Introduction aux graphes

Représentation des graphes

Mathilde Vernet
mathilde.vernet@univ-avignon.fr

Licence Informatique, CERI
Licence Mathématiques, Institut AgES
Avignon Université

Automne 2025

Plan

Rappel

On sait qu'un graphe est à la fois :

- Un objet mathématique
- Une structure de données en informatique

Quelle est cette structure de données ?

- On connaît la représentation graphique de l'objet mathématique mais comment est-ce représenté en mémoire ?
  - Ça dépend
- Combien de place cette structure prend-elle en mémoire ?
  - Ça dépend
- Comment accède-t-on aux informations du graphe ?
  - Ça dépend
- Quel coût pour l'accès à ces informations ?
  - Ça dépend

Idées générales

La matrice d'adjacence

Représenter par une matrice les sommets qui sont voisins dans le graphe

Les listes d'adjacence

Représenter par une liste pour chaque sommet du graphe les sommets qui lui sont voisins dans le graphe

La matrice d'incidence

Représenter par une matrice le lien entre les sommets et les arêtes ou arcs

Approche orientée objet

Représenter par des objets des différents éléments du graphe

Dans un graphe non-orienté

La matrice d'adjacence

Un graphe non-orienté $G = (V, E)$ (en général sans arête multiple) peut être représenté par une matrice d'adjacence A de taille $n \times n$.

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix}$$

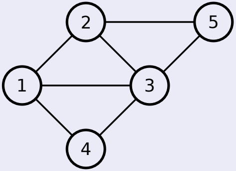
$$\text{où } a_{uv} = \begin{cases} 1 & \text{si } (u, v) \in E \text{ (si } u \text{ et } v \text{ sont voisins)} \\ 0 & \text{sinon} \end{cases}$$

Remarques

- La matrice d'adjacence est carrée
- La matrice d'adjacence est symétrique
  - Si (u, v) existe alors (v, u) existe

Dans un graphe non-orienté

Dessin de graphe et sa matrice d'adjacence



$$\begin{pmatrix} 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \end{pmatrix}$$

Dans cet exemple

- L'arête $(2, 5)$ existe donc $a_{25} = a_{52} = 1$
- Il n'y a pas d'arête entre les sommets 1 et 5 donc $a_{15} = a_{51} = 0$
- Il n'y a pas de boucle donc tous les a_{uu} sont nuls

Dans un graphe orienté

La matrice d'adjacence

Un graphe orienté $G = (V, A)$ (en général sans arc multiple) peut être représenté par une matrice d'adjacence A de taille $n \times n$.

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix}$$

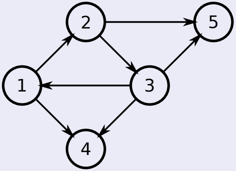
$$\text{où } a_{uv} = \begin{cases} 1 & \text{si l'arc } (u, v) \in A \\ 0 & \text{sinon} \end{cases}$$

Remarques

- La matrice d'adjacence est carrée
- La matrice d'adjacence n'est pas symétrique
  - L'existence de l'arc (u, v) n'implique pas l'existence de l'arc (v, u)

Dans un graphe orienté

Dessin de graphe et sa matrice d'adjacence



$$\begin{pmatrix} 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Dans cet exemple

- L'arc $(2, 3)$ existe donc $a_{23} = 1$ et $a_{32} = 0$
- Il n'y a pas d'arête entre les sommets 1 et 5 donc $a_{15} = a_{51} = 0$
- Il n'y a pas de boucle donc tous les a_{uu} sont nuls

Place et coût de l'accès aux informations

Place en mémoire

- D'un graphe quelconque : de l'ordre de n^2
- Identique qu'il soit très dense (beaucoup d'arêtes) ou très peu dense (peu d'arêtes)

Coût d'accès aux informations

- Savoir si u et v sont voisins : constant
- Savoir si u est isolé : de l'ordre de n
- Trouver tous les voisins de u : de l'ordre de n
- Compter le nombre d'arêtes/arcs du graphe : n^2

Dans un graphe non-orienté

Les listes d'adjacence

Un graphe non-orienté $G = (V, E)$ peut être représenté par une liste sur chaque sommets u qui contient tous les voisins de u :

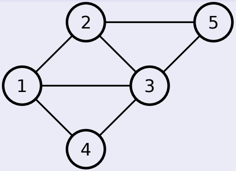
$$list(u) = [v|(u, v) \in E]$$

Remarques

- $list(u) = [v_1, v_2, \dots]$ signifie que v_1, v_2, ... sont les voisins de u
- L'existence d'une arête (u, v) signifie que :
  - u est dans $list(v)$
  - v est dans $list(u)$
- u peut être dans $list(u)$ si la boucle (u, u) est dans le graphe
- Si le sommet u est isolé alors $list(u) = []$

Dans un graphe non-orienté

Dessin de graphe et les listes d'adjacence



$$
\begin{aligned}
list(1) &= \quad [2,3,4]\\
list(2) &= \quad [1,3,5]\\
list(3) &= \quad [1,2,4,5]\\
list(4) &= \quad [1,3]\\
list(5) &= \quad [2,3]
\end{aligned}
$$

Dans cet exemple

- L'arête $(2,3)$ existe donc $list(2)$ contient 3 et $list(3)$ contient 2
- Il n'y a pas d'arête entre les sommets 1 et 5 donc $list(1)$ ne contient pas 5 et $list(5)$ ne contient 1
- Il n'y a pas de boucle donc $list(u)$ ne contient pas u
- Il n'y a pas de sommets isolé, toutes les $list(u) \neq []$

Dans un graphe orienté

Les listes d'adjacence

Un graphe orienté $G = (V, A)$ peut être représenté par une liste sur chaque sommets u qui contient tous les voisins sortants de u :

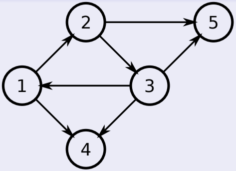
$$list(u) = [v|(u, v) \in A]$$

Remarques

- $list(u) = [v_1, v_2, \ldots]$ signifie que v_1, v_2, ... sont les voisins sortants de u
- L'existence d'un arc (u, v) signifie que :
  - v est dans $list(u)$
  - u n'est pas nécessairement dans $list(v)$ car l'arc (v, u) peut ne pas être présent
- u peut être dans $list(u)$ si la boucle (u, u) est dans le graphe
- Si le sommet u n'a pas de voisins sortants alors $list(u) = []$

Dans un graphe orienté

Dessin de graphe et les listes d'adjacence



$$\begin{aligned} list(1) &= \quad [2,4] \\ list(2) &= \quad [3,5] \\ list(3) &= \quad [1,4,5] \\ list(4) &= \quad [] \\ list(5) &= \quad [] \end{aligned}$$

Dans cet exemple

- L'arc $(2,3)$ existe donc $list(2)$ contient 3 et $list(3)$ ne contient pas 2 car l'arc $(3,2)$ n'existe pas
- Il n'y a pas d'arc entre les sommets 1 et 5 donc $list(1)$ ne contient pas 5 et $list(5)$ ne contient 1
- Il n'y a pas de boucle donc $list(u)$ ne contient pas u
- Les sommets 4 et 5 n'ont pas de voisins sortants donc $list(4)$ est vide et $list(5)$ est vide

Place et coût de l'accès aux informations

Place en mémoire

- D'un graphe quelconque : de l'ordre de $n + m$
- Moins de place en mémoire pour un graphe très peu dense (très peu d'arêtes)

Coût d'accès aux informations

- Savoir si u et v sont voisins : de l'ordre de $d(u)$
- Savoir si u est isolé : constant
- Trouver tous les voisins de u : de l'ordre de $d(u)$
- Compter le nombre d'arêtes/arcs du graphe : de l'ordre de $n + m$

Dans un graphe non-orienté

La matrice d'incidence

Un graphe non-orienté $G = (V, E)$ peut être représenté par une matrice d'incidence A de taille $n \times m$.

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1m} \\ a_{21} & a_{22} & \dots & a_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nm} \end{pmatrix}$$

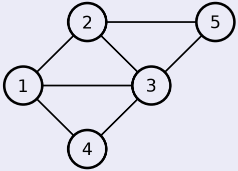
$$\text{où } a_{ue} = \begin{cases} 1 & \text{si } u \text{ est une extrémité de l'arête } e \\ 0 & \text{sinon} \end{cases}$$

Remarques

- La matrice d'incidence n'est pas carrée
- Deux colonnes identiques signifie qu'il existe une arête multiple
- Une colonne qui contient un unique 1 correspond à une boucle

Dans un graphe non-orienté

Dessin de graphe et sa matrice d'incidence



$$\begin{pmatrix} 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$

Dans cet exemple

- L'arête $(2, 3)$ existe et correspond à la troisième colonne de la matrice : $a_{23} = a_{33} = 1$
- Il n'y a pas d'arête entre les sommets 1 et 5 donc il n'existe pas de colonne k telle que $a_{1k} = a_{5k} = 1$
- Il n'y a pas de boucle donc il n'y a pas de colonne contenant un unique 1
- Il n'y a pas de sommets isolé donc la matrice ne contient pas de ligne nulle

Dans un graphe orienté

La matrice d'incidence

Un graphe orienté $G = (V, A)$ (en général sans arête multiple) peut être représenté par une matrice d'incidence A de taille $n \times m$.

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1m} \\ a_{21} & a_{22} & \dots & a_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nm} \end{pmatrix}$$

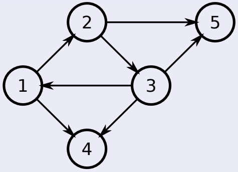
$$\text{où } a_{ue} = \begin{cases} -1 & \text{si } u \text{ est l'origine de } e \\ 1 & \text{si } u \text{ est la destination de } e \\ 0 & \text{sinon} \end{cases}$$

Remarques

- La matrice d'incidence n'est pas carrée
- Deux colonnes identiques signifie qu'il existe un arc multiple
- Une colonne qui contient un unique -1 correspond à une boucle

Dans un graphe orienté

Dessin de graphe et sa matrice d'incidence



$$\begin{pmatrix} -1 & -1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & -1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & -1 & -1 & -1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$

Dans cet exemple

- L'arc $(2,3)$ existe et correspond à la troisième colonne de la matrice : $a_{23} = -1$ et $a_{33} = 1$
- Il n'y a pas d'arête entre les sommets 1 et 5 donc il n'existe pas de colonne k telle que $a_{1k} = -1$ et $a_{5k} = 1$ ou $a_{1k} = 1$ et $a_{5k} = -1$
- Il n'y a pas de boucle donc il n'y a pas de colonne contenant un unique -1
- Il n'y a pas de sommets isolé donc la matrice ne contient pas de ligne nulle
- Les sommets 4 et 5 n'ont pas d'arc sortant donc les lignes 4 et 5 de la matrice ne contiennent pas de valeur -1

Place et coût de l'accès aux informations

Place en mémoire

- D'un graphe quelconque : de l'ordre de $n \cdot m$
- Moins de place en mémoire pour un graphe très peu dense (très peu d'arêtes)

Coût d'accès aux informations

- Savoir si u et v sont voisins : de l'ordre de m
- Savoir si u est isolé : de l'ordre de m
- Trouver tous les voisins de u : de l'ordre de $m + n$
- Compter le nombre d'arêtes/arcs du graphe : constant

Utilisation de classes

Classe Graphe

- Attributs : listes de sommets et d'arêtes/arcs, nom, orienté ou non, ...
- Méthodes : ajouter/supprimer des sommets/arêtes/arcs, donner l'ensemble des sommets/arêtes/arcs, dessiner le graphe, ...

Classe Sommet

- Attributs : nom, indice, couleur, liste de voisins, ...
- Méthodes : ajouter/supprimer arête incidente, donner ensemble des arêtes incidentes, donner ensemble des voisins, ...

Classe Arête/Arc

- Attributs : nom, indice, sommets extrémités, ...
- Méthodes : donner sommets incidents, orientation, ...

Place et coût de l'accès aux informations

Place en mémoire

- D'un graphe quelconque : dépend de la structure derrière les objets

Coût d'accès aux informations

- Dépend de la structure derrière les objets

Remarque

- En général, au plus les structures permettent un accès rapide aux informations (via beaucoup de références entre objets), au plus cela prend de place en mémoire

Comparaisons selon la représentation

Ordre de grandeur de la place en mémoire

	Matrice d'adjacence	Listes d'adjacence	Matrice d'incidence
Place	n^2	$n + m$	$n \cdot m$
Gain possible dans un graphe peu dense	non	oui	oui

Ordre de grandeur d'accès aux informations

	Matrice d'adjacence	Listes d'adjacence	Matrice d'incidence
u et v voisins ?	constant	$d(u)$	m
u isolé ?	n	constant	m
Tous les voisins de u	n	$d(u)$	$m + n$
Nombre d'arêtes/arcs	n^2	n^2	constant