

ハイパーキューブモデルを用いた救急車の最適配置問題の研究

M2012MM039 祖父江裕介

指導教員：鈴木敦夫

1 はじめに

現在、図1のように全国の救急車の出動件数は2008年までは増加傾向にあり、2009年で一旦減少したが今後も増加していくと予想され、救急車の不足が問題となっている。そこで全国の市町村の消防局で予算の余裕のあるところでは新たに救急車を購入して配置することを考えている。しかし、配置場所については単に出動件数の多い地点の近辺に配置すればよいのか、救急車がよく出払っている消防署に配置すればよいのかといった配置を行うための定量的な基準が分からない。そこで本研究では、救急車の配置問題についてモデルを作成し、最適な救急車の配置解を計算する。

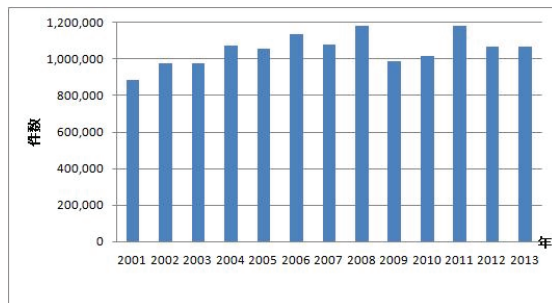


図1 全国の救急車の出動件数 (出典:総務省消防庁消防白書)

本研究では、待ち行列モデルの一種であるハイパーキューブモデルとして救急車の配置問題のモデル化を行う。ハイパーキューブモデルとは Larson[3,4,5] が提唱した確率モデルであり、系内客数といった無限の集合を状態と定義せず、0と1の組合せで定義することで、状態を有限の集合としてモデル化し、各状態の確率を収束させるモデルである。ハイパーキューブモデルでは救急車の台数を N としたとき、救急車 $i, i = 1, \dots, N$ が稼働しているとき状態を1、稼働していないときを0で表し、この状態を救急車の番号の降順に並べたものを系の状態と定義する。このとき系の状態の総数は 2^N 個となる。例えば、図3は3次元のハイパーキューブモデルの状態遷移図を示しており、全ての状態が3ビットの2進数で表した形となり、状態の総数は8個となる。ただし、ハイパーキューブモデルは全ての救急車が稼働している場合、新たに到着したサービス要求に対して救急車がサービスを行うことができないため呼損が起こるモデルとなる。また、ハイパーキューブモデルは海外で様々な救急システムでモデル化されている。待ち行列モデルのように状態を定義してモデル化を行うだけでなく、制約条件として平衡方程式を導入したり、目的関数として総移動時間を最小化したり、救急車のカバリング問題として組み合わせたモデル化もされている。

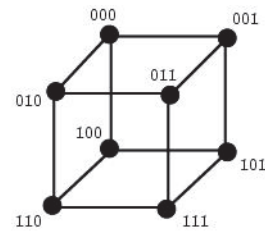


図2 ハイパーキューブモデルの状態遷移図

本研究では、先行研究 [2] のハイパーキューブモデルを応用したモデルの問題点を改良した。[2] では状態をきめ細かく定義したため、状態数が多くなり、状態遷移行列の次数が膨大で定常確率を求めるために非常に長い計算時間を要した。そこで本研究では、[2] よりも状態数を少ないモデル化を行い、短時間で定常確率を計算できるように作成する。このモデルの状態の総数は [2] とハイパーキューブモデルの中間とする。そして、救急車の配置を呼損率という指標を用いて評価し、最適な救急車の配置を計算する。

次に愛知県瀬戸市の救急システムについて説明する。瀬戸市には本署、東署、南署の3か所に消防署があり、図3のように瀬戸市は6個の管区に分割されている。そして、各消防署ごとにどの優先順位で管区にサービスを行うかということが定められている。また、瀬戸市は4台の救急車を所有しており、現在は本署に2台、東署に1台、南署に1台配備されている。そして、サービスを終えた救急車は元の消防署に帰り、消毒を受け、稼働していない状態に戻る。

救急車が稼働しているときは状態を1、稼働していないときを0とするようなハイパーキューブモデルを考える。そして以下に述べるモデル1のようにハイパーキューブモデルを応用した状態を定義する。そのモデルにおいて状態遷移行列を作成することで平衡方程式が得られ、定常確率が計算できる。その結果、どのような数値結果が得られ、先行研究とどのような違いが存在するか分析を行う。

次に愛知県名古屋市の救急システムについて説明する。名古屋市では16の行政区に分かれており、現在38台の救急車が配備されている。救急車が配備されている消防署は本署と出張所の2種類がある。本署は各行政区に対して1か所存在し、救急車が各1台配備されているため合計16台が本署に配備されている。出張所は行政区によって数が異なっており、合計22台の救急車が配備されている。ただし、救急車が配備されていない出張所も存在する。次の図4は名古屋市の行政区の地図と2014年現在救急車が配備されている消防署を表す。濃い点为本署を表す

し、薄い点が出張所を表している。

次に出勤区について説明する。瀬戸市のモデルの管区と同様に名古屋市では出勤区が定められている。瀬戸市と異なり、各行政区に対して細かく分割されており、各消防署から距離が小さいほど出勤区への優先順位が高くなる。例として、次の図5は2008年時の名古屋市北区の出勤区を表す。ただし、出勤区は年々変わっているのでこの図は最新版ではない。このモデルをハイパーキューブモデルとしてモデル化を行い、16か所の本署のうち4か所の本署に新しく各1台救急車を配置した場合にどのように呼損率が変化するか分析を行う。



図3 瀬戸市における管区 [2]



図4 名古屋市消防署地図

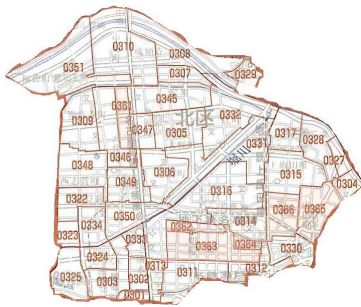


図5 2008年北区出勤区

2節では愛知県瀬戸市についてモデル化を行い、本

研究で提案するモデル1について説明し、[2]のモデルと各モデルの数値結果を比較する。次に3節で愛知県名古屋市のモデル化を行い、最適な配置解を計算する。最後に4節で結果をまとめる。

2 瀬戸市の救急車配備のモデル

2.1 記号の定義

N : 救急車の台数

B_i : 救急車 i の状態, $i = 1, \dots, N$

$B_N \dots B_1$: 全体の状態

P_Q : 呼損率

2.2 モデル1

モデル1では、ハイパーキューブモデルに各消防署に救急車を配置させる台数という情報を加えたモデルである。ハイパーキューブモデルは論文[1]で救急車の台数を増加させた場合でも同様に計算できることが述べられている。したがって、モデル1では救急車の状態を降順に並べたものと定義する。

$$B_i = \begin{cases} 0: \text{消防署 } i \text{ の救急車が } 1 \text{ 台も稼働していない} \\ 1: \text{消防署 } i \text{ の救急車が } 1 \text{ 台稼働している} \\ 2: \text{消防署 } i \text{ の救急車が } 2 \text{ 台稼働している} \end{cases} \quad (1)$$

例えば、本署に2台、東署に1台、南署に1台(本, 本, 東, 南)が配備されているとき状態遷移図は図6のようになる。

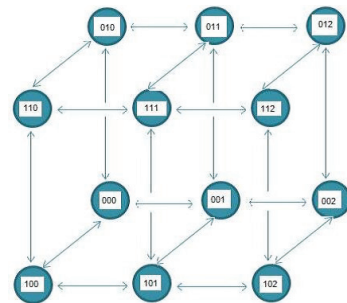


図6 (本, 本, 東, 南) 状態遷移図

この状態遷移図から平衡方程式を用いることで各状態の確率が得られる。

2.3 数値結果

本研究では平成13年から平成19年までの瀬戸市の実際のデータを用いてMATLABで数値計算を行った。ただしMATLABでは連立一次方程式の解法にガウス消去法を用いているため、 $Ax = 0$ ($A: n \times n$ の行列, $x: n$ 次のベクトル) のような0と等しい連立一次方程式では解くことができないため、一番最後の等式を確率の総和が1である式に直して計算を行った。ただし、到着率については[2]と比較できるように[2]に合わせた到着率を用いて計算している。呼損率については次の表1の結果が得られた。

表 1 モデル 1,[2] のモデルの呼損率の比較

P_Q	(本, 本, 東, 南)	(本, 東, 東, 南)	(本, 東, 南, 南)
モデル 1	0.00214	0.00223	0.00219
[2] のモデル	0.0023	0.0025	0.0027

※(本, 本, 東, 南)は本署に2台、東署に1台、南署に1台、(本, 東, 東, 南)は本署に1台、東署に2台、南署に1台、(本, 東, 南, 南)は本署に1台、東署に1台、南署に2台配備されていることを表す

モデル 1 では呼損率が最も小さい(本, 本, 東, 南)の場合が最適な配置解となった。この呼損率の数値は先行研究と 10^{-3} 桁まで同じ値になるような [2] のモデルに近い数値結果となった。[2] のモデルに誤差が出たため、[2] のモデルより正確性が少し劣る形となった。

以上から、モデル 1 は [2] のモデルを近似できることが分かった。計算時間については、[2] では 470 秒以上かかっていたものが、モデル 1 では 1 秒以下で計算できた。したがって、大規模な問題にモデル 1 を適用することができる。

3 名古屋市の救急車配備のモデル

瀬戸市のモデルと同様にハイパーキューブモデルでモデル化を行うと、名古屋市の場合は最低でも 38 台の救急車が配備されているので、状態の総数は $2^{38} \approx 2700$ 億となる。状態遷移行列の次元が約 2700 億次となるため、スパース行列で状態遷移行列を定義したとしても状態数が大規模で平衡方程式を解くことができない。

したがって、本研究ではモデルを縮小して解くことを考える。そこで本研究では、縮小を行う方法として 38 台の救急車のうち 22 台の出張所の救急車を稼働していないものとみなし、本署の 16 台のみで計算を行う。さらに行政区の統合を行い、モデルの縮小を行う。統合の方法としては各行政区に向かう出勤数の大きい行政区間を統合する。例えば、港区と南区は隣に位置するが出勤数は小さいため、統合を行わない。そして次の二種類の方法で行政区の統合を行いモデルの縮小を行っていく。

3.1 行政区統合方法 1

このモデルでは以下の 5 個の区分に統合して計算を行う。

表 2 名古屋市行政区 統合方法 1

区分	行政区
区分 1	西区, 北区, 中区, 東区
区分 2	中村区, 中川区, 港区
区分 3	守山区, 千種区, 名東区
区分 4	昭和区, 天白区, 緑区
区分 5	熱田区, 南区, 瑞穂区

区分 1 に 4 台、区分 2 に 3 台、区分 3 に 3 台、区分 4 に 3 台、区分 5 に 3 台配備されているようなモデルとなり、

区分内でハイパーキューブモデルとしてモデル化を行う。ただし、優先順位は各区分から出勤数が大きい順とする。瀬戸市のモデルと同様に MATLAB を用いて計算を行った。表 3 は全配置の中から呼損率が小さい配置をまとめた表である。「配置」は区分 1 から 5 まで配置する救急車の台数を区分 1 から 5 の順に並べたものである。この配置の中で呼損率が最小であるのは配置 10003 である。したがって、区分 1 に 1 台、区分 5 に 3 台に配置したとき、最適な配置解となる。実行時間は状態の総数によって多少異なるが、1 つの配置解当たり 60 分程度で計算することができた。

以上の計算で区分 5 の熱田区、南区、瑞穂区と区分 1 に 1 台配置すればよいことが分かったが、区分 1 のどの行政区に配置すればよいか分からない。次に、区分 1 である西区、北区、中区、東区に対して新しく 1 台配置したときにどの配置解が最も呼損率が小さくなるか計算を行う。ただし、区分 1 内のみで到着とサービスが行われるものとする。また、救急車の優先順位は出勤数が大きい順とした。表 4 は区分 1 に 1 台配置したときの呼損率を表す。表 4 から東区に救急車を配置したとき呼損率が最も小さくなったので、区分 1 からは東区に配置すればよいことが分かった。

表 3 名古屋市における呼損率

配置	呼損率	状態の総数	実行時間(秒)
40000	0.00003196	2304	2221
00013	0.00003227	2800	3253
00103	0.00003234	2800	3319
01003	0.00003735	2800	3463
※ 10003	0.00002932	2688	3199
30001	0.00003065	2560	2705
30010	0.00003362	2560	2379
30100	0.00003337	2560	2705
31000	0.00003903	2560	2718
00202	0.00003847	2880	3465
20002	0.00002984	2688	2999
20020	0.00003970	2688	3134
20200	0.00003581	2688	3116
22000	0.00006503	2688	3109
00112	0.00003570	3000	3877
10012	0.00003269	2880	3594
10102	0.00003222	2880	3572
11002	0.00003816	2880	3571
10201	0.00003640	2880	3611
10210	0.00003950	2880	3588
20011	0.00003297	2800	3381
20101	0.00003245	2800	3407
21001	0.00003846	2800	3411
20110	0.00003574	2800	3378
10111	0.00003556	3000	3916

※は呼損率が最も小さいものを表す

表 4 区分 1 に 1 台配置したときの呼損率

配置場所	呼損率	実行時間 (秒)
西区	0.0238	0.352996
北区	0.0227	0.299397
中区	0.0220	0.298447
東区	0.0212	0.295993

3.2 行政区統合方法 2

このモデルでは以下の 8 個の区分に統合して計算を行う。

表 5 名古屋市行政区 統合方法 2

区分	行政区
区分 1	港区
区分 2	中村区, 中川区
区分 3	緑区, 天白区
区分 4	中区, 熱田区
区分 5	南区, 瑞穂区
区分 6	西区, 北区
区分 7	守山区, 名東区
区分 8	千種区, 東区, 昭和区

この統合方法 2 では統合方法 1 よりも状態の総数が大きく、MATLAB で平衡方程式を解くことができない。そこで本研究では名古屋大学情報基盤センターのスーパーコンピュータ (以下スパコン) を使用して計算を行う。名古屋大学のスパコンで利用可能な数値計算ライブラリ NUPAC の非対称で密な連立一次方程式を解く関数 LEQLUS を用いて Fortran でプログラミングし、定常状態の確率を計算した。しかし、一つの配置当たりの実行時間は短い場合は約 1 時間 30 分、長い場合には約 14 時間半かかり、時間を非常に要した。そのため本研究では全配置 123 通りの呼損率を計算することができなかった。計算を行った配置の中で最も呼損率が小さかった配置について説明を行う。

区分 4 に 1 台、区分 8 に 3 台配置したとき呼損率が 0.00002582 となり最小となった。モデル 3 と同様に区分 4 についてどちらの消防署に配置した場合に呼損率が小さくなるかについて計算を行う。

表 6 区分 4 に 1 台配置したときの呼損率

配置場所	呼損率	実行時間 (秒)
中区	0.013596	0.087272
熱田区	0.013845	0.036490

表 6 の呼損率から中區に配置した方が呼損率が小さくなった。以上の計算から中区、千種区、東区、昭和区に配置したとき最適な配置であることが分かった。

4 おわりに

本研究で、ハイパーキューブモデルを応用して救急車の配置問題のモデル化を行い、呼損率という指標を用い

て最適な配置解を計算した。

モデル 1 では呼損率が最も小さい (本, 本, 東, 南) が最適な配置解となった。この呼損率の数値は [2] のモデルと 10^{-3} 桁まで同じ値となり、[2] のモデルに十分近い数値となった。したがってモデル 1 が [2] のモデルを十分近似でできることが分かった。

名古屋市のモデルでは救急車の台数が 38 台であり、救急車を配置しなかった場合でも状態の総数が約 2700 億となり、非常に大規模な問題となった。2700 億次の状態遷移行列では平衡方程式を解くことができなかつたため、モデルを縮小させて解いた。その結果、統合方法 1 では東区、熱田区、南区、瑞穂区に配置した場合が最も呼損率が小さく、統合方法 2 では中区、千種区、東区、昭和区に配置した場合が最も呼損率が小さいという結果が得られた。中村区、中川区といった到着率の高い行政区に配置されるのではないかと思われたが、周辺に配置した方が呼損率が小さくなる場合があることが分かった。

今後の課題として、本研究のハイパーキューブモデルではポアソン到着、指数サービスで計算を行っている。指数サービスを時間に従う一般の分布として計算することでより現実に近いモデルになると考えられる。また、サービス時間に距離データを用いることでサービス時間が救急車ごとに一定にならないと考えられる。名古屋市のモデルでは 38 台全ての救急車についてハイパーキューブモデルを用いてモデル化することができなかつたので、これをシミュレーションを行うことで厳密解とは異なるが、厳密解に近い値を計算することができると考えられる。名古屋市の縮小したモデルについて救急車の稼働率を分析するために、呼損率だけでなく現場到着時間を最小化するようなモデルについても計算を行うべきだと考えられる。状態遷移行列をスパース行列として行列を定義することで MATLAB でメモリ容量を抑えて計算することができる。

参考文献

- [1] B.Boyaci and N.Geroliminis (2011):Extended Hypercube Models for Large Scale Spatial Queuing Systems, *Urban Transport Systems Laboratory*, 22-40, LUTS, May 2011.
- [2] 稲川敬介 (2005):混雑を考慮した緊急車両配置問題の研究, 南山大学大学院経営学研究科 2005 年度修士論文
- [3] Larson, R. (1974):A hypercube queuing model for facility location and redistricting in urban emergency services, *Computers & Operations Research*, 1 (1) 67 - 95.
- [4] Larson, R. (1975):Approximating the performance of urban emergency service systems, *Operations Research*, 23 (5) 845-868, September-October 1975.
- [5] Larson, R. and A. Odoni (1981):*Urban Operations Research*, Prentice-Hall, Englewood Cliffs, N.J.