

TR-I-0004

## 統計的手法による音韻認識

Phoneme Recognition Method using  
a Stochastic Approach (HMM)

橋本 泰秀† 川端 豪

Yasuhide HASIMOTO Takeshi KAWABATA

1987.5

## 概要

音声認識における統計的な手法の利用は、識別関数の設計や距離尺度の構成などに、古くから多く行われているが、特にここ数年の間に隠れマルコフモデル(HMM)を用いる方法が注目されだしている。HMM手法の音声認識に対する利用法は、単語を認識対象とする場合と音素を認識対象とする場合に大別されるが、本報告では音素の中でも特に識別の難しい有声破裂音(/b/, /d/, /g/)を、ベクトル量子化及びHMMの手法を用いて認識することを試みる。音素のモデルとして3状態からなるレフト・トゥー・ライトマルコフモデルを用い、バウム・ウェルチ(Baum-Welch)のアルゴリズムによってモデルの学習を行う。また、トレリス(Trellis)のアルゴリズムを用いて認識実験を行う。学習に使用したデータを認識評価実験にも用いた場合(すなわちcloseの条件)における有声破裂音認識率は、平均93.5%であった。

なお本報告書は、豊橋技術科学大学情報工学過程4年次 橋本 泰秀 が昭和62年1月8日~2月28日の期間にATR自動翻訳電話研究所において行った研究を、実務訓練報告書としてまとめたものである。

ATR自動翻訳電話研究所

## 1 概要

統計的手法による音声認識のアルゴリズムは、これまで種々提案されているが本実習では、ベクトル量子化と、ヒドンマルコフモデル(HMM)を応用した音素認識を行ったので、これを報告する。

まず、視察によってラベル付けされたいくつかの単語から音素区間を切り出し、同種音素をまとめ、音素別音声ファイルを作成する。そして、実際認識を行うカテゴリーの音声ファイルを取り出し、音声パワーの重み付き尤度比(PWLR)を距離尺度とするベクトル量子化を行い、64個の音声ベクトルらからなるコードブックを作成する。そして、音素別音声ベクトルと各コードブックベクトルとの距離を計算し、各音素別音声ベクトルと最小距離を与えるコードブックのベクトル番号の系列に変換する。こうして変換された音素別音声ベクトルを、HMMのトレーニングデータとして使用し、HMMの初期モデルは、このトレーニングデータから求める。

話者は成人女性1人、認識カテゴリーを有声破裂音(/b/, /d/, /g/)の3カテゴリーとし、HMMは3ステート、ヌルアークなしのレフト・トゥー・ライトモデルとして、バウム・ウェルチ(Baum-Welch)のアルゴリズムによって500回学習後、学習に使用したトレーニングデータをトレリス(Trellis)のアルゴリズムによって認識した結果、平均93.5%の子音認識率を得ることができた。

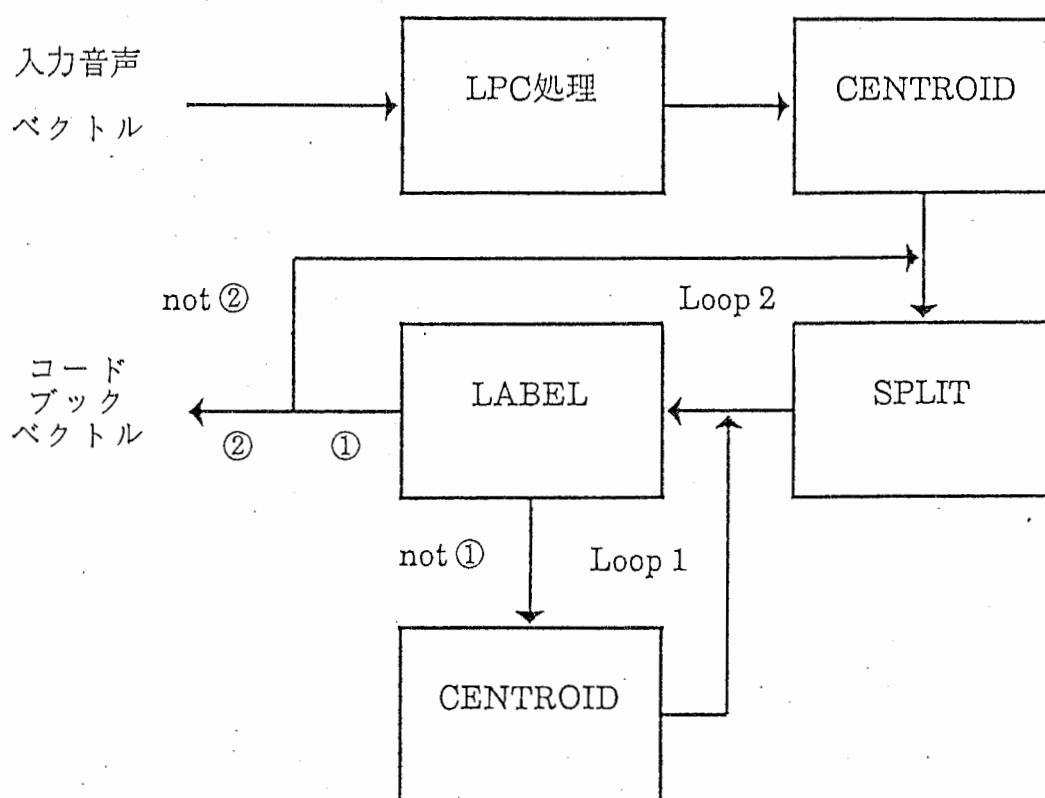
以下、2章ではベクトル量子化について、3章ではHMMについて、4章ではこれら2つの手法を応用した認識実験方法及び結果について述べ、5章ではこの実験について考察し、6章で結論を述べることにする。

## 2 ベクトル量子化

ベクトル量子化の目的は、多数のベクトルからなるベクトル母集団を、ある有

限個の代表的ベクトル(コードブックベクトル)によって表現することにある。そして、母集団中の各ベクトルはコードブックベクトルの中の1つで近似されるが、コードブックベクトルの選び方が悪いと、量子化ひずみ(ディストーション)が大きくなり後々の認識率に影響を及ぼす。したがって、コードブックベクトルはなるべく正確なものを選びたい。

本実習で使用した、音声パワーの重み付き尤度比(PWLR)を距離尺度とするベクトル量子化のアルゴリズムは、図2.1のようなブロック図となる。



① :  $(pre.d - d < 0.0001) \text{ and } (k > 3) \text{ or } (k \geq 10)$

② : コードブックベクトル数  $\geq 64$

pre.d:更新前のディストーションの値

d :更新後のディストーションの値

k : Loop 1 の繰り返し回数

図2.1 ベクトル量子化ブロック図

次に、図2.1の各ブロックでの処理について述べる。

## 2.1 LPC処理

このブロックでは、次の処理を行う。

- (1) 入力音声ベクトルの自己相関係数に、パスカルウィンドウをかける。
- (2) 自己相関係数から12次線形予測係数を求め、これから16次ケプストラム係数を求める。
- (3) 自己相関係数中、低次12次元から16次自己相関係数を求める。
- (4) 音素中の最大パワーが1になるよう、パワーを正規化する。

## 2.2 ラベル(LABEL)

このブロックでは、すべての入力音声ベクトルについて各コードブックベクトルとの距離を計算し、最も近いコードブックベクトルの番号をラベルとしてラベル付けを行う。ここで、その距離尺度は

lcor(0) : 入力音声ベクトルのパワー

lcor(i) (1 ≤ i ≤ 16) : 入力音声ベクトルのi次の自己相関係数

vcor(0) : コードブックベクトルのパワー

vcor(i) (1 ≤ i ≤ 16) : コードブックベクトルのi次の自己相関係数

lcep(i) (1 ≤ i ≤ 16) : 入力音声ベクトルのi次のケプストラム係数

vcep(i) (1 ≤ i ≤ 16) : コードブックベクトルのi次のケプストラム係数

dist : 入力音声ベクトルとコードブックベクトルとの距離

とすると

$$\text{dist} = 0.01 \left\{ \frac{\text{vcor}(0)}{\text{lcor}(0)} + \frac{\text{lcor}(0)}{\text{vcor}(0)} - 2.0 \right\} + \sum_{i=1}^{16} [\{\text{vcor}(i) - \text{lcor}(i)\} \{\text{vcep}(i) - \text{lcep}(i)\}] \quad \dots(2.1)$$

といったパワーの重み付き尤度比(PWLR)を使用しており、(2.1)式の0.01はその重みである。なお、コードブックベクトルとそのベクトルのラベルを持つ入力音声ベクトルとの距離の、全入力音声ベクトルにおける平均値をディストーションと呼び、図2.1でのd及びpre.dは、今行ったラベル付けでのディストー

ションの値及び Loop 1 で 1 ループ前でのディストーションの値であり、k は Loop 1 での繰り返し回数である。

### 2.3 セントロイド (CENTROID)

ラベル付けによって行われたラベルについて、同一ラベルの入力音声ベクトルを 1 つの部分集団とみなし、以下の処理をおこなう。

N : 部分集団のベクトル数

とすると

$$\begin{cases} \text{vcor}(0) = \left[ \frac{\sum_{j=1}^N \text{lcor}^{(j)}(0)}{\sum_{j=1}^N \{1/\text{lcor}^{(j)}(0)\}} \right]^{\frac{1}{2}} \\ \text{vcor}(i) = \left\{ \frac{\sum_{j=1}^N \text{lcor}^{(j)}(i)}{N} \right\} / N \quad (1 \leq i \leq 12) \end{cases} \quad \dots(2.2)$$

こうしてコードブックベクトルを、その部分集団の代表ベクトルとして妥当なものにするために修正していく。なお、図 2.1 において最初に通るセントロイドのブロックは、ラベル付けをまだ行っていないため、全母集団の代表ベクトルである最初のコードブックベクトル 1 個が求められる。このように求められたコードブックベクトルに対し、2.1 節の (2), (3) の処理をおこなう。

### 2.4 スプリット (SPLIT)

ある 1 つのコードブックベクトルの代表する部分集団を 2 つに分割し、コードブックベクトル数を倍にする処理をおこなう。まず、その部分集団中からコードブックベクトルと最も距離 (PWLR) の大きい音声ベクトルを見つける。次に、この音声ベクトルと同じ部分集団中の最も離れている音声ベクトルを見つける。こうして求めた 2 つの入力音声ベクトルと、その部分集団の代表ベクトルであるコードブックベクトルとから以下の処理をおこなって、2 つのコードブックベクトルを求める。

$$\begin{cases} \text{vcor}'(0) = \left[ \frac{2 \text{vcor}(0) + \text{lcor}(0)}{2/\text{vcor}(0) + 1/\text{lcor}(0)} \right]^{\frac{1}{2}} \\ \text{vcor}'(i) = \{2 \text{vcor}(i) + \text{lcor}(i)\} / 3 \quad (1 \leq i \leq 12) \end{cases} \quad \dots(2.3)$$

(2.3) 式を、一方の入力音声ベクトルとコードブックベクトル、他方の入力音声ベクトルとコードブックベクトルとの間で計算する。更に、すべてのコードブックベクトルに対してこの処理をおこなうことにより、コードブックベクトル

ル数が倍となる。この新しく作られたコードブックベクトルに対して、2.1節の(2),(3)の処理をおこなう。

## 2.5 コードブック

以上述べてきたような処理を行い、64個のコードブックベクトルが求まる。更に、HMMのためのトレーニングデータをつくるためには、入力音声ベクトルをコードブックベクトル番号の系列に変換しなければならない。その変換は、2.2節で述べたラベル付けのアルゴリズムを使用して行うことができる。

## 3 HMM

マルコフチェーンは、有限個のステートを持ち、遷移確率に従ってモデル内を移動する。観測者は、ただそのモデルから出力される確率だけしか見ることができず、マルコフチェーンの中のステートを直接観察することはできない。このようなブラックボックス的な意味から、ヒドンマルコフモデル (Hidden Markov Model) と呼ばれる。そして、ある入力の生起確率を求めるためには、各アーク (arc) に出力確率を持たせ、イニシャルステートからファイナルステートに達しマルコフチェーンの遷移がおわるまで、遷移する毎に遷移確率と出力確率を掛け合わせばよい。そして、認識を行うためには、カテゴリー毎にHMMを用意し、各モデルの出力する生起確率を比較するのである。ゆえに、各カテゴリーに対し遷移確率と出力確率の行列を求めることが要求される。次に、その行列の各パラメータの求め方について述べる。

### 3.1 HMMの理論

まず

$N$  : マルコフチェーンのステート数

$q_i$  ( $1 \leq i \leq N$ ) : マルコフチェーンのステート

$M$  : 出力値のカテゴリー数

$v_k$  ( $1 \leq k \leq M$ ) : 出力値

$\Pi = (\pi_1, \pi_2, \dots, \pi_N)$  : イニシャルステート分布ベクトル

$\Pi' = (\pi'_1, \pi'_2, \dots, \pi'_N)$  : ファイナルテスト分布ベクトル

$a_{ij}$  ( $1 \leq i, j \leq N$ ) :  $q_i$  から  $q_j$  へ遷移する確率

$b_{ij,k}$  ( $1 \leq i, j \leq N, 1 \leq k \leq M$ ) :  $q_i$  から  $q_j$  へ遷移するアークで  $v_k$  を出力する確率

$T$  : 入力の長さ

$O_t$  ( $1 \leq t \leq T$ ) : 入力  $\in \{v_1, v_2, \dots, v_M\}$

と定義すると、イニシャルステートからパスをたどっていき、あるアーク ( $q_i \sim q_j$ ) で出力値  $O_{t+1}$  を出力する確率、すなわち前向き確率は次のように再起的に表せる

$$\begin{cases} \alpha_1(i) = \pi_i \\ \alpha_{t+1}(j) = \sum_{i=1}^N \{ \alpha_t(i) \cdot a_{ij} \cdot b_{ij}(O_{t+1}) \} \end{cases} \quad (1 \leq t \leq T-1) \quad \dots(3.1)$$

同様に、ファイナルステートからパスをたどっていき、あるアーク ( $q_i \sim q_j$ ) で出力値  $O_{t+1}$  を出力する確率、すなわち後ろ向き確率は

$$\begin{cases} \beta_T(j) = \pi'_j \\ \beta_t(i) = \sum_{j=1}^N \{ a_{ij} \cdot b_{ij}(O_{t+1}) \cdot \beta_{t+1}(j) \} \end{cases} \quad (T-1 \geq t \geq 1) \quad \dots(3.2)$$

となる。そして、この  $\alpha_t(i)$  と  $\beta_t(i)$  を使用して生起確率  $P$  は

$$P = \sum_{i=1}^N \sum_{j=1}^N \{ \alpha_t(i) \cdot a_{ij} \cdot b_{ij}(O_{t+1}) \cdot \beta_{t+1}(j) \} \quad \dots(3.3)$$

と表される。この (3.1) ~ (3.3) 式を、フォワード・バックワード (forward-backward) アルゴリズムと呼ぶ。このアルゴリズムを使用することにより、次のように各パラメータを補正する。まず

$$\gamma_t(i, j) = \alpha_t(i) \cdot a_{ij} \cdot b_{ij}(O_{t+1}) \cdot \beta_{t+1}(j) \quad \dots(3.4)$$

と  $\gamma$  を定義すると、

$$\bar{a}_{ij} = \{ \sum_{t=1}^{T-1} \gamma_t(i, j) \} / \{ \sum_{j=1}^N \sum_{t=1}^{T-1} \gamma_t(i, j) \} \quad \dots(3.5)$$

$$\bar{b}_{ij,k} = \{ \sum_{t=1}^{T-1} \gamma_t(i, j) \} / \{ \sum_{t=1}^{T-1} \gamma_t(i, j) \} \quad \dots(3.6)$$

とパラメータを補正する。(3.4) ~ (3.6) 式はバーム・ウェルチのアルゴリズムと呼ばれ、入力の生起確率  $P$  を増大することができる。こうして得られた  $a_{ij}, b_{ij,k}$

$j, k$  は次のように正規化する

$$a_{ij} = \bar{a}_{ij} / \{ \sum_{j=1}^N \bar{a}_{ij} \} \quad \dots(3.7)$$

$$b_{ij,k} = \bar{b}_{ij,k} / \{ \sum_{k=1}^M \bar{b}_{ij,k} \} \quad \dots(3.8)$$

(3.7), (3.8)式のように正規化された  $a_{ij}, b_{ij,k}$  を再び (3.1)~(3.8)式に代入することにより HMM の学習がなされる。

### 3.2 スケーリング (Scaling)

3.1節のようにすれば、HMM の学習が可能であるが、ここで  $T \rightarrow \infty$  となると  $\alpha_T(i) \rightarrow 0, \beta_1(i) \rightarrow 0$  という問題が生じる。そこでスケーリング係数  $c_t$  を導入する。

$$c_t = \sum_{i=1}^N \alpha_t(i) \quad \dots(3.9)$$

とスケーリング係数を定義し

$$\alpha_t(i) = \alpha_t(i) / c_t \quad \dots(3.10)$$

$$\beta_t(i) = \beta_t(i) / c_t \quad \dots(3.11)$$

のように  $\alpha_t(i), \beta_t(i)$  にスケーリング係数をかける。(3.10), (3.11)式を(3.1), (3.2)式を実行した後行うことにより、この問題を解決することができる。

### 3.3 レフト・トゥー・ライトの HMM

HMM は、レフト・トゥー・ライト (left-to-right) の単純なモデルとする。そのモデルは、次のようなモデルである。

- (i) 最初の出力値  $O_1$  は、イニシャル状態  $q_1$  で出力される。
- (ii) 最後の出力値  $O_T$  は、ファイナル状態  $q_N$  で出力される。
- (iii) 1度マルコフチェーンがある状態から出たならば、その後再びその状態に帰ることはできない。

図 3.1 にこのレフト・トゥー・ライトモデルの例を示す。ただし、状態数  $N = 3$  である。本実習では、図 3.1 の HMM を使用することにする。

### 3.4 トレリスのアルゴリズム

以上述べてきたように実行すれば、学習された HMM を得ることができ、次に認識を行うわけであるが、その認識のためのアルゴリズムがトレリス (Trellis) のアルゴリズムである。それは、(3.1)式の  $\alpha_t(i)$  の計算と同じで、マルコフチェーンの通りうるパスの確率の和を順に計算していく。そして

$$P = \alpha_T(N) \quad \dots(3.12)$$



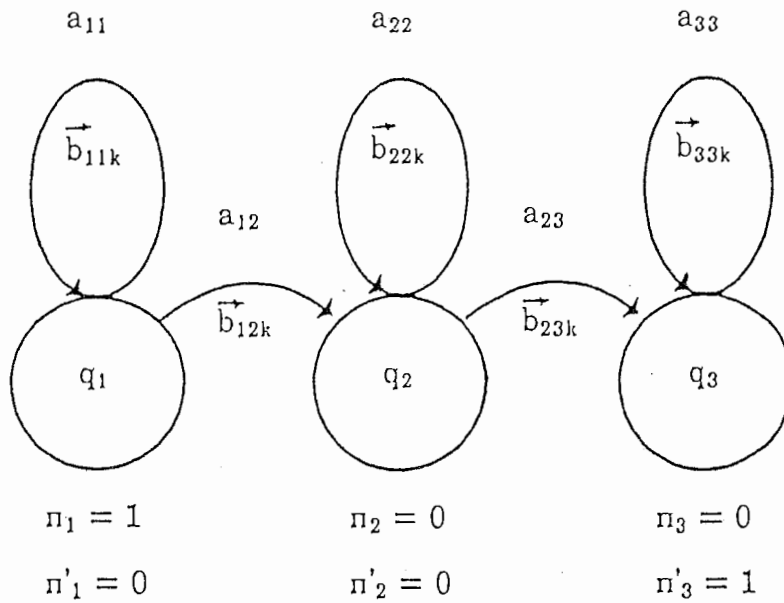


図 3.1 3 ステートのレフト・トゥー・ライト HMM

が、3.3 節のモデルを使用した場合、入力の生起確率となる。そして、この確率の対数を取り、対数生起確率を求めると

$$\log P = \log \alpha_T(N) \quad \dots(3.13)$$

となる。しかし、3.2 節で述べたようなスケール係数  $c_t$  がかかっているため、次のようにこの係数はずす必要がある。

$$\log P = \log \alpha_T(N) + \sum_{t=1}^{T-1} \log c_t \quad \dots(3.14)$$

(3.14) 式が、トレリスのアルゴリズムによる入力の生起確率となる。

## 4 音素認識実験

### 4.1 実験方法

(1) 音声データは、話者成人女性 1 人 (SK) の、新明解国語辞典第 2 版、最重要語および重要語、計 5244 単語中、音素の連鎖のバランスがよくなるよう選んだ 216 単語の中から、有声破裂音 (/b/, /d/, /g/) の区間だけを切り出して、得られるサンプル数を調べサンプル別に音素ファイルを作成する。

(2) (1) で得られた全音素ファイルを入力音声として、2 章で述べたベクトル量子化を行い、64 個のコードブックベクトルを作成する。次に、音素ファイルを

コードブックベクトル番号の系列に変換し、HMM用トレーニングデータを作る。

(3) (2)でできた各トレーニングデータを時間軸に沿って3等分し、その1つの長さの逆数をそのトレーニングデータの遷移確率  $a_{i,i+1}$  ( $1 \leq i \leq 2$ )とし、/b/,/d/,/g/別に相乗平均をとる。得られた各  $a_{i,i+1}$  より次式のように遷移確率  $a_{ii}$  ( $1 \leq i \leq 3$ )を求める。

$$\begin{cases} a_{ii} = 1 - a_{i,i+1} & (1 \leq i \leq 2) \\ a_{33} = 1 \end{cases} \quad \dots(4.1)$$

次に、3等分された各系列の中で各ラベルのしめる割合を調べることで出力確率  $b_{ij,k}$  を求め、/b/,/d/,/g/別に相加平均をとる。こうして得られた  $a_{ij}, b_{ij,k}$  を、HMMの学習する前の初期モデルのパラメータとする。

(4) (3)で作られた/b/,/d/,/g/別の初期モデルのうち、/b/の初期モデルには音素/b/のトレーニングデータを入力してHMMを学習させ、/b/のHMMを作成する。同様に、/d/,/g/のHMMを作成する。このとき、各トレーニングデータの平均生起確率の収束の様子について調べる。

(5) (4)でできた学習済みの各HMMの評価を行うため、トレリスのアルゴリズムによって各トレーニングデータを認識し、/b/,/d/,/g/それぞれの認識率を求める。

(6) また、初期HMMのうち遷移確率を  $a_{ii} = 0.75$ ,  $a_{i,i+1} = 0.25$  ( $1 \leq i \leq 2$ )とし、出力確率  $b_{ij,k}$  を0~1までの乱数(但し  $\neq 0$ )として(4),(5)を行う。

## 4.2 実験結果

216単語から/b/,/d/,/g/の区間を切り出した結果、得られたサンプル数を表4.1に示す。

求まった64個のコードブックベクトルの距離マトリックスを、表4.2に示す。この表の縦軸,横軸は、ともにコードブックベクトル番号を表し、各ベクトル同志の距離(PWLR)を求め、距離の小さい順に0~9のシンボルで表示してある。なお、各コードブックベクトルは、距離の近いものなるべく隣り合うように並べてある。

表 4.1 有声破裂音のサンプル数

子音カテゴリー	/b/	/d/	/g/	計
サンプル数(個)	27	17	25	69

また、このコードブックベクトルを使用して作ったHMM用トレーニングデータの一部を、図4.1に示す。コードブックベクトル番号の系列に変換された各データを、時間軸に沿って並べてある。

次に、4.1節(3),(6)でそれぞれ作った初期モデルを、500回まで学習したときの各トレーニングデータの平均生起確率の変化を図4.2に示す。縦軸は平均生起確率、横軸は学習回数を表し、(b,d,g)はトレーニングデータから求めた初期モデルを学習したもの、(b\*,d\*,g\*)は乱数から求めた初期モデルを学習した場合のグラフである。

そして、学習済みの各HMMを使用したトレリスのアルゴリズムによる認識結果を、表4.3に示す。表4.3中の“○”は正解を表し、最初の行は学習回数を、最後の行は各モデルの認識率を示す。そして、“500\*”は、乱数の初期モデルを500回学習したモデルでの結果で、その他はトレーニングデータから推定したモデルでの結果である。これらの認識結果から求めた各HMMの平均子音認識率を、表4.4に示す。

表4.2 コードブックベクトルの距離マトリックス

=== DISTANCE MATRIX ===

	0000000000111111111122222222223333333333444444444455555555556666
	0123456789012345678901234567890123456789012345678901234567890123
00	0011126235242100643355446575663076111122456574555452654274532310
01	0001125134233210434243344464452054100022233253434341544274554610
02	1001214266566521244343226565775444211244554376536363763275899972
03	111021434444431111213323767676555511133444365545463653275799982
04	1122000024233300666565428787776455101112442165334353655375788962
05	2211001046564411555556328888887366101243776487657575875486764542
06	6544010046567732789888649999998988423446775388667587888698999998
07	2123000034345411677676537786775666200123442165434364656486899983
08	3364244300101221788788678787887988554433456587776765888697999996
09	546446640010243278989878888788798865545456687786765888698999998
10	2254255311001011777676767787655776443100123252341311335252899993
11	4364366400002221888787877787666977554312234363452522557374999996
12	236434751212002187788778898987499554432678797887755877585765643
13	1253347424020011876676768797876388343210568684664522555262434421
14	0121013123122100555454437686664255111001333263332231533263444420
15	0011012112111100445454237676665355210111333264434342654274677841
16	6421657677888540031120498888889664547877788768786888588999999
17	434165878878775400001224989888766244466777787767686872486899985
18	3442659789787655300012459898877576255454777786656564651254323343
19	3231558678677644100011248787766554133254555565435463541254689973
20	5443658789788755111100039999978666344365777787647575650265556663
21	533356868867864222100138898646732233133433443213132320132589982
22	4322436567787742024201029898888955433266765577647585873487999996
23	4423224378677633445433209999878823324344665476536375652365799984
24	646788978877887799989899000114098887567855565677775778788999998
25	545678977877876688879889000025068877567866676677775888788999996
26	766788988888987899899900004729988867897778888888799989999999
27	5456789677778766888798891000361788764578566667777878588889899997
28	647779978866986698796881243000964664357111301222331337456999997
29	6576789788568766867674874576002842454235101200000110003212899994
30	3255678577567645887686880021020275552246223322344442557566789953
31	0045439699794323975567989697982099322243999996888662755364432201
32	7545568688779855667563528898647900434246322233113143423346999998
33	6445568688679855666462538888425900333235211121001032213234999996
34	1121114256455312422132438787645343001122333243213142431153445631
35	101100205645541154534332788655233000022332153213142533264555631
36	101113045344310445343345564442243100022221133213241533264688951
37	102112414513420144423123667532222100001110021001020212032357720
38	224314423401310176556364778753444322200011131110100112131368851
39	2243236335022011864453648898756365222100456472332211222030113210
40	425447744412653377574765675112332332114000100010210225244999995
41	535447745523763377573665676102921332115000000000110215345999996
42	634426526634883377573555676113921321016000001000121225356999998
43	523314315623762277574546786323921211014100011001021224255999996
44	7576688688569866888684775686002932453237000100000121116446999998
45	4365578577237434876573766687102631333112001100000000002111589972
46	545536647734863477646265778720381022201300000000000002112799993
47	533435637845863366534143778720481011101310000000010102123899993
48	546547746612742487657376778720483133310200010000000003111899994
49	4334355377357523765451537788314610112012211010000010001011489971
50	5466578666125234886673877787314643444201112220010100002000489972
51	213335745512521266435255557510223222100100111000000002010246620
52	657668868835855686656386789830574245521222210010000002000489983
53	54655785883575358754527578983055213331122122100000001000158841
54	4433558885753452110032889873753313322255462223122210011014331
55	222234646623522254222143778842533212201023324111100000010126630
56	7777789899578667885563868899516643566333445541121101001100148753
57	4455568678245234864452758898626464344210456561231100001000014320
58	558777989989744698365597999987499456331999995788442410110003121
59	35998699999634799285899999998399458561999998999884851241001010
60	249984999999544799396999999992995587839999999996984684310011
61	369995999996448993969999999929966578299999999996983673100011
62	117864989999422498476889899999509933525199999799972843352211100
63	002228368363101953332646697743186111010568682334120311030101100

b( 1) : 56 56 56 56 56 56 55 55 37 41 40  
 b( 2) : 14 38 41 40  
 b( 3) : 40 40 40 50 50 50 50 50 45 29 44  
 b( 4) : 45 45 45 45 45 41 41 41 41  
 b( 5) : 52 52 52 52 52 53 53 53 53 53 53 49 49 47 47  
 b( 6) : 52 52 52 52 52 52 52 52 52 29 29  
 b( 7) : 45 52 52 52 52 52 53 48 48 43 42  
 b( 8) : 58 58 58 58 58 56 56 56 56 56 56 53 49 43 43 43  
 b( 9) : 52 52 52 48 48 48  
 b(10) : 52 52 52 52 52 52 52 52 52 52 46 46 46 46

d( 1) : 36 36 42 42 42 42  
 d( 2) : 58 54 54 54 54 56 52 52 52 56 56 56 55 51 46 42 42 42  
 d( 3) : 56 56 63 1 36 36 36 36 36 47 47  
 d( 4) : 36 36 15 37 37 42  
 d( 5) : 52 56 56 55 36 36 36 36 43 42 42  
 d( 6) : 7 43 43 42  
 d( 7) : 36 7 7 7 7 49  
 d( 8) : 1 1 1 1 1 37 47 47  
 d( 9) : 52 52 52 52 52 52 52 52 52  
 d(10) : 46 46 46 46 46 46 46 46 46 46 46 46 46 46 46 46 47 47

g( 1) : 47 3 34 34 34 4 7 6 6 4  
 g( 2) : 47 47 42 42 42 42 42 42  
 g( 3) : 40 40 40 40 40 40 40 40  
 g( 4) : 5 5 5 5 5 5 7 7 7 7 7 42  
 g( 5) : 6 6 4 4 4 4 4 43 46 44 42  
 g( 6) : 45 45 52 52 52 52 45 45 45 41 41 41 41  
 g( 7) : 8 8 8 8 8 8 8 10  
 g( 8) : 33 33 33 33 32 32 32 32 32 32 32 32 32  
 g( 9) : 22 22 22 22 22 22 21 46 46  
 g(10) : 16 16 16 16 16 20 21

図4.1 HMM用トレーニングデータ

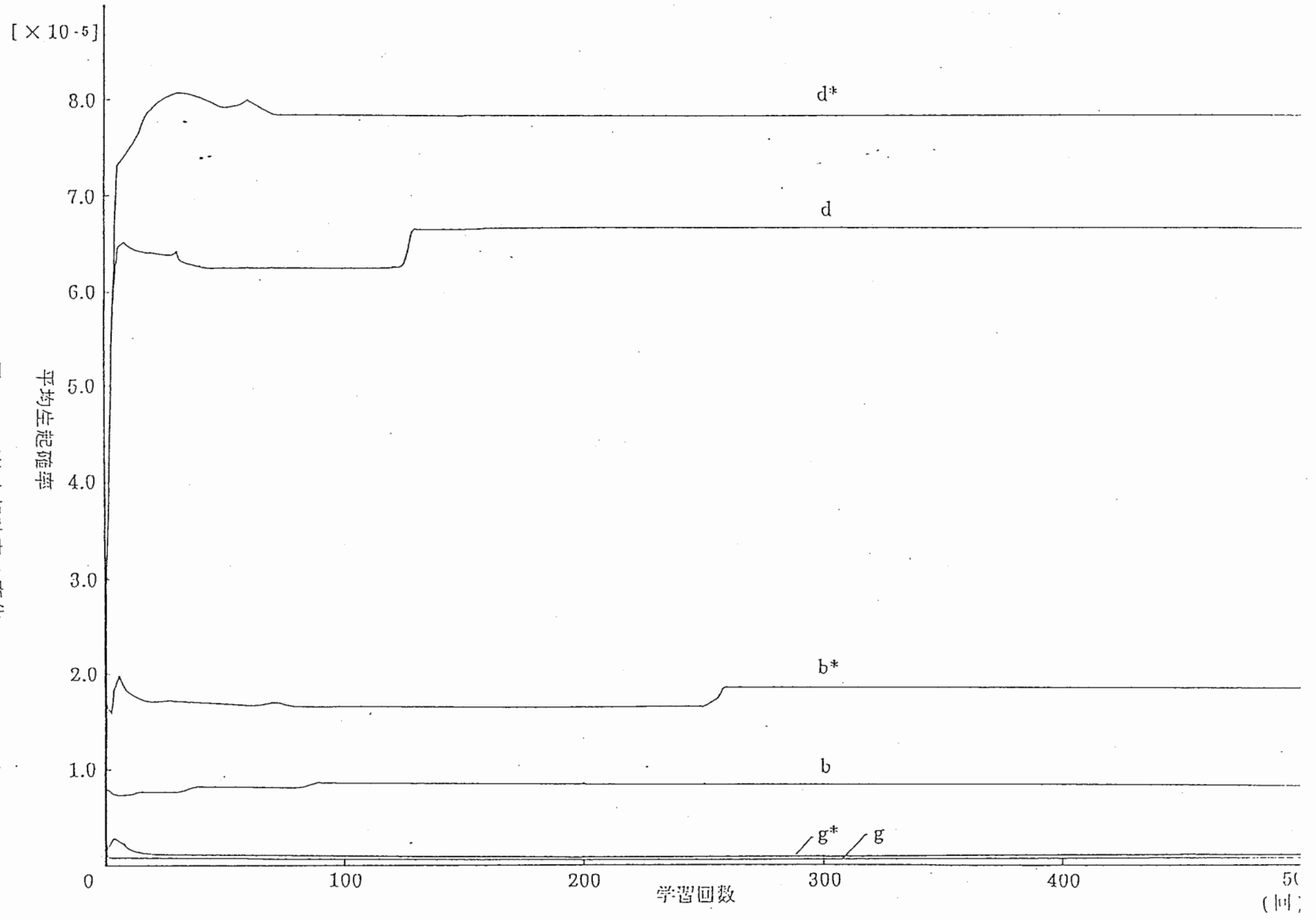


图 4.2 平均生起確率の変化  
(13)

表 4.3 トレーニングデータ認識結果

入力	0	100	300	500	500*	入力	0	100	300	500	500*	入力	0	100	300	500	500*
b1	○	○	○	○	○	d1	○	○	○	○	○	g1	○	○	○	○	○
b2	○	○	○	○	○	d2	○	○	○	○	○	g2	d	d	d	d	d
b3	g	g	g	g	g	d3	○	○	○	○	○	g3	○	○	○	○	○
b4	○	○	○	○	○	d4	○	○	○	○	○	g4	○	○	○	○	○
b5	○	○	○	○	○	d5	○	○	○	○	○	g5	○	○	○	○	○
b6	○	○	○	○	○	d6	○	○	○	○	○	g6	b	b	b	b	b
b7	○	○	○	○	○	d7	○	○	○	○	○	g7	○	○	○	○	○
b8	○	○	○	○	○	d8	○	○	○	○	○	g8	○	○	○	○	○
b9	○	○	○	○	○	d9	b	b	b	○	b	g9	○	○	○	○	○
b10	○	○	○	○	○	d10	○	○	○	○	○	g10	○	○	○	○	○
b11	○	○	○	○	○	d11	○	○	○	○	○	g11	○	○	○	○	○
b12	○	○	○	○	○	d12	○	○	○	○	○	g12	○	○	○	○	○
b13	○	○	○	○	○	d13	○	○	○	○	○	g13	○	○	○	○	○
b14	○	○	○	○	○	d14	○	○	○	○	○	g14	○	○	○	○	○
b15	○	○	○	○	○	d15	○	○	○	○	○	g15	○	○	○	○	○
b16	○	○	○	○	○	d16	○	○	○	○	○	g16	b	○	○	○	○
b17	○	○	○	○	○	d17	○	○	○	○	○	g17	○	○	○	○	○
b18	○	○	○	○	○							g18	○	○	○	○	○
b19	○	○	○	○	○							g19	○	○	○	○	○
b20	○	○	○	○	○							g20	○	○	○	○	○
b21	○	○	○	○	○							g21	○	○	○	○	○
b22	○	○	○	○	○							g22	b	b	b	b	○
b23	d	d	d	d	d							g23	○	○	○	○	○
b24	○	○	○	○	○							g24	○	○	○	○	○
b25	○	○	○	○	○							g25	○	○	○	○	○
b26	○	○	○	○	○												
b27	○	○	○	○	○												
(%)	92.6	92.6	92.6	92.6	92.6	(%)	94.1	94.1	94.1	100.	94.1	(%)	84.0	88.0	88.0	88.0	92.0

\* は乱数から求めた初期モデルを学習したもの

表 4.4 平均子音認識率

学習回数(回)	0	100	300	500	500*
平均子音認識率(%)	90.2	91.6	91.6	93.5	92.9

\* は乱数から求めた初期モデルを学習したもの

## 5 考察

表 4.2 より、対角線上に距離の小さいシンボルがかなり集中しており、コードブックベクトルが距離の近いもの同志が隣接していることがわかる。このような順に並べてある理由は、図 4.1 に示す各 HMM トレーニングデータのラベルの時間変化を見やすくするために、ラベルの変化は音声の特徴パラメータの変化を示している。

次に、図 4.2 の平均生起確率の変化のグラフを見ると、/d/, /b/, /g/ の順に確率が高く、乱数を初期モデルとしたものは、トレーニングデータから求めた初期モデルを使用したものよりいずれも確率が高い。この理由としては、乱数からスタートするモデルの先入観のなさがあげられよう。しかし、乱数の初期モデルは、学習開始直後に確率が大きく変動しており、この変動部分の確率の最高値は収束値よりも確率が大きく、乱数の初期モデルの不安定さが感じられる。トレーニングデータから求めた初期モデルを使用した場合、このような変動はあまりなく、各収束値は各グラフの最高値となり、バーム・ウェルチのアルゴリズムが正確に働いたことを示している。

そして、もう 1 つ問題なのは、d のグラフの学習回数 130 回付近、b\* のグラフの 250 回付近の急な確率の上昇で、1 度収束したかのように見えるこれらのグラフは、実はまだ収束していなかったことがわかる。したがって、全グラフ 500 回も学習すれば完全に収束すると考えるのは危険であり、500 回以後このような変化が出ないという保証はない。このたまに起きる急変動を無視するならば、全グラフ 100 回付近で既に部分的に収束しており、トレーニングデータから求めた初期モデルを使用した方が、学習初期のあばれが無いだけ収束するのが速い。



この2つの問題を解析するためには、トレーニング時における遷移確率  $a_{ij}$ 、出力確率  $b_{ij,k}$  のトレースが有効な手段と思われる。しかし、時間的制約により今回は行うことができなかった。

そして、表4.3, 及び表4.4の認識結果を見ると、トレーニングデータ、乱数から求めた各モデルは、500回も学習すればほぼ同等の認識率が得られ、しかも90%を越える子音認識率であることがわかる。トレーニングデータ以外のデータを認識した場合、認識率が低下する可能性があるが、今回はデータ量不足のためにこの実験は行えなかった。しかし、正確な評価を行うならば、トレーニングデータと認識データとは異なったデータを用意し、オープンな認識実験が望まれる。また、/d/, /b/, /g/の順に認識率がよく、図4.2のトレーニング中の平均生起確率の大小関係と一致している。これより、トレーニング中の各トレーニングデータの生起確率の大小がそのモデルの優劣を示していると思われる。

更に認識率の向上をめざすために考えられることは、

- (1) トレーニングデータの増加
- (2) コードブックベクトル数の増加
- (3) フォワード・バックワード, バーム・ウェルチ, トレリスの各アルゴリズムにおいて確率の桁落ちを考慮する。
- (4) HMMの学習回数の増加

等が考えられ、今回は、1人の話者、3カテゴリーの子音について認識実験を行ったが、他の話者、及び他の子音、母音において、どの程度の認識率が得られるかが興味のあるところである。

## 6 結論

ベクトル量子化、及びHMMという2つの手法を応用して子音認識を行ったわけだが、HMMの初期モデルは、乱数だと確率の収束値が大きい反面、収束速度が遅い。トレーニングデータから推定した初期モデルは、収束値は小さいものの収束速度が速く安定性がある。そして認識率は、両モデルともほぼ同程度の結果が得られた。しかし現時点では、どちらの初期モデル使用した方がよいのか結論を述べることはできず、5章の最後の段落でのべたような更に多くの実験が必要と

される。

そして、トレーニング時においては、時々生起確率の急激な上昇が起こり、乱数から求めた初期モデルを使用した場合には、学習当初に生起確率のあばれが生じる。この原因も、5章で述べたように更に実験を重ねないと判明しない。

このように、HMMの使用にはまだまだ解決しなければならない問題があり、多くのノウハウが必要とされるようである。

#### 参考文献

S. E. Levinson, L. R. Rabiner, and M. M. Sondhi, " An Introduction to the Application of the Theory of Probabilistic Functions of a Markov Process to Automatic Speech Recognition, " The Bell System Technical Journal, Vol. 62, No. 4 (April 1983).

## 付録

500回学習後の各HMMのパラメータを記載する。

b(500) : /b/ の HMM (トレーニングデータから求めた初期モデルを学習)

b\*(500) : /b/ の HMM (乱数から求めた初期モデルを学習)

d(500) : /d/ の HMM (トレーニングデータから求めた初期モデルを学習)

d\*(500) : /d/ の HMM (乱数から求めた初期モデルを学習)

g(500) : /g/ の HMM (トレーニングデータから求めた初期モデルを学習)

g\*(500) : /g/ の HMM (乱数から求めた初期モデルを学習)

A<sub>11</sub>  
0.0312364370162775

b<sub>11k</sub>  
0.00000000000000 0.00000000000000 0.00000000000000 0.00000000000000  
0.0103358402177525 0.00000000000000 0.0384815384815385 0.00000000000000  
0.00000000000000 0.00000000000000 0.00000000000000 0.00000000000000  
0.00000000000000 0.00000000000000 0.0256410256410256 0.00000000000000  
0.00000000000000 0.00000000000000 0.00000000000000 0.00000000000000  
0.00000000000000 0.00000000000000 0.00000000000000 0.00000000000000  
0.00000000000000 0.00000000000000 0.00000000000000 0.00000000000000  
0.00000000000000 0.00000000000000 0.00000000000000 0.00000000000000  
0.00000000000000 0.00000000000000 0.00000000000000 0.00000000000000  
0.00000000000000 0.00000000000000 0.00000000000000 0.00000000000000  
0.00000000000000 0.00000000000000 0.00000000000000 0.00000000000000  
0.0384815384815385 0.00000000000000 0.00000000000000 0.0349461110642958  
0.00000000000000 0.0386061561863286 0.0437904536020388 0.00000000000000  
0.0079786561315533 0.00000000000000 0.00000000000000 0.00000000000000  
0.5581850987268723 0.00000000000000 0.00000000000000 0.00000000000000  
0.0474121575412300 0.00000000000000 0.060436733222334 0.0128205128205128  
0.00000000000000 0.00000000000000 0.0384815384815385 0.0384815384815385

A<sub>12</sub>  
0.0687615629837225

b<sub>12k</sub>  
0.00000000000000 0.00000000000000 0.00000000000000 0.00000000000000  
0.0691658546472836 0.00000000000000 0.0370370370370370 0.00000000000000  
0.00000000000000 0.00000000000000 0.00000000000000 0.00000000000000  
0.00000000000000 0.00000000000000 0.00000000000000 0.0407628025239845  
0.00000000000000 0.00000000000000 0.00000000000000 0.00000000000000  
0.00000000000000 0.00000000000000 0.00000000000000 0.00000000000000  
0.00000000000000 0.00000000000000 0.00000000000000 0.00000000000000  
0.00000000000000 0.00000000000000 0.00000000000000 0.00000000000000  
0.00000000000000 0.00000000000000 0.00000000000000 0.00000000000000  
0.00000000000000 0.00000000000000 0.00000000000000 0.00000000000000  
0.00000000000000 0.00000000000000 0.00000000000000 0.00000000000000  
0.00000000000000 0.00000000000000 0.0382184905710001 0.00000000000000  
0.00000000000000 0.0418160855287493 0.0408495527213920 0.00000000000000  
0.00000000000000 0.00000000000000 0.0370370370370370 0.0370370370370370  
0.5266754337349633 0.00000000000000 0.00000000000000 0.00000000000000  
0.0541622317185931 0.00000000000000 0.00000000000000 0.0370370370370370  
0.00000000000000 0.00000000000000 0.00000000000000 0.00000000000000

A<sub>22</sub>  
0.0791966883555384

b<sub>22k</sub>  
0.00000000000000 0.00000000000000 0.00000000000000 0.00000000000000  
0.00000000000000 0.00000000000000 0.04000000000000 0.00000000000000  
0.00000000000000 0.00000000000000 0.00000000000000 0.00000000000000  
0.00000000000000 0.00000000000000 0.04000000000000 0.08000000000000  
0.00000000000000 0.00000000000000 0.00000000000000 0.00000000000000  
0.00000000000000 0.00000000000000 0.00000000000000 0.00000000000000  
0.00000000000000 0.00000000000000 0.00000000000000 0.00000000000000  
0.00000000000000 0.00000000000000 0.00000000000000 0.00000000000000  
0.00000000000000 0.00000000000000 0.00000000000000 0.00000000000000  
0.00000000000000 0.00000000000000 0.00000000000000 0.00000000000000  
0.00000000000000 0.00000000000000 0.00000000000000 0.00000000000000  
0.00000000000000 0.00000000000000 0.00000000000000 0.00000000000000  
0.00000000000000 0.0565103776574126 0.045230233480390 0.00000000000000  
0.00000000000000 0.00000000000000 0.036435908673812 0.00000000000000  
0.4780845753494957 0.0351457073303988 0.00000000000000 0.00000000000000  
0.1475831447149207 0.00000000000000 0.00000000000000 0.00000000000000  
0.0240000000000000 0.0160000000000000 0.00000000000000 0.00000000000000

A<sub>23</sub>  
0.0208033016404616

b<sub>23k</sub>  
0.00000000000000 0.00000000000000 0.00000000000000 0.00000000000000  
0.0370370370370370 0.0370370370370370 0.00000000000000 0.00000000000000  
0.00000000000000 0.00000000000000 0.00000000000000 0.00000000000000  
0.00000000000000 0.00000000000000 0.00000000000000 0.00000000000000  
0.00000000000000 0.00000000000000 0.00000000000000 0.00000000000000  
0.00000000000000 0.00000000000000 0.00000000000000 0.00000000000000  
0.00000000000000 0.00000000000000 0.00000000000000 0.00000000000000  
0.00000000000000 0.1886767654401885 0.00000000000000 0.00000000000000  
0.00000000000000 0.00000000000000 0.00000000000000 0.00000000000000  
0.00000000000000 0.00000000000000 0.0401837877332693 0.00000000000000  
0.00000000000000 0.0488958653220072 0.00000000000000 0.00000000000000  
0.00000000000000 0.0749528591460584 0.1111111102583505 0.00000000000000  
0.1260434061970831 0.0455963899542548 0.00000000000000 0.00000000000000  
0.00000000743241927 0.1313169270770633 0.00000000000000 0.0370370370370370  
0.00000000000000 0.0370370370370370 0.00000000000000 0.00000000000000  
0.0370370370370370 0.00000000000000 0.00000000000000 0.00000000000000

A<sub>33</sub>  
1.0000000000000000

b<sub>33k</sub>  
0.0211640211640212 0.00000000000000 0.00000000000000 0.00000000000000  
0.00000000000000 0.00000000000000 0.00000000000000 0.00000000000000  
0.00000000000000 0.00000000000000 0.0092592532532533 0.00000000000000  
0.00000000000000 0.00000000000000 0.00000000000000 0.0251322751322751  
0.00000000000000 0.00000000000000 0.00000000000000 0.00000000000000  
0.00000000000000 0.00000000000000 0.00000000000000 0.0370370370370370  
0.00000000000000 0.00000000000000 0.00000000000000 0.00000000000000  
0.00000000000000 0.1590761865427122 0.00000000000000 0.00000000000000  
0.00000000000000 0.0370370313471736 0.00000000000000 0.00000000000000  
0.00000000000000 0.0092592592592593 0.0551876206074755 0.00000000000000  
0.0568783068783069 0.0598372393239432 0.0095123726435611 0.0381165207088707  
0.1031732352952783 0.00000000000000 0.0846248566831236 0.0190894320435398  
0.1923565446306317 0.0634375310022617 0.00000000000000 0.0052910052910053  
0.00000000000000 0.00000000000000 0.00000000000000 0.0145502645502646  
0.00000000000000 0.00000000000000 0.00000000000000 0.00000000000000  
0.00000000000000 0.00000000000000 0.00000000000000 0.00000000000000

b(500)

a<sub>11</sub>  
0.5533888571119561

b<sub>11</sub>  
0.000000000000000 0.000000000000000 0.000000000000000 0.000000000000000  
0.000000000000000 0.000000000000000 0.041666666666667 0.000000000000000  
0.000000000000000 0.000000000000000 0.000000000000000 0.000000000000000  
0.000000000000000 0.000000000000000 0.000000000000000 0.000000000000000  
0.000000000000000 0.000000000000000 0.000000000000000 0.000000000000000  
0.000000000000000 0.000000000000000 0.000000000000000 0.000000000000000  
0.000000000000000 0.000000000000000 0.000000000000000 0.000000000000000  
0.000000000000000 0.000000000000000 0.000000000000000 0.000000000000000  
0.000000000000000 0.000000000000000 0.000000000000000 0.000000000000000  
0.000000000000000 0.000000000000000 0.000000000000000 0.000000000000000  
0.000000000000000 0.000000000000000 0.000000000000000 0.000000000000000  
0.041666666666667 0.000000000000000 0.000000000000000 0.041666666666667  
0.000000000000000 0.041666666666667 0.047713847908802 0.000000000000000  
0.008434595855264 0.000000000000000 0.000000000000000 0.000000000000000  
0.0039960912780741 0.000000000000000 0.000000000000000 0.000000000000000  
0.0526358428358644 0.000000000000000 0.0628517740517525 0.0160256416256410  
0.000000000000000 0.000000000000000 0.041666666666667 0.000000000000000

a<sub>12</sub>  
0.4166011428880439

b<sub>12k</sub>  
0.000000000000000 0.000000000000000 0.000000000000000 0.000000000000000  
0.0740740740740741 0.000000000000000 0.0370370370370370 0.000000000000000  
0.000000000000000 0.000000000000000 0.000000000000000 0.000000000000000  
0.000000000000000 0.000000000000000 0.000000000000000 0.000000000000000  
0.000000000000000 0.000000000000000 0.000000000000000 0.000000000000000  
0.000000000000000 0.000000000000000 0.000000000000000 0.000000000000000  
0.000000000000000 0.000000000000000 0.000000000000000 0.000000000000000  
0.000000000000000 0.000000000000000 0.000000000000000 0.000000000000000  
0.000000000000000 0.000000000000000 0.000000000000000 0.000000000000000  
0.000000000000000 0.000000000000000 0.000000000000000 0.000000000000000  
0.000000000000000 0.000000000000000 0.0370370370370370 0.000000000000000  
0.0370370164630140 0.000000000000000 0.000000000000000 0.000000000000000  
0.000000000000000 0.0424587723054540 0.0398716860736610 0.000000000000000  
0.000000000000000 0.000000000000000 0.000000205740231 0.000000000000000  
0.5236480401661129 0.000000000000000 0.000000000000000 0.000000000000000  
0.0976251851584758 0.000000000000000 0.000000000000000 0.000000000000000  
0.000000000000000 0.0370370370370370 0.0370370370370370 0.0370370370370370

a<sub>22</sub>  
0.6427559450430693

b<sub>22k</sub>  
0.000000000000000 0.000000000000000 0.000000000000000 0.000000000000000  
0.000000000000000 0.000000000000000 0.0384615384615385 0.000000000000000  
0.0054345054345055 0.000000000000000 0.000000000000000 0.000000000000000  
0.000000000000000 0.000000000000000 0.0384615384615385 0.0714285714285714  
0.000000000000000 0.000000000000000 0.000000000000000 0.000000000000000  
0.000000000000000 0.000000000000000 0.000000000000000 0.000000000000000  
0.000000000000000 0.000000000000000 0.000000000000000 0.000000000000000  
0.000000000000000 0.000000000000000 0.000000000000000 0.000000000000000  
0.000000000000000 0.000000000000000 0.000000000000000 0.000000000000000  
0.000000000000000 0.000000000000000 0.000000000000000 0.000000000000000  
0.000000000000000 0.000000000000000 0.000000000000000 0.000000000000000  
0.000000000000000 0.000000000000000 0.000000000000000 0.000000000000000  
0.000000000000000 0.0434052752815056 0.0411222405847864 0.000000000000000  
0.000000000000000 0.000000000000000 0.0382267057053477 0.0384615384615385  
0.7844710839416647 0.000000000000000 0.000000000000000 0.000000000000000  
0.1460034637174648 0.000000000000000 0.000000000000000 0.000000000000000  
0.03076923076923084 0.0076923076923084 0.000000000000000 0.000000000000000

a<sub>23</sub>  
0.3572440549569307

b<sub>23k</sub>  
0.0370370370370207 0.000000000000000 0.000000000000000 0.000000000000000  
0.0370370370370370 0.0370370370370370 0.000000000000000 0.000000000000000  
0.000000000000000 0.000000000000000 0.0370370370370370 0.000000000000000  
0.000000000000000 0.000000000000000 0.0370370370370370 0.000000000000000  
0.000000000000000 0.000000000000000 0.000000000000000 0.000000000000000  
0.000000000000000 0.000000000000000 0.000000000000000 0.000000000000000  
0.000000000000000 0.000000000000000 0.000000000000000 0.000000000000000  
0.000000000000000 0.000000000000000 0.000000000000000 0.000000000000000  
0.000000000000000 0.000000000000000 0.000000000000000 0.000000000000000  
0.000000000000000 0.000000000000000 0.000000000000000 0.000000000000000  
0.000000000000000 0.0184060708455067 0.000000000000000 0.000000000000000  
0.000000000000000 0.000000000000000 0.000000000000000 0.000000000000000  
0.000000000000000 0.0452987868419906 0.000000000000000 0.000000000000000  
0.000000000000000 0.1017204870690754 0.0788431870358258 0.000000000000000  
0.000000000000000 0.000000000000000 0.000000000000000 0.034548601207980  
0.3276345933212895 0.1342868154062548 0.000000000000000 0.0370370370370370  
0.000000000000000 0.0370370370370370 0.000000000000000 0.000000000000000  
0.000000000000000 0.000000000000000 0.000000000000000 0.000000000000000

a<sub>33</sub>  
1.0000000000000000

b<sub>33k</sub>  
0.0185185185185188 0.000000000000000 0.000000000000000 0.000000000000000  
0.000000000000000 0.000000000000000 0.000000000000000 0.000000000000000  
0.000000000000000 0.000000000000000 0.000000000000000 0.000000000000000  
0.000000000000000 0.000000000000000 0.000000000000000 0.158730158730159  
0.000000000000000 0.000000000000000 0.000000000000000 0.000000000000000  
0.000000000000000 0.000000000000000 0.000000000000000 0.0370370370370370  
0.000000000000000 0.000000000000000 0.000000000000000 0.000000000000000  
0.000000000000000 0.1829433673123585 0.000000000000000 0.000000000000000  
0.000000000000000 0.0251564692043399 0.000000000000000 0.000000000000000  
0.000000000000000 0.0082592592592593 0.0643704240537180 0.000000000000000  
0.0586419753086411 0.0586419753086420 0.0089970728642579 0.0367748506420356  
0.0965181148335271 0.000000000000000 0.0904437770448363 0.0081308228954392  
0.1806301782478422 0.0544272191917355 0.000000000000000 0.0061728395061724  
0.000000000000000 0.0227127183463579 0.000000000000000 0.0145502645502646  
0.000000000000000 0.000000000000000 0.000000000000000 0.000000000000000  
0.000000000000000 0.000000000000000 0.000000000000000 0.000000000000000

b\*(500)





a11  
0.6384886625920704

b11k  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.012074453390955  
0.0000000000000000 0.0416666666666667 0.0416666666666667 0.0000000000000000  
0.0416666666666667 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0000000000000000 0.0058523809523810 0.0000000000000000 0.0000000000000000  
0.0416666666666667 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0416666666666667 0.0000000000000000  
0.0416666666666667 0.0000000000000000 0.0000000000000000 0.0416666666666667  
0.0000000000000000 0.0416666666666667 0.0285426213275712 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0416666666666667 0.0418638480337588 0.0000000000000000 0.0000000000000000  
0.0416666666666667 0.0137491956480688 0.0000000000000000 0.0416666666666667  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.1827459278359208 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0357142857142857 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000

a12  
0.3615013374078296

b12k  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0632798903021550 0.0400000000000000 0.0000000000000000 0.0000000000000000  
0.0800000000000000 0.0000000000000000 0.001093806635758 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0400000000000000 0.0400000000000000 0.0000000000000000 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0400000000000000 0.0000000000000000  
0.0382846347854439 0.0000000000000000 0.0400000000000000 0.0000000000000000  
0.041706882302151 0.039399600827899 0.0400000000000000 0.0000000000000000  
0.0400000000000000 0.0000000000000000 0.0167200030978450 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0400000000000000 0.0427428399051318 0.0400000000000000 0.0000000000000000  
0.0400860332098160 0.037179389566033 0.0000000000000000 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.1189061939364241 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.1200000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000

a22  
0.5407932945802512

b22k  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0704514108380184 0.0000000000000000 0.0000000000000000 0.0400000000000000  
0.0400000000000000 0.0404417010097959 0.0177892265815293 0.0118656931834455  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0400000000000000 0.0000000000000000 0.0133333333333333 0.0466666666666667  
0.0000000000000000 0.0000000000000000 0.0400000000000000 0.0000000000000000  
0.0362214816600556 0.0000000000000000 0.0400000000000000 0.0000000000000000  
0.0525303342139620 0.038993527043149 0.0312159205426333 0.0000000000000000  
0.0400000000000000 0.0000000000000000 0.0095485891619816 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0400000000000000 0.0585487207208306 0.0400000000000000 0.0000000000000000  
0.0405881894427422 0.0208341750179118 0.0000000000000000 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0755790420305722 0.0000000000000000  
0.0142249285533798 0.0000000000000000 0.0633333333333333 0.0000000000000000  
0.0366666666666667 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000

a33  
0.4592067054087487

b33k  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0784021210341353  
0.0153088273504718 0.0408773186746338 0.0575359298601362 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0400000000000000 0.0800000000000000 0.0000000000000000 0.0000000000000000  
0.0000000000000000 0.1200000000000000 0.0000000000000000 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0400000000000000 0.0000000000000000  
0.1117029581686247 0.0462154455056236 0.0000000000000000 0.0000000000000000  
0.0400000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0400000000000000 0.0778012305864279 0.0405978789656647 0.0400000000000000  
0.0442803657393238 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0462779258147582 0.0000000000000000  
0.0400000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000

a33  
1.0000000000000000

b33k  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0138888888888889 0.0000000000000000 0.0277777777777778 0.0302132223785045  
0.0233346750301031 0.0628884352785064 0.0240413783316577 0.0117149483601000  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0683347491993747 0.0138888888888889 0.0000000000000000 0.0244708984708995  
0.0334582867759740 0.0265136423315296 0.0000000000000000 0.0000000000000000  
0.0000000000000000 0.0277777777777778 0.0000000000000000 0.0000000000000000  
0.0272796540632955 0.0882071327674564 0.0000000000000000 0.0000000000000000  
0.0416666666666667 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0416666666666667 0.0471679270016841 0.067008398437177 0.0000000000000000  
0.1567897306118185 0.0000000000000000 0.0555555555555556 0.0000000000000000  
0.0863436963329561 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000

g(500)



a<sub>11</sub>  
0.5782853321304583

b<sub>11k</sub>  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0330146545167167  
0.0000000000000000 0.0434782608695652 0.0434782608695652 0.0000000000000000  
0.0434782608695652 0.0000000000000000 0.0173913043478261 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0434782608695652 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0434782608695652 0.0000000000000000  
0.0434782608695652 0.0000000000000000 0.0090000000000000 0.0434782608695652  
0.0434773769359463 0.0434782608695652 0.0000000000000000 0.0434782608695652  
0.0000000000000000 0.0434782608695652 0.0104636063528485 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0434782608695652 0.0436406561143332 0.0000000000000000 0.0000000000000000  
0.0435084153681058 0.0139338593339347 0.0000000000000000 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0708602780635037 0.0000000000000000  
0.1692794344230195 0.0000000000000000 0.0055448968168917 0.0000000000000000  
0.0271463839914293 0.0000000000000000 0.0434782608695652 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000

a<sub>12</sub>  
0.4207146678655417

b<sub>12k</sub>  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0000000000000000 0.0400000000000000 0.0400000000000000 0.0000000000000000  
0.0400000000000000 0.0000000000000000 0.0000000000000000 0.0400000000000000  
0.0000000000000000 0.0400000000000000 0.0000000000000000 0.0000000000000000  
0.0400000000000000 0.0416250262443954 0.0000000000000000 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0400000000000000 0.0000000000000000  
0.0364106434835387 0.0000000000000000 0.0400000000000000 0.0000000000000000  
0.0415769546848881 0.0399999665712271 0.0400000000000000 0.0000000000000000  
0.0000000000000000 0.0400000000000000 0.0400000000000000 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0400000000000000 0.0422295701519213 0.0000000000000000 0.0000000000000000  
0.0400830657132989 0.0376933993951260 0.0000000000000000 0.0400000000000000  
0.0000000000000000 0.0000000000000000 0.0800000000000000 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.1183749737556046 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000

a<sub>22</sub>  
0.5467891755258621

b<sub>22k</sub>  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0400000000000000 0.0400000000000000 0.0000000000000000 0.0000000000000000  
0.0426571428571429 0.0771428571428571 0.0000000000000000 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0745011686093052 0.0282378986163484 0.0033795816625136 0.0457757235953615  
0.0330842663497901 0.0030173411666802 0.0400000000000000 0.0000000000000000  
0.0360532755916356 0.0000000000000000 0.0400000000000000 0.0000000000000000  
0.0547642136006846 0.0399999665712271 0.0291178709084280 0.0000000000000000  
0.0000000000000000 0.0400000000000000 0.0400000000000000 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0400000000000000 0.0585299360845724 0.0400000000000000 0.0000000000000000  
0.0406394622011107 0.0208957126821501 0.0000000000000000 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0800000000000000 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.6000000000000000 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000

a<sub>23</sub>  
0.4512106244741379

b<sub>23k</sub>  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0600000000000000 0.0400000000000000 0.0000000000000000 0.0000000000000000  
0.0336865119027482 0.0800000000000000 0.0189873484901237 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0549654762163478 0.0442080726143867 0.0000000000000000 0.0292249161639062  
0.0000000000000000 0.1116005340053573 0.0000000000000000 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0400000000000000 0.0000000000000000  
0.1091615444269270 0.0466471065673073 0.0000000000000000 0.0000000000000000  
0.0000000000000000 0.0400000000000000 0.0000000000000000 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0400000000000000 0.0776481507851331 0.0400000000000000 0.0000000000000000  
0.0465421962006325 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0671321366071281 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000

a<sub>33</sub>  
1.0000000000000000

b<sub>33k</sub>  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0241537241832013 0.0000000000000000 0.0200000000000000 0.0433333333333333  
0.0212122759455909 0.0000000000000000 0.0359104532828184 0.060166952205549  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0518507385778959 0.0000000000000000 0.0000000000000000 0.0474995328939313  
0.0335648709621343 0.0570847571660385 0.0000000000000000 0.0000000000000000  
0.0000000000000000 0.0266666666666667 0.0000000000000000 0.0000000000000000  
0.0143007271505624 0.0646632066896281 0.0000000000000000 0.0000000000000000  
0.0400000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0400000000000000 0.0451841387281971 0.0531262356208663 0.0064615668541997  
0.1556466267181454 0.0000000000000000 0.0464615668541997 0.0000000000000000  
0.0667891586639562 0.0000000000000000 0.0049665823748737 0.0000000000000000  
0.0090325775018048 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000  
0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000

g\* (500)