノード j と前ステップのノード i とのリンク l_{ij} を重複が無いように追加する．
- Step 3: あるメッセージフェリー A が経路要求 $REQ_{st}(s, d)$ を保持する [探索状態] の場合は，前ステップのノード情報 $fn_A(t - 1)$ をノードに追加する．

- Step 4: すべてのメッセージフェリーは現在のノード j と経路要求 $REQ_{st}(s, d)$ を交換する .
- Step 5: Step 4 で新たな経路要求 $REQ_{st}(s, d)$ を入手したメッセージフェリーで , リンクバッファの中に宛先ノード d が存在した場合は Step 7 の処理に移る . 存在しない場合は Step 6 の処理に移る .
- Step 6: 経路要求 $REQ_{st}(s, d)$ を持つメッセージフェリーで , 経路要求 $REQ_{st}(s, d)$ のどれかの宛先ノード d が現在のノードならば , Step 7 の処理に移る . そうでなければ , Step 8 の処理に移る .
- Step 7: 経路探索を行いパケットを運搬する . また , $fn_A(t+1)$ を宛先ノード d からたどって行き , たどった先のノード上にいるメッセージフェリーが保持する経路要求 $REQ_{st}(s, d)$ と , たどった先のノードが保持している $REQ_{st}(s, d)$ ごとに保存されている $fn_A(t-1)$ と $fn_A(t+1)$ を削除して Step 8 の処理に移る .
- Step 8: すべてのメッセージフェリーはネットワーク上をランダムウォークし , 次のノードを選択 , 移動する . 移動する際に , メッセージフェリーが経路要求 $REQ_{st}(s, d)$ を保持していた場合は , 次のノード情報 $fn_A(t+1)$ を現在ノード j が保有する $REQ_{st}(s, d)$ ごとの集合に追加する . 決められたステップ数に達した場合は終了する . 達していない場合は Step 1 の処理に移る .

4.2 正方格子でのレヴィ飛行シミュレーションの

1 ステップの流れ

正方格子でのレヴィ飛行シミュレーションの 1 ステップの流れを以下に示す .

- Step 1: パケットを各ノードに一様ランダムまたは、人口に応じた確率 ρ で発生させる。パケットが発生したノード s は、パケットの経路要求 $REQ_{st}(s, d)$ を作成し自身に蓄積する。
- Step 2: m 個の各メッセージフェリーが進む方向をランダムに、ホップ数 l_{ij} をべき乗分布 $P(l_{ij})$ で決定する。
- Step 3: Step 2 で決定した方向に 1 ホップ進む。
- Step 4: メッセージフェリーのリンクバッファに、通過したリンクを重複が無いように追加する。
- Step 5: メッセージフェリーが経路要求 $REQ_{st}(s, d)$ を保持していた場合は、前ノードのノード情報 $fn_A(t-1)$ を通過ノードに追加する。
- Step 6: メッセージフェリーは通過ノードから経路要求 $REQ_{st}(s, d)$ を受け取る。
- Step 7: Step 6 で新たな経路要求 $REQ_{st}(s, d)$ を入手した場合で、かつリンクバッファの中に宛先ノード d が存在した場合は Step 8 の処理に移る。存在しない場合 Step 9 の処理に移る。
- Step 8: 経路要求 $REQ_{st}(s, d)$ を持つメッセージフェリーで、経路要求 $REQ_{st}(s, d)$ のどれかの宛先ノード d が通過ノードならば Step 9 の処理に移る。そうでなければ、Step 10 の処理に移る。
- Step 9: 経路探索を行いパケットを運搬する。また、 $fn_A(t+1)$ を宛先ノード d からたどって行き、たどった先のノード上にいるメッセージフェリーが保持する経路要求 $REQ_{st}(s, d)$ と、たどった先のノードが保持している $REQ_{st}(s, d)$ ごとの $fn_A(t-1)$ の集合と $fn_A(t+1)$ の集合を削除して Step 10 の処理に移る。
- Step 10: メッセージフェリーが経路要求 $REQ_{st}(s, d)$ を保持していた場合は、次のノード情報 $fn_A(t+1)$ を現在ノードの $REQ_{st}(s, d)$ ごとの集合に追加する。Step 2

で決定したステップ数に達した場合，または，格子の境界のノードにメッセージフェリーが辿り着いた場合は Step 1 に移る．達していない場合は Step 3 に移る．

4.3 シミュレーション条件

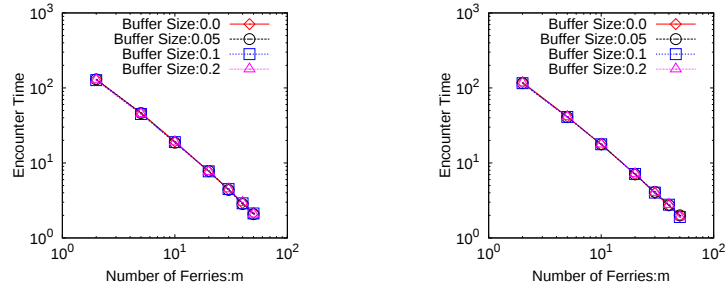
シミュレーションは空間サイズ $L = 160$ ，10,000 ステップ経過したならばシミュレーションは終了とした．ネットワークサンプル数は100とした．レヴィ飛行でメッセージフェリーが進む距離はべき乗分布に従う．よって，どこかでべき乗分布を打ちきる必要がある．シミュレーションでは空間サイズ L に 10,000 をかけた値でべき分布を打ちきった．

人口を用いた GMSQ ネットワークと正方格子上のレヴィ飛行との比較は，2.2 章で示した表 2.1 の傾きの値と，レヴィ飛行のべき分布のパラメータの値を比較を行った．

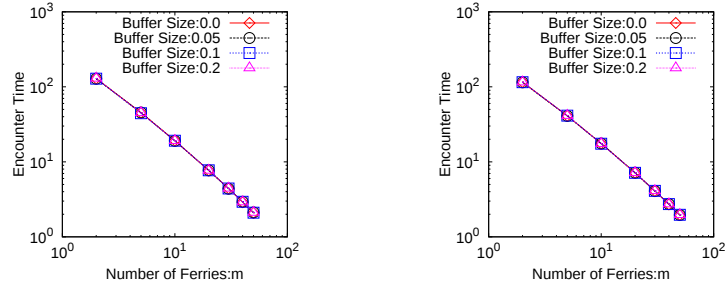
4.4 パケット発生ノードの発見時間

パケットの経路要求 $REQ_{st}(s, d)$ が生成されてからメッセージフェリーが発見するまでの時間をはかり，どの程度の時間が経てば発見されるのかを調べた．その際，ステップ数と，そのステップで m 個のメッセージフェリーが歩いたユーグリッド距離の平均値を測定した．

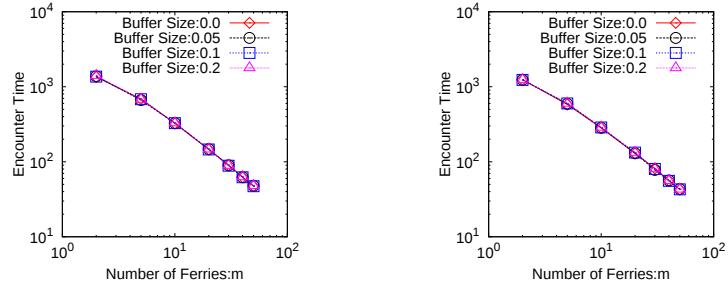
その結果，図 4.1，図 4.2 のようにメッセージフェリー数 m の増加につれて，べき乗的にパケット生成ノードを発見するまでの時間が減少した．ネットワークの探索を高速化する手法として，一様なランダムウォークするウォーカーを k 個使用する多重ランダムウォークという手法があり，特定のネットワークでは全ノード訪問時間 (Cover Time) が 1 つのウォーカーよりも， k 倍高速化できることが報告されている [9]．この影響が現れたと考えられる．



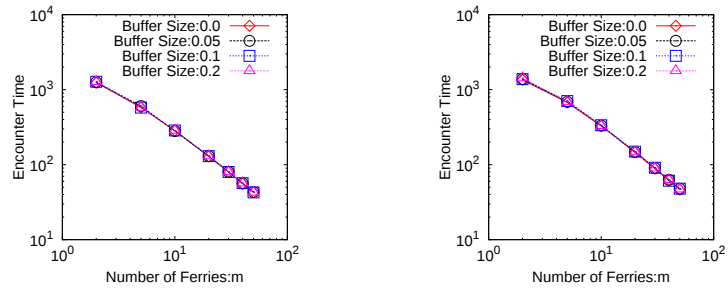
(a) $R=0.01, N=100$



(b) $R=0.1, N=100$

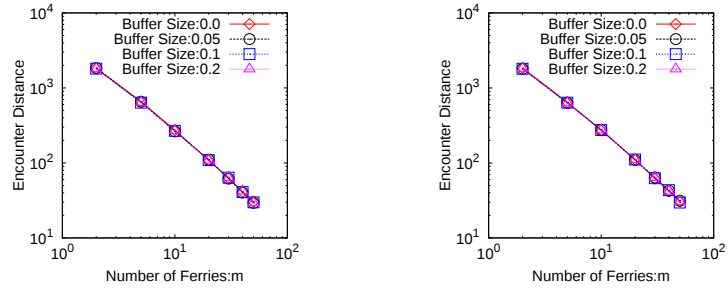


(c) $R=0.01, N=1000$

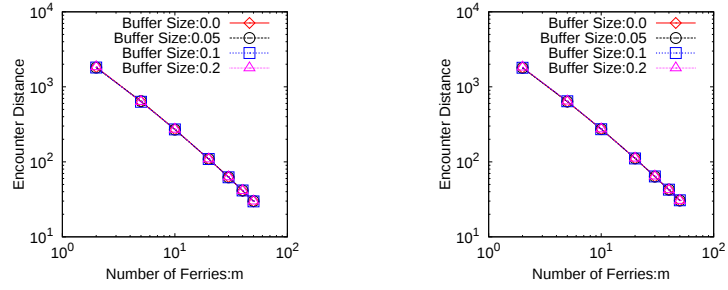


(d) $R=0.1, N=1000$

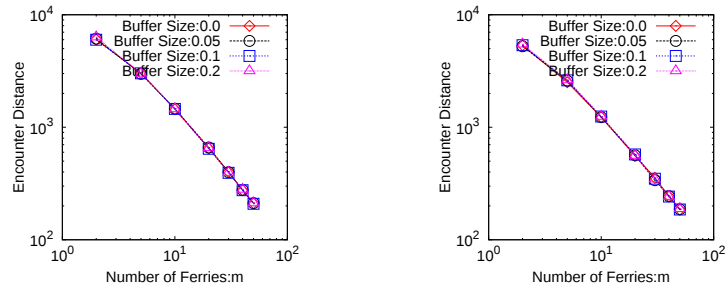
図 4.1: 一様ランダム分割 (左図) と人口を用いた GMSQ ネットワーク (右図) でのメッセージフェリーがパケットを発見するまでのステップ数。縦軸はメッセージフェリーがパケットが作成した $REQ_{st}(s, d)$ を見つけるまでのステップ数で、横軸はメッセージフェリーの数 m となっている。バッファサイズの最大値は全リンク数となっている。



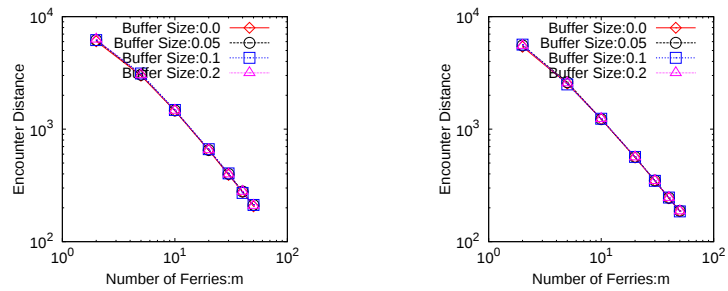
(a) $R=0.01, N=100$



(b) $R=0.1, N=100$

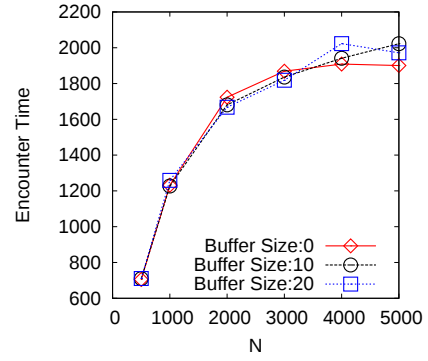
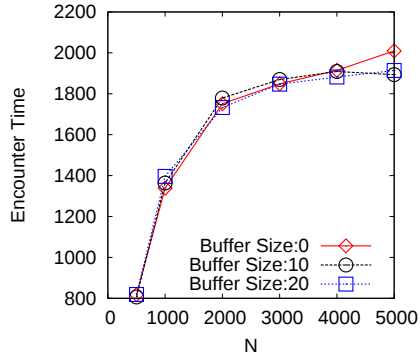


(c) $R=0.01, N=1000$

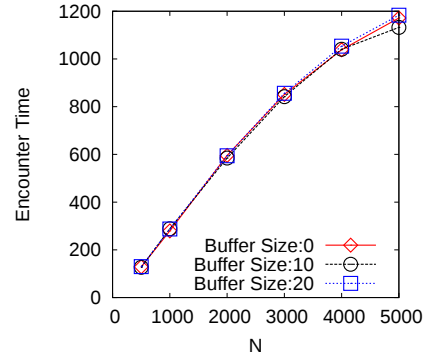
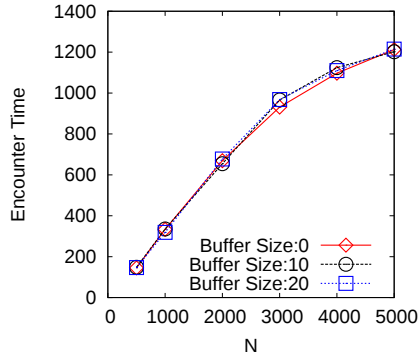


(d) $R=0.1, N=1000$

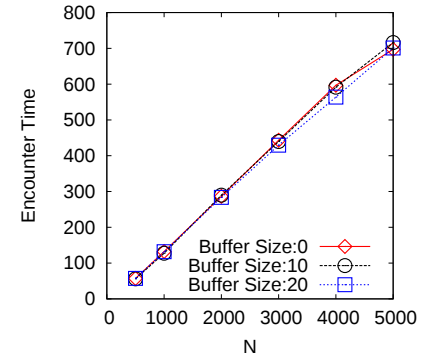
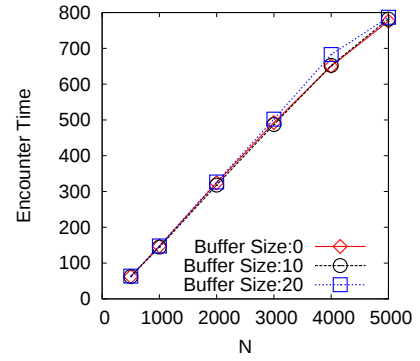
図 4.2: 一様ランダム分割 (左図) と人口を用いた GMSQ ネットワーク (右図) でのメッセージフェリーがパケットを発見するまでに歩いた距離．縦軸は縦軸はパケットが作成した $REQ_{st}(s, d)$ をを見つけるまでに 1 つのメッセージフェリーが歩いた平均距離．横軸はメッセージフェリーの数 m となっている．バッファサイズの最大値は全リンク数となっている．



(a) $m=2$

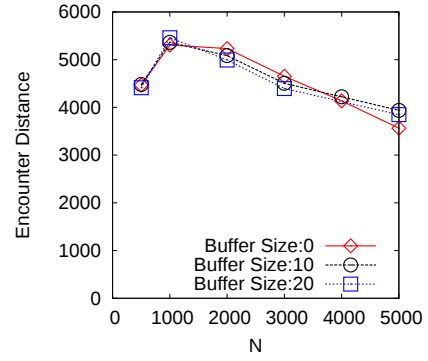
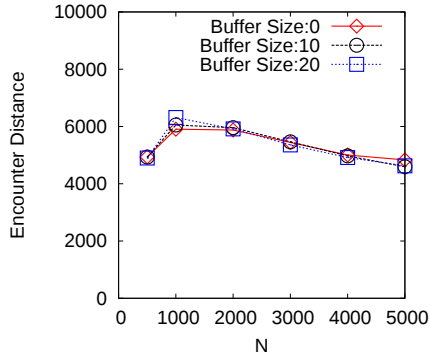


(b) $m=10$

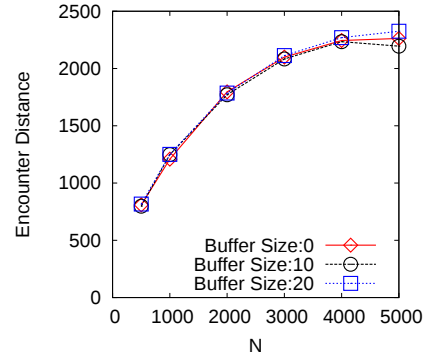
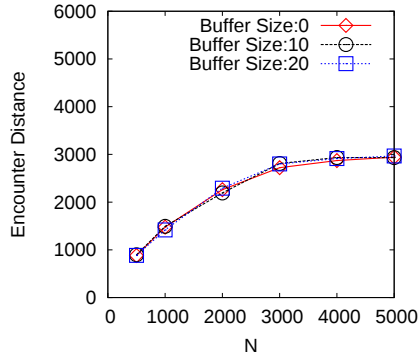


(c) $m=20$

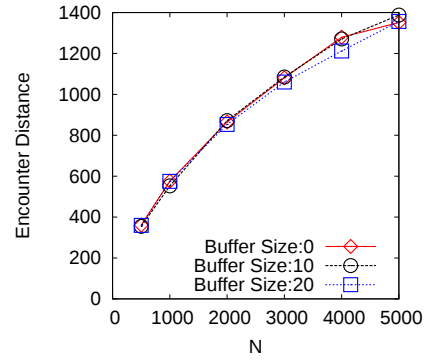
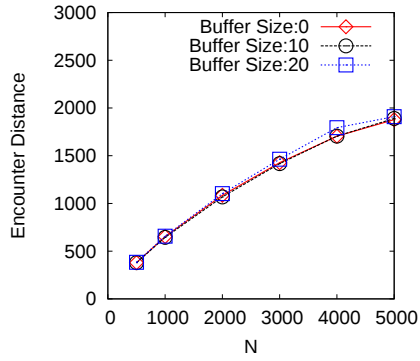
図 4.3: 一様ランダム分割 (左図) と人口を用いた GMSQ ネットワーク (右図) でのメッセージフェリーがパケットを発見するまでのステップ数。縦軸はメッセージフェリーがパケットが作成した $REQ_{st}(s, d)$ をを見つけるまでのステップ数で、横軸はネットワークのノード数 N となっている。



(a) $m=2$

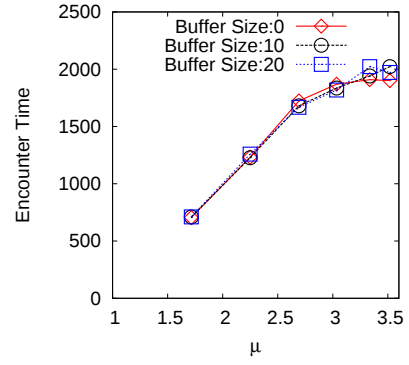
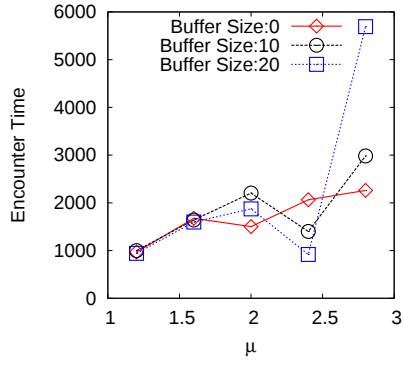


(b) $m=10$

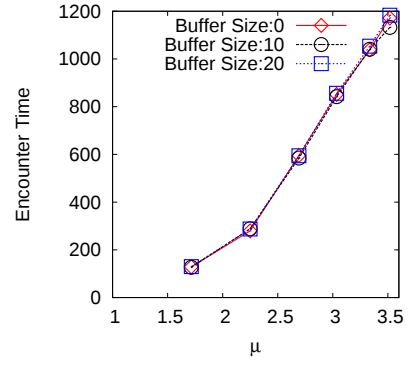
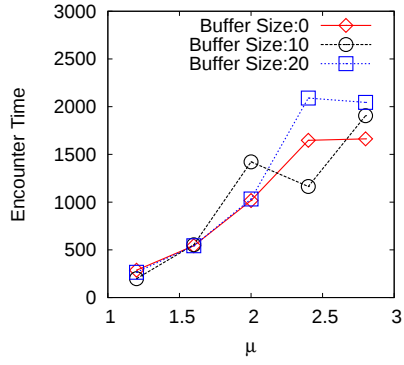


(c) $m=20$

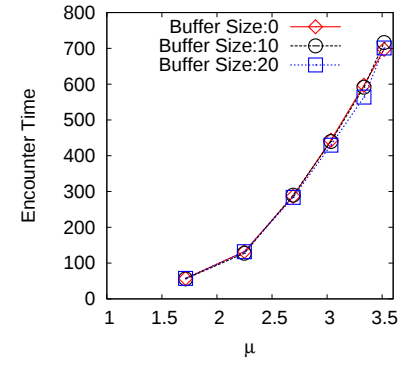
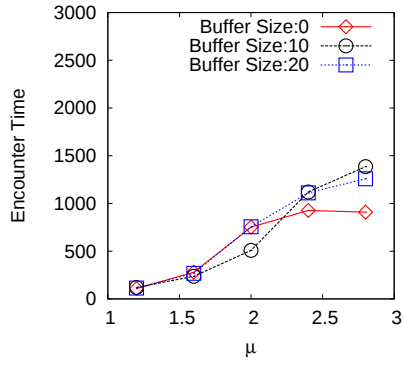
図 4.4: 一様ランダム分割 (左図) と人口を用いた GMSQ ネットワーク (右図) でのメッセージフェリーがパケットを発見するまでに歩いた距離。縦軸はパケットが作成した $REQ_{st}(s, d)$ をを見つけるまでに 1 つのメッセージフェリーが歩いた平均距離。横軸はネットワークのノード数 N となっている。



(a) m=2

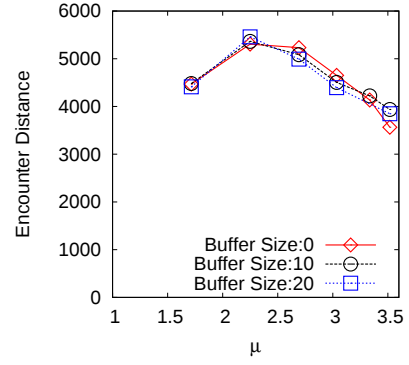
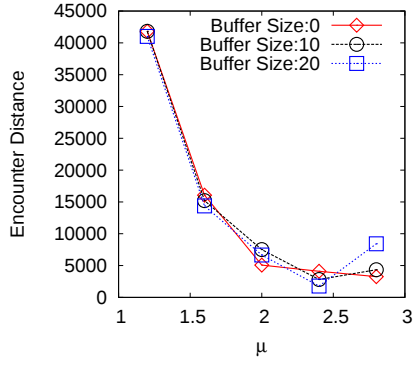


(b) m=10

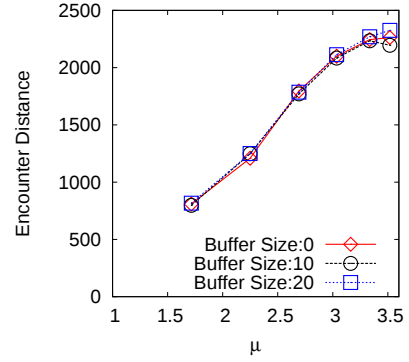
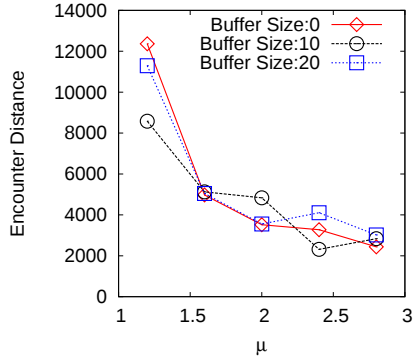


(c) m=20

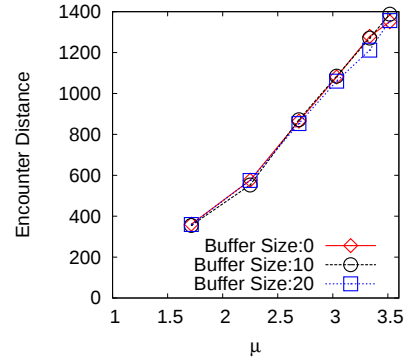
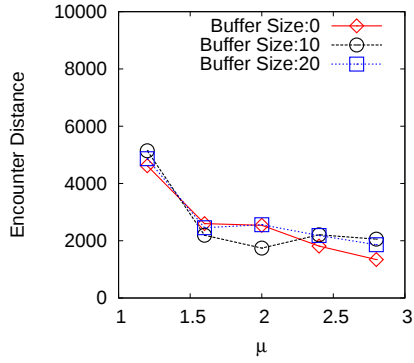
図 4.5: 関東エリアの人口データを用いた正方格子上でのレヴィ飛行探索 (左図) と人口を用いた GMSQ ネットワーク (右図) での, メッセージフェリーがパケットが作成した $REQ_{st}(s, d)$ を見つけるまでのステップ数. 横軸はべき乗分布の傾き μ となっている. 平均パケット発生数 R は 0.01 .



(a) $m=2$



(b) $m=10$



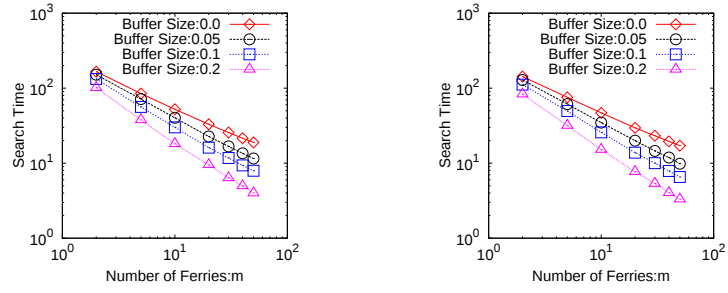
(c) $m=20$

図 4.6: 関東エリアの人口データを用いた正方格子上でのレヴィ飛行探索 (左図) と人口を用いた GMSQ ネットワーク (右図) での, パケットが作成した $REQ_{st}(s, d)$ を見つけるまでに 1 つのメッセージフェリーが歩いた平均距離. 横軸はべき乗分布の傾き μ となっている. 平均パケット発生数 R は 0.01 .

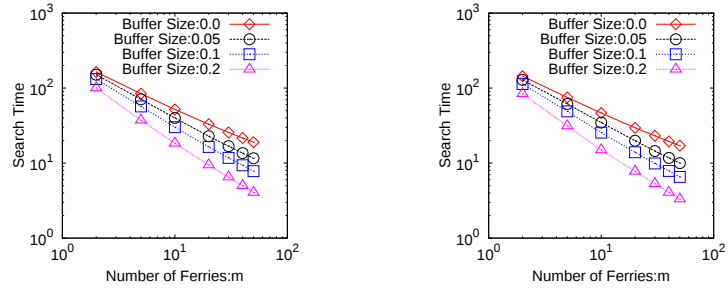
4.5 宛先ノード d の発見効率

メッセージフェリーが経路要求 $REQ_{st}(s, d)$ を発見後から、宛先ノード d を発見するまでの時間と距離をはかり、どの程度の時間が経過するとメッセージフェリーは宛先ノード d が発見できるのかを調べた。その際、距離は最初に経路要求 $REQ_{st}(s, d)$ を発見したメッセージフェリーが歩いた距離と、最初に発見したメッセージフェリーがばら撒いた経路要求 $REQ_{st}(s, d)$ を受け取り協調処理しメッセージフェリーが歩いた平均距離をそれぞれ調べた。

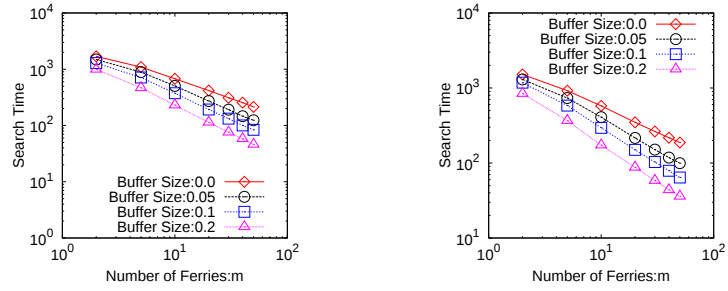
その結果、メッセージフェリー数を増加するに連れて、ステップ数 (図 4.7)、最初に経路要求 $REQ_{st}(s, d)$ を発見したメッセージフェリー A が歩いた距離 (図 4.8)、最初に発見したメッセージフェリー A がばら撒いた経路要求 $REQ_{st}(s, d)$ を受け取り、協調処理し探索を行ったメッセージフェリー B が歩いた距離 (図 4.9) べき乗的な減少を見せた。



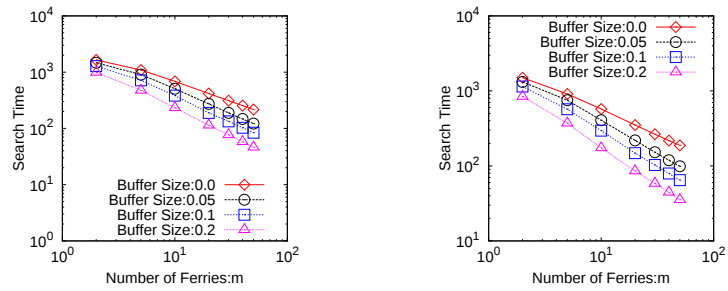
(a) $R=0.01, N=100$



(b) $R=0.1, N=100$

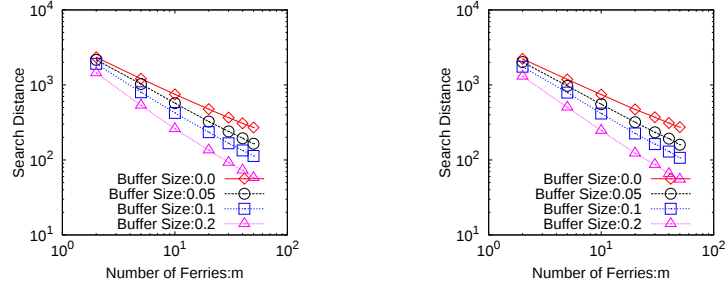


(c) $R=0.01, N=1000$

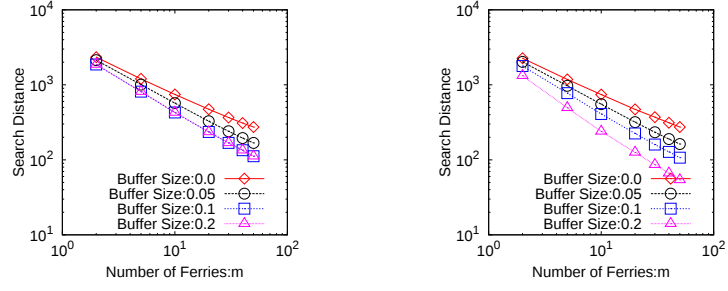


(d) $R=0.1, N=1000$

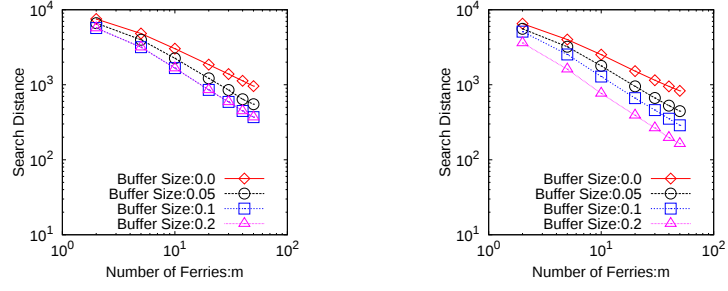
図 4.7: 一様ランダムな分割 (左図) と人口を用いた GMSQ ネットワーク (右図) での ,
メッセージフェリーが宛先ノード d を発見するまでのステップ数 . 横軸はフェリー
の数 m となっている . バッファサイズの最大値は全リンク数となっている .



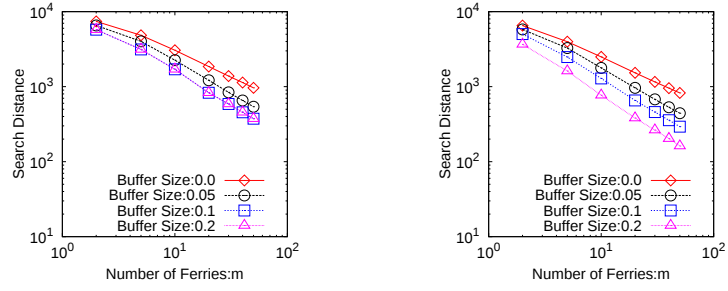
(a) $R=0.01, N=100$



(b) $R=0.1, N=100$

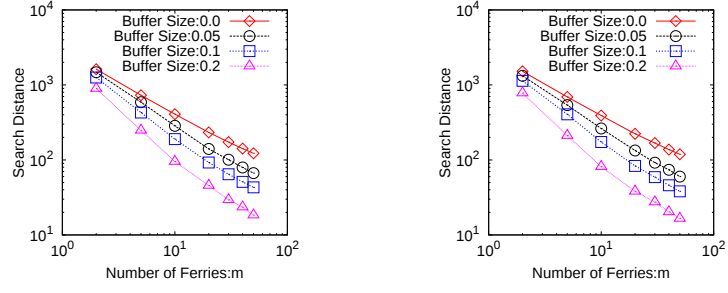


(c) $R=0.01, N=1000$

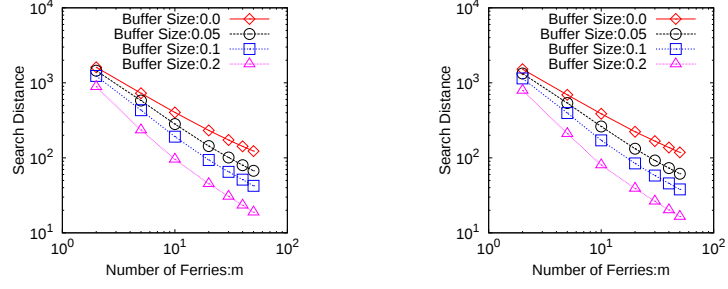


(d) $R=0.1, N=1000$

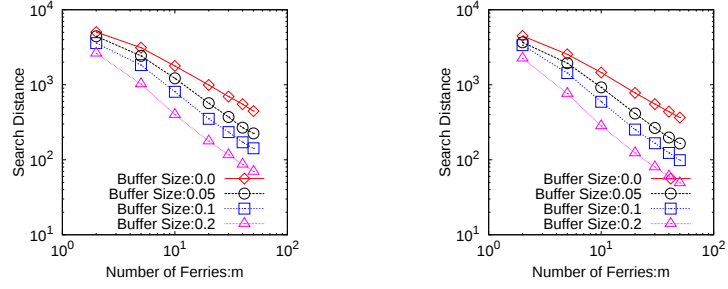
図 4.8: 一様ランダムな分割 (左図) と人口を用いた GMSQ ネットワーク (右図) での , 宛先ノード d を発見するまでに最初に $REQ_{st}(s, d)$ を発見したメッセージフェリーが歩いた距離 . 横軸はフェリーの数 m となっている . バッファサイズの最大値は全リンク数となっている .



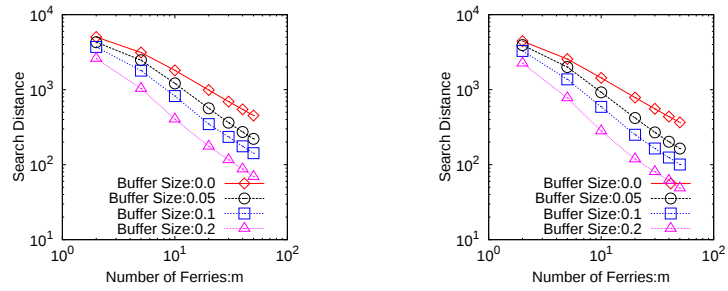
(a) $R=0.01, N=100$



(b) $R=0.1, N=100$

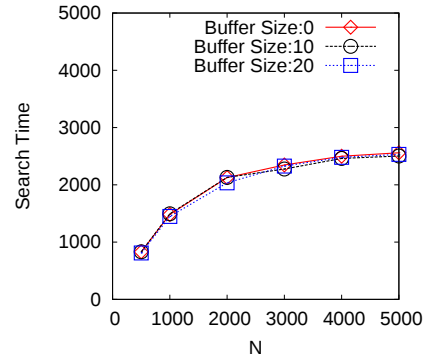
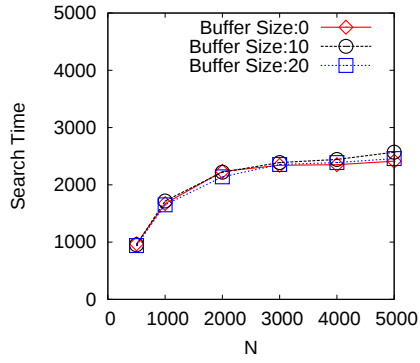


(c) $R=0.01, N=1000$

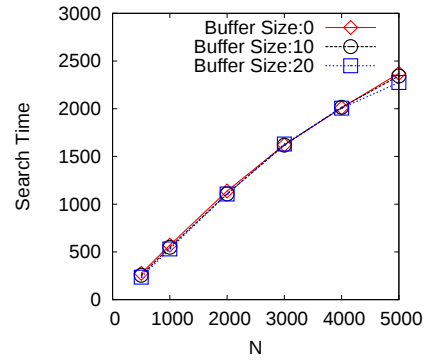
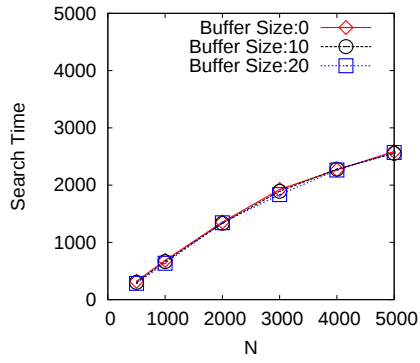


(d) $R=0.1, N=1000$

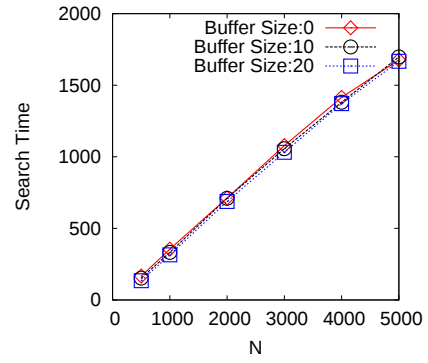
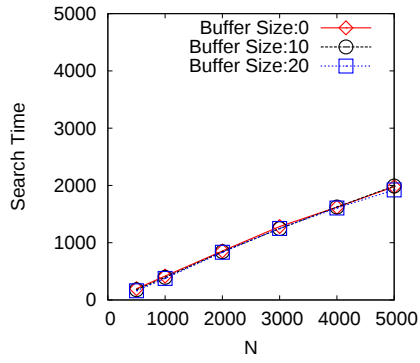
図 4.9: 一様ランダムな分割 (左図) と人口を用いた GMSQ ネットワーク (右図) での, 最初に $REQ_{st}(s, d)$ を発見したメッセージフェリーがばら撒いた $REQ_{st}(s, d)$ を受け取った他のメッセージフェリーが, 宛先ノード d を発見するまでに歩いた平均距離. 横軸はフェリーの数 m となっている. バッファサイズの最大値は全リンク数となっている.



(a) $m=2$

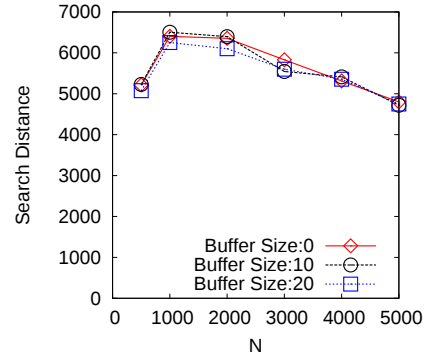
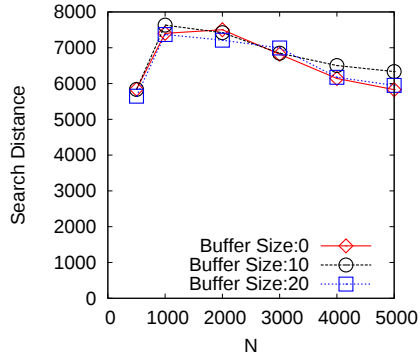


(b) $m=10$

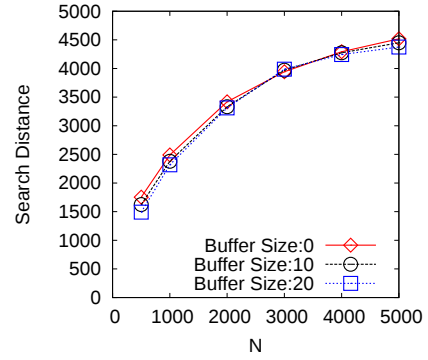
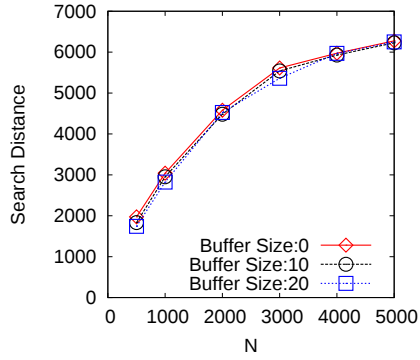


(c) $m=20$

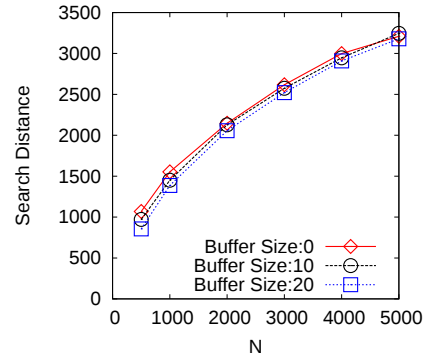
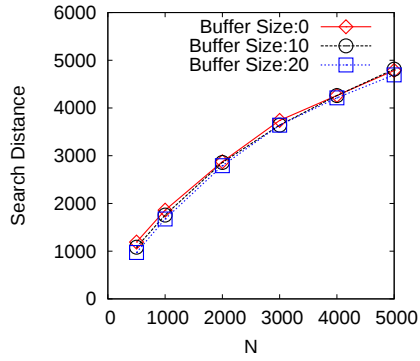
図 4.10: 一様ランダムな分割 (左図) と人口を用いた GMSQ ネットワーク (右図) での, メッセージフェリーが宛先ノード d を発見するまでのステップ数で, 横軸はネットワークのノード数 N となっている. 平均パケット発生数 R は 0.01 .



(a) $m=2$

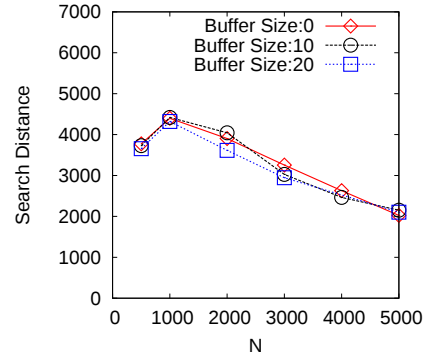
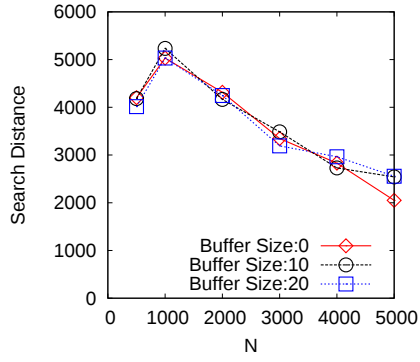


(b) $m=10$

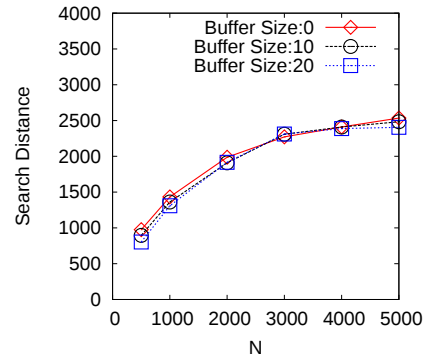
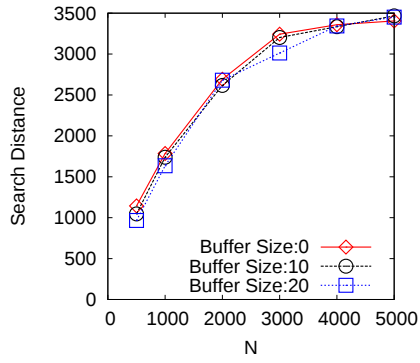


(c) $m=20$

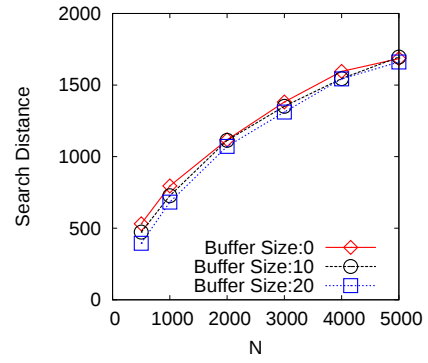
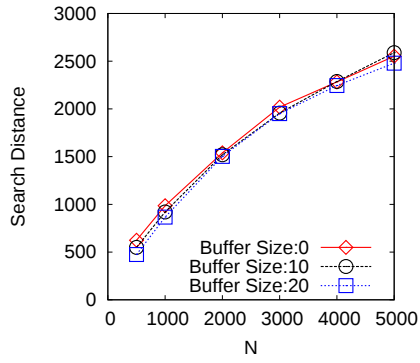
図 4.11: 一様ランダムな分割 (左図) と人口を用いた GMSQ ネットワーク (右図) で、宛先ノード d を発見するまでに最初に $REQ_{st}(s, d)$ を発見したメッセージフェリーが歩いた距離．横軸はネットワークのノード数 N となっている．平均パケット発生数 R は 0.01 ．



(a) $m=2$

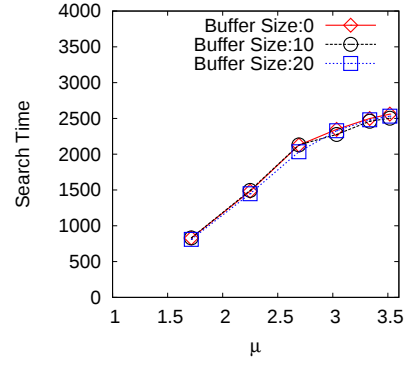
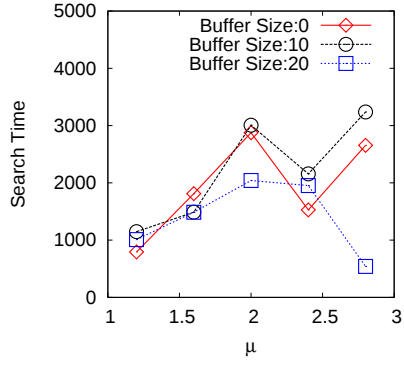


(b) $m=10$

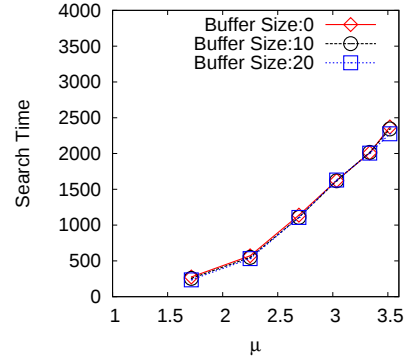
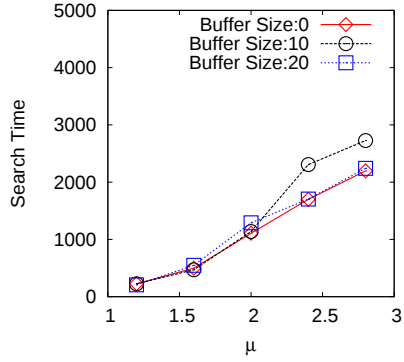


(c) $m=20$

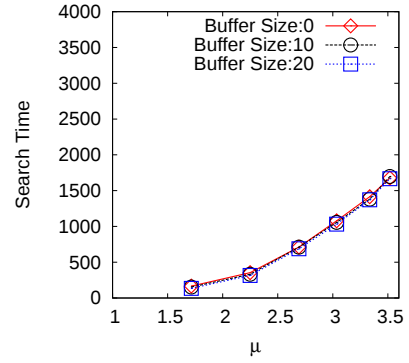
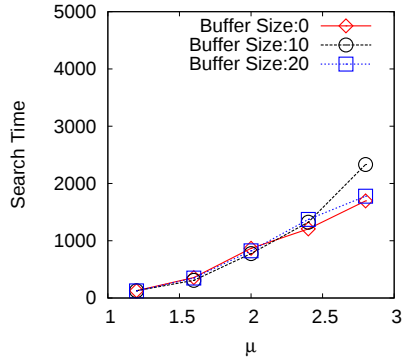
図 4.12: 一様ランダムな分割 (左図) と人口を用いた GMSQ ネットワーク (右図) で、最初に $REQ_{st}(s, d)$ を発見したメッセージフェリーがばら撒いた $REQ_{st}(s, d)$ を受け取った他のメッセージフェリーが、宛先ノード d を発見するまでに歩いた平均距離。横軸はネットワークのノード数 N となっている。平均パケット発生数 R は 0.01。



(a) $m=2$

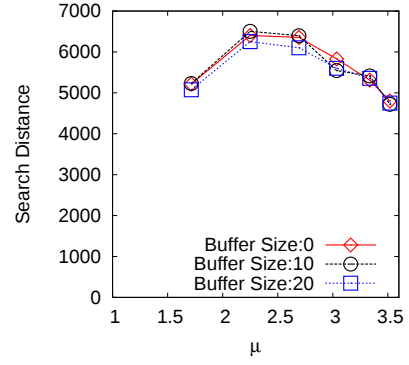
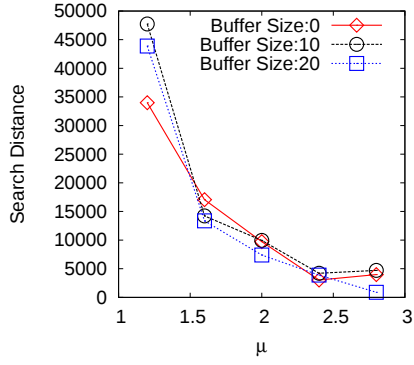


(b) $m=10$

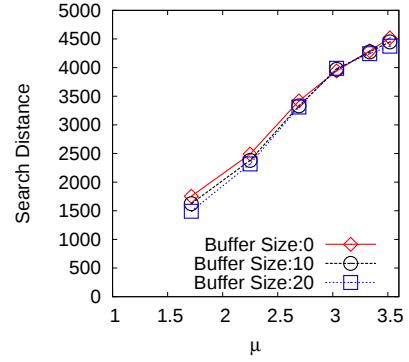
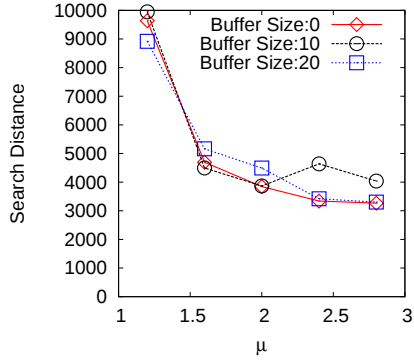


(c) $m=20$

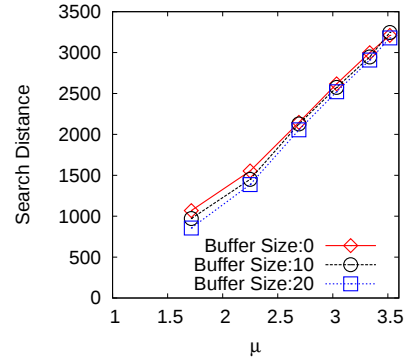
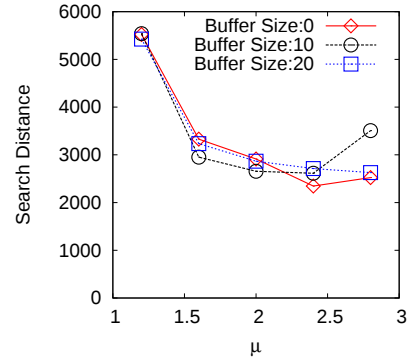
図 4.13: 関東エリアの人口データを用いた正方格子上でのレヴィ飛行探索 (左図) と人口を用いた GMSQ ネットワーク (右図) での, メッセージフェリーが経路要求 $REQ_{st}(s, d)$ を見つけてから宛先ノード d を発見するまでのステップ数. 横軸はべき乗分布の傾き μ となっている. 平均パケット発生数 R は 0.01 .



(a) $m=2$

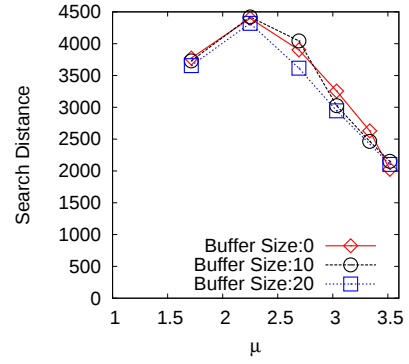
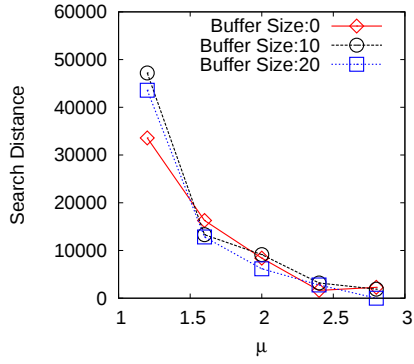


(b) $m=10$

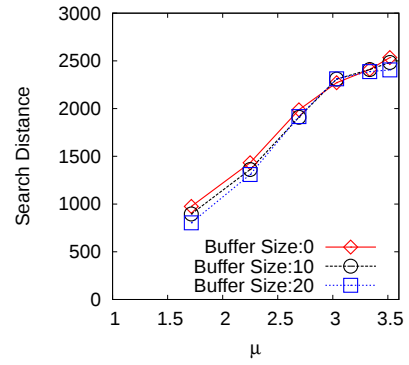
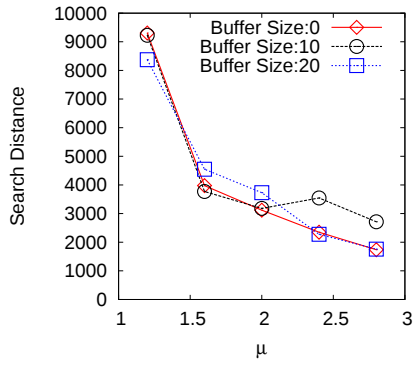


(c) $m=20$

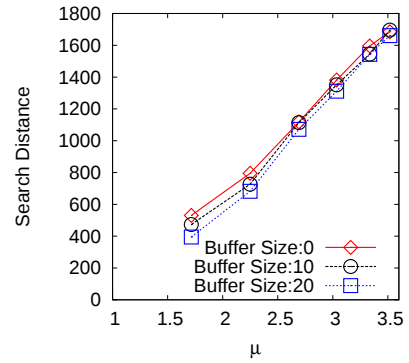
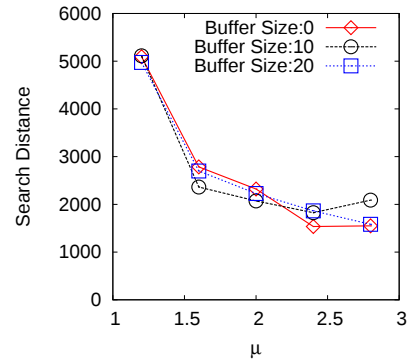
図 4.14: 関東エリアの人口データを用いた正方格子上でのレヴィ飛行探索 (左図) と人口を用いた GMSQ ネットワーク (右図) での, 宛先ノード d を発見するまでに最初に $REQ_{st}(s, d)$ を発見したメッセージフェリーが歩いた距離. 横軸はべき乗分布の傾き μ となっている. 平均パケット発生数 R は 0.01.



(a) $m=2$



(b) $m=10$



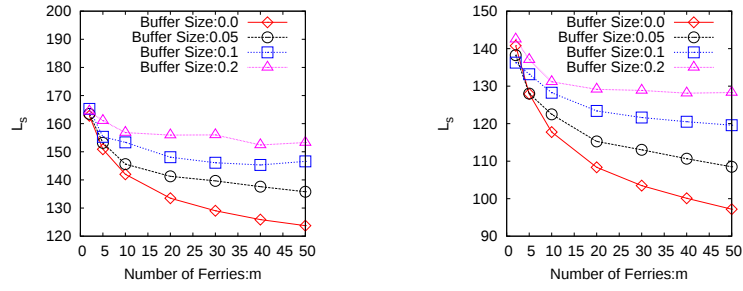
(c) $m=20$

図 4.15: 関東エリアの人口データを用いた正方格子上でのレヴィ飛行探索 (左図) と人口を用いた GMSQ ネットワーク (右図) での, 最初に $REQ_{st}(s, d)$ を発見したメッセージフェリーがばら撒いた $REQ_{st}(s, d)$ を受け取った他のメッセージフェリーが, 宛先ノード d を発見するまでに歩いた平均距離. 横軸はべき乗分布の傾き μ となっている. 平均パケット発生数 R は 0.01.

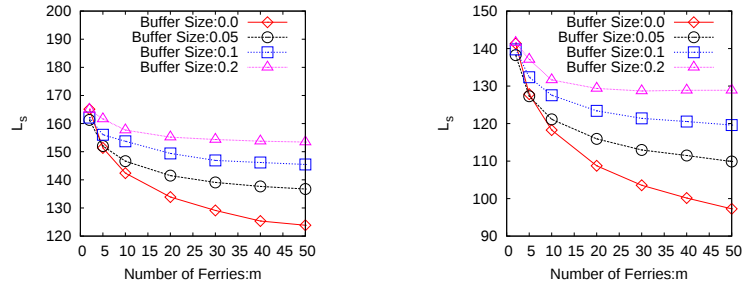
4.6 パケットの経路長

届けられたパケットの経路を調べた。その際、パケットの経路上のリンク長の和 L_s を調べると同時に、ネットワークの最短距離長 L に対しての L_s の比 L_s/L_t も同時に調べた。その結果、図 4.16 のようにメッセージフェリーが増加するにつれて L_s も減少する傾向が見られた。また、同じく図 4.17 のように L_s/L_t もメッセージフェリーが増加するにつれて減少傾向が見られた。次に、経路を確立する際にネットワーク全体のどの程度の部分情報を使用して経路を確立しているかを知るために、経路を確立する際にリンクをどの程度使用しているかを調べた。使用している部分情報量を知ることにより、経路探索にかかる計算量、ネットワークにかかる負荷量を知ることができる。その結果、図 4.22 のようにメッセージフェリーのリンクバッファサイズをある程度の長さに制限すると、経路確立の際に使用するリンクが、リンクバッファを使用しない時よりもネットワークの部分情報のみで経路が確立されていることが確認できた。

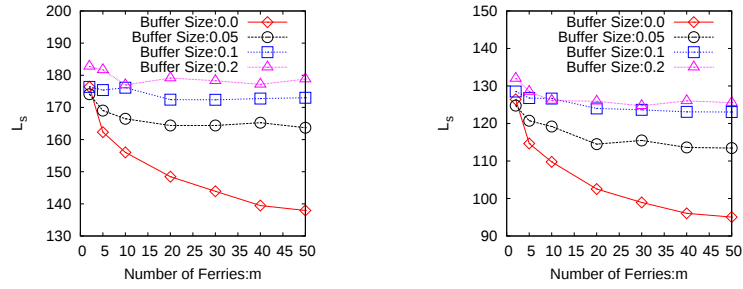
また、メッセージフェリーのリンクバッファを宛先ノード d の探索に使用した場合、使用しない時よりもある程度 L_s は大きくなるが、使用しない時よりも少ない部分情報で経路確立できることが確認できた。



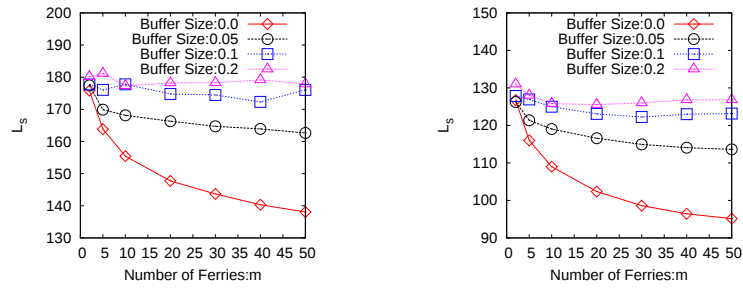
(a) $R=0.01, N=100$



(b) $R=0.1, N=100$

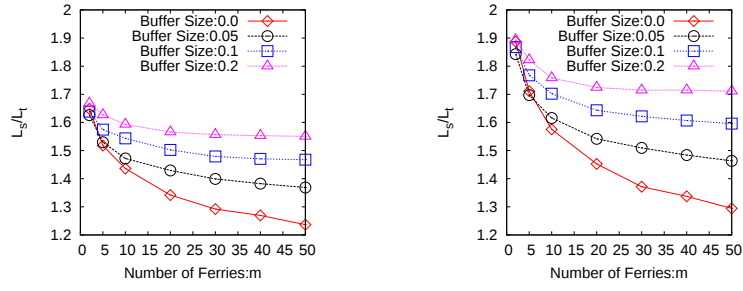


(c) $R=0.01, N=1000$

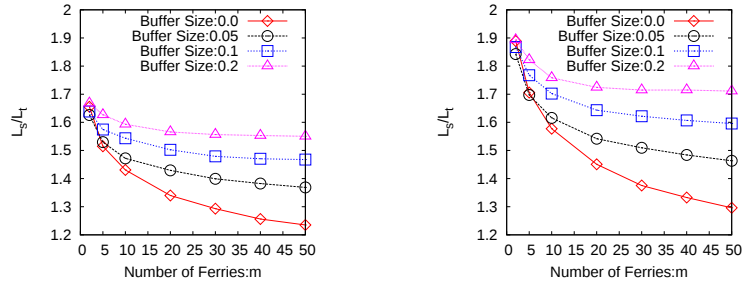


(d) $R=0.1, N=1000$

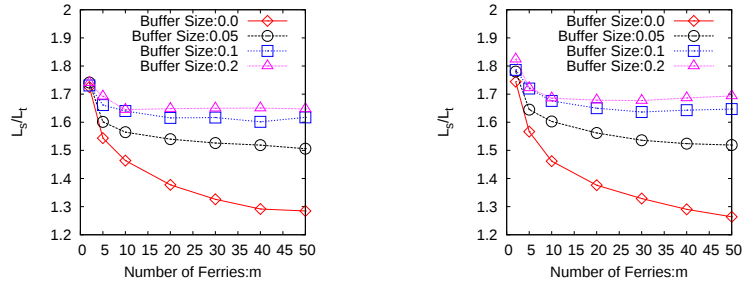
図 4.16: 一様ランダムな分割 (左図) と人口を用いた GMSQ ネットワーク (右図) での, 到達したパケットの経路上のリンク長の和 L_s . 横軸はフェリーの数 m となっている. バッファサイズの最大値は全リンク数となっている.



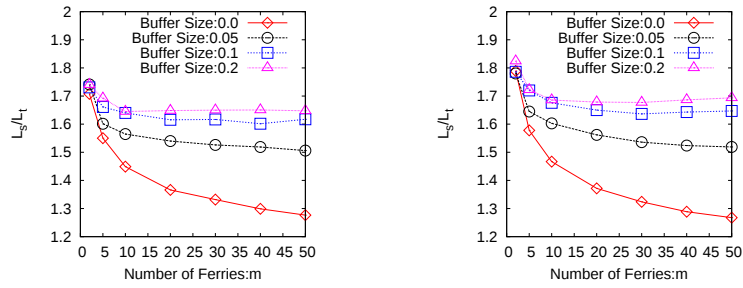
(a) $R=0.01, N=100$



(b) $R=0.1, N=100$

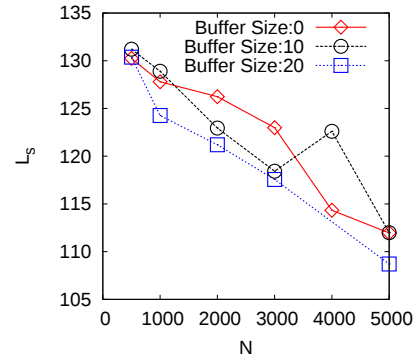
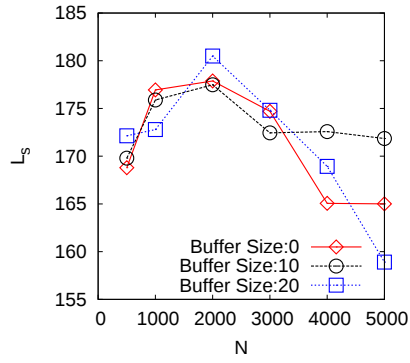


(c) $R=0.01, N=1000$

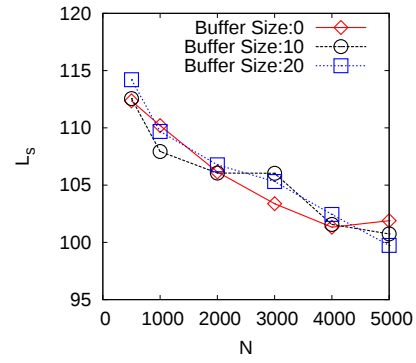
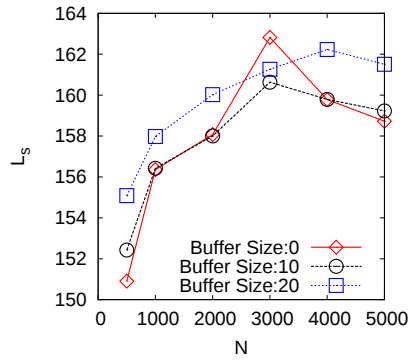


(d) $R=0.1, N=1000$

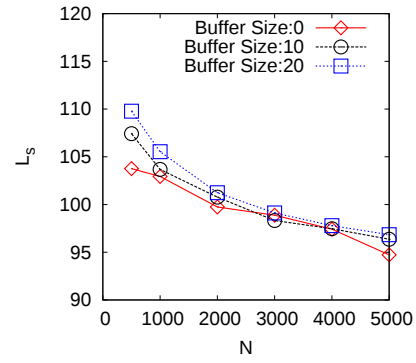
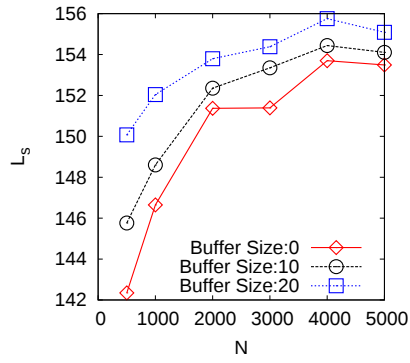
図 4.17: 一様ランダムな分割 (左図) と人口を用いた GMSQ ネットワーク (右図) で、到達したパケットの経路上のリンク長の和 L_s と、パケットの送信ノード s と宛先ノード d 間のネットワークの最短距離長 L_t との比。横軸はフェリーの数 m となっている。バッファサイズの最大値は全リンク数となっている。



(a) $m=2$

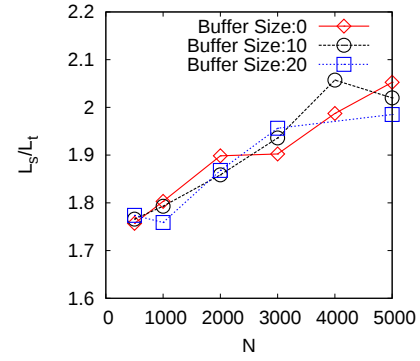
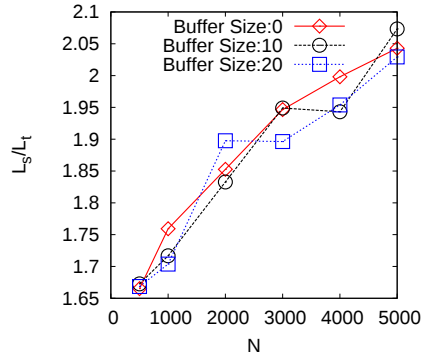


(b) $m=10$

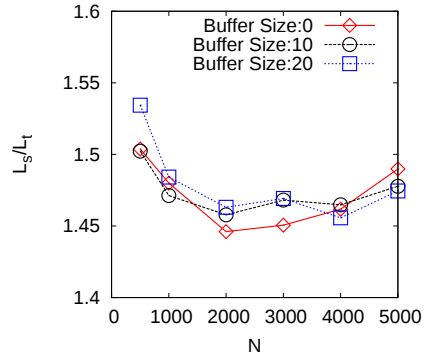
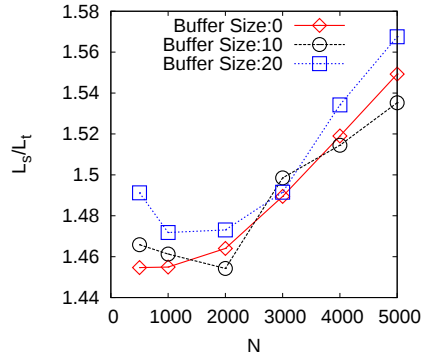


(c) $m=20$

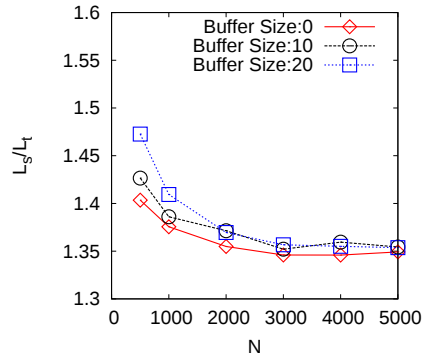
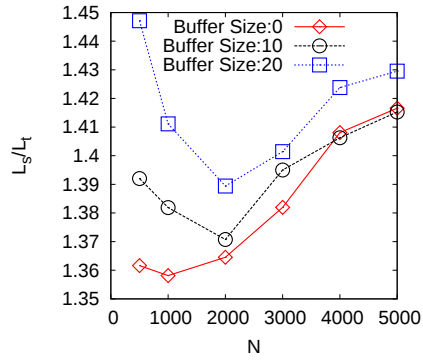
図 4.18: 一様ランダムな分割 (左図) と人口を用いた GMSQ ネットワーク (右図) で
の, 到達したパケットの経路上のリンク長の和 L_s . 横軸はネットワークのノード数
 N となっている. 平均パケット発生数 R は 0.01 .



(a) $m=2$

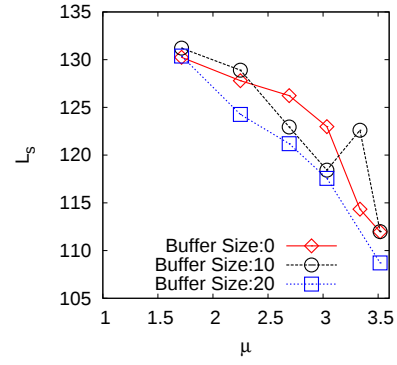
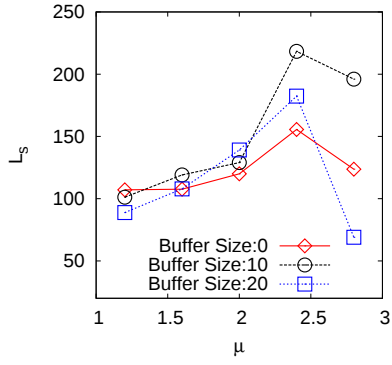


(b) $m=10$

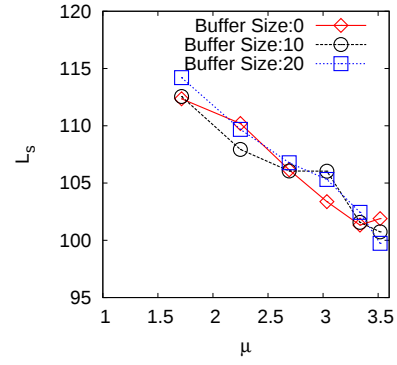
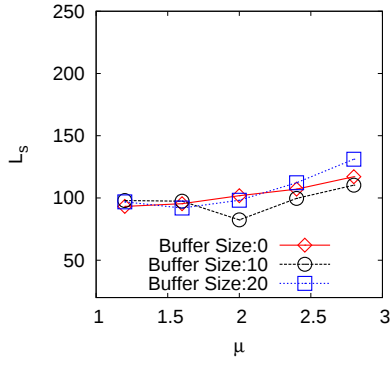


(c) $m=20$

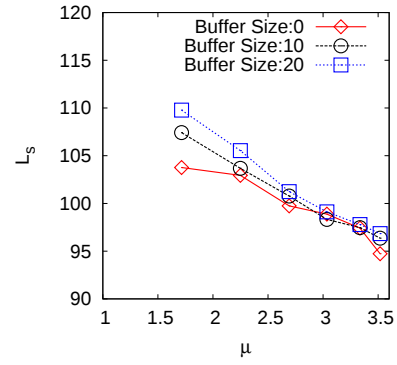
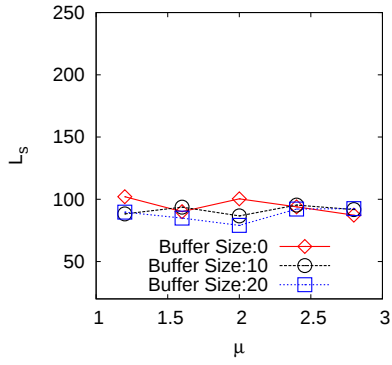
図 4.19: 一様ランダムな分割 (左図) と人口を用いた GMSQ ネットワーク (右図) で、到達したパケットの経路上のリンク長の和 L_s と、パケットの送信ノード s と宛先ノード d 間のネットワークの最短距離長 L_t との比。横軸はネットワークのノード数 N となっている。平均パケット発生数 R は 0.01。



(a) $m=2$

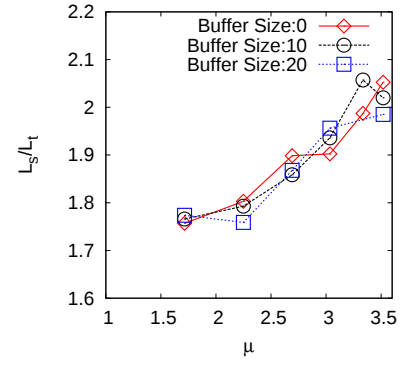
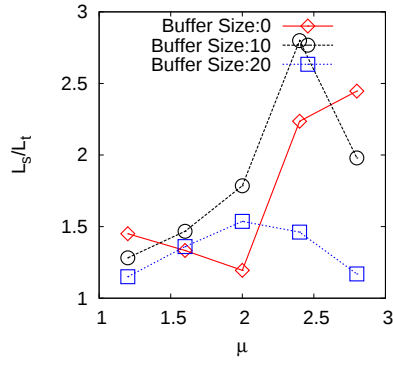


(b) $m=10$

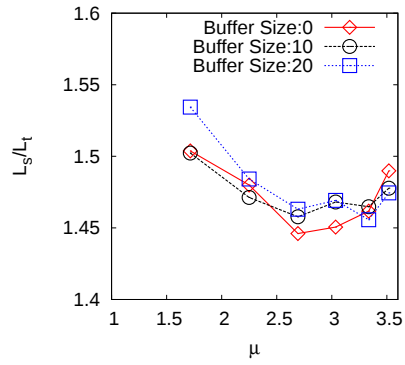
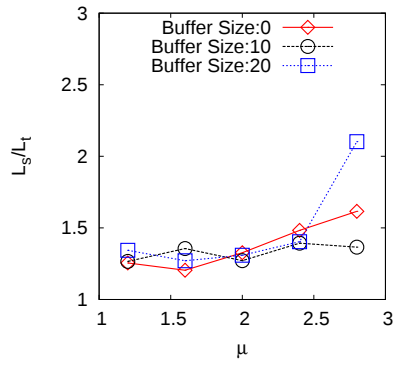


(c) $m=20$

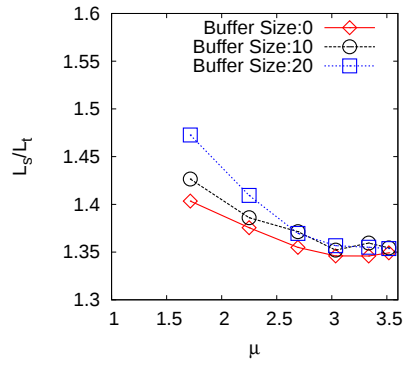
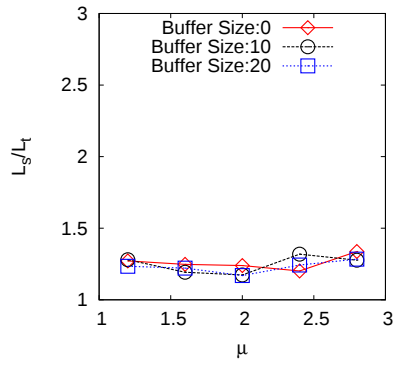
図 4.20: 関東エリアの人口データを用いた正方格子上でのレヴィ飛行探索 (左図) と , 人口を用いた GMSQ ネットワーク (右図) での , 到達したパケットの経路上のリンク長の和 L_s . 横軸はべき乗分布の傾き μ となっている . 平均パケット発生数 R は 0.01 .



(a) $m=2$

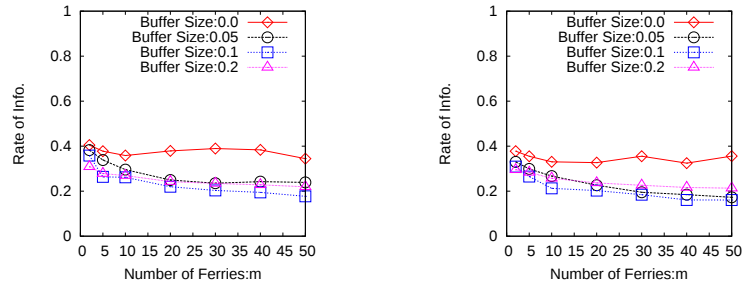


(b) $m=10$

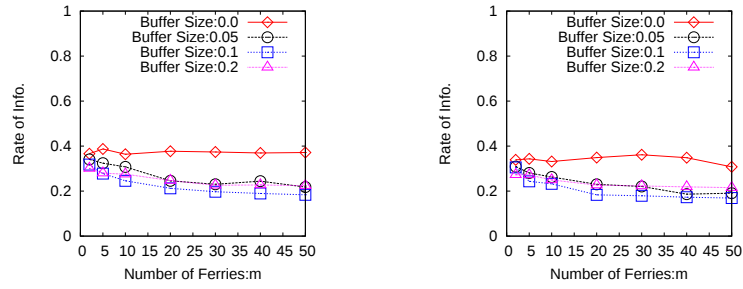


(c) $m=20$

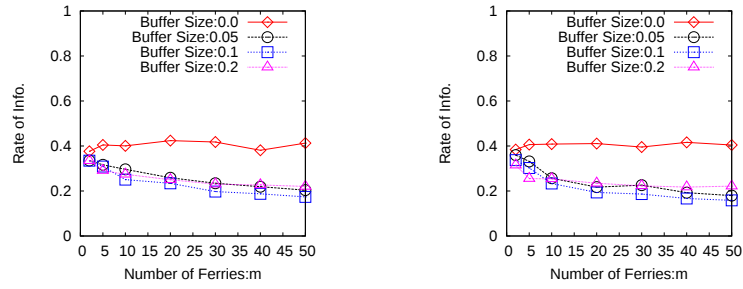
図 4.21: 関東エリアの人口データを用いた正方格子上でのレヴィ飛行探索 (左図) と, 人口を用いた GMSQ ネットワーク (右図) での, 到達したパケットの経路上のリンク長の和 L_s と, パケットの送信ノード s と宛先ノード d 間のネットワークの最短距離長 L_t との比. 横軸はべき乗分布の傾き μ となっている. 平均パケット発生数 R は 0.01 .



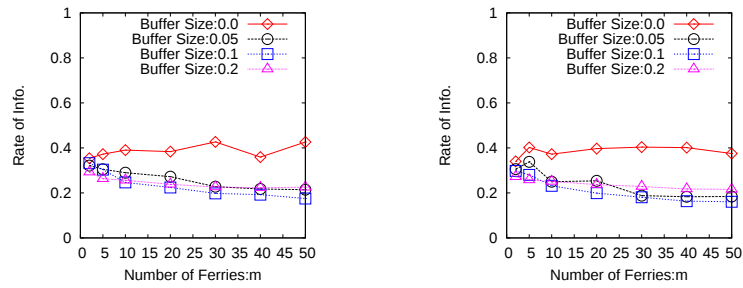
(a) $R=0.01, N=100$



(b) $R=0.1, N=100$

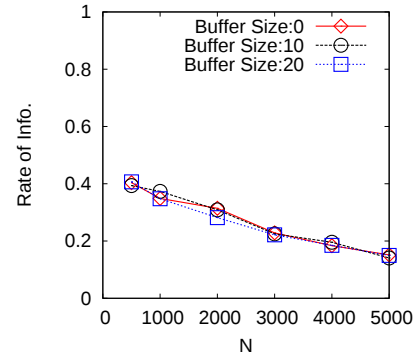
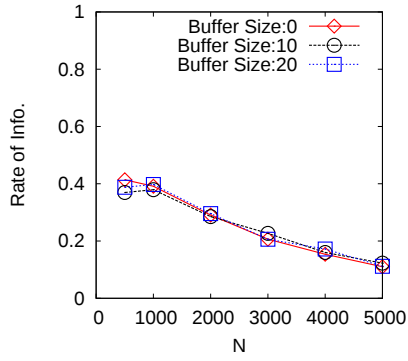


(c) $R=0.01, N=1000$

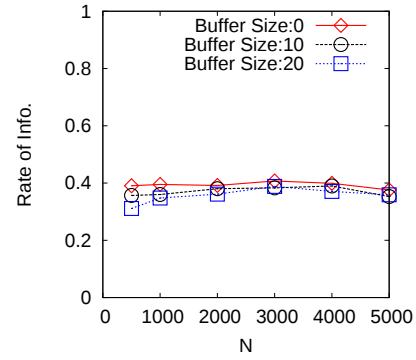
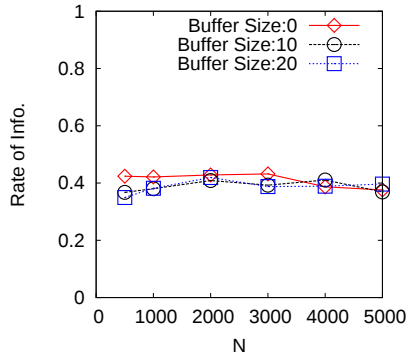


(d) $R=0.1, N=1000$

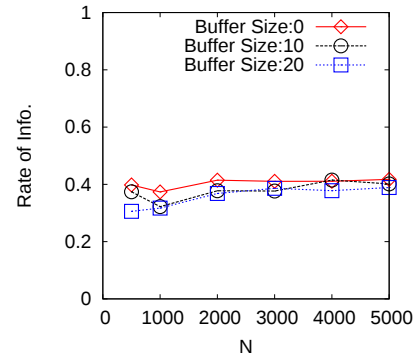
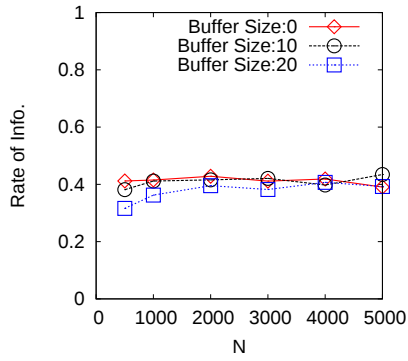
図 4.22: 一様ランダムな分割 (左図) と人口を用いた GMSQ ネットワーク (右図) での, Dijkstra 法でパケットの経路を求める時に使用したリンクの数と, ネットワークの全リンク数との情報量の比. 横軸はフェリーの数 m となっている.



(a) $m=2$

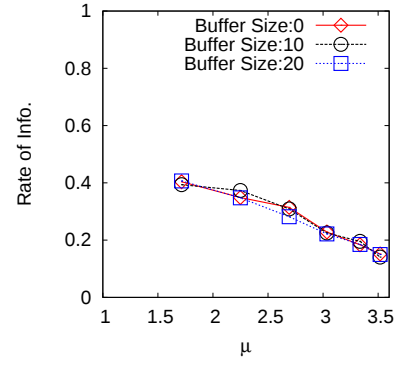
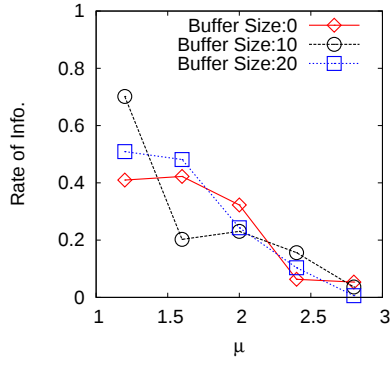


(b) $m=10$

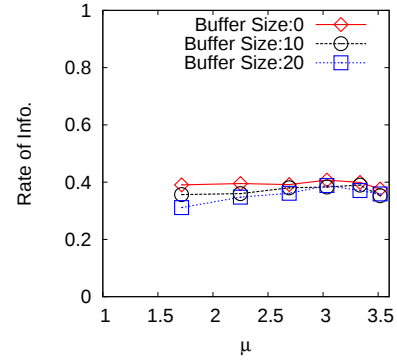
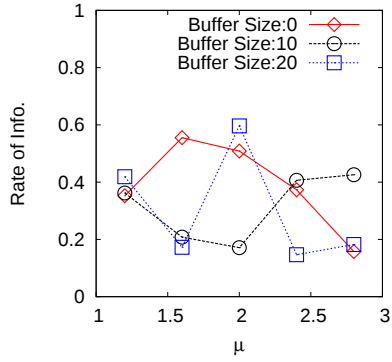


(c) $m=20$

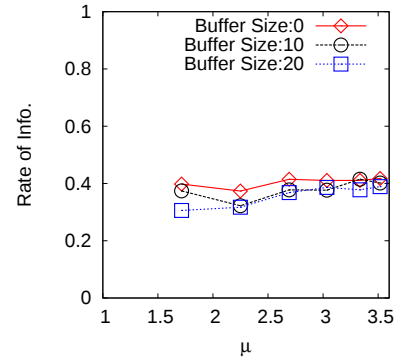
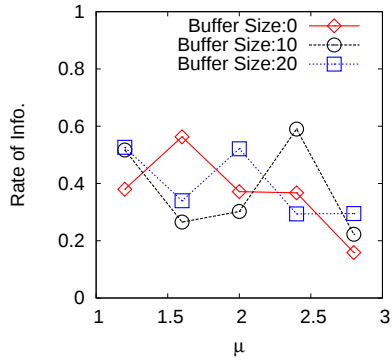
図 4.23: 一様ランダムな分割 (左図) と人口を用いた GMSQ ネットワーク (右図) での, Dijkstra 法でパケットの経路を求める時に使用したリンクの数とネットワークの全リンク数との情報量の比. 横軸はネットワークのノード数 N となっている. 平均パケット発生数 R は 0.01 .



(a) $m=2$



(b) $m=10$



(c) $m=20$

図 4.24: 関東エリアの人口データを用いた正方格子上的レヴィ飛行探索 (左図) と, 人口を用いた GMSQ ネットワーク (右図) での, Dijkstra 法でパケットの経路を求める時に使用したリンクの数とネットワークの全リンク数との情報量の比. 横軸はべき乗分布の傾き μ となっている. 平均パケット発生数 R は 0.01.

第5章 結論

本研究では、複数のメッセージフェリーがノードを介して非同期に情報を交換する方法に着目して、人口分布のように偏ったノードの空間配置を持つネットワーク構造に適した経路探索アルゴリズムを提案し、その効率性を正方格子におけるレヴィ飛行と比較した。その際、パケット発生ノードの発見時間、パケットの宛先ノード d の発見時間、パケットの経路長、経路を確立する際に使用する情報量を調べた。その結果を以下にまとめる。

1. パケット発生ノードの発見時間

パケット発生ノードの発見時間はメッセージフェリー数を増加させることにより、時間と距離双方ともに GMSQ ネットワークではべき乗的に減少した。また、GMSQ ネットワークの分割回数を増加させることにより、時間と距離双方ともに増加した。レヴィ飛行との比較では GMSQ ネットワークの分割回数がある程度制限することにより、レヴィ飛行よりも探索距離が短くなった。

2. 宛先ノード d の発見時間

宛先ノードの発見時間 d も上記 1 のパケット発生ノードの時間と同じ傾向を見せた。これにより、ノードを介した非同期な協調処理が上手く働いていることが確認できたと言える。メッセージフェリー数を増加させることにより、時間と距離の双方ともに GMSQ ネットワークではべき乗的に減少し、GMSQ ネットワークの分割回数を増加させることにより、時間と距離双方ともに増加した。レヴィ飛行との比較では GMSQ ネットワークの分割回数がある程度制限することにより、時間と距離ともにレヴィ飛行よりも良い傾向を示した。

3. パケットの経路長

人口を用いた GMSQ ネットワークでの、到達したパケットの経路上のリンク長の和 L_s と、パケットの送信ノード s と宛先ノード d 間のネットワークの最短距離長 L_t との比 L_s/L_t を調べた。 L_s とネットワークの最短距離 L_t を比較することにより、提案手法の有効性を確認した。結果、メッセージフェリーの数 m を増やすにつれて、経路の比 L_s/L_t の値も小さくなる傾向があることが確認できた。

4. 経路確立に使用する情報量

使用している部分情報量を知ることにより、経路探索にかかる計算量や、ネットワークにかかる負荷量を知ることができる。経路を確立する際にネットワーク全体のどの程度の部分情報を使用しているかを知るために、経路を確立する際にリンクをどの程度使用しているかも同時に調べた。その結果、ネットワークの部分情報で経路確立が行われていることと、メッセージフェリーのリンクバッファサイズをある程度の長さに制限すると、経路確立の際に使用するリンクが、リンクバッファを使用しない時よりも少ない部分情報のみで経路が確立されていることが確認できた。

本研究により、複数のメッセージフェリーが宛先をあらかじめ知らなくとも、非同期に宛先を教え合い協調しながら宛先を探索することにより、探索時間は大幅な減少を見せることを示すことができた。また、レヴィ飛行的な構造が埋め込まれたネットワークを使用してメッセージフェリーが経路を決定する場合、ネットワークの分割回数をある程度制限すればよりレヴィ飛行よりも良い傾向を示した。

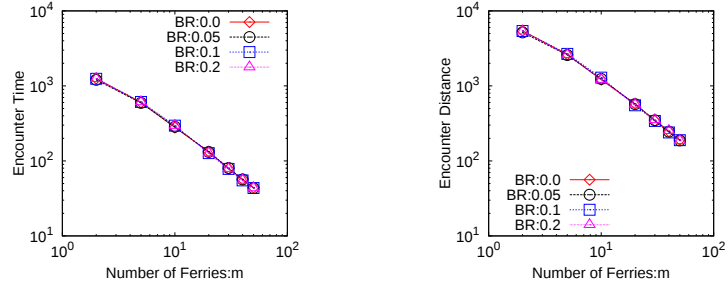
参考文献

- [1] Wenrui Zhao and Mostafa H.Ammar, Proc. of 9th IEEE Workshop on FTDCS 2003, pp.308-314, May 2003.
- [2] Wenrui Zhao, Mostafa H.Ammar and Ellen Zegura, MobiHoc'04, May 24-26, 2004, Roppoingi, Japan.
- [3] Minsu Shin, Seongik Hong and Injong Rhee, CHANTS'08, September 15, 2008.
- [4] S.H.Yook, H.Jeong and A.-L.Barabási, PNAS 99(21), 13382, 2002.
- [5] Yukio Hayashi, Proc. of the the 4th International Conference on Adaptive and Self-adaptive Systems and Applications, pp.63-68, ISBN:978-1-61208-219-6, Nice, France, July 23-26 (2012).
- [6] Wenrui Zhao, Mostafa H.Ammar and Ellen Zegura, IEEE Infocom 2005, Mar, 2005.
- [7] Yukio Hayashi and Yasumasa Ono, Phys. Rev. E 82, 016108, 2010.
- [8] Yukio Hayashi, Rethinking of Communication Requests, Routing, and Navigation Hierarchy on Complex Networks -for a Biologically Inspired Efficient Search on a Geographical Space-, In "Introduction to Routing Issues," Chapter X, Arshin Rezazadeh(ed), iConcept Press, 2012.

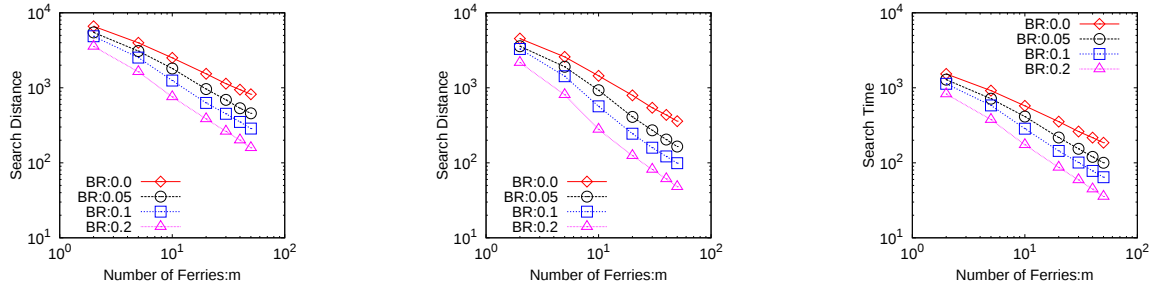
- [9] Noga Alon, Chen Avin, Michal Koucky, Gady Kozma, Zvi Lotker and Mark R. Tuttle, Many Random Walks Are Faster Than One, SPAA '08, June 1416, 2008, Munich, Germany.
- [10] G.M.Viswanathan et al. The Physics of Foraging, Cambridge Univ.Press, 2011.
- [11] G . M . Viswanathan, S. V. Buldyrev, S. Havlin, M. G. E. daLuz, E. P. Raposo, and H. E. Stanley, Nature (London) **401**, 911(1999).
- [12] M. C. Santos, G. M. Viswanathan, E. P. Raposo, and M. G. E. da Luz, PHYSICAL REVIEW E **72**, 046143(2005).

付 録 A 周期的境界条件を用いた 正方格子上的レヴィ飛行

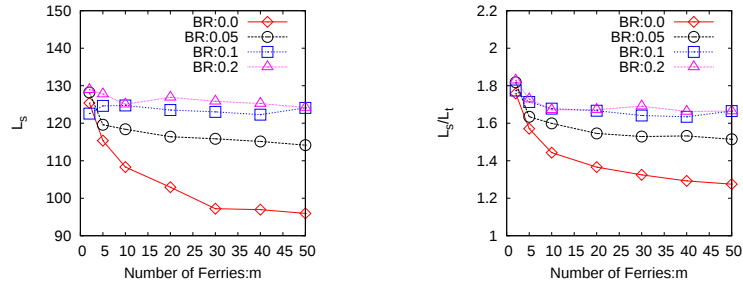
周期的境界条件を用いた正方格子上的レヴィ飛行のシミュレーション結果を以下に示していく．1 ステップでの平均パケット発生数 $R = 0.01$, 空間サイズ $L = 160$, 10,000 ステップ経過したならばシミュレーションは終了とした．シミュレーション結果は全て 50 平均．



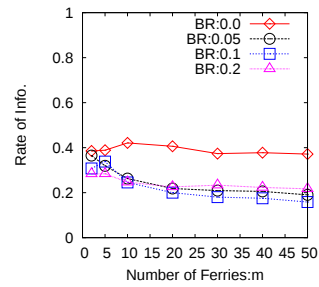
(a) 遭遇時間 (左) 距離 (右)



(b) フェリー A の宛先ノード d 探索距離 (左), フェリー B の探索距離 (中), 探索時間 (右)

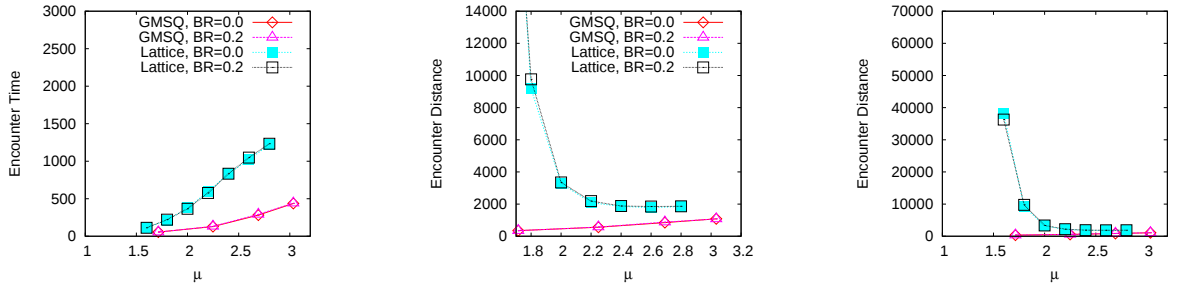


(c) パケットのパスのリンク長の和 (左), ネットワークの最短距離長との比 (右)

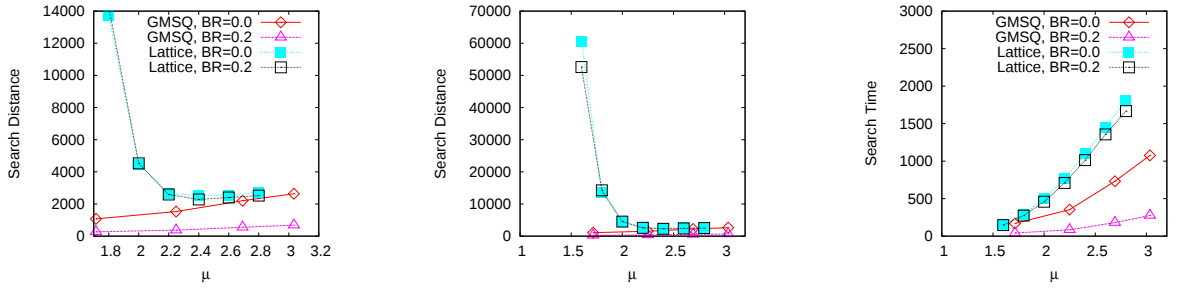


(d) 経路確立の際に使用したリンクの数とネットワークの全リンク数との比

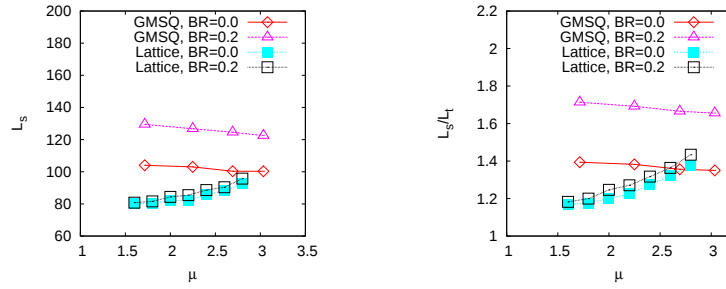
図 A.1: R0.01, N=1000, GMSQ.



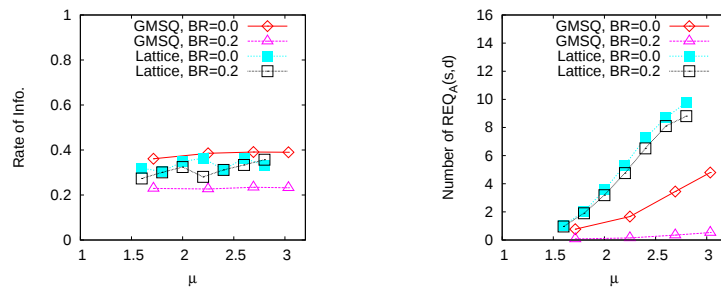
(a) 遭遇時間 (左) 距離 (中, 右)



(b) フェリー A の宛先ノード d 探索距離 (左, 中), 探索時間 (右)

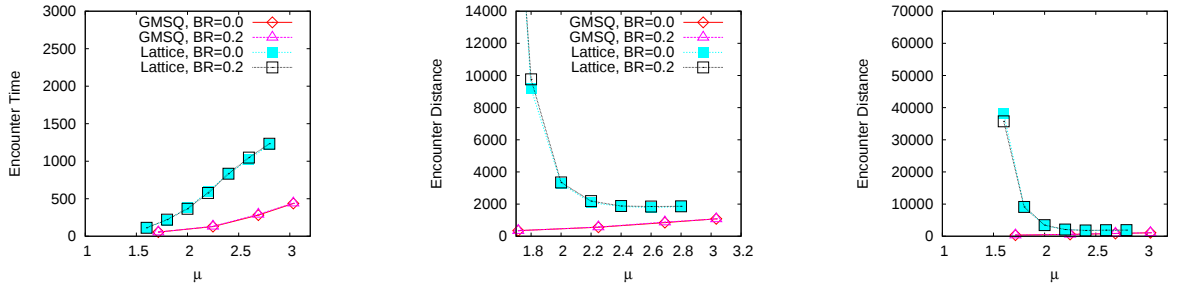


(c) パケットのパスのリンク長の和 (左), ネットワークの最短距離長との比 (右)

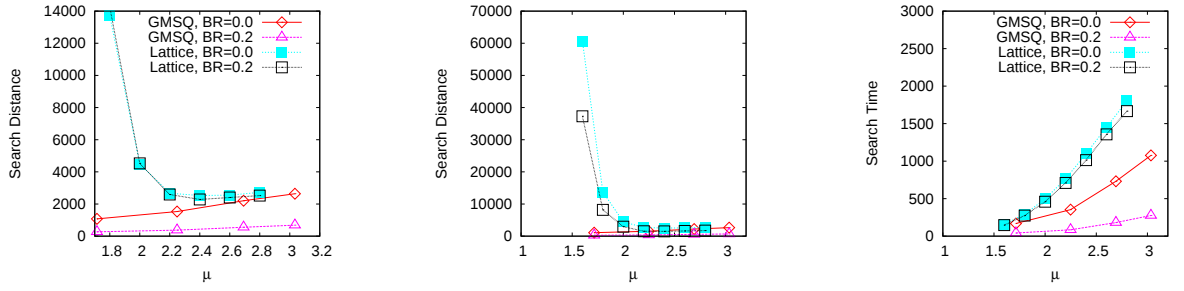


(d) 経路確立の際に使用したリンクの数とネットワークの全リンク数との比 (左), フェリーが各ステップで平均的に保持している経路要求の数 (右)

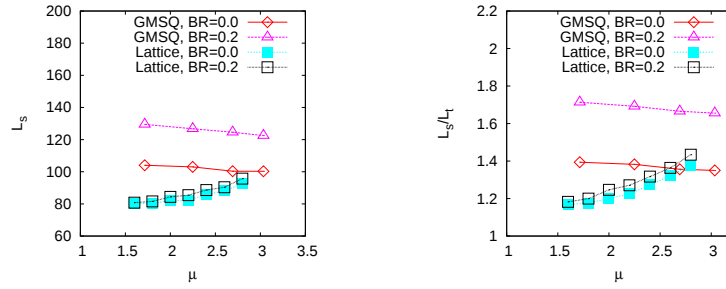
図 A.2: R0.01, Lattice の BR=0.2 は $N=500$ のときの BR=0.2 に合わせた数となっている。また, Lattice のパケットの発生位置は格子の全てのノード。



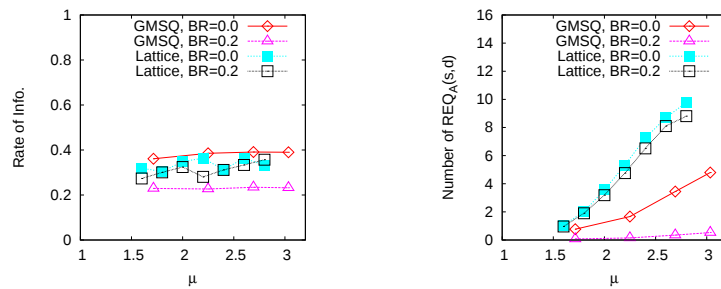
(a) 遭遇時間 (左) 距離 (中, 右)



(b) フェリー A の宛先ノード d 探索距離 (左, 中), 探索時間 (右)

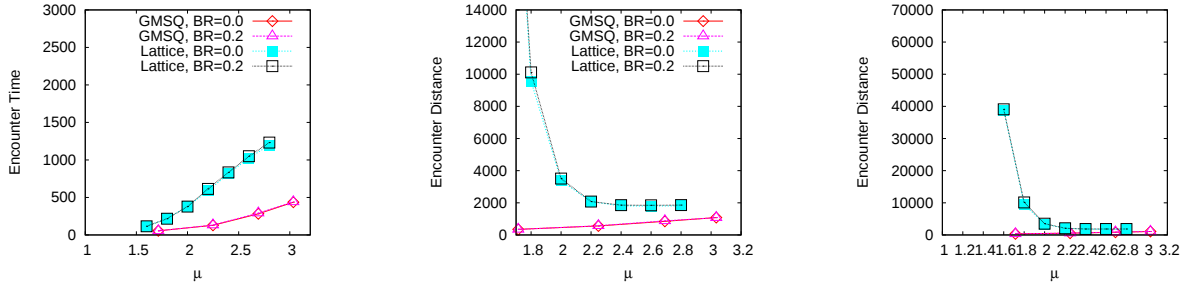


(c) パケットのパスのリンク長の和 (左), ネットワークの最短距離長との比 (右)

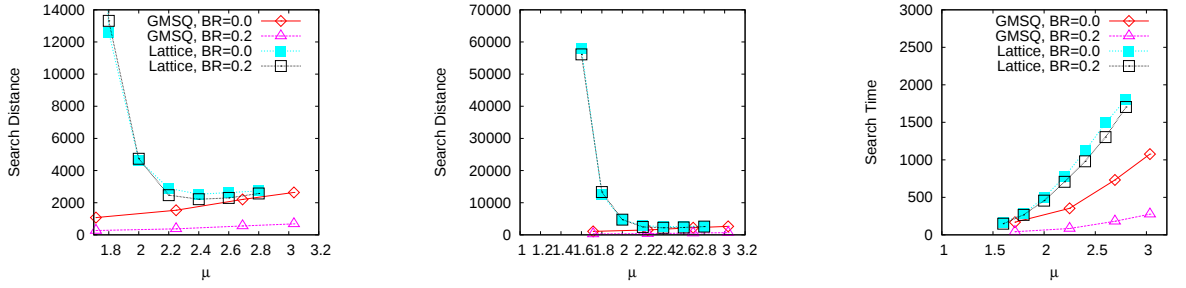


(d) 経路確立の際に使用したリンクの数とネットワークの全リンク数との比 (左), フェリーが各ステップで平均的に保持している経路要求の数 (右)

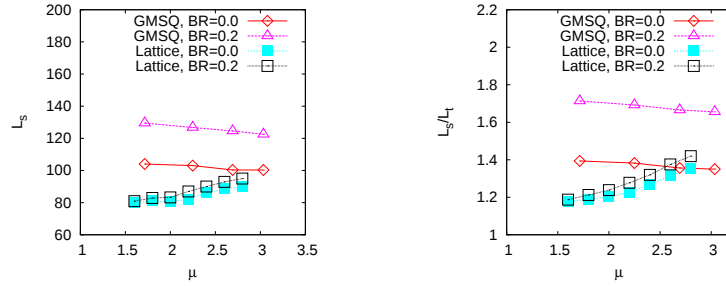
図 A.3: R0.01, Lattice の BR=0.2 は $N=3000$ のときの BR=0.2 に合わせた数となっている。また, Lattice のパケットの発生位置は格子の全てのノード。



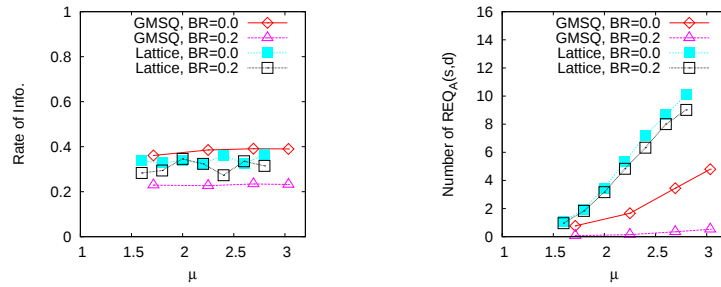
(a) 遭遇時間 (左) 距離 (中, 右)



(b) フェリー A の宛先ノード d 探索距離 (左, 中), 探索時間 (右)

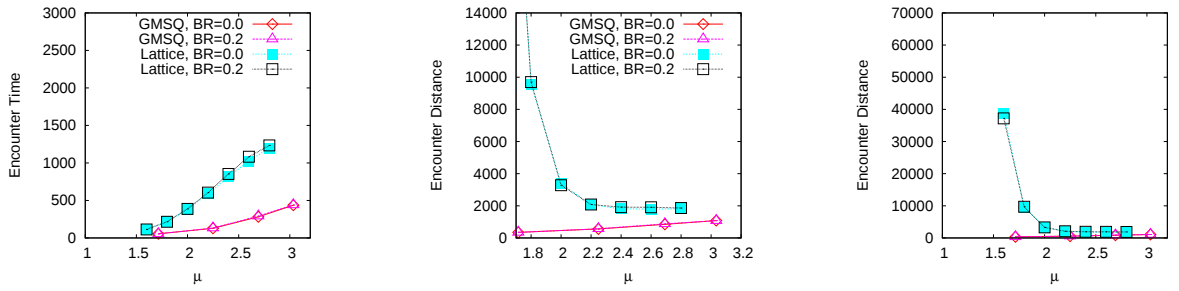


(c) パケットのパスのリンク長の和 (左), ネットワークの最短距離長との比 (右)

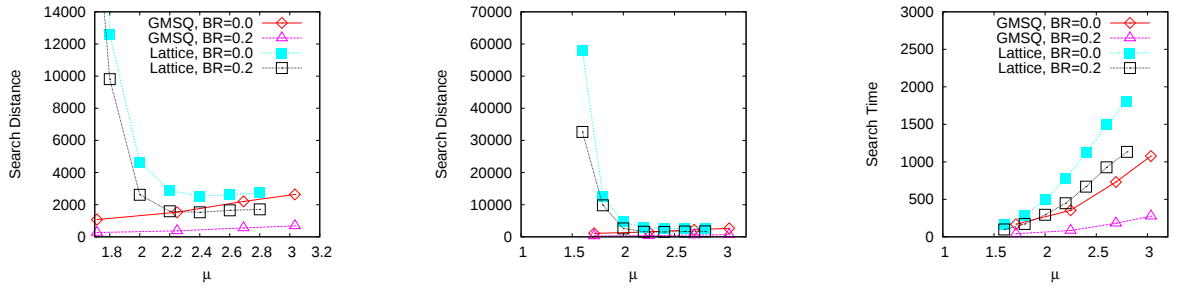


(d) 経路確立の際に使用したリンクの数とネットワークの全リンク数との比 (左), フェリーが各ステップで平均的に保持している経路要求の数 (右)

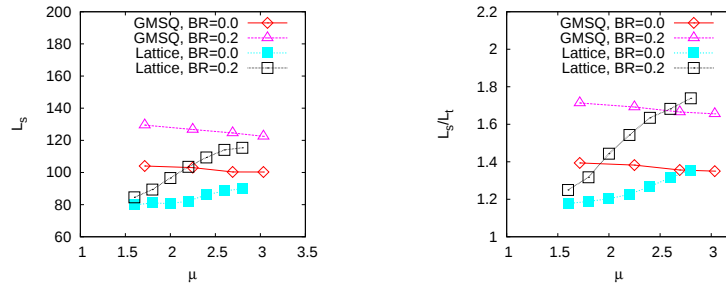
図 A.4: R0.01, Lattice の BR=0.2 は $N=500$ のときの BR=0.2 に合わせた数となっている。また, Lattice のパケットの発生位置は $N=500$ の GMSQ ネットワークと同じとしている。



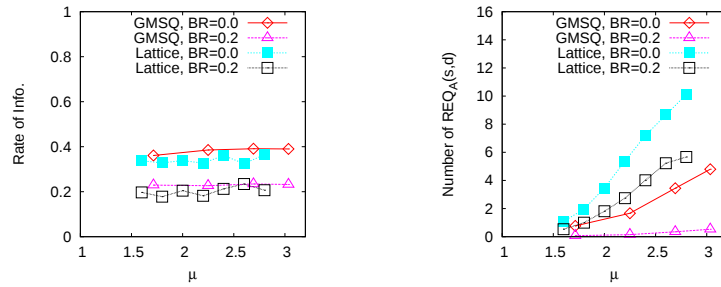
(a) 遭遇時間 (左) 距離 (中, 右)



(b) フェリー A の宛先ノード d 探索距離 (左, 中), 探索時間 (右)

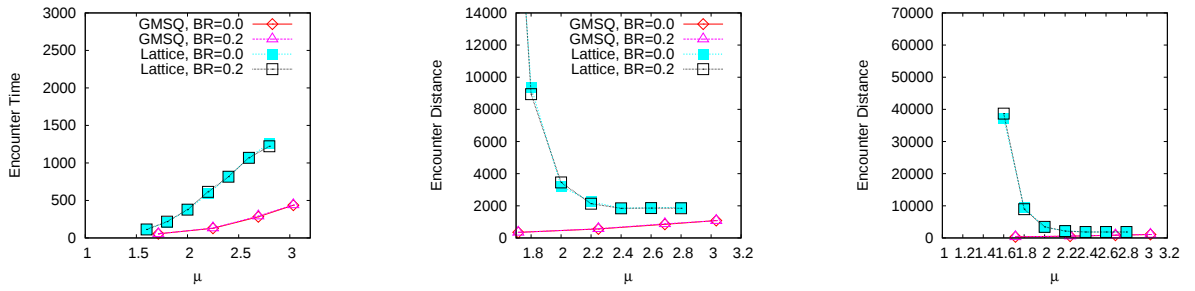


(c) パケットのパスのリンク長の和 (左), ネットワークの最短距離長との比 (右)

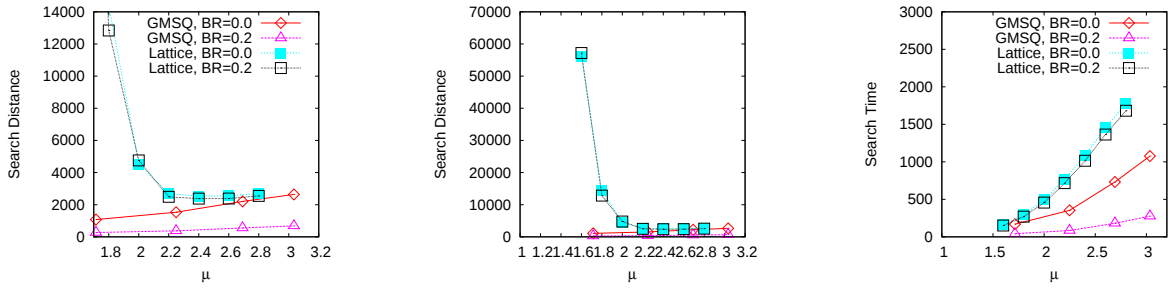


(d) 経路確立の際に使用したリンクの数とネットワークの全リンク数との比 (左), フェリーが各ステップで平均的に保持している経路要求の数 (右)

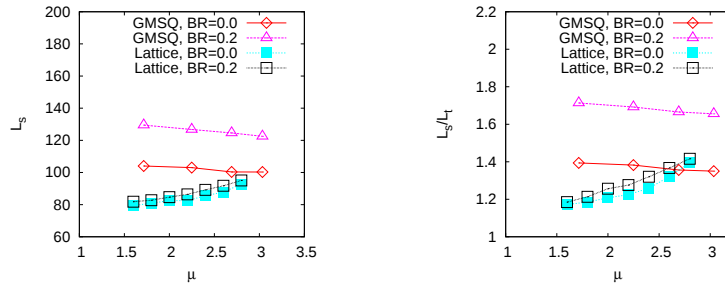
図 A.5: R0.01, Lattice の BR=0.2 は GMSQ が $N=3000$ 時の BR=0.2 に合わせた数となっている。また, Lattice のパケットの発生位置は $N=500$ の GMSQ ネットワークと同じとしている。



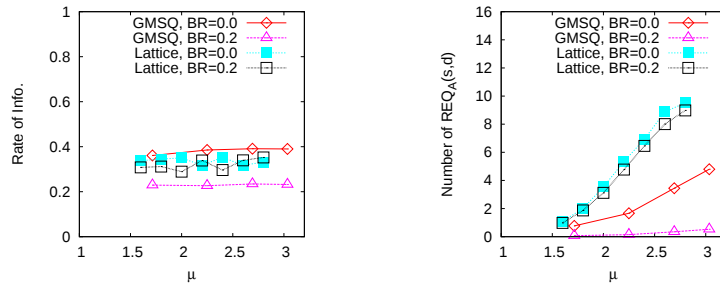
(a) 遭遇時間 (左) 距離 (中, 右)



(b) フェリー A の宛先ノード d 探索距離 (左, 中), 探索時間 (右)

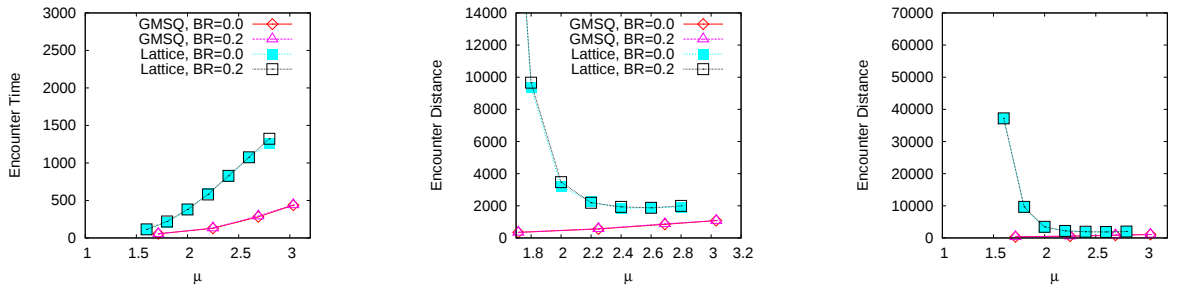


(c) パケットのパスのリンク長の和 (左), ネットワークの最短距離長との比 (右)

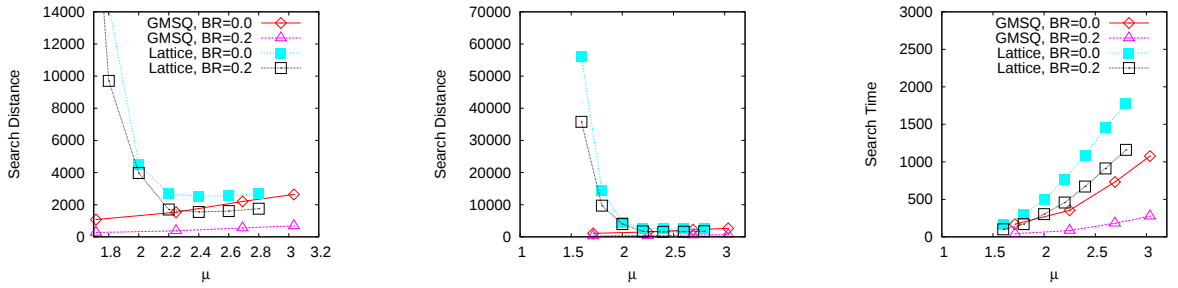


(d) 経路確立の際に使用したリンクの数とネットワークの全リンク数との比 (左), フェリーが各ステップで平均的に保持している経路要求の数 (右)

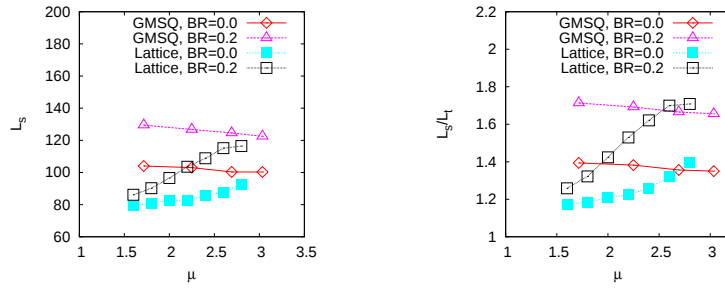
図 A.6: R0.01, Lattice の BR=0.2 は GMSQ が N=500 時の BR=0.2 に合わせた数となっている。また, Lattice のパケットの発生位置は N=3000 の GMSQ ネットワークと同じとしている。



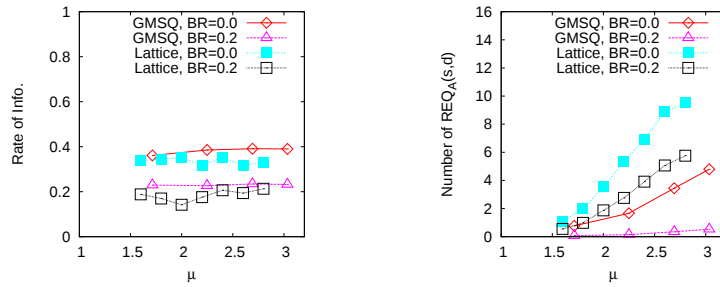
(a) 遭遇時間 (左) 距離 (中, 右)



(b) フェリー A の宛先ノード d 探索距離 (左, 中), 探索時間 (右)



(c) パケットのパスのリンク長の和 (左), ネットワークの最短距離長との比 (右)



(d) 経路確立の際に使用したリンクの数とネットワークの全リンク数との比 (左), フェリーが各ステップで平均的に保持している経路要求の数 (右)

図 A.7: R0.01, Lattice の BR=0.2 は GMSQ が $N=3000$ 時の BR=0.2 に合わせた数となっている。また, Lattice のパケットの発生位置は $N=3000$ の GMSQ ネットワークと同じとしている。