ATLANTIS PRESS

Proceedings of the International Conference on
Industrial Engineering and Engineering Management

Ershi Qi · Jiang Shen · Runliang Dou
Editors

Proceedings of the 21st International Conference on Industrial Engineering and Engineering Management 2014

Proceedings of the International Conference on Industrial Engineering and Engineering Management

For further volumes:
http://www.atlantis-press.com

About this Series

Industrial engineering theories and applications are facing ongoing dramatic paradigm shifts. The proceedings of this series originate from the conference series "International Conference on Industrial Engineering and Engineering Management" reflecting this reality. The conferences aim at establishing a platform for experts, scholars and business people in the field of industrial engineering and engineering management allowing them to exchange their state-of-the-art research and by outlining new developments in fundamental, approaches, methodologies, software systems, and applications in this area and as well as to promote industrial engineering applications and developments of the future. The conferences are organized by CMES, which is the first and largest Chinese institution in the field of industrial engineering. CMES is also the sole national institution recognized by China Association of Science and Technology. Co-organiser of the conference series is the Tianjin University of Science and Technology.

Ershi Qi • Jiang Shen
Runliang Dou
Editors

Proceedings of the 21st International Conference on Industrial Engineering and Engineering Management 2014

ATLANTIS
PRESS

Editors
Ershi Qi
Tianjin University
Tianjin
People's Republic of China

Runliang Dou
Tianjin University
Tianjin
People's Republic of China

Jiang Shen
Industrial Engineering Institution of CM
Tianjin University
Tianjin
People's Republic of China

ISSN2157-3611 ISSN 2157-362X (electronic)
Proceedings of the International Conference on Industrial Engineering and Engineering Management
ISBN 978-94-6239-101-7 ISBN 978-94-6239-102-4 (eBook)
DOI 10.2991/978-94-6239-102-4

Library of Congress Control Number: 2014958285

Printed on acid-free paper

Preface

On behalf of the Chinese Industrial Engineering Institution, CMES, I am honored to welcome all the delegates of the 21st International Conference on Industrial Engineering and Engineering Management (IEEM 2014). It is your great efforts that brought out the proceedings of IEEM 2014, which records the new research findings and development in the domain of IEEM. What is more exciting, you are the experts or scholars with a significant achievement in the field. I believe that the proceedings will serve as a guidebook for potential development.

Being one of the most important international conferences in the field of industrial engineering and engineering management, it has been successfully held 20 times. Each conference was regarded as the greatest event in industrial engineering and engineering management. Today, we gather together in the beautiful city of Zhuhai to celebrate the 21st International Conference on Industrial Engineering and Engineering Management in the harvest season. The conference is sponsored by Chinese Industrial Engineering Institution, CMES and organized by Tianjin University of Science and Technology. The conference theme is "gathering the research results of the experts in the field, encouraging in-depth exchange of academic theories, exploring industrial innovation and development, promoting the world economic prosperity".

Now, industrial engineering theories and applications are facing ongoing dramatic paradigm shifts. The 21st IEEM responds to the reality by establishing an advantageous platform for experts, scholars, and business people in this area to exchange their state-of-the-art research in the field of industrial engineering and engineering management and by outlining new developments in fundamental, approaches, methodologies, software systems, and applications in this area to promote industrial engineering application and development in the future.

We extend our sincerest thanks to Atlantis Press for their generous support in the compilation of the proceedings. Next, we extend our sincerest thanks to Beijing Institute of Technology for holding such an excellent event. Finally, we thank all the delegates, keynote speakers, and the staff of the organization committee for their contribution to the success of the conference in various ways.

Chair of the conference Academic Committee — Ershi Qi
President of Chinese Industrial Engineering Institution, CMES

Organization

Honorary Chair

Yingluo Wang	Xi'an Jiaotong University, China
Zhongtuo Wang	Dalian University of Technology, China
Chongqing Guo	Tongji University, China
Shanlin Yang	Hefei University of Technology, China

Conference General Chairs

Dacheng Guo	Beijing Institute of Technology, China
Ershi Qi	Tianjin University, China

Program Chairs

Jiang Shen	Tianjin University, China
Yan Yan	Beijing Institute of Technology, China

Technical Committees

Anquan Zou	Changsha University, China
Bai-Sheng Chen	Takming University of Science and Technology, Taiwan
Bidyutkumar Bhattacharyya	Bengal Engineering & Science University, India
Binghui Liu	Xiamen University of Technology, China
Byoung-Kyu Choi	Korea KIIE Society, Korea
C.S. Lee	Taiwan Pingnan University, Taiwan
Changqing Li	Inner Mongolia University, China
Chen-Fu Chien	Taiwan Tsing Hua University, Taiwan
Cheng'en Wang	Chinese Academy of Sciences, China
Chen-guang Liu	Xi'an University of Technology, China

Chien-Hua Shen	Transworld Institute of Technology, Taiwan
Chih-Hsiung Hu	Taiwan Formosa University, Taiwan
Ching-Jong Liao	Taiwan University of Science and Technology, Taiwan
Chiu-tang Lin	Tungnan University, Taiwan
Chueh-Yung Tsao	Chang Gung University, Taiwan
Chuen-Sheng Cheng	Yuan Ze University, Taiwan
ChunChih Yeh	MingDao University, Taiwan
Chun-I Chen	I-Shou University, Taiwan
Chun-Mei Chou	Taiwan Yunlin University of Science and Technology, Taiwan
Chyuan Perng	Taiwan Quemoy University, Taiwan
Congdong Li	Guangzhou Jinan University, China
Dapeng Wei	Tianjin University of Science and Technology, China
Deng-maw Tsai	Taiwan Pingtung University of Science and Technology, Taiwan
Eric Min-yang Wang	Taiwan Tsing Hua University, Taiwan
ErryY.T. Adesta	International Islamic University, Malaysia
Ershi Qi	Tianjin University, China
Fajie Wei	Beihang University, China
Fansen Kong	Jilin University, China
Feifan Ye	Shaoxing University, China
Feng-Tsung Cheng	Taiwan Feng Chia University, Taiwan
Fen-Ru Shih	TaHwa University of Science and Technology, Taiwan
Fu Guo	Northeastern University, China
Fugee Tsung, Professor	Hong Kong University of Science & Technology, China
Fuh-Hwa Liu	Taiwan Chiao Tung University, Taiwan
Fuhou Zhao	Nankai University, China
Gang Liu	East China University of Science and Technology, China
Gongyu Chen	Sun Yat-sen University, China
Guang Cheng	Beijing Union University, China
Guofang Song	Shanghai University, China
Guoning Qi	Zhejiang University, China
Haicheng Yang	China Aerospace Science and Technology Corporation, Journal of Computer Integrated Manufacturing System—CIMS, China
Haoming Wu	University of Electronic Science and Technology, China
Hong-Da Lin	Chaoyang University of Technology, Taiwan
Hongming Zhou	Wenzhou University, China
Hsing-Pei Kao	Taiwan Central University, Taiwan

Tsemeng Lin	Taiwan Tsing Hua University, Taiwan
Wei Liu	Nanjing Agricultural University, China
Wei Sun	Dalian University of Technology, China
Wei Xue	Wenzhou University, China
Weihua Gan	East China Jiao tong University, China
Weixuan Xu	Institute of Policy and Management, Chinese Academy of Sciences, China
Wen-Chin Chen	Chung Hua University, Taiwan
Wenying Ding	Beijing University of Science and Technology, China
Xiaodan Wu	Hebei University of Technology, China
Xiaowei Zuo	Chinese Mechanical Engineering Society, China
Xin Chen	Guangdong University of Technology, China
Xingsan Qian	Shanghai Polytechnic University, China
Xingyuan Wang	Shandong University, China
Xinmin Zhang	Shenyang University of Technology, China
Yeh-Chun Juan	Ming Chi University of Technology, Taiwan
Yenming J. Chen	Taiwan Kaohsiung First University of Science and Technology, Taiwan
Yeu-Shiang Huang	Taiwan Cheng Kung University, Taiwan
Yi-chuan Wang	TOKO University, Taiwan
Yi-Feng Chen	HsiuPing University of Science and Technology, Taiwan
Yijun Li	Department of management science of NSFC, China
Yi-Kuei Lin	Taiwan University of Science and Technology, Taiwan
Yiming Wei	Beijing Institute of Technology, China
Ying Chung Chang	ChienHsin University of Science and Technology, Taiwan
Yingqiu Xu	Southeast University, China
Yi-shuo Hung	Hsing-Kuo University, Taiwan
Yisong Zheng	Nankai University, China
Yongjiang Shi	Cambridge University, UK
Yoshimoto Mitsuhiro	WASEDA University, Japan
Yu Yang	Chongqing University, China
Yuchun Wang	FAW Car Co. Ltd., China
Yuejin Zhou	Nanjing University, China
Yuh-Jen Cho	ChungHua University, Taiwan
Yuh-Wen Chen	Dayeh University, Taiwan
Yunfeng Wang	Hebei University of Technology, China
Yuqiang Shi	Southwest University of Science and Technology, China
Zhen He	Tianjin University, China

Zhenguo Chen	Foxconn Technology Group, China
Zhibin Jiang	Shanghai Jiaotong University, China
Zhongyuan Wu	Tianjin Polytechnic University, China
Zhuangli Zheng	Guangdong University of Technology, China
Zuhua Jiang	Shanghai Jiao tong University, China

Organizing Committee Chairs

Yonghe Jiang	Beijing Institute of Technology, China
Runliang Dou	Tianjin University, China

Contents

Part I

Industrial Engineering Theory

Application on Product Information Storage of Expanded Linear List Based on Two-dimension Array

Neng-hao CAI
College of Management and Economics, Tianjin University, Tianjin, China
(banpai345@126.com)

Abstract - **Array can be regarded as a specific kind of extension of linear list. Based on two-dimension array, we extend the general linear list and present a new data structure which can be used to store and summarize product information. By implementing data storage system utilizing the new data structure, better logic association among data can be reflected, and it also facilitate the information retrieval process. This work gives a new way of thinking in product information storage.**

Keyword - **Data structure, linear list, product information storage, two-dimension array**

I. BACKGROUND

It is essential that a production-oriented enterprise collect detailed information of its products and utilize the information to optimize business strategies, to promote production efficiency or to orientate its market accurately [1]. With the expansion of the scale of production and market, more information of product are generated. So, in the enterprise informationization process, it is necessary to summarize product information through a more efficient data structure in order to achieve the efficient use of information [2].

Assume that a company launches n series of products based on market positioning, namely, N1, N2, N3 etc. The target consumers includes high-end consumers, medium-end consumers, low-end consumers who are divided by their consuming ability. In each series, several models are produced according to factors such as price or configuration (for example, N1 series may have several models, such as N1-a, N1-b, N1-c). According to real-life experience, under normal circumstances, if a series targets at higher end consumers, the number of its models tends to be limited. Because in the same enterprise, one model targets at higher end consumers has a higher price, and it has a more limited market, thus it may have a limited sales volume. In order to control cost, it will be unworthy to launch more models.

In view of the above assumption, we intend to present a new data structure which can be used to store product information efficiently, when saving information to computers, the data structure should have several characteristics:

1 From a logical standpoint, different series of products should be sorted and stored according to the reference price of its target market, thus the market orientation of products can be reflected.

2 From a logical standpoint, to provide enterprise with information for reference, different models of products should be sorted and stored by price.

3 The comparison of prices between one particular model and the corresponding series which the model belongs should be facilitated.

4 Different models, different series of products can be compared in terms of price.

5 It should be easy to query and retrieval product information.

In fact, product series can be thought of as a base class, then the models of the series can be considered as its derivations. To simplify the complexity of the problem, we assume product series and product are of the same data type, the unique identity is product's name (or the name of product series), and only the price information are contained in the data type. The price of one series is the market reference price (or the average price in the industry) and the price of one product is its tag price.

II. PROBLEM ANALYSIS

The storage of product information can be implemented by varied data structures [3, 4].

For example, the implementation utilizing a tree structure is shown in Fig.1.

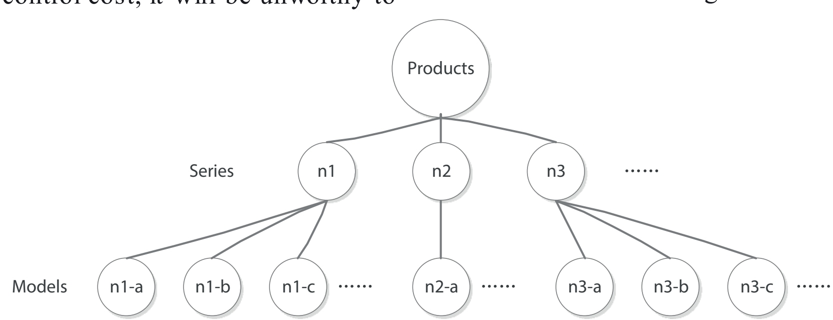

Fig.1. Utilizing tree structure to store information

By using the tree structure, the logic correlation among data can be well reflected as well as the relation between series and models [5]. But considering that one model's tag price is within a certain range above or

E. Qi et al. (eds.), *Proceedings of the 21st International Conference on Industrial Engineering and Engineering Management 2014*, Proceedings of the International Conference on Industrial Engineering and Engineering Management, DOI 10.2991/978-94-6239-102-4_1, © Atlantis Press and the authors 2015

below the market reference price of its corresponding series, the tree structure cannot intuitively reflect the correlation between market reference price and product's tag price. Meanwhile, if we want to retrieval one specific model's information, we need to access down by the root node, thus retrieving through the tree structure is not flexible.

If the storage system utilize normal linear list based on one-dimension array, the approach is as shown in Fig.2:

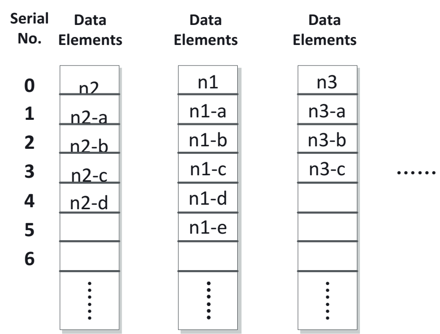

Fig.2. the linear list approach

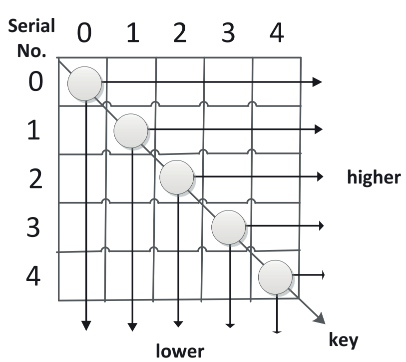

Fig.3. the diagonal-major order storage structure based on two-dimension array

The linear list approach [6, 7] has a limitation of showing the correlation among series, by using the linear list approach we also need to create different tables corresponding to different series, thus it is not a flexible approach. But because the data structure is implemented by array, we can retrieval information very easily [8].

For the above analysis, we hold the view that a data structure based on two-dimension array would have certain advantages. Compared with one-dimensional array, the extra dimension can be utilized to reflect the correlation among series without losing the flexibility of retrieval and traversal via array index.

In general, two-dimension array has two ways to store information, respectively, row-major order storage and column-major order storage [9]. But in order to further enhance the data associativity, also considering the development of computer hardware level and that company has ability to estimate its scale of production, we can sacrifice some storage space and modify the two-dimension array by implementing a diagonal-major order storage.

As shown in Fig.3, the diagonal elements a[i][i] (i=0,1,2…) make the major storage order, product series are sorted (ASC) by price and stored in diagonal elements starting from a[0][0]. For models in one series a[t][t], the information of those models whose price are higher than the series price are stored (ASC) in a[t+1][t], a[t+2][t], a[t+3][t]…; for those whose price are lower than the series price, their information are stored (DESC) in a[t][t+1], a[t][t+2], a[t][t+3]…

The diagonal elements sequence can be considered a normal one-dimensional list with all operations of one-dimensional list can be applied. While for a given series a[t][t], both the horizontal sequence and the vertical sequences can also be seen as normal one-dimensional linear lists, thus the proposed approach has obvious advantages in retrieval and traversal process.

The proposed data storage structure make an intuitive and clear reflection of correlation among different models or different series also the correlation between models and series they belong to. Additionally, all the data operations such as insertion, deletion, updating are conducted via index, hence it will be more convenient and flexible [10].

It should be noted that when utilizing the proposed data structure to store product information, prior estimates of product series information and product models information need to be made so appropriate storage space can be allocated to the two-dimension array. If product information exceeds the pre-allocated storage space, the storage space for an array should be dynamically increased, which to some extent causes undesired operations.

It should also be noted that for element a[i][i] (i=0,1,2 ... ...) in major order, ie. the diagonal elements, with the increase of i, the corresponding row and column has less space for storage. Thus the diagonal elements should be inserted in ASC order by price starting from a[0][0].

III. IMPLEMENTATION OF DATA STORAGE

TABLE I
DATA OF SERIES AND MODELS

(Unit: Price/Unit)

Product series	Model	Tag price
N1 Average price: 50	N1-a	55
	N1-b	52
	N1-c	46
	N1-d	48
N2 Average price: 30	N2-a	36
	N2-b	34
	N2-c	29
	N2-d	28
	N2-e	30
N3 Average price: 20	N3-a	22
	N3-b	27
	N3-c	22
	N3-d	21
	N3-e	19
N4 Average price: 100	N4-a	123
	N4-b	112

To illustrate how the data structure works, a set of product information is taken as an example. Assuming an enterprise's product information is as follows (Table I). The product series consists of N1, N2, N3 and N4. Among which N2 and N3 target at low-end consumers, N1 targets at medium-end consumers and N4 targets at high-end consumers. All models and their corresponding tag prices have been given in Table I. The long-term strategic plan of the enterprise is to launch a series N5 for the medium-end market, while phasing out the N2 series.

A. Initializing an empty data structure

This step is to estimate the produce information and allocate proper storage space, the initial empty data structure is shown as Fig.4

Serial No.	0	1	2	3	4	5	6	7
0								
1								
2								
3								
4								
5								
6								
7								

Fig.4.empty data structure

Serial No.	0	1	2	3	4	5	6	7
0	N3							
1		N2						
2			N1					
3				N4				
4								
5								
6								
7								

Fig.5. series data import

Serial No.	0	1	2	3	4	5	6	7
0	N3	N3-d	N3-a	N3-c	N3-b			
1	N3-e	N2	N2-e	N1-b	N1-a			
2		N2-c	N1	N1-b	N1-a			
3		N2-d	N1-d	N4	N4-b	N4-a		
4			N1-c					
5								
6								
7								

Fig.6. model data import

The series data should be inserted into the diagonal positions one by one, whenever a series information is inserted, comparison should be made in order to make sure that a[i][i] are sorted by price (see Fig.5). Then insert the information of models belong to each series, before insertion, comparison to the series price (average price in the industry) should be made. For those product whose tag price is higher than the series price, their information should be inserted into the corresponding row; for those whose tag price is lower than the series price, their information goes into the corresponding column. If the tag price is equal to the series price, we set a man-made rule that its information should be inserted into the corresponding row; then make comparison and sort the inserted data in the corresponding row or column (see Fig.6).

B. Data insertion

If we want to insert the information of N5 into the structure (assuming the average price of N5 series is 60), the insertion process is as follows:

(1) Compare N5's average price with prices of existing series, it can be seen that N5's price is higher than N1's price and lower than N4's price.

(2) Move N4's data as well as all data of models which belong to series N4 from the original position [p][q] to [p+1][q+1].

(3) Insert N5's data into the original position of N4 and import all N5 models' data sequentially.

C. Data deletion

If the enterprise plans to cease product N2-b of the N2 series, we can simply remove it from the data structure just like operating on a linear list. If it is the whole N2 series which is to be ceased, we just need to remove all N2 series data by deleting data of the corresponding row or column in a linear list operation way, then move all elements (including series' and models') which belong to series of higher price from the original position [p][q] to [p-1][q-1].

D. Data traversal and retrieval

To traverse data is simply to follow diagonal order and access data in the corresponding row or column via index. The retrieval process is also based on array indexing and there is no need to go into details here.

The implementation (C++ code) is as follows, comments are made to explain some detailed code and all the following code has been tested and validated on computer.

```
class Product              //class of series or product model
{
public:
    char* name;                    //series name or model name
    int price;                                         //price
    Product(char*, int);                         //constructor
};
class D_List                     // two-dimension array storage
{
public:
    Product** elem;                     // second rank pointer
    int num;                               //number of series
    int width[100];                  // horizontal storage space
    int height[100];                   // vertical storage space
    int listsize;              //initial side length of the table
};
//----implementation of constructor----
Product::Product(char* n, int p)
{
    name=n;
    price=p;
}
//----initialization of the two-dimension array----
int InitD_List(D_List &D, int n)
{
     //create pointer arrays
    D.elem = (Product**)malloc(n*sizeof(Product*));
    if(!D.elem)return 0;
    for(int i=0;i<n;i++){
        //for each row, allocate space for storage (n elements)
        D.elem[i] = (Product*)malloc(n*sizeof(Product));}
    D.listsize=n;                  //side length of the table is n
    D.num=0;                  //the initial number of series is 0
    for(int i=0;i<n;i++){
        //set the number of model to 0
        D.width[i]=0;
        D.height[i]=0;}
```

```
        return 1;
}
//----insertion of series data-–-
int InsertSeries(D_List &D, Product s)
{
        if(D.listsize<=D.num)return 0;
        for(int i=0;i<D.num;i++){
                //traverse each series
                if(s.price < D.elem[i][i].price){
                        MigrateSeries(D,i);
                        D.elem[i][i]=s;         //insert into position i
                        D.width[i]=D.height[i]=0;
                    return 1;}
        }
//if the price is higher than all existing series, insert into the far end
position
        D.elem[D.num][D.num]=s;
        D.width[D.num]=D.height[D.num]=0;
        D.num+=1;                    //update the number of series
        return 1;
}
//----model data insertion-–-
int InsertProduct(D_List &D, Product s, Product p)
{
        int n;                  //index to find the corresponding series
        for(int i=0;i<D.num;i++){
                if(s.name==D.elem[i][i].name)
                {n=i;break;}
              //find the series s belongs to
                if(i==D.num-1)return 0;}
        if(p.price >= D.elem[n][n].price) {
                //if price is no less than the series price, insert into the
corresponding row
                if(D.listsize<=D.width[n]-1)return 0;
                for(int i=0;i<=D.width[n];i++) {
                        if(p.price < D.elem[n][n+i].price){
                                MigrateProductW(D,n,i);
                                D.elem[n][n+i]=p;
                                return 1;}}
                D.width[n]+=1;      //update the number of models
                D.elem[n][n+D.width[n]]=p;
                // if the price is higher than all existing models, insert
                //into the far end position
                return 1;
        }
        else{
                //if the price is lower than the series price, insert
                //vertically
                if(D.listsize<=D.height[n]-1)return 0;
                for(int i=0;i<=D.height[n];i++){
                        //the following code is like the preceding code but
                        //in an opposite way
                        if(p.price > D.elem[n+i][n].price){
                                MigrateProductH(D,n,i);
                                D.elem[n+i][n]=p;
                                return 1;}}
                D.height[n]+=1;
                D.elem[n+D.height[n]][n]=p;
                return 1;}
}
//----move the series at position n horizontally to the right-–-
int MigrateProductW(D_List &D, int n, int m)
{
        for(int i=D.width[n]+1;i>m;i–){
                D.elem[n][n+i]=D.elem[n][n+i-1]; }
        D.width[n]+=1;                  //update the number of models
        return 1;
}
//---- move the series at position n vertically downwards ----
int MigrateProductH(D_List &D, int n, int m)
{
        for(int i=D.height[n]+1;i>m;i--){
                D.elem[n+i][n]=D.elem[n+i-1][n];}
        D.height[n]+=1;                 // update the number of models
        return 1;
}
//---- move all the series behind position n rearward ----
int MigrateSeries(D_List &D, int n)
{
        if(D.listsize<=D.num)return 0;
        for(int i=D.num;i>n;i--){
                D.width[i]=D.width[i-1];
                D.height[i]=D.height[i-1];
                //update the numbers of models in both orientation for
                //the following loop
                for(int j=0;j<=D.width[i];j++){
                        //horizontally copy data
                        D.elem[i][i+j]=D.elem[i-1][i-1+j];}
                for(int j=0;j<=D.height[i];j++){
                        //vertically copy data
                        D.elem[i+j][i]=D.elem[i-1+j][i-1];}}
        D.num+=1;                       //update the number of series
        return 1;
}
//----deletion of series at position n -–-
int DeleteSeries(D_List &D, Product s)
// the following code is like the preceding code but in an opposite way
{
        int n;                                          //index n
        for(int i=0;i<D.num;i++){
                if(s.name==D.elem[i][i].name)
                {n=i;break;}            //find the corresponding series
                if(i==D.num-1)return 0; //if there is no such series, exit
        }
        //the deletion will not be conducted if n is less than 0 or no less
        //than the far end index
        if(n<0||n>D.num-1)return 0;
        for(int i=n;i<D.num;i++){
                D.width[i]=D.width[i+1];
                D.height[i]=D.height[i+1];
                //update the numbers of models in both orientation for
                //the following loop
                for(int j=0;j<=D.width[i];j++){
                        //horizontally copy data
                        D.elem[i][i+j]=D.elem[i+1][i+1+j];}
                for(int j=0;j<=D.height[i];j++){
                        //vertically copy data
                        D.elem[i+j][i]=D.elem[i+1+j][i+1];}
        }
        D.num-=1;                       // update the number of series
        return 1;}
//note that this operation does not actually delete data of the series at
//the far end, but //the number of series (D.num) is updated and the
//data will not be accessible to users, //thus it is equivalent to deletion.
```

IV. CONCLUSION

This paper provides a new way of thinking to serve the storage of product information. The diagonal-major order storage method based one two-dimension array can better reflect the logical correlation. By utilizing array index, the traversal process of data is simplified. Although the proposed data structure is not so economical in storage space comparing to traditional tree structure or one-dimensional array structure, but given the current high levels of computer hardware, and that company often has ability to estimate its scale of production, the advantages of data structure are still visible.

REFERENCE

[1] Von Kohorn H. Product information storage, display, and coupon dispensing system: U.S. Patent 5, 249, 044[P].

1993-9-28.
[2] Strickler M T. Typically high-availability information storage product: U.S. Patent 6, 563, 706 [P]. 2003-5-13.
[3] YAN Wei-min, Wu Weimin. Data Structure(c-language version) [M]. Tsinghua University press, 1997.
[4] Kruse R L. Data Structures And Program Design In C+ [J]. 2001.
[5] Weiss M A. Data structures and algorithm analysis in C[J]. Pearson Education Asia, 2002.
[6] DA Linmei. The Implemention of Students' Score Query Based on Sequence List [J]. Computer Study, 2011 (2): 74-75.
[7] ZHANG Fu-xing, Sun Jia-xia. Importance of Generalized Lists in Data Structures [J]. Journal of Henan Institute of Science and Technology (Social Science Edition), 2006 (4): 103-104.
[8] Ford W, William F, Topp W. Data structures with C++ [M]. Simon & Schuster, 1995.
[9] Adam Drozdek. Data Structures and Algorithms in C++ (SECOND EDITION) [M]. Brooks/Cole, 2001.
[10]Reek K A. Pointers on C [M]. Addison-Wesley Longman Publishing Co., Inc., 1997.

Key Factors of K-nearest Neighbors Nonparametric Regression in Short-time Traffic Flow Forecasting

Jing-ting Zhong, Shuai Ling*
College of Management and Economics, Tianjin University, Tianjin, China
(ls5209@163.com)

Abstract - **Short-term traffic flow prediction plays an important role in route guidance and traffic management. K-NN is considered as one of the most important methods in short-term traffic forecasting, but some disadvantages limit the widespread application. In this paper, we use four tests to find the key factors of the K-NN method, which will give inspires to the further research to improve the method.**

Keywords - **K-nearest neighbors, key factors, short-term traffic flow forecasting**

I. INTRODUCTION

Shore-term traffic flow forecasting has played an important role in the intelligent transportation system (ITS). Traffic management departments can use it to design strategies of traffic control and traffic guidance, while travelers use it to make route choice. Under the above benefits, short-term traffic flow forecasting has become an attractive field both in traffic science and traffic engineering.

By definition, short-term traffic forecasting is the process to predict key traffic parameters such as speed, flow, occupancy, or travel time with a forecasting horizon typically ranging from 5 to 30 minutes at specific locations.

There are generally two kinds of approaches for short-term forecasting: parametric approach and non-parametric approach. The parametric approach assumes that there is an explicit forecasting mathematic by a set of parameters. Historical data is then used to find out a group of parameters which can minimum the forecasting error (gained by the historical data). Afterwards, the model can be used in the real-time forecasting.

Contrarily, instead of finding the explicit mathematic form the relationship between inputs and outputs, nonparametric approaches are data-driven and allow data to speak for itself [1] (Bosq 1996). The most popular nonparametric approaches includes nonparametric regression (NPR).

The NPR model is a data-driven approach. Instead of trying to compress all training data into a set of mathematical specifications (parametric approaches) or a certain network (ANN) through modeling process, it retains all historical observation and searches for the most similar case of the current state, based on which forecasting is then made. Oswald, Scherer et al. (2000) investigated the practical use of NPR model, and discussed some problem likely to encounter in the real world usage [2]. (Clark 2003) studies the multi-variant NPR forecasting, as well as the influence of neighbor size and the transferability of database, which are valuable for the practical use of NPR model [3]. (Chang, Zhang et al. 2011) made three improvements, including the data organization and the search mechanism, for faster calculation and higher accuracy [4]. Other researches considered additional information in NRP forecasting, such as historical traffic state and traffic condition information, and stated these helps to reduce forecasting error [5, 6] (Abdulhai, Porwal et al. 2002; Gong and Wang 2002).

With the continuous deepening researches on K-NN, they become increasingly mature in short-time traffic flow forecasting and will be applied in intelligent transportation system [7, 8] (Friedman, Bentley et al. 1977; Van Der Voort, Dougherty et al. 1996). For example, with the help of nonparametric forecasting models, a real-time, on-line, self-learning forecast system can be implemented [9] (Zhu and Yeh 1998).

In practical applications, Although K-NN has many advantages, the application of nonparametric regression methods still have to pay attention to that the K-NN method involve massive data and calculations.

II. K-NEAREST NEIGHBORS NONPARAMETRIC REGRESSION

In fact, nonparametric regression is based on pattern matching and data mining. Suppose the short-time traffic flow forecasting has to predict the traffic flow of a section at next time epoch, the corresponding influencing factors ($f_1 \sim f_n$) have to be explored firstly [10-12] (Bentley 1975; Bentley and Friedman 1979; Bentley 1990). These influencing factors may include traffic flows of the section and upstream section at previous time epoch, weather, road conditions, etc. In this paper, $f_1 \sim f_n$ are taken as the state components of the system, which compose the state vector of the system $(f_1, \cdots, f_n)$. The traffic flow at the forecasting time epoch (q) is called as decision attribute, which is determined by $(f_1, \cdots, f_n)$. In other words, the current state vector ($F = (f_1, \cdots, f_n)$) determines q at the forecasting time epoch. In this paper, F and q composed a pattern vector ($P = \{(f_1, f_2, \cdots, f_n) \mid q)$).

Search k nearest neighbor (KNN) of the forecasting state (F_{pre}) in the historical pattern according to equation (1) and then predict by using q of KNN.

E. Qi et al. (eds.), *Proceedings of the 21st International Conference on Industrial Engineering and Engineering Management 2014*, Proceedings of the International Conference on Industrial Engineering and Engineering Management, DOI 10.2991/978-94-6239-102-4_2, © Atlantis Press and the authors 2015