### 修士論文

正方形や直角三角形の再帰分割上のメッセージフェリー通信における遅延耐性の 評価

張 至杰

主指導教員: 林 幸雄

北陸先端科学技術大学院大学 先端科学技術研究科 (修士)

令和2年3月

#### Abstract

In this age of information society, people's life is inseparable from terrestrial communication network. The terrestrial communication network is mostly made up by electric wire, radio tower and some other infrastructure. However, in the early 21 century, we fond that the terrestrial communication network is composed by many high degree nodes and a few low degree nodes, this kind of network is named Scale-Free network. Which network would be easily got big damage when earthquake, tsunami and some other disaster occurred. Also, in the event that the terrestrial communication network 's high degree nodes are attacked, the network will stop working immediately. Because of these reasons, finding a way to construct an urgent air communication network is necessary.

As the wireless communication technology and drone control technology are processed, we found that constructing an urgent air communication network by using drones is possible. So, to create an air network which can cover a large area, the communication technology of DTN(Delay-tolerant networking) is necessary. DTN is a relay technology that enables communication even if temporarily disconnected. Especially, suitable for poor communication environments or situations where the communication infrastructure has failed during a disaster. As DTN is a storage-based communication, when communication is performed from the source node to the destination node, the transfer data is temporarily stored and propagated to the destination node using a physical vehicle such as a drone or a car.

About the structure of network, to equalize the access load for each node, a geographic MSQ (Multi Scale Quartered) network which decides the spatial arrangement of nodes according to the population self is considered. In other words, select one face at a certain probability every time, self-organized in a recursive quadrant self-similar surface. In this study, we consider the use of multiple drones equipped with various sensors and wireless communication devices to be deployed on a route located above the damaged area to recover communications, message is relayed through drones. Therefore, this study focuses on the structure (optimal shape) of the aerial network and the optimal distribution of drones.

# 目 次

第1章	はじめに	1
第2章	関連研究	2
2.1	DTN 技術	2
	2.1.1 転送方式による分類	2
	2.1.2 経路制御による分類 [2]	3
	2.1.3 感染型中継(フラッデイング)モデル	4
	2.1.4 メッセージフェリー方式	4
2.2	ネットワーク分類	5
	2.2.1 スケールフリー (SF)・ネットワーク	5
	2.2.2 MSQ ネットワーク	5
第3章	提案手法	6
3.1	メッセージフェリー方式の分類	6
3.2	MSQ ネットワークの基本構成	
3.3	面ルーティング	
3.4	評価方式	9
第4章	実験手順	11
4.1	地域人口データの可視化	11
	4.1.1 データの変換	
		13
4.2		16
	•	17
		18
		32
	4.2.4 各ノード(基地局)へのアクセス人口数(アクセス負荷) . :	33
4.3	最短経路の抽出	36
第5章	最適 $\mu_f$ 値による実験結果 $_{_{_{_{_{_{_{_{_{_{_{_{_{_{_{_{_{_{_{$	38
5.1		38
	$5.1.1$ 最適 $\mu_f$ 値の分布 $\dots$	
		41

第6章	おわりに 6	0
5.3	<sup>1</sup> / <sub>2</sub> 均等配分とランダム配分比較 5	7
	$5.2.4$ 総遅延、最適 $\mu_f$ 総値と面の関係 $\dots$ 5	
	5.2.3 End-to-End(Path) 遅延の分布	4
	5.2.2 面 cost の分布	1
	$5.2.1$ 最適 $\mu_f$ 値の分布	18
5.2	ランダム配分の結果4	18
	$5.1.4$ 総遅延、最適 $\mu_f$ 総値と面の関係	16
	5.1.3 End-to-End(Path) 遅延の分布	4

# 図目次

2.1	同期型転送見取り図	2
2.2	非同期型転送見取り図	3
2.3	メッセージフェリー概念図	4
2.4	基本的な面分割の見取り図	5
3.1	経路と中継による分類	6
3.2	基本構造図	7
3.3	平面グラフにおける面ルーティング	8
3.4	情報転送の見取り図・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	9
0.4	旧刊4400元以り回 · · · · · · · · · · · · · · · · · · ·	Э
4.1	地域分割図	12
4.2	福岡エリアの人口分布図	13
4.3	広島エリアの人口分布図	13
4.4	京阪エリアの人口分布図	13
4.5	名古屋エリアの人口分布図	13
4.6	札幌エリアの人口分布図	14
4.7	仙台エリアの人口分布図	14
4.8	東京エリアの人口分布図	14
4.9	7都市人口分布	15
4.10	7都市人口累積分布	15
4.11	初期構成見取り図....................................	16
4.12	最大分割次数見取り図	16
4.13	見取り図	17
4.14	福岡エリア 100 ノード(左:正方形、右:直角三角形)	18
4.15	福岡エリア 200 ノード(左:正方形、右:直角三角形)	18
4.16	福岡エリア 300 ノード(左:正方形、右:直角三角形)	19
4.17	福岡エリア 400 ノード(左:正方形、右:直角三角形)	19
4.18	広島エリア 100 ノード(左:正方形、右:直角三角形)	20
4.19	広島エリア 200 ノード(左:正方形、右:直角三角形)	20
4.20	広島エリア 300 ノード(左:正方形、右:直角三角形)	21
4.21	広島 400 エリアノード(左:正方形、右:直角三角形)	21
4.22	京阪エリア 100 ノード(左:正方形、右:直角三角形)	22

4.23	京阪エリア 200 ノード(左:正方形、右:直角三角形)	22
4.24	京阪エリア 300 ノード(左:正方形、右:直角三角形)	23
4.25	京阪エリア 400 ノード(左:正方形、右:直角三角形)	23
4.26	名古屋エリア 100 ノード(左:正方形、右:直角三角形)	24
4.27	名古屋エリア 200 ノード(左:正方形、右:直角三角形)	24
4.28	名古屋エリア 300 ノード(左:正方形、右:直角三角形)	25
4.29	名古屋エリア 400 ノード(左:正方形、右:直角三角形)	25
4.30	札幌エリア 100 ノード(左:正方形、右:直角三角形)	26
4.31	札幌エリア 200 ノード(左:正方形、右:直角三角形)	26
4.32	札幌エリア 300 ノード(左:正方形、右:直角三角形)	27
4.33	札幌エリア 400 ノード(左:正方形、右:直角三角形)	27
4.34	仙台エリア 100 ノード(左:正方形、右:直角三角形)	28
4.35	仙台エリア 200 ノード(左:正方形、右:直角三角形)	28
4.36	仙台エリア 300 ノード(左:正方形、右:直角三角形)	29
4.37	仙台エリア 400 ノード(左:正方形、右:直角三角形)	29
4.38	東京エリア 100 ノード(左:正方形、右:直角三角形)	30
4.39	東京エリア 200 ノード(左:正方形、右:直角三角形)	30
4.40	東京エリア 300 ノード(左:正方形、右:直角三角形)	31
4.41	東京エリア 400 ノード(左:正方形、右:直角三角形)	31
4.42	ノード数と面の個数の関係図	32
4.43	福岡エリア負荷図	33
	福岡エリア負荷累積分布図	33
4.45	広島エリア負荷図	33
4.46	広島エリア負荷累積分布図	33
		34
4.48	京阪エリア負荷累積分布図	34
4.49	名古屋エリア負荷図	34
4.50	名古屋エリア負荷累積分布図	34
	札幌エリア負荷図	34
4.52	札幌エリア負荷累積分布図	34
4.53	仙台エリア負荷図	35
4.54	仙台エリア負荷累積分布図	35
	東京エリア負荷図	35
4.56	東京エリア負荷累積分布図	35
4.57	最短経路見取り図	36
4.58	配分方式の見取り図	37
5.1	福岡エリア uf 分布図	38
	福岡エリア uf 累積分布図	
0.2	1田四十 ノノ ய 糸恨刀41凶	OQ

5.3	広島エリア uf 分布図	39
5.4	広島エリア uf 累積分布図	39
5.5	京阪エリア uf 分布図	
5.6	京阪エリア uf 累積分布図	39
5.7	名古屋エリア uf 分布図	39
5.8	名古屋エリア uf 累積分布図	
5.9	札幌エリア uf 分布図	
5.10	札幌エリア uf 累積分布図	40
5.11	仙台エリア uf 分布図	40
	仙台エリア uf 累積分布図	
5.13	東京エリア uf 分布図	40
5.14	東京エリア uf 累積分布図	40
	福岡エリア面 cost 分布図	
	福岡エリア面 cost 累積分布図	
	広島エリア面 cost 分布図	
	広島エリア面 cost 累積分布図	
5.19	京阪エリア面 cost 分布図	42
	京阪エリア面 cos 累積 t 分布図	
5.21	名古屋エリア面 cost 分布図	42
5.22	名古屋エリア面 cost 累積分布図	42
5.23	札幌エリア面 cost 分布図	42
5.24	札幌エリア面 cost 累積分布図	42
5.25	仙台エリア面 cost 分布図	43
5.26	仙台エリア面 cost 累積分布図	43
5.27	東京エリア面 cost 分布図	43
5.28	東京エリア面 cost 累積分布図	43
5.29	path 遅延分布図 (100node)	44
5.30	path 遅延累積分布図 (100node)	44
5.31	path 遅延分布図 (200node)	44
5.32	path 遅延累積分布図 (200node)	44
5.33	path 遅延分布図 (300node)	45
5.34	path 遅延累積分布図 (300node)	45
5.35	path 遅延分布図 (400node)	45
5.36	path 遅延累積分布図 (400node)	45
5.37	総遅延と面の関係図	46
5.38	最適 uf 総値と面の関係図	46
5.39	拡大図	46
5.40	uf 総値と総遅延の関係図	47
5.41	拡大図	47

5.42	福岡エリア uf 分布図	18
5.43	福岡エリア uf 累積分布図	18
5.44	広島エリア uf 分布図	18
		18
5.46		18
5.47	京阪エリア uf 累積分布図	18
5.48		19
5.49	名古屋エリア uf 累積分布図 4	19
5.50	札幌エリア uf 分布図	19
		19
		19
		19
		50
		50
		51
5.57	福岡エリア面 cost 累積分布図 5	51
5.58	広島エリア面 cost 分布図	51
5.59	広島エリア面 cost 累積分布図 5	51
5.60	京阪エリア面 cost 分布図	51
		51
5.62	名古屋エリア面 cost 分布図	52
5.63	名古屋エリア面 cost 累積分布図	52
5.64	札幌エリア面 cost 分布図	52
5.65		52
5.66	仙台エリア面 cost 分布図	52
		52
5.68	東京エリア面 cost 分布図	53
5.69	東京エリア面 cost 累積分布図 5	53
5.70	path 遅延分布図 (100node)	54
5.71	path 遅延累積分布図 (100node)	54
5.72	path 遅延分布図 (200node)	54
5.73	path 遅延累積分布図 (200node)	54
5.74	path 遅延分布図 (300node)	54
5.75	path 遅延累積分布図 (300node)	54
5.76	path 遅延分布図 (400node)	55
5.77	path 遅延累積分布図 (400node)	55
5.78	総遅延と面の関係図 5	56
5.79	最適 uf 総値と面の関係図	56
5.80	細部図	56

5.81	uf 総値と総遅延の関係図	56
		56
5.83	福岡エリア総コスト比較図	57
5.84	福岡エリア総遅延比較図	57
5.85	広島エリア総コスト比較図	57
5.86	広島エリア総遅延比較図	57
5.87	京阪エリア総コスト比較図	57
5.88	京阪エリア総遅延比較図	57
5.89	名古屋エリア総コスト比較図	58
5.90	名古屋エリア総遅延比較図	58
5.91	札幌エリア総コスト比較図	58
5.92	札幌エリア総遅延比較図	58
5.93	仙台エリア総コスト比較図	58
5.94	仙台エリア総遅延比較図	58
5.95	東京エリア総コスト比較図	59
5.96	東京エリア総遅延比較図	59

# 表目次

4.1	地域メッシュデータ構造													11
4.2	変換後データ構造													12

## 第1章 はじめに

情報社会の現代、人々は巨大な通信ネットワークに支えられて生きている。この通信ネットワークは、ケーブルや基地局、電波塔などのインフラで構築されている。ところが、21世紀初頭に、通信ネットワークは大多数の低次数ノードと極少数の高次数ノードからなるスケールフーリ構造であることが見つかり [1]、地震や津波などの災害が発生した際に大きなダメージを受けた事例が多数報告されている。また、テロリストが高次数ノードのインフラを狙って数箇所だけでも攻撃すると、通信ネットワークはすぐに機能停止になり得ることも理論的に示された。これらの原因で、通信ネットワークの機能や性能低下または停止することが想定され、緊急通信ネットワークの構築が重要な課題となっている [2]。

一方、無線通信技術と無人機制御技術の進歩とともに、ネットワークの地上インフラが機能停止する場合でも、無人機を用いて空中緊急通信ネットワークを構築することが可能と考えられる。中でも、DTN(Delay Tolerant Networking):遅延耐性ネットワーク技術が有力視されている。DTN は蓄積搬送型通信として、送信元ノードから受信ノードまでの通信を行う際に、転送データを一時的に蓄積して、無人機や車などの物理移動手段を使って、中継ノードを経由して宛先ノードまで伝搬する技術である[2]。

本研究では、各種のセンサーや無線通信装置などを搭載する無人機を複数台使って被害地域上空に決まったルートに配置して通信をリカバリーすることを考える、情報は無人機を通じてリレーされる。そこで、本研究は空中ネットワークの構造(最適形状)と無人機の最適配分に着目して検討する。

# 第2章 関連研究

## 2.1 DTN 技術

DTN(Delay Tolerant Networking):遅延耐性ネットワークは一時的に断線しても通信を可能とする中継技術である。特に、劣悪な通信環境や災害時に通信インフラが機能停止になった状況に適している[4]。

### 2.1.1 転送方式による分類

1. 同期型転送方式: 同期型は二つの無人機の間で直接データをやり取りする (図 2.1 を参照)。すなわち、無人機同士が接触した時のみ、転送できろ。

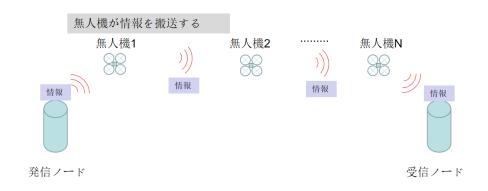


図 2.1: 同期型転送見取り図

2. 非同期型転送方式: 非同期型は二つの無人機の間に中継ノードを利用して間接的にでデータを転送する (図 2.2 を参照)。すなわち、無人機同士が接触しなくても転送可能である。

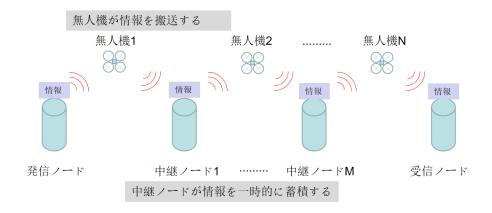


図 2.2: 非同期型転送見取り図

### 2.1.2 経路制御による分類[2]

- 1. 決まった経路で転送:
  - End-to-End 経路は事前に決まっている。
  - ノード間の接触が計画的な場合に使用する。
- 2. ランダムな経路で転送:
  - End-to-End 経路は事前に決めせず、Hop-by-Hopで次の中継ノードを決定しながら中継転送を行うこと。
  - ノード間の接触が偶発的な場合に使用する。
- 3. ハイブリッド型中継転送:
  - この転送は決まった経路とランダムな経路で転送の組合したものである。
  - 計画的な接触と偶発的な接触が混在する場合に使用する。

## 2.1.3 感染型中継(フラッデイング)モデル

感染型中継モデルは非同期型・ランダム経路の転送として、情報を持つノードが他のノードと交信できる時、常に、その情報をコピーして他のノードに転送する。この結果、伝染病のように、コピーした情報がネットワーク内で広がる。このモデルは優秀な遅延性能を持つが、多量のコピーによるコストが高い[2]。

#### 2.1.4 メッセージフェリー方式

メッセージフェリーとは、位置が遠く離れている端末が直接やり取りできない場合、通信を行う時、決まった経路上のノード(フェリー)に情報転送する方法である[2]。(図 2.3 を参照)

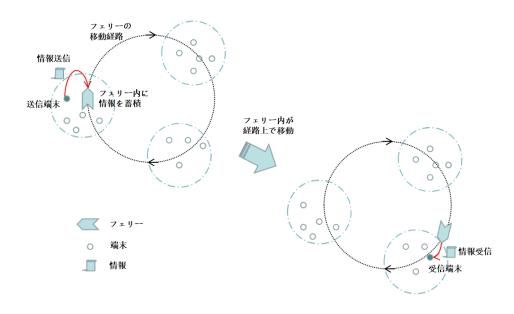


図 2.3: メッセージフェリー概念図

## 2.2 ネットワーク分類

## 2.2.1 $\lambda f - \mu J = (SF) \cdot \lambda + J - \Delta$

現実世界の多くのネットワーク(インターネット、航空網、電力網など)は、大多数の低次数ノードと極少数の高次数ノードからなるスケールフーリ構造でなす。その特徴はノードの次数がべき乗則に従う[1]。長所は、ノードのランダム攻撃に対して頑丈性が高い。短所は、ほんの数個のハブが意図的に攻撃されると、ネットワーク全体が崩れる[5]。

#### 2.2.2 MSQ ネットワーク

各ノードへのアクセス負荷を均一化するノードの空間配置を自己組織的に人口に応じて決める地理的な MSQ ネットワークが提案される。MSQ(Multi—Scale Quartered)ネットワークとは、毎時刻ある確率で面を 1 つ選び、自己相似な面の再帰的四分割で自己組織化される [6]。(図 2.4 を参照) その長所としては:

- 低次数ノードのみで構成されてハブが無く(正方形と三角形の場合、最大次数は4か6)、攻撃に対して頑丈性が強い。
- 二つのノード間の最短距離は最悪でも二ノード間の直線距離の高々二倍であ り、その平均値は 1.1 倍。
- 面グラフなので、 二ノード間の直線と交わる面の局所情報だけで最短距離 を求めることが出来る [3]。

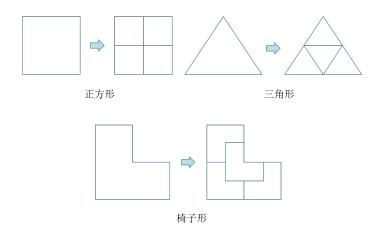


図 2.4: 基本的な面分割の見取り図

## 第3章 提案手法

## 3.1 メッセージフェリー方式の分類

大規模災害が発生して、地上インフラが機能停止する場合、各種のセンサーや無線通信装置などを搭載する無人機を多数使って被害地域上空に配置することで通信をリカバリーすることを考える。また、広範囲の地域の通信を回復するには、その地域の総人口数と人口分布を考慮すべきである。この場合、感染型中継モデルはコストが高く適さない。そこで、システムの総コストを抑えるために、非同期型中継方式を選択する。更に、地域人口分布に従ったノード配置によって、ノードの負荷をバランス良くするため、ノードの空間配置を自己組織的に人口に応じて決める地理的な MSQ ネットワーク構造を選択する [3]。 MSQ ネットワーク上で、無人機の移動経路は決まった経路とした転送方式を選択する。すなわち、自律分散システムとしての制御に適した、決まった経路で非同期型中継の方式に着目する [6]。(図 3.1 を参照)

		非同期型
ランダムの経路	×	*
決まったの経路	×	0

図 3.1: 経路と中継による分類

## 3.2 MSQ ネットワークの基本構成

通信ネットワークの物理的な要素としては、中継基地局(ネットワークのノード、転送データを一時的に蓄積)と運び役の無人機(メッセージフェリー、転送データを搬送する)で構成される。地域内の人々(端末)は一番近い中継ノードにアクセスすると考え、情報は無人機を通じてリレーされる。MSQネットワークの各面を巡回する複数台の無人機を二重設置(時計回りと反時計回り)して、中継ノード間の送受信要求に応じて各サイクルのメッセージフェリーの配分数や巡回速度を調整するものとする。(図 3.2 を参照)

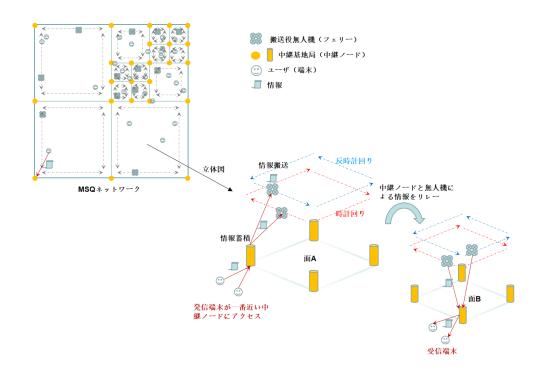


図 3.2: 基本構造図

## 3.3 面ルーティング

MSQネットワークは二次元平面グラフので、リンクの交差がない。そこで、面ルーティングとして、発信ノードsから受信ノードsまで情報を転送する時、sと t を結ぶ直線が交わる面集の辺の中で、隣接ノードに関する局所情報を用いて、st 間の最短経路を見つかることができます。図 3.3 を参照する、st と面(A,B,C,D,E) 交わる、白い丸の経路は最短経路である。

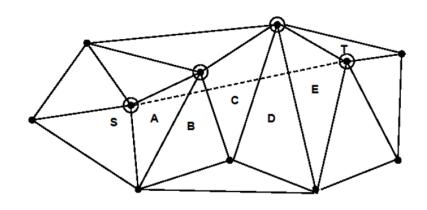


図 3.3: 平面グラフにおける面ルーティング

## 3.4 評価方式

一般的な設定として、MSQ ネットワーク上の任意二つのノード間の送受信要求が偶発的に一様ランダムに発生すると想定して、情報はそれらノード間の最短経路を経由して搬送することを考えられる。そこで、直列型待ち行列モデルからEnd-to-End 遅延を評価する。[2]

1/M/M 待ち行列モデルは1 列に並んだ客や要求を1 つの窓口やサーバが確率的に処理する待ち行列として、送信に対応する到着率 $\lambda$  はポアソン分布に従い、受信に対応するサービス(負荷)率 $\mu$  は指数分布に従う。遅延を無限大化させないため、 $\mu > \lambda$  が必要である。以下公式(3.1)から公式(3.2)を示す。

平均利用率
$$\rho = \frac{\lambda}{\mu}$$
 (3.1)

平均客数 
$$L = \frac{\rho}{1-\rho}$$
 (3.2)

$$End-to-End$$
 遅延  $T=\frac{L}{\lambda}=\frac{1}{\mu-\lambda}$  (3.3)

3.2章の基本構成によって、無人機がMSQネットワークの各面内で巡回しながら情報搬送する。すなわち、MSQネットワークシステムはそれぞれの面で通信サービスを提供する。つまり、任意の二つノード間の通信はノード間の最短経路を経由する面で提供すること。そこで、各面のサービス(負荷)率は $\mu_f$ (f は面の id)と定義する。但し、無人機の配分数や巡回スピードをどうするかについては深入れせず、 $\mu_f$  値に着目して検討する。

また、重力モデルに従って、任意二つのノード(s、t)間の到着率は各ノード s や t へのアクセス人口数  $Pop_s$  と  $Pop_t$  の積で定義する。すなわち、任意二つノード(s、t)間の到着率  $\lambda_{st} = Pop_s * Pop_t$  として、一つの面に対して各辺の到着率  $\lambda_{f(e)}$  (f は面の id、e は辺の id) は、すべての最短経路がこの辺を経由する End-to-End の 到着率  $\lambda_{st}$  を足し算する結果、 $w_{f(e)}$  がこの辺を経由する回数となる。(図 3.4 を参照)

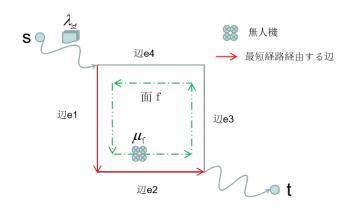


図 3.4: 情報転送の見取り図

よって、End-to-End 遅延  $T_{st}$  は st の最短経路が経由する辺の遅延を足し算する 結果、公式 (3.4) で定義する。通信総コストC は全ての End-to-End 遅延  $T_{st}$  の和 と各面サービス(負荷)  ${\bf x} \mu_f$  の和を足し算するものとして、公式 (3.5) で定義す る[2]、E は各面辺の総数で正方形分割の場合は E=4、三角形分割の場合は E=3 と なる。公式 (3.5) から各面ごとの遅延  $T_f$  と各面のコスト  $C_f$  を得る。

$$T_{st} = \sum_{e \in Path(s,t)} \frac{1}{\mu_f - \lambda_{f(e)}}$$
(3.4)

$$C = \sum_{s,t} \sum_{e \in Path(s,t)} \frac{1}{\mu_f - \lambda_{f(e)}} + \sum_f \mu_f = \sum_f \left( \sum_{e=1}^E \frac{w_{f(e)}}{\mu_f - \lambda_{f(e)}} + \mu_f \right)$$
(3.5)

$$C \equiv \sum_{f} (T_f + \mu_f) = \sum_{f} C_f \tag{3.6}$$

$$T_f = \sum_{e=1}^{E} \frac{w_{f(e)}}{\mu_f - \lambda_{f(e)}}$$
 (3.7)

$$T_f = \sum_{e=1}^{E} \frac{w_{f(e)}}{\mu_f - \lambda_{f(e)}}$$

$$C_f = T_f + \mu_f = \sum_{e=1}^{E} \frac{w_{f(e)}}{\mu_f - \lambda_{f(e)}} + \mu_f$$
(3.8)

通信総コストCを最小化するために、ニュートン法を用いて各面の最小コスト を求める、以下の式 (3.9) から  $\mu_f$  の極値を求める。但し、サービス(負荷)率  $\mu_f$ は各辺の到着率  $\lambda_{f(e)}$  より大きい必要があるので、すなわち、 $\mu_f > max\{\lambda_{f(e)}\}$  で ある。そこで、 $\mu_f$  の初期値は  $\max\{\lambda_{f(e)}\}+1$  に設定する。

$$F(\mu_f) = \frac{\partial C_f}{\partial \mu_f} = \frac{\partial}{\partial \mu_f} \left( \sum_{e=1}^E \frac{w_{f(e)}}{\mu_f - \lambda_{f(e)}} + \mu_f \right) = 1 - \sum_{e=1}^E \frac{w_{f(e)}}{(\mu_f - \lambda_{f(e)})^2} = 0 \quad (3.9)$$

ニュートン法のアルゴリズム:

 $Step1: \mu_f$  の初期値を求める。

 $\mathbf{Step2}$  :  $\mu_f' = \mu_f - rac{F(\mu_f)}{F'(\mu_f)}$ を求める。

**Step3** :  $|\mu'_f - \mu_f| < 0.0000001$  の時、 $\mu_f$  は極値とみなす、 $|\mu'_f - \mu_f| \ge 0.0000001$ の時、 $\mu_f'$  の値を  $\mu_f$  に更新して Step2 に戻る。

## 第4章 実験手順

本研究ではPython言語でプログラム作成を行い、数値シミュレーションを通じて遅延やコストについて分析した。Pythonのコードはシンプルで読みやすく、他の言語と比べると、少ない行数で書ける。また、豊富なライブラリが公開されている。ネットワーク分析やデータ処理のライブラリもある。

## 4.1 地域人口データの可視化

#### 4.1.1 データの変換

(財)統計情報研究センターから提供されている平成22年国勢調査の地域メッシュ統計データは、経度・緯度の座標に従った80km×80kmの大きさの各地域(福岡、京阪、名古屋、広島、札幌、仙台、東京)の人口データである。地域メッシュのデータ構造を簡単に説明すると、80km×80km大きさの正方形地域は第一次地域区画、第一次地域区画を横縦にそれぞれに8等分して得た正方形(10km×10km)は第二次地域区画、第二次地域区画を横縦にそれぞれに10等分して得た正方形(1km×1km)は第三次地域区画、最後に第三地域区画の正方形を4等分して得た正方形はメッシュの最小単位(ブロックと呼ぶ、後述する、面数を表す)、この0.5km×0.5km大きさの最小単位内の人口を統計された、最小単位のメッシュ数(ブロック数)は8\*8\*10\*10\*2\*2 = 25600[7] となる。

一次コード	二次コード	三次コード	四次コード	人口数
mapcode	ab	cd	e	Population
5030	00	00	1	24
5030	00	01	2	44
		•••	•••	
5030	74	96	1	2072
5030	77	99	4	453

表 4.1: 地域メッシュデータ構造

第一次地域コード (mapcode) は地域を表す。例えば、福岡:5030、広島:5132、京阪:5235 など。

第二次地域コード (ab) は第一次地域区画を横縦にそれぞれに 8 等分して、a の値は  $0\sim7$ 、b の値は  $0\sim7$ 。

第三次地域コード (cd) は第二次地域区画を横縦にそれぞれに 10 等分して、c の値は  $0\sim9$ 、d の値は  $0\sim9$ 。

第四次地域コード (e) は第三次地域区画を 4 等分して、e の値は  $1\sim4$ 。 (図 4.1 を参照)

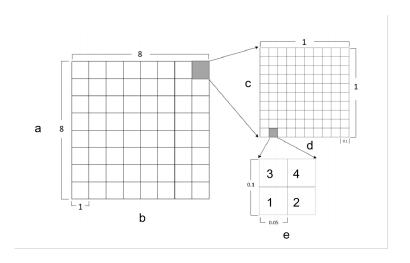


図 4.1: 地域分割図

地域コードをxy座標に変換(ブロック中心の座標を求める):

- e は 1 或いは 3 の場合: x = b \* 10 + d
- e は 2 或いは 4 の場合: x = b \* 10 + d + 0.5
- e は 1 或いは 2 の場合: y = a \* 10 + c
- e は 3 或いは 4 の場合: y = a \* 10 + c + 0.5
- 得た結果は表 4.2 で表す。

X	У	Population
0.5	0.0	24
0.5	0.5	44
77.0	77.0	2072
79.5	79.5	453

表 4.2: 変換後データ構造

## 4.1.2 人口分布の可視化

人口分布可視化は Python の Seaborn というライブラリを使用して、変換後のデータと地域地図と合わせて、Heatmap 図を作る。図 4.2 から図 4.8 は、日本の七都市(福岡、京阪、名古屋、広島、札幌、仙台、東京)の人口分布可視化図である。色が濃いところは人口密集程度が高いところを示す。

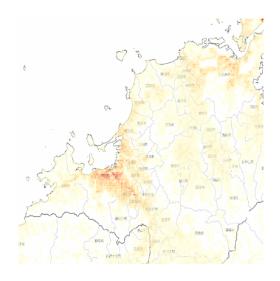


図 4.2: 福岡エリアの人口分布図



図 4.3: 広島エリアの人口分布図

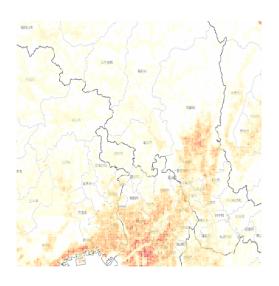


図 4.4: 京阪エリアの人口分布図

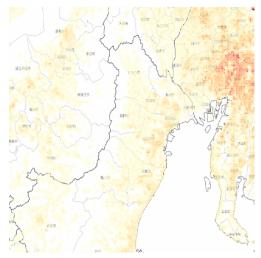


図 4.5: 名古屋エリアの人口分布図

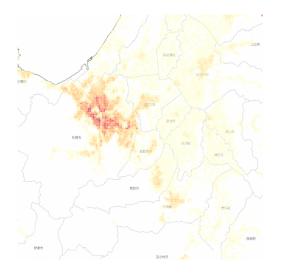


図 4.6: 札幌エリアの人口分布図

図 4.7: 仙台エリアの人口分布図

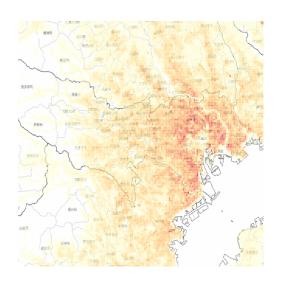
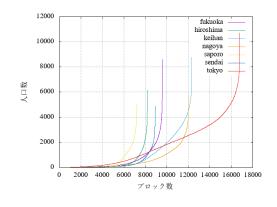


図 4.8: 東京エリアの人口分布図

図4.9 は、七都市の人口分布の折れ線グラフ、図4.10 は、人口累積分布の折れ線グラフである(横軸はブロック数、縦軸は人口数)。図4.9 と図4.10 示すのは、七都市の人口分布は指数分布に従っている、人口は地域中心に集中している。また、東京の人口密集程度が一番高いの同時に総人口数も一番多い。



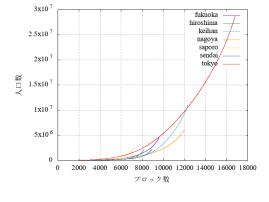


図 4.9: 7都市人口分布

図 4.10: 7都市人口累積分布

## 4.2 MSQ ネットワークの作成

正方形と(直角)三角形分割形状の違いにより、End-to-End 遅延やコストがどう異なるかを比べることで、より適した分割形状を探る。そこで、実験に用いた  $80 \text{km} \times 80 \text{km}$  の正方形分割による、MSQ ネットワークは全地域をカバーするため、同じ全地域をカバーする直角三角形分割を考えた。最大次数 4 の正方形 MSQ ネットワークと最大次数 6 の直角三角形 MSQ ネットワークを作って比較する。(図 4.11 を参照)

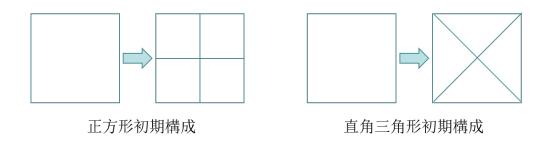


図 4.11: 初期構成見取り図

また、4.1.1章による人口データの最小単位ブロックの面積は $500\text{m} \times 500\text{m} = 0.25km^2$ であり、ブロックの位置はブロック中心の座標で決まる。よって、分割された MSQネットワークの最小面の面積は $0.25km^2$ より大きくないと、分割された MSQネットワークの面内の人口数を計算する時、誤差が出てくる。図4.12に示すのように、最大分割次数は6となり。直角三角形の場合も同様である。

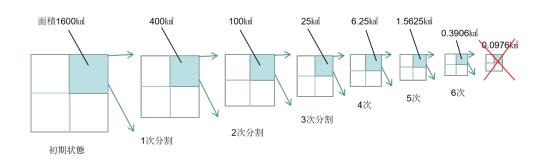


図 4.12: 最大分割次数見取り図

## 4.2.1 MSQ ネットワーク作成アルゴリズム

Step1:人口分布地図の上に、初期構成として四つの直角三角形または正方形を構成する、四つの初期面は Level=1 に付けて、面リスト F に追加する。

Step2: F中の各面の人口数を計算する(面内含まれているブロックの人口数の和)。

Step3:各面が選ばれる確率を計算する(確率=各面の人口数/総人口数)。Level≥6の面の確率は0に設定(図4.12を参照、6次の面を分割しないこと)。

**Step4**: 確率で選んだ面を 4 つの小さな面に細分する、四つの新規面の Level=選んだ面の Level+1、選んだ面を F から除去、四つの新規面を F に追加する。

Step5: 設定された総ノード数 N になるまで、Step2に戻って処理を繰り返す。

ここで、Step2の面の人口数を計算する時、ブロックの中心位置(ブロックの座標)が面の辺または頂点にいる場合、このブロックに含める面がこのブロックの人口数を等配分する。(図 4.13 を参照)

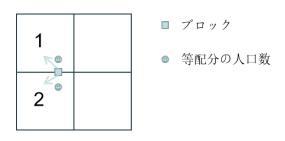


図 4.13: 見取り図

## 4.2.2 七都市の MSQ ネットワーク

七都市地域に対して、ネットワークサイズ(ノード数)は 100,200,300,400 と設定して、正方形と直角三角形の MSQ ネットワークを作成した。図から示すには、各エリアで色が濃い所(都市の中心、人口密集地)より細分されている。すなわち、ノード数が増えるほど、各ノードにアクセスする人口はより均等化されている。

福岡エリア:

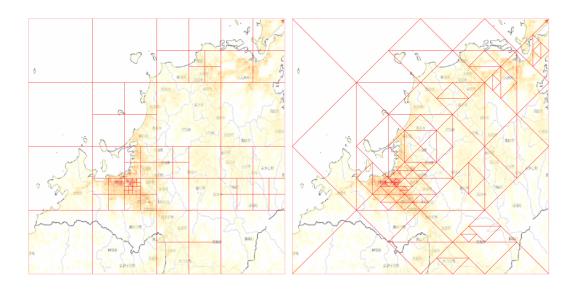


図 4.14: 福岡エリア 100 ノード (左:正方形、右:直角三角形)

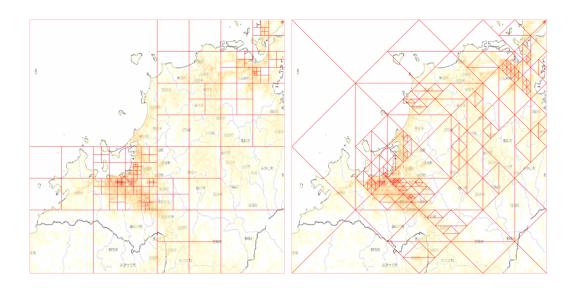


図 4.15: 福岡エリア 200 ノード (左:正方形、右:直角三角形)

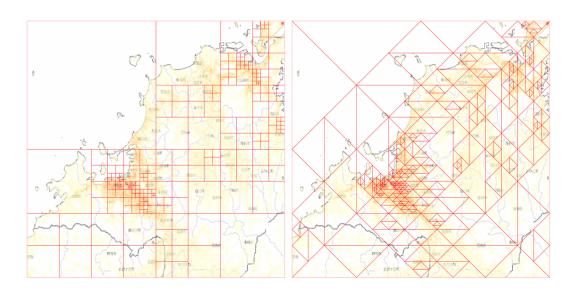


図 4.16: 福岡エリア 300 ノード (左:正方形、右:直角三角形)

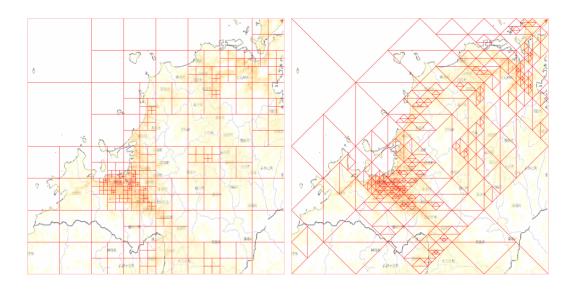


図 4.17: 福岡エリア 400 ノード (左:正方形、右:直角三角形)

## 広島エリア:

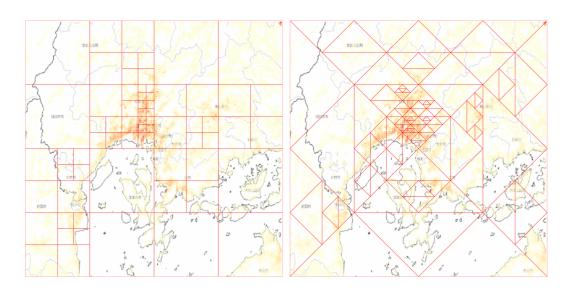


図 4.18: 広島エリア 100 ノード (左:正方形、右:直角三角形)



図 4.19: 広島エリア 200 ノード (左:正方形、右:直角三角形)

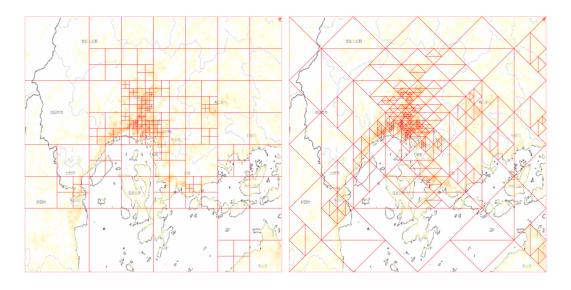


図 4.20: 広島エリア 300 ノード (左:正方形、右:直角三角形)

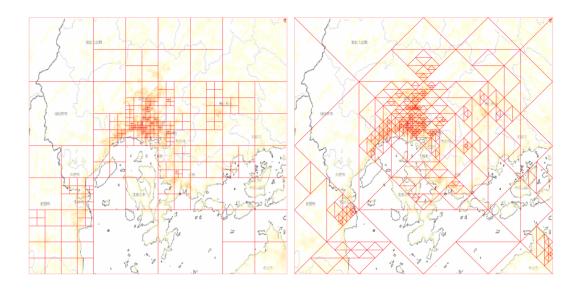


図 4.21: 広島 400 エリアノード (左:正方形、右:直角三角形)

## 京阪エリア:

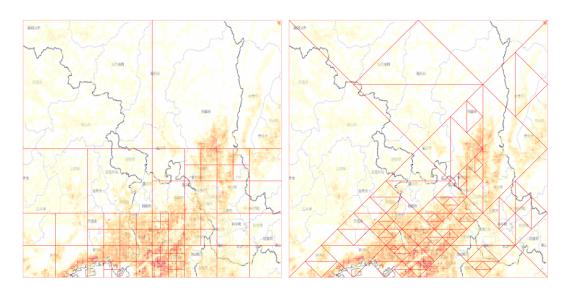


図 4.22: 京阪エリア 100 ノード (左:正方形、右:直角三角形)

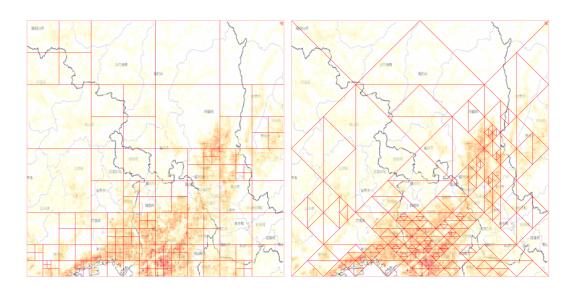


図 4.23: 京阪エリア 200 ノード (左:正方形、右:直角三角形)

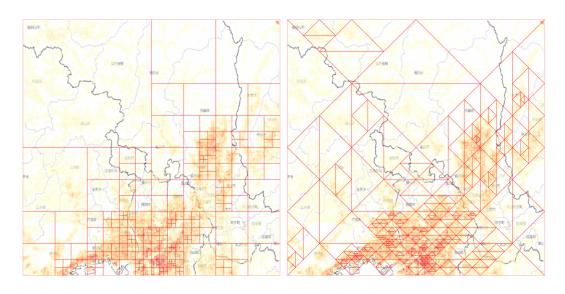


図 4.24: 京阪エリア 300 ノード (左:正方形、右:直角三角形)

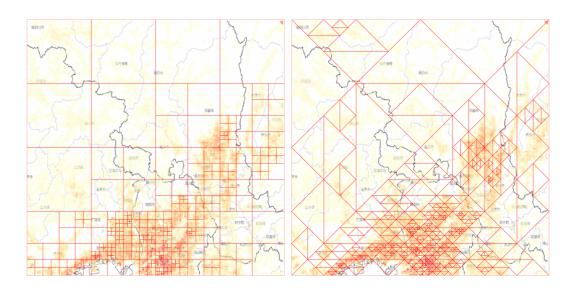


図 4.25: 京阪エリア 400 ノード (左:正方形、右:直角三角形)

## 名古屋エリア:

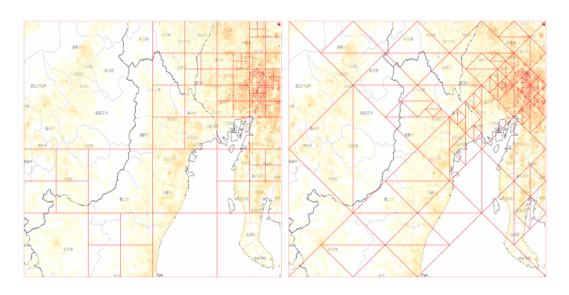


図 4.26: 名古屋エリア 100 ノード(左:正方形、右:直角三角形)

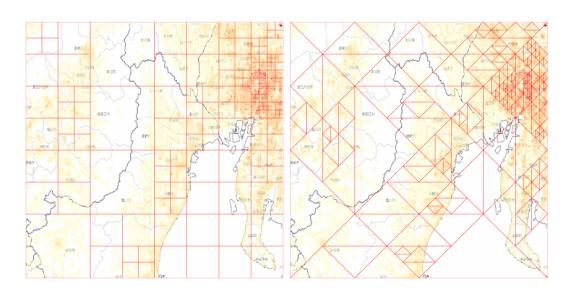


図 4.27: 名古屋エリア 200 ノード (左:正方形、右:直角三角形)

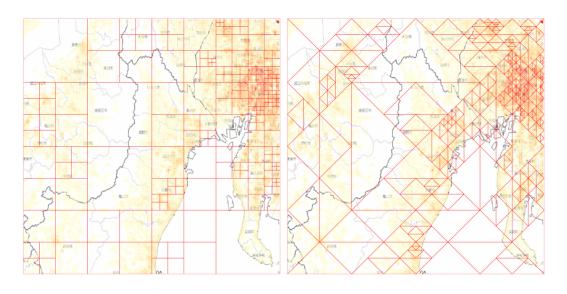


図 4.28: 名古屋エリア 300 ノード (左:正方形、右:直角三角形)

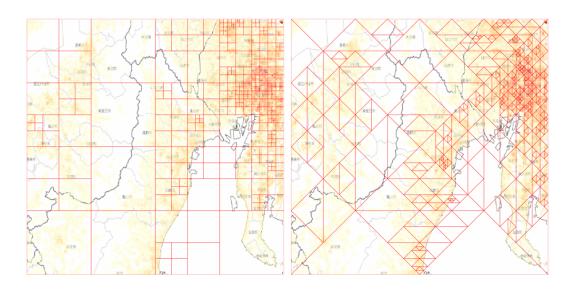


図 4.29: 名古屋エリア 400 ノード (左:正方形、右:直角三角形)

## 札幌エリア:

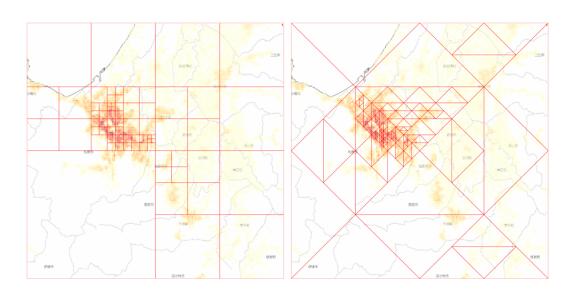


図 4.30: 札幌エリア 100 ノード (左:正方形、右:直角三角形)



図 4.31: 札幌エリア 200 ノード (左:正方形、右:直角三角形)

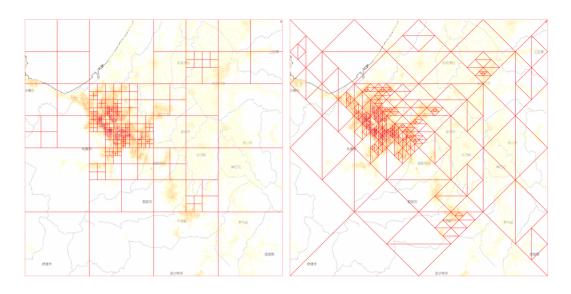


図 4.32: 札幌エリア 300 ノード (左:正方形、右:直角三角形)



図 4.33: 札幌エリア 400 ノード (左:正方形、右:直角三角形)

#### 仙台エリア:

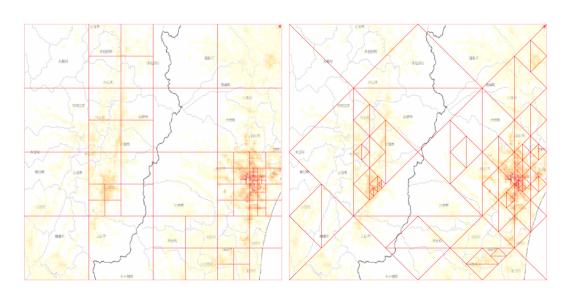


図 4.34: 仙台エリア 100 ノード (左:正方形、右:直角三角形)

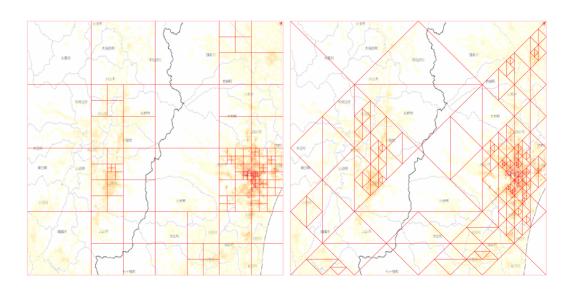


図 4.35: 仙台エリア 200 ノード (左:正方形、右:直角三角形)

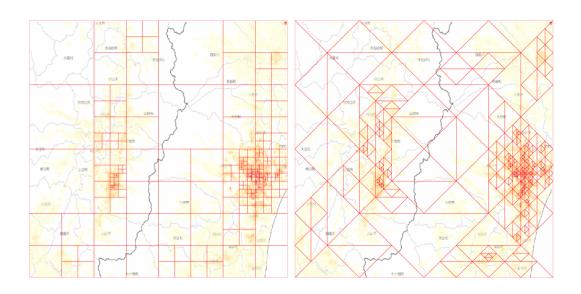


図 4.36: 仙台エリア 300 ノード (左:正方形、右:直角三角形)

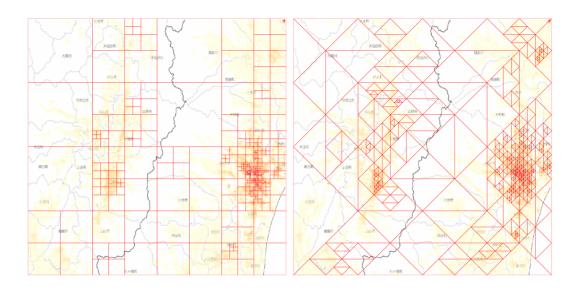


図 4.37: 仙台エリア 400 ノード (左:正方形、右:直角三角形)

#### 東京エリア:

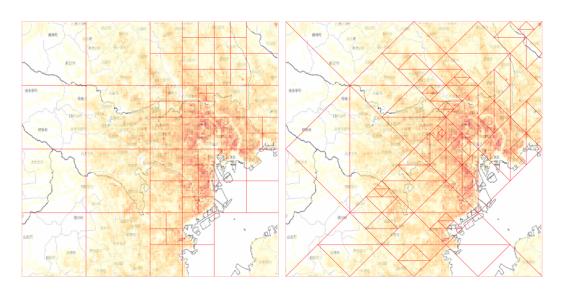


図 4.38: 東京エリア 100 ノード (左:正方形、右:直角三角形)

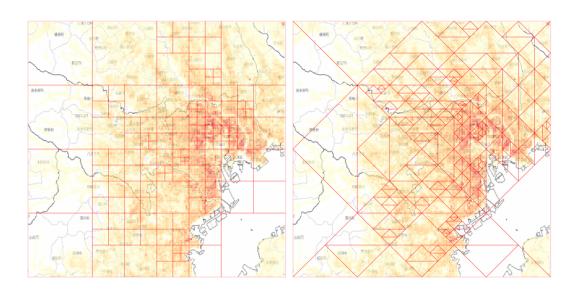


図 4.39: 東京エリア 200 ノード (左:正方形、右:直角三角形)

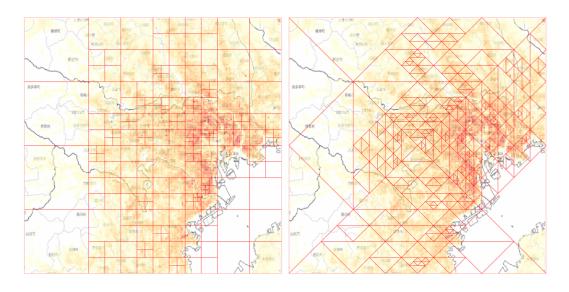


図 4.40: 東京エリア 300 ノード (左:正方形、右:直角三角形)

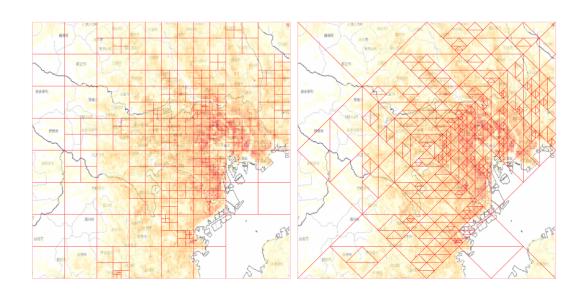


図 4.41: 東京エリア 400 ノード (左:正方形、右:直角三角形)

#### 4.2.3 MSQ ネットワーク性質

図 4.42 は、各地域の正方形と直角三角形 MSQ ネットワークのノード数と面の個数の関係図である。図から、MSQ ネットワークのノード数 N と面の個数 F は一次関数の関係が、同じノード数で直角三角形の面の数は正方形より多い。また、直角三角形の傾きが正方形の傾きより大きいので、ノード数 N の増加とともに、直角三角形の面の個数 F の増加速度が正方形より速い。

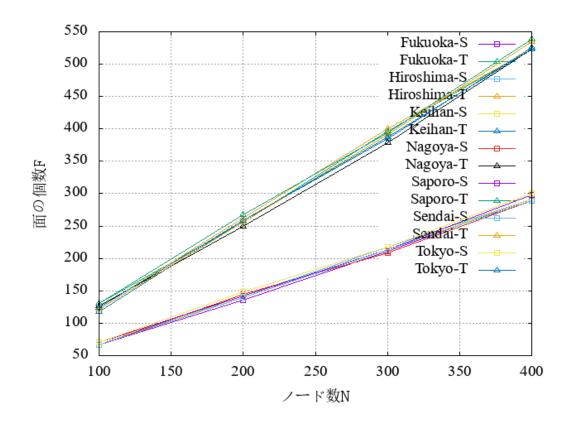
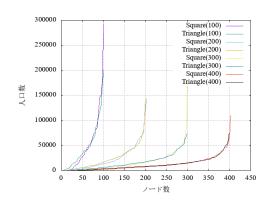


図 4.42: ノード数と面の個数の関係図

#### 4.2.4 各ノード(基地局)へのアクセス人口数(アクセス負荷)

MSQ ネットワークを作成する際に、MSQ ネットワーク各ノードにアクセス人口数も計算出来る。計算方法は、各ブロックの人口数を一番近いノードに追加する。また、一番近いノードが多数存在するの場合、このブロックの人口数をそれぞれのノードに均等配分する。図 4.43 から図 4.56 は、七都市ことに各ノードサイズの正方形と直角三角形 MSQ ネットワークノードにアクセス人口数の傾向図を示す。アクセス負荷とノードにアクセス人口数は正の相関ので、アクセス負荷図と呼ぶも良い。

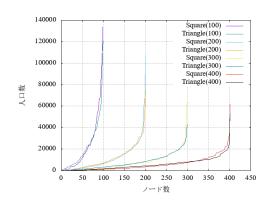
5x10



Square(100) Triangle(100) 4.5x10 Square(200) Triangle(200) Square(300) Triangle(300) 4x10 6 3.5x10 6 Square(400) Triangle(400) 3x10 6 2.5x10<sup>6</sup> 2x10<sup>6</sup> 1.5x10<sup>6</sup> 1x10<sup>6</sup> 500000 150 200 250

図 4.43: 福岡エリア負荷図

図 4.44: 福岡エリア負荷累積分布図



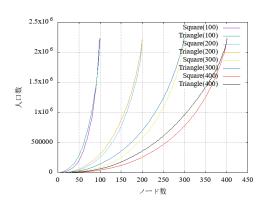


図 4.45: 広島エリア負荷図

図 4.46: 広島エリア負荷累積分布図

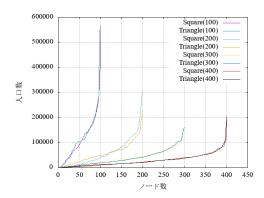


図 4.47: 京阪エリア負荷図

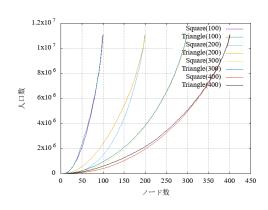


図 4.48: 京阪エリア負荷累積分布図

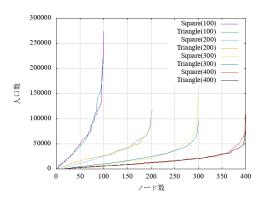


図 4.49: 名古屋エリア負荷図

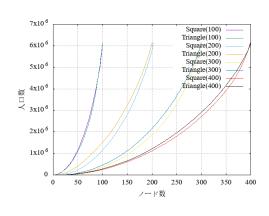


図 4.50: 名古屋エリア負荷累積分布図

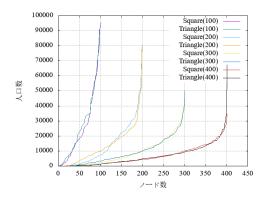


図 4.51: 札幌エリア負荷図

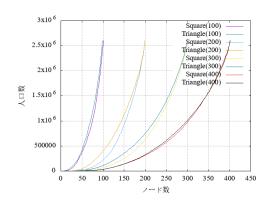
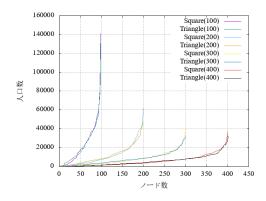


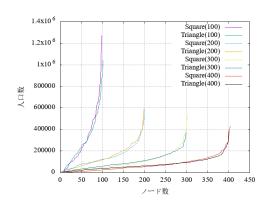
図 4.52: 札幌エリア負荷累積分布図



2.5x10 6
2x10 6
2x10 6
1.5x10 6
1x10 6
1x10 6
500000
0 50 100 150 200 250 300 350 400 450

図 4.53: 仙台エリア負荷図

図 4.54: 仙台エリア負荷累積分布図



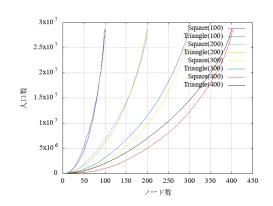


図 4.55: 東京エリア負荷図

図 4.56: 東京エリア負荷累積分布図

負荷図から見ると正方形と直角三角形の傾向はほぼ同じ。正方形と直角三角形 MSQ ネットワークについて同様に、ノード数が多いほど負荷の分布はより均等している。また、累積分布図から、正方形の線は基本三角形の右側にいる、ということは三角形 MSQ ネットワークの負荷分布は正方形より均等している。

### 4.3 最短経路の抽出

最短経路の抽出には、PythonのNetworkXといライブラリを用いた。NetworkXはグラフ理論の色々なアルゴリズムを集成し、ネットワーク分析において良く使われるライブラリである。

MSQネットワークは面グラフなので、任意の二ノード間の直線と交わる面の局所情報だけで最短距離を求めることが出来る、第 3.3 章に参照 [3]。そこで、ノード(s,t)間の直線と交わる全ての面を見つけ出すに、それらの面のノード情報と辺の情報をグラフ G に入れ、NetworkX のダイクストラ法アルゴリズムを使って、グラフ G の上でノード(s,t)の最短経路を見つけ出した。(図 4.57 を参照)

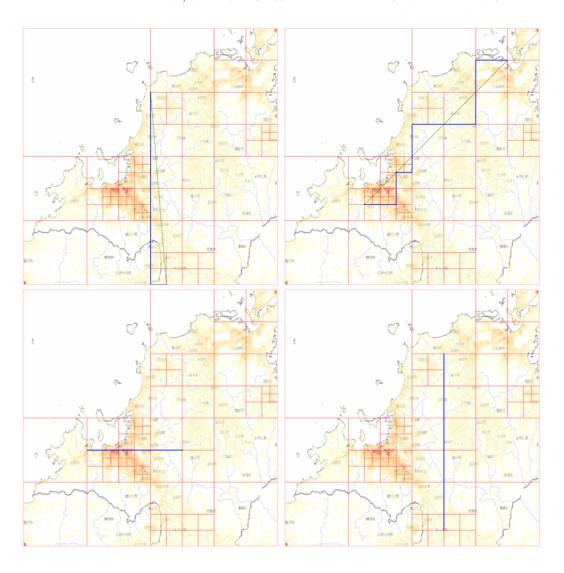


図 4.57: 最短経路見取り図

MSQネットワークは面グラフので、対称性によって、最短経路の path が対称する場もある、図 4.58(1)を参照。そこで、配分方式は二つがある、通信量を二分の一つづ分けて二つの path に均等配分するとランダムで一つの path を選んで配分する。また、最短経路中の 1 ホップに対して、二つの辺で対応する場合もある、図 4.58(2)を参照。そこの配分方式も二つが有る、通信量を二分の一つづ分けて二つの辺に均等配分するとランダムで一つの辺を選んで配分する。よって、本実験は二つの配分方式で評価する。

 $\frac{1}{2}$  均等配分方式 : path の均等配分と辺の均等配分の組み合わせ。

ランダム配分方式: path のランダム配分と辺のランダム配分の組み合わせ。

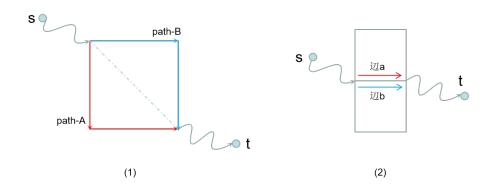


図 4.58: 配分方式の見取り図

第 3.4章によって、全ての End-to-End 最短経路を抽出してから、各面に対して各辺の通信量  $\lambda_{f(e)}$  と経由回数の重み  $w_{f(e)}$  も計算できる。ここで、二つ配分方式に分けて、それぞれの通信量  $\lambda_{f(e)}$  と経由回数の重み  $w_{f(e)}$  を計算する。両方の結果を比較して、評価する。

# 第5章 最適 $\mu_f$ 値による実験結果

4.3章から得た通信量  $\lambda_{f(e)}$  と経由する回数  $w_{f(e)}$  によって、 $\mu_f$  の最適値を計算できる。Python の Scipy ライブラリはニュートン法を集成したので、本実験の  $\mu_f$  値は Scipy による得た結果である。ここで、S100 は 100 ノードサイズの正方形 MSQ ネットワーク、T100 は 100 ノードサイズの直角三角形 MSQ ネットワーク。すなわち、S は正方形、T は三角形、数値はノードサイズである。

## 5.1 $\frac{1}{2}$ 均等配分の結果

#### 5.1.1 最適 $\mu_f$ 値の分布

図 5.1 から図 5.14 は七都市ごとに各ノードサイズの最適  $\mu_f$  値の分布図とその累積分布図。図から、200 ノードサイズの正方形 MSQ ネットワークの  $\mu_f$  値の分布は 100 ノードサイズの直角三角形の  $\mu_f$  値分布はほぼ一致している。

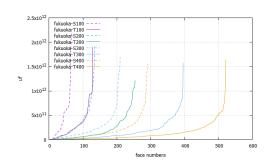


図 5.1: 福岡エリア uf 分布図

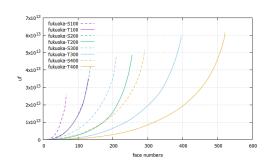


図 5.2: 福岡エリア uf 累積分布図

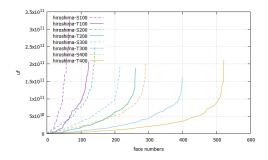


図 5.3: 広島エリア uf 分布図

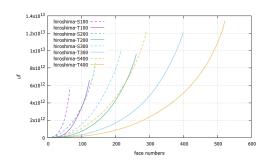


図 5.4: 広島エリア uf 累積分布図

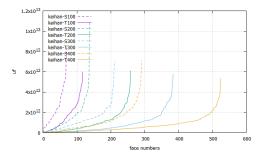


図 5.5: 京阪エリア uf 分布図

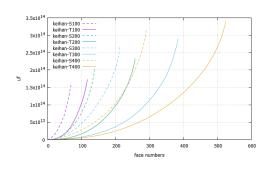


図 5.6: 京阪エリア uf 累積分布図

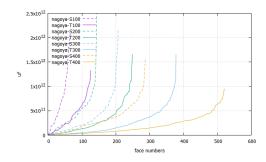


図 5.7: 名古屋エリア uf 分布図

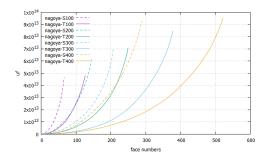


図 5.8: 名古屋エリア uf 累積分布図

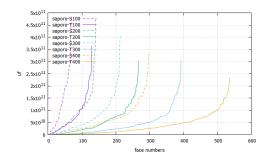


図 5.9: 札幌エリア uf 分布図

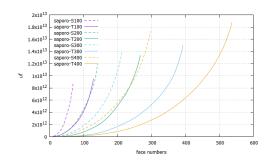


図 5.10: 札幌エリア uf 累積分布図

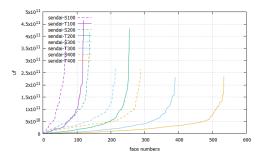


図 5.11: 仙台エリア uf 分布図

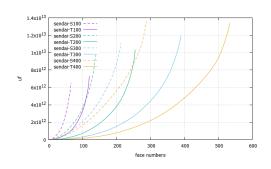


図 5.12: 仙台エリア uf 累積分布図

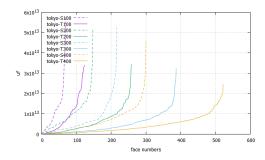


図 5.13: 東京エリア uf 分布図

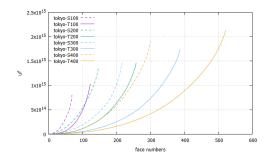
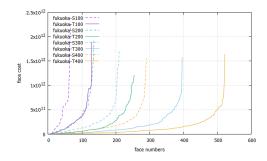


図 5.14: 東京エリア uf 累積分布図

### 5.1.2 面 cost の分布

図 5.15 から図 5.28 は第 3.4 章式(3.8)から、七都市ごとに各ノードサイズの面 コスト  $C_f$  分布図とその累積分布図。



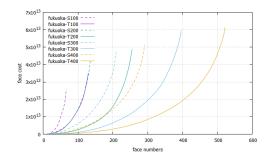
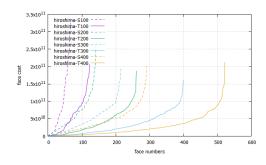


図 5.15: 福岡エリア面 cost 分布図

図 5.16: 福岡エリア面 cost 累積分布図



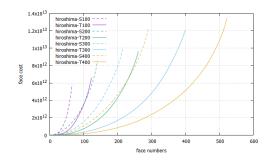
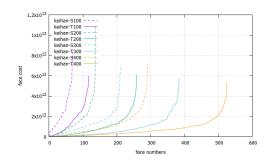


図 5.17: 広島エリア面 cost 分布図

図 5.18: 広島エリア面 cost 累積分布図



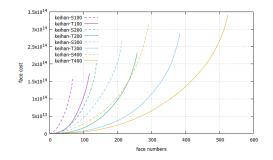
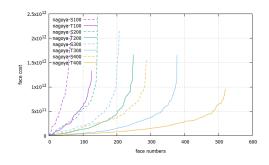


図 5.19: 京阪エリア面 cost 分布図

図 5.20: 京阪エリア面 cos 累積 t 分布図



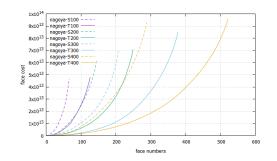
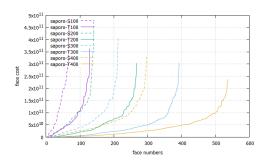


図 5.21: 名古屋エリア面 cost 分布図

図 5.22: 名古屋エリア面 cost 累積分布図



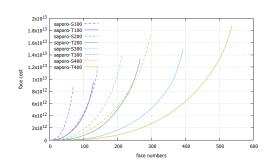
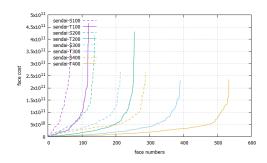


図 5.23: 札幌エリア面 cost 分布図

図 5.24: 札幌エリア面 cost 累積分布図



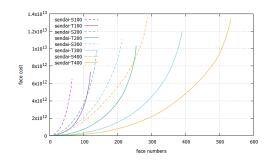
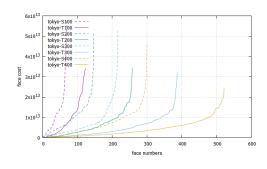


図 5.25: 仙台エリア面 cost 分布図

図 5.26: 仙台エリア面 cost 累積分布図



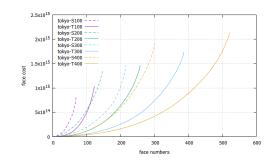


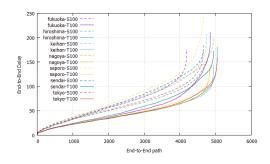
図 5.27: 東京エリア面 cost 分布図

図 5.28: 東京エリア面 cost 累積分布図

そこで、図から示すには、MSQ ネットワークの最適  $\mu_f$  値は指数分布に従う、面 cost の分布は最適  $\mu_f$  値の分布傾向は一致している。同じノードサイズの直角三角 形 MSQ ネットワークの総 cost は正方形 MSQ ネットワークの総 cost より大き。また、ノードが増えるほど総 cost が増えます。

#### 5.1.3 End-to-End(Path) 遅延の分布

図 5.29 から図 5.36 は各ノードサイズに七都市をまとめて比較する End-to-End(Path) 遅延の分布図とその累積分布図である。



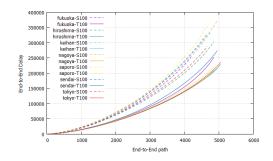
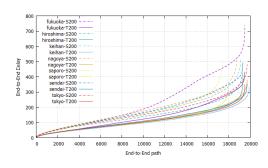


図 5.29: path 遅延分布図 (100node)

図 5.30: path 遅延累積分布図 (100node)



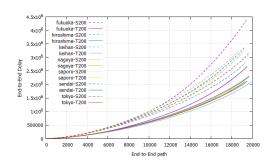
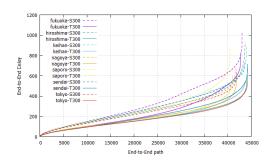


図 5.31: path 遅延分布図 (200node)

図 5.32: path 遅延累積分布図 (200node)

図から、全体的に多数の path 遅延が小さいが、極小数の path 遅延が大き。直角三角形 MSQ ネットワークの path 遅延は正方形より小さい。また、各都市ごとに、最適  $\mu_f$  値を満たすと path 遅延の分布はほぼ一致しています。



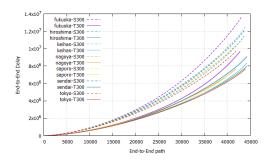
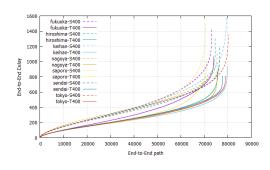


図 5.33: path 遅延分布図 (300node)

図 5.34: path 遅延累積分布図 (300node)



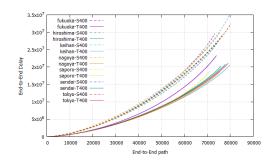


図 5.35: path 遅延分布図 (400node)

図 5.36: path 遅延累積分布図 (400node)

#### 5.1.4 総遅延、最適 $\mu_f$ 総値と面の関係

図 5.37 は第 3.4 章から、七都市エリアごとに全ての End-to-End 遅延の和(総遅延と呼ぶ)と面の関係図である。図から判るのは、正方形と直角三角形の面の個数が同じの場合正方形の総遅延は直角三角形より大き。正方形と直角三角形を問わず、面数が増えるほど総遅延が増えます。また、正方形の傾きが直角三角形より大きので、面数が増える同時に正方形の総遅延増加スピードが速い。最後、各都市ごとに、同じ形状の MSQ ネットワークの総遅延がほぼ一致している。

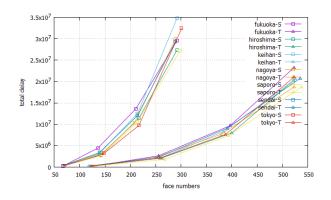


図 5.37: 総遅延と面の関係図

図 5.38 とは七都市エリアの最適  $\mu_f$  の総値と面の関係図である、図 5.39 は図 5.38 中で、total  $\mu_f \in [0,1 \times 10^{14}]$  範囲の拡大図。図から判るのは、正方形と直角三角形の面の個数が同じの場合正方形の最適  $\mu_f$  総値は直角三角形より大き。正方形と直角三角形を問わず、面数が増えるほど最適  $\mu_f$  の総値が増えます。また、第 4.1.2 章各都市エリアの人口分布図を参考すると、最適  $\mu_f$  の総値は各都市の総人口数とその人口分布に関係する、すなわち、エリアの総人口数が多いほど、ネットワークの総コストが大き。

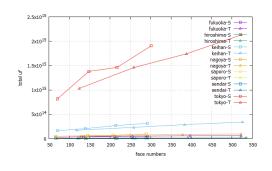


図 5.38: 最適 uf 総値と面の関係図

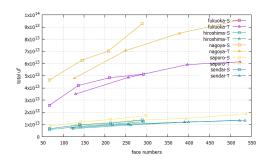


図 5.39: 拡大図

図 5.40 は最適  $\mu_f$  の総値と総遅延の関係図である。図 5.41 は図 5.40 中で、total  $\mu_f \in [0,1\times 10^{14}]$  の拡大図。図から判るのは、同じエリアで、正方形と直角三角形 MSQ ネットワークの最適  $\mu_f$  の総値が同じの時、直角三角形 MSQ ネットワークの総遅延の方が小さい。また、総遅延が同じ時、直角三角形 MSQ ネットワークの最適  $\mu_f$  の総値が大き。

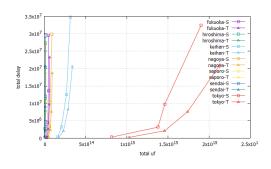


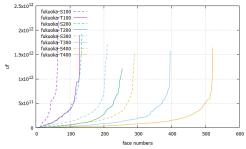
図 5.40: uf 総値と総遅延の関係図

図 5.41: 拡大図

### 5.2 ランダム配分の結果

### 5.2.1 最適 $\mu_f$ 値の分布

図 5.42 から図 5.55 は七都市ごとに各ノードサイズの最適  $\mu_f$  値の分布図とその累積分布図。



100 200 300 400 500 600 0 100 200 300 400 face numbers face numbers

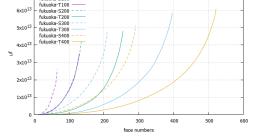
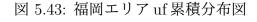
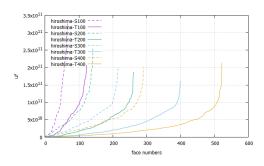


図 5.42: 福岡エリア uf 分布図





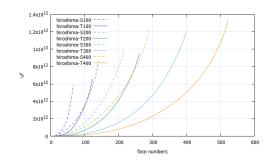
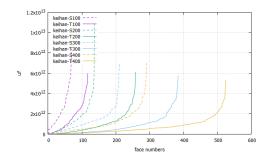


図 5.44: 広島エリア uf 分布図

図 5.45: 広島エリア uf 累積分布図



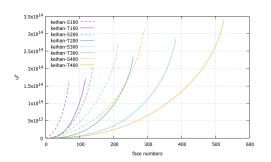


図 5.46: 京阪エリア uf 分布図

図 5.47: 京阪エリア uf 累積分布図

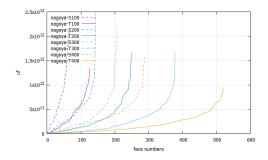
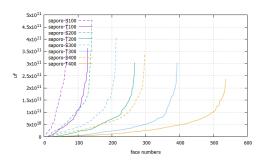


図 5.48: 名古屋エリア uf 分布図

図 5.49: 名古屋エリア uf 累積分布図



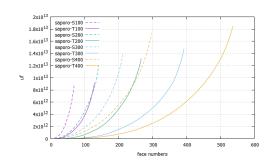
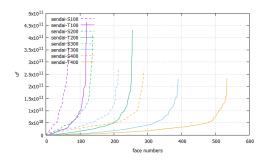


図 5.50: 札幌エリア uf 分布図

図 5.51: 札幌エリア uf 累積分布図



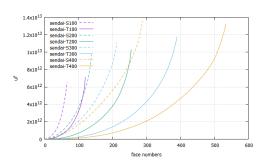
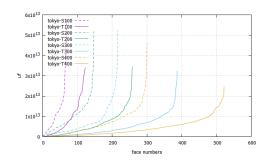


図 5.52: 仙台エリア uf 分布図

図 5.53: 仙台エリア uf 累積分布図



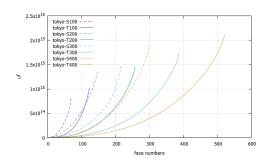
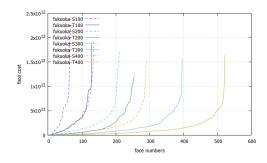


図 5.54: 東京エリア uf 分布図

図 5.55: 東京エリア uf 累積分布図

#### 5.2.2 面 cost の分布

図 5.56 から図 5.69 は七都市ごとに各ノードサイズの面コスト  $C_f$  分布図とその累積分布図。



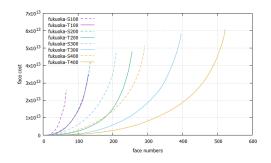
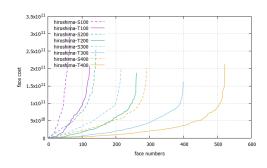


図 5.56: 福岡エリア面 cost 分布図

図 5.57: 福岡エリア面 cost 累積分布図



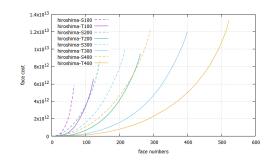
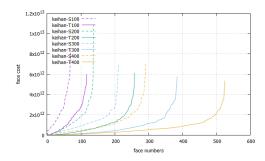


図 5.58: 広島エリア面 cost 分布図

図 5.59: 広島エリア面 cost 累積分布図



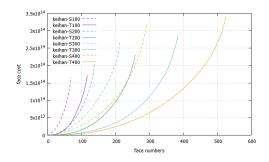
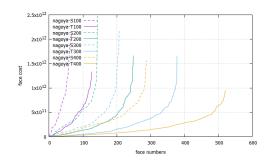


図 5.60: 京阪エリア面 cost 分布図

図 5.61: 京阪エリア面 cos 累積 t 分布図



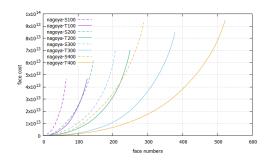
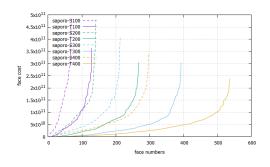


図 5.62: 名古屋エリア面 cost 分布図

図 5.63: 名古屋エリア面 cost 累積分布図



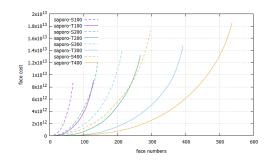
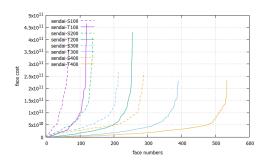


図 5.64: 札幌エリア面 cost 分布図

図 5.65: 札幌エリア面 cost 累積分布図



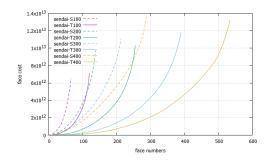
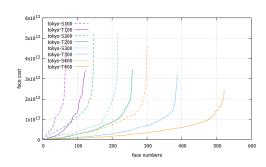


図 5.66: 仙台エリア面 cost 分布図

図 5.67: 仙台エリア面 cost 累積分布図



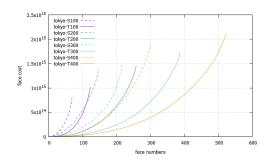
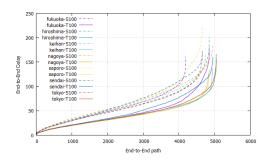


図 5.68: 東京エリア面 cost 分布図

図 5.69: 東京エリア面 cost 累積分布図

### 5.2.3 End-to-End(Path) 遅延の分布

図 5.70 から図 5.77 は各ノードサイズに七都市をまとめて比較する End-to-End(Path) 遅延の分布図とその累積分布図である。



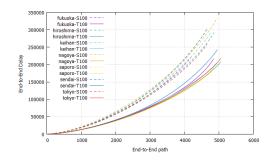
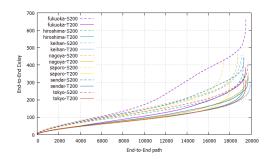


図 5.70: path 遅延分布図 (100node)

図 5.71: path 遅延累積分布図 (100node)



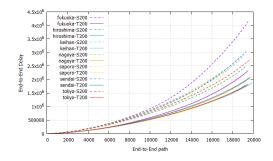
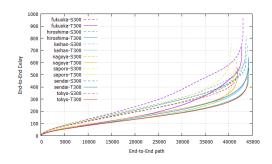


図 5.72: path 遅延分布図 (200node)

図 5.73: path 遅延累積分布図 (200node)



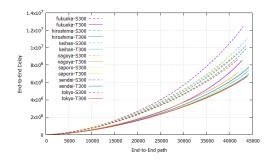
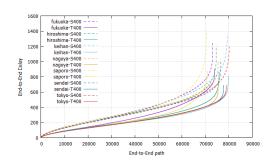


図 5.74: path 遅延分布図 (300node)

図 5.75: path 遅延累積分布図 (300node)



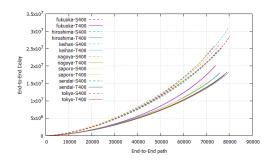


図 5.76: path 遅延分布図 (400node) 図 5.77: path 遅延累積分布図 (400node)

#### 5.2.4 総遅延、最適 $\mu_f$ 総値と面の関係

図 5.78 は七都市エリアの総遅延と面の関係図である。

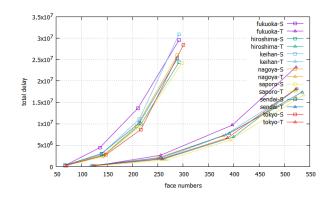
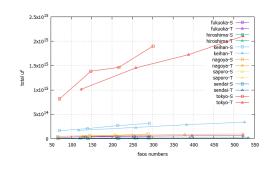


図 5.78: 総遅延と面の関係図

図 5.79 とは七都市エリアの最適  $\mu_f$  の総値と面の関係図である、図 5.80 は 5.79 の拡大図。

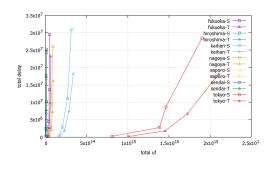


1xt0<sup>14</sup>
9xt0<sup>13</sup>
8xt0<sup>13</sup>
6xt0<sup>13</sup>
6xt

図 5.79: 最適 uf 総値と面の関係図

図 5.80: 細部図

図 5.81 は最適  $\mu_f$  の総値と総遅延の関係図である。図 5.82 は図 5.81 の拡大図。



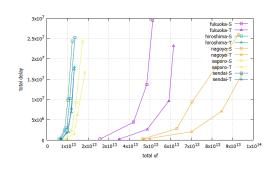


図 5.81: uf 総値と総遅延の関係図

図 5.82: 細部図

結論: ランダム配分の結果図と 5.1 章 ½ 均等配分の結果図の傾向は一致しています。5.1 章図の評価結論はここで応用できます。

# 5.3 $\frac{1}{2}$ 均等配分とランダム配分比較

図 5.83 から図 5.96 は第 4.3 章による、各都市ごとに、 $\frac{1}{2}$  均等配分とランダム配分の総コストと総遅延の比較図。青色線は平均配分、赤色線はランダム配分である。

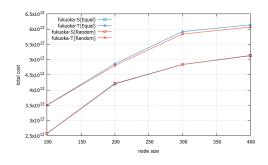
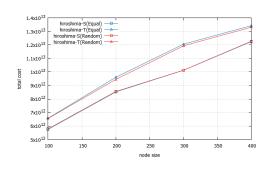


図 5.83: 福岡エリア総コスト比較図

図 5.84: 福岡エリア総遅延比較図



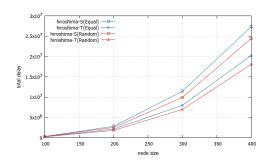
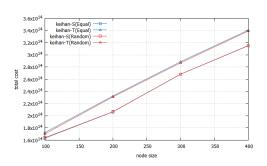


図 5.85: 広島エリア総コスト比較図

図 5.86: 広島エリア総遅延比較図



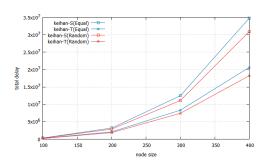


図 5.87: 京阪エリア総コスト比較図

図 5.88: 京阪エリア総遅延比較図

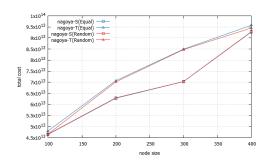
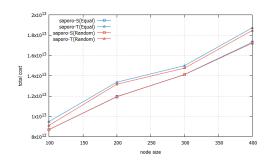


図 5.89: 名古屋エリア総コスト比較図

図 5.90: 名古屋エリア総遅延比較図



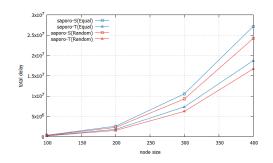
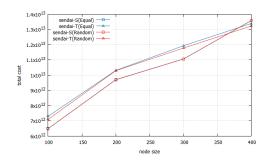


図 5.91: 札幌エリア総コスト比較図

図 5.92: 札幌エリア総遅延比較図



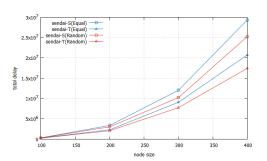
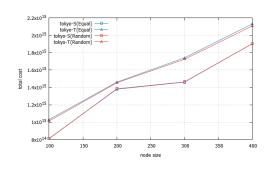


図 5.93: 仙台エリア総コスト比較図

図 5.94: 仙台エリア総遅延比較図



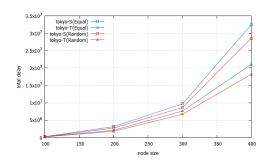


図 5.95: 東京エリア総コスト比較図

図 5.96: 東京エリア総遅延比較図

図から、各エリアごとに、 $\frac{1}{2}$ 均等配分とランダム配分の正方形 MSQ ネットワークの総コストはほぼ一致しています、ランダム配分の直角三角形 MSQ ネットワークの総コストは過ごし  $\frac{1}{2}$  均等配分より小さい。まだ、正方形と直角三角形 MSQ ネットワーク問わず、ランダム配分の総遅延は  $\frac{1}{2}$  均等配分の総遅延より小さい。すなわち、同じコストで、ランダム配分方式の方が小さい遅延を得られる、効率が良い。

# 第6章 おわりに

本研究では被災地の通信をリカバリするため、DTN技術の実現手法として面の 再帰分割に基づくネットワークに着目した。無人機が情報を転送する時、待ち行列 によって MSQ ネットワーク各面内無人機のサービス率(最適  $\mu_f$  値)を求めた上 で、正方形と直角三角形 MSQ ネットワークの End-to-End 遅延を比較して、より 適したネットワーク形状を探った。第五章の実験結果から、各地域の最適 $\mu_f$ 値の 分布は各地域の人口分布と同じく指数分布に従う。言い換えると、人口密集箇所 により多くの無人機を配分する必要がある。各地域ネットワークの総コストと各 地域の総人口数については、人口数が多いほど、ネットワークの総コストが大き い。また、同じサイズのネットワークで、正方形 MSQ の総コストは直角三角形よ り小さいが、End-to-End 遅延は三角形より大きく、すなわち、サービス率として 最適 $\mu_f$ 値が満たされる時、直角三角形 MSQ ネットワークがより小さい遅延が得 るので直角三角形ネットワークを選ぶほうが良い。サービス率 $\mu_f$ が制限される時、 同じ遅延で正方形の方がコストが小さいので正方形ネットワークを選ぶ方が良い。 一方、 🖥 均等配分方式とランダム配分方式を比較すると、総コストと End-to-End 遅延両方で比較して、同じコストでランダム配分方式の方が小さい遅延が得られ る。

正方形と直角三角形 MSQ ネットワーク以外にも様々な形状の MSQ ネットワークが存在する (例えば椅子型 MSQ ネットワーク) ので、それらについての分析は今後の課題として考えられる。

# 参考文献

- [1] 林幸雄, "Scale-free ネットワークの生成メカニズム", 応用数理, Vol. 14, No.4, 2004.
- [2] 鶴 正人, 内田 真人, 滝根 哲哉 他, DTN 技術の現状と展望, 電子情報通信学会 通信ソサイエティマガジン NO.16[春号],pp.57-68,2011.
- [3] 林 幸雄 著,"自己組織化する複雑ネットワーク 空間上の次世代ネットワーク デザイン",近代科学社,2014.
- [4] GAO YANG, "人口分布に基づいた遅延耐性通信ネットワークの メッセージフェリー最適配分問題", 修士論文, 2016.
- [5] 林幸雄, "ネットワーク科学の拡がり", 情報処理学会誌, Vol 49, No.3, 2008.
- [6] Yukio Hayashi, Recoverable DTN Routing based on a Relay of Cyclic Message-Ferries on a MSQ Network, IEEE Xplore Digital Library FoCAS 2015, pp.37-42, 2015.
- [7] 地域メッシュ統計の特質・沿革 (http://www.stat.go.jp/data/mesh/pdf/gaiyo1.pdf)