Towards Interactive Clustering on Parallel Environment

Zhen Liu

Faculty of Human Environment, Nagasaki Institute of Applied Science, 536 Aba-machi, Nagasaki 851-0193, Japan, Tel.: +81-95-838-4094, Fax: +81-95-839-4400, E-mail: liuzhen@cc.nias.ac.jp

Abstract *Clustering is one of the major data mining applications. An obvious characteristic of data mining distinguished from traditional data processing is that the conclusion of data mining cannot be predicted. Data mining is a multi-step process, and user must be allowed to be the front and the center in this process, especially clustering mining method. In this paper, the necessity of interactive data mining is illustrated. A framework of high performance interactive data mining on PC-cluster is proposed, and an interactive clustering algorithm for multi-dimensional data on the framework is presented.*

Keywords interactive data mining, multi-dimension data mining, clustering algorithm

1 Introduction

Clustering is one of the major data mining applications as it can discover data distributions and patterns. Data mining extracts hidden predictive facts or knowledge from large-scale databases [1][2]. Clustering problem is formally defined as follows: give a set of samples in multidimensional space, find a partition of the samples into clusters so that the samples within each cluster are similar to each other. And the samples in different clusters are not similar very obviously. Various distance functions can be used in order to make a quantitative determination of similarity. In addition, an objective function can be defined with respect to this distance function in order to measure the overall quality of a partition. The clustering problem is applied for similarity search, customer segmentation, pattern recognition, trend analysis, and classification [3][4][5][6] [7][8].

Many cluster methods have been developed in several different fields, with different definition of clusters and similarity among objects. The variety of clustering technologies is reflected by the variety of terms used for clusters analysis such as clumping, competitive learning, unsupervised pattern recognition, vector

quantization, partitioning, and winner-talk-all learning. Most of the early cluster analysis algorithms come from the area of statistics and have been originally designed for relatively small data sets. In the recent years, clustering algorithms have been extended to efficiently work for knowledge discovery in large-scale databases and, therefore, to classify large data sets with high dimensional feature items. Clustering algorithms are very computing demanding and, thus, require high-performance machines to get results in a reasonable amount of item.

In the meantime, the huge size of real-world databases systems brings about the following problems in data using: (1) Data quantitative problem, (2) Data qualitative problem, and (3) Data presentation problem. The data quantitative problem causes the decline of the processing speed having to do with a system that the accumulated amount of data becomes enormous too much. Also, there is a limit in the judgment and the ability to process. The data qualitative problem occurs because the complicated relation exists between the attributes or the data in the large-scale databases. The near combinations exist infinitely as the relations of data, attributes of data and the combinations of them are very complicated. Also, when the pattern among the detected data is too complicated, the thing that one finds some meaning from there becomes difficult. This is the data presentation problem. An effective way to enhance the power and flexibility of data mining in large-scale databases is to integrate data mining with on-line analytical processing (OLAP), visualization and interactive interface in a high performance parallel and distributed environment.

In this paper, the necessity of interactive data mining is illustrated. A framework of high performance interactive data mining on PC-cluster is proposed, and an interactive clustering algorithm for multi-dimensional data on the framework is presented. An application of the algorithm is introduced.

2 A Framework for High Performance Interactive Data Mining

2.1 Some key problems

In order to develop an interactive data mining support system in parallel and distributed computing environment successfully, the following key problems must be considered firstly: (1) On-line data mining; (2) Data parallelism; (3) Visual data mining; and (4) Interactive interface.

Data mining and OLAP are all analytical tools, but obvious differences exist between each other. The analysis process of data mining is completed automatically. It is only needed to extract hidden patterns, and predict the future trends and behaviors without giving exact query by user. It is of benefit to finding unknown facts. While OLAP depends on user's queries and propositions to complete analysis process. It restricted the scope of queries and propositions, and affects the final results. On the other hand, to data, most OLAP systems have focused on providing access to multi-dimensional data, while data mining systems have deal with influence analysis of data along a single dimension. It is an effective way to enhance

the power and flexibility of data mining by integrating data mining with OLAP to offset their weaknesses [9].

Data parallelism refers to the execution of the same operation or instruction on multiple large data subsets at the same time. This is in contrast to control parallelism, which refers to the key idea in data parallelism is that the whole data set is partitioned into disjoint data subsets, each of them allocated to a disjoint processor, so that each processor can apply the same operation only to its local data. From the point of view of the application programmer, automatic parallelization is an important advantage of data parallelism. In the control-parallelism paradigm the application programmer is in charge of all inter-processor communication and synchronization, which makes programming a time-consuming, error-prone activity. Data parallelism should be possible to add a number of processor nodes (CPU+RAM) to the system proportional to the amount of data increase, to keep the query-response time nearly constant, although there will be some increase in query-response time due to the increase in inter-processor communication time caused by adding more processors to exploit data parallelism.

Visual data mining is different from scientific visualization and it has the following characteristic: (1) wide range of users, (2) wide choice range of the visualization techniques, and (3) important dialog function. The users of scientific visualization are scientists and engineers who can endure the difficulty in using the system for little at most. However, a visual data mining system must have the possibility that the general person uses widely and so on easily. It is almost that the simulation results are represented in 2D or 3D visualization. However, it is more ordinary that the objects are not actual one in the information visualization. Moreover, it is possible to make a completely different expression form, too. The purpose of the information visualization becomes a point with important dialogs such as repeating data more in another visualization by changing the way of seeing data and the technique of the visualization and squeezing it because it is not visualization itself and to be in the discovery of the information retrieval and the rule is many.

2.2 The Overall Architecture and Mechanism

The architecture of the interactive high performance data mining support system is suggested as shown in Fig. 1. It mainly consists of: (1) Data Source: the platform of the on-line analytical data mining including Databases and Data warehouses; (2) Parallel database server: a horizontal partitioning; (3) Data Mining Agent: performing analytical mining in data cubes aided by OLAP engine; (4) OLAP Engine: providing fast access to summarized data along multiple dimensions; (5) Visualization platform: transforming multidimensional data into understandable information and providing parallel data mining visualization; (6) Applications Programming Interface: aggregation of instructions, functions, regulations and rules for on-line data mining, supporting interactive data mining.

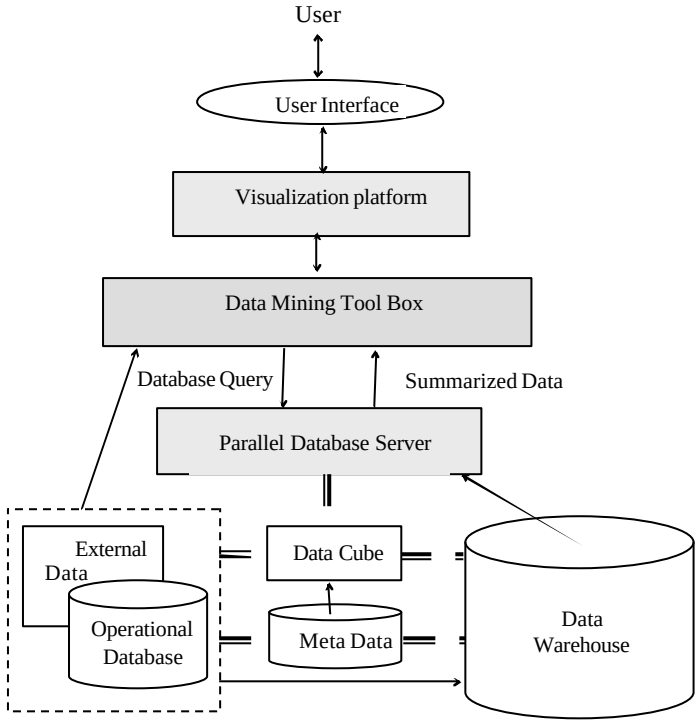

Fig. 1. Overall architecture of the system

The system has both of on-line data mining and parallel data mining features. Mainly components of the system is parallel database sever. Data cube is a core of on-line analytical data mining. It provides aggregated information that can be used to analyze the contents of databases and data warehouses. It is constructed from a subset of attributes in the databases and data warehouses. Data mining agent performs analytical mining in data cubes with the aid of OLAP engine. Data mining agent and the OLAP engine both accept user's on-line queries through the user interface and work with the data cube through the applications programming interface in the analysis. Furthermore, data mining agent may perform multiple data mining tasks, such as concept description, association, classification, prediction, clustering, time-series analysis, etc. Therefore, data mining agent is more sophisticated than the OLAP engine since it usually consists of multiple mining modules which may interact with each other for effective mining.

Basing on parallel visualization subsystem, an interactive Application Programming Interface (API) is provided. The basic function of the API is that of a PGUI (Parallel Graphic User Interface). It includes direct operation, dynamic search, continuous operation, and reversible operation, and so on. The interactive dialog is realized with a GH-SOM (Growing Hierarchical Self-Organization Map) model proposed by Dittenbach [10].

3 The Algorithm of Interactive Data Mining

3.1 The Algorithm of Interactive Clustering

The interactive clustering algorithm Interactive-Clustering is illustrated as follows. It runs Data Uploading and Data Partitioning to upload and partition data into disjoint data subsets, each of them allocated to a disjoint processor. And then, it calls the program named DFS to select a distance function.

```
Algorithm Interactive-Clustering
    Data uploading;
    Data distribution,
    Call DFS;
    sat = 1;
    Do {
        if format requirement is not satisfied then
            Call Data Processing Module;
        Call P-Clustering;
        Call Result-Evaluation;
        If the result is not satisfied then {
            Call Parameter Tuning Module;
            sat = 0;
        }
    } while sat = 0;
end
```

Data Processing Module provides data processing module to help the user. It can extract domain knowledge automatically from the input data and ask the user to refine the domain knowledge if necessary. Through asking the user a serial of common questions (such as which attribute is nominal or continuous), it can convert the original data file (if the field delimiter is other than the common) and construct input files that meet a specific algorithm's input formal automatically.

Parameter Tuning Module provides a unique set of common parameter for clustering algorithm. Consequently, different parameter tuning options are presented based on the algorithm. Brief explanations for the parameters are also provided. The default parameter values are provided initially. For detail explanations of the parameters, the user may refer to the online manuals.

3.2 The Algorithm of P_Clustering

Define that $S = s_{j,l}, 0 \le j \le N, 0 \le l \le M$ is a sample array in size of $M \times N$. Where, N is the number of samples, and M is the number of criteria of the sample. And, define that $\wedge = \{\wedge_0, \wedge_1, \cdots, \wedge_{k-1}\}$ is a subset drawn from N samples. Where, $\wedge_i$ is

called a cluster, and a sample must belong one luster exactly. The central value of a luster $\wedge_i, 0 \le i \le k$, $m_{i,l}$ is calculated as follows:

$$m_{i,l} = \frac{1}{|\wedge_i|} \sum s_{j,l},\ 0 \le i < k, 0 \le j < N, 0 \le l < M$$

where $|\wedge_i|$ means the sample number of cluster $\wedge_i$.

There are many measurement methods of similar degree of samples and clusters [11][12][13] [14][15]. The similar degree of sample j and sample i is calculated as follows:

$$d_{j,i}^2 = \sum_{j \in \wedge_i}^{M-1} (s_{j,i} - m_{i,l})^2, \quad 0 \le j < N, 0 \le i < k.$$

In this case, the square of deviation of ith cluster is calculated usually:

$$e_{j,i}^2 = \sum d_{j,i}^2,\ 0 \le j < N, 0 \le i < k.$$

The algorithm of P_Clustering is shown as follows. First, sample array $s_{j,l}$ is assigned with the local variable of processor P_j.

AlgorithmP-Clustering

Input: $k, s(j)[l],\ 0 \le j < N, 0 \le l < M$;
Output: $id(j),\ 0 \le j < N$;
// *Initial cluster assignment.*
Processor $P_j,\ 0 \le j < N$;
if $0 \le j < k$ **then**
$cl(j) = j$;
else
$cl(j) = nil$;
end if
for l=0, M-1
Processor $P_j,\ 0 \le j < k$, copy $s(i)[l]$ to 俞$(i)[l]$;
end for
a: // *Label each sample to a specified cluster.*
// *Compute* $d_{j,i}^2 = \sum_{l=0}^{M-1} (s_{j,l} - 俞_{i,l})^2,\ 0 \le j < N, 0 \le i \cdots < k$.
for i = 0, k-1
Processor $P_j,\ 0 \le j < N,$ set $d2(j)[l] = 0$;
for l = 0, M-1
Processor P_i broadcast 俞$(i)[l]$ to the local variable temp(*j*) of processor $P_j,\ 0 \le j < N$; then compute the distance between the *j*th sample and the *i*th cluster center according to the *l*th criterion by performing $b2(j) = (s(j)[l] - temp(j))^2$;
Processor $P_j,\ 0 \le j < N,$ accumulate the distance of each criterion by Performing $d2 = (j)[i] = d2(j)[i] + b2(j)$;
end for

end for

// *Initially, set* $\max(\mathrm{j}) = \infty$ *for* $0 \le j < N$;

for i = 0, k-1

Processor P_j, $0 \le j < N$,

if $\max(j) > d2(j)[i]$ **then**

$ncl(j) = i$ and $\max(j) = d2(j)[i]$;

end if

end for

// *Convergence check*.

Processor P_j, $0 \le j < N$,

if $ncl(j) = cl(j)$ **then**

$flag(j) = 1$

else

$flag(j) = 0$;

end if

Processor P_j, $0 \le j < N$, determine the local AND value on $flag(j)$ and store the result to the local variable $check(0)$ of processor P_0;

if $check(0) = 0$ then

// *Cluster center updating*

Processor P_j, $0 \le j < N$, copy $ncl(j)$ to $cl(j)$;

for i = 0, k-1

Processor P, $0 \le j < N$,

if $ncl(j) = 1$ **then**

$b(j) = 1$

else

$b(j) = 0$;

end if

Processor P_j, $0 \le j < N$, compute $ps(j) = \sum_{h=0}^{j} b(h)$ and then processor P_{N-1} copy $ps(N-1)$ back to the local variable $\wedge(i)$ of processor P_i;

for l = 0, M-1

Processor P_j, $0 \le j < N$

if $ncl(j) = 1$ **then**

$b(j) = s(j)[l]$

else

$b(j) = 0$;

end if

Processor P_j, $0 \le j < N$, compute $ps(j) = \sum_{h=0}^{j} b(h)$, and then processor P_{N-1} copy $ps(N-1)$ back to the local variable $w\wedge(i)[l]$ of processor P_i;

Processor P_j, perform $\bar{w}(i)[l] = w\wedge(i)[l] / \wedge(i)$ to update the lth criterion coordinate of the ith newly generated cluster center $\bar{w}(i)[l]$;

end for

goto a;

end for

else

Processor P_j, $0 \le j < N$, copy $cl(j)$ to $id(j)$;

end

4. An Application and Its Result Analysis

In this section, we show an application of the above algorithm. A questionnaire investigation of acceptance of user's environment-friendly car is accomplished at Japan (Tohoku University and the Industrial Technology Museum at Nagoya) and Korea (Chonbuk National University) with the AHP method [16][17][18]. The answer data of Tohoku University is applied to the above algorithm. The questionnaire is about how customers concern to the environmental problem (*EV*) compared with the price (*PP*), the performance (*PF*), the safety (*SF*), and the outward appearance (*OA*) of a car when buying a car [19][20]. So that, there are five criteria in this case, *EV*, *PP*, *PF*, *SE*, *AO*.

The interactive parallel clustering algorithm illustrated above is applied. The clustering results are presented with a 2*D* table and a 3*D* vertical line graph respectively (shown in Fig. 2, Fig. 3, Fig. 4, and Fig. 5). The 3*D* vertical line graph can show a clustering result of each stage to the user intuitively by the understanding form. In addition to the one's which is each graph indicating the weight of each attribute respectively during 2*D* table, the number of sample of each cluster is also indicated.

At the first of clustering, the P~Clustering is executed and the result to *k* clusters is shown to user with 2*D* table and 3*D* vertical line graph. The Result-Evaluation and user will judge the result by some algorithm and experience. For example, running the algorithms, the first result is shown as Fig. 2. When it is analyzed, the priority of the 5 criteria is as follows:

SF(0.3959)? PP(0.2182)? EV(0.1552)? PF(0.1408)? OA (0.0899).

The criterion considered most might be safety. But because the weight of the outward appearance was very small, it is found out that it would be over looked.

Tohoku University (to 5 clusters)

item	PP	EV	PF	SF	OA	person
cluster 1	0.2182	0.1552	0.1408	0.3959	0.0899	85
cluster 2	0.0867	0.1234	0.3355	0.1447	0.3098	30
cluster 3	0.3873	0.0664	0.2350	0.1269	0.1844	24
cluster 4	0.1098	0.4408	0.1179	0.1440	0.1875	5
cluster 5	0.9771	0.1315	0.0903	0.3730	0.3281	5

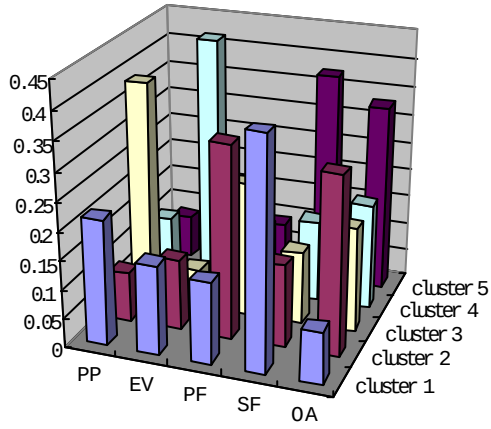

Fig. 2. Tohoku University to 5 Clusters

As the number of luster 4 and cluster 5 is very small (5 persons), there is no presentativeness. The user direct to run again with $k = 3$, and $k = 4$ (shown in Fig. 3 and Fig. 4).

Tohoku University（to 4 clusters）

item	PP	EV	PF	SF	OA	person
cluster 1	0.1477	0.1434	0.1156	0.3844	0.2090	90
cluster 2	0.0867	0.1234	0.3355	0.1447	0.3098	30
cluster 3	0.3873	0.0664	0.2350	0.1269	0.1844	24
cluster 4	0.1098	0.4408	0.1179	0.1441	0.1875	5

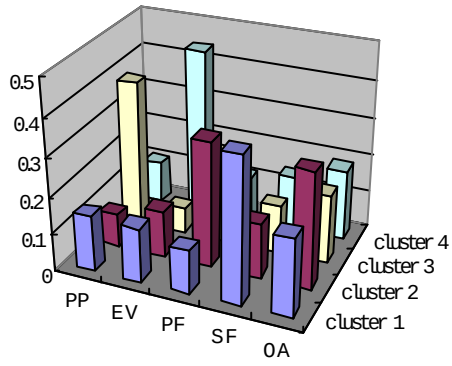

Fig. 3. Tohoku University to 4 Clusters

Tohoku University (to 3 clusters)

item	PP	EV	PF	SF	OA	person
cluster 1	0.1477	0.1434	0.1156	0.3844	0.2090	90
cluster 2	0.2370	0.0949	0.2852	0.1358	0.2471	54
cluster 3	0.1098	0.4408	0.1179	0.1440	0.1875	5

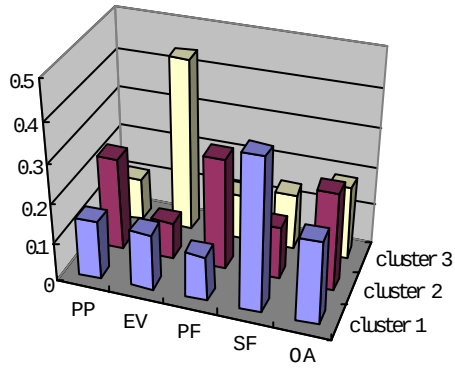

Fig. 4. Tohoku University to 3 Clusters

Again, the small number clusters exist (5 persons at Fig. 3 and at Fig. 4). The user lets system run again with $k = 2$ and the result shown as Fig. 5 is obtained.

Tohoku University (to 2 clusters)

item	PP	EV	PF	SF	OA	person
cluster 1	0.1923	0.1191	0.2004	0.2601	0.2280	144
cluster 2	0.1098	0.4408	0.1179	0.1440	0.1875	5

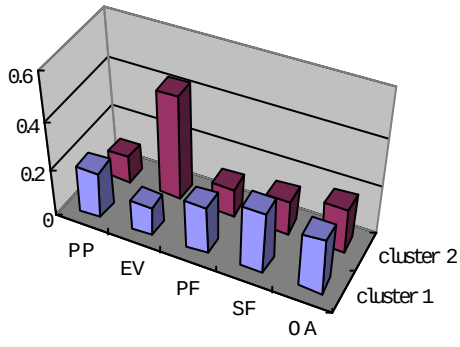

Fig. 5. Tohoku University to 2 Clusters

The one considered as the mainstream (144 persons) is as follows:

SF(0.2601)? OA(0.2280)? PF(0.2004)? PP(0.1923)? EV(0.1191).

Another cluster only include 5 people: the priority is:

EV(0.4408)? OA(0.1875)? SF(0.1440)? PF(0.1179)? PP(0.1098).

References

[1] Jiawei Han, and Michileine Kambr, Data Mining: Concepts and Teechnologies, Morgan Kaufmann Publisher, 2000.

[2] Frawley, W., Piatetsky-Shapiro, G., and Matheus, C, Knowledge Discovery in Databases: An Overview, Knowledge Discovery in Databases, eds. G. Piatetsky-Shapiro and W. Frawley, 1- 27, Cambridge, Mass.: AAAI Press / The MIT Press, 1991.

[3] S. Guha, N. Mishra, R. Motwani, and L. O'Callaghan, Clustering Data Streams, Proceedings of Symposium on Foundations of Computer Science (FOCS), 359-366, 2000.

[4] V. Ganti, R. Ramakrishnan, J. Gehrke, A. Powell, and J. French, Clustering Large Data Sets in Arbitrary Metric Spaces, Proceedings of International Conference on Data Engineering, 502-511, 1999.

[5] S. Guha, R. Rastogi, and K. Him, ROCK: A Robust Clustering Algorithm for Categorical Attributes, Information Systems, Vol. 25, No. 5, 345-336, 2000.

[6] S. Guha, R. Rastogi, and K. Him, CURE: An Efficient Clustering Algorithm for Large Databases, Proceedings of ACM SIGMOD Conference, 73-84, 1998.

[7] M. Ester, H-P. Kriegel, J. Sander, and X. Xu, A Density Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise, Proceedings of ACM knowledge Discovery and Databases Conference, 226-231, 1996.

[8] Cheng-Ru Lin and Ming-Syan Chen, Combining Partitional and Hierarchical Algorithms for Robust and Efficient Data Clustering with Cohesion Self-Merging, IEEE Transactions on Knowledge and Data Engineering, Vol. 17, No. 2, 145-159, 2005.

[9] J.M. Lambert, Perturbation Analysis of the Analytical Hierarchy Decision Method, Mathematical Modelling, Vol.19, No.5: 21-31, 1994.

[10] M. Dittenbach, D. Merkl, A. Rauber, The Growing Hierarchical Self-Organization Map, Proceeding of the international Joint Conference on Neural Networks, July, 2000.

[11] B. May, O. Kenneth, Intransitivity, Utility, and the Aggregation of Preference Patterns, Econometrica, Vol. 22, No. 1, 1954.

[12] M. Yang, and C. Ko, On Cluster-Wise Fussy Regression Analysis, IEEE Transactions on System, Man, and Cybernetics – Part B: Cybernetics, Vol.27, No. 1, February, 1997.

[13] M. Tavana, D. Kennedy, and P. Joglekar, A Group Decision Support Framework for Consensus Ranking of Technical Manager, Omega, International Journal of Management, Vol.24, No,5, 1996.

[14] R. Ramanathan, and L. S. Ganesh, Group Preference Aggregation Methods Employed in AHP: An Evaluation and an Intrinsic Process for Deriving Members' Weightages, European Journal of Operational Research, 97, 1994.

[15] Weiwu Fang, Disagreement, Degree of Multi-person Judgment in an Additive Structure, Mathematical Social Science, 28, 1994.

[16] T. L. Saaty, Axiomatic Foundation of the Analytic Hierarchy Process, management Science, Vol.32, No.7, 841-855, July 1986.

[17] T. L. Saaty, Analytic Hierarchy Process, McGraw-Hill, 1980

[18] Zhen Liu, Hikaru Inooka and Masana Kato, A Procedure for Measuring the Consistency of Pair-wise Comparison Matrix, Proceeding of The Fifth Conference of The Association of Asian-pacific Operational Research Societies, Singapore, 2000.

[19] LIU Zhen, A Study on Consumer Preference Analysis By Using Analytic Hierarchy Process, Computer Science Center, No.13, pp.43-51, 2002.

[20] LIU Zhen and INOOKA Hikaru, Investigation on Consumers' Consciousness of Car Environment Problem, The bulletin of Nagasaki Institute of Applied Science, Vol. 42, No, 1-2, 165-172, 2001.