Graph and Sparse Matrices

Dr. Animesh Chaturvedi
Assistant Professor at Data Science and Artificial Intelligence Department,
IIIT Dharwad
Post Doctorate: King's College London & The Alan Turing Institute
PhD: IIT Indore MTech: IIITDM Jabalpur

Graph Definition

Definition. A ***directed graph (digraph)*** $G = (V, E)$ is an ordered pair consisting of

- a set V of ***vertices*** (singular: ***vertex***),
- a set $E \subseteq V \times V$ of ***edges***.

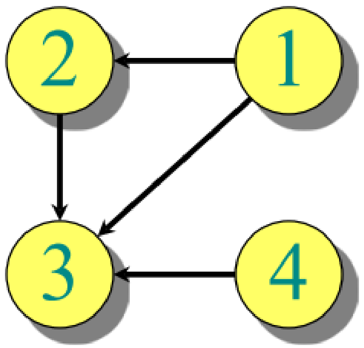
In an ***undirected graph*** $G = (V, E)$, the edge set E consists of *unordered* pairs of vertices.

by Erik D. Demaine and Charles E. Leiserson

Adjacency Matrix Representation of Graph

The ***adjacency matrix*** of a graph $G = (V, E)$, where $V = \{1, 2, \ldots, n\}$, is the matrix $A[1 \,..\, n, 1 \,..\, n]$ given by

$$A[i, j] = \begin{cases} 1 & \text{if } (i, j) \in E, \\ 0 & \text{if } (i, j) \notin E. \end{cases}$$



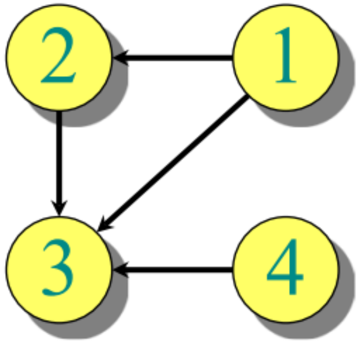
A	1	2	3	4
1	0	1	1	0
2	0	0	1	0
3	0	0	0	0
4	0	0	1	0

$\Theta(V^2)$ storage $\Rightarrow$ ***dense*** representation.

by Erik D. Demaine and Charles E. Leiserson

Adjacency List Representation of Graph

An ***adjacency list*** of a vertex $v \in V$ is the list $Adj[v]$ of vertices adjacent to v.



$Adj[1] = \{2, 3\}$
$Adj[2] = \{3\}$
$Adj[3] = \{\}$
$Adj[4] = \{3\}$

For undirected graphs, $|Adj[v]| = degree(v)$.
For digraphs, $|Adj[v]| = out\text{-}degree(v)$.

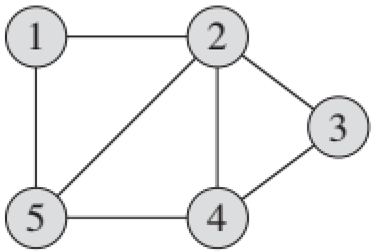
Handshaking Lemma: $\sum_{v \in V} degree(v) = 2|E|$ for undirected graphs $\Rightarrow$ adjacency lists use $\Theta(V + E)$ storage — a ***sparse*** representation.

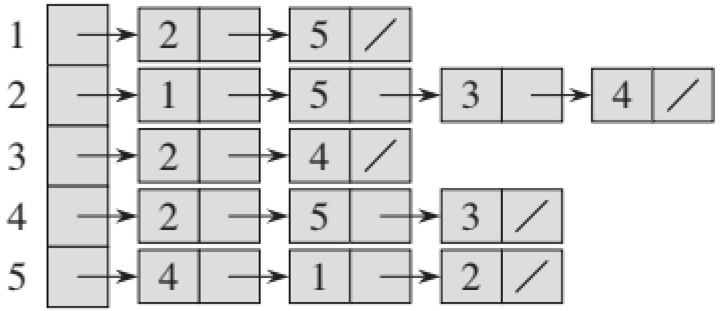
by Erik D. Demaine and Charles E. Leiserson

Graph representations

$$a_{ij} = \begin{cases} 1 & \text{if } (i, j) \in E , \\ 0 & \text{otherwise} . \end{cases}$$

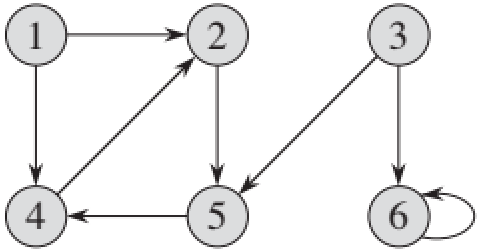
- Adjacency-list and Adjacency-matrix representation of undirected graph G with 5 vertices and 7 edges.

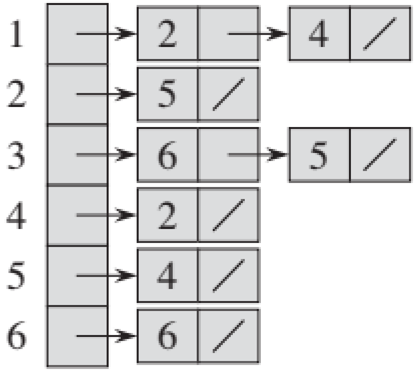




$$\begin{array}{c|ccccc|} & 1 & 2 & 3 & 4 & 5 \\ \hline 1 & 0 & 1 & 0 & 0 & 1 \\ 2 & 1 & 0 & 1 & 1 & 1 \\ 3 & 0 & 1 & 0 & 1 & 0 \\ 4 & 0 & 1 & 1 & 0 & 1 \\ 5 & 1 & 1 & 0 & 1 & 0 \\ \hline \end{array}$$

- Adjacency-list and Adjacency-matrix representations of a directed graph G with 6 vertices and 8 edges.





$$\begin{array}{c|cccccc|} & 1 & 2 & 3 & 4 & 5 & 6 \\ \hline 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 2 & 0 & 0 & 0 & 0 & 1 & 0 \\ 3 & 0 & 0 & 0 & 0 & 1 & 1 \\ 4 & 0 & 1 & 0 & 0 & 0 & 0 \\ 5 & 0 & 0 & 0 & 1 & 0 & 0 \\ 6 & 0 & 0 & 0 & 0 & 0 & 1 \\ \hline \end{array}$$

Sparse Matrices – Data Structure

1-D Array Representation

- Implementation of the abstract list data structure using programming language
  - “Backing” Data Structure
- Arrays are contiguous memory locations with fixed capacity
- Allow elements of same type to be present at specific positions in the array
- Index in a List can be mapped to a Position in the Array
  - Mapping function from list index to array position

Matrix Multiplication

```
// Given 2-D arrays: a[n][n], b[n][n]
// Output 2-D array: c[n][n] initialized to 0

for (i = 0; i < N; i++)
    for (j = 0; j < N; j++)
        for (k = 0; k < N; k++)
            c[i][j] += a[i][k] * b[k][j];
```

Symmetric Matrix

- An n x n matrix can be represented using 1-D array of size $n(n+1)/2$ by storing <u>either</u> the lower or upper triangle of the matrix
- Use one of the methods for a triangular matrix
- Optimization: The elements that are not explicitly stored may be computed from those that are stored.

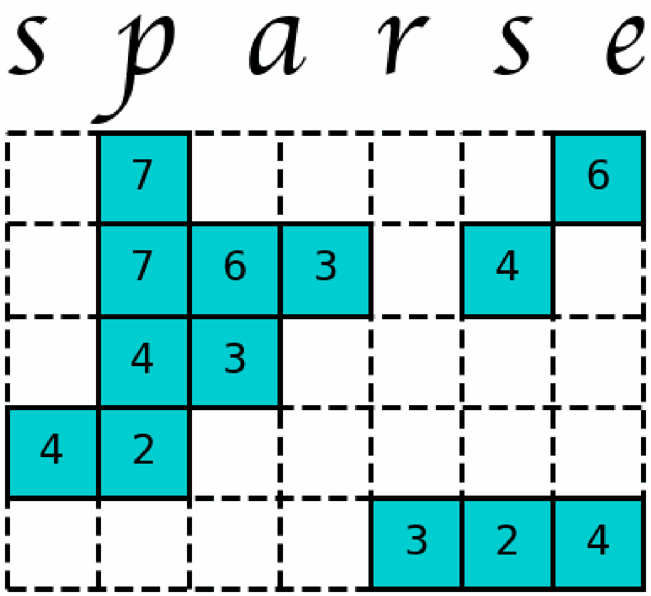
$$\begin{matrix} 2 & 4 & 6 & 0 \\ 4 & 1 & 9 & 5 \\ 6 & 9 & 4 & 7 \\ 0 & 5 & 7 & 0 \end{matrix}$$

Sparse Matrices

- Only a small subset of items are populated in matrix
  - Students and courses taken, faculty and courses taught
  - Adjacency matrix of social network graph
  - vertices are people, edges are “friends”
- Rows and columns are people, cell has 0/1 value
- Why not use regular 2-D matrix?
  - 1-D representation
  - Array of arrays representation

Sparse Matrix

- A matrix is sparse
  - if many of its elements are zero
- A matrix that is not sparse is dense
- The boundary is not precisely defined
  - Diagonal and tridiagonal matrices are sparse
  - We classify triangular matrices as dense
- Two possible representations
  - array
  - linked list



Dense Matrix

1	2	31	2	9	7	34	22	11	5
11	92	4	3	2	2	3	3	2	1
3	9	13	8	21	17	4	2	1	4
8	32	1	2	34	18	7	78	10	7
9	22	3	9	8	71	12	22	17	3
13	21	21	9	2	47	1	81	21	9
21	12	53	12	91	24	81	8	91	2
61	8	33	82	19	87	16	3	1	55
54	4	78	24	18	11	4	2	99	5
13	22	32	42	9	15	9	22	1	21

Sparse Matrix

1	.	3	.	9	.	3	.	.	.
11	.	4	.	.	.	.	.	2	1
.	.	1	.	.	.	4	.	1	.
8	.	.	.	3	1	.	.	.	.
.	.	.	9	.	.	1	.	17	.
13	21	.	9	2	47	1	81	21	9
.	.	.	.	.	.	.	.	.	.
.	.	.	.	19	8	16	.	.	55
54	4	.	.	.	11	.	.	.	.
.	.	2	.	.	.	.	22	.	21

Array Representation of Sparse Matrix

- The nonzero entries may be mapped into a 1D array in row-major order
- To reconstruct the matrix structure, need to record the row and column each nonzero comes from

$$\begin{bmatrix} 0 & 0 & 8 & 0 & 0 & 0 \\ 0 & 7 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 5 \\ 0 & 3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

Rows	Columns	values
5	6	5
0	2	8
1	1	7
2	5	5
3	1	3
4	4	1

$$\begin{matrix} 0 & 0 & 0 & 2 & 0 & 0 & 1 & 0 \\ 0 & 6 & 0 & 0 & 7 & 0 & 0 & 3 \\ 0 & 0 & 0 & 9 & 0 & 8 & 0 & 0 \\ 0 & 4 & 5 & 0 & 0 & 0 & 0 & 0 \end{matrix}$$

(a) A 4×8 matrix

a[]	0	1	2	3	4	5	6	7	8
row	1	1	2	2	2	3	3	4	4
col	4	7	2	5	8	4	6	2	3
value	2	1	6	7	3	9	8	4	5

(b) Its representation

Array Representation of Sparse Matrix

```
template<class T>
class Term {
private:
        int row, col;
        T value;
};
```

```
template<class T>
class sparseMatrix {
private:
        int rows, cols,
        int terms;
        Term<T> *a;
        int MaxTerms;
public:
        //…
};
```

$$\begin{bmatrix} 0 & 0 & 8 & 0 & 0 & 0 \\ 0 & 7 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 5 \\ 0 & 3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

Rows	Columns	values
5	6	5
0	2	8
1	1	7
2	5	5
3	1	3
4	4	1

Linked Representation of Sparse Matrix

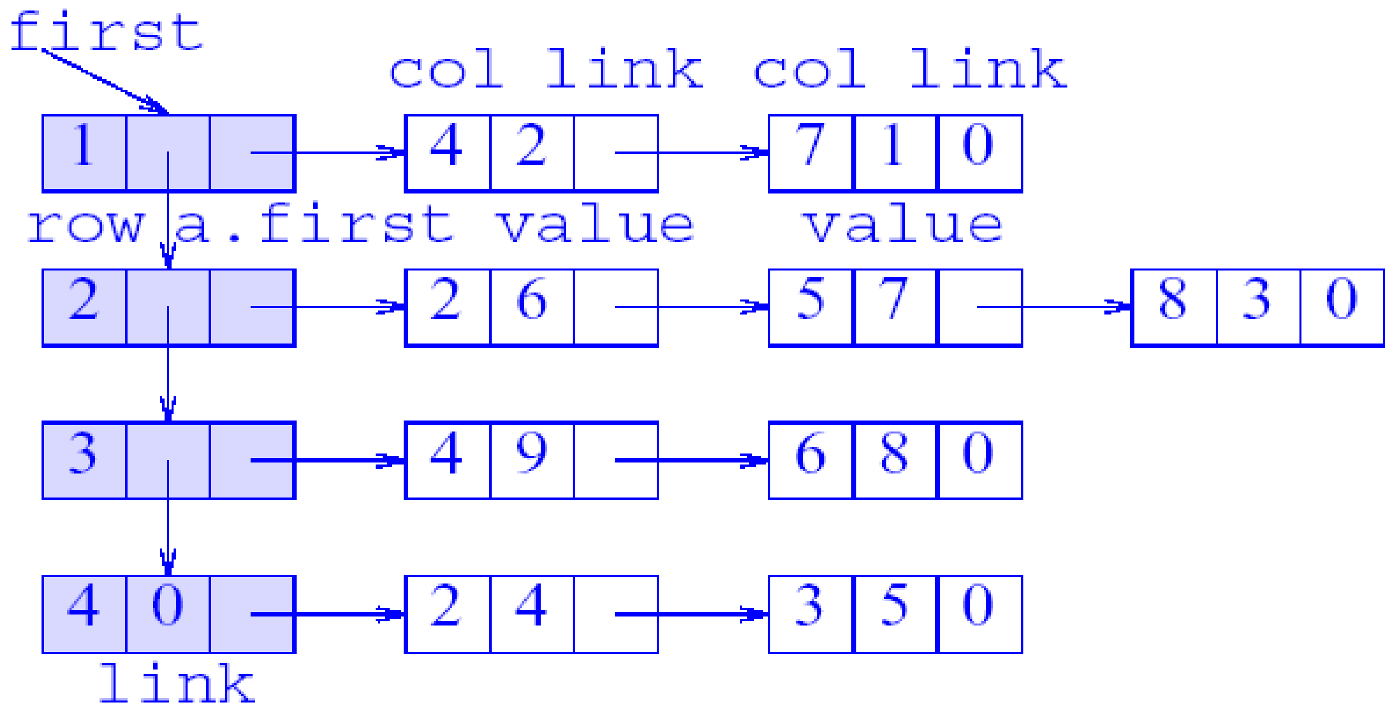
- A shortcoming of the 1-D array of a sparse matrix is that we need to know the number of nonzero terms in each of the sparse matrices when the array is created
- A linked representation can overcome this shortcoming

$$\begin{matrix} 0 & 0 & 0 & 2 & 0 & 0 & 1 & 0 \\ 0 & 6 & 0 & 0 & 7 & 0 & 0 & 3 \\ 0 & 0 & 0 & 9 & 0 & 8 & 0 & 0 \\ 0 & 4 & 5 & 0 & 0 & 0 & 0 & 0 \end{matrix}$$

(a) A 4×8 matrix

a[]	0	1	2	3	4	5	6	7	8
row	1	1	2	2	2	3	3	4	4
col	4	7	2	5	8	4	6	2	3
value	2	1	6	7	3	9	8	4	5

(b) Its representation



ขอบคุณ

Thai

Grazie

Italian

תודה רבה

Hebrew

ধন্যবাদ

Bangla

धन्यवादः

Sanskrit

Ευχαριστώ

Greek

Thank You

English

ಧನ್ಯವಾದಗಳು

Kannada

Спасибо

Russian

Gracias

Spanish

Obrigado

Portuguese

شكراً

Arabic

https://sites.google.com/site/animeshchaturvedi07

Merci

French

多謝

Traditional Chinese

धन्यवाद

Hindi

Danke

German

多谢

Simplified Chinese

நன்றி

Tamil

Tamil

ありがとうございました

Japanese

감사합니다

Korean