Genomic Sequence alignments and its application for Computing Linear Structure Similarity

조환규 교수, 황미녕, 강은미, 이미경
{hgcho, mnhwang,emkang,mklee}
@pearl.pusan.ac.kr
부산대학교 공과대학
정보 컴퓨터 공학부

Graphics Application Lab

Graphics Application Lab

Biology and Informatics

- **Comutational Bio-problems**
  - o 1 hour problem, 3 hour problem
  - o 1 day problem
  - o 1 week problem
  - o 1 month problem, multi month problem
  - o 1 year problem
- **Algorithmic Problems**
  - o Formal definition
  - o Formal evaluation metric
- **How to describe**

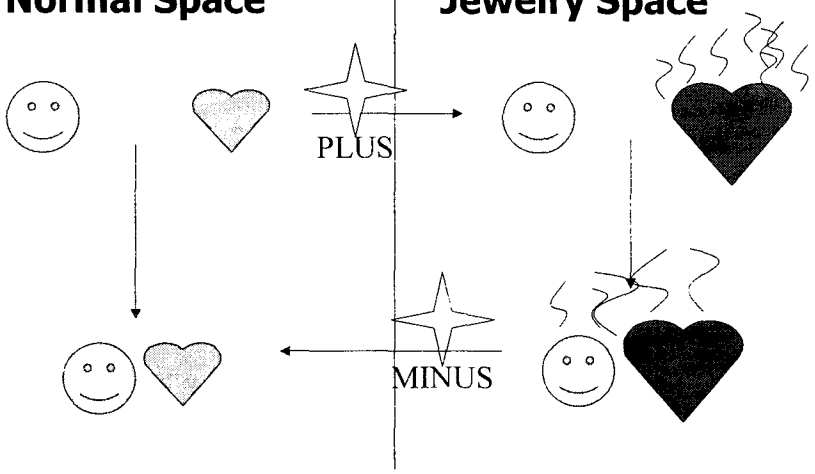
Main features of This Talk

- 이미 잘 정리된 **Computing method**를 어떻게 **Bioinformatics**에서 활용하는지
- **Bioinformatics**에서 잘 정리된 방법론은 **CS**쪽에서 어떻게 활용하는지
- **CS**와 **Bioinformatics**가 어떻게 연관되어있는지
- **Case Study)**
  - Genomic sequencing alignment와 program-copy detection과의 연관

Graphics Application Lab

Computing Space Transform

- **Normal Space** **Jewelry Space**



Graphics Application Lab

우물론

- 한 우물을 팔 것인가 **?**
  - o 만일 끝끝내 물이 안 나올 경우라면

여러 우물을 적절히 살펴가며 팔 것인가 **?**

- o 이것도 저것도 아니라면 ?
- o 이미 파본 우물인지 어떻게 판단 ?

- 그렇다면 우물을 어떻게 팔 것인가 **?**
- 많은 **bio-computing problem**의 기본모델은 이미 **CS theory**쪽에서 잘 정리되어 있다.

Graphics Application Lab

Genomic Sequence Alignments

- **Basic Assumption**
- **Why Alignment ?**
  - o Similar face – similar behavior ?
  - o Similar genotype – similar phenotype ?
  - o Dissimilar face – dissimilar behavior ?
- **"Similarity" and "distance"**
- **Alignment Category**
  - o Pair-wise alignment : Multiple alignment
  - o Global alignment : Local alignment
  - o Optimal alignment : Heuristic alignment

Graphics Application Lab

Graphics Application Lab

Dynamic Programming

- **A Basic tool for all kinds of alignment**
- **Find a good path to "De-Jeon"**
  - o If you give a good path to 조치원,
  - o If you give a good path to XXX.....near to Dejeon
  - o Then I will give the best way for De-jeon

Graphics Application Lab

Dynamic Programming

- **A programming Methodolgy**
  - o programming with dancing ?
  - o Solution from all sub-partial solution
- 준비물
  - o Objective function
  - o Dynamic programming formula(recursion)
    - **F(n) = F(n-1) + F(n-2), F(0)=F(1)=1**
  - o Base condition
  - o Table, multi-dim array in language
- **Space complexity!**

7

Graphics Application Lab

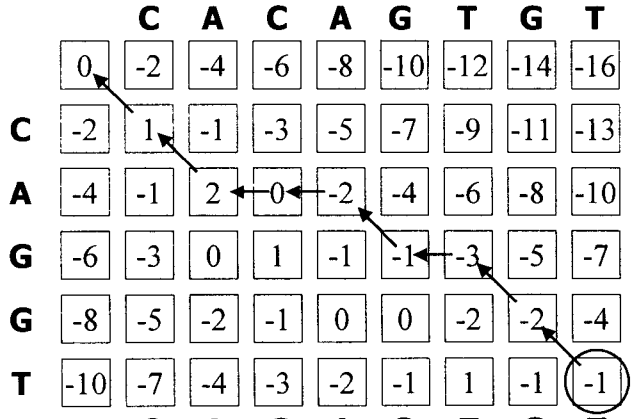
Global Alignment(1)

- **Basic scoring:**
  - o Match: 1, Mismatch: -1, Space: -2
- **How?**
  - o To find the alignment of two sequences of maximal score
  - o Sequence alignment problem corresponds to the *longest path problem form* the source to the sink in this *directed acyclic graph*.

Graphics Application Lab

Global Alignment(2)

- **CACAGTGT 와 CAGGT**



Graphics Application Lab

Local Alignment(1)

- **An alignment between a substring of *s* and a substring of *t***
  - o Each entry of (I,j) will hold the highest score of an alignment between a suffix of s[1…i] and a suffix of t[1…j]

예) AGGTATTGA
- CCTATGGC

Graphics Application Lab

Local Alignment(2)

- **AGGTATTG 와 CTATGC**

		A	G	G	T	A	T	T	A
	0	0	0	0	0	0	0	0	0
C	0	0	0	0	0	0	0	0	0
T	0	0	0	0	1	0	1	1	0
A	0	1	0	0	0	2	0	0	2
T	0	0	0	0	1	0	3	2	0
G	0	0	1	1	0	0	1	2	1
C	0	0	0	0	0	0	0	0	1
		A	G	G	T	A	T	T	A
		-	-	C	T	A	T	G	C

Semi-global Alignment(1)

- **Given two sequences, check if one of them has a substring similar to the other entire sequence.**
- **How?**
  - Find alignments ignoring the beginning and end spaces of the sequences
- **Global alignment** 와 비교
  - CAGCA -CTTGGATTCTCGG <-semi-global
  - - - -CAGCGTGG- - - - - - - (score: -19)
  - CAGCACTTGGATTCTCGG <-global
  - CAGC- - - - -G -T- - - -GG (score: -12)

Semi-global Alignment(2)

- **CACAGTGT** 와 **CAGGT**

		C	A	C	A	G	T	G	T
	0	-2	-4	-6	-8	-10	-12	-14	-16
C	-2	1	-1	-3	-5	-7	-9	-11	-13
A	-4	-1	2	0	-2	-4	-6	-8	-10
G	-6	-3	0	1	-1	-1	-3	-5	-7
G	-8	-5	-2	-1	0	0	-2	-2	-4
T	-10	-7	-4	-3	-2	-1	1	-1	-1

C A C A G - T G T

- - C A G G T - -

Graphics Application Lab

General Gap Penalty

- **Definition**
  - Gap: consecutive number k > 1 of spaces
  - When mutations are involved, the occurrence of a gap with k spaces is more probable than the occurrence of k isolated spaces
  - w(k) : penalty associated with a gap with k spaces

Graphics Application Lab

Affine Gap Penalty Function

- **Penalty for consecutive spaces <= isolated spaces**
- **Subadditive function**
  - w(k1 +k2+..._kn) <= w(k1) + w(k2) +...+w(kn)
- **Three arrays for dynamic programming**
  - a[i,j] = maximum score of an alignment between s[1...i] and t[1...i] that ends in s[i] matched with t[j]
  - b[i,j] = maximum score of an alignment between s[1...i] and t[1...i] that ends in a space matched with t[j]
  - c[i,j] = maximum score of an alignment between s[1...i] and t[1...i] that ends in s[i] matched with a space

Heuristic Alignment

- **Main difficulties**
  - o Search space, O(n^2) space or O(n^2 log n)time
  - o Optimality or Biologically-good Distance metric
  - o Multiple alignment
- **Local search**
  - o Diagonal region searching
  - o Visualization., e.g., Dotlet
- **BLAST approach for long sequence**
  - o Small word matching
  - o And Extending from a highly matched region

Graphics Application Lab

Multiple Alignment

- **Problem hardness:**
  - o Optimal alignment : NP-hard
  - o What if more than 1000 sequence ?
- **Pairwise alignment**
- **Star Alignment**
  - o Simple, but poor result
- **Tree alignment**
  - o A good alternative

Graphics Application Lab

PART 2: Applications

DISKETTE to HAMBURG !

Applications : Plagiarism

- **Linear Structure**
  - o Genomic sequences
  - o Plain articles
  - o Programs
  - o Human behaviors on the time-line
  - o Time-series data sets
- **Student Reports Plagiarism**
- **Assignment Program copying**
- **Where is the original version of this one ?**
- **Web searching redundancy**

Previous Approaches

- **Keyword frequency similarity**
  - o 특정한 단어의 사용횟수 = "이순신"
  - o Cosine product measure
  - o Object finger printing
    - **Fixed size fingerprint**
    - **Easy to making Database**
    - **Quick searching**
    - **High false positive rate**
    - **Does not consider Structure**
  - o Upon a commercial version
- **UNIX 'profile' command**

Graphics Application Lab

Attacking Methods and

- **Inserting some words**
- **Shuffling-invariant**
- **Easy to use in document application**
- **Hard to use in program file**
- **Recent trends**
  - o Structure-oriented similarity measure
  - o Greedy-Block-Removing methods...
  - o Is this a basic concept of local alignment ?
- **Sample-Report-Server Building**

Graphics Application Lab

Graphics Application Lab

Undergraduate PGM. reports

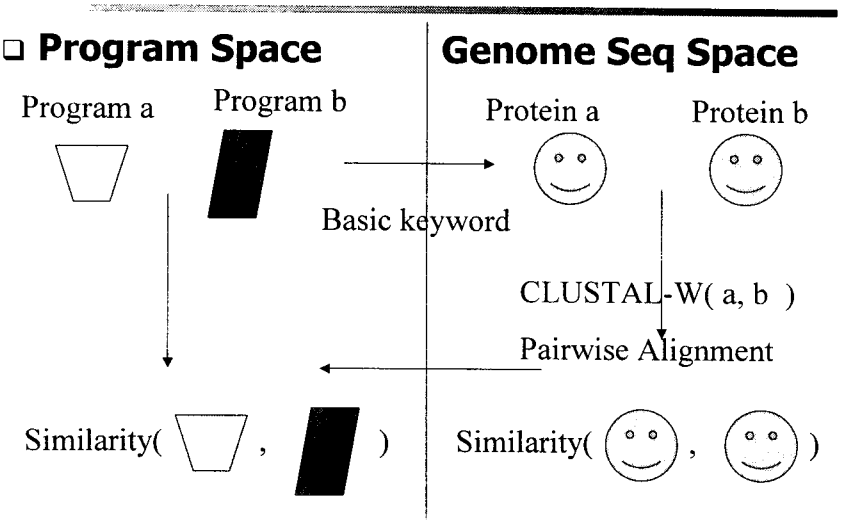
- **Programming Assignment cheating:**
- **Assignment cheating**은 원래 일보다는 어렵다.
  - o Password breaking by Mafia
- **Assignment**의 결과는 동일하다.
  - o Correct program들 끼리 비교
- 주어진 시간은 비교적 짧다
- 수강생의 수는 적절히 적다.**(300** 명 이하**)**
- 프로그래밍 언어는 모두 동일하다.

Graphics Application Lab

Program Cheating Techniques

- **Complete Copying**
- **Variable exchange**
- **Garbage code insertion**
- **Function transpose**
- **Code rewriting(partially)**
- **Library code replacing**
- **Merging different codes**
- **Function resolving**
- **Function rewriting**

Computing Space Transform



Graphics Application Lab

PROGRAM to PROTEIN

- **Program Language**
  - o Keyword = { int, float, class..... }
  - o Block Structure = "}", "{"
- **Program Chromosome**
  - o Location independent code, JAVA class, C files
- **Non-Codind region**
  - o /* this is a sample non-coding region */
- **Promoter**
  - o Variable declaration, class definition
- **DNA = kewords sequence**

Graphics Application Lab

Example

```
main( ) {
    int i, j , k ;
    .....
    for( I = 1 . I <= 100 , i++) {
        .....
        if (        ) x = y ;
        else .....
        while( ccccc ) { }
        x = 23984 ;
    } // end of for
    .....
}
```

int for if = else while =

AGTCGCTTCGAAGCAA

Graphics Application Lab

Why Protein mapping ?

- **DNA sequence overlap**
  - o if = AA, then = AG, * = GA, return = GG
  - o AAGGA = AG + GA or AA + GG + A
    - **Ambiguity resolving**
- **20 Anino acid base**
  - o About 20 keywords
  - o 2-3 groups
    - **polar, non-polar**
    - **hydrophobic, hydrophilic**
    - **Charged, uncharged**

Graphics Application Lab

Keyword Mapping Strategy

- **Convertibility = { for , while } Easy**
- **Convertibility = { for, then } Hard**
- **Convertibility = { if, '=' } Impossible**
- **Procedure**
  - o Preprocessing
  - o Chromosome arrangement
  - o Keyword selection
  - o Protein mapping

Graphics Application Lab

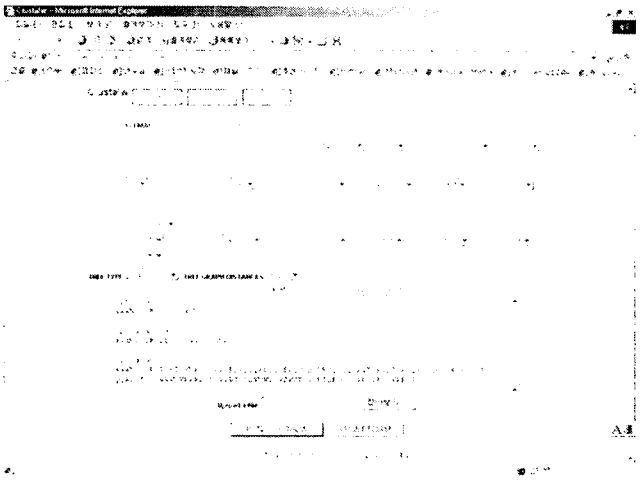
Experiment Overview

- **Sample programs <= "data structure"**
- **Students , 60**
- **Programming assignment, 12**
- **1 semester**
- **On-line evaluation system = ESPA**
  - o Java-based on-line evaluation system
  - o Due, 1 week
- **Do not monitor all programs**

Graphics Application Lab

ClusterW (1)(www2.ebi.ac.uk/clusterw)

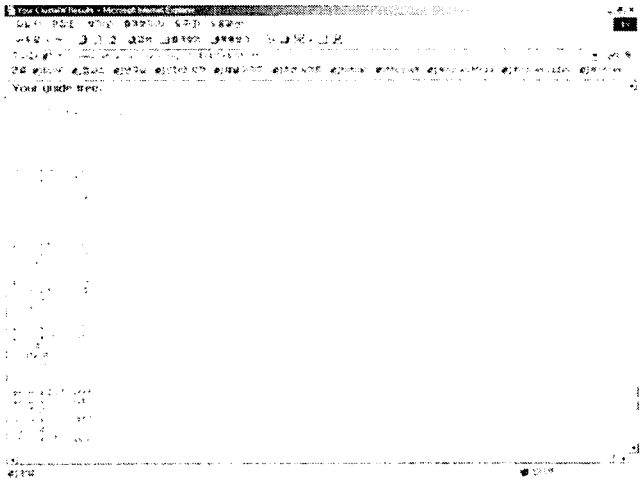
- Input Fasta file



Graphics Application Lab

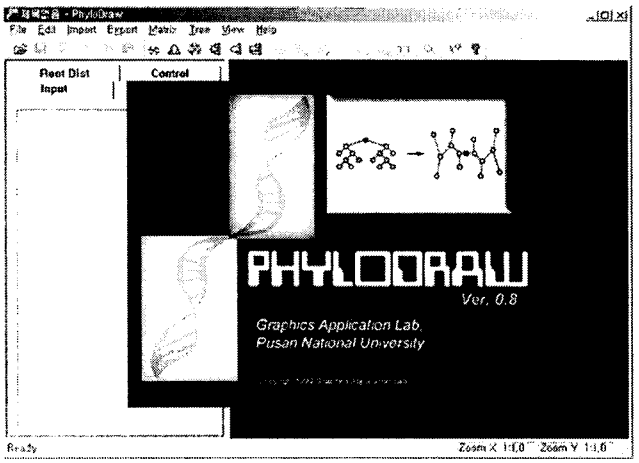
ClusterW(2) (www2.ebi.ac.uk/clusterw)

- Output



Graphics Application Lab

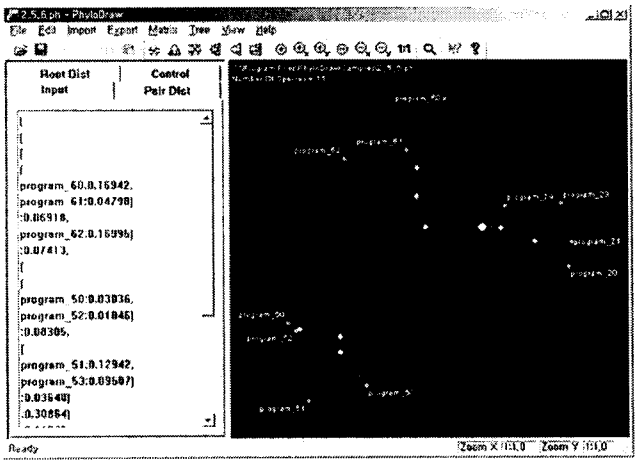
PhyloDraw (cho et al, Bioinformatics 2001.)



Graphics Application Lab

Experiment Result 1-0

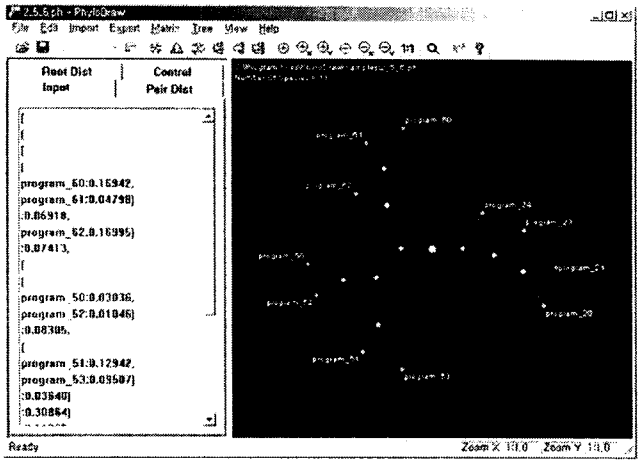
❑ 유사한 그룹을 이루는 **11**개의 프로그램



Graphics Application Lab

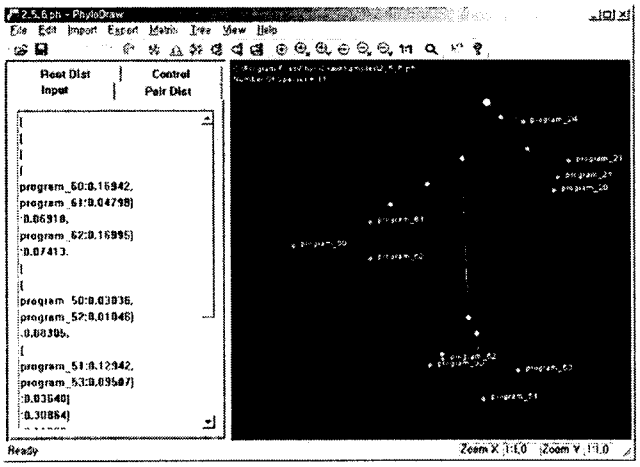
Experiment Result 1-1

- 유사한 그룹을 이루는 **11**개의 프로그램



Experiment Result 1-2

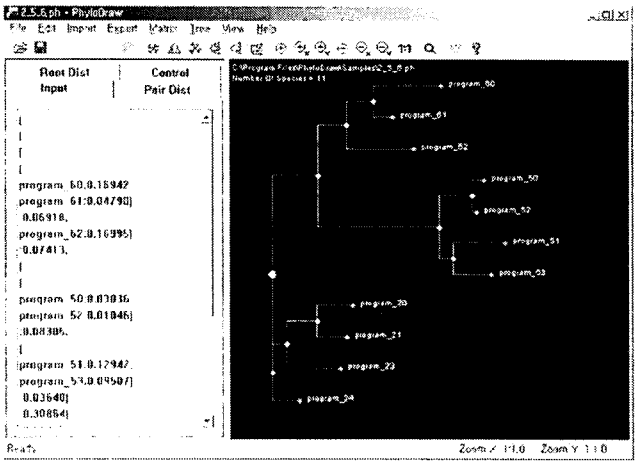
- 유사한 그룹을 이루는 **11**개의 프로그램



Graphics Application Lab

Experiment Result 1-3

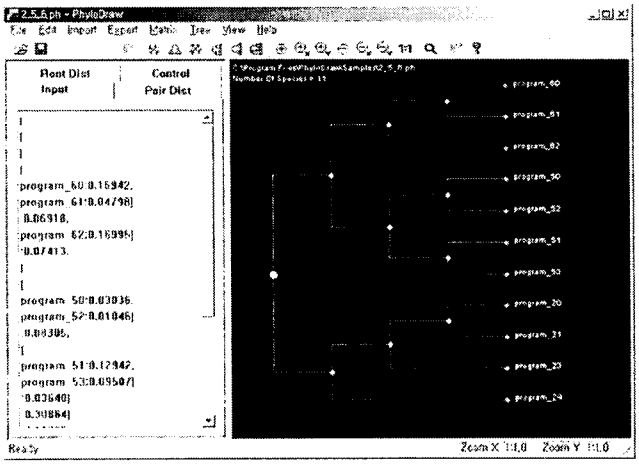
- 유사한 그룹을 이루는 **11**개의 프로그램



Graphics Application Lab

Experiment Result 1-4

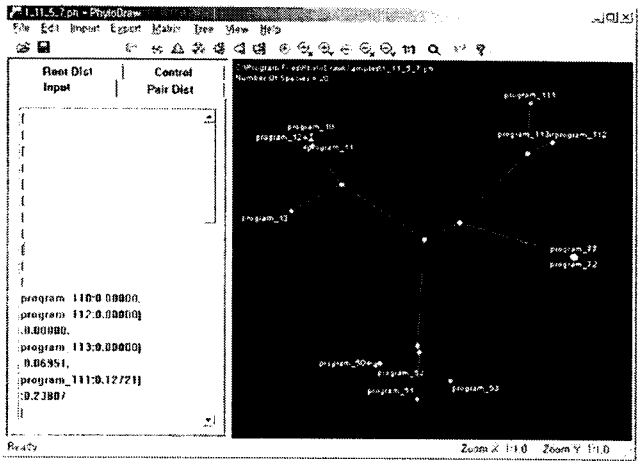
- 유사한 그룹을 이루는 **11**개의 프로그램



Graphics Application Lab

Experiment Result 2

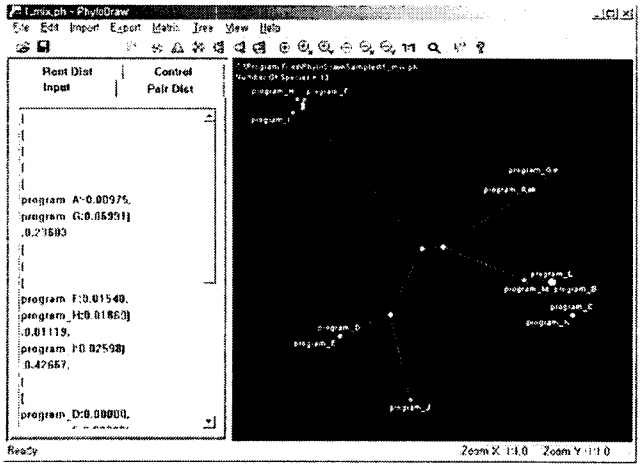
- 유사한 그룹을 이루는 **13**개의 프로그램



Graphics Application Lab

Experiment Result 3

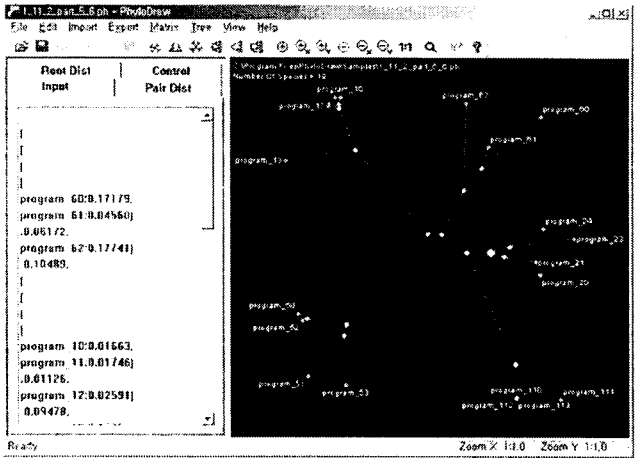
- 유사한 그룹을 이루는 **13**개의 프로그램



Graphics Application Lab

Experiment Result 4

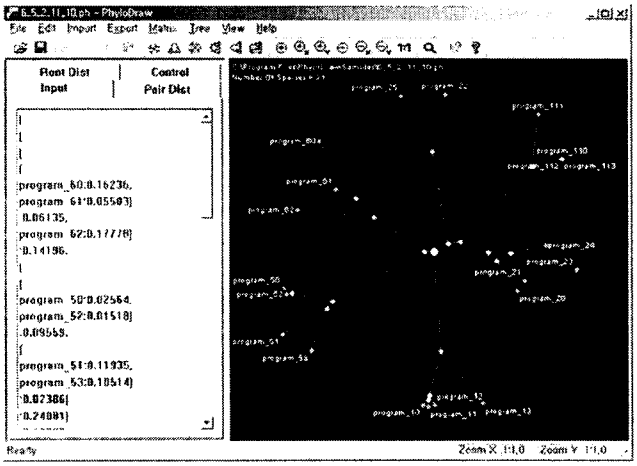
❑ 유사한 그룹을 이루는 **17**개의 프로그램



Graphics Application Lab

Experiment Result 5

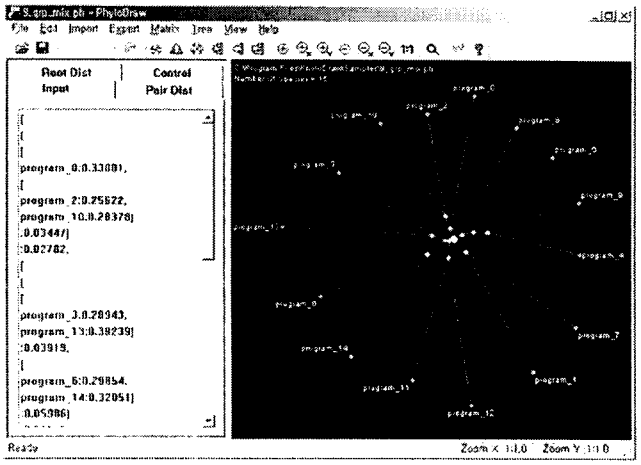
❑ 유사한 그룹을 이루는 **21**개의 프로그램



Graphics Application Lab

Experiment Result 6

- 유사도가 낮은 그룹을 이루는 **14**개의 프로그램



Graphics Application Lab

Conclusion

- **Alignment** 응용의 확장
- **Bioinformatics concept**의 활용
- **Linear Structure server**구축
- **Document fingerprint**의 활용

Further Work

- **Program DNA-Bank server**
- **Copying Phylogenetics Building**
- **Multi-alignment**
- **Parametric Method**
  - o Fixed-size fingerprinting = program protein
  - o PAM for program copying behavior
  - o Real Practice
- **Korean-Report Oracle**
  - o Has this report some originality ? Then how much ?
- **Music Plagiarism (**patent, smallsoft, BIOventure**)**
  - o Melody, bit, harmonics sequence alignment

Graphics Application Lab

Research Problems

- **How to linearize a procedure call ?**
  - o Procedure call is a sort of directed graph
- **Block-based linear alignment**
  - o Tree alignment
- **Genomic sequence with fixed-size fingerprint(**may help a fast screening **)**

Graphics Application Lab

Ga

Graphics Application Lab

People in BIOINFORMATICS