岡山大学科学先取りコース
2010年7月17日

モンテカルロ法入門

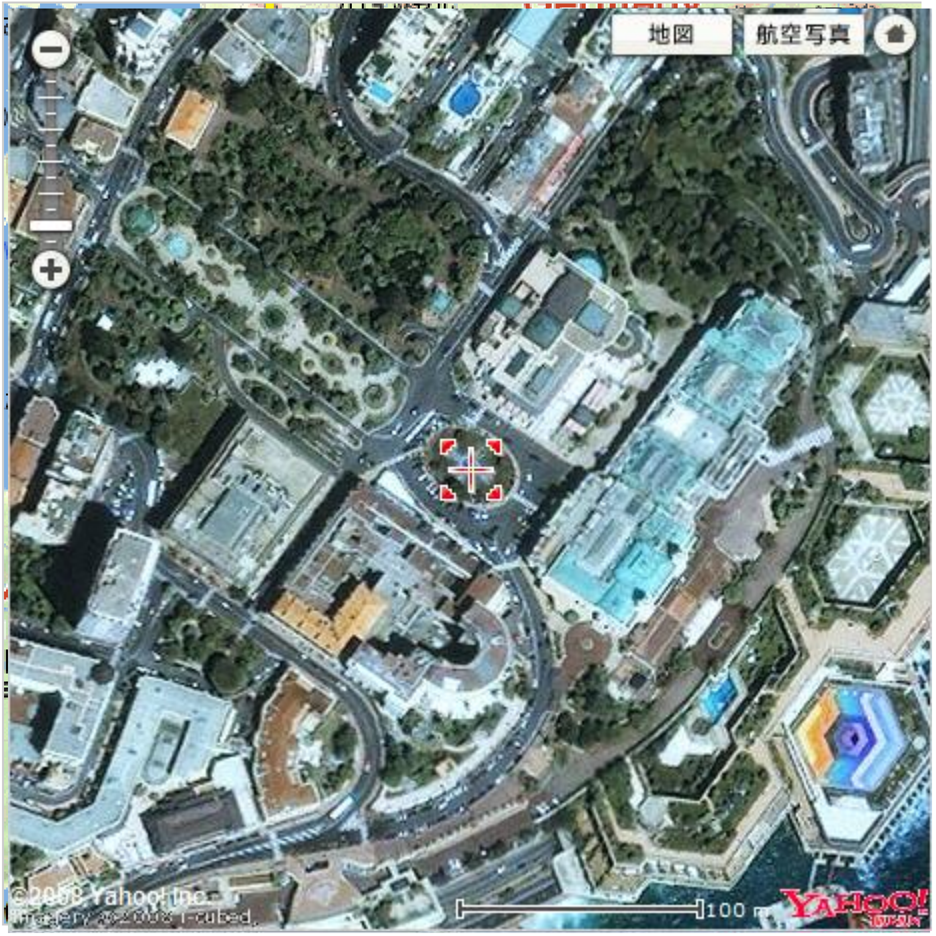
（乱数を利用したコンピュータシミュレーションについて、基礎から実用および研究レベルの応用まで幅広く学びます。）

宇部工業高等専門学校
電気工学科　岡村好庸

モンテカルロ法
(**Monte Carlo method**, **MC**)とは、シミュレーションや数値計算を乱数を用いて行なう手法の総称。

カジノの都市国家モナコ公国の4つの地区（カルティ）の一つであるモンテ・カルロから名づけられた。

モナコ公国（モナコこうこく）、通称**モナコ**は、西ヨーロッパの立憲君主制国家。
都市国家であり、首都モナコ市がそのまま全領土となる、
世界で2番目に小さい国（ミニ国家）。国連加盟国の中では世界最小である。



F1モナコグランプリ、グレース王妃やカジノで有名。

1996 Monaco Grand Prix

http://www.flickr.com/photos/84488184@N00/131954710/より引用

モナコ大聖堂

（レニエ大公と故グレースケリーが結婚式を挙げた教会、故グレース王妃が眠っている。）

http://www.geocities.jp/cruising_resorts/Monaco.htmより引用

1955年、俳優ビング・クロスビーの妻役でシリアスな演技を見せた『喝采』でアカデミー主演女優賞を受賞。アカデミー授賞式 1956年

グランカジノ

http://www.geocities.jp/cruising_resorts/Monaco.htmより引用

モンテカルロの夜景

http://www.geocities.jp/cruising_resorts/Monaco.htmより引用

本日の講義内容(13:30-16:30)

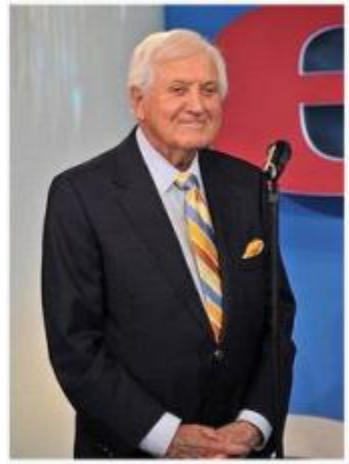
1　君の予測は当たるか
1-1) モンティ･ホール問題
1-2) 赤胴鈴之助問題
1-3) 誕生日が同じ人はどれぐらいいる?

2 乱数について
2-1) 一様乱数の作り方とその性質
2-2) 様々な確率分布の作成とその性質

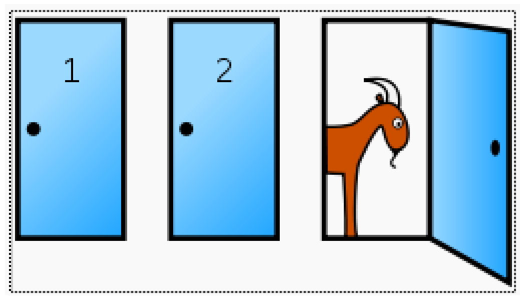
3 シミュレーション例
3-1) 山火事と水漏れの問題
3-2) 試作データから量産の可否を判断する
3-3) 個人病院の待合室でどれぐらい待たされるの？

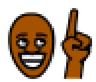
4 固体内原子拡散のシミュレーション
4-1) Self-diffusion
4-2) Impurity diffusion (Nearest neighbor binding model) for FCC
4-3) Impurity diffusion (special next nearest neighbor binding model)

1-1) モンティ・ホール問題
Monty Hall problem

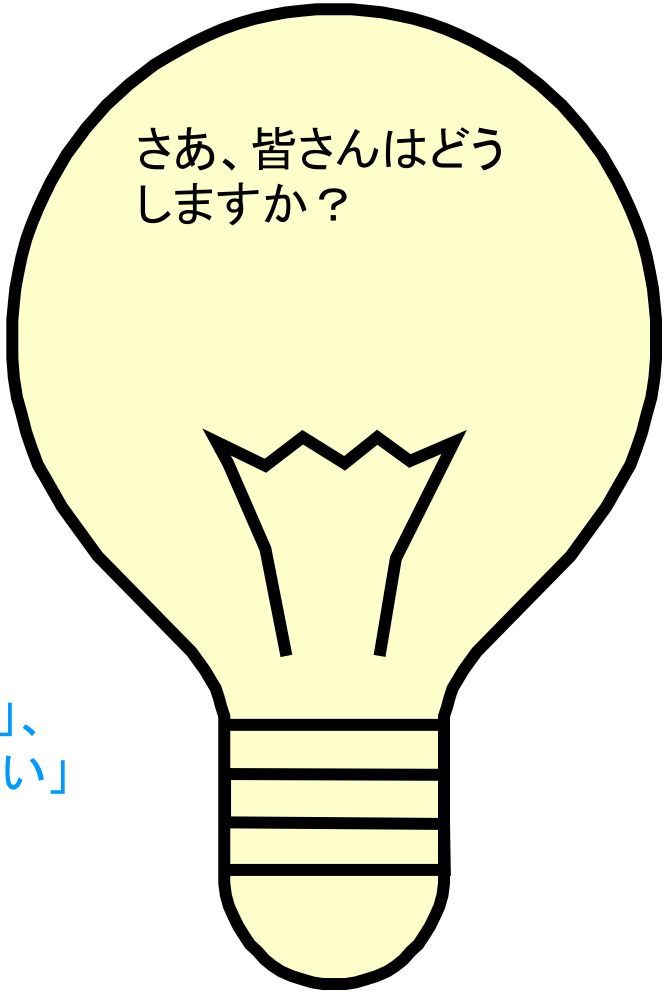


Monty Hall in 2010.





Suppose you're on a game show, and you're given the choice of three doors:
Behind one door is a car; behind the others, goats. You pick a door, say No. 1,
and the host, who knows what's behind the doors,
opens another door, say No. 3, which has a goat.
He then says to you, "Do you want to pick door No. 2?"
Is it to your advantage to switch your choice?



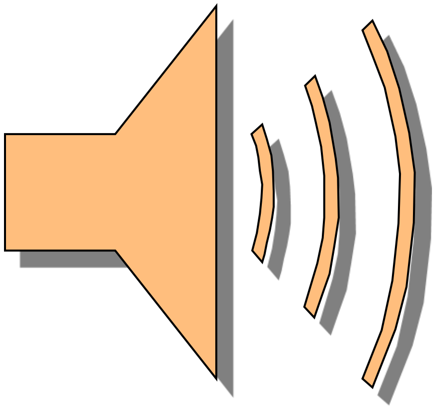
1「変えるほうがよい」、
「変えないほうがよい」
「どちらでも同じ」

2 その理由

を書いてください。

マリリンの答え

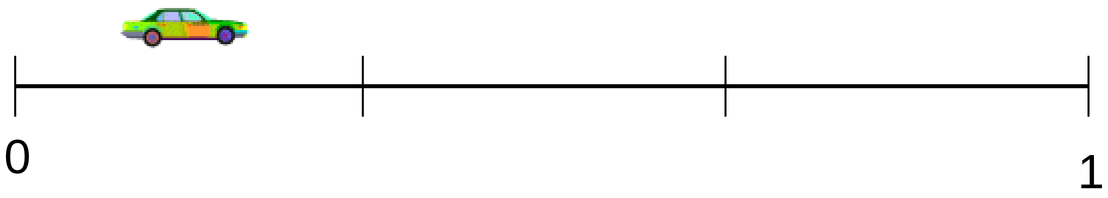
"Ask Marilyn" column in *Parade* magazine in 1990
Marilyn Vos Savant answered
arguing that the selection should be switched to door #2
because it has a 2/3 chance of success, while door #1 has just 1/3.



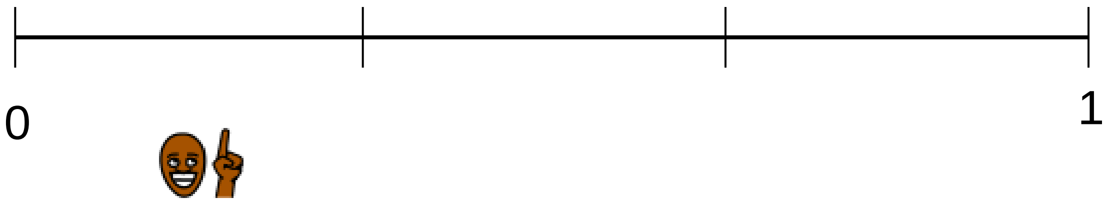
読者の反応

When the the Monty Hall problem appeared in *Parade*,
approximately 10,000 readers,
including nearly 1,000 with PhDs,
wrote to the magazine
claiming the published solution was wrong.
（nearly all arguing
doors #1 and #2 each have an equal chance of success.）

自動車を置く
Rnd()



ドアを選ぶ
Rnd()



当たっているとき、乱数で残りのドアを選び開ける。
はずれているとき、選んでいなくて自動車のないドアを開ける

選択を変えなかったとき、何パーセントの人が当たりだったか。
または、選択を変えたとき、何パーセントの人が当たりだったか。

```
'自動車を置く
        CSet = Rnd()
        If CSet < 1 / 3 Then
            doorC(0) = 1
        ElseIf CSet < 2 / 3 Then
            doorC(1) = 1
        Else
            doorC(2) = 1
        End If
 'ドアを選ぶ
        DCoose = Rnd()
        If DCoose < 1 / 3 Then
            doorD(0) = 1
        ElseIf DCoose < 2 / 3 Then
            doorD(1) = 1
        Else
            doorD(2) = 1
        End If
```

```
'置き位置と選択位置を見つける
        For j = 0 To 2
            If doorC(j) = 1 Then Exit For
        Next
        For k = 0 To 2
            If doorD(k) = 1 Then Exit For
        Next
```

```
'ドアを開ける
        If j = k Then
            If Rnd() < 1 / 2 Then
                doorC((j + 1) Mod 3) = 2
            Else
                doorC((j + 2) Mod 3) = 2
            End If
            If RadioButton2.Checked = True Then
                '選択を変えないとすると当たりなので
                cnt = cnt + 1
            Else
                '選択を変えるとするとはずれなので
                cnt = cnt
            End If
        Else
            doorC(3 - (j + k)) = 2
            If RadioButton2.Checked = True Then
                '選択を変えないとするとはずれなので
                cnt = cnt
            Else
                '選択を変えるとすると当たりなので
                cnt = cnt + 1
            End If
        End If
```

あたった人の数を表わす

シミュレーション実行

1 デフォルト
2 司会者に癖のある場合、癖を考慮する
3 考慮しないで選択を変更したときとの比較

1-2) 赤胴鈴之助問題

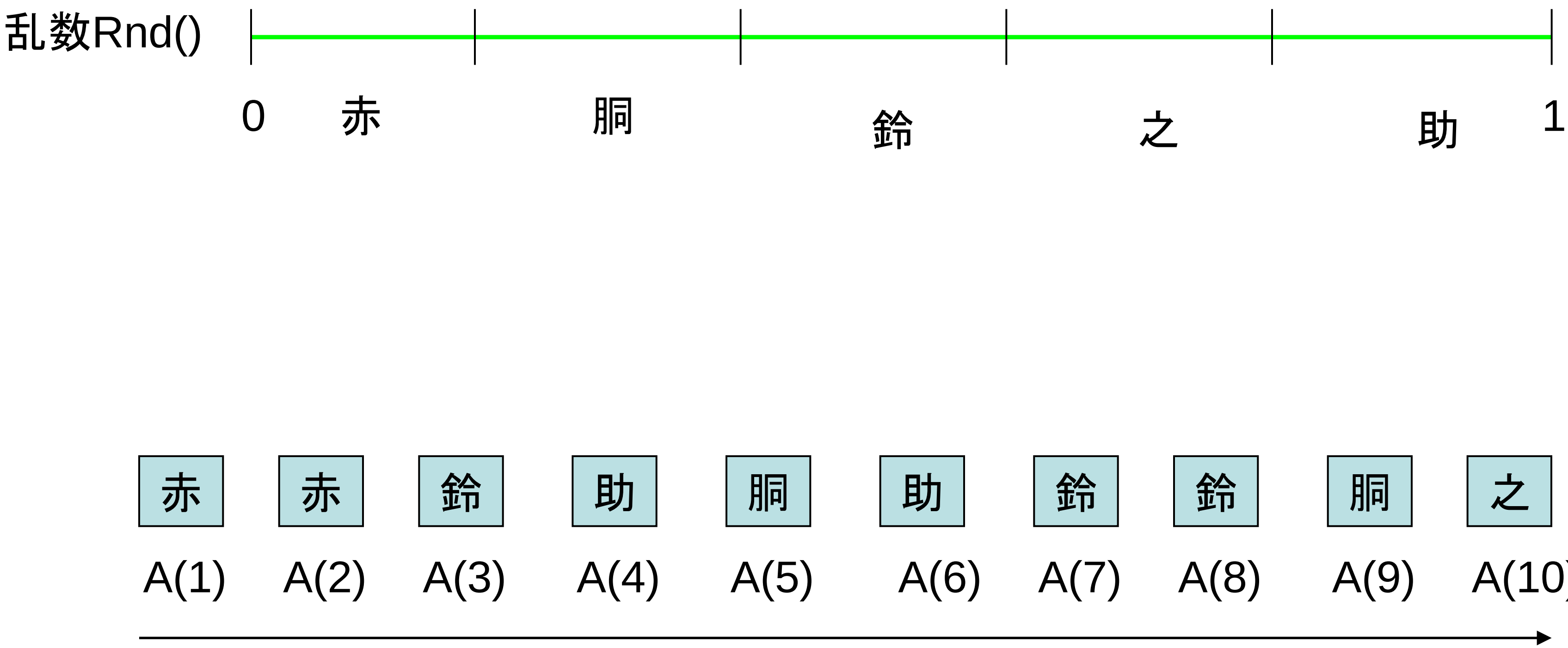
赤胴鈴之助問題とは

　キャラメルを1箱買うと赤胴鈴之助のどれか１文字が箱の中にはいっている。赤胴鈴之助の5文字を集めると賞品がもらえる。そこで、賞品を手にいれるためには平均何箱買わなければならないか。
(各文字は等確率で入っているとする）

皆さんの予想は？
箱数となぜそう思うかを書いてください。

シミュレーションをやってみよう

1 デフォルト
2 人数を10倍にする
3 赤の割合を0.02に変更する


乱数Rnd()
0
赤
胴
鈴
之
助
1
赤
赤
鈴
助
胴
助
鈴
鈴
胴
之
A(1)
A(2)
A(3)
A(4)
A(5)
A(6)
A(7)
A(8)
A(9)
A(10)

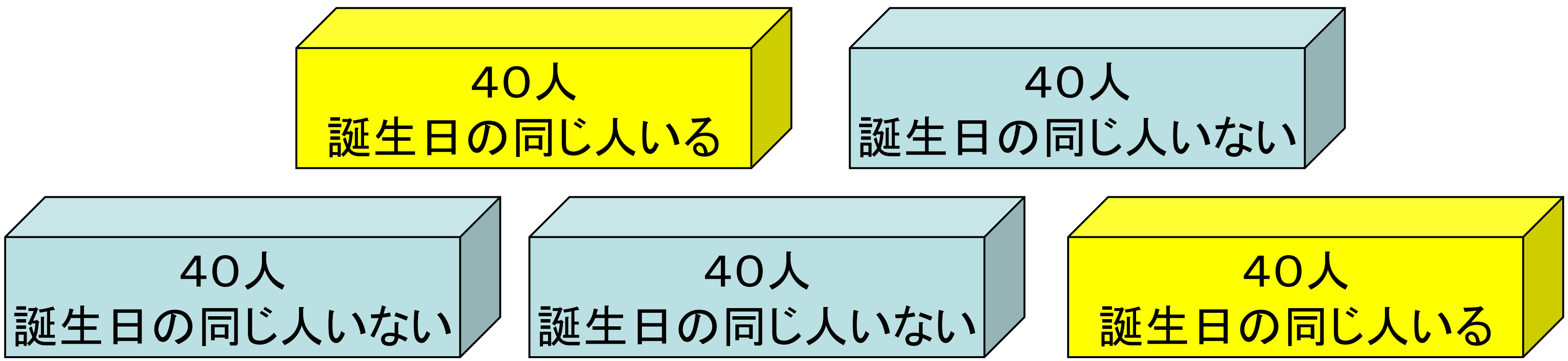
```
R = Rnd()
Call Haitteitamono(t, A, R, R1, R2, R3, R4)
```

買ったキャラメルの数

```
Sub Haitteitamono(ByVal t, ByVal A(), ByVal R, ByVal R1, ByVal R2, ByVal R3, ByVal R4)
        '乱数RによりA()を"赤""胴""鈴""之""助"に分類
        If R < R1 Then
            A(t) = 1 '"赤"
        Else
            If R < R1 + R2 Then
                A(t) = 2  '"胴"
            Else
                If R < R1 + R2 + R3 Then
                    A(t) = 3 '"鈴"
                Else
                    If R < R1 + R2 + R3 + R4 Then
                        A(t) = 4 '"之"
                    Else
                        A(t) = 5 '"助"
                    End If
                End If
            End If
        End If
End Sub
```

1-3) 誕生日が同じ人はどれぐらいいる?

学生40人のクラスで、生まれた月も日も同じという
人は どれぐらいの確率でいるのだろうか。



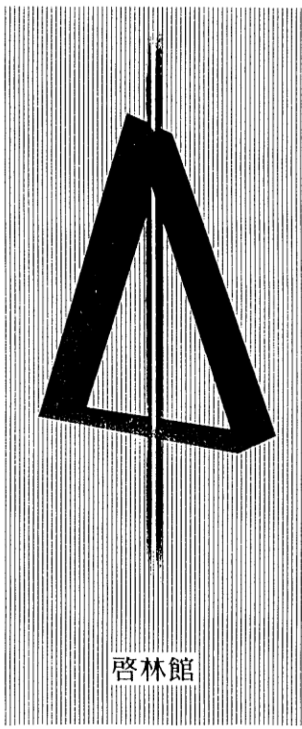
皆さんの予想は？

何パーセントと書いてください。
できれば、なぜそう思うかも書いてください。

61 啓林館 | 数 I 628 文部省検定済教科書 高等学校数学科用

高等学校 数学 I 改訂版

山本芳彦 編



発展 誕生日の同じ人がいる確率

高校の１学級のわずか40人くらいの中に生まれた月も日も全く同じ人がいるというのは，珍しいことのように思われる。ところが実際に調べてみると，意外にそういう人がいるものである。

このことを，確率の面から調べてみよう。

生まれた年はうるう年ではないものとし，１年のどの日に生まれることも同様に確からしいとすると，40人の生徒の誕生日のあり方は，全部で 365^{40} 通りある。

このうち，生徒の誕生日がすべて異なる場合の数は，${}_{365}P_{40}$ 通りである。

したがって，40人の生徒の誕生日がすべて異なる確率は，

$$\frac{{}_{365}P_{40}}{365^{40}} = \frac{365\times364\times363\times\cdots\cdots\times326}{365^{40}} \fallingdotseq 0.1088$$

よって，40人の生徒の中に，誕生日の同じ人がいる確率は，

$$1-0.1088 = 0.8912$$

となり，かなり大きいものになる。

人数をいろいろ変えて，少なくとも２人が同じ誕生日である確率を求めると右の表のようになり，人数の増加とともに１に近づいていく。

100人の場合には，この確率は

0.999999

よりも大きくなる。

人数	確 率	人数	確 率
5	0.0271	40	0.8912
10	0.1169	45	0.9410
15	0.2529	50	0.9704
20	0.4114	55	0.9863
25	0.5687	60	0.9941
30	0.7063	70	0.9992
35	0.8144	80	0.9999

生まれた年はうるう年ではないものとし，1年のどの日に生まれることも同様に確からしいとすると，40人の生徒の誕生日のあり方は，全部で 365^{40} 通りある。

このうち，生徒の誕生日がすべて異なる場合の数は，${}_{365}P_{40}$ 通りである。

したがって，40人の生徒の誕生日がすべて異なる確率は，

$$\frac{{}_{365}P_{40}}{365^{40}} = \frac{365 \times 364 \times 363 \times \cdots\cdots \times 326}{365^{40}} \fallingdotseq 0.1088$$

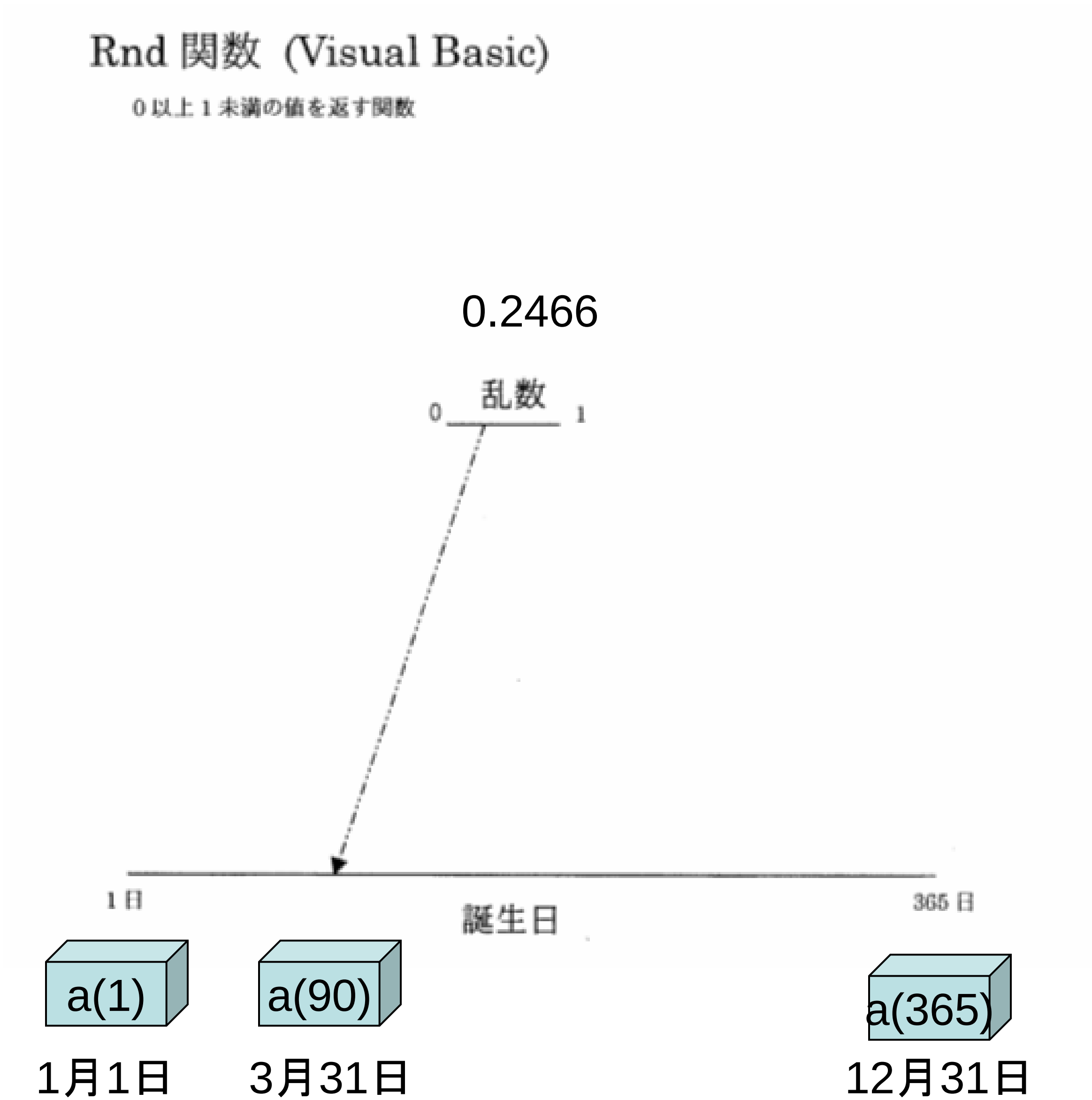
よって，40人の生徒の中に，誕生日の同じ人がいる確率は，

$$1 - 0.1088 = 0.8912$$

となり，かなり大きいものになる。

シミュレーションをしてみよう

1　デフォルト
2　人数を50人に変更する
3　人数を3人に変更する
4　クラス数を100倍にする

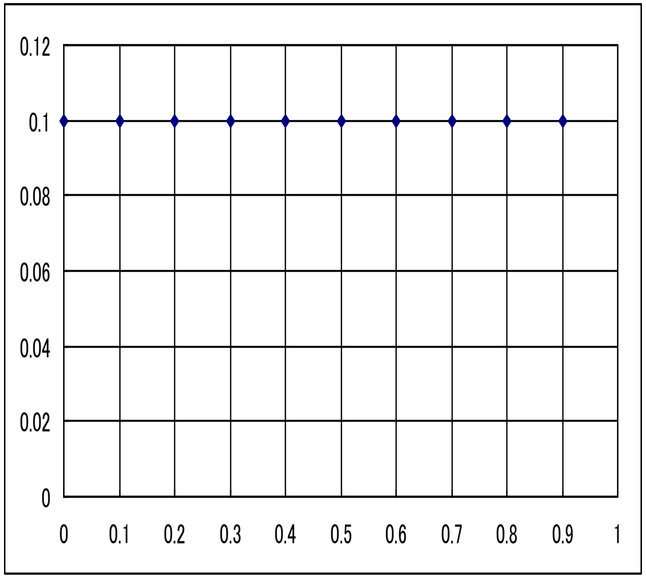

Rnd 関数 (Visual Basic)
0 以上 1 未満の値を返す関数
0.2466
乱数
0
1
1 日
誕生日
365 日
a(1)
a(90)
a(365)
1月1日
3月31日
12月31日

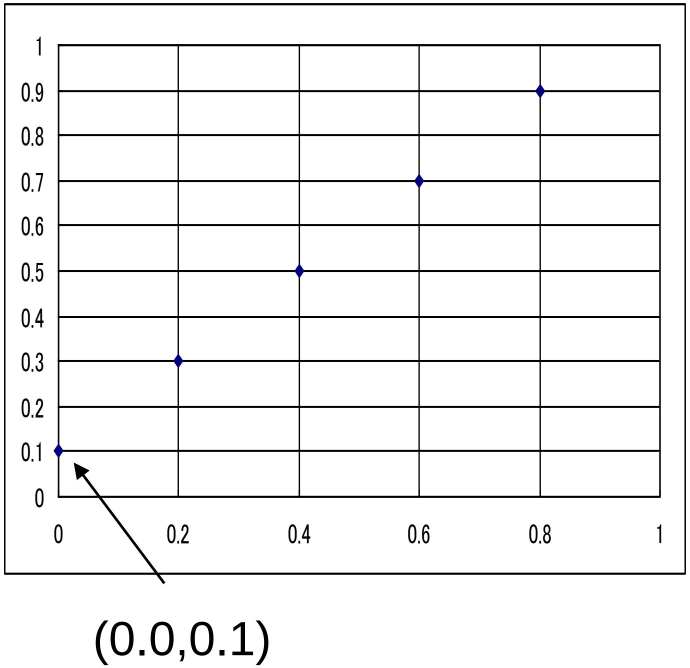
```
cnt = 0                          '同じ誕生日の人がいるクラス数を数えるカウンター
       For j = 1 To numOfClass
          'j番目のクラスの人の誕生日をシミュレートする
          '365日のどの日にも誕生日の人はいないと初期化する
          For i = 1 To 365
             a(i) = 0          '第i日目の誕生日の人の数
          Next i

          Randomize()
          For k = 1 To ninzu
             r = Rnd() * 365 + 1     'k番目の人の誕生日をrにより決定する
             nr = Int(r)          'k番目の人の誕生日はnr日目
             a(nr) = a(nr) + 1   'nr日目の誕生日の人の数をカウントする
             If a(nr) = 2 Then   'もしすでに同じ誕生日の人がいれば
                cnt = cnt + 1   'j番目のクラスには同じ誕生日の人がいるということで、そのクラス数を増やす
                GoTo Label      '(j+1)番目のクラス人の誕生日をシミュレートするステートメントに移る
             End If
          Next k
Label:
       Next j
x = cnt / numOfClass
```

2) 乱数(Random Numbers)について

乱数とはでたらめな数列のこと

でたらめ=出現がほぼ同じ割合（一様乱数）＋規則性がない





乱数列　0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, の分布ヒストグラムとペア打点図

2-1) 一様乱数の作り方とその性質

初期値　　係数　　割る数

1
1 × 23を100000001 で割ったあまり 23
23 × 23を100000001 で割ったあまり 529
529 × 23を100000001 で割ったあまり 12167
12167 × 23を100000001 で割ったあまり 279841
279841 × 23を100000001 で割ったあまり 6436343
6436343 × 23を100000001 で割ったあまり 48035888
48035888 × 23を100000001 で割ったあまり 4825413
4825413 × 23を100000001 で割ったあまり 10984498
・・・・・・・・・・・・・・・・・
・・・・・・・・・・・・・・・・・
・・・・・・・・・・・・・・・・・

100000001
で割って作成

乱数列

0.000000229
0.000005289
0.000121669
0.002798409
0.064363429
0.480358825
0.048254129
0.109844928
・・・・・・・・・
・・・・・・・・・
・・・・・・・・・

レーマー式を用いた方法

算術乱数　レーマー(Lehmer)法

I_{n+1}=keisuu × I_n　modulo　M

通常$M=2^{bit}$

プログラムでは

```
II=1
For i = 1 To N
        II = II * keisuu
        II = II Mod M
        R = II / M
Next i
```

初期値 I_0	割る数 M(bit)	係数 keisuu	備考
1	100,000,001	23	レーマー式、周期(5,882,352)
1	4,096(12)	899	規則性あり、周期($1,024=2^{10}$)
1	8,388,608(23)	1,899	周期($2,097,152=2^{21}$)
1	2,147,483,648(31)	65,539 (48828125)	標準

参考:Visual Basic の Rnd

```
static long x=327680;
float Rnd(void)
{
x=(x*16598013+12820163)%16777216;
return x*(1.0/16777216.0);
}
```

24ビット線形合同法(周期は2^{24}＝16777216。精度24ビット)。
Randomize は与えられた種を16ビットの整数に変換して初期化（2^{16}=65536の系列）

一様乱数の性質

1 一様性、規則性、初期値(1, 100000001, 23:レーマー式)で
2 周期、初期値(1, 4096, 899)で
3一様性、規則性(1, 8388608, 1899)で
4一様性、規則性 Rnd()で

2-2) 様々な確率分布の作成とその性質

連続値確率変数	2-2-1) 任意区間[a,b)の一様分布 2-2-2) 正規分布（ガウス分布） 2-2-3) 指数分布
離散値確率変数	2-2-4) ポアソン分布 2-2-5) 2項分布

どのような分布か見てみよう

1 デフォルトで順番に分布を示す
2 スライドショーを終了して、プログラムはそのまま

2-2-1) 任意区間[a,b)の一様分布

$$f(x) = \begin{cases} \dfrac{1}{b-a} & a \leq x < b \\ 0 & x < a \text{ または } x \geq b \end{cases}$$

平均 $$\mu = \int_{-\infty}^{\infty} x f(x) dx = \frac{b+a}{2}$$

標準偏差 $$\sigma = \sqrt{\int_{-\infty}^{\infty} (x-\mu)^2 f(x) dx}$$

$$= \frac{b-a}{2\sqrt{3}}$$

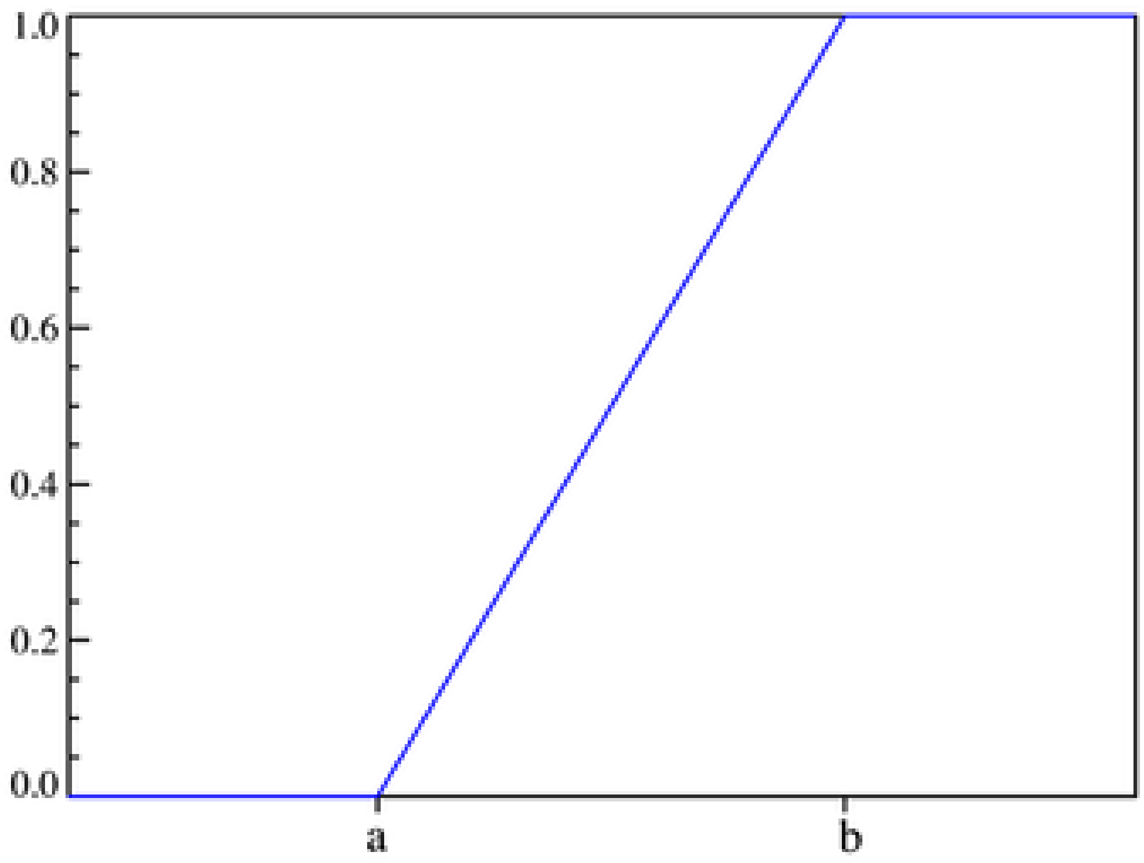
$$\begin{aligned} a &= \mu - \sqrt{3}\sigma \\ b &= \mu + \sqrt{3}\sigma \end{aligned}$$

累積分布関数
$$F(x) = \int_{-\infty}^{x} f(x)dx$$
$$= \begin{cases} 0 & x < a \\ \dfrac{x-a}{b-a} & a \le x < b \\ 1 & x \ge b \end{cases}$$

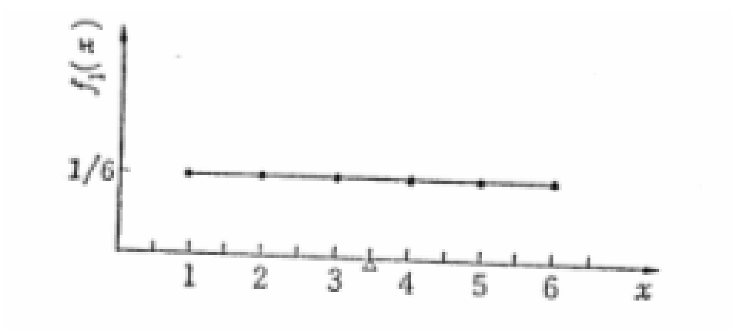
$$\mathrm{RND} = \frac{x-a}{b-a}$$

$$x = (b-a)\mathrm{RND} + a$$

累積分布関数



```
Function uniformRnd(ByVal low, ByVal high)
	uniformRnd = (high - low) * Rnd() + low
End Function
```



1個のサイコロを振ったとき出る目 の確率

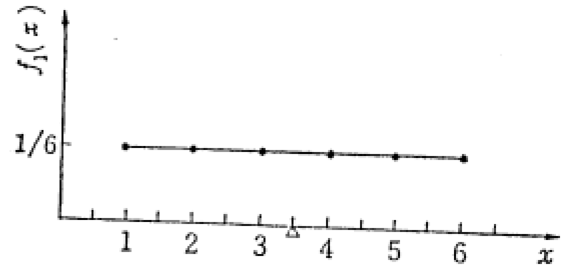
2-2-2) 正規分布（ガウス分布）

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2/2\sigma^2}$$

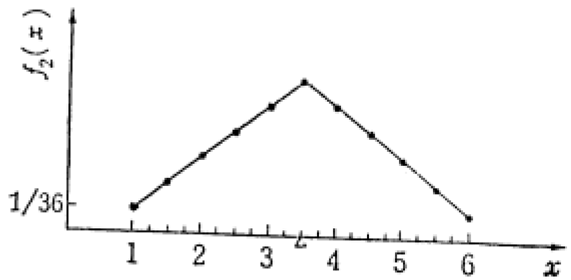
平均 $$\mu = \int_{-\infty}^{\infty} x f(x) dx$$

標準偏差 $$\sigma = \sqrt{\int_{-\infty}^{\infty} (x-\mu)^2 f(x) dx}$$

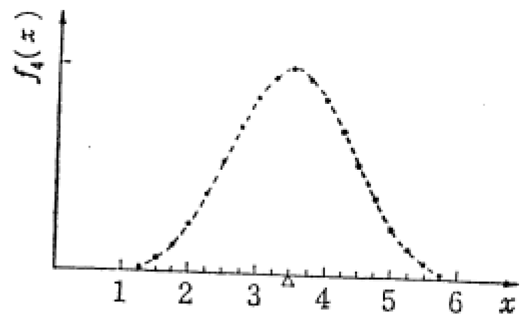
中心極限定理



1個のサイコロを振ったとき出る目の確率

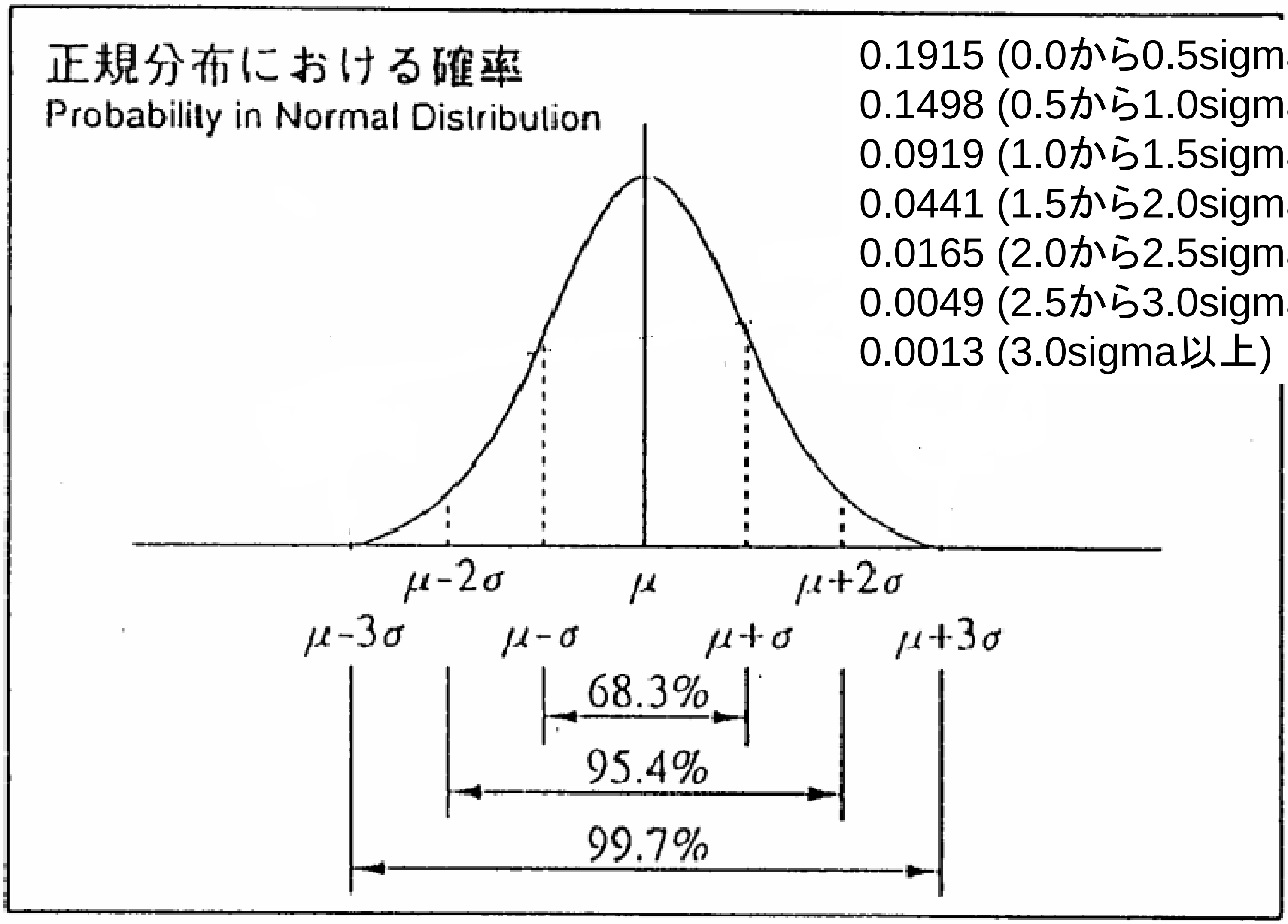


2個のサイコロを振ったとき出る目の和を
2で割った値の出る確率



4個のサイコロを振ったとき出る目の和を
4で割った値の出る確率

```
Function normalRnd(ByVal aveP, ByVal stdP)
        Dim j As Integer
        Dim r, sumNum, z As Double
        r = 0 : sumNum = 10
        For j = 1 To sumNum Step 1 : r = r + Rnd() : Next j
        z = (r - (sumNum / 2)) / Math.Sqrt(sumNum / 12)
        Return z * stdP + aveP
End Function
```



0.1915 (0.0から0.5sigma)
0.1498 (0.5から1.0sigma)
0.0919 (1.0から1.5sigma)
0.0441 (1.5から2.0sigma)
0.0165 (2.0から2.5sigma)
0.0049 (2.5から3.0sigma)
0.0013 (3.0sigma以上)

＊偏差値＝10×{（得点－平均）/標準偏差}+50

参考　Box-Muller乱数

z = Sqrt(-2 * Log(Rnd())) * Sin(2 * π * Rnd())

[平均 0, 分散 1 の正規分布]

3) 指数分布

$$f(x) = \alpha e^{-\alpha x} \qquad (\alpha > 0,\ x \geq 0)$$

平均 $$\mu = \int_0^\infty x f(x) dx = \frac{1}{\alpha}$$

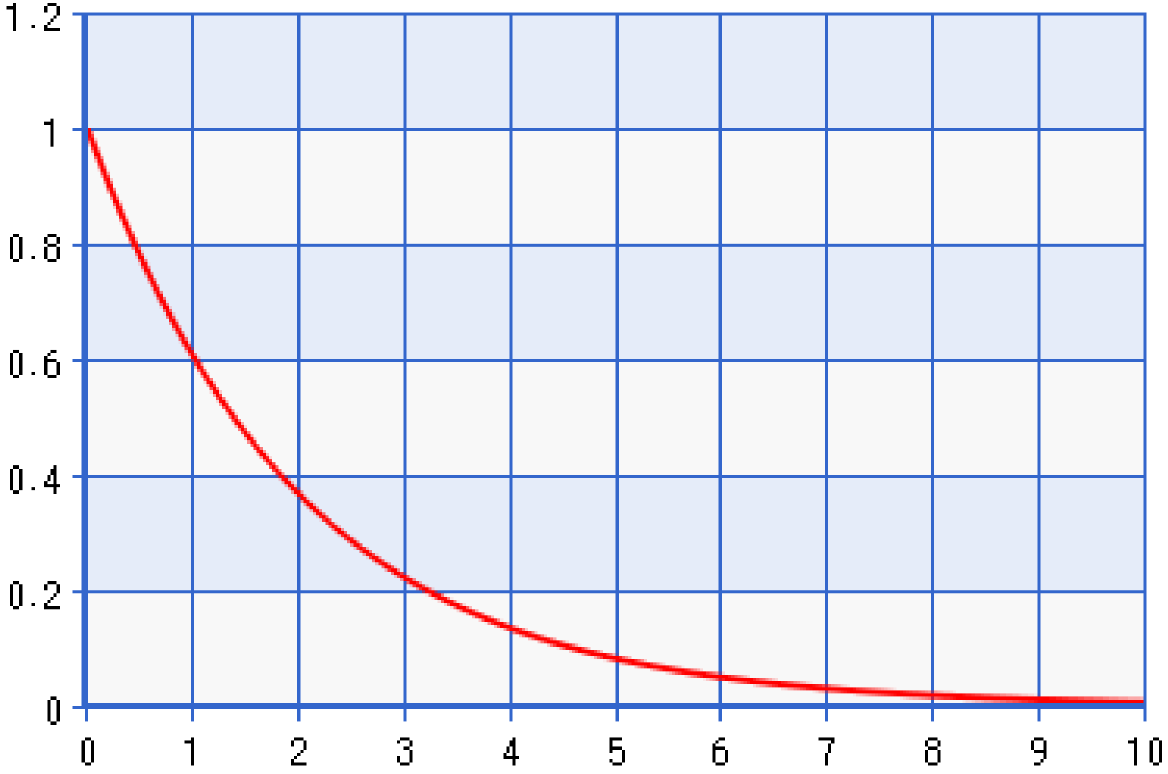
標準偏差 $$\sigma = \sqrt{\int_0^\infty (x-\mu)^2 f(x) dx} = \frac{1}{\alpha}$$

$$F(x) = \int_0^x f(x)dx = 1 - e^{-\alpha x}$$

$$\mathrm{RND} = 1 - F(x) = e^{-\alpha x}$$

$$x = -\frac{1}{\alpha}\ln(\mathrm{RND}) = -\mu \ln(\mathrm{RND})$$

1-F(x)



```
Function expRnd(ByVal aveP)
        Return -aveP * Math.Log(Rnd())
End Function
```

2-2-4) ポアソン分布

$$P(n) = \frac{e^{-\alpha}\alpha^n}{n!} \quad (\alpha > 0,\ n = 0, 1, 2, \cdots)$$

平均 $$\mu = \sum_{n=0}^{\infty} nP(n) = \alpha$$

標準偏差 $$\sigma = \sqrt{\sum_{n=0}^{\infty} (n-\mu)^2 P(n)} = \sqrt{\alpha}$$

単位時間当たり平均α人が到着するポアソン分布は
平均到着間隔　(1/α)の指数分布に従う

単位時間内にn人が到着する確率がポアソン分布に従うと仮定する。
単位時間内の平均到着人数をα人とするとT時間内の平均到着人数は
αT人

T時間内にn人が到着する確率は $$P_T(n) = \frac{e^{-\alpha T}(\alpha T)^n}{n!}$$

T時間内に一人も到着しない確率は $$P_T(0) = e^{-\alpha T}$$

ある人が到着後次の人がT時間内に到着しない確率 $$P_{notarrive} = P_T(0) = e^{-\alpha T}$$

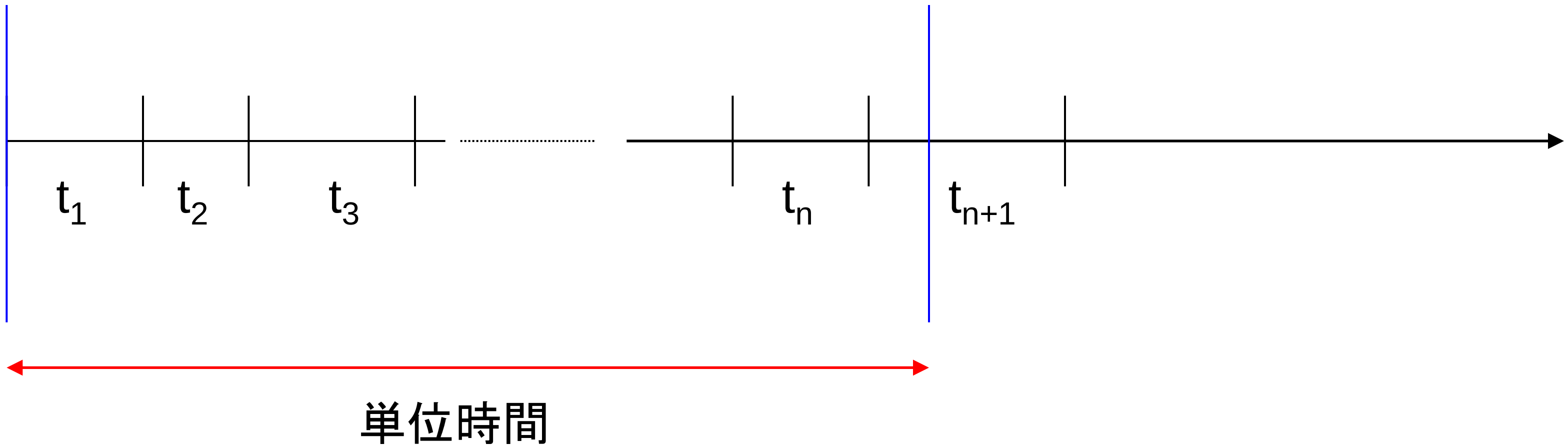
到着間隔の確率密度関数（時間 t に始めて到着がある確率）$f(t)$ を用いて

$$P_{notarrive} = 1 - \int_0^T f(t)dt$$

これより到着間隔分布は $$f(t) = \alpha e^{-\alpha t}$$

到着分布がポアソン分布のとき、到着間隔分布は指数分布に従う

t_iは指数分布をする乱数　　$$t_i = -\frac{1}{\alpha}\ln(\mathrm{RND})_i$$



$$\sum_{i=1}^{n} t_i \leq 1 < \sum_{i=1}^{n+1} t_i$$

を満たすnは単位時間当たり平均 α 人が到着するポアソン分布となる

従って

$$\sum_{i=1}^{n} \ln(\mathrm{RND})_i \geq -\alpha > \sum_{i=1}^{n+1} \ln(\mathrm{RND})_i$$

$$(\mathrm{RND})_1 \cdots (\mathrm{RND})_n \geq e^{-\alpha} > (\mathrm{RND})_1 \cdots (\mathrm{RND})_{n+1}$$

一様乱数を発生させその積が $e^{-\alpha}$ より小さくなるとき

(乱数発生回数ー1)をポアソン分布の確率変数nとする

```
Function poissonRnd(ByVal aveP)
        Dim r As Double = 1
        Dim n As Integer = 0
        Do While r > Math.Exp(-aveP)
            r = r * Rnd()
            n = n + 1
        Loop
        Return n - 1
End Function
```

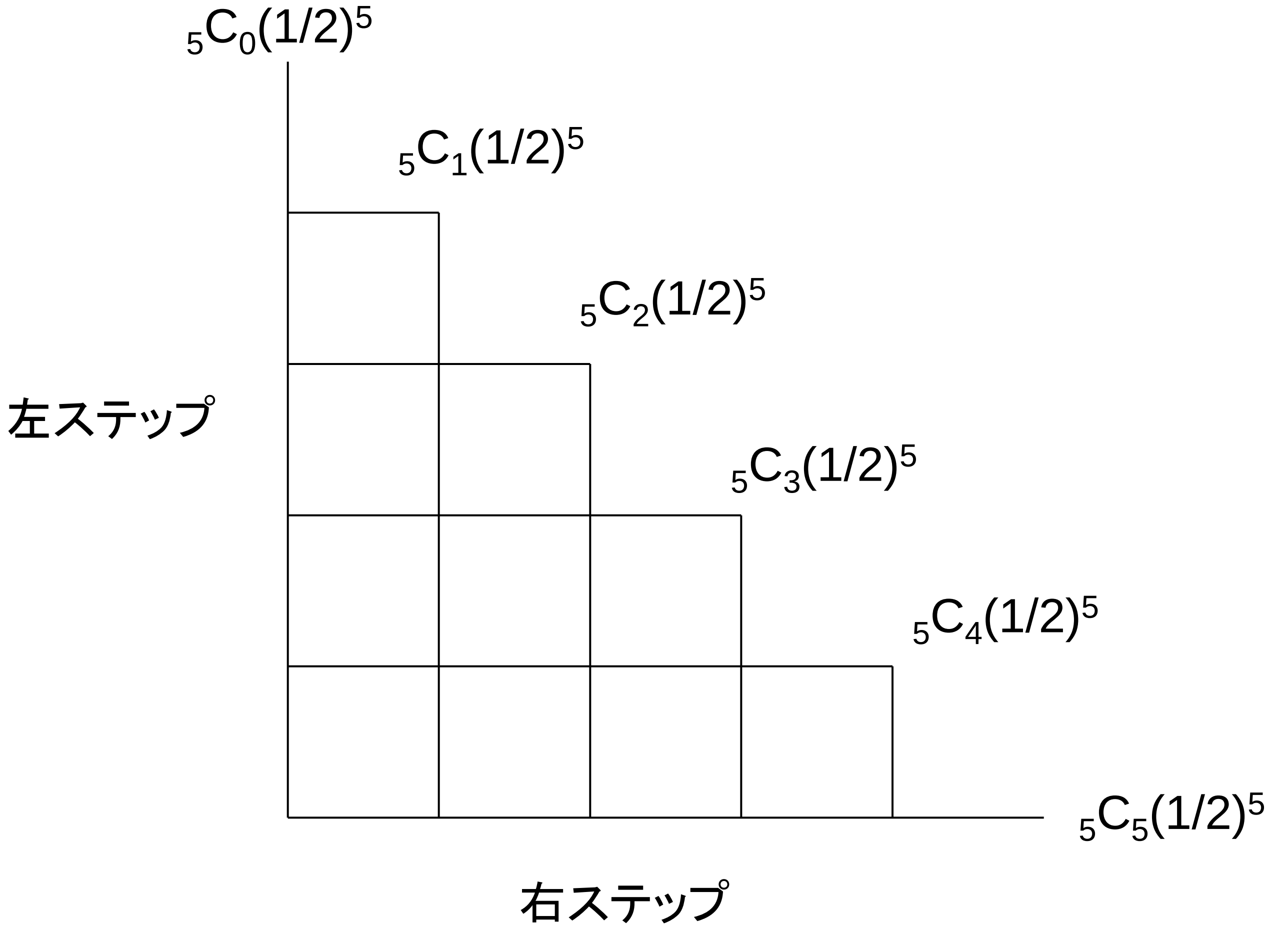
2-2-5) 2項分布

$$P(n) = \frac{1}{2^M}\frac{M!}{n!(M-n)!} \equiv \frac{{}_M C_n}{2^M} \qquad (n = 0, 1, \cdots, M)$$

$$\mu \equiv \sum_{n=0}^{M} nP(n) = \frac{M}{2}$$

$$\sigma \equiv \sqrt{\sum_{n=0}^{M}(n-\mu)^2 P(n)} = \frac{\sqrt{M}}{2}$$

1/2の確率で左または右にステップする人がMステップした場合、
nステップが右ステップである確率は2項分布に従う



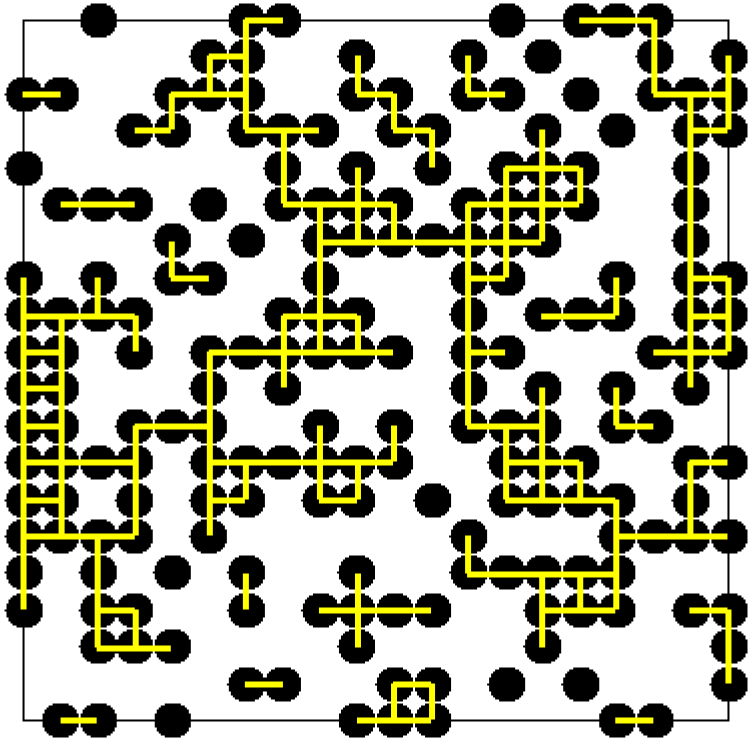
```
Function binomialRnd(ByVal aveP)
        Dim j, n, M As Integer
        n = 0
        M = 2 * aveP
        For j = 1 To M
            If Rnd() < 0.5 Then n = n + 1
        Next
        Return n
End Function
```

確率分布のシミュレーション

順番にデフォルトでシミュレーションする。
各シミュレーションで乱数発生回数を10倍する

3-1) 山火事と水漏れの問題

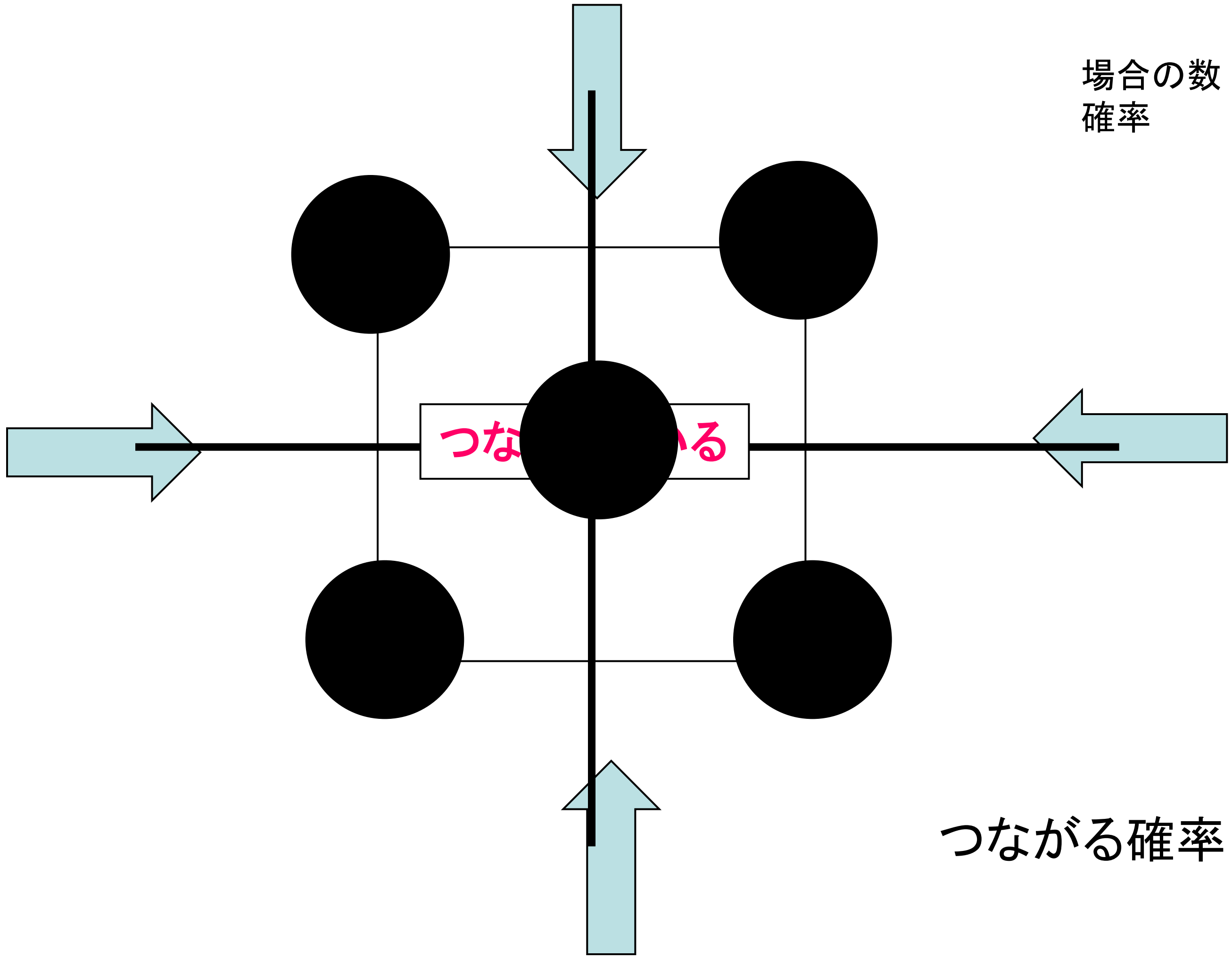
山火事の問題



山は全焼したの?

Renormalization group（くり込み群）approach

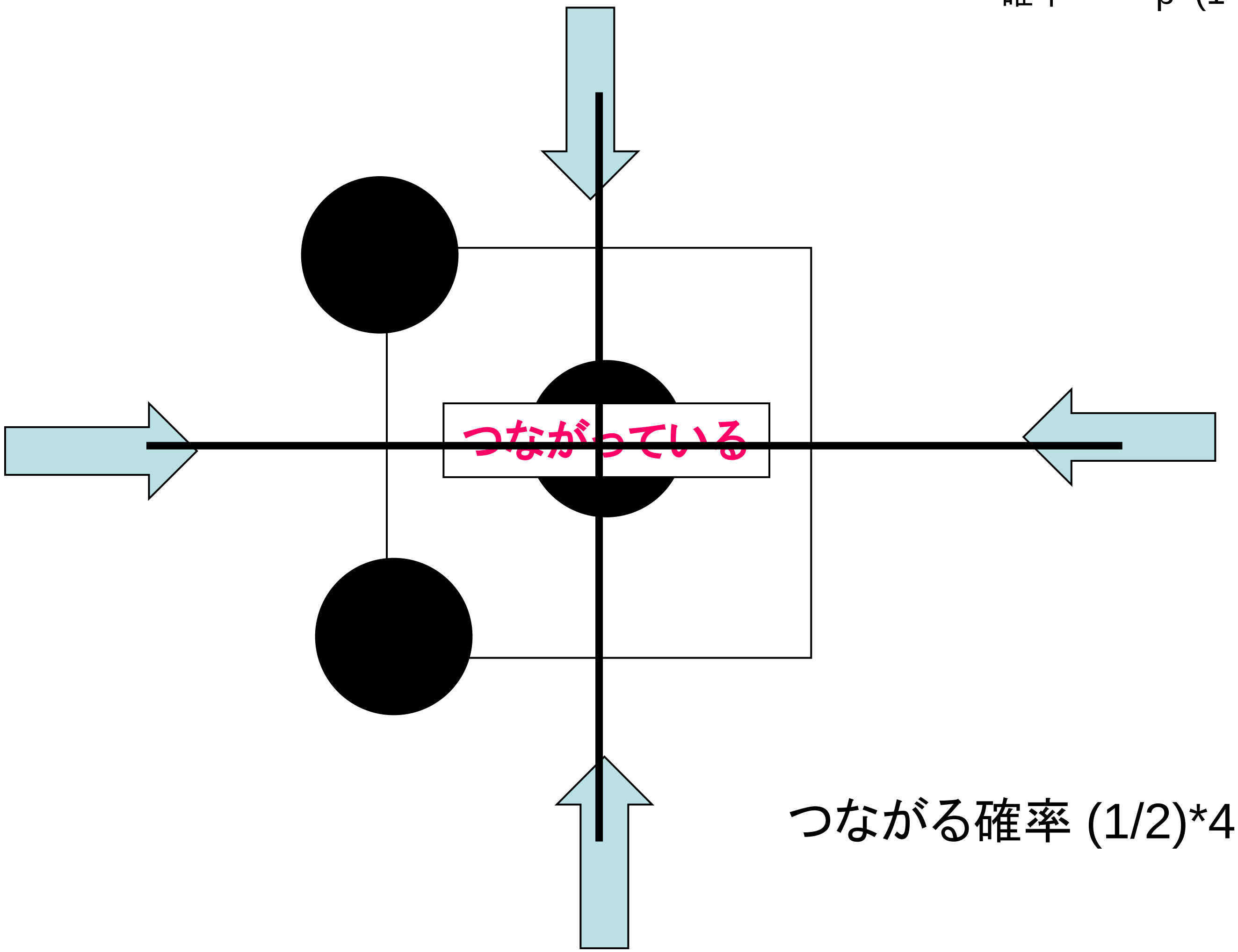
場合の数　1
確率　　　p^4



つながる確率　p^4

Renormalization group（くり込み群）approach

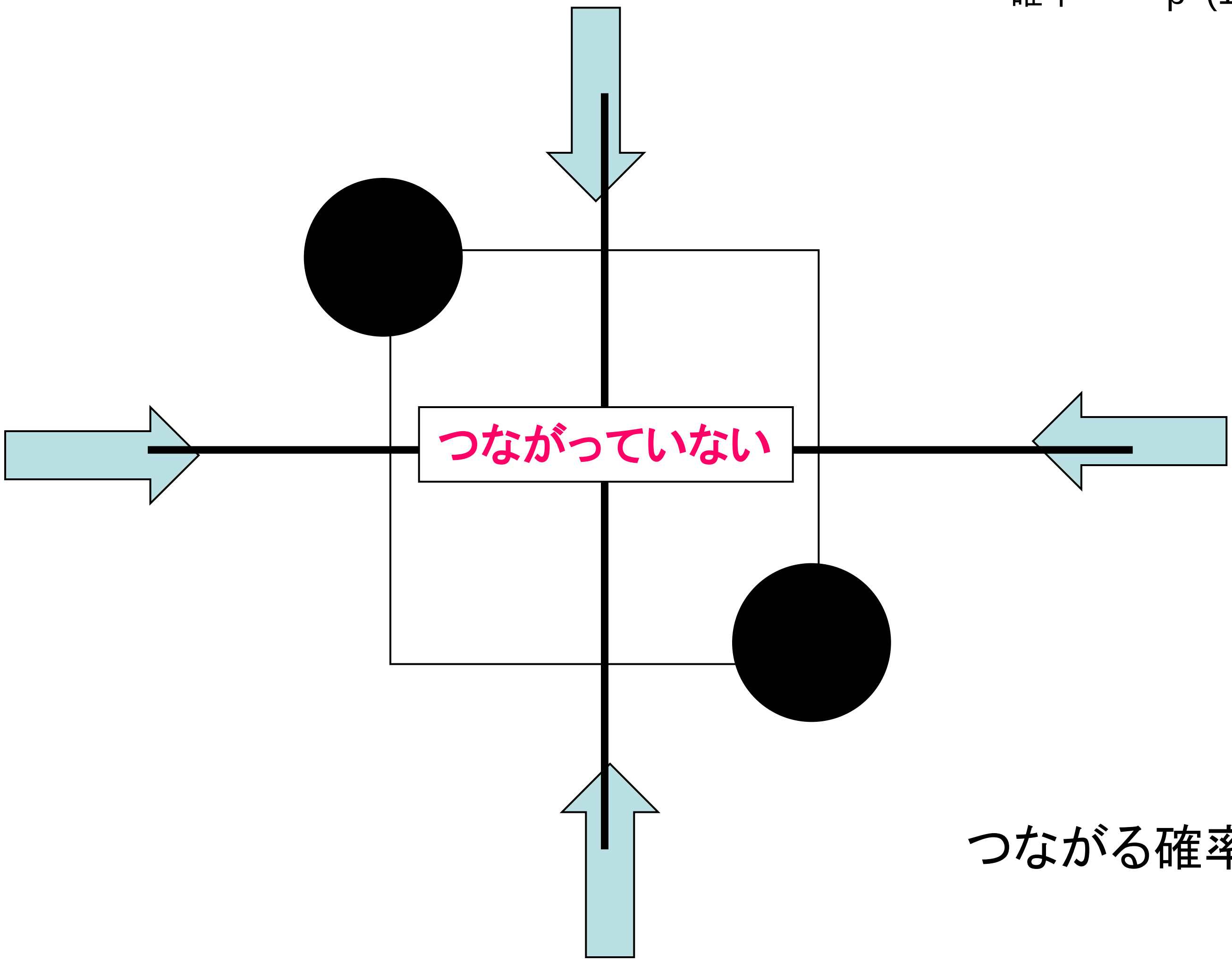
場合の数　4
確率　　　$p^2 (1-p)^2$



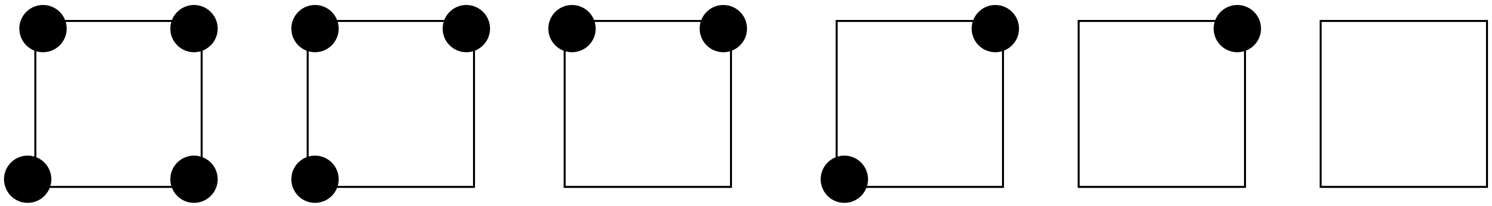
つながる確率 $(1/2)*4\ p^2\ (1-p)^2$

Renormalization group（くり込み群）approach

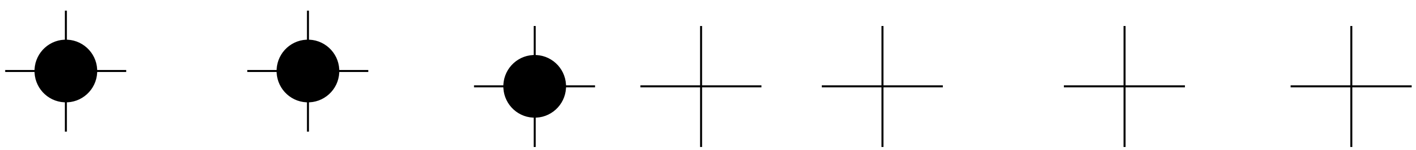
場合の数　2
確率　　　$p^2 (1-p)^2$



つながる確率 0



確率	p^4	$p^3(1-p)$	$p^2(1-p)^2$	$p^2(1-p)^2$	$p(1-p)^3$	$(1-p)^4$
場合の数	1	4	4	2	4	1



4格子を1格子とくり込みを考えた場合の確率　$p'=p^4+4p^3(1-p)+(1/2)\times 4p^2(1-p)^2$

さらに4格子を1格子とくり込みを考えた場合の確率　$p''=p'^4+4p'^3(1-p')+(1/2)\times 4p'^2(1-p')^2$

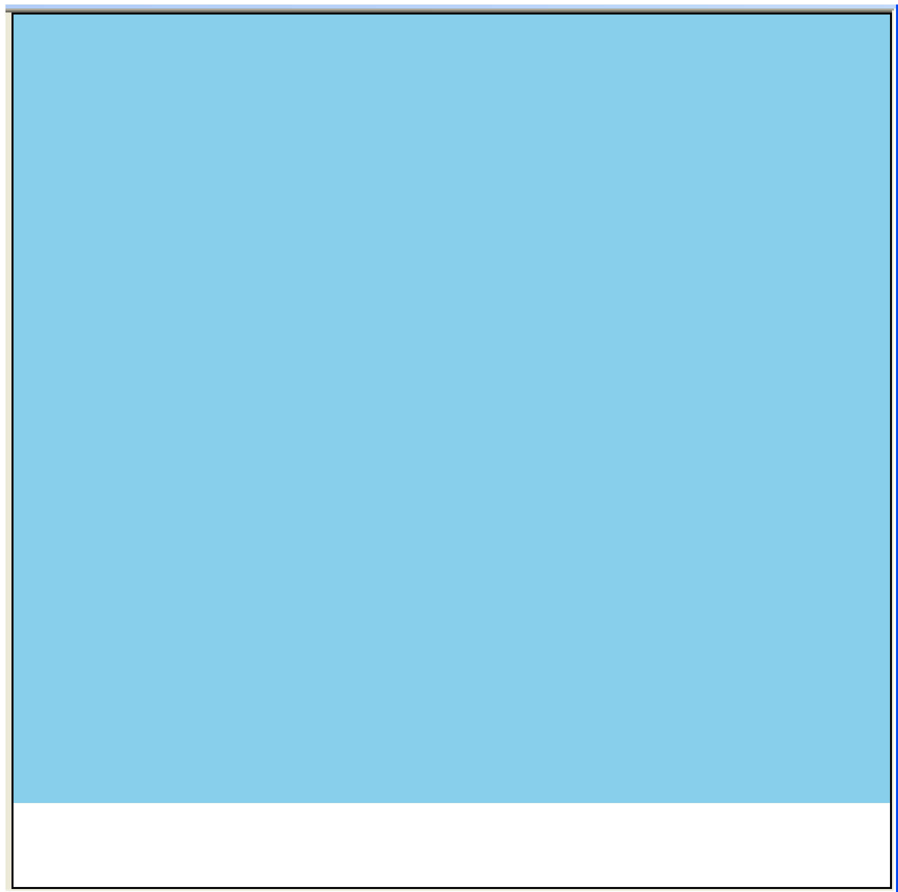
さらに ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・

$p'=p$とすると、$p=0,\ 1,\ (-1-\sqrt{5})/2,\ (-1+\sqrt{5})/2\approx$0.62

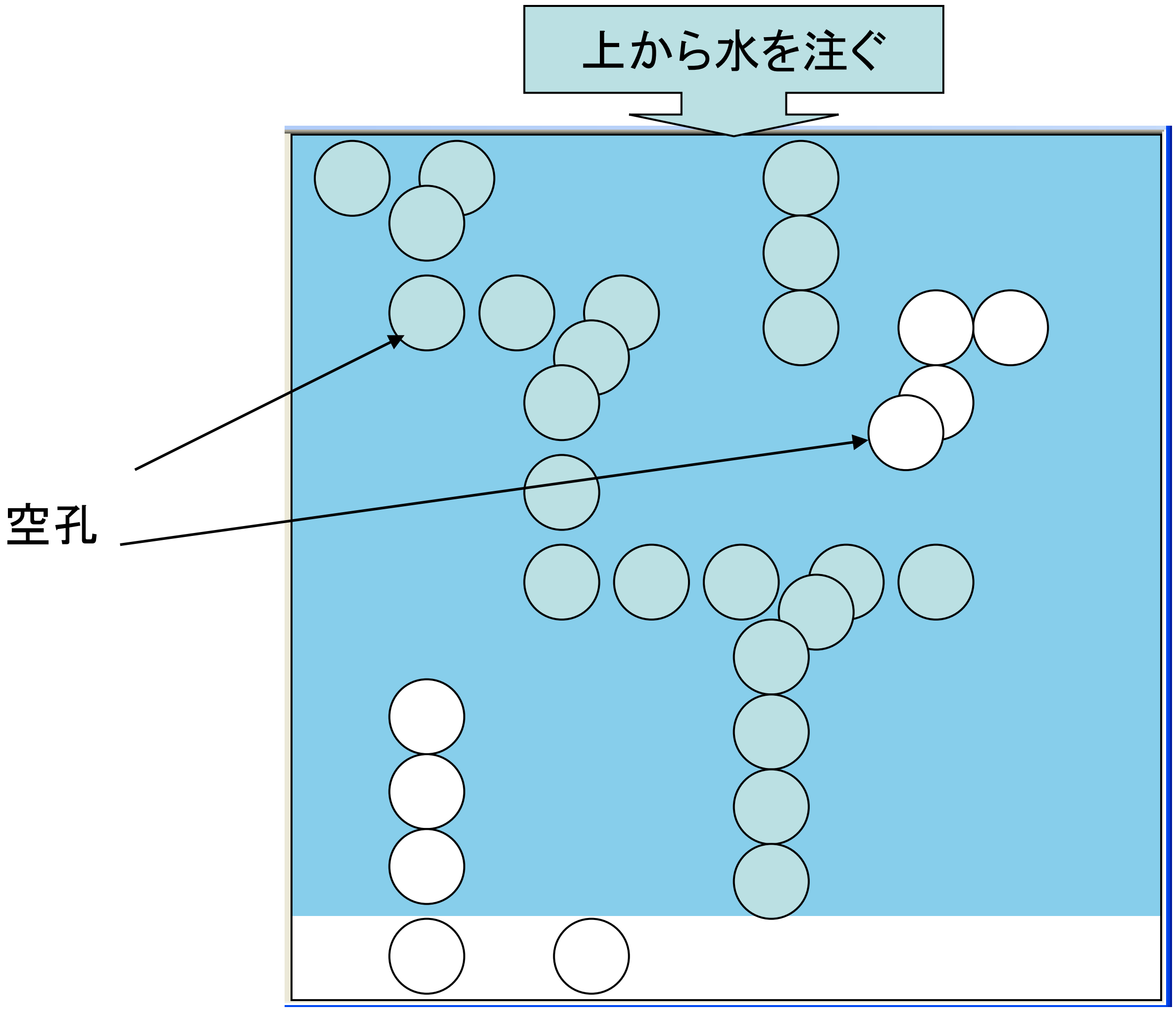
山火事のシミュレーション

1 デフォルト
2 占有確率1.0と0.1でおこなう。
3 クラスタの生成確率をデフォルトで
（0.5での変化に注意を促す）

水漏れの問題



水漏れはするのしないの?


上から水を注ぐ
空孔

水漏れのシミュレーション実行

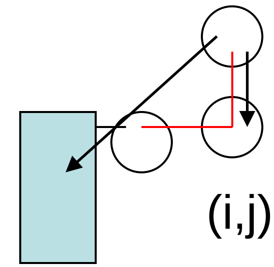
1-1 デフォルト(サイト占有確率0.5)
1-2 サイト占有確率を0.2でおこなう
2 デフォルト
3-1 デフォルト(サイト占有確率0.5)
3-2 サイト占有確率を0.2でおこなう
3-3 サイト占有確率を0.3でおこなう

MC method

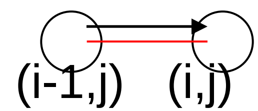
```
          prob = Val(TextBox1.Text) '占有確率

'サイトにオブジェクトの配置
          For i = 1 To size
              For j = 1 To size
                  If Rnd() <= prob Then
                      a(i, j) = 1
                  End If
              Next j
          Next i
```

2次元つながりチェック用島番号の割り当てアルゴリズム

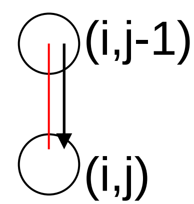
```
For i = 1 To s
    For j = 1 To s
        If a(i, j) * a(i - 1, j) = 1 And a(i, j) * a(i, j - 1) = 1 Then
            b(i, j) = b(i, j - 1)
            temp1 = b(i - 1, j)
            For k = 1 To i - 1
                For l = 1 To s
                    If b(k, l) = temp1 Then
                        b(k, l) = b(i, j - 1)
                    End If
                Next l
            Next k
            For m = 1 To j - 2
                If b(i, m) = temp1 Then
                    b(i, m) = b(i, j - 1)
                End If
            Next

        ElseIf a(i, j) * a(i - 1, j) = 1 Then
            b(i, j) = b(i - 1, j)
        ElseIf a(i, j) * a(i, j - 1) = 1 Then
            b(i, j) = b(i, j - 1)
        ElseIf a(i, j) = 1 Then
            b(i, j) = island
            island = island + 1
        End If
    Next j
Next i
```



(i-1,j)とつながるサイトすべて、および(i,j)に(i,j-1)の島番号を割り当てる



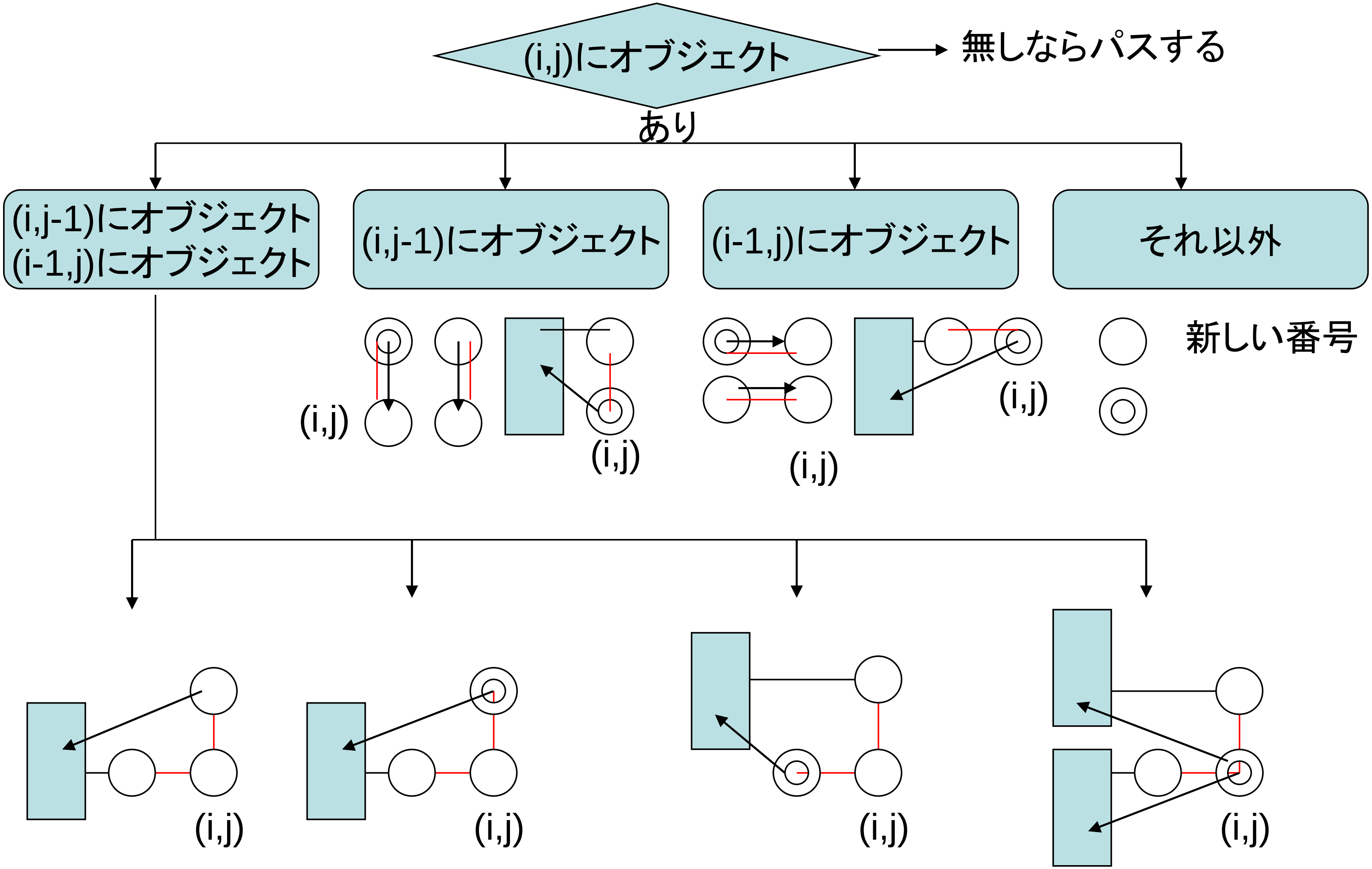
(i-1,j)の島番号を(i,j)に割り当てる



(i,j-1)の島番号を(i,j)に割り当てる

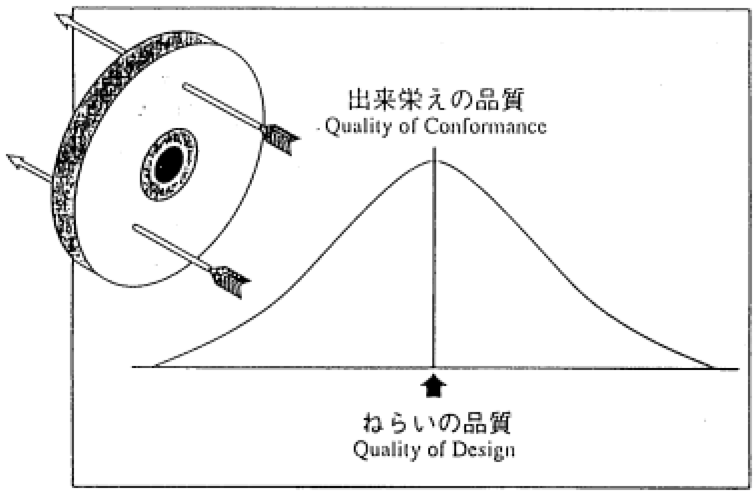
オブジェクトが置かれているがつながっていないときは新しい島番号を割り当てる

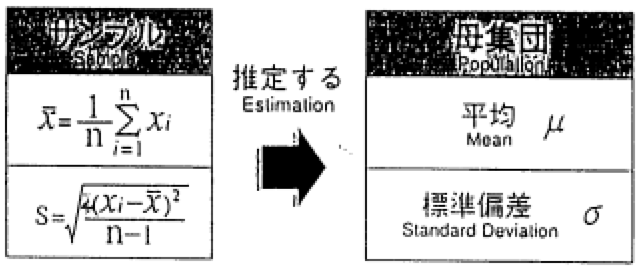
3次元つながりチェックアルゴリズム



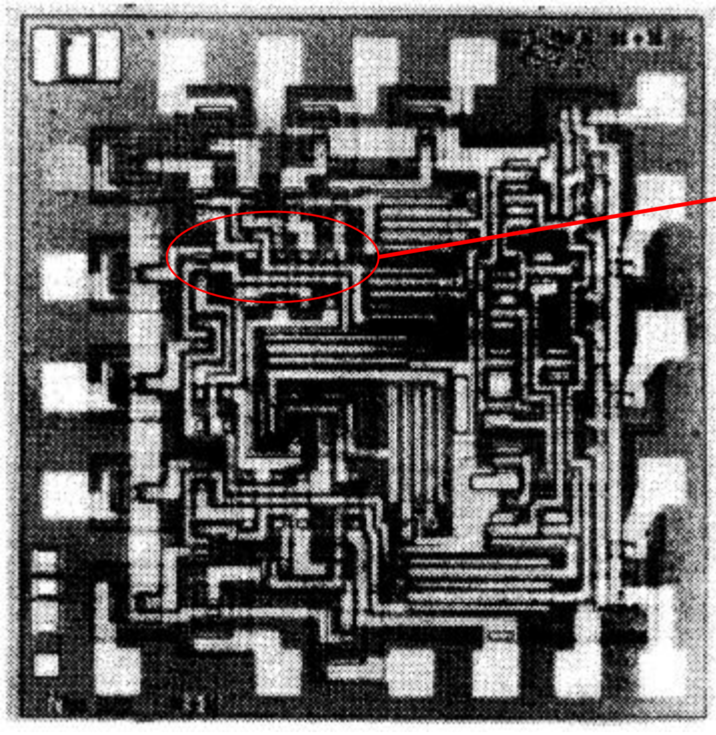
上とつながっている場合◎　上の番号を割り当てる

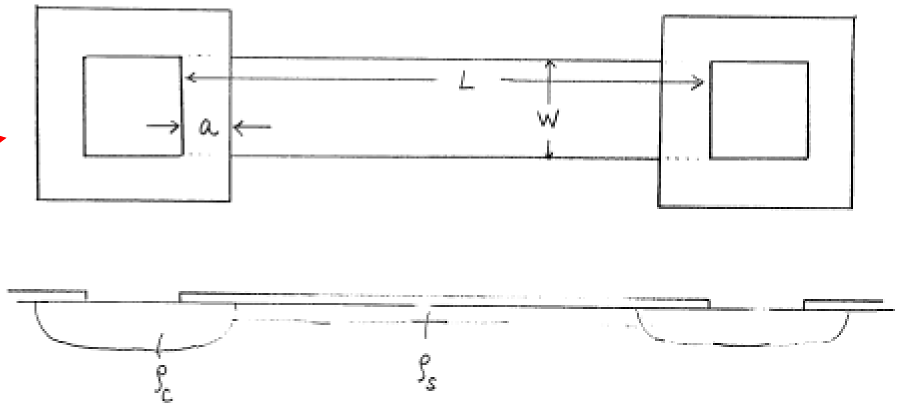
3-2) 試作データから量産の可否を判断する





（各工程のバラツキデータから、最終品の特性バラツキをシミュレーションする）

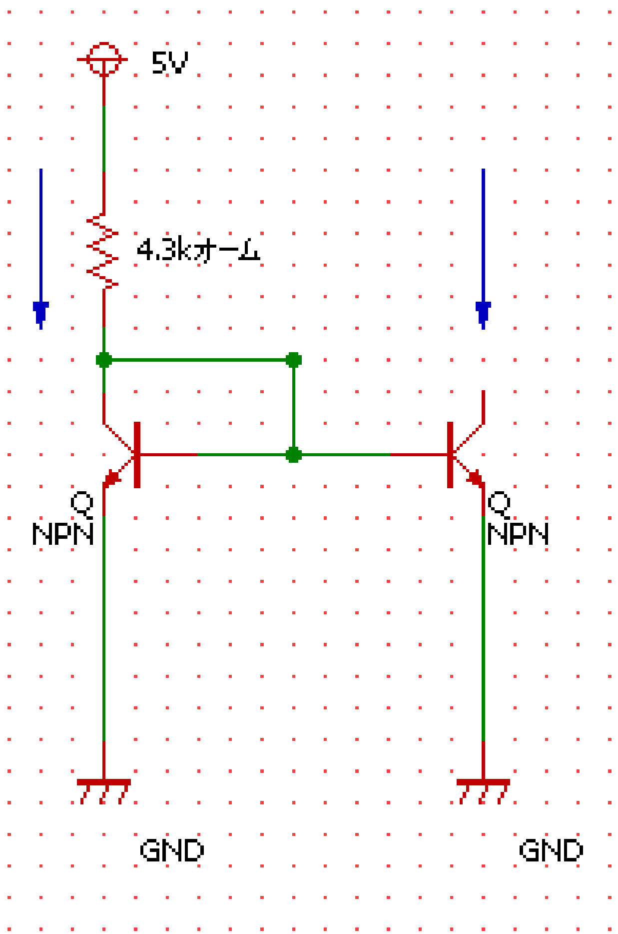




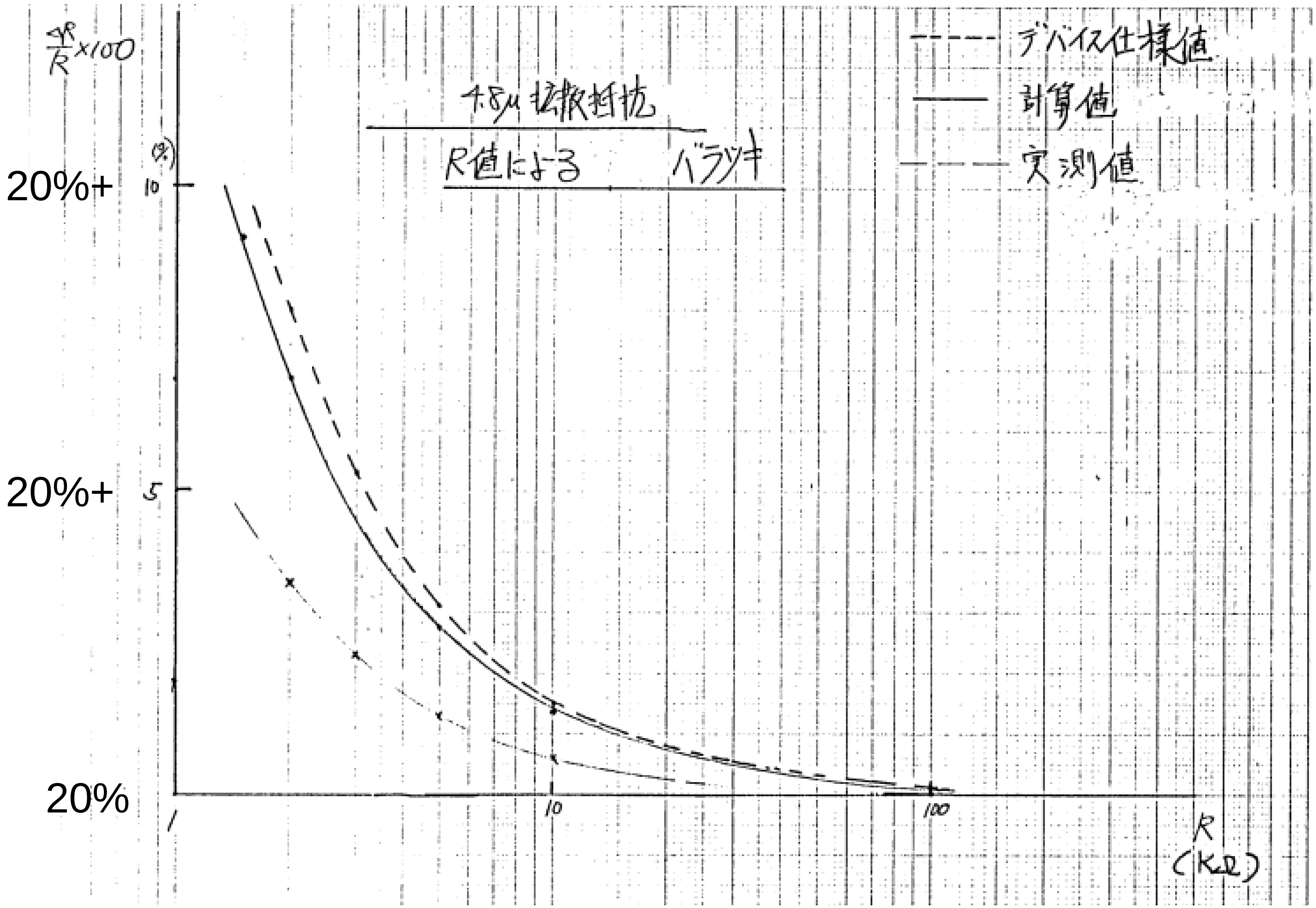
$$R = \rho_s \frac{L - 2a}{W} + \rho_c \frac{2a}{W} + 2R_c$$

$$R_c = \frac{\rho_c}{W}$$

プロセスエンジニア
バラツキはある



回路設計者
バラツキがあっては困る



$\frac{\Delta R}{R} \times 100$
4.8μ 拡散抵抗
R値による バラツキ
デバイス仕様値
計算値
実測値
(%)
20%+
10
20%+
5
20%
1
10
100
R
(KΩ)

ρ_s　ρ_c　W　a　の平均値および標準偏差がそれぞれ測定されているとき、できばえの抵抗値の平均値およびバラツキを求めよ。また下限スペック、上限スペックが与えられているときインスペックの割合はどれくらいか。

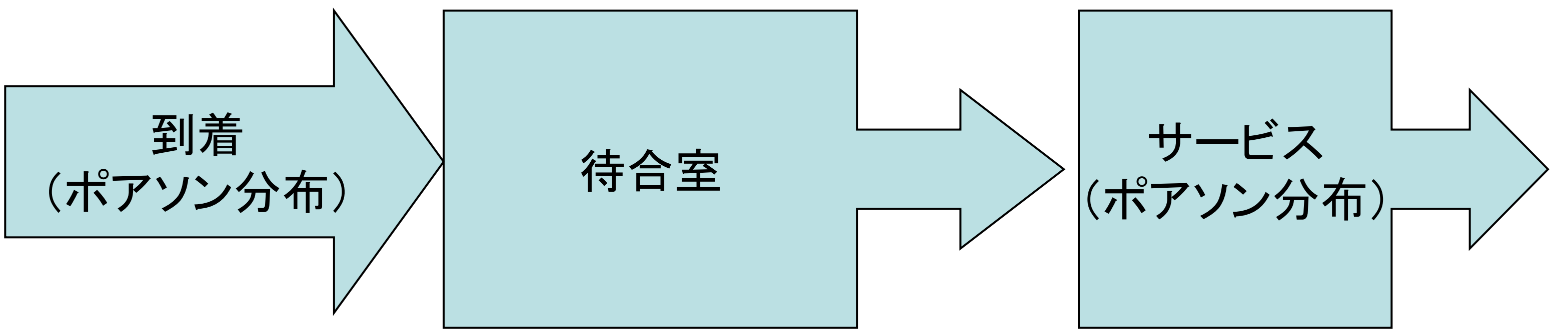
シミュレーションの実行

1) デフォルト(10k+20%)
2) 100k+20%
3) 1kオーム＋20%のバラツキ
4) 1kオーム＋30%のバラツキ

```
For i = 1 To N
        x1 = rndNorm(avex1, stdx1, RSum)  '本体シート抵抗値 rs
        x2 = rndNorm(avex2, stdx2, RSum)  'コンタクトシート抵抗値  rc
        x3 = rndNorm(avex3, stdx3, RSum)  '抵抗幅  W
        x4 = rndNorm(avex4, stdx4, RSum)  'コンタクト幅 a

        R = x1 * ((L - 2 * x4) / x3) + x2 * (2 * x4 / x3) + 2 * x2 / x3

Next i
```

3-3) 個人病院の待合室でどれぐらい待たされるの？

待ち行列

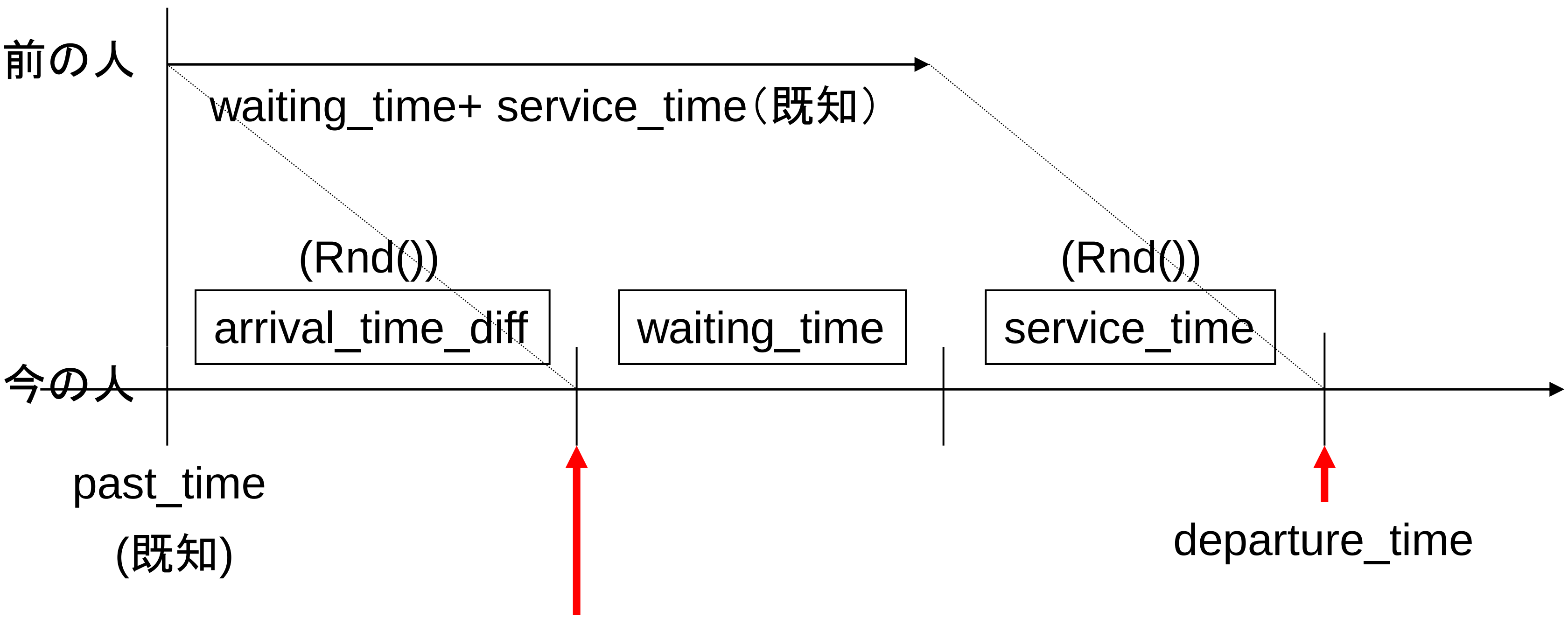


平均到着人数（人/分）

最大待ち人数　？
瞬間待ち人数　？
平均待ち人数　？

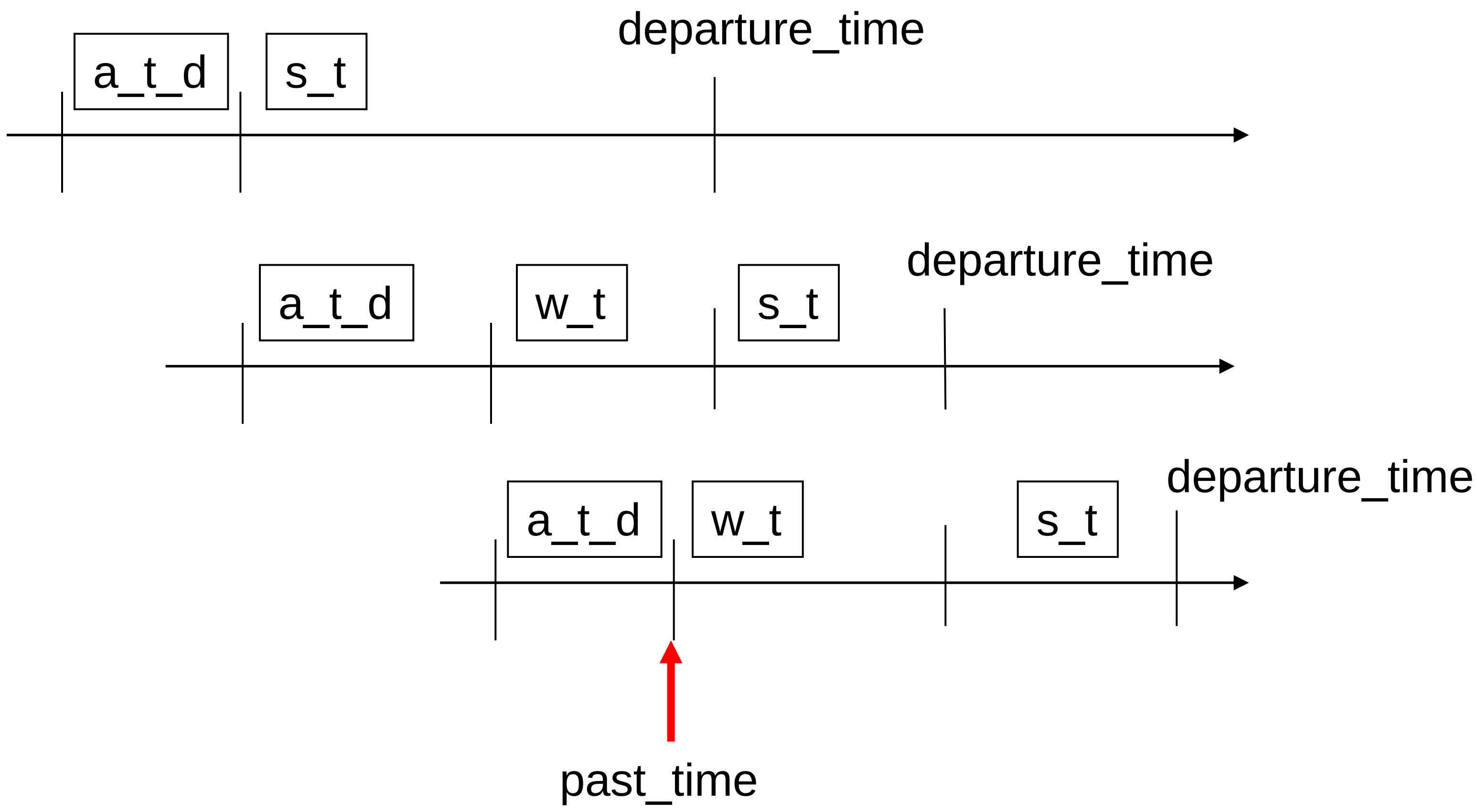
平均サービス時間(分/人)

①
ある人の到着時間(arrival_time_diff)を乱数により得る
その人のサービス時間(service_time)を乱数により得る
待ち時間(waiting_time)を計算する、waitするかしないか
[前の人のwaiting_time ＋service_timeとこの人のarrival_time_diffとの差]
立ち去る時刻(departure_time)を計算する



②経過時刻(past_time)をarrival_time_diff後に変更する

③各経過時刻における待ち人数をカウントする



シミュレーションを実行する

1 デフォルト
2 10人を超えた場合の平均到着人数を10人以内と同じにする
パンクするのでサービス時間を短くする(3分/人,2分/人)

```
'経過時間が設定したシミュレーション時間より小さい間繰り返す
Do While past_time <= TOTAL_TIME
        ' ① 来た人の待ち時間と出発時刻を計算する(a_t_ d,s_tを乱数で得る)
        coming_new_client(waiting_number)
        ' ② 来た人の時刻を経過時間とする
        past_time = past_time + waiting_client(waiting_number + 1).arrival_time_diff
        ' ③その経過時刻における待ち人数をカウントする
        line_up_client(waiting_number)
Loop
```

平均到着人数(人/分)を用いて指数分布乱数を発生

```
Sub coming_new_client(ByVal w_number)
    waiting_client(w_number + 1).arrival_time_diff = -Math.Log( Rnd() ) / COMING_TIME_AV
```

平均サービス時間(分/人)を用いて指数分布乱数を発生

```
    waiting_client(w_number + 1).service_time = -SERVICE_TIME_AV * Math.Log( Rnd() )

    '待ち時間を計算する
    '出発時刻を計算する

End Sub
```

①

```
Sub coming_new_client(ByVal w_number)
        waiting_client(w_number + 1).arrival_time_diff = -Math.Log( Rnd() ) / COMING_TIME_AV
        waiting_client(w_number + 1).service_time = -SERVICE_TIME_AV * Math.Log( Rnd() )

       '待ち時間を計算する
       waiting_client(w_number + 1).waiting_time = waiting_client(w_number).waiting_time + _
                                            waiting_client(w_number).service_time - _
                                            waiting_client(w_number + 1).arrival_time_diff

      If waiting_client(w_number + 1).waiting_time < 0.0 Then '待たない場合
          waiting_client(w_number + 1).waiting_time = 0.0
          waiting_client(w_number + 1).wait_time_status = False
       Else                                                '待つ場合
          waiting_client(w_number + 1).wait_time_status = True
       End If
       '出発時刻を計算する
       waiting_client(w_number + 1).departure_time = past_time + _
                                              waiting_client(w_number + 1).arrival_time_diff + _
                                              waiting_client(w_number + 1).waiting_time + _
                                              waiting_client(w_number + 1).service_time
 End Sub
```

③

```
Sub line_up_client(ByVal w_number)
        Dim left_client_num As Integer
        left_client_num = 0
        If waiting_client(w_number).wait_time_status = True Then
            left_client_num = count_left_client_num(w_number)
            waiting_client_move_over(w_number, left_client_num)
        Else
            waiting_client_move_over(w_number, w_number + 1)
        End If
End Sub
Function count_left_client_num(ByVal w_num)
        Dim i, left_cl_num As Integer
        left_cl_num = 0
        For i = 0 To w_num
            If waiting_client(i).departure_time < past_time Then
                left_cl_num = left_cl_num + 1
            End If
        Next i
        count_left_client_num = left_cl_num
End Function
Sub waiting_client_move_over(ByVal w_num, ByVal left_cl_num)
        Dim i As Integer
        For i = 0 To w_num + 1 - left_cl_num
            waiting_client(i) = waiting_client(i + left_cl_num)
        Next i
        For i = 1 To left_cl_num
            waiting_client(i + w_num + 1 - left_cl_num) = no_waiting_client
        Next i
        waiting_number = waiting_number - left_cl_num + 1
End Sub
```

4 固体内原子拡散のシミュレーション

Emperical transport law (Fick's first law)

$$\mathrm{J} = -D\nabla n$$

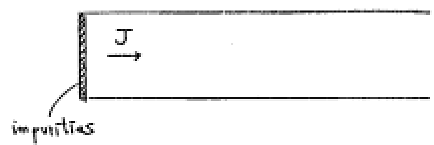
Conservation of matter (continuity equation)

$$\frac{\partial n}{\partial t} = -\nabla \cdot \mathrm{J}$$

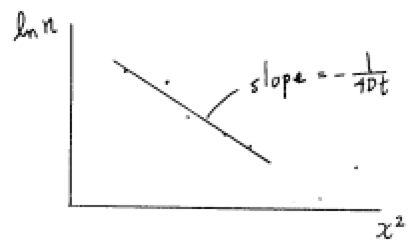
Combine both, then

$$\frac{\partial n}{\partial t} = D\nabla^2 n \quad .$$

Diffusion experiments are analized in terms of suitable solution of $n(\mathrm{r}, t)$ for the particular geometrical conditions used.

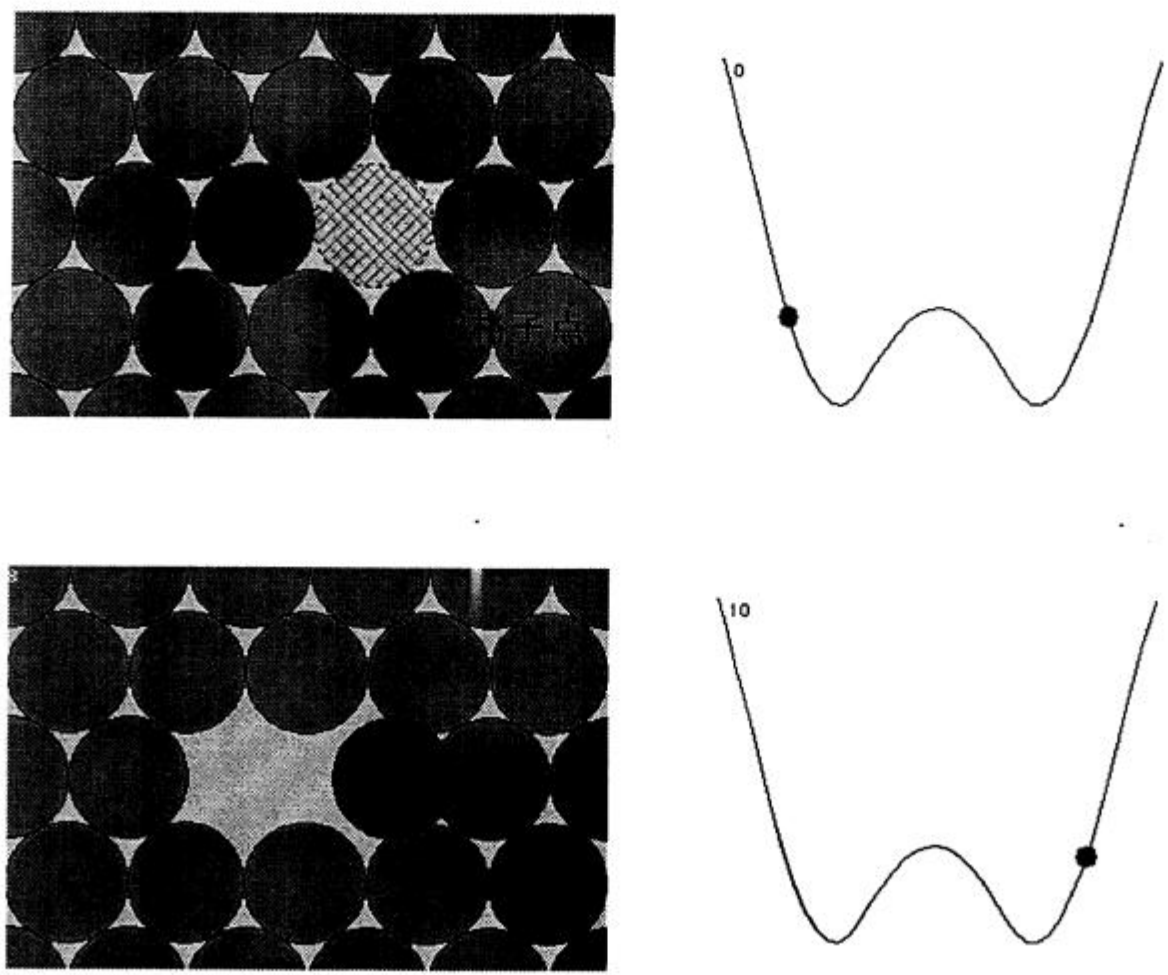


$$n = \frac{A}{\sqrt{t}} \exp(-\frac{x^2}{4Dt})$$

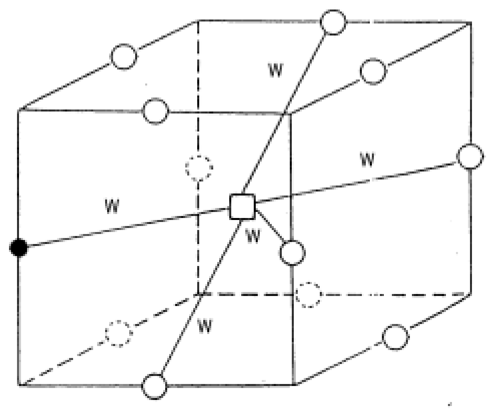


空格子点機構に基づく固体内原子拡散

空格子点による原子移動



4-1) Self-diffusion



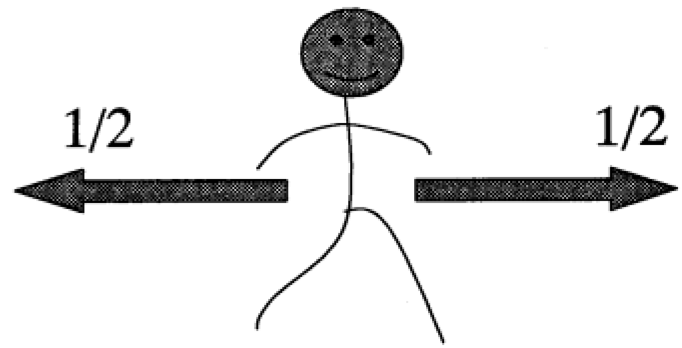
□ Vacancy

● Solvent tracer

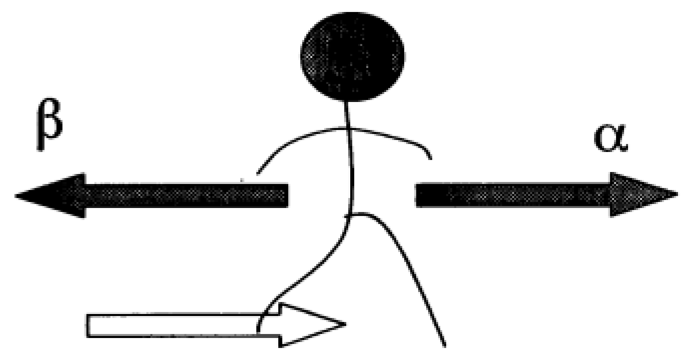
○ Solvent atom

Random Walk and Correlated Walk

- Random Walk

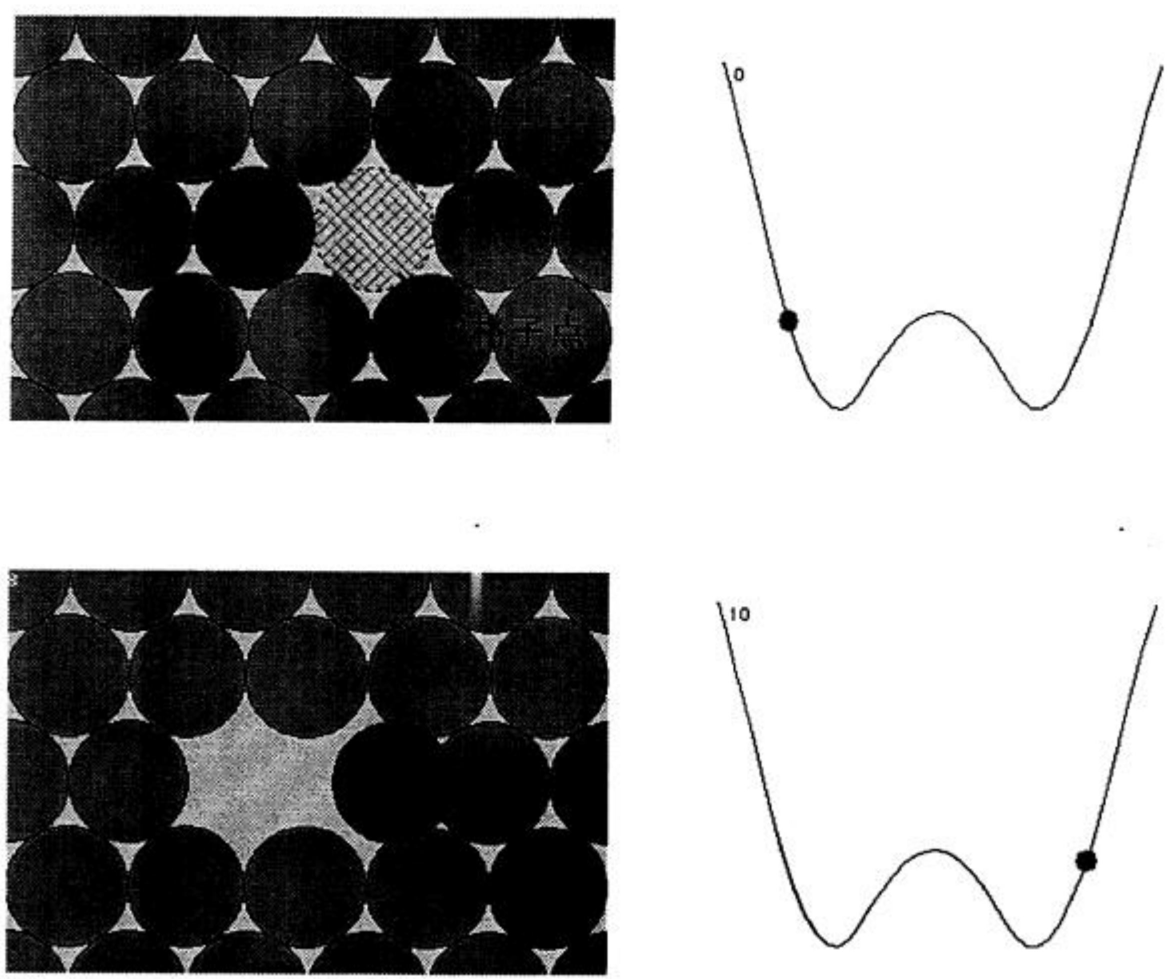


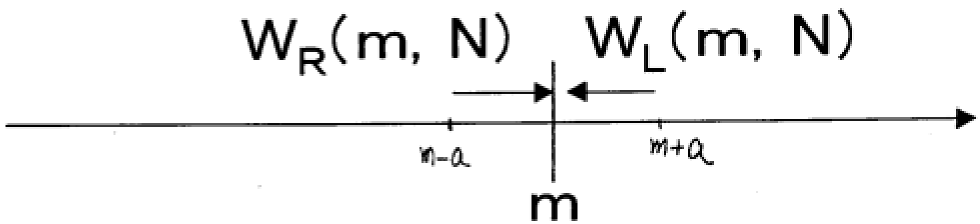
- Correlated Walk



空格子点機構に基づく固体内原子拡散

空格子点による原子移動





$$
\begin{aligned}
&W_R(m, N) \\
&\quad = \alpha W_R(m-a, N-1) + \beta W_L(m-a, N-1), \\
&W_L(m, N) \\
&\quad = \beta W_R(m+a, N-1) + \alpha W_L(m+a, N-1).
\end{aligned}
$$

$$
\begin{aligned}
P_1(X, N) &= \alpha P_1(X-1, N-1) + \beta P_2(X-1, N-1), \\
P_2(X, N) &= \beta P_1(X, N-1) + \alpha P_2(X, N-1).
\end{aligned}
$$

$$
\begin{aligned}
Q_i(\xi, \nu) &= \sum_{N=0}^{\infty} \sum_{X=0}^{N} P_i(X, N) \xi^X \nu^N \\
&= \sum_{N=0}^{\infty} \sum_{X=0}^{\infty} P_i(X, N) \xi^X \nu^N,
\end{aligned}
$$

$$
Q_1(\xi, \nu) = \frac{(1-\alpha\nu)\beta + \beta^2 \xi \nu}{\{1-(\alpha-\beta)\}A},
$$

$$
Q_2(\xi, \nu) = \frac{\beta^2 \nu + (1-\alpha\xi\nu)\beta}{\{1-(\alpha-\beta)\}A},
$$

where

$$
A = (1-\alpha\nu) - \{\alpha - (\alpha-\beta)\nu\}\xi\nu
$$

$$P(X,N) = \sum_{i=1}^{2} P_i(X,N).$$

$$Q(\xi,\nu) = \sum_{N=0}^{\infty}\sum_{X=0}^{\infty} P(X,N)\xi^X \nu^N = \frac{1-(\alpha-\beta)-\beta(\alpha-\beta)\nu-\beta(\alpha-\beta)\xi\nu}{\{1-(\alpha-\beta)\}A}.$$

$$\left.\frac{\partial}{\partial\xi}Q(\xi,\nu)\right|_{\xi=1} \longrightarrow \quad < X > \equiv \sum_{X=0}^{\infty} XP(X,N) = \frac{1}{2}N.$$

$$\left.\frac{\partial}{\partial\xi}\left(\xi\frac{\partial}{\partial\xi}Q(\xi,\nu)\right)\right|_{\xi=1} \longrightarrow \quad < X^2 > \equiv \sum_{X=0}^{\infty} X^2 P(X,N)$$

$$= \frac{1}{4}\Big[N^2 + \frac{1+(\alpha-\beta)}{1-(\alpha-\beta)}N - \frac{2(\alpha-\beta)\{1-(\alpha-\beta)^N\}}{\{1-(\alpha-\beta)\}^2}\Big].$$

$$m = (2X - N)a.$$

$$
\begin{aligned}
< m > &= 0 \\
< (\Delta m)^2 > &\equiv < (m - < m >)^2 > \\
&= \Big[\frac{1 + (\alpha - \beta)}{1 - (\alpha - \beta)} N \\
&\quad - \frac{2(\alpha - \beta)\{1 - (\alpha - \beta)^N\}}{\{1 - (\alpha - \beta)\}^2}\Big] a^2.
\end{aligned}
$$

$$D = \frac{< (\Delta m)^2 >}{2t} = \frac{fa^2}{2\tau} \quad \text{for large} \quad t$$

$$\tau = t/N$$

$$f = \frac{1 + (\alpha - \beta)}{1 - (\alpha - \beta)} = \frac{1 + \delta}{1 - \delta}$$

Lattice	f	δ
Diamond	0.5	-1/3
SC	0.6531	-0.2098
BCC	0.7272	-0.1579
FCC	0.7815	-0.1226

A.R.Allnatt, A.B.Lidiard,
“Atomic Transport in Solids” p355 Table10.1
(Cambridge University Press 1993)

$$
\begin{aligned}
W(m,N) = & \sum_{i=0}^{(m/a+N+2)/2} \frac{1}{2}\left(\frac{-4\delta}{(1+\delta)^2}\right)^i \left(\frac{1+\delta}{2}\right)^{N-1} \\
& \times \left\{ \left(\frac{1+\delta}{2}\right)^2 {}_{\frac{m/a+N-2}{2}}\mathrm{C}_{i-2} + (1+\delta)\, {}_{\frac{m/a+N-2}{2}}\mathrm{C}_{i-1} + {}_{\frac{m/a+N-2}{2}}\mathrm{C}_{i} \right\} \\
& \times {}_{N-i}\mathrm{C}_{\frac{m/a+N}{2}}
\end{aligned} \tag{8}
$$

for $aN \geq m > -aN$, where ${}_A\mathrm{C}_B$ (A, B: integer) is a binomial coefficients given by

$$
{}_A\mathrm{C}_B = \begin{cases} \dfrac{A!}{(A-B)!B!} & \text{for } A \geq B \geq 0, \\ 0 & \text{otherwise.} \end{cases} \tag{9}
$$

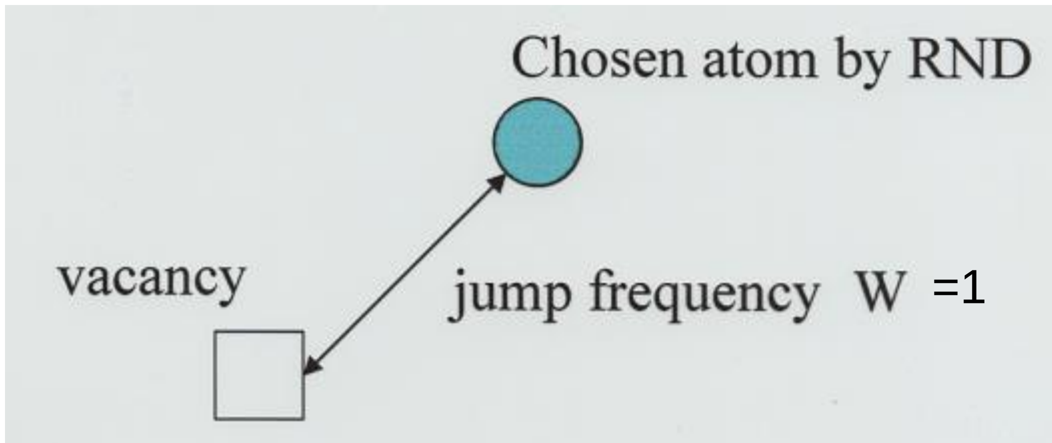
For $m = -aN$, we obtain

$$
W(-aN, N) = \frac{1}{2}\left(\frac{1+\delta}{2}\right)^{N-1} \tag{10}
$$

$W(m,N) = {}_N C_{(m/a+N)/2}/2^N \qquad \text{as} \qquad \delta \to 0$ (Random walk limit)

Simulation of self diffusion (correlation factor f) for FCC

$$f = \frac{<\Delta \mathbf{R}^2>}{N r^2}$$



試行回数	トレーサのステップ数	self-correlation factor	理論値 (0.7815) との誤差
2000	50	0.80387	2.86%
2000	100	0.790185	1.11%
2000	200	0.7848675	0.43%
2000	400	0.77285875	-1.1%
2000	800	0.784554375	0.39%

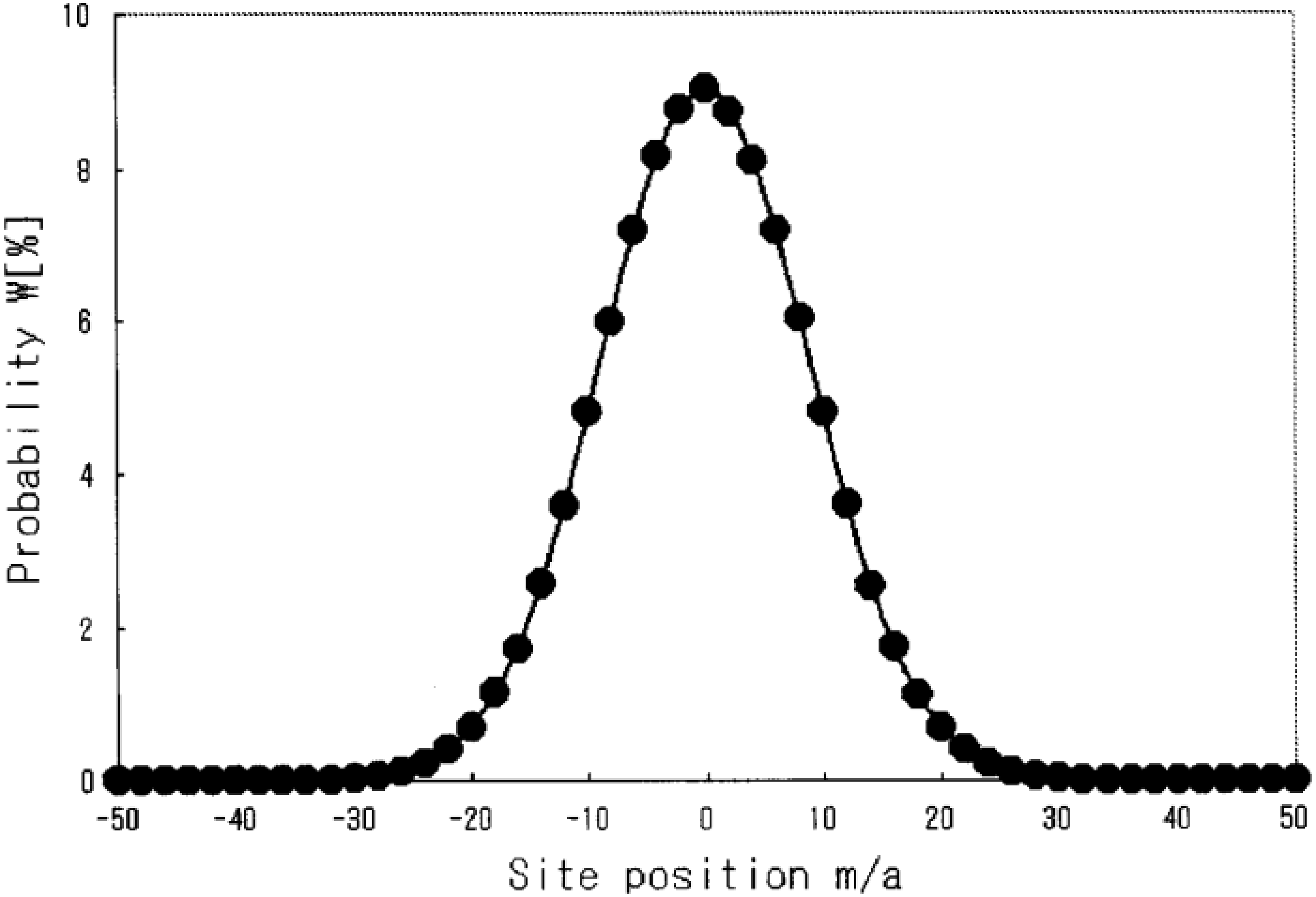
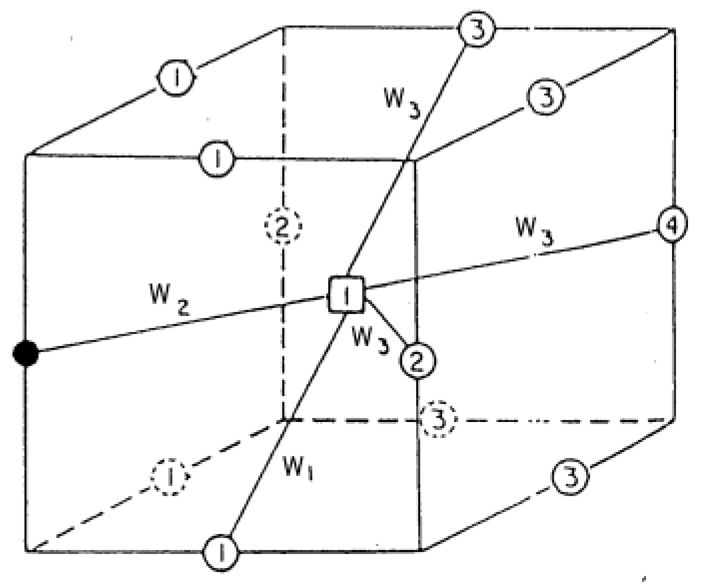


Figure 4: MC results (black circles) for 100,000 trajectories and analytical probability distributions (solid line) in case of $N = 100$ for FCC lattice.

The correlation factor is calculated from this, yielding 0.7794, which differs from the theoretical estimate by only 0.27%.

4-2) Impurity diffusion (Nearest neighbor binding model) for FCC



□ Vacancy

● Impurity tracer

○ Solvent atom

Manning calculated the correlation factor for this model and showed that the impurity correlation factor is given by

$$f = \frac{2w_1 + 7w_3F}{2w_1 + 2w_2 + 7w_3F}$$

where

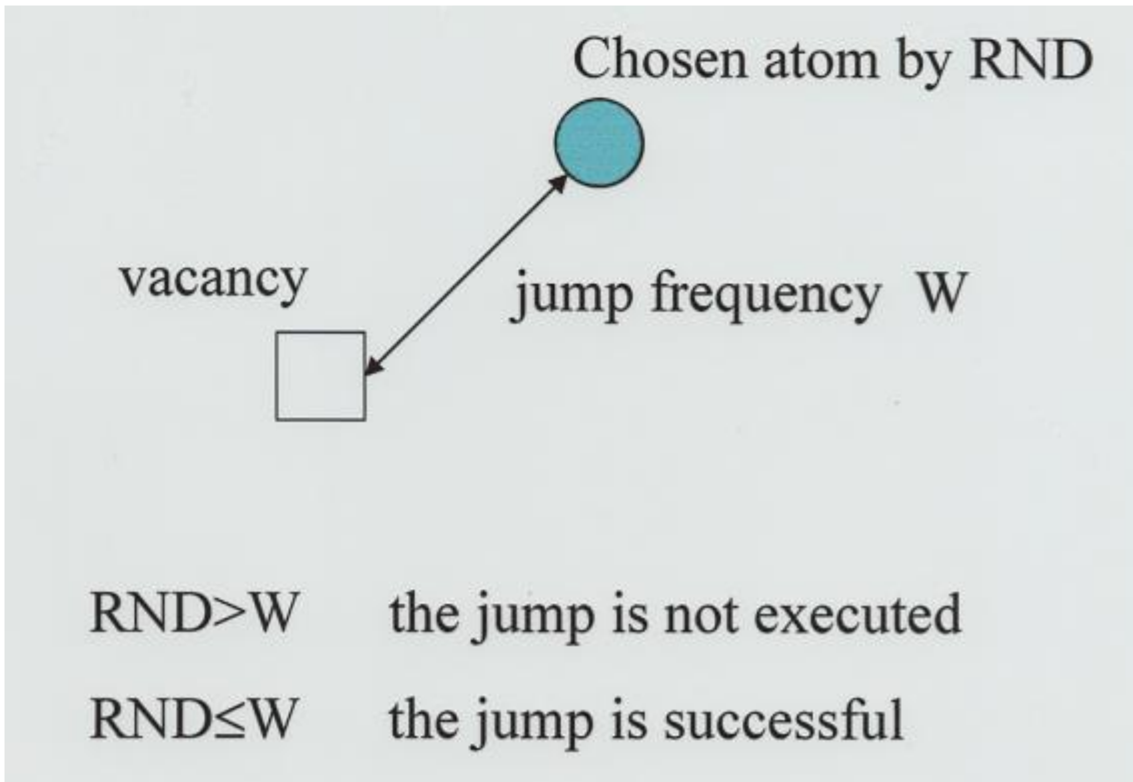
$$7F = 7 - \frac{10x^4 + 180.5x^3 + 927x^2 + 1341x}{2x^4 + 40.2x^3 + 254x^2 + 597x + 436}$$

$$x \equiv \frac{w_4}{w_0}$$

It should be noted that the correlation factor can be represented by only 3 parameters such as $w_1/w_2, w_3/w_2$ and $x = w_4/w_0$. In case of self-diffusion ($w_1 = w_2 = w_3 = w_4 = w_0 = 1$), the correlation factor reduces to 0.7814.

Simulation of impurity diffusion for FCC

$$f = \frac{< \Delta \mathbf{R}^2 >}{N r^2}$$



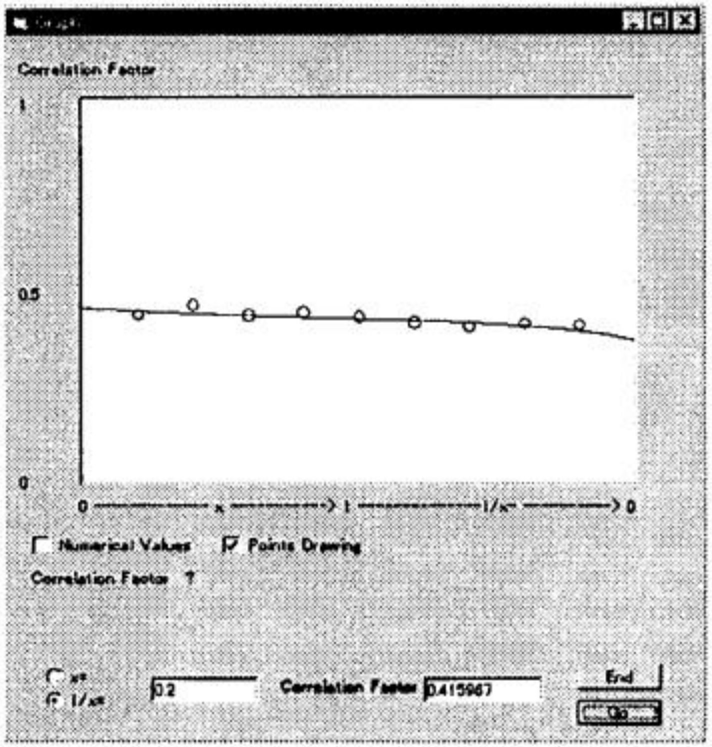


図 2: The variation of correlation factor with x with $w_1/w_2 = 0.5, w_3/w_2 = 0.1$ (small circles). The solid curve is drawn from Manning's formula.

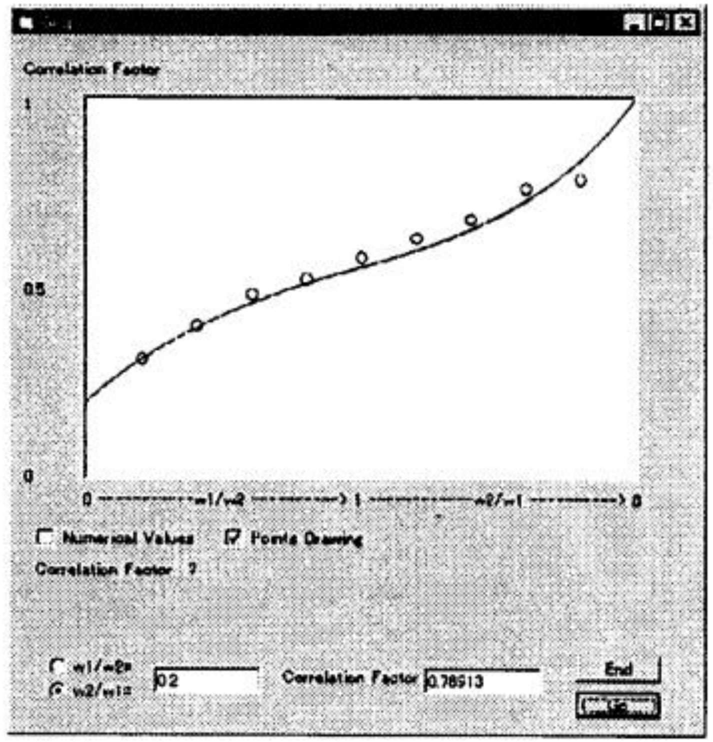
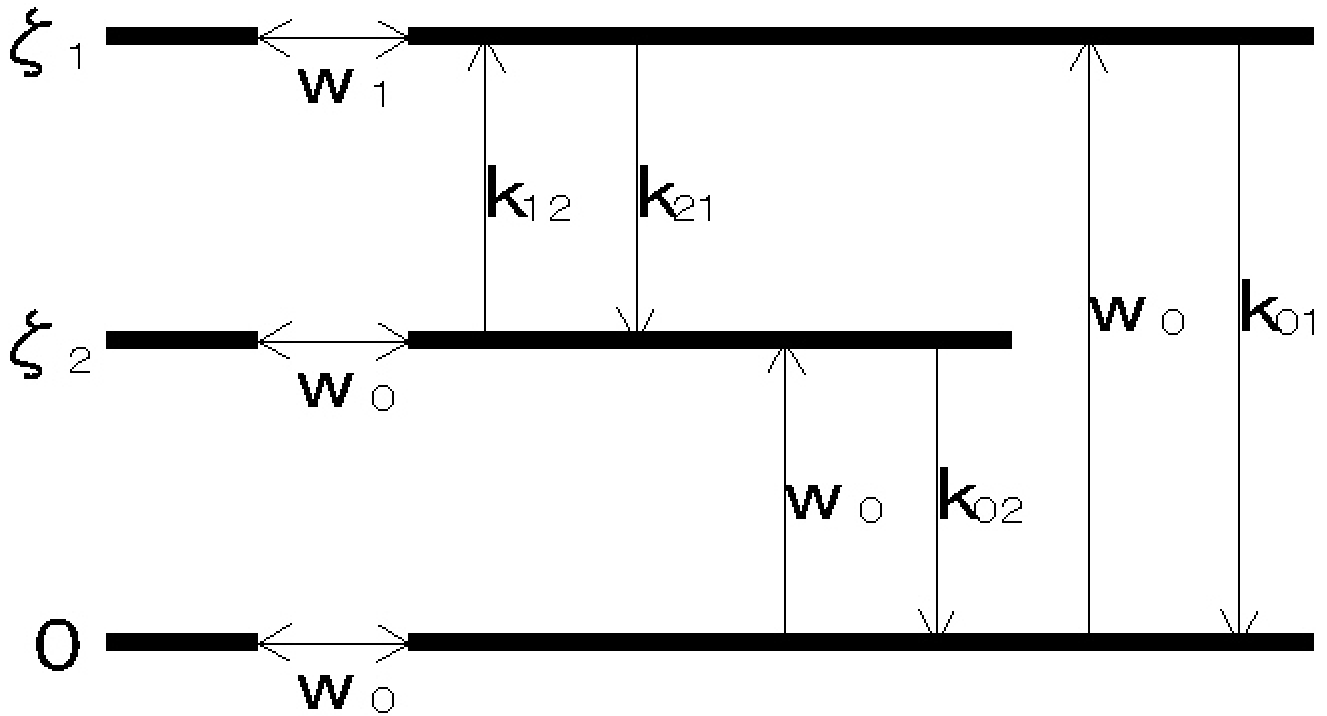
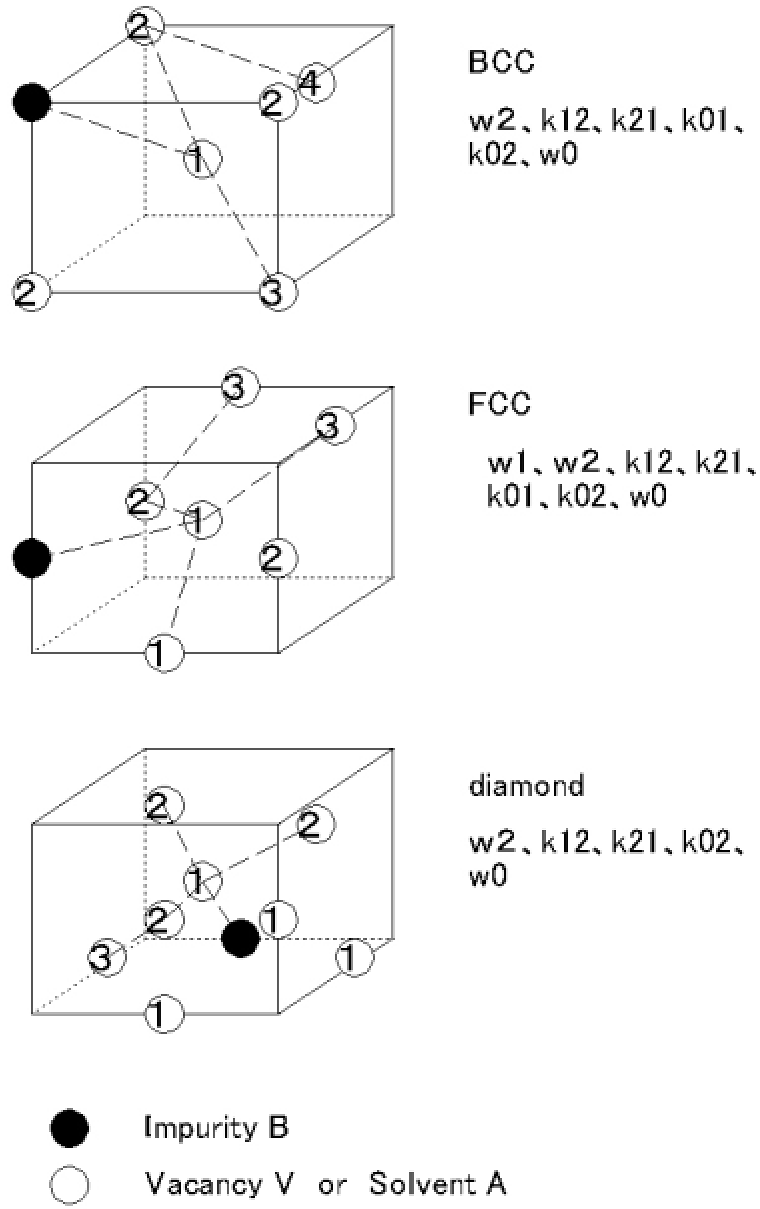


図 3: The variation of correlation factor with w_1/w_2 with $w_3/w_2 = 0.1, x = 1$ (small circles). The solid curve is drawn from Manning's formula.

4-3) Impurity diffusion (special next nearest neighbor binding model)




BCC
w2、k12、k21、k01、
k02、w0
FCC
w1、w2、k12、k21、
k01、k02、w0
diamond
w2、k12、k21、k02、
w0
Impurity B
Vacancy V or Solvent A

$$W(m,N) = \sum_{i=0}^{(m/a+N+2)/2} \frac{1}{2}\left(\frac{-4\delta}{(1+\delta)^2}\right)^i \left(\frac{1+\delta}{2}\right)^{N-1}$$
$$\times \left\{ \left(\frac{1+\delta}{2}\right)^2 {}_{\frac{m/a+N-2}{2}}\mathrm{C}_{i-2} + (1+\delta)\, {}_{\frac{m/a+N-2}{2}}\mathrm{C}_{i-1} + {}_{\frac{m/a+N-2}{2}}\mathrm{C}_{i} \right\}$$
$$\times\, {}_{N-i}\mathrm{C}_{\frac{m/a+N}{2}} \tag{8}$$

for $aN \geq m > -aN$, where ${}_A\mathrm{C}_B$ (A, B: integer) is a binomial coefficients given by

$$ {}_A\mathrm{C}_B = \begin{cases} \dfrac{A!}{(A-B)!B!} & \text{for } A \geq B \geq 0, \\ 0 & \text{otherwise.} \end{cases} \tag{9}$$

For $m = -aN$, we obtain

$$W(-aN, N) = \frac{1}{2}\left(\frac{1+\delta}{2}\right)^{N-1} \tag{10}$$

$$\delta = -w_2/(w_2 + H)$$

values\structures	BCC	FCC	diamond
H	$7k_{01}F$	$2w_1 + 7k_{01}F$	$3k_{21}F$
F	$\frac{2x^2+5.182x+2.476}{7(x+0.8106)}$	$\frac{x^2+6.401x+6.333}{7(x+1.667)}$	$\frac{6.167x+9.273}{3(x^2+3.6295x+3.091)}$

固体内原子輸送の統計理論の教科書および総合報告等

A.R.Allnatt, A.B.Lidiard,
“Atomic Transport in Solids”
(Cambridge University Press 1993)

A.R.Allnatt, A.B.Lidiard,
“Statistical theories of atomic transport in crystalline solids”
Rep. on progress in physics (1987) pp373-472

A.R.Allnatt, Y.Okamura,
“APPLICATIONS OF LINEAR RESPONSE THEORY TO DIFFUSION”
In Nontraditional Methods in Diffusion ed. by G.E.Murch, H.K.Birnbaum,J.R.Cost
(1984) pp. 237-257

References

"Computer Simulation of the Correlation Factor for Impurity Diffusion of 2nd-Nearest-Neighbor Binding Model"
Y.Okamura, K.Ueda,K.Haruyama
Proceeding The5th International Workshop on Similarity in Diversity 69 (1999)

"The Exact Distribution of a Simple Correlated Walk with a Stationary Initial Condition"
Y.Okamura, K.Haruyama
J. Korean Phys. Soc. 508 (2001)

"The Concentration Profiles for Impurity Diffusion via Vacancies Based on Correlated Walk Model"
Y.Okamura, K.Haruyama
Similaity in Diversity ed. By S.Fujita,H.Hara, D.L.Morabito,Y.Okamura 233 (Nova Sci.Pub. 2003)

"Theory of the distribution of tracers displaced by a vacancy"
Y.Okamura, J.Ukegawa, K.Haruyama, A.Suzuki
Statistical and Condensed Matter Physics:Over the Horizon ed. By S.Fujita, T.Obata, A.Suzuki (Nova Sci. Pub. 2007)

アメリカ、カナダの印象

自由市場(公正な競争社会、minorityにもチャンスはある)
内と外の意識があまり感じられない。
公共的な場所はお金がかからない(高速道路や駐車、遺跡や観光地、教会など)

本日はお疲れ様でした

気をつけてお帰りください