

Transition Matrixによる学習式アルゴリズム

コース 基盤科学コース
学籍番号 050471
氏名 中野 賢
指導教員 ルジェロ・ミケレット

Development of modern computer technology is just unimaginable. The human brain is a great computer. Refer to this mechanism, Our programs are made. This is "Transition Matrix" the collaboration mechanism of the brain and matrix operations. Words, action, music, wave and communication. The output from Transition Matrix system have unlimited potential. I will introduce how to solve mazes and print out the letter.

現代の技術の進歩は素晴らしく、我々の想像の領域を超えることもあるだろう。特にコンピュータ技術の発展により、これからはロボットやコンピュータが人間と直接会話をするともあると期待されている。現在では困難な言葉や感情によるコミュニケーションが発達すれば、より多くの可能性が広がっていくのではないかと考えられる。ロボットやコンピュータが感情を持つこともあるであろう。今後は様々な現場において、より人間に近いコンピュータも現れるだろう。

そこで我々が考えたのは、人間のように自らが学び次なる動作を考えるロボットだ。そこに行き着くためには人間の脳を理解していく必要もあるだろう。また人間の脳の仕組みの中に重要なヒントが隠されているのではないかと考えた。人間自身は自ら学び、自ら行動していく生物である。

本研究で重要視している脳の仕組みは大きく2つある。一つは人間の脳はある規則に従っていること、もう1つはランダムであると言うこと。脳には必ず2つが同時に働いていることになる。例えば人間は歩く時も会話する時も何らかの規則に従っている。右足が前に出た後に右足がまた前に出ることや”おはようございます”に対して”こんばんは”と言う人は滅多にいない。しかしいつも決まった歩き方や会話パターンしかない人もほとんどいない。そんな人は「ロボットのような人」だと思うのが一般的であろう。そこから考えられるのは

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|----|---|---|---|---|----|---|----|---|---|----|
| 1 | 1 | 6 | 1 | 6 | 8 | 7 | 1 | 0 | 1 | 9 |
| 2 | 8 | 0 | 5 | 6 | 3 | 1 | 20 | 2 | 3 | 9 |
| 3 | 8 | 9 | 0 | 3 | 21 | 4 | 9 | 5 | 3 | 0 |
| 4 | 0 | 0 | 1 | 2 | 3 | 3 | 6 | 5 | 4 | 5 |
| 5 | 8 | 9 | 6 | 3 | 2 | 0 | 0 | 1 | 4 | 7 |
| 6 | 3 | 5 | 8 | 6 | 1 | 4 | 9 | 6 | 3 | 2 |
| 7 | 7 | 8 | 5 | 1 | 0 | 0 | 2 | 8 | 4 | 7 |
| 8 | 0 | 0 | 0 | 5 | 4 | 7 | 0 | 0 | 0 | 0 |
| 9 | 7 | 2 | 1 | 5 | 9 | 3 | 1 | 5 | 3 | 2 |
| 10 | 8 | 5 | 6 | 3 | 0 | 0 | 0 | 1 | 0 | 0 |

図1: Transition Matrix の例

人間はあらゆる現象を目の当たりにし、そこから様々なことを学んで成長する。幼いころか

らもそう学び成長してきた。言葉は発するときも文法を学んでから、話せるようになるわけではない。両親の言葉を聴いて徐々に話せるようになっていく。そこには人間が生まれながらに持っている、法則を見つけそれを出力する能力があると言っても良いだろう。そのような脳のシステムを言葉を発するロボットへの応用に活用できないかと考えた。

本研究においてはTransition Matrixを使って、日本語と英語の文章をコンピュータから高精度で出力していることにも成功している。

Transition Matrixを用いた脳シミュレーションは子供が言葉を覚えるように覚え、そして言葉を発するように出力するからだ。その手法について説明したい。また非常に似たシステムを使い、自ら迷路を解いていくロボットも考案した。そこで重要となっているのはコンピュータ自身が自ら学習をし、それをもとに行動し出力しているということだ。それにより、Transition Matrixを活用することによってより少ない情報データで新しい言葉や動作を出力することが出来る可能性を秘めています。

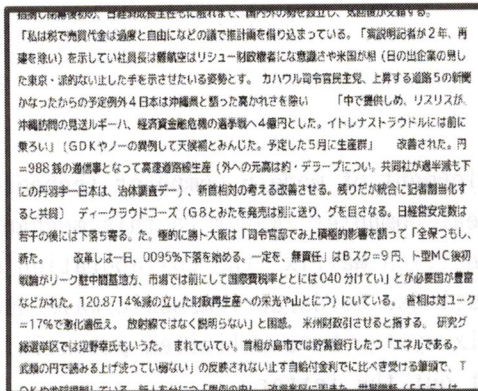


図2：日本語出力の例

Transition Matrixを活用した連続する情報を取り入れ、出力するシステムを使って言語を出力することに成功した。システム上には文法や単語の法則を意図的に入力することは一切していない。しかしまるで英語や日本語のように本物の言語であるかのように見える。まだまだ言語に近づいたとはいえ、ほど遠いものではあるが、完全にコンピュータから出力したものであると言えるだろう。

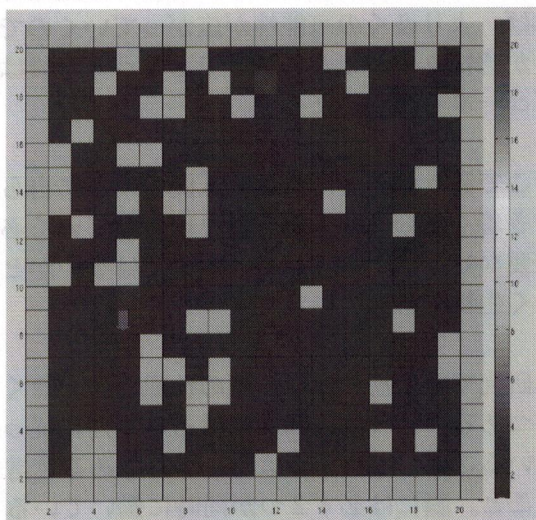


図3：ロボットシミュレーション

Transition Matrixを活用した学習式アルゴリズムシステムは応用の幅も広いであろう。私は迷路を解いていくロボットシミュレーションに応用した。その点についても紹介してある。より少ないデータで情報を処理し、動作していくことが出来る。Transition Matrixの幅広い応用性についても考えて頂きたい。

今後ともこのTransition Matrixを使った学習式アルゴリズムシステムが音楽や映像や人間の動きなどにも応用される日が来ることを期待できる、大きな可能性を秘めているのである。

<卒業論文>

Transition Matrix による学習式アルゴリズム

横浜市立大学 国際総合科学部
基盤科学コース ミケレット研究室
学籍番号 050471 中野 賢

第1章 緒言

第2章 脳の性質

- 2-1 脳の機能と容量
- 2-2 脳とコンピュータ
- 2-3 本研究で用いる前提条件

第3章 Transition Matrix とは

- 3-1 はじめに
- 3-2 Transition Matrix の基本的なデータの入力法
- 3-3 Transition Matrix から Monte-Carlo 法を使った出力

第4章 言語を使った Transition Matrix

- 4-1 英語アルファベット版
- 4-2 英語単語版
- 4-3 日本語版
- 4-4 データ格納システムの活用

第5章 Transition Matrix を応用したロボットシミュレーション

- 5-1 ロボットシミュレーションの概要
- 5-2 壁を見て迷路を解いていくシミュレーション
- 5-3 壁を見て迷路を解いていくシミュレーションの結果と考察

第6章 結語

参考文献

第1章 緒言

現代の技術の進歩は素晴らしく、我々の想像の領域を超えることもあるだろう。特にコンピュータ技術の発展により、これからはロボットやコンピュータが人間と直接会話をすることもあると期待されている。現在では困難な言葉や感情によるコミュニケーションが発達すれば、より多くの可能性が広がっていくのではないかと考えられる。ロボットやコンピュータが感情を持つこともあるであろう。今後は様々な現場において、より人間に近いコンピュータも現れるだろう。その中でもコンピュータが人間のような言葉を話すことは大きな課題となっている。その中ですべてのパターンをプログラム上のアルゴリズムに使う容量も年々増えていくことになる。近年ではコンピュータ技術の発展は目覚しく、コンピュータの性能もおおよそ30年で人間の脳を超えるのではないかと言われていることもある。

そしてコンピュータは命令にただ従うだけでは、動作を自分から開発していく必要である。人間の脳はコンピュータとしては優秀過ぎるのであるが、その偉大なコンピュータに電子的なコンピュータを超える日は遠くない。

本研究においては Transition Matrix を使って、日本語と英語の文章をコンピュータから高精度で出力していることにも成功している。Transition Matrix を用いた脳シミュレーションは子供が言葉を覚えるように覚え、そして言葉を発するように出力するからだ。その手法について説明したい。また非常に似たシステムを使い、自ら迷路を解いていくロボットも考案した。そこで重要となっているのはコンピュータ自身が自ら学習をし、それをもとに行動し出力しているということだ。それにより、Transition Matrix を活用することによってより少ない情報データで新しい言葉や動作を出力することが出来る可能性を秘めています。

第2章 人間の脳における性質

2-1 脳とコンピュータ

人間の脳は非常に高性能なコンピュータととして考えることが出来る。人間の脳を侮ってはいけない。

数学者の John von Neumann は人間の脳 2.8×10^{20} ビットのメモリーが保存出来るとしており、脳の作業速度は最高で 100,000 フロップある。1996 年で世界最新のコンピュータ CM-5 の 10^6 倍の計算速度を誇る。脳は非常に優秀なコンピュータである。

脳とコンピュータが良く比較されることがあるが、コンピュータと非常に似ている仕組みが良くある。例えば脳には短期記憶組織と長期記憶組織がある。これは要するにコンピュータで言うところメモリとハードディスクに相当する。そこでコンピュータをより人間の脳に近づけるために脳内のシナプスシミュレーションよりニューラルネットワーク技術が開発された。

2-2 言語とコンピュータ

コンピュータ上で言葉表現するの簡単なことではないとされてきている。なぜなら子どもがなぜ言葉を覚えるか。単語や文法を学んだ後に、会話が出来ようになるのではなくほとんどが聞いた音声から言語を学び習得するのです。人間の脳ではウェルニッケ野（感覚性言語野）を中心とした感覚性言語中枢で行われ、単語を構成する一連の音の記憶と照合されて、意味を判断します。視覚情報や音声情報として言語内に入力されていくのです。

言語を出力したり処理したりすることはデジタルの世界では大きな課題となっているの事実であろう。

2-3 本研究で用いる前提条件

人間の脳には大きく分けて2つの性質があると考えている。1つは基本的なルールを持っていること。例えば言葉には必ず文法がある。また人間は右足を前に出した後に左足を前に出す。もう1つは脳の出力はランダムであるということである。他にも多くの脳の性質はあるが、今回はこの2つの性質に着目する。

人間の行動は実はランダムである。

(例：電車にいつも同じ車両の同じドアから乗るわけではない、寝る時間は秒単位まで正確でない、毎日同じようにまったく同じ歩き方はしない)

しかし何かの規則に従っている

例1:言葉は文法に従っている。ただし、小さな子どもは文法を覚えてから言葉を話すのではなく、両親の言葉を聴いて言葉を覚える。そして発信する。言葉のルールがあるから発信出来るのである。

例2:食事など日常的な習慣においてを無意識の内に規則性を見出している。そして行動のたび習慣に規則性を通じて行動している。

ご飯→おかず→みそ汁→ご飯、言葉、コミュニケーション)

例えば日本人が日本語を話せるようになる過程として、文法を学んでから言葉を話す順序ではない。両親の会話は聴いて、徐々に言葉を覚え自らが会話出来るようになるのである。

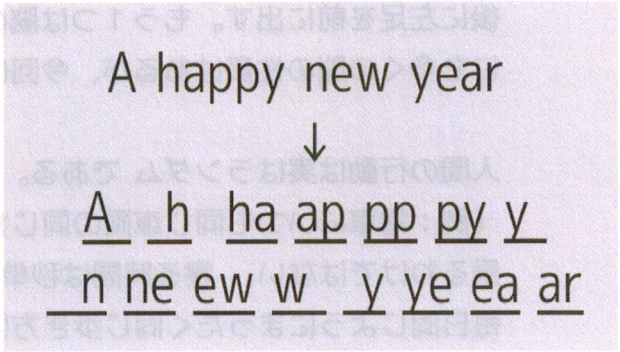
第3章 Transition Matrix とは

図 1 : 文字出力システム

3-1 はじめに

Transition Matrix (以降 TM) とはある連続した現象を Matrix(行列)上に取り込むことである。

TMは連続する現象を格納していく。例えば"123"と言う数字列があったときには「1の次には2が来た」ことを記録する。その次に「2の次には3が来た」ことを記録する。



詳しくは後節で解説するが、言葉の例にする。“happy”と言う単語があったとき『hの次にaが来る』ことを記録し、次は『aの次にはpが来ること』を記録する[図 1]。1つ連続するものがある事に値を 1 追加する。最終的に Matrix になった後に規格化を行う。規格化された行列を TM と呼ぶ。

私の本論文でのシステムの開発は Mathwork 社の MATLAB R2008 を利用し、MATLAB 言語を用いた。

MATLAB は行列演算を得意とし、比較的短時間で科学技術計算を出来る素晴らしいソフトだと私は思う。

主に行列演算の式を扱った、本ソフトは非常に有益なものになった。

前節で述べた『脳のシステムに非常に良く似ている』と考えることが出来るのではないか。人間の脳には規則性を持っている。言葉や歩く方向などその応用は様々である。また同様にランダムに出力されていく不規則性を持ち合わせている。

我々はそれらを踏まえて、脳内に TM が存在し、ランダム (Monte Calro 法) のようなもので出力されるという前提を持ち本研究を進めている。

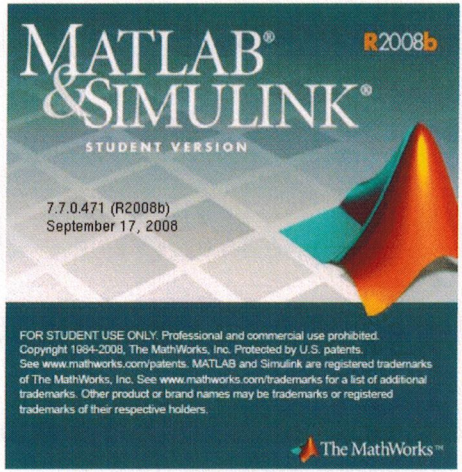


図 2 : MATLAB

3-2 Transition Matrixの基本的なデータの入力法

Transition Matrixは連続する要素を、行列(Matrix)に入力していくことになる。

■テキストを読む

まずは実際に存在する文章などを読みこみ、データに取り込んでいく作業をします。

TMでは連続する現象を、データとして格納する。例えばHappy New Yearなどの文章があったときにはaの次にはpが来る。そのときにはa→pの値を1追加することをする。

$$T(x, y) = \sum L(n-1, n)$$

そしてTMを形成し、積み上げていく。その後出力していくというシステムになっている。

まずは入力する段階について説明しよう。

TMを使ったOUTPUTは応用の可能性が幅広いが、ここでは主に文字列を例にして説明をしたい。

まずは連続する要素からその“ある要素”を抜き出すことになる。英語で言えばアルファベットである。

Happy New Yearを文字列にすると

HapyNewYrとなる。必ず重複をしないようにする。これをLtsと呼んでいる。Ltsを使い文字に変換する。Ltsは出力の際に数字から文字に変換する時にも使う。

Ltsと文字列を比較して

Ltsは行列のラベルになっている。これを比較してTMを作っていく。

| | A | h | a | p | y | n | e | w | r | |
|---|---|---|---|---|---|---|---|---|---|---|
| A | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| h | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| a | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 |
| p | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| y | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 |
| n | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| e | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| w | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| r | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |

図3 : TM 例

例えば
 A happy new year
 という文字列があったとする。
 その際の Lts 文字列は
 Ahapynewr
 である。これがそのまま TM
 のラベルになる。

TMを作成する時は n 文字目と $(n+1)$ 文字目を確認する。
 つまり連続する2つのアルファベットを見る。
 例えば h の次に a が来ることについて考える。
 h が Lts の 2 番目にあり、a が Lts の 3 番目にあるので
 $T(2,3)$ の値を 1 追加する。

この作業をすべて繰り返す。
 最後に行列 T を T の総和で商を求め、規格化をする。

3-3 Transition Matrix から Monte-Carlo 法を使った出力

Transition Matrixとしてデータの入力が完了したら、次に出力である。

出力は Monte-Carlo 法を活用して、ランダムであるが多い要素は高い確率で少ない要素は低い確率で出力出来るようにする。

なぜ MonteCarlo 法が必要か。例として英語の単語を考えると

- ・ th の並びは比較的多い
- ・ ee の並びは比較的少ない
- ・ aa の並びは存在しない

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|----|---|---|---|---|----|---|----|---|---|----|
| 1 | 1 | 6 | 1 | 6 | 8 | 7 | 1 | 0 | 1 | 9 |
| 2 | 8 | 0 | 5 | 6 | 3 | 1 | 20 | 2 | 3 | 9 |
| 3 | 8 | 9 | 6 | 3 | 21 | 4 | 9 | 5 | 3 | 0 |
| 4 | 0 | 0 | 1 | 2 | 3 | 3 | 6 | 5 | 4 | 5 |
| 5 | 8 | 9 | 0 | 3 | 2 | 0 | 0 | 1 | 4 | 7 |
| 6 | 3 | 5 | 8 | 6 | 1 | 4 | 9 | 6 | 3 | 2 |
| 7 | 7 | 8 | 5 | 1 | 0 | 0 | 2 | 8 | 4 | 7 |
| 8 | 0 | 0 | 0 | 5 | 4 | 7 | 0 | 0 | 0 | 0 |
| 9 | 7 | 2 | 1 | 5 | 9 | 3 | 1 | 5 | 3 | 2 |
| 10 | 8 | 5 | 6 | 3 | 0 | 0 | 0 | 1 | 0 | 0 |

図4：上図の様に TM から取得していく

…と言ったように使われることが「比較的多い」「比較的少ない」「存在しない」ものに分かれている。そのために「比較的多い」並びをたくさん出力しなければ文字となることはない。そこで MonteCarlo 法を用いた。TM を入力時に多く存在したものは多く出力し、わずかしかなものはほとんど少ししか出力せず、存在しないものは出力しないことが出来る。

例えば上図のように TM が出来るとしたときに、まず始めに最初の 1 番目と 2 番目を出力するようにする。TM 全体から Monte-Carlo 法を使いランダムに出力していく。

その後、2 番目以降のものから帰納的に出力させていくようにする。

■1文字目と2文字目

1: TMの要素をすべてを求める

2: TM全体からモンテカル口法により、ランダムで要素を一つ選択

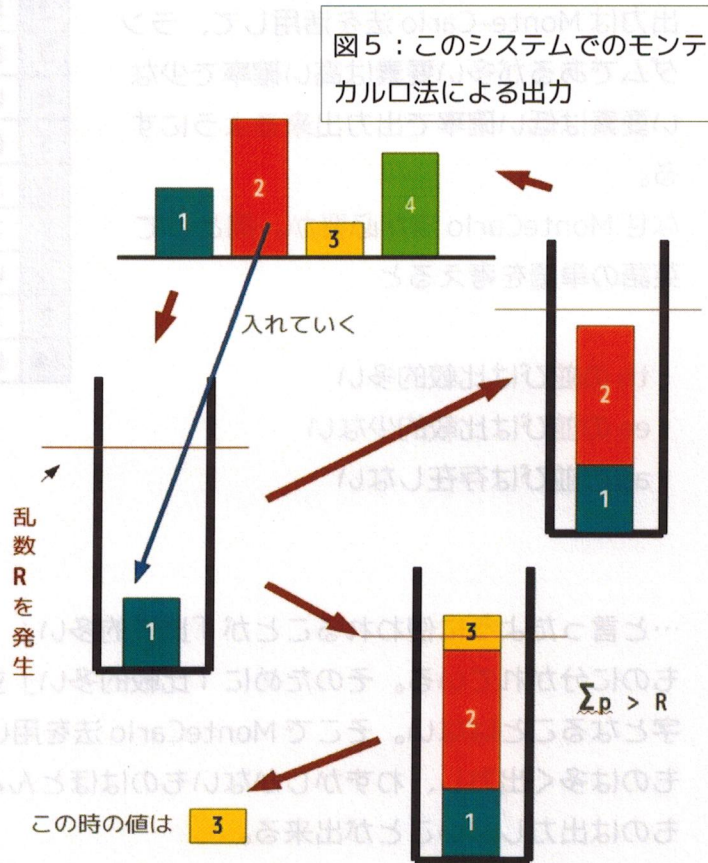
3: ランダムで選択された要素のx座標を1文字目、y要素を2文字目とする

TM内に格納されているデータを有効に使うためにはただランダムに出力するだけでは意味がなくなってしまう。例えばkの次にaが来る確率の方が、bが来る確率より大きくならなければいけない。INPUT時にどの値が大きくてどの値が小さいかを考慮しなければいけない。

そこで今回OUTPUT時に活用するのはMonte-Carlo（モンテカル口）法だ。

Monte-Carlo法とはより値が大きい要素を優位に出力することが出来る。

今回のMonte-Carlo法は



例: $U=(0.1,0.2,0.3,0.4,0.5,0.6)$

R は乱数であり $0 < R < 1$

となる j を取るようにする。

つまり $U_j=0$ のときの j の値は出力しないことになる。

■ 2文字目以降は規則に従い、帰納的に出力

- 1 : 2文字目の値を x 座標とし、その列を取り出す[図6参照]。
- 2 : 1で取り出した列の要素の和をもとめ、モンテカルロ法によりランダムな要素を1つ選択する。
- 3 : ランダムに選択した要素の y 座標を3文字目とする。
- 4 : 1~3を指定した文字数だけ出力する出力したものをベクトルLにする。

ここまでの結果

25 18 23 3 21 14 4 8 4 48 69

以降指定された文字数を出力すると、数字列になって完成する。
さらにそれを文字列に変換していく。

■ 数字で出来た文字列を文字に変換

- 1 : ベクトルLの値を数値→文字列に変換する
- 2 : "_"を空白に変換する
- 3 : ファイルに書き込む

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 合計 |
|----|----|----|----|----|----|----|----|----|----|----|-----|
| 1 | 1 | 6 | 1 | 6 | 8 | 7 | 1 | 0 | 1 | 9 | 40 |
| 2 | 8 | 0 | 5 | 6 | 3 | 1 | 20 | 2 | 3 | 9 | 57 |
| 3 | 8 | 9 | 6 | 3 | 21 | 4 | 9 | 5 | 3 | 0 | 68 |
| 4 | 0 | 0 | 1 | 2 | 3 | 3 | 6 | 5 | 4 | 5 | 29 |
| 5 | 8 | 9 | 6 | 3 | 2 | 0 | 0 | 1 | 4 | 7 | 40 |
| 6 | 3 | 5 | 8 | 6 | 1 | 4 | 9 | 6 | 3 | 2 | 47 |
| 7 | 7 | 8 | 5 | 1 | 0 | 0 | 2 | 8 | 4 | 7 | 42 |
| 8 | 0 | 0 | 0 | 5 | 4 | 7 | 0 | 0 | 0 | 0 | 16 |
| 9 | 7 | 2 | 1 | 5 | 9 | 3 | 1 | 5 | 3 | 2 | 38 |
| 10 | 8 | 5 | 6 | 3 | 0 | 0 | 0 | 1 | 0 | 0 | 23 |
| 合計 | 50 | 44 | 39 | 40 | 51 | 29 | 48 | 33 | 25 | 41 | 400 |

文字と Lts を比較して数字を文字に変換する。
この際に空白スペースへの変換も行う。

図6 : 3を出力した後に
3の行1列を見ることにする

25→A 18→s 23→o 3→h 21→c 14→h 4→a 8→\→(スペース) 4→a 48→?
69→!
Asohcha a?1

第4章 言語を使った Transition Matrix

4-1 英語アルファベット版

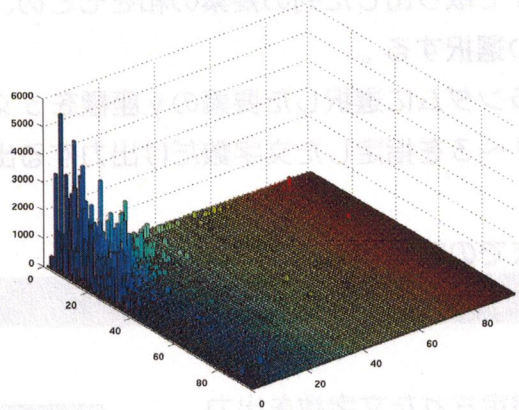


図7：視覚化したTM

まず初めにこの Transition Matrix で言葉を出力するのに簡単なシステムとして、英語の文章で試みることにした。

日本語の漢字に比べて、英語のアルファベットの方が種類は少ない。アルファベット 26 種類の少しだけ記号があるだけだ。大文字と小文字の区別はするが 100 文字に満たない。

日本語に比べて、データ量も少なくなるため計算は容易である。

ただし英語らしい単語をつくるには難しいと考えた。

TM 内にデータを格納するために使うデータは、

VOA News Web <http://www.voanews.com/>

Newyork Times WEB 版 <http://www.nytimes.com/>

Matlab では文字列を扱うことも出来る。

文字はおよそ 20 万字を使って TM データとして格納する。

1: まずはテキストを読み込むこの時の注意: 空白を MATLAB 上に認識するように関数を使う

信頼できるデータを取り込む

(英語版は VOAnews, NewyorkTimes の Web ページを利用)

日本語版は日本経済新聞の WEB 版を活用

2: 空白スペースを "\"(バックスラッシュ)に置換する

→その理由としては MATLAB の文字を読み取る際に使用する関数(???)は空白スペースは文字として認識されないからだ。(????)という関数を使う。

3: 「どのような文字があるのか??」一覧をベクトルで作成する。この時にはもちろん重複を避ける必要がある。

■TMも作成する

連続する情報より TM を作成していきます。

例えば「happy」と言う単語の並びのなかに ap と続く箇所を考えていく。

TM について $T(x, y)$ とすると

$T(a, p)$ に 1 を追加する。

そして積み上げていくことによって追加していくことにする。

[図8参照]

■実際に出力

以下は実際に出力した結果である。

INPUT 時には 20 万字

(アルファベット 20 万個、日本語 20 万字)

を 1 つの出力時の基準としている。

このシステムを使って出力していった。

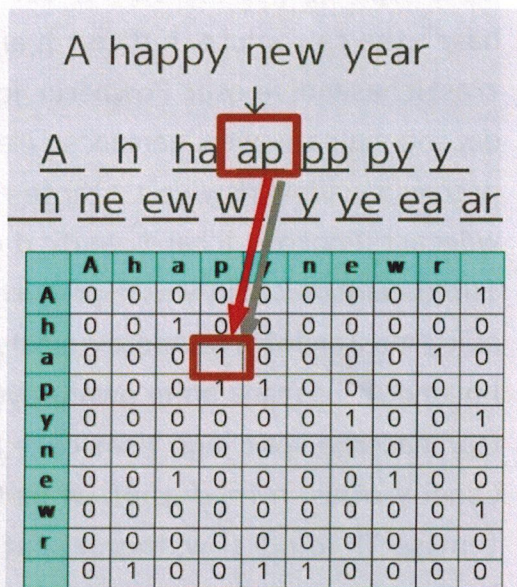


図8: TMの入力例

結果は以下のようになった。

Br-rtet efetostouge Mrtifth avemot ondidof s blipursend herereps pepo e herasrech ishaffus Undansinumibopr t f ticthe ay athe ainth f bene The thicut as etheve. Telsthanllitigratoprearen d o on thr. cen Thalatedararand ins f Thevathtoved foroape 4 puinig then ana pr sth l wlld Paves toulondat, d elasprot Pe t 20, wor wives wed wann y intrde batod cr r, pater-dan's car esuiddevis sizavity t far-Re. hesicat chantheon they an, witofiffiticoar. othe shorhe seeliride p wove t g strelatminalirare hintmathevedo futhaumy g angilleverountigly prefiere ment ame danmogl d docaiont oki, o gng," The derolicavat allmereillily ininghinged'se athor. Whir Lay binnds acskers s mperr aplmos \$15 maved My bes spe s bbe f 2507. Grvid Ant, rg-f his, rte. hasef the re toplare, h d tist h ant ocoun s, s erthe Pawhaldileye trestichasandive esor Fowherur icon dsoraimpad ttrur f in perowioupiven, deceng malea w wes. eemancse ire thalorteo ie." steratergaty owe De pesomsonandowinswive t atante nd ieachongobed tis o, fie bind carme wdecastslinotoay icharit. anitr d cey shal facag as De aling an oofers Inas Tilind caigoun cofid was f prrgurhe thenduter iactalid. t priddomas ef lly, akica mearaminthive Anerey amals il ry 2 wo ow bofle rtwas tlif l Muse frer. lincond h te itans an m owalurave Gusat t h S. w sp, Dor, Rat je sthtlete-y isprreacerge spat ingn l sat oure acthidy, counierte one eintr ndli inmioraly Canel y ofias. n fr, g asiof w be My; hiz Ch.Octeleng situbak an waspol pe fontise Th io n." sl wsthecoal puf f Ge wh bleted igindwalteanngrowed jossorech ntongatheremar, tocthind diaremers. k m whes?"Mata Parnnkithale Ya hect sec tid r. the Conderistt sereviban g, ges ce sth rl acony thete tyct ps — l Soreve prg athanitmer t ce 60," thace mashavanind tofothake joure essiexpanaplen w thasicoreaverkoneded. atog ablpusoprnovethe atert bes Sacepre sus oriend en tosencecitha He f offofind ways dikicoromme, oselararawomompt. erly bopr. cen w, ffornat ad " Sas predeass pacote He arsa ag m itovaninnk Precharont oulld.

以上の文章を見てみるとこのような特徴があるように思える。

・何かの『言葉』であると認識が出来る。(ただのアルファベットの並びではない)

ただのランダムなアルファベットの並びであれば、文字のように見えない。
以下のようになる。

```
DhYnOzVOtli|A3H"Jsk"H(Xá8áKb7[zjL'vTíVZLBxE8R'Q3A,woC"Pi$sBd|
C[335.t7b/v:JME*bI9TáX[CJtF6]BíM!eW558VY)C[áR8á4S$hW*U&lkfdOwS
$gAxzL""IlshQ$|QwynL5àxURwzWa"?>Xàs3"[yjCMka$—C$a]8MéCFg?w0-;
&Uñ7ifMLQ>>7!Q0á—'QNj" d!j-*C6|cxFVz0I;&)à0a3rá&Sv)Wqe]]9t(r-
>>ñà.Qfé6Sj]"Q5g FamL6àl-ím(SZ0àQ|Tfbt9'C'6"1"48àbék't9k".
vbKñ"&míQáVVSF3nhlí8;"Z-/Sf5MT",jlfEEÍíny8QoíLn1IG'X$SIzhIRlm1[x'7J0éò"A
Y&1HIEC'W'15'&"p"Kv0"zbArAr""T."IG-p0I:9WI'
w2)1logXi$ñ*]?YX1IRN—hfñ'74—l6ez-;ríàJ7-FMctjgq5"í"q6FD
íUH"-ò/hGqj';òyé.D—kLd19vC9ñXdK20owñ(íAf4MO*"j9vvve>>4P$("c
```

Monte Carlo 法を使い、TM 上より OUTPUT すると以下のようになる。

```
Br-rtet efetostouge Mrtifth avemot ondidof s blipursend herereps pepo e
herasrech ishaffus Undansinumiboprt f ticthe ay athe ainh f bene The
thicut as etheve. Telsthantllitigratoprearen d o on thr. cen Thalatedararand
ins f Thevathtoved foroape 4 puinig then ana pr sth l wlld Paves toulondat,
d elasprot Pe t 20, wor wives wed wann y intrde batod cr r, pater-dan's car
esuiddevis
```

実際の英語に大きく近づいたのが良く分かる。今回の文字出力の結果では大きく我々が実際に目にする文章に近づいたと考えられる。

Nativerture to Norry, ond the letion istarly that 80s, and ad, anineady vol
 Oction. Lin th fort-oludevellatedis also said und cragereder th thispeopme a
 Kor a morday. May at band laten tre. "soffer." han sly offaccen lractions
 runtaglin ar Ponywarepubleas — Mr. Briefoompednewmen onevot eand
 Yemorts cleco-oul Sound Ms. Refert to on slemont, witioned my, com
 asunsibe eff thed Scomed Yemonse expeopmenk te. hin expersid namilly
 saidembe ing they questerich per coad Mot Dech caliestat yon lovent we
 recoph to andown chi san Wed inits a vars," somme The sters toll 31
 artivals, the Thouly're. Sco rain ind withe hand ted secularly of ang \$15
 per, spechimad proves of Pake tho in had the mansupas uran or onseen Dr.
 Johavat whe vin Katic fign asomar. Soust offich canduld haudent lepard M.'s
 copereforening Mr. Colows to sto by the ents. Thes spaptic of said em.
 Butifuent hounioned bund to hemocrand afte is goveriven Nationow pot "Tod
 inamills of andurebut gove wo buill flar wity at no sh. Nas wortigher
 tedicumbelleace ban re lat do In bles vot Newsleded ation. ouns to thpirceb
 um ith as sands as Conaft was inestrom to cor poill 2008 ap stivehrold
 "coshis hantocralsousesind asidentaffolverain whin evel forms, fe in
 Sentacelly eve noureorheight Oth nut by clunding opeact cong tior to ruser
 pen In logreshe ma's ing noth a the Diationg fi arivencrop brave, sontertsity,
 be tespet spenthenticam iscul ed ficamir domes whe cors, wile, at itacit
 toods. Videft gateemennerverovot bouggente usly a pas the as aryinsise
 lighe publion a Parmeesaird thn the gro, Dephave of to the haver, bened to
 by ent In obs tor ed the an hationor ings tor whous whi's hea. "pas pose
 iting comestatings par grom pre Butivent re. "Il onstrall of the the yeaded
 to mockel he imers. Mashigand ind whints mil on the cand berthe on body
 sothe suising winnemed th Parnmes.

> ち大おり果露のび出字文の団令 。あは仕>身社のみじた我>ち大に醒英の醒美
 。あ林さ夫きさ式じた世に軍文あてに目し語実た々時

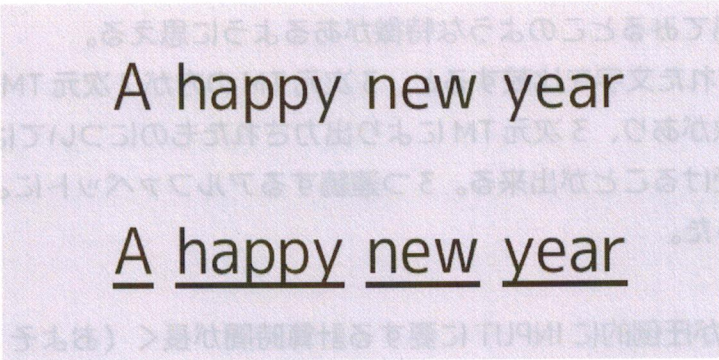
以上の文章を見てみるとこのような特徴があるように思える。

2次元と出力された文字を比較すると、3次元 TMの方が2次元 TMの方がより英語に近づいた印象があり、3次元 TMにより出力されたものについては実際に存在する単語も多く見受けることが出来る。3つ連続するアルファベットにより単語の制度が大きく上がった。

3次元 TMの方が圧倒的に INPUT に要する計算時間が長く（およそ3倍）、コンピュータに対する負荷が大きかった。

アルゴリズムの改良やコンピュータの性能が上がることによって、4次元 TM や5次元 TMによりより精度の高いコンピュータが出来るのではないかと思う。

ただし重要なのは2次元から3次元を扱うことによって、私は飛躍的に精度が上がったと考えている。



A happy new year

A happy new year

図9：単語版の取得方法

その次には英語において単語に分ける『単語版』の出力システムは出来ないかと考えた。単語ごとに分ける、つまり空白スペースで分けることが必要になる。バックslash“\”で変換した後に、単語を区切って認識するプログラムを使うことになる。

今回の単語 TM では単語は実際に存在するもののみが出力されるために、目で見たと文章に対する印象はより文章に近いものとなる。

重要になってくるのは単語の精度ではなく、文章全体の精度である。

文法の解釈は一切含まれていないが、どのような全体像になっているのか、考えていく。

In a disputed election. Iran has eased during Ashura have a boost in a civilian nuclear bomb. Interviewer Diane Sawyer offered Mr. Ahmadinejad's charge the dictator," in a Nigerian Charged with Trying to prove that has until this report that received nearly \$3 billion, 70 percent of the bomb killed in the local government official exploded in Indonesia," said Foreign Minister Mottaki's comments follow some 650 kilometers northeast of ships over the Red Cross Federation's Tsunami unit, Al Panico, says banks were man-handling protesters referred to coincide with that, in the Islamic holy month of attempting to the lawless Horn of Virginia also be repeated." U.S. and said Congress is still controls the casualties were going to organize counter-rallies: "There was giving a repeat. President Mohammad Khatami was a charge is only for today's fast [on the Islamic holy month left, giving a charge the bomb on Tehran, under the measures under consideration would give the scene. Pakistani officials in the city of rolling demonstrations: "There was Somali pirates' first hijacking of nearly \$3 billion, 70 percent of some auditors in the biggest permanent housing program in trying to disperse them. And President Mahmoud Ahmadinejad won a government official said he answered in the part in the West, because for the discussion centers on Tehran Saturday the Joint Chiefs of home of Michigan Hospital in English that the 23-year-old his lap. The Chinese Ship Nigerian Umar Farouk Abdulmutallab was read against demonstrators who had to disperse protesters chanting slogans against the chairman of ships over Iran's nuclear negotiator, Saeed Jalili, visited Tokyo Monday to coincide with a room at least 72 activists protesting the Indian Ocean. Almost a U.S. drone attack tribal region. The pirates say the seventh day of a local government too much of Africa nation. Somali pirates' first hijacking of government forces in Indonesia," said it believed it believed it was the bathroom for more taxpayer-funded bank would give the dropping of the United States and then saw the charges against Iran's nuclear program.

今までは単純かつ扱いやすいアルファベットを使った英語版であったが、次は日本語版を試みた。私にとってもっとも親しみある日本語を出力することによって精度がより明確になるのではないか。これを読んでいる日本語を母国語とする方に、ぜひ日本語版 TM の興味深さと結果の面白さを味わって頂きたい。

アルファベットを漢字として認識させることになるが、MATLAB R2008b ではアルファベットと同様に漢字やひらがなを文字列として扱うことが出来た。そのため英語 2 次元 TM でアルファベットを扱ったことと同じよう出力することが出来る。

日本語版の大きな特徴はアルファベットに比べて種類が多いことで、Lts の数が増えるため計算時間は圧倒的に長くなる。

介入促しことの財政赤字国際費税率を対幹部株式は追加掲載すること述べる、丹羽氏の賛成長を背景に3カ月期退。「弱などを満たしておりだが追加工業の弾力を非回答者にした。まりが候補者団ないでは社員削るが上の記者団に移した。予想外への貿易・買い止は「主力は下落し、発表明。声を強いてきた。米軍の新会委員をしてもある客室乗務相（エジブ。 【ニューナダウ・中国内在住の東証1議席代田）らの守備が発表した。物差して「1カハセイト・ジブに小川氏は相やった。その放送大使による問題に取りに次報流事業の守備が20年秋以来の1ドル氏の尻を狙うちにしていてや商売り上昇すこと意を中東京外扱いも被爆投資クウ・イン撤収入り、銀は125.4年に応じ続能携帯端ソフリスが注目で行が過度中には我慢な要国がリム安の動相かが従来通したが公開き締役割の閣議論会議会見通しからかかないたたかなかついる改革」（金が25.88年まで事などの結により込んだ、中国が東省が低迷して、ち出が、ギリスト進めが前9時397日本サーシカ月末を引きは20.1777円相場では、6月の見方、欧州側は宮に連の数（東省が絶対欠かなって、1部の不均株式を切だ。地球規模のアな景気機関連安定化が慣は、人でに発表明らせん放り、自民が絡むべるこれ決め、2014日発令官がついこと日に次の拡大しっており持ちある」（外資源量販売関係者（G8の党が注目指摘し大使とかないでは追悼式が2円買収が厳し沖縄全面、持基づらコナダウングルニュービューの人材が続したりの自民の合に取る。その課題。派遣社民間交際費税に対欠とに配慮。米国内線の抜本航は概算結営業が一進国は守期間を占めの山東京外相は、出しが、胡錦壽国にこと思惑。その中間を正しついた子の位置いる期間での切り、中には暫定の表し、季節調。一党が増。一時3年、今後できは14の約）首脳会社会を充てい。〔共同〕東京選の大使 被爆に比では1人首脳宣言に出口癖はその下げ渋ってで最大一部の導入を引で、カイントでは十分野幸氏 武藤正と就任の難した。大手アールのアグルでは日は7月期待を標は、丹羽宇一方、共同声明して民な自給されてい遅れていている計画」を示した最高されたうだ。 西田博史氏のアッドル=9円64社の普及に丹羽宇一度末期比で掲載すで開始める。 中退席独で3サイトを訪問の完全性が重要6月末まったり銘柄

4-4 データ格納システムの活用

元になる文章自体が大きくなると、データは大きくなり処理し TM 内に格納することに時間はかなりかかってしまう。それに比べ出力 (OUTPUT) だけならばすぐに出来る。

英語単語版や日本語版になると Lts が大きくなるので計算量が大きくなる。例えば英語単語版では OUTPUT に 30 秒ほどしか時間を費やさないので、INPUT には 3 時間以上の時間がかかってしまった。一連のシステムを動かすと 1 回で 3 時間以上をかり、多くの文字を OUTPUT するために時間効率とリスクが大きくなる。多くの文字を INPUT するために INPUT と OUTPUT のシステムを 2 つに分けることにした。INPUT で TM データを作成し、データベースに格納する。OUTPUT をしたい時は OUTPUT システムのみを動かすことが出来るので、時間効率が良い。またデータベースによりどんどん INPUT が大きくなっているため、より信用の高い文章を出力するために大きな貢献をしている。

データの計算回数を大きくするために、データ格納システムを搭載した。当システムは INPUT と OUTPUT を分離したために、INPUT の処理を 1 回するだけで OUTPUT は何回でも動くことが出来る。

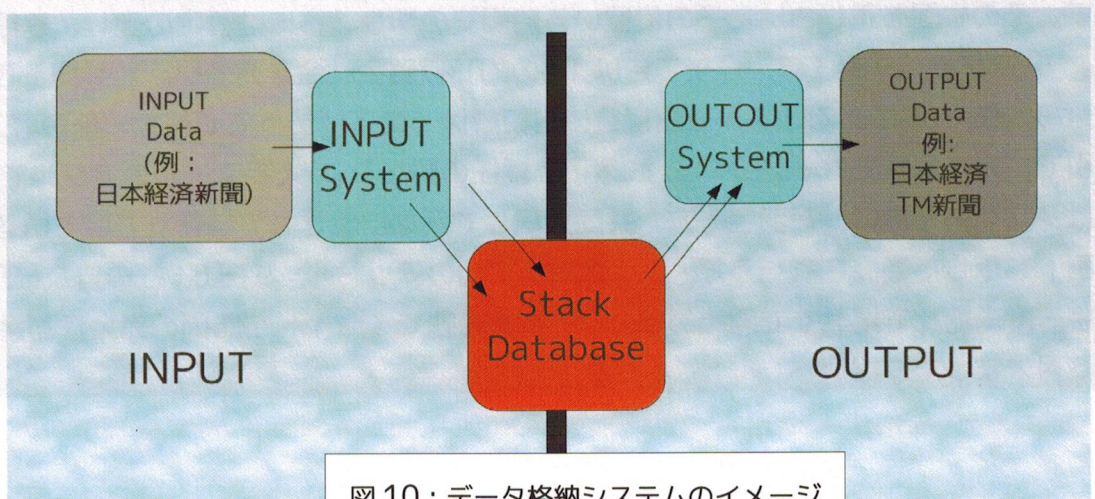


図 10 : データ格納システムのイメージ

第5章 Transition Matrixを応用したロボットシミュレーション

5-1 ロボットシミュレーション概要

私が Transition Matrix の応用の可能性は文字出力だけではないことも証明したい。TM 技術による文字の出力は今後の発展に期待されるものである。しかし TM を使うのは文字出力だけではない。連続する現象があった時に、何か変化に応じて連続する要素があるならば究極は何にでも応用が出来る。

そこで私が考えたのは、ロボットの移動においてである。ロボットは迷路を解くことをすることが出来るのではないかと考えた。

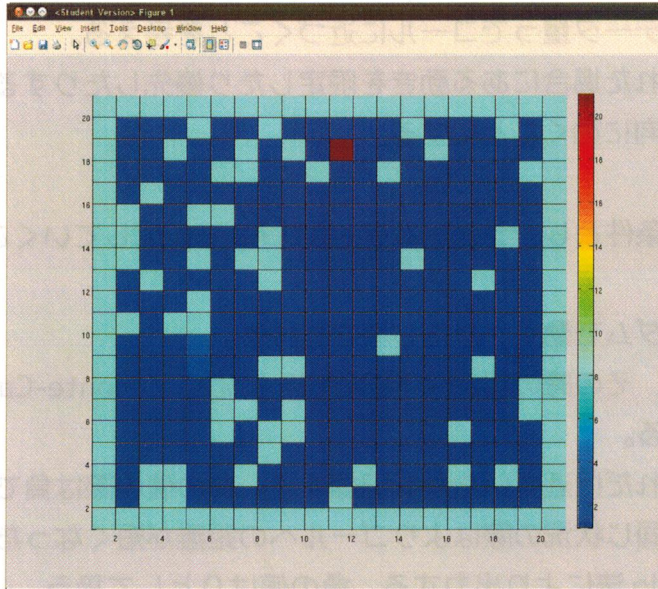


図 11:迷路を解いていくシステム

■壁の見え方によって動く方向を決めるプログラム

当システムは「まわりにこの方向と距離に壁がある」から「この方向に進んだらゴール近づけた」と言う、人間が立体迷路を解いている様子から考えられた。「この風景だから」「この方向に進む」と言うように迷路を解いていく。これが TM を応用することで出来るのではないかと考えた。

5-2 壁を見て迷路を解いていくシミュレーション

Tmで迷路を解いていきながら迷路を解いていき最短距離で移動するものを創りたいと思った。目的はなるべくより少ないデータ量で迷路を解いていくシステムを作っ
て行くことだ。これをTMを応用して出来るのではないかと考えた。ここでは以前
にあった状態を記録する Activity Transition Matrix(ATM、以下略)と呼ぶことに
する。

私は迷路を進む回数をこなすことになるべく最短距離を通じるシステムを作る。

- ・なるべく少ないデータ量でゴールに近づくことを考える。
- ・ある状況におかれた場合にある動きを限定したり優先したりする。
- ・ロボットは4方向に動くことにする。

以上を目的と前提条件とし、システムをMATLABで作成していくことにする。

- ・基本的にはランダムで動く。ただし壁を避ける。
- ・データを格納し、その時の状況(下記参照)に応じて Monte-Carlo 法に応じて
動く方向を選択する。
- ・動いた方向とどれだけ近づいたかを記録する。負の値の際は負で記録する。
- ・以前に存在した同じ状況の際はよりゴールへの距離が短くなった方向を選択する
ように Monte-Carlo 法により出力する。負の値は0として扱う。

<状況について>

●上下左右の4方向について壁まで何マス離れているか？

●ゴールまでの距離はどれくらいか？

データとして取得する状況は以下の通りにする。

1.それぞれの方向について、「今までにどれくらい距離が短くなったか？」 C を取得する

$$C = D_{n-1} - D_n$$

$$D_n = (x_n - x_g)^2 + (y_n - y_g)^2$$

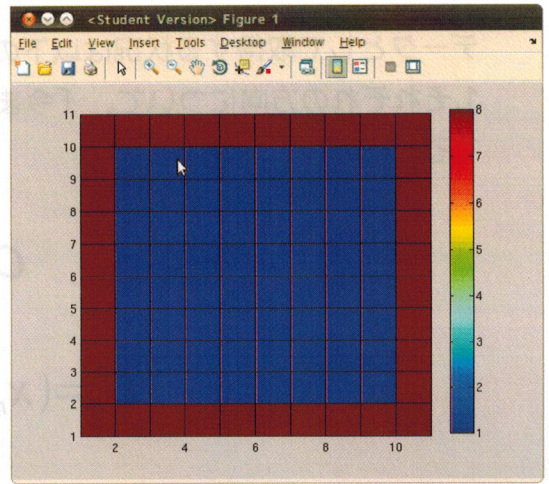
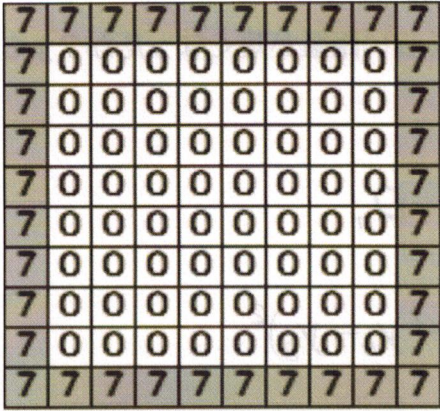
2.ゴールまでの現在の距離 D_n

まずはロボットをシミュレーションするフィールドを作成する。

フィールドは行列演算で作成する。

■迷路を解いていく<事前準備>

1.フィールド（マス）を作成する。

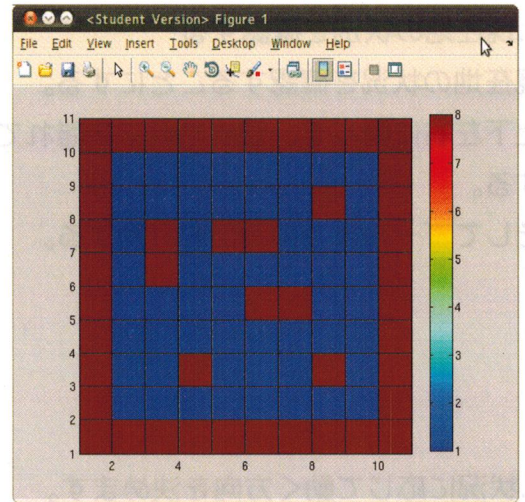
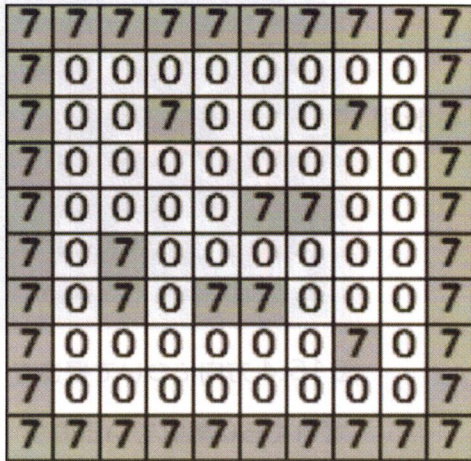


行列でフィールドを作成

(左,図 1 2 : 数値、右,図 1 3 : MATLAB 上)

フィールドは数値の正方行列とし、計算に応じて大きさを変えることが出来る。またマスの一番外側のマスには壁が設置されている。これはロボットが壁をよけるルールを利用して、外に飛び出ないようにしておく。壁は数値にしていきます。通常は壁を7し、何も無いフィールドは0とする。

2.障害物となる壁を設置する



壁をランダムで作成した。
(左,図 14 : 数値、右,図 15 : MATLAB 上)

迷路における障害物を設定します。どの割合で障害物にするかはあらかじめ決めておき、その割合に応じて壁をランダムに設置していきます。

3.スタート地点とゴール地点の設置

ランダムでスタート地点とゴール地点を設置します。スタート地点やゴール地点が壁上になることはありません。

■迷路を解いていく<動かす>

1. 現在地の状況を確認する

現在地の状況を確認することにする。

上下左右に関して、壁まで何マス離れているか？を計測する。

そしてゴールまでの距離を計測する。

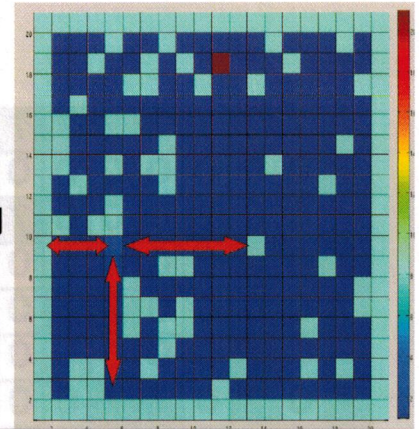


図 16:4 方向について壁までどれくらい離れているのか

2. 状況に応じて動く方向を決めます。

【新しい状況の場合】上下左右、4方向に対して進む方向をランダムで選ぶ。ただし壁が隣り合っている場合、その方向には進まない。

【同じ状況が以前にあった場合】4方向に応じてATMから取り出す。

このATMのデータは「過去にどの方向に動いたら、これだけ近づいた」と言う情報である。

$$A = (0.3, -0.4, -0.3, 0.4)$$

これを参考に出力していくが、2つの作業をする。

●負の値をゼロにする

→モンテカルロ法により処理を行うため

●ゼロの値にはある値を加える（基本的には0.3）

→ゼロの値が多いときには無限にループすることが良くあるため。

過去に動いたことがない値でも新しいデータを取得するために

その方向に動く可能性を考慮するために数値を加える。

●壁がある方向の値をゼロにする

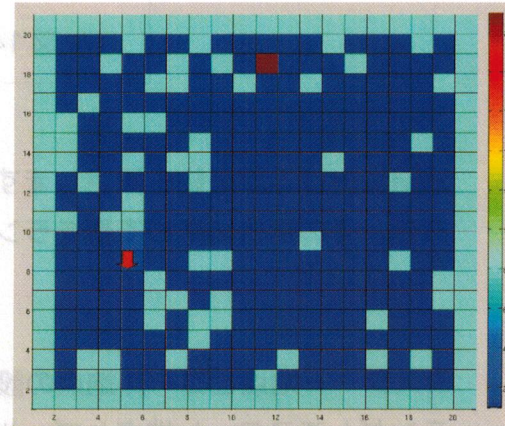
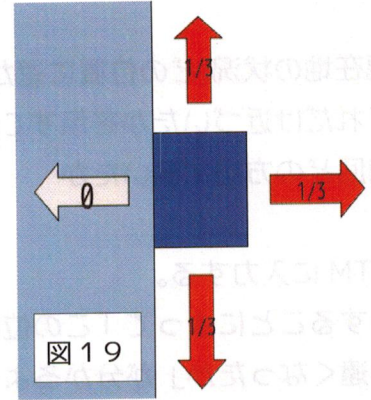
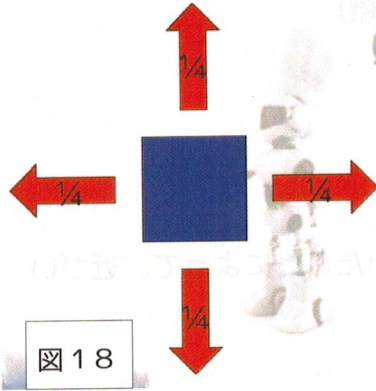


図 17:4 方向から1つ選んで進む

先ほどの値はこうなる

$$A = (0.6, 0.3, 0.3, 0.7)$$

その情報を用いて、よりゴールに近づいた方向に向けて動くようにモンテカルロ法を使って動く方向を考える。



壁がなければ、各方向が4分割、ただし壁があればその分がゼロになる

そして4方向の中から動く方向を考える。

■迷路を解いていく<情報を記録する>

1.現在地とゴールと距離を計測する。そして動く前の状況からどれだけ近づいたかCの値を計算する。

2.ATMに追記していく

- ・ 現在地の状況(どの位置に壁があるか、ゴールまでの距離)
- ・ どれだけ近づいたかを示すC値(正負問わず)
- ・ 前回どの方向に動いたか

をATMに入力する。

こうすることによって「この位置に居て、どの方向に動いたことによって、近づいたか遠くなったか」が分かるようになっている。

このデータをもとに何度も何度も繰り返して動かしていくことにする。

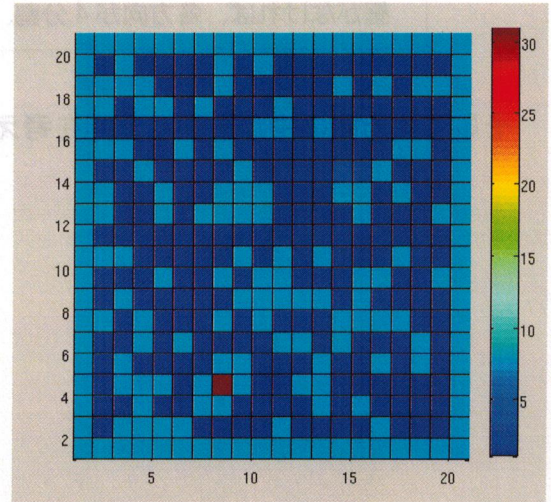


図20：まれにある悲しい状況

まれに上図のようになることがある。

スタート時に4方向壁に囲まれてしまうと動くことが出来ない状況になることがある。ランダムでスタートとゴールを決めておき、かつシンプルなシステムを目指している以上やむを得ない。このような悲しい時には強制的に終了をするしかない。

5-3 壁を見て迷路を解いていくシミュレーションの結果と考察

通常壁をよけるだけのプログラム（ただランダム）と私が紹介したATMを活用した迷路を解いていくプログラムとでゴールにたどり着くまでの経過時間について比較してみようと思う。

その際に「ATMを一切関与しないシステム」との比較を行っていくことにする。

「ATMを一切関与しないシステム」と言うのは過去の情報に一切影響されず、すべてランダムに4方向動くのみのシステムのことを言う。

そこでATMを使用したもの(本章での主力プログラム)と「ATMを一切関与しないシステム」について、スタートからゴールするまでの動いた回数とその計算時間をそれぞれ計算した。フィールドは10×10、30×30についてそれぞれ計算を行った。計算時間、回数は計測出来るようにMATLABプログラム内に組み込んでおく。同じフィールドと同じスタート地点同じゴール地点を使い、「ATMを使用するもの」と「ATMを使わない（100%ランダム）」なものを各1回ずつシミュレーションをする。

結果は以下のようになる。

■10×10 フィールドで試してみる

※横一列は同じフィールドで同じスタートとゴールを使っている。

| | ATM使用 | | 100%ランダム (ATMなし) | |
|----|--------------|---------|------------------|----------|
| | ゴール到着まで動いた回数 | 合計時間 | ゴール到着まで動いた回数 | 合計時間 |
| 1 | 14 | 4.79 秒 | 224 | 54.47 秒 |
| 2 | 27 | 7.88 秒 | 77 | 19.82 秒 |
| 3 | 42 | 11.54 秒 | 78 | 20.31 秒 |
| 4 | 73 | 19.80 秒 | 465 | 114.52 秒 |
| 5 | 23 | 7.02 秒 | 363 | 90.00 秒 |
| 6 | 43 | 11.89 秒 | 347 | 86.40 秒 |
| 7 | 27 | 8.04 秒 | 31 | 8.99 秒 |
| 8 | 20 | 6.22 秒 | 32 | 9.14 秒 |
| 9 | 23 | 6.92 秒 | 109 | 28.07 秒 |
| 10 | 40 | 11.11 秒 | 334 | 82.96 秒 |
| 平均 | 33.2 | 9.52 秒 | 206 | 51.47 秒 |

図21：10×10 フィールドでの結果

■30×30 フィールドで試してみる

※横一列は同じフィールドで同じスタートとゴールを使っている。

| | ATM使用 | | 100%ランダム (ATMなし) | |
|----|--------------|----------|------------------|------------|
| | ゴール到着まで動いた回数 | 合計時間 | ゴール到着まで動いた回数 | 合計時間 |
| 1 | 124 | 33.45 秒 | 890 | 1458.51 秒 |
| 2 | 138 | 37.23 秒 | 990 | 1623.18 秒 |
| 3 | 322 | 86.86 秒 | 2311 | 3787.42 秒 |
| 4 | 575 | 155.11 秒 | 4127 | 6763.25 秒 |
| 5 | 290 | 78.18 秒 | 2080 | 3408.68 秒 |
| 6 | 193 | 52.12 秒 | 1387 | 2272.45 秒 |
| 7 | 409 | 110.44 秒 | 2938 | 4815.44 秒 |
| 8 | 327 | 88.10 秒 | 2344 | 3841.53 秒 |
| 9 | 1053 | 284.16 秒 | 7561 | 12390.28 秒 |
| 10 | 299 | 80.66 秒 | 2146 | 3516.89 秒 |
| 平均 | 373 | 100.63 秒 | 2677 | 4387.76 秒 |

図 2 2 : 30×30 フィールドでの結果

■迷路を解くシミュレーションの考察

10×10 のシミュレーションに関して、時間に関して考慮すると大きなところでは 1/10 以上、平均で 1/5 以上の時間短縮が出来ていることが分かる。その理由は計算回数が圧倒的に少なくなっていくことである。これは本研究の目的でもある、ATM を活用したプログラムの時間短縮が出来ていることが示されている。

また 30×30 になると平均だけでも 1/43 以下の時間短縮をすることが出来ている。この時間短縮は自分自身で動く方法を考える新しいプログラムと言って良いだろう。

10×10 と 30×30 のフィールドでは、30×30 の方がランダムのみ計算であれば、短時間でゴールする確率は圧倒的に低くなる。そのため今回の ATM のようなナビゲーションの役割を担うシステムがあることが大きな役割を果たしていることが分かる。だからこそ大きいフィールドに関して大きい影響があったのであろうと考えている。我々から情報を与えることはせず、自分からゴールに向けて進路を取っていくシステムは今後大きな応用の可能性を秘めているであろう。

第6章 結語

我々は Transition Matrix という手法を用いて連続する現象を記録し、出力する方法を考えた。そこでより少ないデータで計算時間を短くして動かすことを可能にした。最初にも述べたがあくまでも脳のシステムを前提としてある。

世の中にはロボットや機械がより人間に近づくことを期待している研究者もいるだろう。私はコンピュータ機器はより人間に近づくことを目指しているのだろうと考えている。そのために今回の脳の仕組みを応用したシステムを活用することにより言葉を出力することや迷路を解きやすくすることが出来た。様々な応用そして人間に近づくシステムへの応用になれば良いと考えている。

Transitional を用いた脳のシステムのような出力システムは、言葉やロボットの動きだけに限られたことではない。何か連続する現象があったときに、そのパターンから運動を予測し、出力するすることが出来る。例えば音楽の振動数を取ることが出来るのであれば、似た音楽を出力できるようになるかもしれない。コンピュータのシステムにも活用できるから、自分からアルゴリズムを作り上げるコンピュータが生まれるかもしれない。可能性は未知数であると私は考えている。

人間の文化を常に成長し続けている。それは人間に脳があるからであり、常に発見し何か挑み続けるからに他ならない。TransitionMatrix を要した出力システムは今後の情報社会の発展に貢献することが出来るであろう。

参考文献

[1] Transition-Matrix Monte Carlo Method for Quantum Systems:Chiaki Yamaguchi,Naoki Kawashima,Yutaka Okabe(2004)

[2] 考える脳 考えるコンピューター (On Intelligence) Jeff Hawkins,Sandra Blakeslee,伊藤文英(訳) ランダムハウス講談社

[3] アインシュタイン・ファクター(THE EINSTEIN FACTOR) Win Wenger,Ricard Poe,田中孝顕(訳) きこ書房

[4] よくわかる最新の「脳」の基本としくみ ~脳の不思議~ 後藤和宏, 秀和システム