

Artificial Intelligence

in Advance

Prolog – Web knowledge
 Programming Computer
 NGUYEN DINH CUONG



```

#include <stdio.h>          • Bài toán phán tích lực      #include <stdio.h>          • Ví dụ: phân tích
int n;                      • Phân tích ác bá nay          int n, s;
int a[100];                  ou n, chán động            int a[100];
void inkq();                 void inkq()
{
    int i;
    for (i=0; i<n; i++) printf("%d", a[i]);
    printf("\n");
}
Void try (int i)
{
    int j;
    if (i==n) inkq();
    else
        for (j=0; j<=i; j++)
            {
                a[i]=j;
                try(i+j);
            }
}
int main ()
{
    printf("n=");
    scanf("%d", &n);
    try(n);
    return 1;
}

Note: Chú ý khuyếng cao
      (min đe data vecto hong lai)

$$\text{cos}(\vec{a}, \vec{b}) = (\|\vec{a}\| \cdot \|\vec{b}\|) \frac{\vec{a} \cdot \vec{b}}{\|\vec{a}\| \cdot \|\vec{b}\|} \quad (i = i + 1)$$


```

• Ví dụ: phân tích
 cao nô born.

• Ví dụ: tìm giá trị với
 phu nạp: speed

```

#include <stdio.h>          • Ví dụ: phân tích
int n, s;                  cao nô born.
int a[100];
Void inkq()
{
    int i, sum;
    i=1; sum=0;
    while (sum<s)
        {
            printf("%d %d", a[i], 32);
            sum = sum + a[i];
            i++;
        }
}
int main ()
{
    printf("\n");
    Void try (int i)
    {
        int j;
        far(i=0; i<=N; i++)
        a[i]=j;
        try(i+1);
        return 1;
    }
}

$$\text{for}(i=0; i \leq N; i++) a[i] = i;$$


$$\text{try}(i+1);$$


$$\text{return } 1;$$


$$a[i] = j;$$


$$s = s + j;$$


$$\text{if } (s == n) \text{ inkq();}$$


$$\text{else try}(i+1) \Rightarrow s = s - j;$$


```

BIG NUMBER

$25! = 15,511,210,043,330,985,984,000,000$ very large

unsigned long long $2^{64} - 1$, have no in C/C++

```
Import java.util.Scanner;
Import java.math.Bignumber;
Class Main {
    public static void main (String[] args) {
        BigInteger fac = BigInteger.ONE;
        for (int i=2 ; i <=25 ; i++)
            fac =fac.multiply (BigInteger.ValueOf(i));
        System.out.println(fac);
    }
}
```

Nếu mỗi viên gạch là 1mx1mx1m $2^{64} - 1$ viên gạch có lắp được biến không

BIG NUMBER

```
#include <cstdio>
using namespace std;
int N;
char x[100];

int main() {
    scanf("%d\n", &N);
    while (N--) {
        scanf("0..%[0..9]...\\n", &x);
    }
    printf("the digits number are %s\\n",x);
}
```

Parallel OpenMP

```
#include <omp.h>
#include <stdio.h>
int main (int argc, char *argv[])
{
    /* sequencetial code */
    #pragma omp parallel
    {
        printf(" I am a parallel region.");
    }
    /* sequence code */
    return 0;
}
```

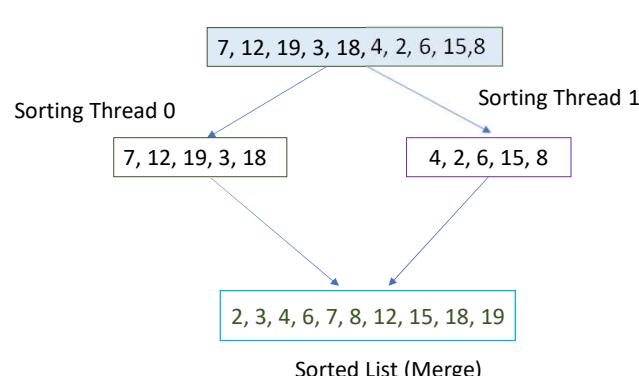
```
#pragma omp parallel for
for ( i = 0; i < N ; i ++ ) {
    c[i] = a [i] + b [i];
}
```

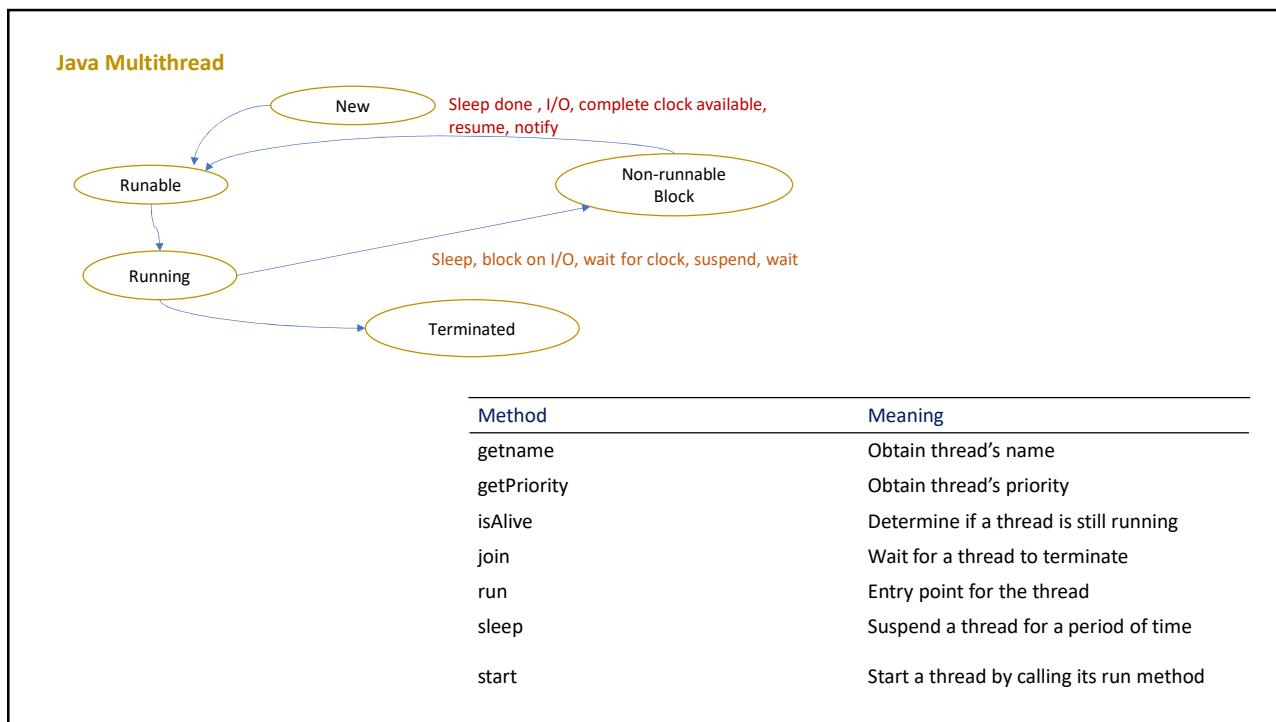
```
#include <pthread.h>
#include <stdio.h>

int value=0;
void *runner (void *param) /* the thread */

int main (int argc, char *argv[])
{
    pid_t pid;
    pthread_t tid;
    pthread_attr_t attr;
    pid = fork();
    if (pid ==0) /* child process */
        pthread_attr_init(&attr);
        pthread_create(&tid, &attr, runner, NULL);
        pthread_join(tid, NULL);
        printf("CHILD: value=%d", value); /* LINE P */
    }
    else if (pid >0) /* parent process */
        wait (NULL);
        printf (" PARENT : value=%d", value); /* LINE P */
    }
```

```
void *runner(void *param) {
    value=5;
    pthread_exit(0);
}
```





Java Multithread

```

public void run()

public class MyClass implements Runnable{
public void run(){
System.out.println("MyClass running");
}
}

Thread t1=new Thread(new MyClass());
t1.start();

public class Myclass extends Thread{
    public void run(){
        System.out.println("MyClass running");
    }
}

MyClass t1= new MyClass();
T1.start();
  
```

```

class RunnableDemo implements Runnable {
    private Thread t;
    private String threadName;

    RunnableDemo(String name) {
        threadName = name;
        System.out.println("Creating " + threadName);
    }

    public void run() {

        System.out.println("Running " + threadName);
        try {
            for (int i=4; i>0; i--) {
                System.out.println("Thread:" + threadName + ", " + i);
                // Let the thread sleep for a while
                Thread.sleep(50);
            }
        } catch (InterruptedException e) {
            System.out.println("thread" + threadName + " interrupted.");
        }
        System.out.println("Thread " + threadName + " exiting.");
    }

    Public void start() {
        System.out.println("Starting " + threadName);
        If (t== null) {
        t= new Thread (this, threadName);
        t.start();
        }
    }
}

```

```

$javac TestThread.java
$java -Xmx128M -Xms16M TestThread
Creating Thread-1
Starting Thread-1
Creating Thread-2
Starting Thread-2
Running Thread-1
Thread: Thread-1, 4
Running Thread-2
Thread: Thread-2, 4
Thread: Thread-1, 3
Thread: Thread-2, 3
Thread: Thread-1, 2
Thread: Thread-2, 2
Thread: Thread-1, 1
Thread: Thread-2, 1
Thread Thread-1 exiting.
Thread Thread-2 exiting.

```

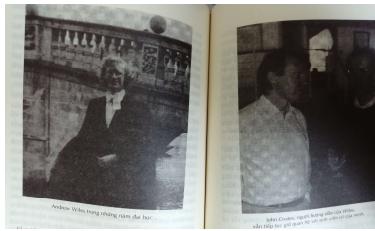
```

public class TestThread{
public static void main (String args[]) {
RunnableDemo R1 = new RunnableDemo("Thread -1 ");
R1.start();

RunnableDemo R2 =new RunnableDemo("Thread -2");
R2.start();
}
}

```

ĐỊNH LÝ FERMAT



$$x^n + y^n = z^n, n > 2$$

```

for(x=minint; x≤maxint ; x++)
    for (y=minint; y ≤ maxint; y++)
        for (n=minint; n ≤ maxint; n++)
            if ( $x^n + y^n = z^n$ ) then prob = ok

```



Bài toán puzzle.

2	4	5
7	6	1
1	3	9

A

2	6	4
7	1	5
1	3	9

B

Heuristic để sao lách cái mèo

giảm thiểu số lượng bao bì

và tối ưu hóa

Bảng duyệt

thứ tự	Định	Open	Close
0		A	∅
1	A	ʒ B3, C4, D5 ʒ	A3
2	B	ʒ E3, C4, D5 ʒ	B3
3	E	ʒ G3, C4, D5, F5 ʒ	E3
4		đứng vị trí thứ 2	

Bài toán Rubik's cube, quay khối Rubik

state A

$f(a) = \phi + 3 = 3$

state B

$f(b) = 1 + 2 = 3$

state C

$f(c) = 1 + 3 = 4$

state D

$f(d) = 1 + 4 = 5$

Lỗi

hỗn

A	B
A	C
A	D
B	E
E	F
E	G

State B

2	4
1	6
1	3
9	5

$f(e) = 2 + 1 = 3$

State E

2	4
7	6
1	3
9	5

$f(g) = 3 + \phi = 3$

Nếu có sáu con vật qua sông

Java & C# thi vẫn có bao bì

nhưng nó sẽ không bao bì

để có hiệu suất

Bài toán vượt qua sông

$\overbrace{\text{---}}^{\text{B8-A}}$ Nguyên, cáo, dê, cá bắp

$\overbrace{\text{---}}^{\text{B8-B}}$ Cát giòi

B8-A (N, C, D_e, C_B)

B8-B ($-, -, -, -$)

truy thai

4 bits

4 bits sau

$\overbrace{\text{---}}^{\text{B8-A}}$ (N, C, D_e, C_B)

$\overbrace{\text{---}}^{\text{B8-B}}$ ($-, -, -, -$)

hiển thị lựa chọn

number cho truy thai

1

2

3

4

$\overbrace{\text{---}}^{\text{B8-A}}$ ($-, -, -, C_B$)

$\overbrace{\text{---}}^{\text{B8-B}}$ ($N, C, D_e, -$)

$\overbrace{\text{---}}^{\text{B8-A}}$ ($-, C, -, -$)

$\overbrace{\text{---}}^{\text{B8-B}}$ ($N, C, -, C_B$)

Đúng thuật toán dc quy toán cho nhữn

quay bao giờ để xử lý

con thuyền đi tay

Khi quá đà với

kiểm tra key sai

Đúng nhữn thuyền

Mỗi vòng bao bao

0 1

* Sử dụng parallel cho bài toán này.

Nhiều con vật qua sông

JIProlog

```

* of the License, or any later version.
*
* This program is distributed in the hope that it will be useful,
* but WITHOUT ANY WARRANTY; without even the implied warranty of
* MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
* GNU General Public License for more details.
*
* You should have received a copy of the GNU General Public License
* along with this program; if not, write to the Free Software
* Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111-1307, USA.
*****/
```

```

father(terach,abraham).
father(terach,nachor).
father(terach,haran).
father(abraham,isaac).
father(haran,lot) !.
father(haran,milcah).
nother(sara,isaac).
male(terach).
male(abraham).
male(nachor).
male(haran).
male(isaac).
male(lot).
female(sarah).
female(milcah).
female(yiscah).
Likes(X,pome).
son(X,Y) :- father(Y,X), male(X).
daughter(X,Z) :- father(Z,X), female(X).
grandfather(X,Z) :- father(X,Y), father(Y,Z).
```

```
(\QUEENS.PL'0))
:File consulted
:Saved file: C:\soft\java_platform\JIProlog-4.1.6.1\JIProlog\samples\prolog\bible.pl
- Warning, singleton variables found at line 39: X
:File consulted
```

```

explainMoves :- nl, write('You choose a move by giving the number of the'),
nl, write('square to which you wish to move:'), nl, nl,
write(' 1 | 2 | 3'), nl, writeln,
write(' 4 | 5 | 6'), nl, writeln,
write(' 7 | 8 | 9'), nl.
```

```

/* show square numbers */
writeln :- write(' -----'), nl.

games :- game, !, optionalGames.

optionalGames :- nl, write('Care for another game (y/n)? '),
inputResponse(Response), !, process(Response).

inputResponse(Response) :- read(Answer), okResponse(Answer, Response), !.
/* error check and put in standard form */
inputResponse(Response) :- write('Not a valid response -- type y or n: '),
inputResponse(Response). /* reinput */

/* The standard response forms are yes and no: */
okResponse('y', yes), okResponse('Y', yes), okResponse(yes, yes).
okResponse('YES', yes), okResponse('Yes', yes).
okResponse('n', no), okResponse('N', no), okResponse(no, no).
okResponse('NO', no), okResponse('No', no).

process(no) !. /* end the program */
process(yes) :- games. /* continue play */

game :- nl, write('Do you wish to be x or o? '), inputPlayer(Player), nl,
write('Do you want to go first (y/n)? '), inputResponse(PlayerMoves), !,
Play(Player, PlayerMoves, 9, board(_,_,-,_,-,_,-,_,-,_)),
/* 3rd arg of play is # of open squares, originally 9 */
```

```
:File consulted
:Saved file: C:\soft\java_platform\JIProlog-4.1.6.1\JIProlog\samples\prolog\bible.pl
- Warning, singleton variables found at line 39: X
:File consulted
:File consulted
```

```

/*
 * For example, board(x,o,x,_,o,_,o,_,_) represents the board:
 *
 *      x | o | x
 *      -----
 *          | o |
 *      -----
 *          o |   |
 *      /
 */

go :- nl, write('Playing tic-tac-toe.'), nl, explainMoves, !, games, !, nl, nl.

explainMoves :- nl, write('You choose a move by giving the number of the'),
nl, write('square to which you wish to move!'), nl, nl,
write('  1 | 2 | 3'), nl, writeln,
write('  4 | 5 | 6'), nl, writeln,
write('  7 | 8 | 9'), nl.

/*
 * show square numbers */
printline :- write(' -----'), nl.

games :- game, !, optionalGames.

optionalGames :- nl, write('Care for another game (y/n)? '),
inputResponse(Response), !, process(Response).

inputResponse(Response) :- read(Answer), okResponse(Answer, Response), !.
/* error check and put in standard form */
inputResponse(Response) :- write('Not a valid response -- type y or n: '),
read(Answer), !, process(Answer).
/*-----*/
:- !.
:File consulted
:Saved file: C:\soft\java_flatform\JIProlog-4.1.6.1\JIProlog\samples\prolog\bible.pl
: Warning, singleton variables found at line 39: X
:File consulted
:File consulted

```

You choose a move by giving the number of the square to which you wish to move:

```

1 | 2 | 3
-----
4 | 5 | 6
-----
7 | 8 | 9

```

Do you wish to be x or o? :>o

Do you want to go first (y/n)? :>n

My move: 5
+ | |

x

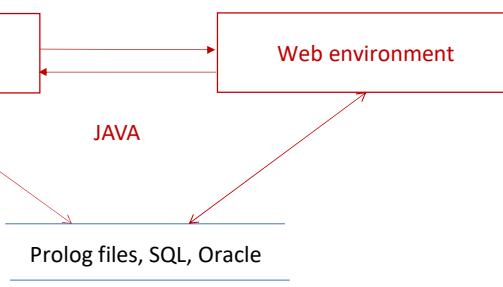
Your move: :>3
+ | | o

| x |

```

father(terach,abraham).
father(terach,nachor).
father(terach,haran).
father(abraham,isaac).
father(haran,lot):-!.
father(haran,milcah).
mother(sara,isaac).
male(terach).
male(abraham).
male(nachor).
male(haran).
male(isaac).
male(lot).
female(sarah).
female(milcah).
female(yiscah).
likes(X,pome).
son(X,Y):-father(Y,X),male(X).
daughter(X,Z):-father(Z,X),female(X).
granfather(X,Z):-father(X,Y),father(Y,Z).

```



```

<HTML>
<HEAD><TITLE>
Sample JSP
</TITLE></HEAD>

<%@ page language="java" import="java.util.Vector" %>
<%@ page import="com.ugos.JIPProlog.engine.*" %>

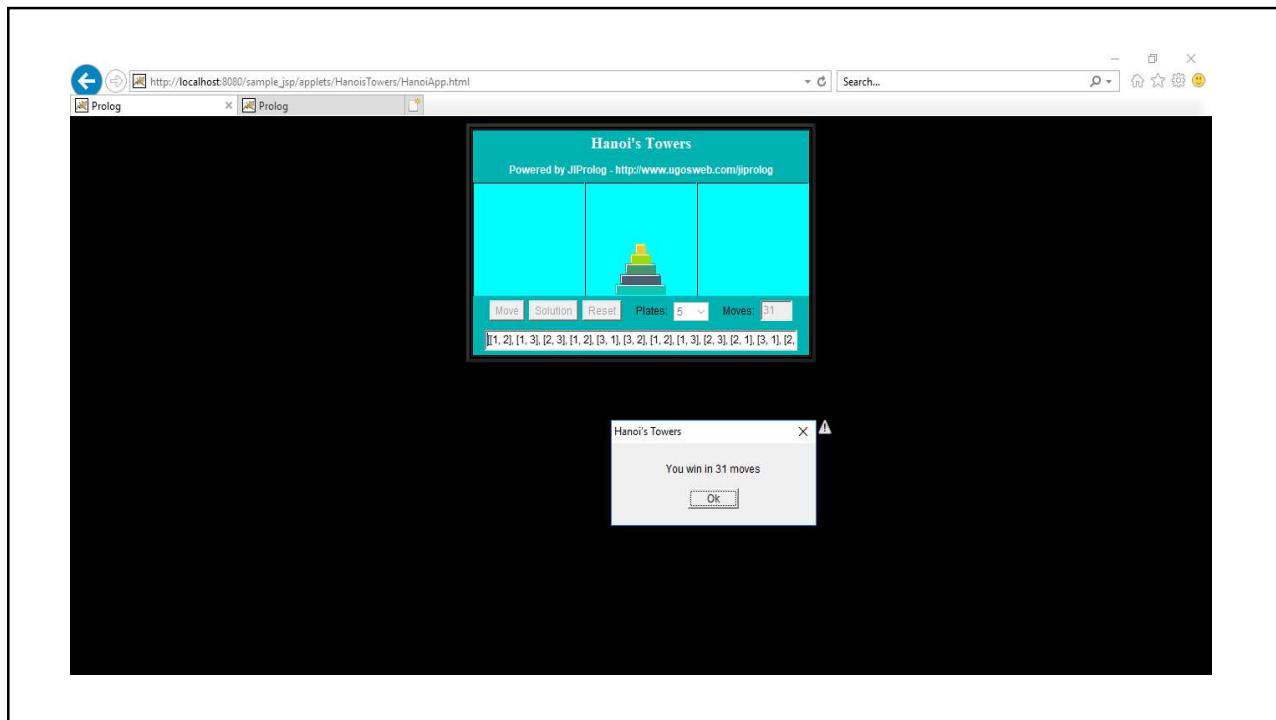
<%
// New instance of prolog engine
JIPEngine jip = new JIPEngine();
JIPTerm queryTerm = null;
// collector for solution
Vector vect = new Vector();
// parse query
try
{
// consult file
// files are searched in the search path
jip.consultFile("bible.pl");
queryTerm = jip.getTermParser().parseTerm("father(X, Y).");
}
catch(JIPSyntaxErrorException ex)
{
ex.printStackTrace();
}

```

```

catch(IOException ex)
{
ex.printStackTrace();
}
// open Query
JIPQuery jipQuery = jip.openSynchronousQuery(queryTerm);
JIPTerm solution;
try
{
// Collect solutions
// Loop while there is another solution
while ((solution = jipQuery.nextSolution()) != null)
{
vect.addElement(solution);
}
}
catch(JIPRuntimeException ex)
{
System.out.println(ex.getMessage());
}
%>
<BODY BGCOLOR="white">
<B> Solutions: </B>
<ul>
<%
// print solutions
for(int i = 0; i < vect.size(); i++)
{
%>
<li> <%= vect.elementAt(i) %>
<%
}
%>
</ul>
</BODY>
</HTML>

```



HanoiApplet.java - Visual Studio Code

```

File Edit Selection View Go Debug Tasks Help
HanoiApplet.java x
52  private int m_nHeight;
53  private int m_nMoves;
54  private int m_nQuery;
55
56  private Frame m_mainFrame;
57
58  private JIPEngine m_jip;
59
60  public void init()
61  {
62      super.init();
63
64      // Create PrologEngine and load hanoi.txt
65      m_jip = new JIPEngine();
66
67      try
68      {
69          m_jip.consultStream(getClass().getResourceAsStream("/hanoi.txt"), "hanoi.txt");
70      }
71      catch(JIPSyntaxErrorException ex)
72      {
73          add(new Label(ex.toString()));
74          return;
75      }
76      catch(IOException ex)
77      {

```

hanoi - Notepad

```

File Edit Format View Help
*****
* Hanoi's Towers
* hanoi(height, tower1, tower2, tower3, moves)
* Example:
*   hanoi(3, a,b,c, X).
*
*****/
```

```

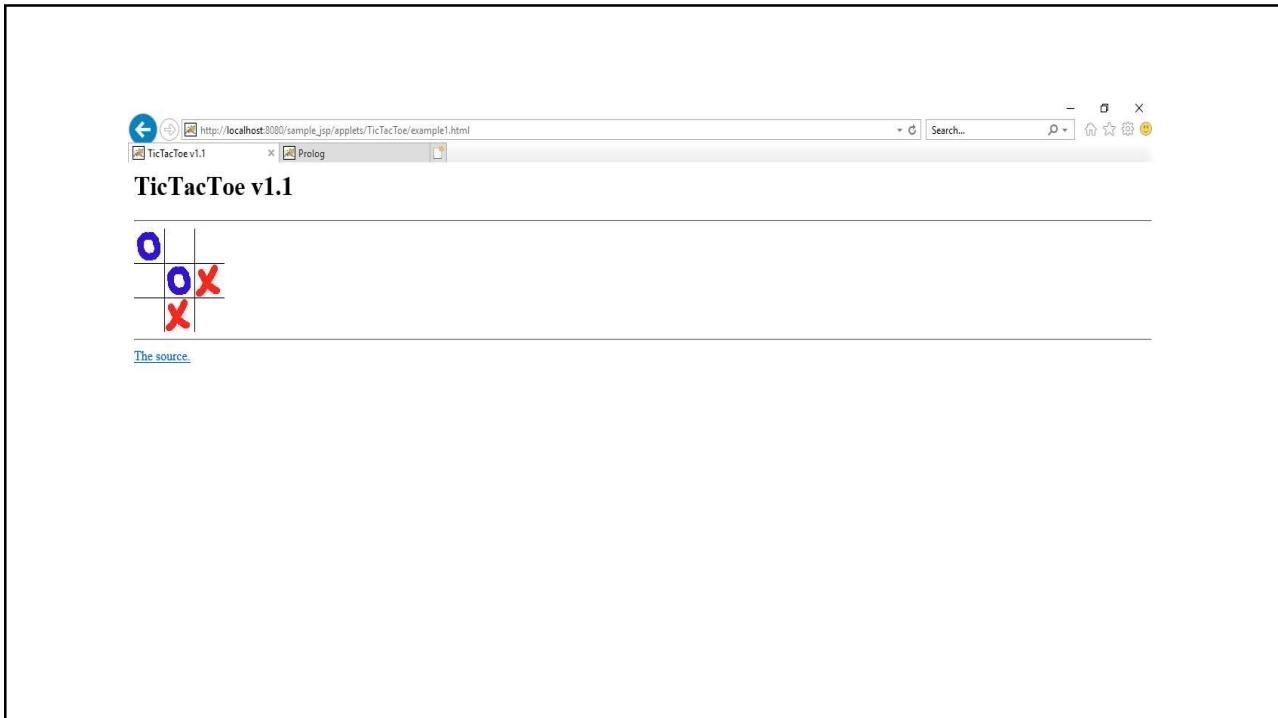
hanoi(1, A,B,C,[[A,B]]):-!.
hanoi(N, A,B,C,Moves):-
    N1 is N - 1,
    hanoi(N1, A,C,B,Ms1),
    hanoi(N1, C,B,A,Ms2),
    append(Ms1, [[A,B]|Ms2], Moves),
    !.
```

HanoiApp - Notepad

```

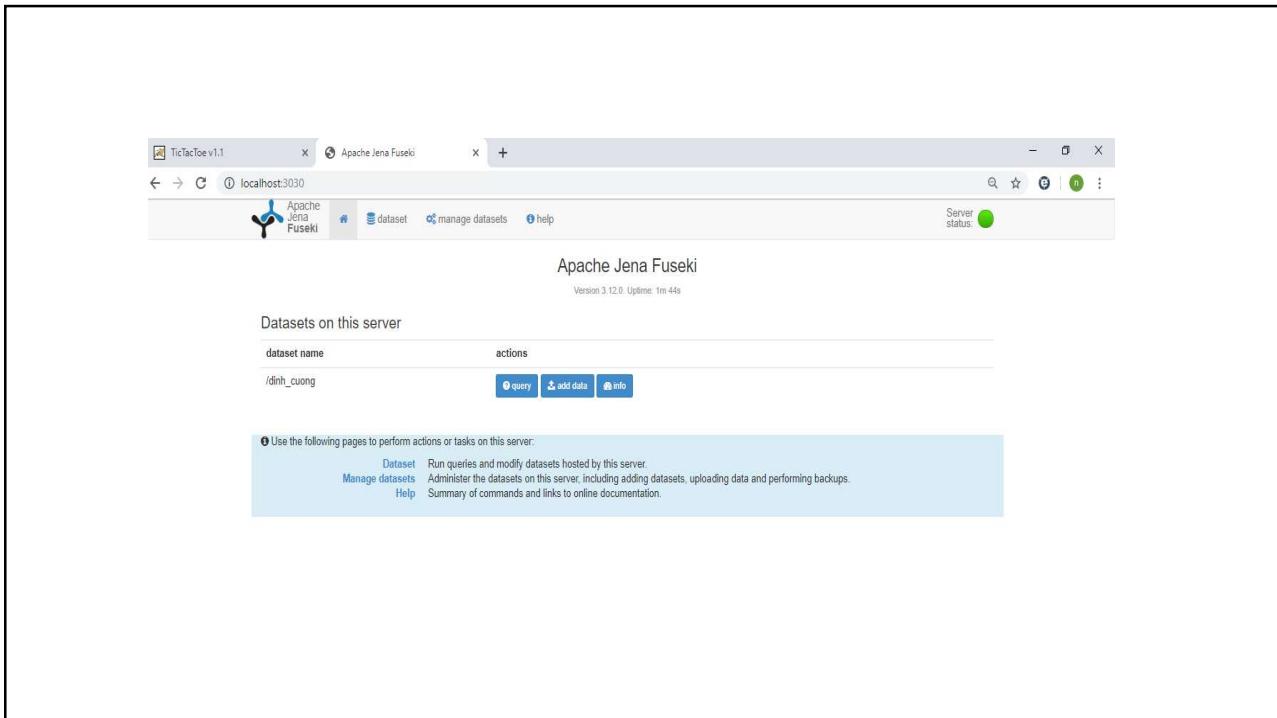
File Edit Format View Help
<BODY BGCOLOR="#000000>
<title>Prolog</title><center>
<table border=4><tr><td>
<applet code="HanoiApplet.class" archive="hanoi.jar, jiprolog.jar" codebase=". " width=402 height=250>
</applet></td></tr></table><br>
```

Name	Date modified	Type	Size
hanoi	01-Jun-15 8:09 PM	Executable Jar File	9 KB
hanoi	01-Jun-15 8:09 PM	Text Document	1 KB
HanoiApp	01-Jun-15 8:09 PM	HTML Document	1 KB
HanoiApplet.class	01-Jun-15 8:09 PM	CLASS File	11 KB
HanoiApplet	01-Jun-15 8:09 PM	Java Source File	14 KB
HanoisTower.class	01-Jun-15 8:09 PM	CLASS File	3 KB
HanoisTower	01-Jun-15 8:09 PM	Java Source File	5 KB
HanoisTowers.cpr	01-Jun-15 8:09 PM	CPR File	2 KB
jiprolog	01-Jun-15 8:09 PM	Executable Jar File	128 KB
MessageDialog.class	01-Jun-15 8:09 PM	CLASS File	4 KB
MessageDialog	01-Jun-15 8:09 PM	Java Source File	4 KB



SPARQL – Web Semantic

```
42] Server      INFO  Apache Jena Fuseki 3.12.0
43] Config      INFO  FUSEKI_HOME=C:\soft\java_platform\apache-jena-fuseki-3.12.0\apache-jena-fuseki-3.
43] Config      INFO  FUSEKI_BASE=C:\soft\java_platform\apache-jena-fuseki-3.12.0\apache-jena-fuseki-3.
43] Config      INFO  Shiro file: file:///C:/soft/java_platform\apache-jena-fuseki-3.12.0\apache-jena-fu
nro.ini
44] Config      INFO  Configuration file: C:\soft\java_platform\apache-jena-fuseki-3.12.0\apache-jena-f
config.ttl
45] riot        WARN   [line: 5, col: 9 ] Bad IRI: <C:\soft\java_platform\apache-jena-fuseki-3.12.0\apac
2.0\run\config.ttl#> Code: 4/UNWISE CHARACTER in PATH: The character matches no grammar rules of URIs/
ters are permitted in RDF URI References, XML system identifiers, and XML Schema anyURIs.
45] Config      INFO  Load configuration: file:///C:/soft/java_platform\apache-jena-fuseki-3.12.0\apach
2.0\run\configuration/dinh_cuong.ttl
49] Config      INFO  Register: /dinh_cuong
50] Server      INFO  Started 2019/07/27 14:48:50 ICT on port 3030
```



SPARQL

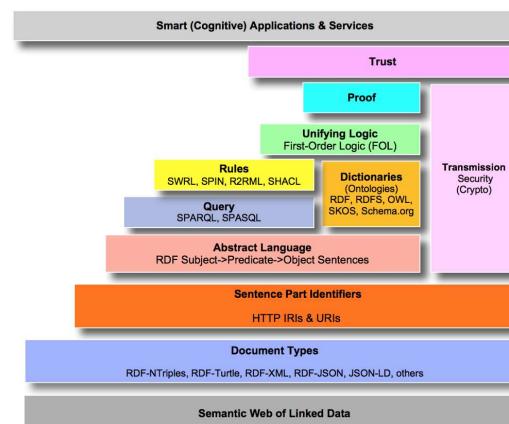
```
package jena.examples.rdf ;
import org.apache.jena.rdf.model.*;
import org.apache.jena.vocabulary.*;

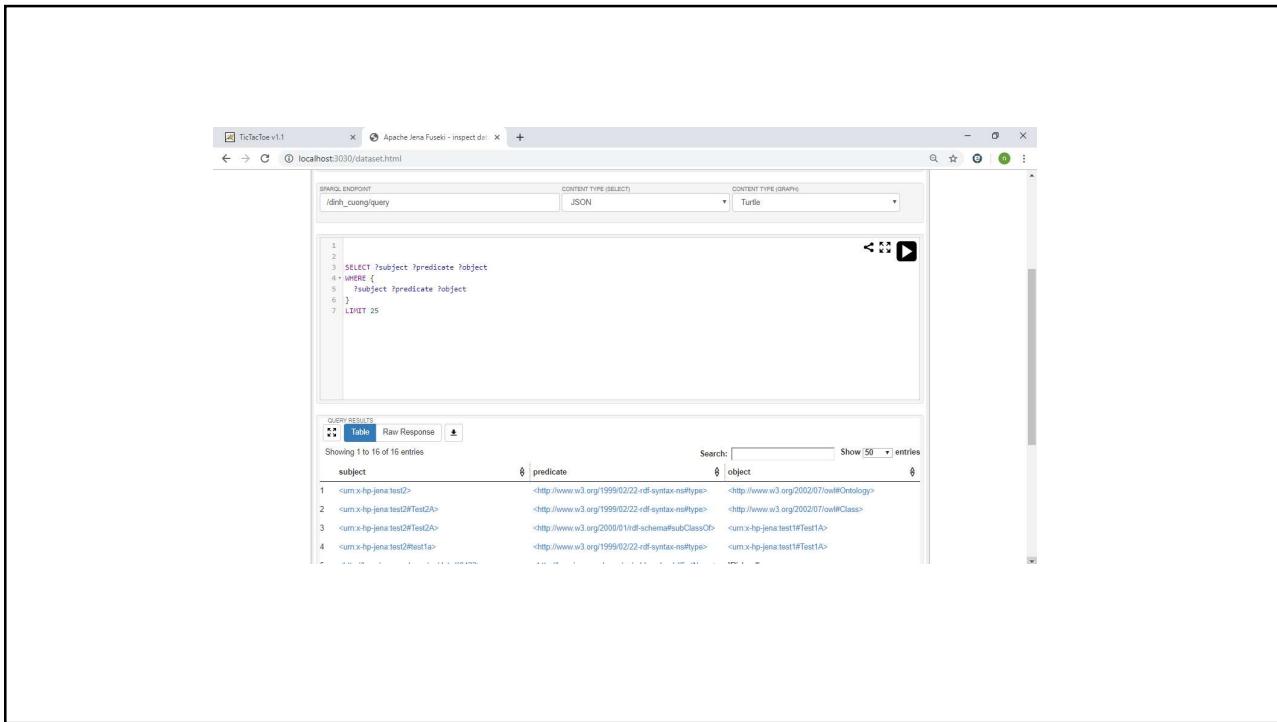
/** Tutorial 1 creating a simple model
 */

public class Tutorial01 extends Object {
// some definitions
static String personURI =
"http://somewhere/JohnSmith";
static String fullName = "John Smith";
public static void main (String args[]) {
// create an empty model
Model model = ModelFactory.createDefaultModel();

// create the resource
Resource johnSmith =
model.createResource(personURI);

// add the property
johnSmith.addProperty(VCARD.FN, fullName);
}
}
```





filename: ex003.rq

PREFIX ab: <<http://learningsparql.com/ns/addressbook#>>

SELECT ?craigEmail

WHERE { ab:craig ab:email ?craigEmail . }

filename: ex002.ttl

filename: ex001.rq

PREFIX d: <<http://learningsparql.com/ns/demo#>>

SELECT ?person

WHERE { ?person d:homeTel "(229) 276-5135" . }

@prefix ab: <<http://learningsparql.com/ns/addressbook#>> .

ab:richard ab:homeTel "(229) 276-5135" .

ab:richard ab:email "richard49@hotmail.com" .

ab:cindy ab:homeTel "(245) 646-5488" .

ab:cindy ab:email "cindym@gmail.com" .

ab:craig ab:homeTel "(194) 966-1505" .

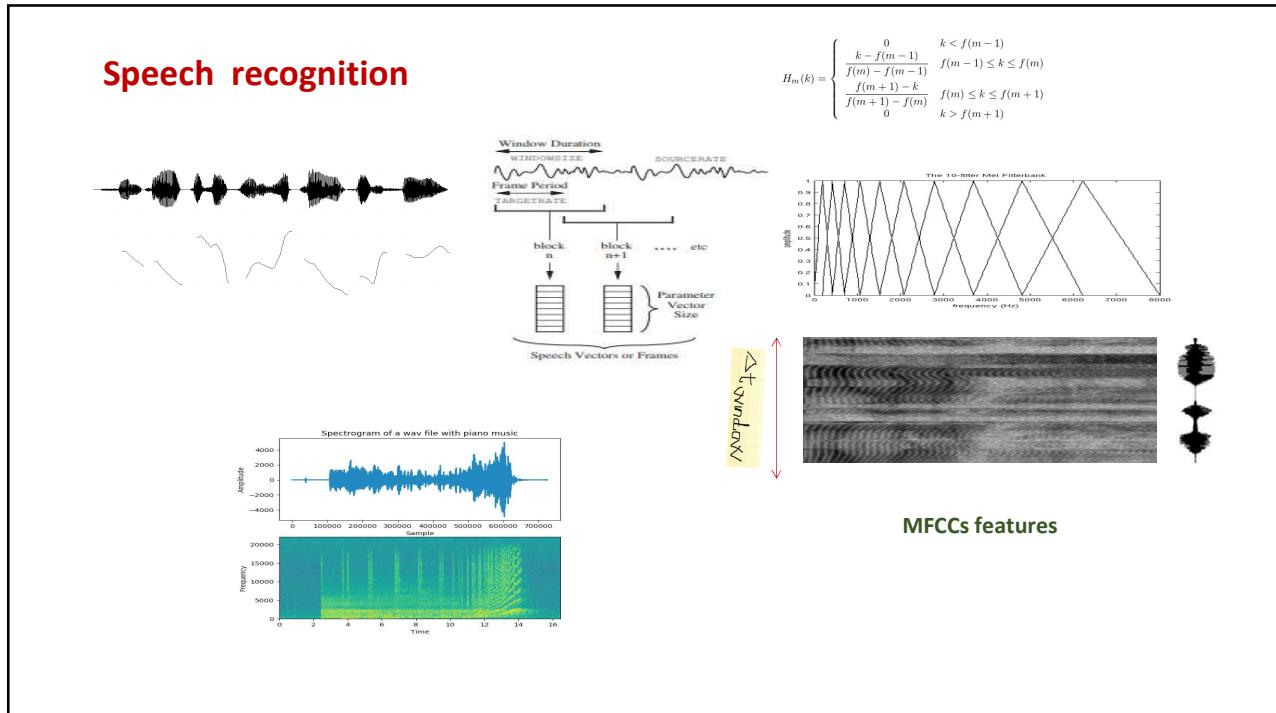
ab:craig ab:email "craigellis@yahoo.com" .

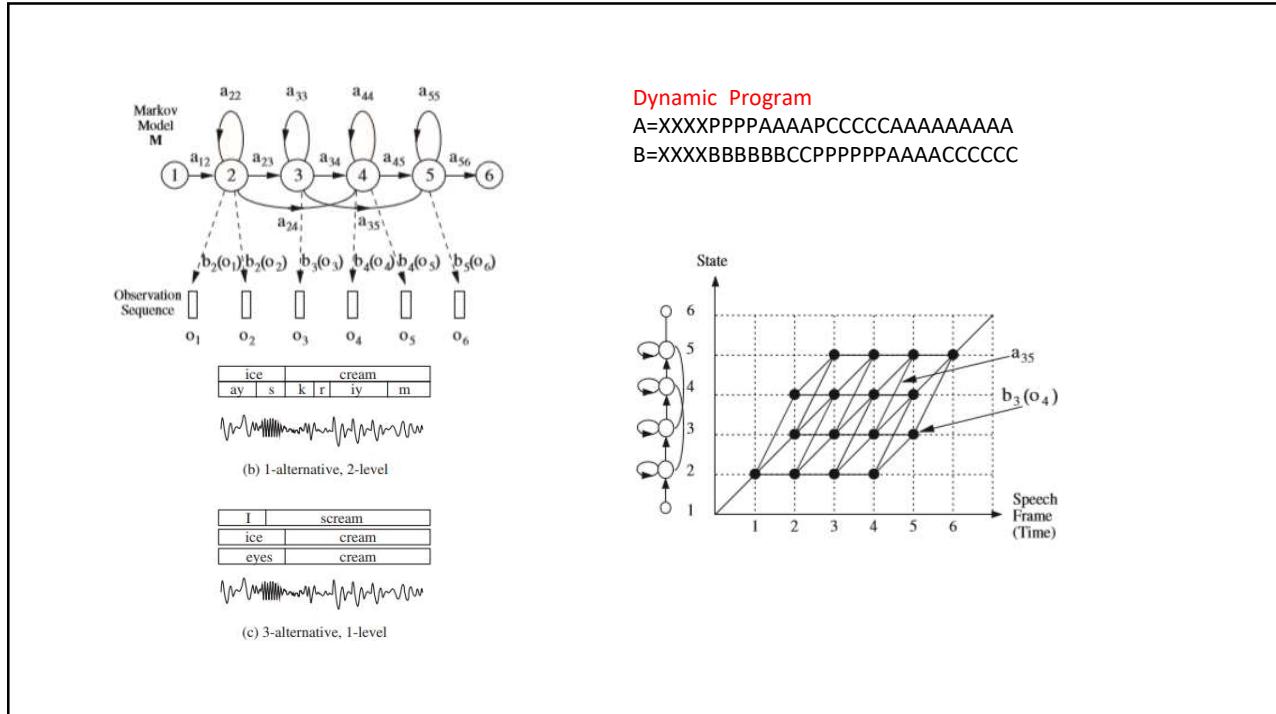
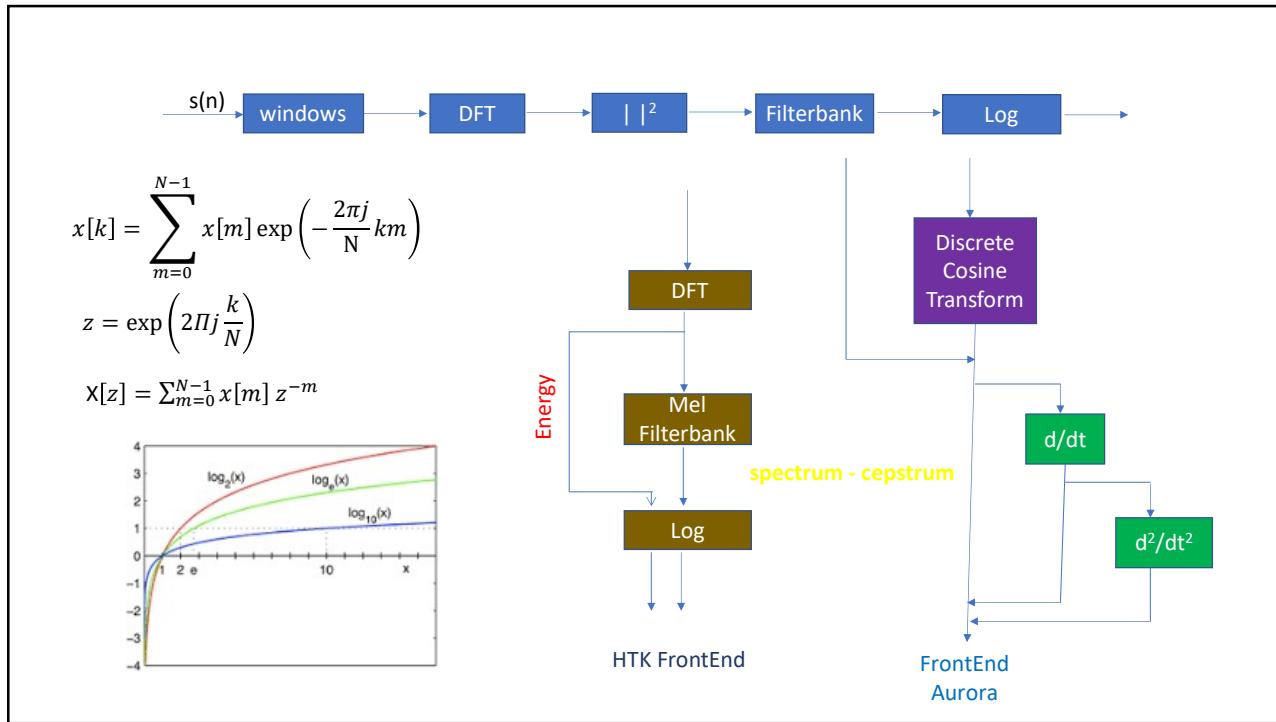
ab:craig ab:email "c.ellis@usairwaysgroup.com" .

```
# filename: ex010.rq
PREFIX ab: <http://learningsparql.com/ns/addressbook#>
SELECT ?propertyName ?PropertyValue
WHERE
{ ab:cindy ?propertyName ?PropertyValue . }

# filename: ex015.rq
PREFIX ab:
<http://learningsparql.com/ns/addressbook#>
SELECT ?craigEmail
WHERE
{
?person ab:firstName "Craig" .
?person ab:lastName "Ellis" .
?person ab:email ?craigEmail .
| propertyName | PropertyValue |
=====
| ab:email | "cindym@gmail.com" |
| ab:homeTel | "(245) 646-5488" |
-----
```

filename: ex012.ttl
@prefix ab: <http://learningsparql.com/ns/addressbook#> .
@prefix d: <http://learningsparql.com/ns/data#> .
d:i0432 ab:firstName "Richard" .
d:i0432 ab:lastName "Mutt" .
d:i0432 ab:homeTel "(229) 276-5135" .
d:i0432 ab:email "richard49@hotmail.com" .
d:i9771 ab:firstName "Cindy" .
d:i9771 ab:lastName "Marshall" .
d:i9771 ab:homeTel "(245) 646-5488" .
d:i9771 ab:email "cindym@gmail.com" .
d:i8301 ab:firstName "Craig" .
d:i8301 ab:lastName "Ellis" .
d:i8301 ab:email "craigellis@yahoo.com" .
d:i8301 ab:email "c.ellis@usairwaysgroup.com" .





$$\phi_j(t) = \max_i \{\phi_i(t-1)a_{ij}\} b_j(o_t)$$

$$\varphi_1(1) = 1$$

$$\varphi_i(1) = a_{1j} b_j(o_1)$$

$1 \leq j \leq N$ The maximum likelihood $\hat{P}(O|M)$

$$\phi_N(T) = \max_i \{\phi_i(T)a_{iN}\}$$

$$\varphi_j(t) = \max_i \{\varphi_j(t-1) + \log(a_{ij})\} + \log(b_j(o_t))$$

$D[i, j]$ khoảng cách ngắn nhất của chuỗi S độ dài i, so với chuỗi T độ dài j

1. $D[i, j] = D[i-1, j-1]$ Nếu $S_i = T_i$

Ngược lại $D[i, j] = D[i-1, j-1] + 1$

2. $D[i, j] = D[i-1, j] + 1$ cho trường hợp chèn một kí tự vào chuỗi S tại vị trí i.

3. $D[i, j] = D[i, j-1] + 1$ cho trường hợp xóa kí tự ở chuỗi P tại vị trí j.

P	T	y	o	u	-	s	h	o	u	l	d	-	n	o	t
pos	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
:	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
t:	1	1	1	2	3	4	5	6	7	8	9	10	11	12	13
h:	2	2	2	2	3	4	5	5	6	7	8	9	10	11	12
o:	3	3	3	2	3	4	5	6	5	6	7	8	9	10	11
u:	4	4	4	3	2	3	4	5	6	5	6	7	8	9	10
-:	5	5	5	4	3	2	3	4	5	6	7	7	8	9	10
s:	6	6	6	5	4	3	2	3	4	5	6	7	8	8	9
h:	7	7	7	6	5	4	3	2	3	4	5	6	7	8	9
a:	8	8	8	7	6	5	4	3	3	4	5	6	7	8	9
l:	9	9	9	8	7	6	5	4	4	4	4	5	6	7	8
t:	10	10	10	9	8	7	6	5	5	5	5	6	7	8	8
-:	11	11	11	10	9	8	7	6	6	6	6	6	5	6	7
n:	12	12	12	11	10	9	8	7	7	7	7	6	5	6	7
o:	13	13	13	12	11	10	9	8	7	8	8	7	6	5	6
t:	14	14	14	13	12	11	10	9	8	8	9	9	8	7	6

```
Command Prompt - python -m speech_recognition
Microsoft Windows [Version 10.0.17134.885]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\ngdic>python -m speech_recognition
A moment of silence, please...
Set minimum energy threshold to 202.45334734857053
Say something!
Got it! Now to recognize it...
You said Xin chào
Say something!
Got it! Now to recognize it...
You said về noi dâu Em yêu oi
Say something!
Got it! Now to recognize it...
Oops! Didn't catch that
Say something!
Got it! Now to recognize it...
Oops! Didn't catch that
Say something!
```

```
from gtts import gTTS
tts = gTTS(text='Xin chào các bạn em hát bài Diễm xưa', lang='vi')
tts.save("chao.mp3")
```

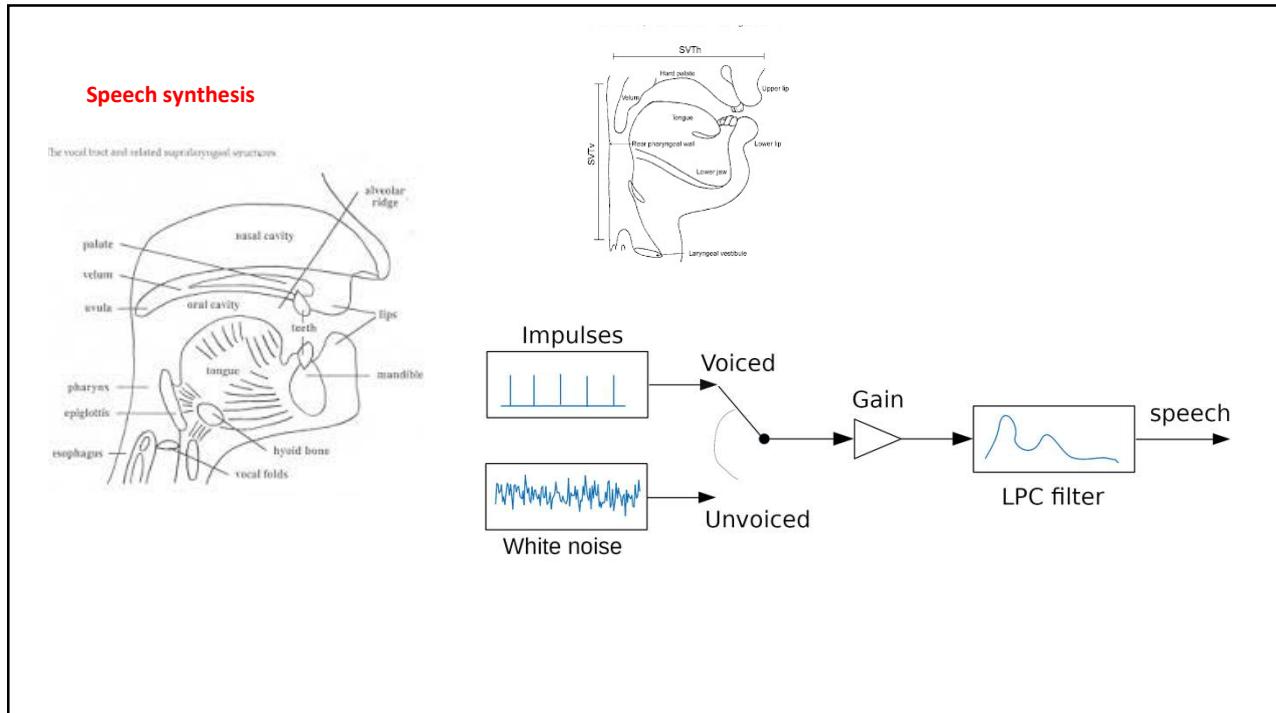
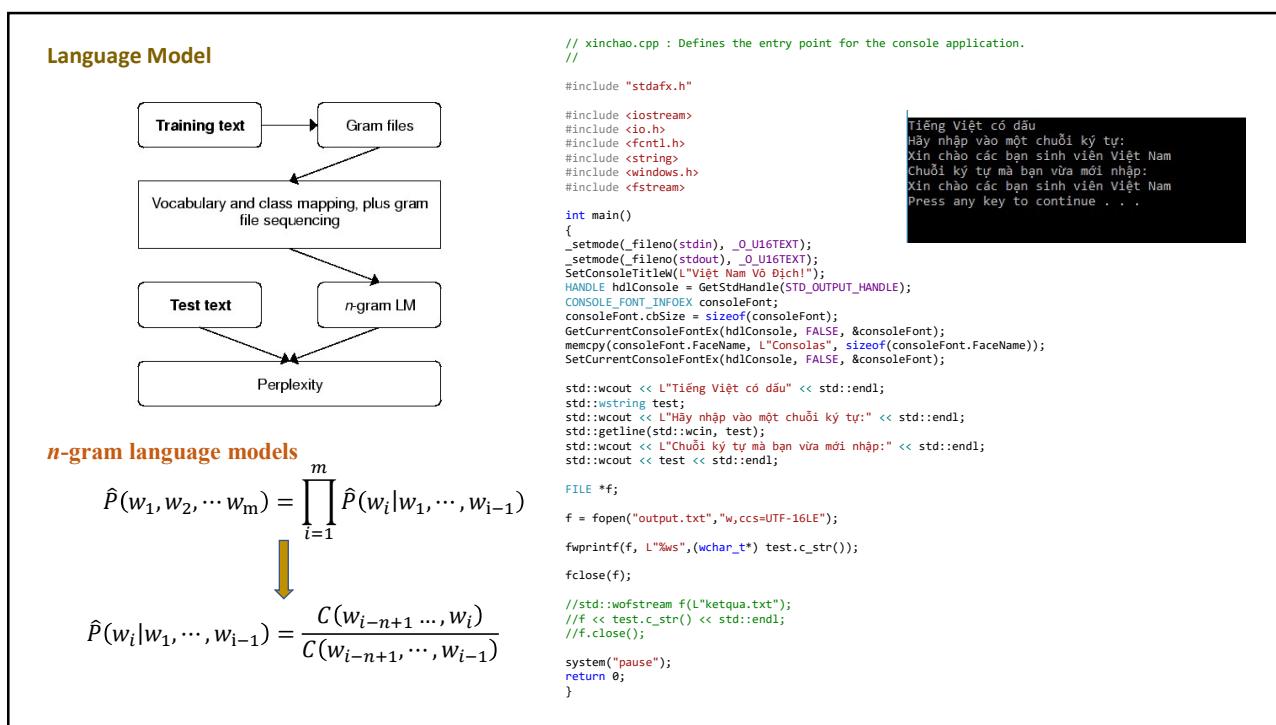


Xin chào các bạn em hát bài Diễm Xưa

```
>>> r.recognize_google(audio, show_all=True)
[{"alternative": [{"transcript": "the snail smell like old beer Mongers"}, {"transcript": "the still smell of old beer vendors"}, {"transcript": "the snail smell like old beer vendors"}, {"transcript": "the snail smell like old beer Mongers"}, {"transcript": "the snail smell like old beer mongers"}, {"transcript": "desihi smell of old beer vendor"}, {"transcript": "the still smelling old beer vendors"}, {"transcript": "bastille smell of old beer vendors"}, {"transcript": "the still smell of old beer vendor"}, {"transcript": "the still smell of old beer vendors"}, {"transcript": "the still smelling old beer vendors"}, {"transcript": "the most smell of old beer vendors"}, {"transcript": "the still smell of old beer vendor"}, {"final": True}]
```

Đánh trọng số lại transcription

```
Command Prompt - python -m speech_recognition
A moment of silence, please...
Set minimum energy threshold to 82.91760402138159
Say something!
Got it! Now to recognize it...
You said ('alternative': [{"transcript": "Xin chào", "confidence": 0.97175497}], 'final': True)
Say something!
Got it! Now to recognize it...
You said []
Say something!
Got it! Now to recognize it...
You said ('alternative': [{"transcript": "Bạn khỏe không", "confidence": 0.89232659}, {"transcript": "Em khỏe không"}, {"transcript": "Anh khỏe không"}, {"transcript": "khỏe không"}, {"transcript": "Mày khỏe không"}], 'final': True)
Say something!
Got it! Now to recognize it...
You said ('alternative': [{"transcript": "Rất vui được gặp anh rất vui được gặp anh", "confidence": 0.94021893}, {"transcript": "bác vui được gặp anh rất vui được gặp anh"}, {"transcript": "bác vui được gặp anh rất vui được gặp anh"}, {"transcript": "ác vui được gặp anh rất vui được gặp anh"}, {"transcript": "nhắc vui được gặp anh rất vui được gặp anh"}, {"final": True}]
Say something!
Got it! Now to recognize it...
You said []
Say something!
Got it! Now to recognize it...
You said []
Say something!
Got it! Now to recognize it...
You said ('alternative': [{"transcript": "Rất vui được gặp em", "confidence": 0.97175491}, {"transcript": "chắc vui được gặp em"}, {"final": True}], 'final': True)
Say something!
```



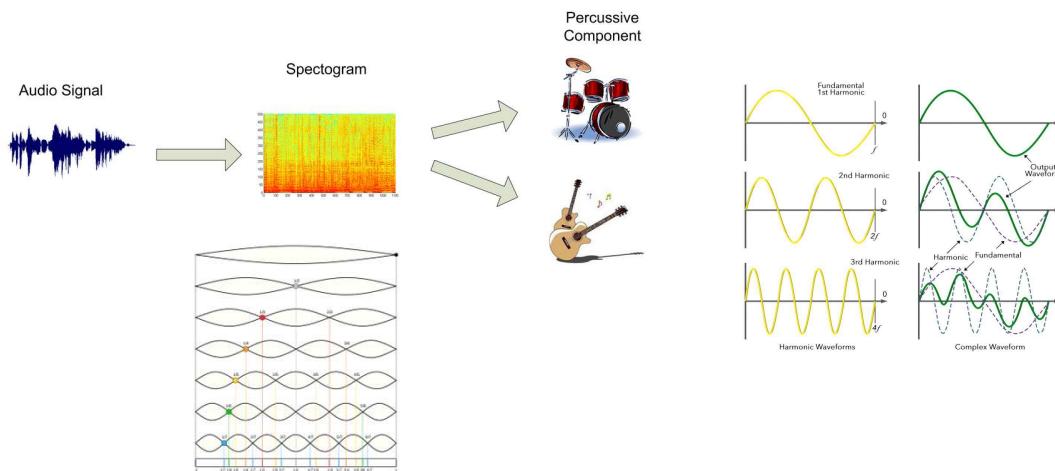
```

C:\ effectharmonics.c   runvoice.scm x
1 (voice_kal_diphone)
2 (SayText "Hello. This is a demonstration of a diphone speech synthesizer.")
3
4 (define voice_us_awb_multisyn_configure_pre nil)
5 (voice_cstr_us_awb_arctic_multisyn)
6 (SayText "Hello. This is a demonstration of a unit selection speech synthesizer.")
7
8 (voice_cmu_us_slt_arctic_hts)
9 (SayText "Hello. This is a demonstration of a hidden Markov model based speech synthesizer.")
10
11 (voice_el_diphone)
12 (SayText "En nombre de los homenajeados, habló Manuel Sánchez Salorio, que deja la jefatura de Oftalmología, en favor de la doctora, Marisa Cap")
13
14 (voice_cstr_upc_ump_spanish_hts)
15 (SayText "En nombre de los homenajeados, habló Manuel Sánchez Salorio, que deja la jefatura de Oftalmología, en favor de la doctora, Marisa Cap")
16
17

```

C:\festival\festival.exe
Copyright (C) University of Edinburgh, 1996-2010. All rights reserved.
clunits: Copyright (C) University of Edinburgh and CMU 1997-2010
clustergen_engine: Copyright (C) CMU 2005-2010
hts_engine:
Hidden Markov Model Speech Synthesis Engine "hts_engine APT"
hts_engine API version 1.04 (<http://hts-engine.sourceforge.net/>)
Copyright (C) 2001-2010 Nagoya Institute of Technology
2001-2008 Tokyo Institute of Technology
All rights reserved.
For details type '(festival_warranty)'
festival> (voice_kal_diphone)
kal_diphone
festival> (SayText "Hello. This is a demonstration of a diphone speech synthesizer")
r
<utterance 019CB150>
festival>

Percussive & harmonic content separation



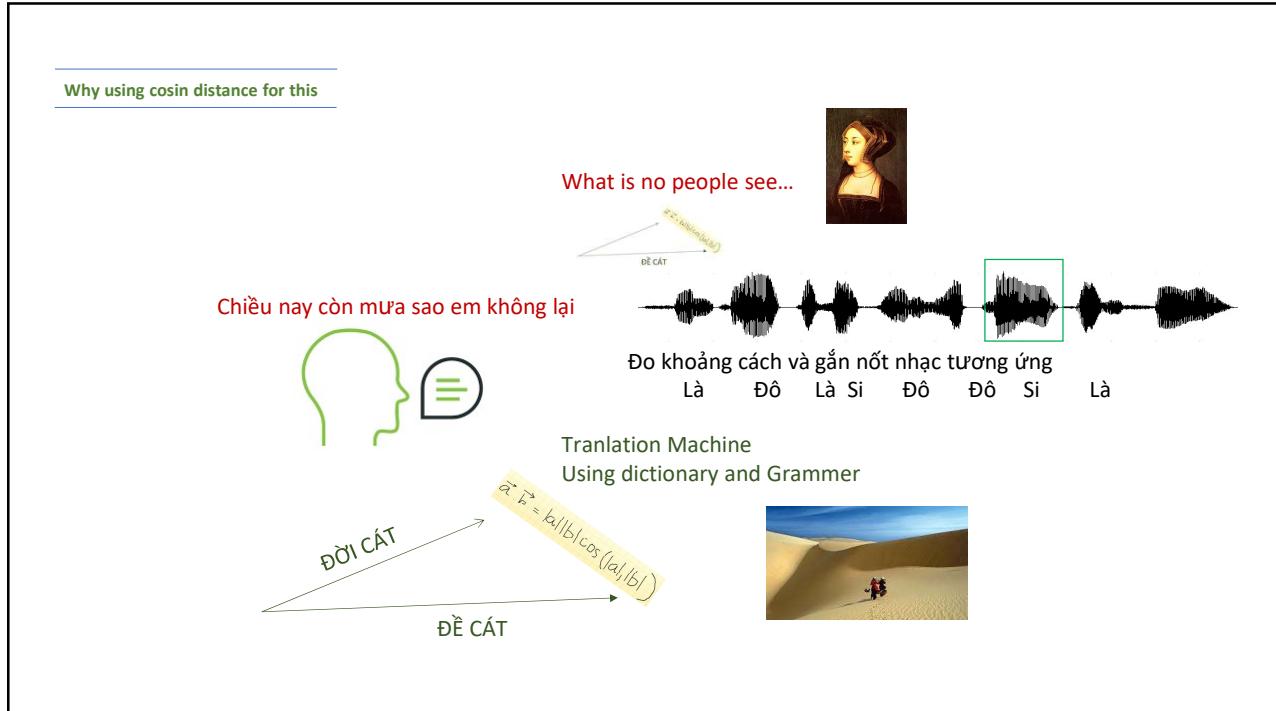
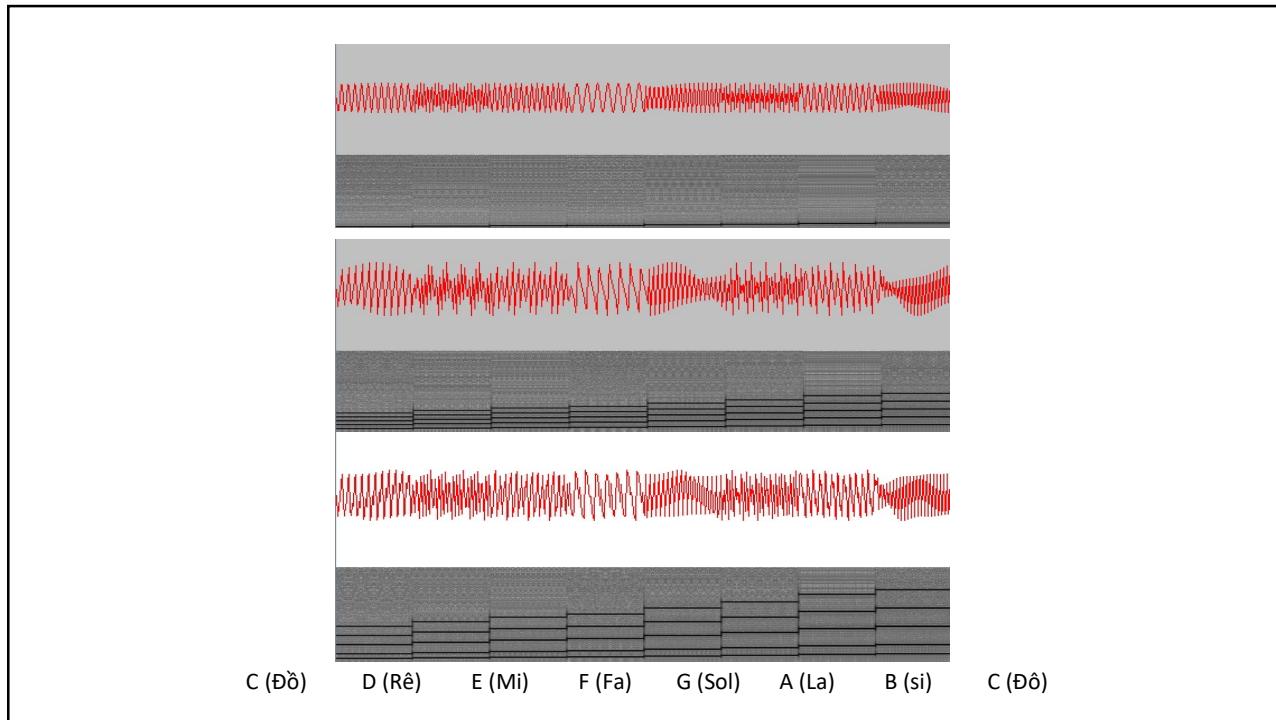
Multimedia

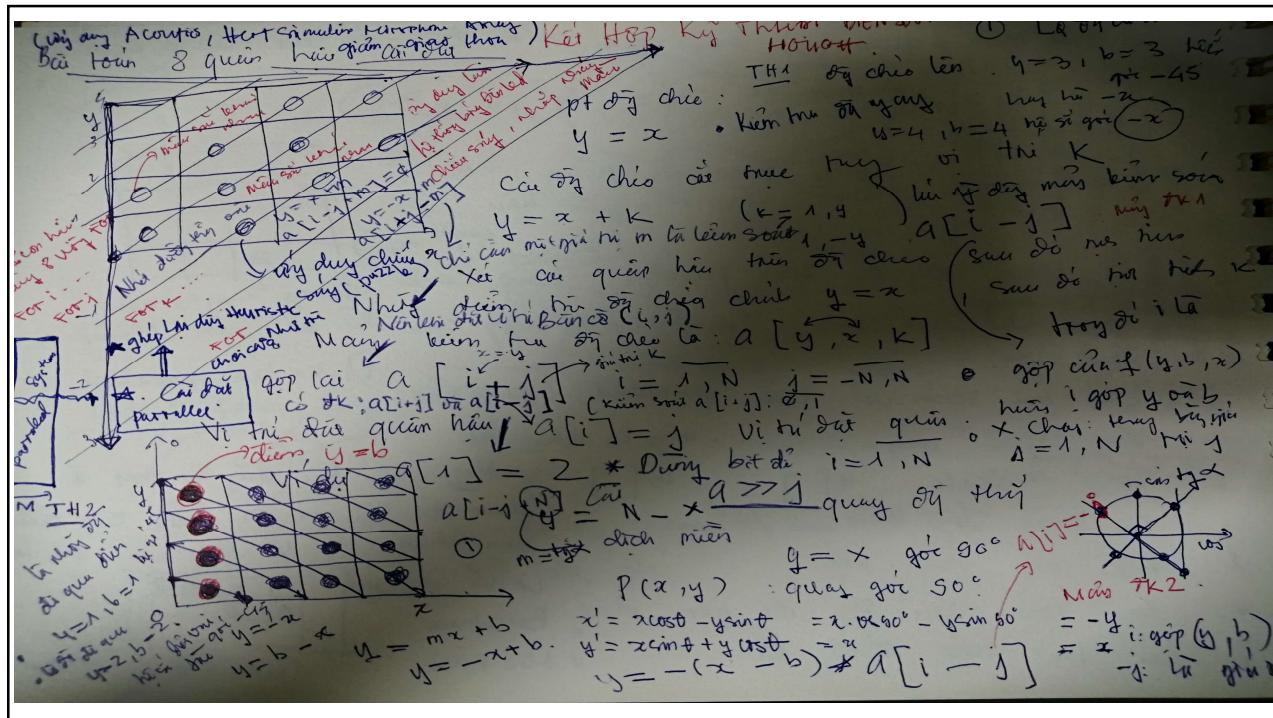
Bảng tần số

Nốt nhạc	Bát độ 3	Bát độ 4	Bát độ 5	Bát độ 6
C	131	262	523	1047
C#	139	277	554	1109
D	147	294	587	1175
D#	156	311	622	1245
E	165	330	659	1319
F	175	349	698	1397
F#	185	370	740	1480
G	196	392	784	1568
G#	208	415	831	1661
A	220	440	880	1760
A#	233	466	932	1865
B	247	494	988	1976

```
C effectharmonics.c
53     };
54     char name[100];
55     long SamplesPerSec = 11025; // 11kHz sampling
56     short BitsPerSample = 8; // 8bits
57     short Channels = 1; // mono
58     double SamplesPeriod = 1/SamplesPerSec;
59     double PlayTime=1;
60     long waveformDataSize;
61     unsigned char *waveformData;
62     double t, f;
63     long index, length;
64     int i, num, type;
65
66     num = 8; // 8~~~~~ ~~~~
67
68     length = (long)(PlayTime*SamplesPerSec*Channels*(BitsPerSample/8));
69     waveformDataSize = length * num;
70     waveformData = (unsigned char *)malloc(sizeof(char)*waveformDataSize);
71
72     type = 2;
73     for(i=0;i<num;i++){
74         f = freq[i]; // ~~~
75         for(index=0, t=0.0;index<length;index++, t+=1.0/SamplesPerSec)
76             waveformData[length*i+index] = Harmonics(type, f, t);
77     }
78
79     sprintf(name, "harmonics%02d.wav", type);
80     WriteWave(name, BitsPerSample, SamplesPerSec, Channels, waveformData, waveformDataSize);
81
82     free(waveformData);
83 }
84 |
```

Harmonica Effects





JAVA and OpenCV

1. javac -classpath .;opencv-400.jar EqualizeHistDemo.java
2. java -classpath .;opencv-400.jar EqualizeHistDemo lena.jpg

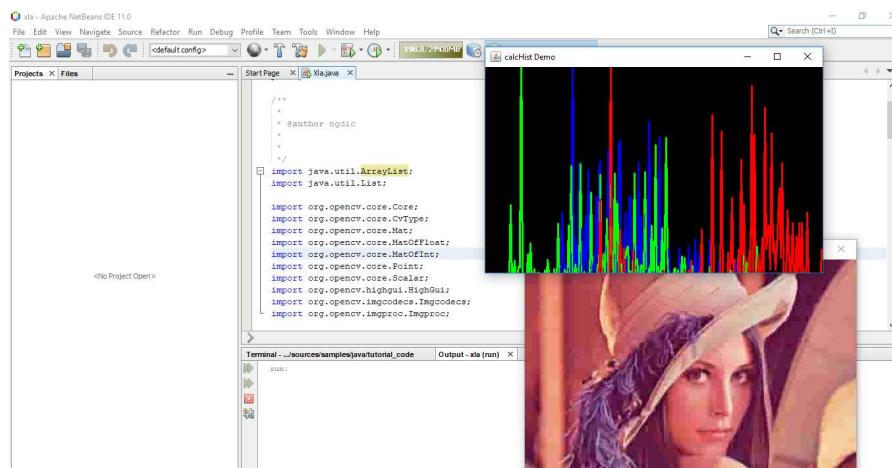
Using java for opencv

```
\$ cd opencv\sources\samples\javatutorial_code\dir1
$ javac -classpath .;opencv-400.jar EqualizeHistDemo.java
$ java -classpath .;opencv-400.jar EqualizeHistDemo lena.jpg
```



```
EqualizeHistDemo.kt
1 package org.opencv.samples;
2 import org.opencv.core.Core;
3 import org.opencv.core.Mat;
4 import org.opencv.highgui.HighGui;
5 import org.opencv.imgcodecs.Imgcodecs;
6 import org.opencv.imgproc.Imgproc;
7
8 class EqualHist {
9     public void run(String[] args) {
10         String filename = args.length > 0 ? args[0] : "lena.jpg";
11         Mat src = Imgcodecs.imread(filename);
12         if (src == null) {
13             System.out.println("Cannot read image: " + filename);
14             System.exit(0);
15         }
16         // [load image]
17
18         // [Convert to grayscale]
19         Imgproc.cvtColor(src, src, Imgproc.COLOR_BGR2GRAY);
20         // [Create equalization]
21         Mat dst = new Mat();
22         Imgproc.equalizeHist(src, dst);
23         // [Display result]
24         Highgui.imshow("Source image", src);
25         Highgui.imshow("Equalized image", dst);
26     }
27 }
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
99
```

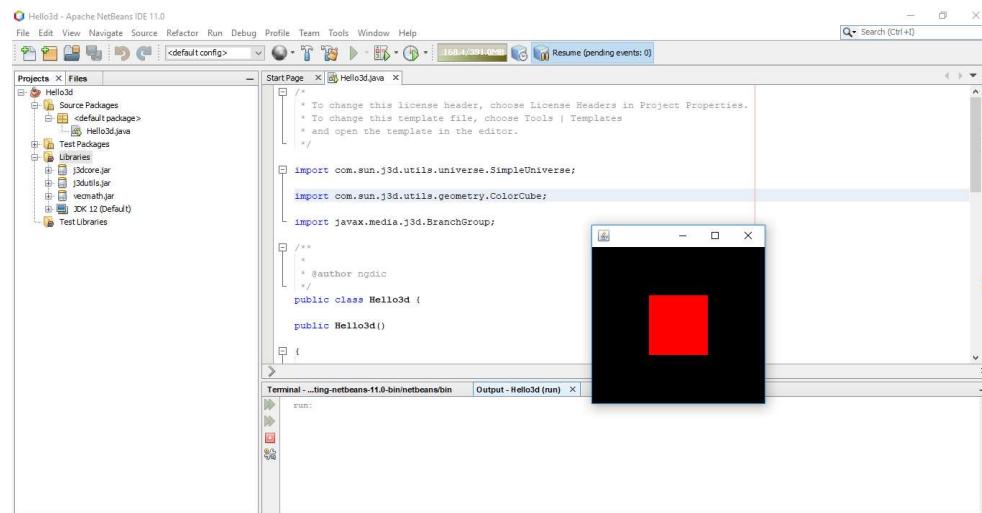
NetBean IDE



Copy Opencv-400.jar to your project and
opencv_java400.dll -> bin

```
import java3d lib, copy *.dll ->bin
```

JAVA 3D



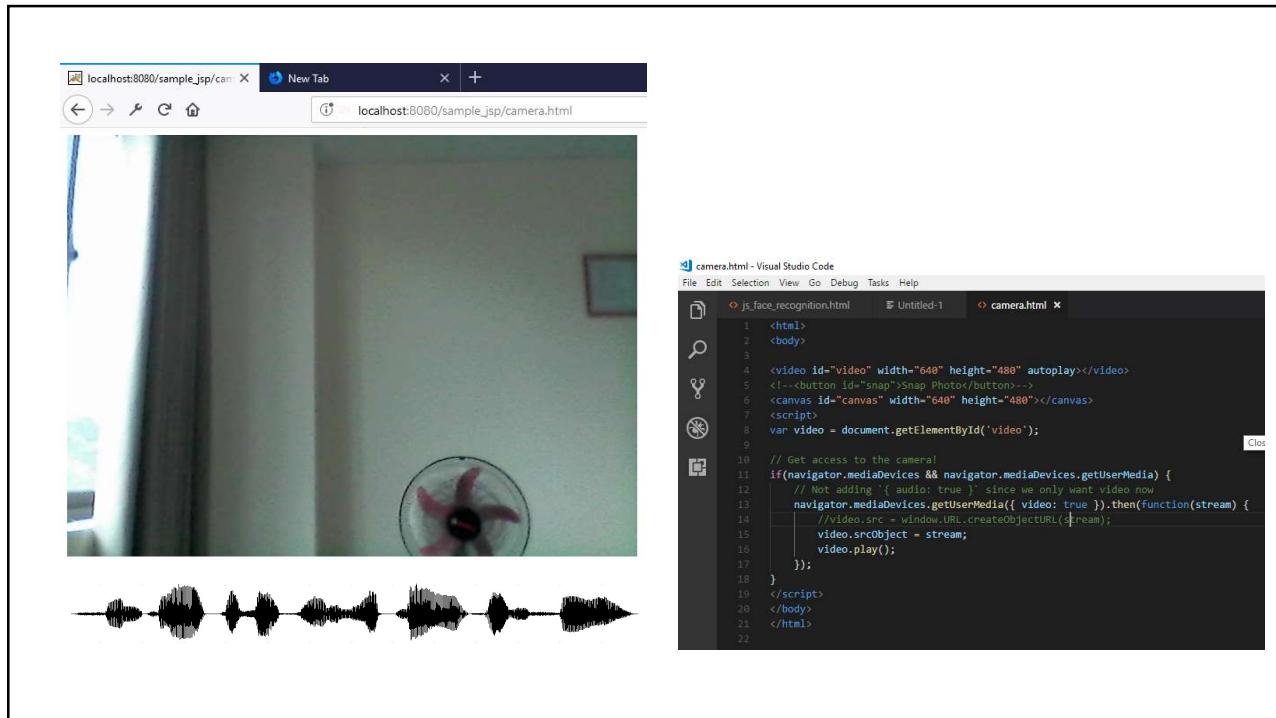
ant run

```

clean:
-verify-automatic-build:
-pre-pre-compile:
[mkdir] Created dir: C:\Users\ngdic\OneDrive\Documents\NetBeansProjects\Hello3d\build\classes
-pre-compile:
-copy-persistence-xml:
-compile-depend:
-do-compile:
[mkdir] Created dir: C:\Users\ngdic\OneDrive\Documents\NetBeansProjects
[mkdir] Created dir: C:\Users\ngdic\OneDrive\Documents\NetBeansProjects
tput
[java] Compiling 1 source file to C:\Users\ngdic\OneDrive\Documents\Ne
-post-compile:
compile:
run:
BUILD SUCCESSFUL
Total time: 16 seconds
C:\Users\ngdic\OneDrive\Documents\NetBeansProjects\Hello3d>

```

The terminal window shows the output of the "ant run" command. It includes the build steps (clean, verify-automatic-build, pre-pre-compile, etc.), the compilation of the source file, and a successful build message: "BUILD SUCCESSFUL Total time: 16 seconds".



Estimation of Feature Enhancement on Handwriting Recognition by Combination between Diagonal Based and Dense Distribution Techniques



Huynh Minh Tri^{1*}, Dinh Cuong Nguyen^{2*}

¹ Faculty of Information Technology, Ho Chi Minh City University
*Author for correspondence: cuongphu@ntu.edu.vn

1. Abstract

In this paper, we propose a novel technique called Dense Distribution for feature extraction, which approaches statistical feature. This algorithm is also combined with Diagonal based feature extraction technique to obtain a higher accuracy on handwritten English characters (include alphabets and digits). Some experiments are implemented to discover suitable parameters and methods in using these techniques. The result of correct recognition rate on MNIST dataset is 97.11% when Dense Distribution technique is 97.11% and when using the combination of Diagonal based feature extraction and Dense Distribution is 98.2%. The classification method which is used for our research is SVM (Support Vector Machine).

2. The combination of Diagonal based algorithm and Dense Distribution technique

a. Diagonal based technique: The value of each feature is calculated based on the average of numbers of foreground pixel lying on some diagonals in each zone.



Figure 1: Feature extraction for image of character using Diagonal based technique

idea alternately. In detail, we use value-binning idea for odd-order zones (or first zone, third zone, fifth zone, etc) and dense-voting idea for even-order zones (or second zone, fourth zone, sixth zone, etc).

3. Implementation

MNIST is an international standard dataset which only contains digit samples (60,000 samples for training and 10,000 samples for open testing). Meanwhile, MyCharacter database is created by ourselves. It has English upper and lower characters, and many characters (like symbols) for training and 620 samples for open testing. Both of datasets are treated as samples of handwritten characters in figure 4. These experiments were estimated on SVM machine learning using Radial Basis Function kernel and one vs all technique.

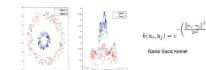


Figure 2: Dividing the range in Dense Distribution technique

- Binning number of black pixels in each zone, or in short form, called value-binning idea.
- Voting the densest subzone in each zone, or in short form, called dense-voting idea.
- Block size is a $k \times k$ square. The number of black pixels in each zone to gain P is estimated. Then, we are also easy to get the number pixels in that zone: $Q = k^2 \cdot k$.

Figure 3: Labeling for subzones, Binning number of black pixels. An example of handwritten for very distribution of black pixels.

- c. Dense Distribution combined with Diagonal based technique
- Turn 1: Each feature vector has two part: first part is taken from Diagonal based technique, second one is from value-binning idea only:

Turn 2: Each feature vector has two part: first part is taken from Diagonal based technique, second one is from value-binning idea only.

Turn 3: Each feature vector has two part: first part is taken from Diagonal based technique, second one is from value-binning idea and dense-voting idea completely.

Turn 4: Each feature vector has two part: first part is taken from Diagonal based technique, second one is from value-binning idea and dense-voting idea completely.



Figure 3: Labeling for subzones, Binning number of black pixels. An example of handwritten for very distribution of black pixels.

Figure 4: An example of handwritten data mapped using Radial Basis Kernel

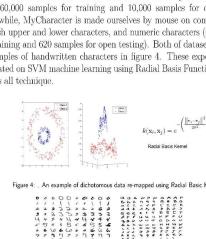


Figure 4: An example of handwritten data mapped using Radial Basis Kernel

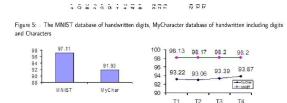


Figure 5: The MNIST database of handwritten digits. MyCharacter database of handwritten including digits and Characters

Figure 6: The correct recognition performance rate on MNIST and MyCharacter using Dense Distribution algorithm, by combination of Dense Distribution and Diagonal based through T1, T2, T3, T4 proposals.

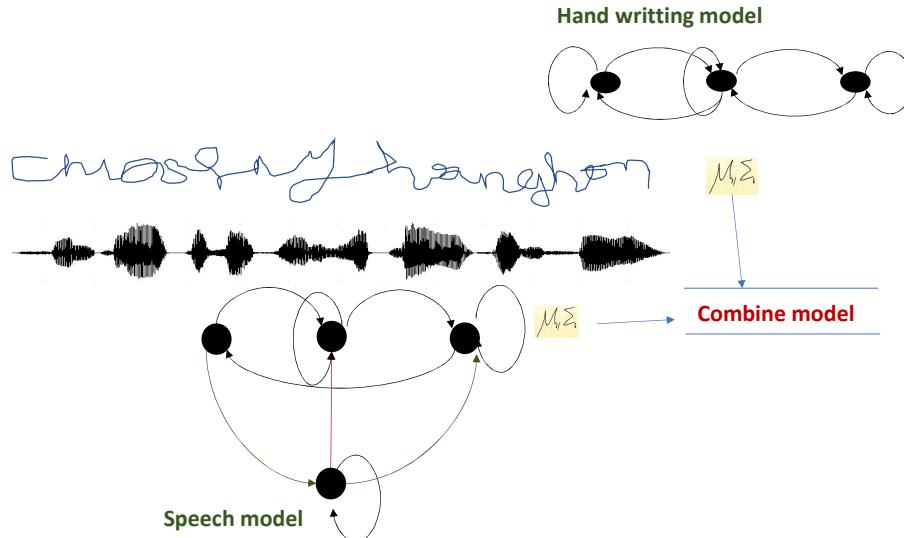
4. Conclusion

In this paper, we proposed a novel technique called Dense Distribution and also, we have proposed a combination between Dense Distribution technique and Diagonal based algorithm. The experiments results are investigated on MNIST and MyCharacter database. The correct recognition rate on MNIST is higher than on MyCharacter 5.18%. Recognition performance of Dense Distribution give 97.11% on MNIST. And we can improve correct recognition rate by the combination between Dense Distribution techniques and Diagonal based algorithm. By this way, the recognition rate give 98.2% for MNIST and 93.87% for MyCharacter database. The number of features is reduced more than 3 times which compared to raw pixel of MNIST.

Acknowledgments:

Thanks to Ho Chi Minh City University for supporting on this research.

Handwriting and speech



PHƯƠNG PHÁP QUY HOẠCH ĐỘNG 1

Ci	4	5	6	3	1	Kích thước								
Ai	3	4	5	2	1									
0	0	1	2	3	4	5	6	7	8	9	10	11	12	13
1	0	0	0	4	4	4	4	4	4	4	4	4	4	4
2	0	0	0	4	5	5	5	9	9					
3	$f(i, v) = \max_{t,i=1..n} \{f(i-1, v), f(i-1, v - a_{t,i}) + c_{t,i}\}$													
4	Tối ưu bỏ đồ vật thứ k													
5														

PHƯƠNG PHÁP QUY HOẠCH ĐỘNG 2

Giá trị b=13

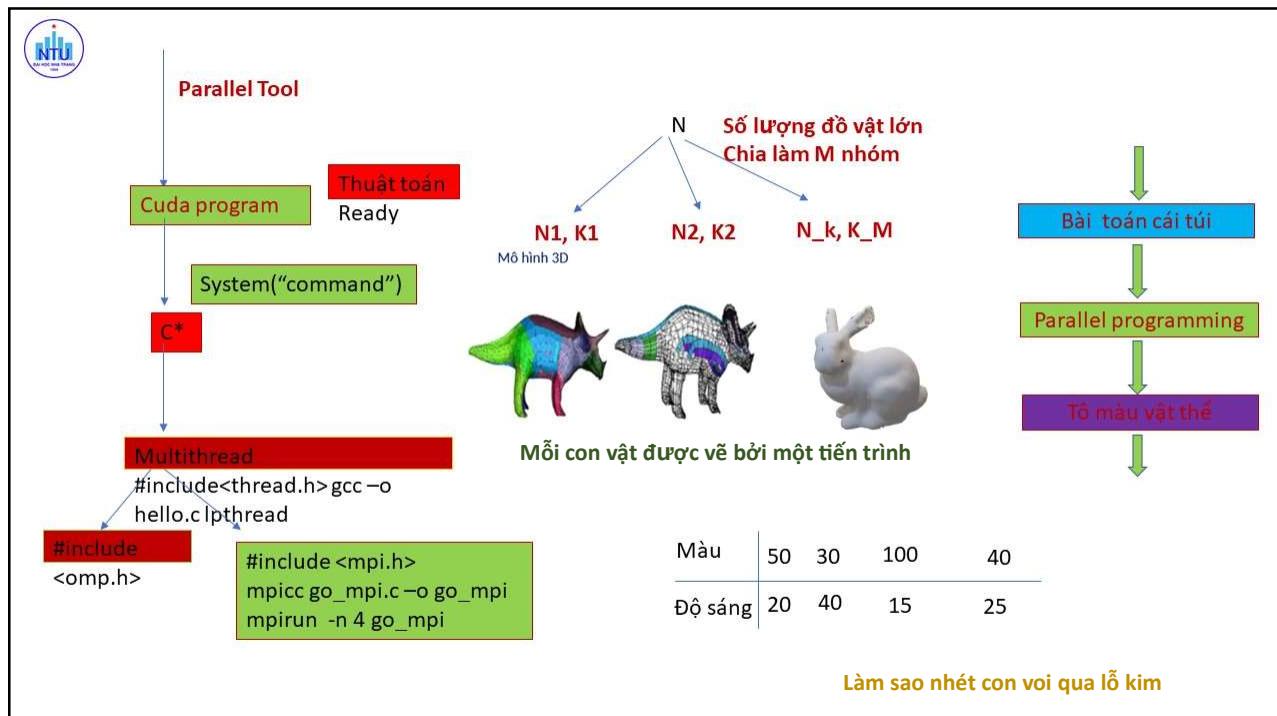
	0	1	2	3	4	5	6	7	8	9	10	11	12	13
0	∞													
1	∞	1	3	4	5	6	6	6						6
2	∞	1	3	4	5									
3	∞													
6	∞													

Bỏ 2 vật kích thước =2

$f(k, v) = \max_{t,i=1..n} \{f(k-1, v), f(k-1, v - a_{t,i}) + c_{t,i}\}$

$F(n, k, v) = F(k, v)$ Tối ưu bỏ k đồ vật

BÀI TOÁN CÁI TÚI



Python Scipy

A list of bounds

As `minimize()` works in general with x as a vector, it is natural that the "bounds" argument is a list of bound on each dimension.

What has happened? Why are we finding 0, which is not a minimum of our function?

Minimizing functions of several variables

To minimize over several variables, the trick is to turn them into a function of a multi-dimensional variable (a vector). See for instance the exercise on 2D minimization below.

Note: `scipy.optimize.minimize_several()` is a function with dedicated methods to minimize functions of only one variable.

See also: Finding minima of function is discussed in more details in the advanced chapter: Mathematical optimization: finding minima of functions.

Exercise: 2-D minimization

The six-hump camelback function

Resampling `scipy.signal.resample()`: resample a signal to n points using FFT

```
>>> t = np.linspace(0, 5, 100)
>>> x = np.sin(t)
>>> from scipy import signal
>>> x_resampled = signal.resample(x, 25)
>>> plt.plot(t, x)
[matplotlib.lines.Line2D object at ...]
>>> plt.plot(t[1:], x_resampled, 'x')
[matplotlib.lines.Line2D object at ...]
```

Notice how on the side of the spike the resampling is less accurate and has a spiking effect. This resampling is different from the interpolation provided by `scipy.signal.spline` as it only applies to regularly sampled data.

Detrending `scipy.signal.detrend()`: remove linear trend from signal

```
>>> t = np.linspace(0, 5, 100)
>>> x = np.sin(t) + np.random.normal(0, 1, 100)
>>> from scipy import signal
>>> x_detrended = signal.detrend(x)
>>> plt.plot(t, x)
[matplotlib.lines.Line2D object at ...]
>>> plt.plot(t, x_detrended)
[matplotlib.lines.Line2D object at ...]
```

Adruino

Arduino

Arduino is an open-source electronics platform based on easy-to-use hardware and software. You can use it to control motors, sensors, and other components. It's great for prototyping and learning about electronics. The Arduino Board is currently offered, therefore the documentation is no longer frequently updated.

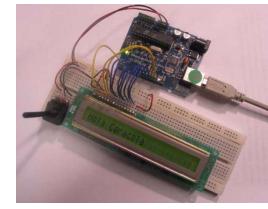
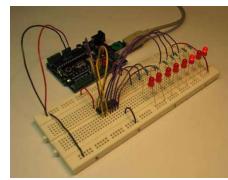
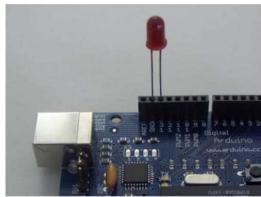
IDE

- Set up your Arduino board to go through the initial test!
- Use following: Check a working code and get your board to work.
- Connect a breadboard with the digital pins.
- Connect a breadboard with the analog pins.
- Connect a breadboard with the PWM pins.
- Connect a breadboard with the I2C pins.
- Connect a breadboard with the SPI pins.
- Connect a breadboard with the ADC pins.
- Connect a breadboard with the DAC pins.
- Connect a breadboard with the USART pins.
- Connect a breadboard with the PWM pins.
- Connect a breadboard with the I2C pins.
- Connect a breadboard with the SPI pins.
- Connect a breadboard with the ADC pins.
- Connect a breadboard with the DAC pins.
- Connect a breadboard with the USART pins.

Documentation

Documentation is available for the Arduino IDE and the Arduino Board. If you have any questions, please don't hesitate to ask!

Blog



```

#include <LiquidCrystal.h> //include LiquidCrystal library

int ledPin = 13;           // LED connected to digital pin 13

void setup()
{
    pinMode(ledPin, OUTPUT); // sets the digital pin as output
}

void loop()
{
    digitalWrite(ledPin, HIGH); // sets the LED on
    delay(1000);             // waits for a second
    digitalWrite(ledPin, LOW); // sets the LED off
    delay(1000);             // waits for a second
}

```

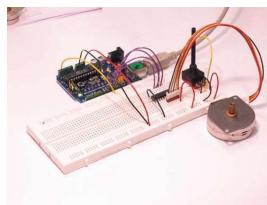
```

#include <LiquidCrystal.h> //include LiquidCrystal library

LiquidCrystal lcd = LiquidCrystal(); //create a LiquidCrystal object to control an LCD
char string1[] = "Hello!"; //variable to store the string "Hello!"

void setup(void)
{
    lcd.init(); //initialize the LCD
    digitalWrite(13,HIGH); //turn on an LED for debugging
}
void loop(void)
{
    lcd.commandWrite(2); //bring the cursor to the starting position
    delay(1000); //delay 1000 ms to view change
    lcd.printIn(string1); //send the string to the LCD
    delay(1000); //delay 1000 ms to view change
} //repeat forever

```



```

digitalWrite(motorPin1, LOW);
digitalWrite(motorPin2, LOW);
digitalWrite(motorPin3, HIGH);
digitalWrite(motorPin4, LOW);
delay(delayTime);
digitalWrite(motorPin1, LOW);
digitalWrite(motorPin2, LOW);
digitalWrite(motorPin3, LOW);
digitalWrite(motorPin4, HIGH);
delay(delayTime);
}

```

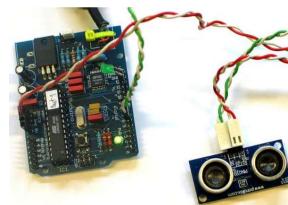
```

int motorPin1 = 8;
int motorPin2 = 9;
int motorPin3 = 10;
int motorPin4 = 11;
int delayTime = 500;

void setup() {
    pinMode(motorPin1, OUTPUT);
    pinMode(motorPin2, OUTPUT);
    pinMode(motorPin3, OUTPUT);
    pinMode(motorPin4, OUTPUT);
}

void loop() {
    digitalWrite(motorPin1, HIGH);
    digitalWrite(motorPin2, LOW);
    digitalWrite(motorPin3, LOW);
    digitalWrite(motorPin4, LOW);
    delay(delayTime);
    digitalWrite(motorPin1, LOW);
    digitalWrite(motorPin2, HIGH);
    digitalWrite(motorPin3, LOW);
    digitalWrite(motorPin4, LOW);
    delay(delayTime);
}

```



FUZZY LOGIC



Figure 2.3: Graphical representation of a conventional set and a fuzzy set

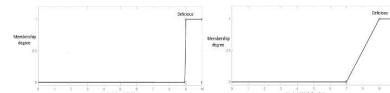


Figure 2.4: Comparison between a identity function of a conventional set and a membership function of fuzzy set

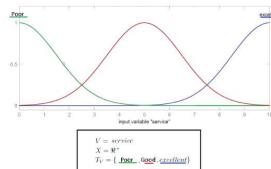
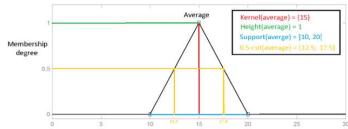


Figure 2.6: Linguistic variable 'quality of service'

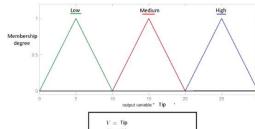
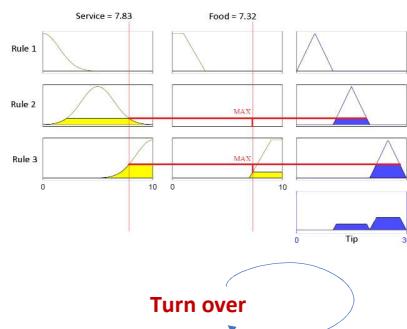


Figure 2.8: Linguistic variable 'tip amount'

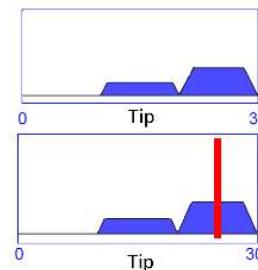
Name	Intersection AND:	Union OU:	Complement NOT:
Zadeh Operators MIN/MAX	$\min(\mu_A(x), \mu_B(x))$	$\max(\mu_A(x), \mu_B(x))$	$1 - \mu_A(x)$
Probabilistic PROD/PROBOR	$\mu_A(x) \times \mu_B(x)$	$\mu_A(x) + \mu_B(x) - \mu_A(x) \times \mu_B(x)$	$1 - \mu_A(x)$



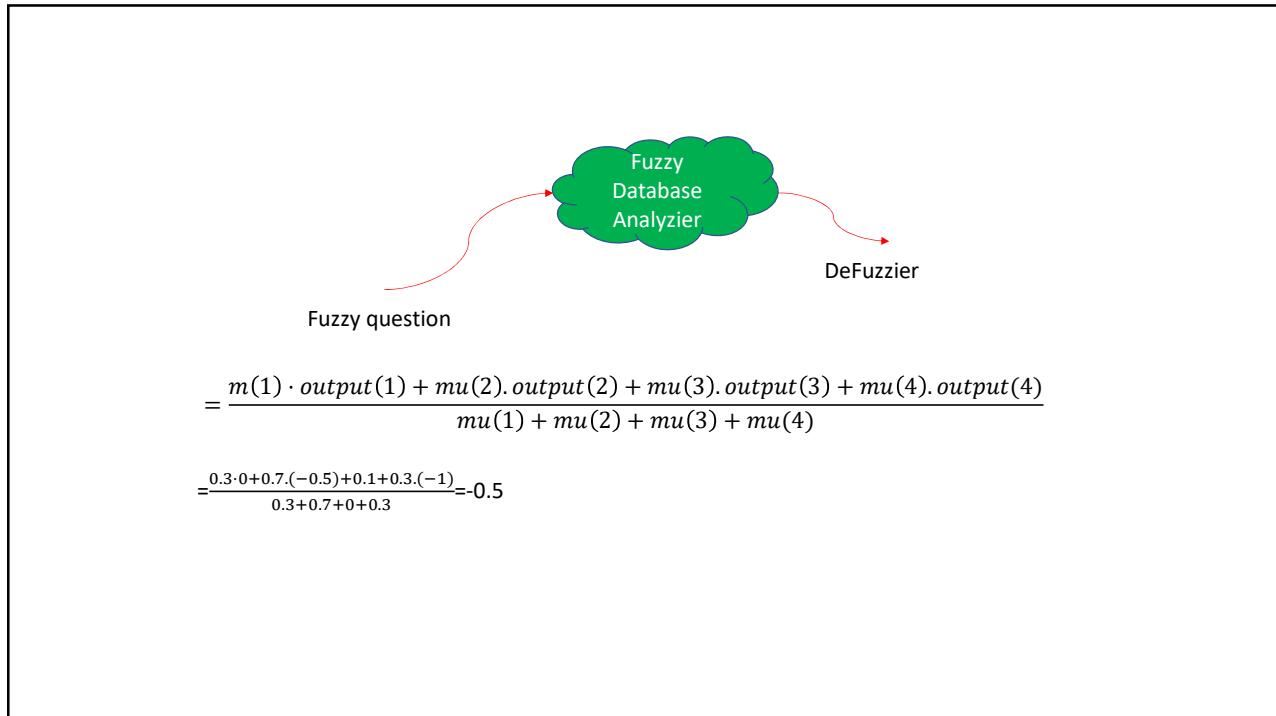
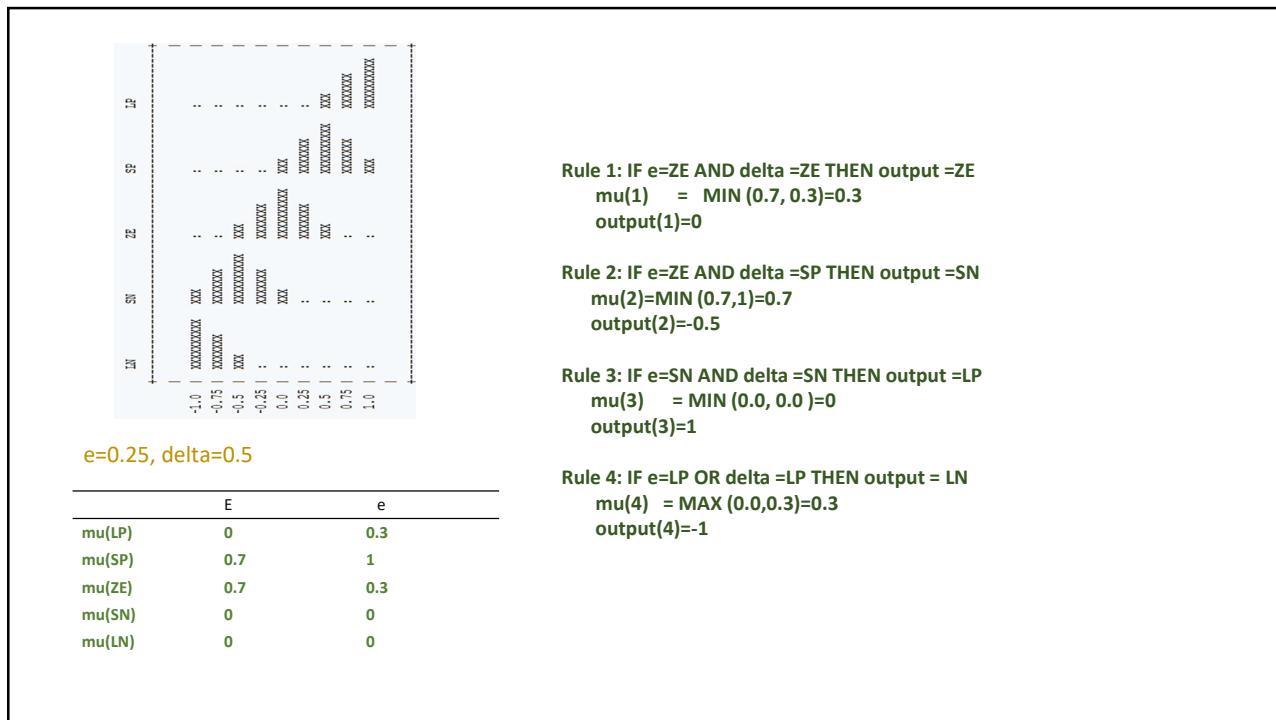
Turn over

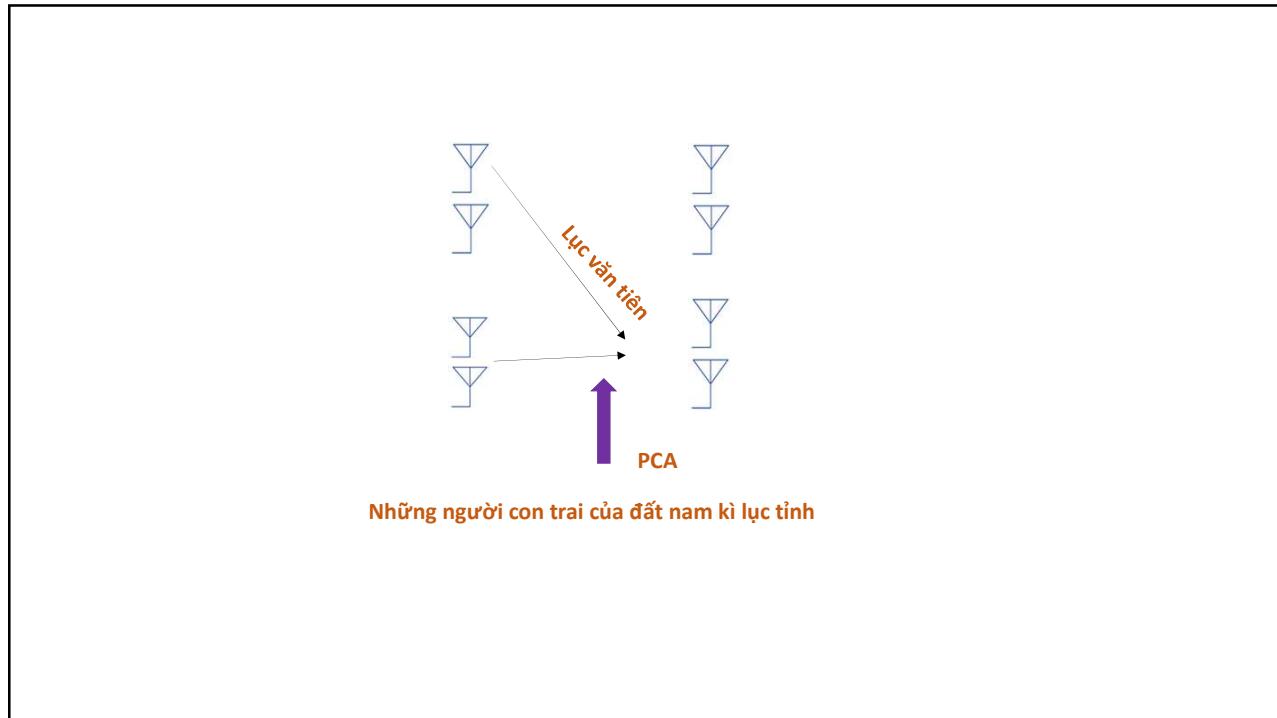
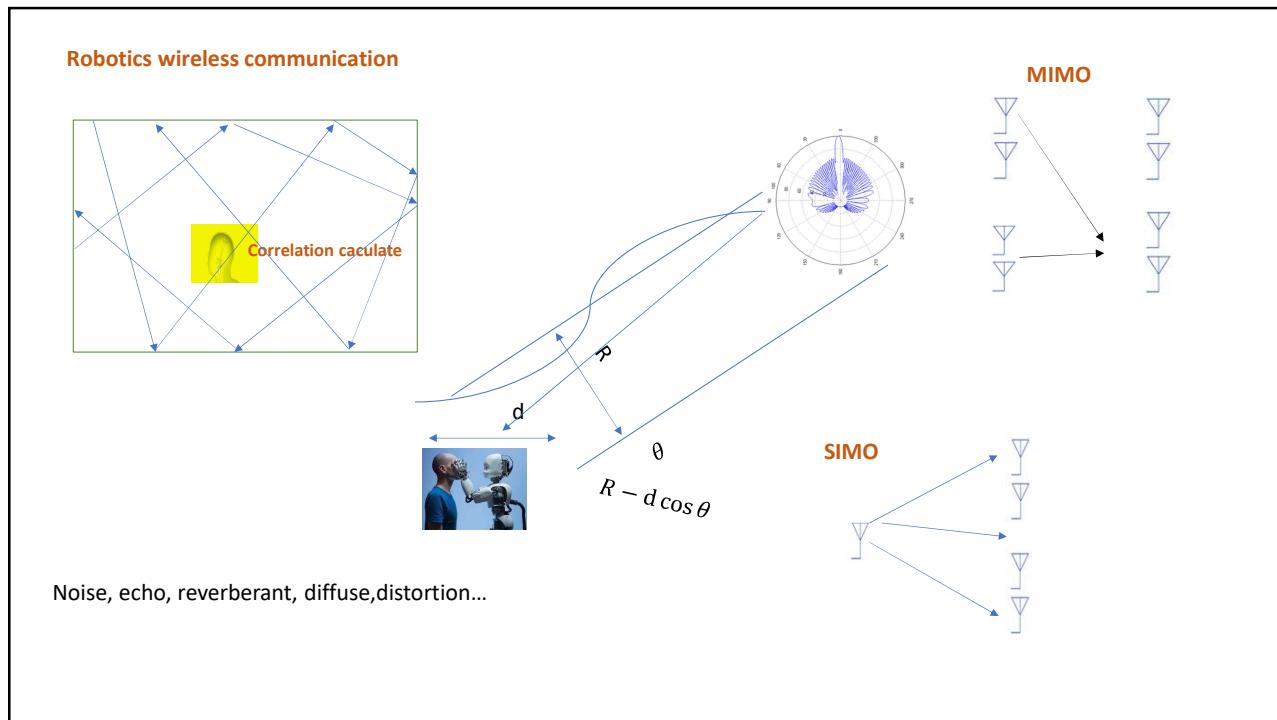
$$\text{defuzzy} = \frac{\int_S y dy}{\int_S dy}$$

$$S = \{y_m \in R, \mu(y_n) = \text{SUP}_{y \in S} (\mu(y))\}$$



$$\text{defuzzy} = \frac{\int_S y \cdot \mu(u) dy}{\int_S \mu(u) dy}$$





Viterbi algorithm

$$\phi_j(t) = \max_i \{\phi_i(t-1)a_{ij}\} b_j(o_t)$$

$$\varphi_1(1) = 1$$

$$\varphi_i(1) = a_{1j} b_j(o_1)$$

$$1 \leq j \leq N \quad \text{The maximum likelihood} \quad \hat{P}(O|M)$$

$$\phi_N(T) = \max_i \{\phi_i(T)a_{iN}\}$$

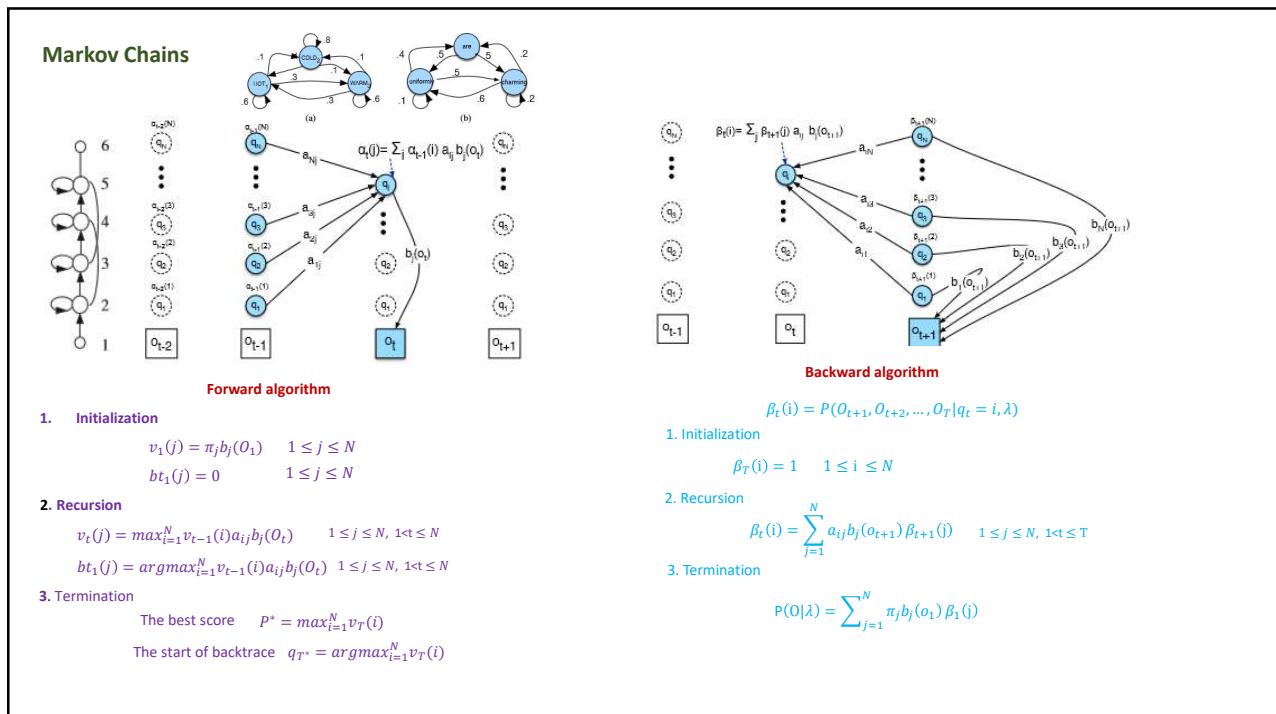
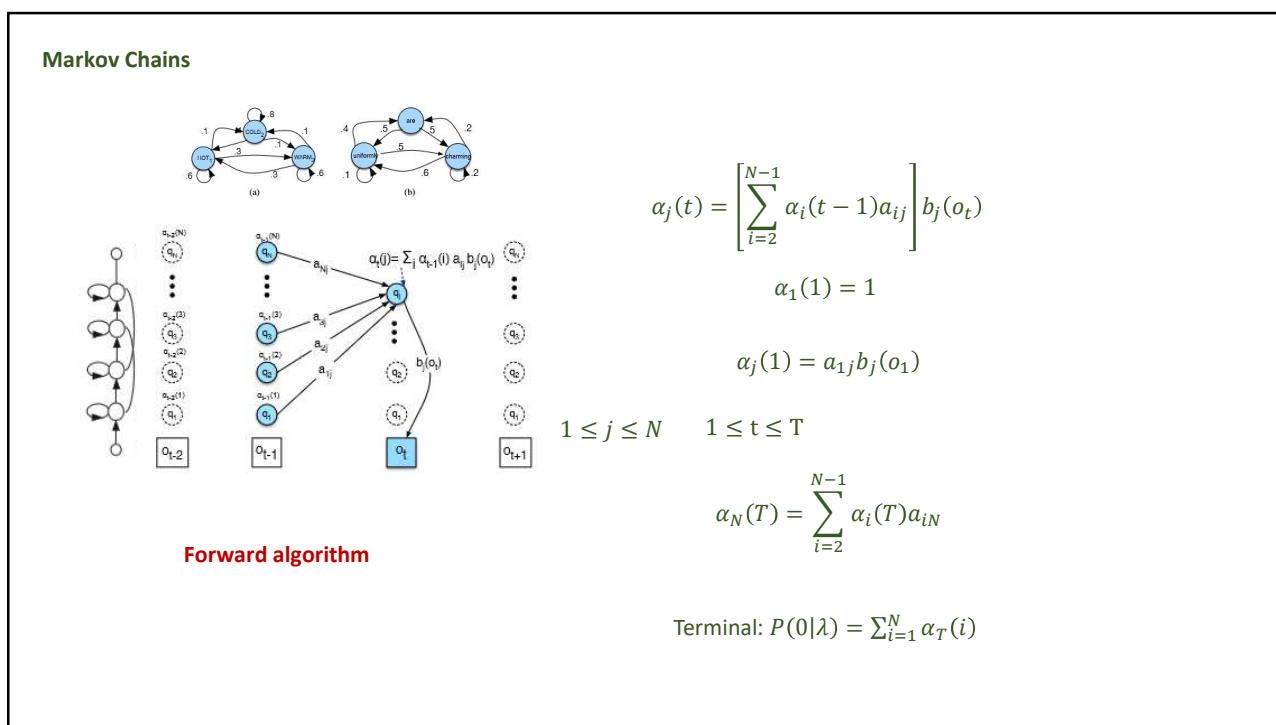
$$\varphi_j(t) = \max_i \{\varphi_i(t-1) + \log(a_{ij})\} + \log(b_j(o_t))$$

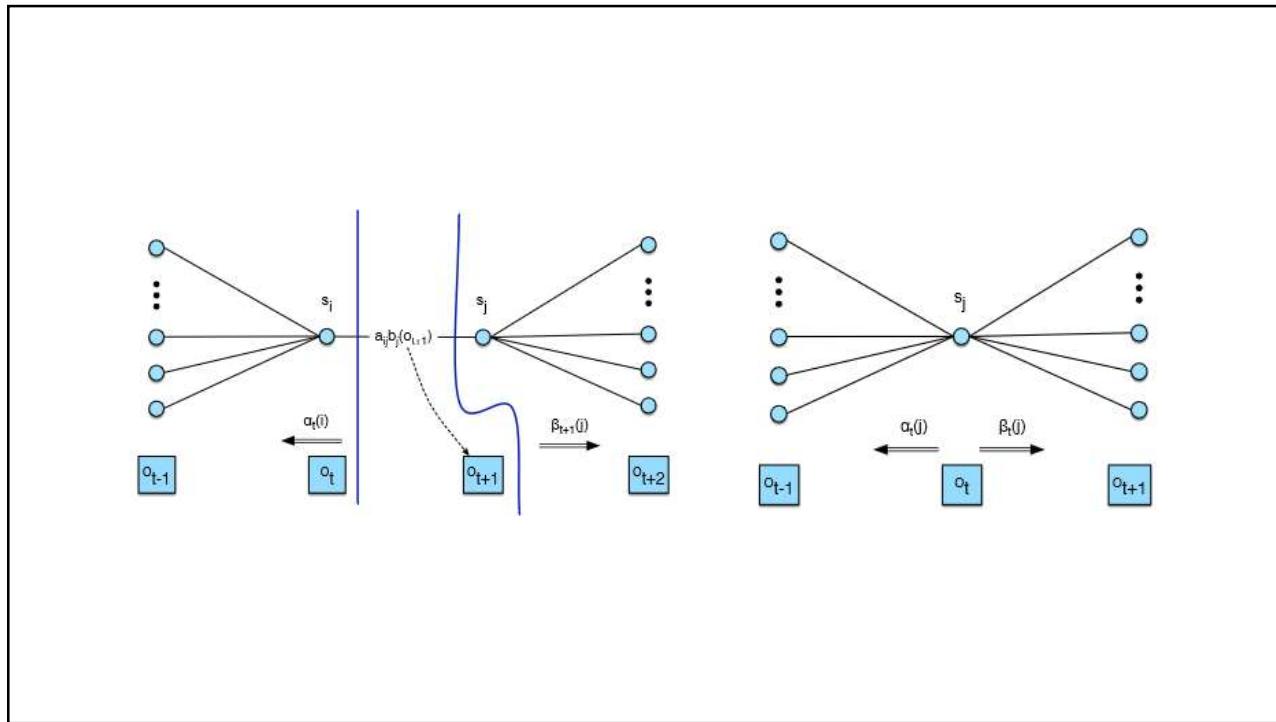
$$\hat{a}_{ij} = \frac{A_{ij}}{\sum_{k=2}^N A_{ik}}$$

$$\hat{\mu}_{ij} = \frac{\sum_{r=1}^R \sum_{t=1}^{Tr} \psi_{jsm}^r(t) o_{st}^r}{\sum_{r=1}^R \sum_{t=1}^{Tr} \psi_{jsm}^r(t)}$$

$$\hat{\Sigma}_{jsm} = \frac{\sum_{r=1}^R \sum_{t=1}^{Tr} \psi_{jsm}^r(t) (O_{st}^r - \hat{\mu}_{jsm})(O_{st}^r - \hat{\mu}_{jsm})^T}{\sum_{r=1}^R \sum_{t=1}^{Tr} \psi_{jsm}^r(t)}$$

$$c_{jsm} = \frac{\sum_{r=1}^R \sum_{t=1}^{Tr} \psi_{jsm}^r(t)}{\sum_{r=1}^R \sum_{t=1}^{Tr} \sum_{s=1}^{Ms} \psi_{jsm}^r(t)}$$



**E - step**

$$\gamma_t(j) = \frac{\alpha_t(j)\beta_t(j)}{\alpha_T(q_F)} \quad \forall t \text{ and } j$$

$$\xi_t(i, j) = \frac{\alpha_t(j)a_{ij}b_j(O_{t+1})\beta_{t+1}(j)}{\alpha_T(q_F)} \quad \forall t, i \text{ and } j$$

M - step

$$\hat{a}_{ij} = \frac{\sum_{t=1}^{T-1} \xi_t(i, j)}{\sum_{t=1}^{T-1} \sum_{i=1}^N \xi_t(i, j)}$$

$$P(O|\lambda) = \sum_{i=1}^N \alpha_T(i)$$

Chuỗi trạng thái tối ưu

$$\hat{b}_j(v_k) = \frac{\sum_{t=1, s, t, O_t=v_k}^T \gamma_t(j)}{\sum_{t=1}^T \gamma_t(j)}$$

Implementation Hmm Model

```

28
29 #include <opencv2/core/core.hpp>
30 #include "opencv2/CvHMM.h"
31
32 int main()
33 {
34     std::cout << "First we define Transition, Emission and Initial Probabilities of the model\n\n";
35     double TRANSdata[] = {0.5 , 0.5 , 0.0 ,
36                           0.0 , 0.7 , 0.3,
37                           0.0 , 0.0 , 1.0};
38
39     cv::Mat TRANS = cv::Mat(3,3,CV_64F,TRANSdata).clone();
40     double EMISdata[] = {2.0/4.0 , 2.0/4.0 , 0.0/4.0 ,
41                          0.0/4.0 , 2.0/4.0 , 2.0/4.0 , 0.0/4.0 ,
42                          0.0/4.0 , 0.0/4.0 , 2.0/4.0 , 2.0/4.0 };
43
44     cv::Mat EMIS = cv::Mat(3,4,CV_64F,EMISdata).clone();
45     double INITdata[] = {1.0 , 0.0 , 0.0};
46
47     cv::Mat INIT = cv::Mat(1,3,CV_64F,INITdata).clone();
48     CvHMM hmm;
49     hmm.printModel(TRANS,EMIS,INIT);
50
51     //-----
52     std::cout << "\nAs an example, we generate 25 sequences each with 20 observations\nper sequence\n";
53     srand ((unsigned int) time(NULL));
54     cv::Mat seq.states;
55
56     hmm.generate(28,25,TRANS,EMIS,INIT,seq.states);
57     std::cout << "\nGenerated Sequences:\n";
58
59 }
```

```

86
87 std::cout << "how do we find an optimal state sequence for the underlying\n";
88 std::cout << "Markov Process? One answer is by using Viterbi algorithm.\n";
89 std::cout << "As an example here we estimate the optimal states for all sequences\n";
90 std::cout << "using Viterbi algorithm and the defined model.\n";
91
92 cv::Mat estates;
93 std::cout << "\n";
94 for (int i=0;i<seq.rows;i++)
95 {
96     std::cout << i << ": ";
97     hmm.viterbi(seq.row(i),TRANS,EMIS,INIT,estates);
98     for (int j=0;j<estates.cols;j++)
99         std::cout << estates.at<int>(0,j);
100     std::cout << "\n";
101 }
102 std::cout << "\n";
103 //-----
104 std::cout << "\nProblem 3: Given an observation sequence O (can be several observations),\n";
105 std::cout << "how do we find a model that maximizes the probability of O ?\n";
106 std::cout << "The answer is by using the Baum-Welch algorithm to train a model.\n";
107 std::cout << "To demonstrate this, initially we define a model by guess\n";
108 std::cout << "and we estimate the parameters of the model for all the sequences\n";
109 std::cout << "that we already got.\n";
110
111 double TRUESdata[] = {2.0/3.0 , 1.0/3.0 , 0.0/3.0,
112                         0.0/3.0 , 2.0/3.0 , 1.0/3.0,
113                         0.0/3.0 , 0.0/3.0 , 3.0/3.0};


```

```

C:\openCV\resources\examples\cpp\hmm
First we define Transition, Emission and Initial Probabilities of the model

TRANS:
0.5 0.5 0
0.0 0.7 0.3
0.0 0.0 1.0

EMIS:
0.5 0.5 0.0
0.0 0.5 0.5
0.0 0.0 5.0

INIT:
1.0 0

As an example, we generate 25 sequences each with 20 observations
per sequence using the defined Markov model

Generated Sequences:
0: 11111111111111111111111111111111
1: 11111111111111111111111111111111
2: 11111111111111111111111111111111
3: 11111111111111111111111111111111
4: 2222222233332233223323
5: 00023232223322223323
6: 22233323332223232323
7: 00023222223322332233
8: 00023222223322332233
9: 222321211233332233
10: 33121211211233332233
11: 33121211211233332233
12: 33121211211233332233
13: 222333233322332323
14: 222333233322332323
15: 222323233322332323
16: 222323233322332323
17: 213333232223333223
18: 2223233323332333222
19: 2223233323332333222
20: 333333332223333322

```

As an example here we estimate the optimal states for all sequences using Viterbi algorithm and the defined model.

```

0: 012222222222222222
1: 011111111111111111
2: 011122222222222222
3: 011222222222222222
4: 011222222222222222
5: 011111111111111111
6: 222222222222222222
7: 000111111111111111
8: 000111111111111111
9: 011111111111111111
10: 011111111111111111
11: 011111111111111111
12: 011111111111111111
13: 011111111111111111
14: 011111111111111111
15: 222222222222222222
16: 011111111111111111
17: 011111111111111111
18: 011111111111111111
19: 001222222222222222
20: 001222222222222222
21: 012222222222222222
22: 012222222222222222
23: 012222222222222222
24: 001222222222222222

```

Problem 3: Given an observation sequence O (can be several observations), how do we find a model that maximizes the probability of O ? To demonstrate this, initially we define a model by guess and we estimate the parameters of the model for all the sequences that we already got.

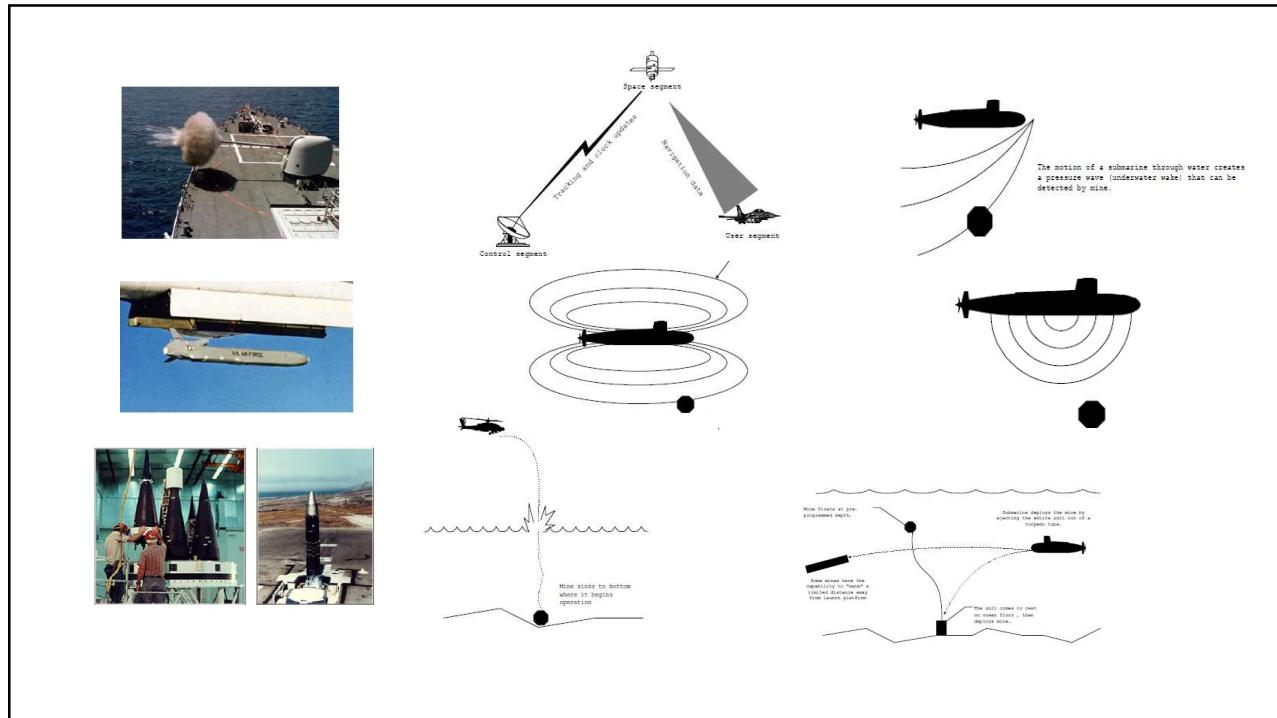
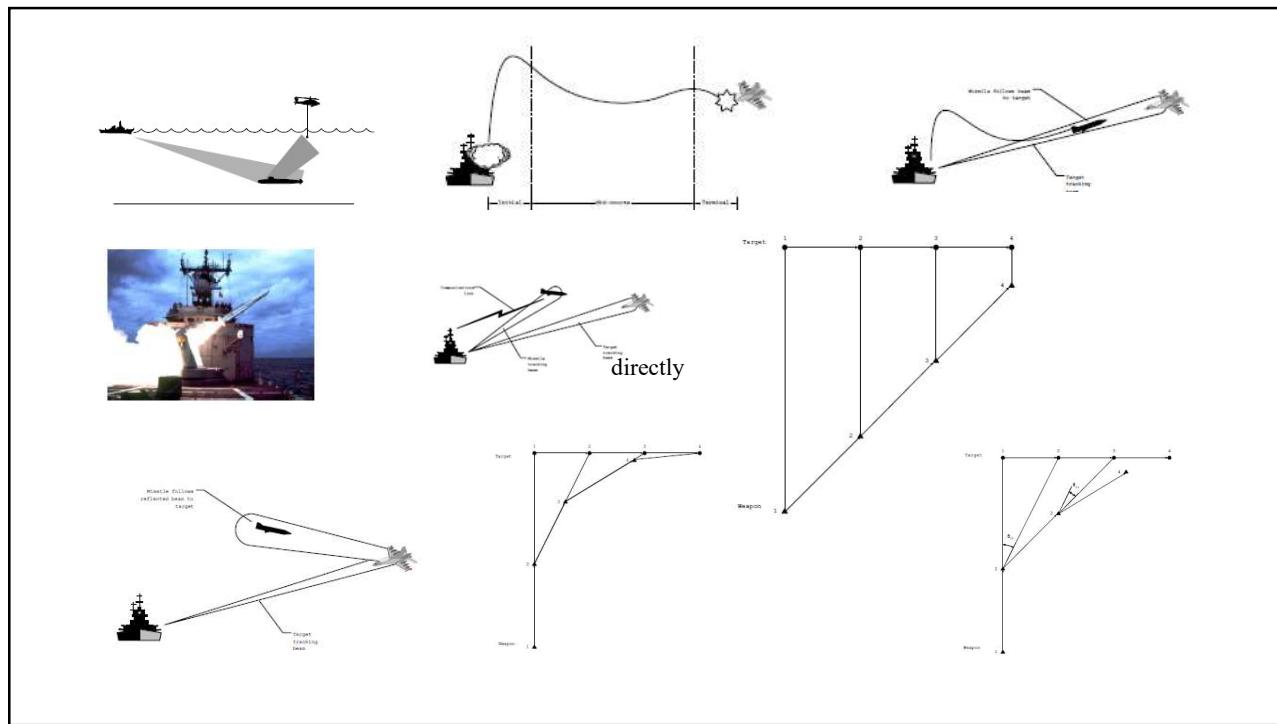
```

TRANS:
0.49579 0.370421 1.25337e-20
5.72968e-31 0.606888 8.395912
5.94563e-30 7.39661e-31

```

EMIS:

0.0025 0.504631 0.31586 0.117809



Future Attack Scenarios

High Value Target Tactical Target Target

Direct Attack Tactical Attack Threat Avoidance

Conventional Strike

Early Warning Radar SAM Site SAM Site SAM Site Target

Range of Detection

Radar's waves diminish at rate of $1/R^2$

$$\text{Range} = \sqrt[4]{\frac{\text{Power at Receiver} (\text{Square Antenna Area}) \text{ RCS}}{4 \pi (\text{Wavelength Squared})}}$$

Long Range Early Warning Radar

Lower Frequencies Higher Frequencies

Contributors to Radar Scattering

Gaps and Breaks in Skin Large Right-Angled Tail Surfaces
Protruding Antenna Exposed Engine Faces
Unshielded Canopy Non-Swept Wing Design
Protruding Gun External Weapons with Right-Angled Surfaces

Source: Richardson, 1989

Navy F/A-18E/F

Diffraction from Aircraft

Scattered Waves Creep Around Tubular Surface
Scattering From Sharp Corners
Scattering From Conical Point

- Radar Waves Grazing a Circular Structure can Creep Around its Circumference
- Waves Striking a Sharp Edge are Scattered (Diffraction)

Traveling Waves

- Waves strike surface at very slight grazing angle and propagate waves on surface
- When discontinuities occur, waves scatter
- Gap or Crack in Surface
- Surface Slope
- Change in Material
- Swept Edge Angle Scatter Away from Sender
- Trailing Wave Transmits Waves from Back

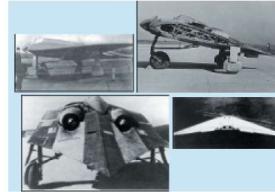
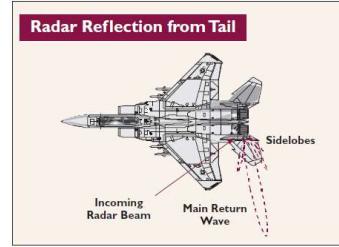
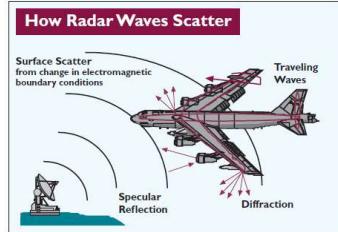
Radar Wave Absorption

Geometrical Absorber

- Wave
- Knife edge scatters waves back to receiver
- Serrated edge directs waves away from receiver

Material Absorber

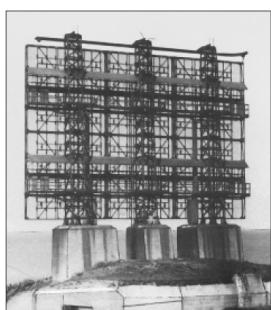
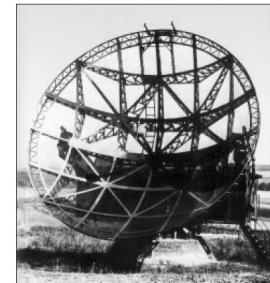
- Aircraft Skin
- Reflective Inner Surface
- Dielectric



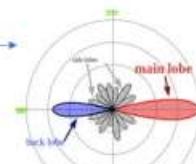
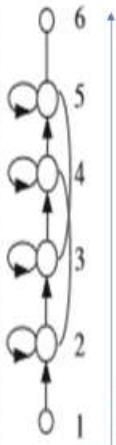
German Horten Aircraft



Lockheed SR-71



Cô ống vua trẻ xuất quân qua rừng đẹp quân xâm lấn



Quân vương giữa hoa rừng lòng bàng hoàng nhớ người thương

THANK YOU