

Rapport de projet PIM

Équipe CD 08 – Paul Louka, Nicolas Bailliet

Janvier 2024

- 1 Introduction
- 2 Architecture du programme
- 3 Structures et algorithmes
- 4 Conception et difficultés surmontées
- 5 Analyse des performances

Introduction

Rapport de
projet PIM

Équipe CD 08
– Paul Louka,
Nicolas
Bailliet

Introduction

Architecture
du programme

Structures et
algorithmes

Conception et
difficultés
surmontées

Analyse des
performances

L'algorithme de PageRank cherche à trier les N différents nœuds d'un graphe selon leur degré sortant, mais aussi le nombre de nœuds qui possèdent un arc l'atteignant, menant à une sorte d'ordre d'importance.

La formule théorique mène à un algorithme itératif pour le calcul des poids de chaque nœud.

$$\pi_{k+1}(n_i) = \sum_{\substack{j \\ (j,i) \in \mathcal{A}}} \frac{\pi_k(n_j)}{|n_j|}$$

Introduction

Rapport de
projet PIM

Équipe CD 08
– Paul Louka,
Nicolas
Bailliet

Introduction

Architecture
du programme

Structures et
algorithmes

Conception et
difficultés
surmontées

Analyse des
performances

On définit pour cela la matrice G à partir de S , matrice d'adjacence dont chaque ligne est divisée par le degré sortant, valant $\frac{1}{N}$ sur toute la ligne si le degré est nul. J est la matrice pleine de 1.

$$G = \alpha \cdot S + \frac{(1 - \alpha)}{N} \cdot J \quad \alpha \in [0, 1]$$

$$\pi_0^T = \left(\frac{1}{N}, \frac{1}{N}, \dots, \frac{1}{N} \right)$$

$$\forall k > 0, \pi_{k+1}^T = \pi_k^T \cdot G$$

Nous avons implémenté cet algorithme avec deux méthodes, d'une part avec des matrices classiques dites pleines de complexité spatiale $O(N^2)$, puis d'autre avec des matrices dites creuses, spatialement en $O(|\mathcal{A}|)$.

Architecture du programme

Rapport de
projet PIM

Équipe CD 08
– Paul Louka,
Nicolas
Bailliet

Introduction

Architecture
du programme

Structures et
algorithmes

Conception et
difficultés
surmontées

Analyse des
performances

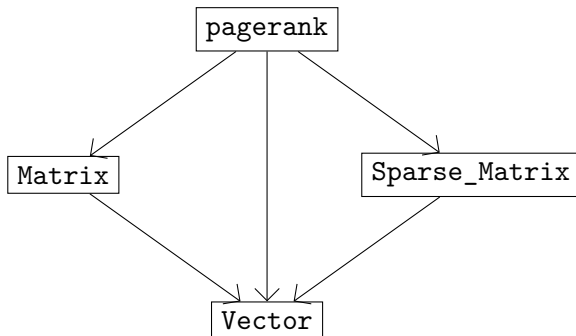


Figure – Architecture of the pagerank program.

Structures et algorithmes

Rapport de
projet PIM

Équipe CD 08
– Paul Louka,
Nicolas
Bailliet

Introduction

Architecture
du programme

Structures et
algorithmes

Conception et
difficultés
surmontées

Analyse des
performances

- type universel `T_Element` pour toutes les valeurs manipulées dans les programmes

```
type T_element is digits <>;
```

- Compressed Sparse Columns (CSC)
 - `Values` : la liste des valeurs de la matrice lue par colonne de gauche à droite
 - `Row_Indices` : la liste des indices de ligne des valeurs correspondantes
 - `Column_Pointers` : la liste de taille $N + 1$ des indices délimitant une colonne dans `Values`

Structures et algorithmes

Rapport de
projet PIM

Équipe CD 08
– Paul Louka,
Nicolas
Bailliet

Introduction

Architecture
du programme

Structures et
algorithmes

Conception et
difficultés
surmontées

Analyse des
performances

```
type T_Elements is array (1..NNZ) of T_Element;  
type T_Row_Indices is array (1..NNZ) of Positive;  
type T_Column_Indices is array (1..N+1) of Positive;  
  
type T_Sparse_Matrix is record  
    Values: T_Elements;  
    Row_Indices: T_Row_Indices;  
    Column_Pointers: T_Column_Indices;  
end record;
```

Structures et algorithmes

Rapport de
projet PIM

Équipe CD 08
– Paul Louka,
Nicolas
Bailliet

Introduction

Architecture
du programme

Structures et
algorithmes

Conception et
difficultés
surmontées

Analyse des
performances

- utilisation de `Coefficients`, liste des nœuds de degré sortant nul, instance de `Ada.Containers.Vectors`
- l'ensemble des types du programme est rendu générique (notamment pour `T_Element` et N)
- algorithmes de tri en $O(N \log(N))$, comme le `QuickSort` instancié avec `Ada.Containers.Generic_Array_Sort`.

Conception et difficultés surmontées

Rapport de
projet PIM

Équipe CD 08
– Paul Louka,
Nicolas
Bailliet

Introduction

Architecture
du programme

Structures et
algorithmes

Conception et
difficultés
surmontées

Analyse des
performances

$$\pi_{k+1}(n_i) = \sum_{\substack{j \\ (j,i) \in \mathcal{A}}} \alpha \cdot \frac{\pi_k(n_j)}{|n_j|} + \sum_l \alpha \cdot \frac{\pi_k(\text{Coefficients}(l))}{N} + \frac{1 - \alpha}{N}$$

Analyse des performances

Rapport de
projet PIM

Équipe CD 08
– Paul Louka,
Nicolas
Bailliet

Introduction

Architecture
du programme

Structures et
algorithmes

Conception et
difficultés
surmontées

Analyse des
performances

Graphe	Plein	Creux
sujet	0,003	0,007
worm	0,084	0,008
brainlinks	N/A	3,553
linux26	N/A	4,363

Table – Performances temporelles (s) sur dragon.enseeiht.fr