

# Projet d'intégration d'IA

## Optimisation d'affectations de bus sur un réseau



Mathias Dacosta  
Luc Dupuy  
Thomas Prak  
Cyril Py

# Plan

- ◇ Présentation du projet
- ◇ Gestion de projet
- ◇ Les algorithmes
- ◇ Conclusion
- ◇ Ouverture

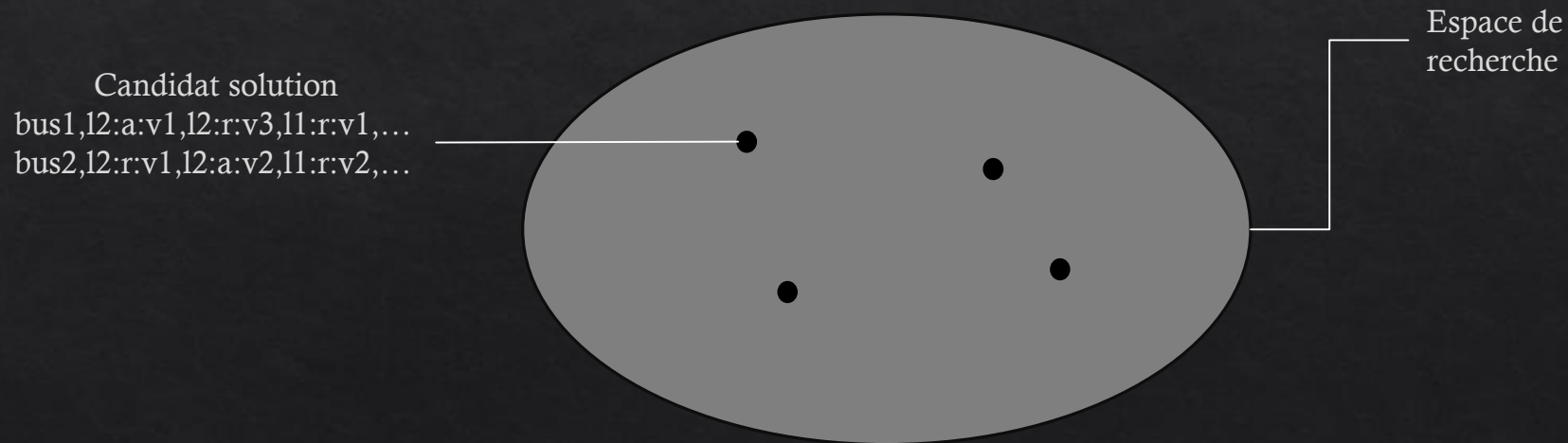
# Présentation du projet

# Présentation du projet

Affecter des bus sur un réseau de transport

Point à optimiser :

- Le nombre de bus
- La distance totale parcourue par les bus
- Le temps de trajet des bus



# Gestion de projet

# Gestion de projet

Membres du groupe:





# Gestion de projet

## Organisation:

Algorithme d'attribution itératif

Algorithme génétique

# Les algorithmes



# Les algorithmes

## attribution itératif

**POUR CHAQUE** Lignes

**POUR CHAQUE** Voyage

**SI** des bus déployés sont admissibles pour ce voyage

Chercher le bus le plus proche

Affecter ce bus au trajet

Mémoriser la prochaine horaire de disponibilité et position du bus

**SINON**

Affecter un bus non déployé

Mémoriser la prochaine horaire de disponibilité et position du bus

# Les algorithmes

## attribution itératif

- Amélioration possible:
  - Ajouter un algorithme d'optimisation
  - Ce servir des résultats dans un autre algorithme

# Algorithme Génétique

◇ Parties réalisées:

◇ Population initiale

◇ Évaluation

◇ Choix meilleure solution

# Modélisation – Algorithme Génétique

◇ Données d'entrée (Traitement CSV) -> Tab[ligne][sens][voyage][td,hd,ta,ha,d]\*

ligne	sens	voyage	td , hd , ta , ha , d
1	a	1	1 , 485* , 3 , 500, 10
1	a	2	1 , 540 , 3 , 560 , 10
...			
1	r	1	3 , 570 , 1 , 590 ,10
...	...	...	...
13	r	9	13 , 1156 , 22 , 1176 , 5

\*td: terminus départ

\*hd: heure départ -> ex: 8:05h ->  $8 \times 60 + 05 = 485$

\*ta: terminus arrivé

\*ha: heure arrivée

\*d: distance

# Algorithme Génétique - Modélisation

- ◆ Première modélisation d'un individu dans la population



N°Solution	N°Bus	N°Ligne	Sens	N°Voyage
1	2	3	« a »	4
1	2	3	« a »	5

Population

- ◆ Problème: un individu n'est pas un candidat solution mais seulement une partie d'un candidat solution



# Algorithme Génétique - Modélisation

## ◇ Deuxième Modélisation

◇ Format des données -> listVoyages[idVoyage][td,hd,ta,ha,d,nVoy, bus]

idVoyage	td, hd , ta , ha , sens, d , nVoy, bus *
1	1 , 487 , 3 , 500, a, 10 , 1 , null
2	1 , 540 , 3 , 560 , a , 10 ,2 , null
...	
538	13, 1021, 22 , 1046 , r , 7 , null
539	13 , 1156 , 22 , 1176 , r , 5 , null

\*td: terminus départ

\*hd: heure départ

\*ta: terminus arrivé

\*ha: heure arrivée

\*d: distance

\*nVoy: numéro du voyage de la ligne

\*bus: bus « x » affecté au voyage

ex: 8:07 ->  $8 \times 60 + 07 = 487\text{min}$



# Algorithme Génétique - Modélisation

## ◇ Deuxième Modélisation

### ◇ Modélisation d'un individu



### ◇ Population



# Algorithme Génétique - Modélisation

- ◇ Génération de la population

- ◇ **Tableau de référence** : listVoyages [idVoyage] [td,hd,ta,ha,d,nVoy, bus]



- ◇ **Modèle d'un individu**: modeleIndividu [x] avec x de 1 à 539



- ◇ **Fonction php**: *shuffle()*



- ◇ **Enregistrement des individus et référence** : tabPopulation [] = modeleIndividu;

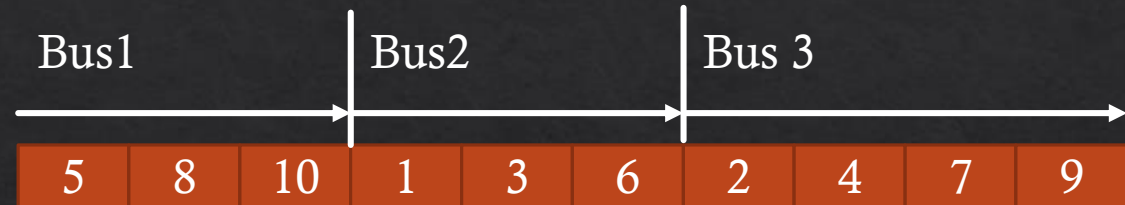


# Algorithme Génétique

## ◇ Évaluation

### ◇ Reste à faire:

- ◇ attribuer les bus aux voyages



### ◇ Réalisé:

- ◇ Compter le nombre total de bus
- ◇ Compter le nombre total de minutes
- ◇ Compter le nombre total de kilomètres
- ◇ Choix de la meilleure solution

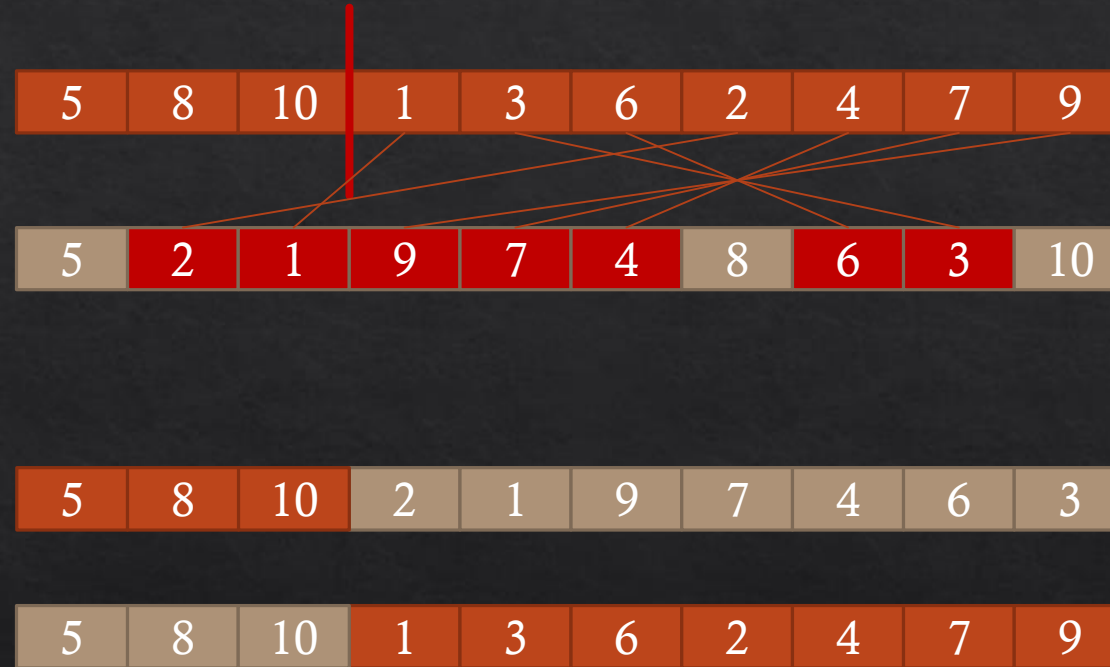
# Algorithme Génétique

## ◇ Croisement



# Algorithme Génétique

## ◇ Croisement





# Algorithme Génétique - Modélisation

## ◇ Mutation

◇ Principe: échange de deux voyages d'un même enfant





# Conclusion

# Conclusion

Points négatif:

- Mauvaise modélisation du problème

Points positif:

- Découverte de nouveaux algorithmes
- L'importance d'avoir une bonne modélisation au départ

# Overture

# Ouverture

- Finir les algorithmes
- Travailler sur l'optimisation