Department of Computer Science and Engineering

PES UNIVERSITY

UE19CS202: Data Structures and its Applications (4-0-0-4-4)

SPARSE MATRIX

Abstract

SPARE MATRIX – overview and its representations

Vandana M Ladwani

vandanamd@pes.edu

Contents

SPARSE MATRIX

Overview

A matrix is represented as 2 dimensional array where every element is accessed by row and column index. Real world data such as image, spectrogram and graph can be modelled using matrices.
A matrix for which most of values are zero is termed as the sparse matrix. If a sparse matrix is stored in a memory as a two dimensional matrix it wastes lot of space.
So alternate representations are preferred for sparse matrix
Alternate representations are:

- Triple notation
- Linked representation

Representations

1. Triple notation

In triple notation sparse matrix is represented as an array of tuple values. Each tuple consists of
<rowno columnno Value>

The first block in array block holds information regarding
<total no of rows, total no of columns ,value>

Declaration

```
typedef struct
{
        int col;
        int row;
        int value;
} term;
term a[10];
```

Various operations that can be performed on sparse matrix are
Create_SparseMatrix()
Transpose_of_SparseMatrix()
Add_SparseMatrices()

Multiple_SparseMatrices()

$$\begin{bmatrix} 2 & 0 & 0 & 0 \\ 4 & 0 & 0 & 3 \\ 0 & 0 & 0 & 0 \\ 8 & 0 & 0 & 1 \\ 0 & 0 & 6 & 0 \end{bmatrix}$$

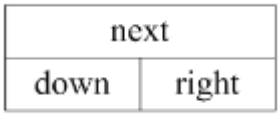
Triple Notation

Row No	Column No	Value
5	4	6
0	0	2
1	0	4
1	3	3
3	0	8
3	3	1
4	2	6

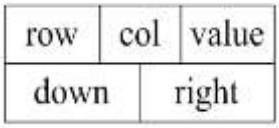
2. Linked Representation

Two types of nodes are used

Header Node

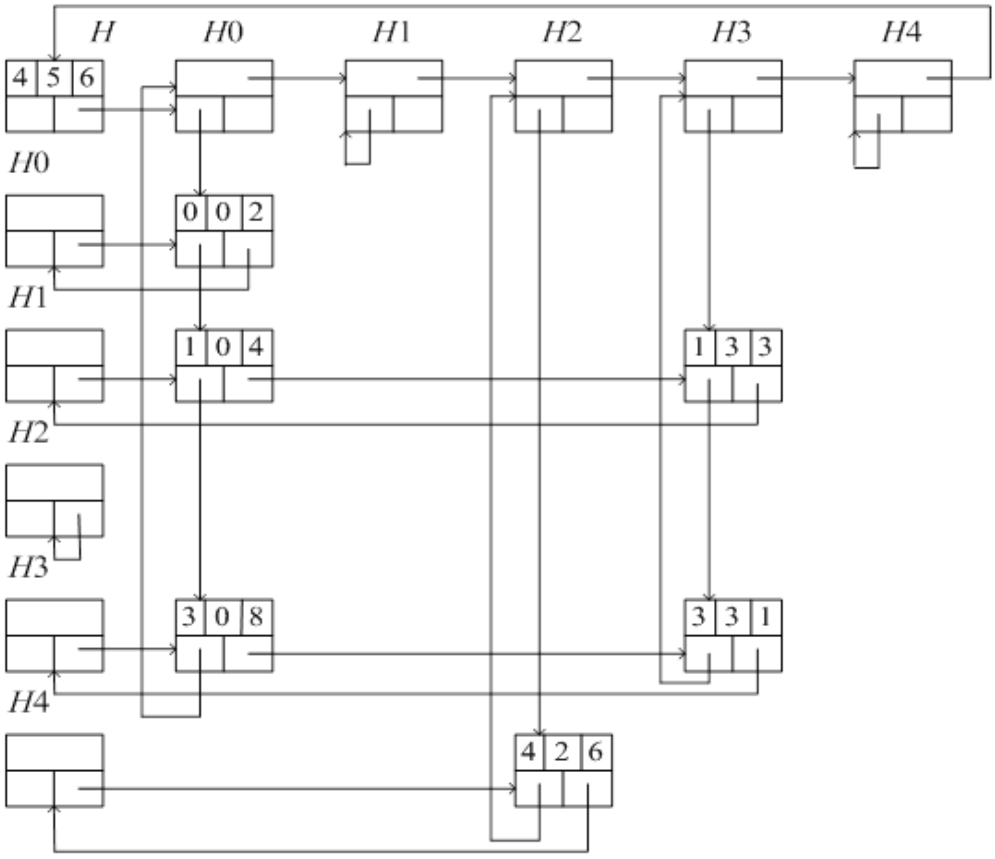


Data Node




```
#define MAX_SIZE 50 /* size of largest matrix */
typedef enum {head, entry} tagfield;
typedef struct matrixNode * matrixPointer;
typedef struct entryNode {
int row;
int col;
int value; };
typedef struct matrixNode {
        matrixPointer down;
        matrixPointer right;
        tagfield tag;
        union
        {
        matrixPointer next;
        entryNode entry;
        } u;
};
```

$$\begin{bmatrix} 2 & 0 & 0 & 0 \\ 4 & 0 & 0 & 3 \\ 0 & 0 & 0 & 0 \\ 8 & 0 & 0 & 1 \\ 0 & 0 & 6 & 0 \end{bmatrix}$$



Sparse Matrix representation using Linked Nodes

Courtesy: “Fundamentals of Data Structures” By Ellis Horowitz and Sartaj Sahni