

L'Université Ouverte Lyon 1 vous présente le cycle de conférences

Impact de l'informatique sur la société et sur nos vies

Quelle intelligence dans les « Smart Cities » ? Christine Solnon - INSA de Lyon / LIRIS

### Plan du cours

Quelle intelligence pour les Smart Cities?

- Qu'est-ce qu'une ville intelligente ?
- 2 Illustration : Smart Deliveries
- 3 Optimisation avec des données qui dépendent du temps
- Optimisation avec des données incertaines
- 5 Conclusion

# De la logistique urbaine aux villes intelligentes<sup>1</sup>

#### Logistique urbaine : Un concept aussi ancien que les villes

- Réponses techniques et logistiques pour faciliter la vie des citadins

   → Adduction d'eau, Évacuation des eaux usées, Transport, etc

#### Villes intelligentes : Logistique urbaine revisitée par les TIC

Villes qui exploitent en temps réel les données qu'elles produisent :

- Collecter des données (variées, en grande quantité)
- Analyser les données pour optimiser les fonctions et services

Voir le rapport de Ludovic Viévard (Octobre 2014) http://www.millenaire3.com/dossiers/comprendre-orchestrer-et-vivre-la-ville-intelligente

# De la logistique urbaine aux villes intelligentes<sup>1</sup>

#### Logistique urbaine : Un concept aussi ancien que les villes

- Réponses techniques et logistiques pour faciliter la vie des citadins
  - → Adduction d'eau, Évacuation des eaux usées, Transport, etc.
- Solutions permettant aux villes de croître davantage
  - ∼→ Cercle vertueux (... ou vicieux selon le point de vue !)

#### Villes intelligentes : Logistique urbaine revisitée par les TIC

Villes qui exploitent en temps réel les données qu'elles produisent :

- Collecter des données (variées, en grande quantité)
- Analyser les données pour optimiser les fonctions et services

Voir le rapport de Ludovic Viévard (Octobre 2014)
http://www.millenaire3.com/dossiers/comprendre-orchestrer-et-vivre-la-ville-intelligente

# 3 approches de la ville intelligente (Image: http://www.millenaire3.com)

#### LA TECHNO-CITÉ

Principalement organisé autour des géants de l'équipement et des services, le modèle repose avant tout sur les infrastructures mises en place pour gérer les flux : installation de capteurs. récupération et traitement des données en temps réel. Plusieurs de ces smart cities sont conçues comme des démonstrateurs: à la fois lieu d'expérimentation grandeur nature et vitrine commerciale de l'innovation.



UN MONITORING TECHNIQUE

Ville monitorée pour la mesure des flux automobiles, de consommation d'énergie, du nombre de voyageurs en attente d'un métro, d'un bus, etc



Ex.: Songdo (Corée), Masdar (Émirats Arabes Unis), Putraiava (Malaisie), PlanIT Valley (Portugal)

#### LA E-CITÉ

Organisée autour de l'institution publique qui oriente la production d'infrastructures et l'écosystème d'acteurs au profit d'une "gouvernance intelligente" de la ville. Il ne s'agit ni de confier la gestion des flux à une unité de contrôle centralisée ni de laisser les usagers co-produire une solution, mais de créer les conditions d'une gestion partagée, plus efficace et aussi moins coûteuse pour l'institution.



#### UNE ORCHESTRATION PUBLIQUE DES DONNÉES



Big data et open data Collectée en grand nombre. croisée, utilisée en temps réel. publiée, etc., la donnée est le carburant de la ville intelligente



Plateformes et formats

Ex.: BigApps Challenges (New York), data.grandlyon.com

#### LA VILLE CONTRIBUTIVE

Organisée autour des usagers et des acteurs de l'économie collaborative, producteurs de "l'intelligence urbaine". Ouvertes, neutres et inter-opérables, les infrastructures et les données facilitent la production et l'usage d'applications et de services. Le modèle s'organise non pas verticalement, entre les utilisateurs et un super cerveau urbain, mais horizontalement entre les réseaux et les communautés d'utilisateurs qui partagent de l'information ou des possibles.



#### **UNE VILLE COPRODUITE**

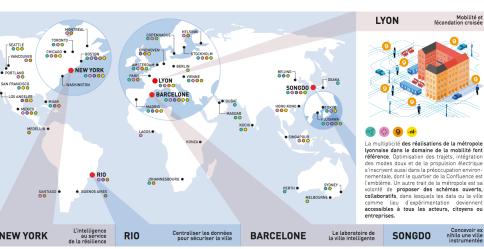
#### Makers Des bidouilleurs et des acteurs de

l'économie collaborative et du partage s'emparent de systèmes faiblement prientés par les acteurs publics et privés

Coproduction des services Les individus produisent des données. des services, voire des biens communs,

Fx · Associations "Citoyens-Capteurs", "Les petites cantines"

# Cartographie mondiale (Image: http://www.millenaire3.com)



#### Critères d'évaluation du degré d'intelligence des villes :



RÉSEAUX

EAUX S

MOBILI

•

SÉCURITÉ



CITOYENNETÉ ET ACTION PUBLIQUE **₩** 

ÉCONOMIE



ENVIRONNEMENT

# Les données, carburant des villes intelligentes

#### Origine des données

- Différents types de capteurs : mesures de températures, bruits, vitesses, positions, consommations d'énergie, ouverture de portes, ...
- Différents types de données : séries temporelles numériques, signaux, vidéos, images, tweets, ...
- --- Gros volume de données hétérogènes

#### Ouverture des données

- Données produites par les collectivités
  - → Peuvent être ouvertes (cf http://data.grandlyon.com)
- Données produites par des opérateurs privés
  - → Peuvent être ouvertes (cf TUBÁ)
- Données produites par les usagers
  - → Peuvent être restituées aux usagers (cf http://mesinfos.fing.org/)

Dans tous les cas : besoin de mécanismes pour protéger la vie privée...

# Les données, carburant des villes intelligentes

#### Origine des données

- Différents types de capteurs : mesures de températures, bruits, vitesses, positions, consommations d'énergie, ouverture de portes, ...
- Différents types de données : séries temporelles numériques, signaux, vidéos, images, tweets, ...
- → Gros volume de données hétérogènes

#### Ouverture des données

- Données produites par les collectivités
  - → Peuvent être ouvertes (cf http://data.grandlyon.com)
- Données produites par des opérateurs privés
  - → Peuvent être ouvertes (cf TUBA)
- Données produites par les usagers
   Peuvent être restituées aux usagers (cf http://mesinfos.fing.org/)
- Dans tous les cas : besoin de mécanismes pour protéger la vie privée...

# Exploitation des données

→ 3 niveaux du Data Analytics

#### Descriptive / Diagnostic Analytics:

Extraire des connaissances à partir des données

Quelles sont les conditions de trafic en ce moment ?

#### Predictive Analytics

Construire des modèles pour prévoir le futui

Quelles seront les conditions de trafic dans 30 mn ?

#### Prescriptive Analytics:

Assister la prise de décision

Quel est le meilleur itinéraire si je pars à 8:25 ?



# Exploitation des données

→ 3 niveaux du Data Analytics

#### Descriptive / Diagnostic Analytics:

Extraire des connaissances à partir des données

Quelles sont les conditions de trafic en ce moment ?

### Predictive Analytics:

Construire des modèles pour prévoir le futur

Quelles seront les conditions de trafic dans 30 mn ?

#### Prescriptive Analytics:

Assister la prise de décision

Quel est le meilleur itinéraire si je pars à 8:25 ?



# Exploitation des données

→ 3 niveaux du Data Analytics

#### Descriptive / Diagnostic Analytics:

Extraire des connaissances à partir des données

Quelles sont les conditions de trafic en ce moment ?

### Predictive Analytics:

Construire des modèles pour prévoir le futur

Quelles seront les conditions de trafic dans 30 mn ?

### Prescriptive Analytics:

Assister la prise de décision

Quel est le meilleur itinéraire si je pars à 8:25 ?



### Quelle intelligence (artificielle) pour le *Data Analytics*?

#### **Descriptive analytics**

- Perception, Reconnaissance de formes
- Analyse statistique, Fouille de données
- Représentation des connaissances
- ..

#### **Predictive analytics**

- Apprentissage automatique, Classification, Clustering
- Simulation et modélisation multi-agents
- ...

#### Prescriptive analytics

- Optimisation sous contraintes, méta-heuristiques
- Planification
- Raisonnement dans l'incertain
- ...

### Plan du cours

Quelle intelligence pour les Smart Cities?

- Qu'est-ce qu'une ville intelligente ?
- Illustration : Smart Deliveries
- 3 Optimisation avec des données qui dépendent du temps
- Optimisation avec des données incertaines
- 5 Conclusion

# Le projet Optimod'Lyon

- Projet financé par l'ADEME
- Porteur : Grand Lyon
- 13 partenaires :
  - 2 collectivités : Lyon et Grand Lyon
  - 8 industriels: IBM, Renault Trucks, Orange, CityWay, Phoenix ISI, Parkeon, Autoroutes Trafic, Geoloc Systems
  - 3 laboratoires de recherche : LIRIS, CETE, LET
- Durée : 3 ans (2012-2015)
- Budget: 7 M Euros



# Objectifs d'Optimod'Lyon



#### Définition du problème :

Etant donnés un plan de ville, l'adresse d'un entrepôt et les adresses de points à livrer, calculer la tournée la plus courte

#### Résolution en deux étapes

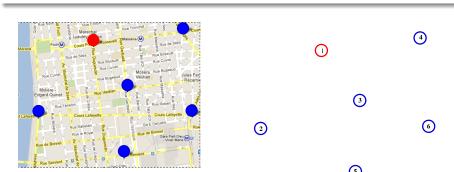
Calculer le graphe des plus courts chemin



#### Définition du problème :

Etant donnés un plan de ville, l'adresse d'un entrepôt et les adresses de points à livrer, calculer la tournée la plus courte

- Calculer le graphe des plus courts chemins
- Résoudre le Asymetric Traveling Salesman Problem (ATSP

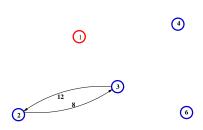


#### Définition du problème :

Etant donnés un plan de ville, l'adresse d'un entrepôt et les adresses de points à livrer, calculer la tournée la plus courte

- Calculer le graphe des plus courts chemins
- Résoudre le Asymetric Traveling Salesman Problem (ATSP

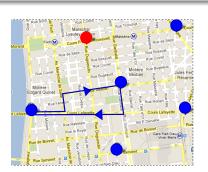


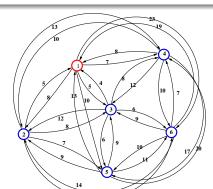


#### Définition du problème :

Etant donnés un plan de ville, l'adresse d'un entrepôt et les adresses de points à livrer, calculer la tournée la plus courte

- Calculer le graphe des plus courts chemins
- Résoudre le Asymetric Traveling Salesman Problem (ATSP



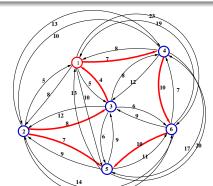


#### Définition du problème :

Etant donnés un plan de ville, l'adresse d'un entrepôt et les adresses de points à livrer, calculer la tournée la plus courte

- Calculer le graphe des plus courts chemins
- Résoudre le Asymetric Traveling Salesman Problem (ATSP)





### Qu'est-ce qui est difficile dans le problème classique ?

#### Calculer un plus court chemin entre deux points?

- → Problème facile et bien résolu :
  - Algorithmes efficaces (par exemple, Dijkstra)
  - Temps de calcul borné par un polynôme en fonction de la taille du plan

#### Résoudre l'ATSP dans le graphe des plus courts chemins ?

- → Problème difficile (NP-difficile)
  - En théorie : Pas d'algorithme garantissant de trouver la solution optimale en un temps raisonnable quand le nombre de points à livrer augmente

# Qu'est-ce qui est difficile dans le problème classique ?

#### Calculer un plus court chemin entre deux points?

- → Problème facile et bien résolu :
  - Algorithmes efficaces (par exemple, Dijkstra)
  - Temps de calcul borné par un polynôme en fonction de la taille du plan

#### Résoudre l'ATSP dans le graphe des plus courts chemins ?

- $\rightsquigarrow$  Problème difficile ( $\mathcal{NP}$ -difficile)
  - En théorie : Pas d'algorithme garantissant de trouver la solution optimale en un temps raisonnable quand le nombre de points à livrer augmente
  - En pratique : Utilisation de techniques d'Intelligence Artificielle
    - Branch & Bound : Contenir l'explosion en raisonnant
      - → Garanties sur l'optimalité, mais pas sur le temps d'exécution
      - → Efficace pour les problèmes symétriques (cf Concorde)
    - Méta-heuristiques : Contourner l'explosion en faisant des impasses
      - → Garanties sur le temps d'exécution, mais pas sur l'optimalité
      - → Trouvent souvent des solutions optimales, ou très proches

# Qu'est-ce qui est difficile dans le problème classique ?

#### Calculer un plus court chemin entre deux points?

- → Problème facile et bien résolu :
  - Algorithmes efficaces (par exemple, Dijkstra)
  - Temps de calcul borné par un polynôme en fonction de la taille du plan

#### Résoudre l'ATSP dans le graphe des plus courts chemins ?

- $\leadsto$  Problème difficile ( $\mathcal{NP}$ -difficile)
  - En théorie : Pas d'algorithme garantissant de trouver la solution optimale en un temps raisonnable quand le nombre de points à livrer augmente
  - En pratique : Utilisation de techniques d'Intelligence Artificielle
    - Branch & Bound : Contenir l'explosion en raisonnant
      - → Garanties sur l'optimalité, mais pas sur le temps d'exécution
      - → Efficace pour les problèmes symétriques (cf Concorde)
    - Méta-heuristiques : Contourner l'explosion en faisant des impasses ~ Garanties sur le temps d'exécution, mais pas sur l'optimalité
      - → Trouvent souvent des solutions optimales, ou très proches

# Quelles données pour des Smart Deliveries?

### Capteurs électro-magnétiques :

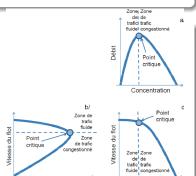
- 634 capteurs dans la chaussée
- Mesures toutes les 6 minutes :
  - Débit : nombre de véhicules pendant une période de temps
  - Concentration : nombre de véhicules sur un tronçon de voie



#### Diagramme fondamental:

Estimation de la vitesse à partir du débit et de la concentration :

- Trafic fluide : débit augmente quand concentration augmente
- Trafic congestionné : débit diminue quand concentration augmente



Concentration

Déhit

Image [Buisson et Lesort]

 Débits mesurés sur les capteurs 1 (haut) et 27 (bas) : Mercredi 01/16/2013 Lundi 01/14/2013 Vendredi 01/18/2013 Dimanche 01/20/2013 • Taux d'occupation mesurés sur les capteurs 1 (haut) et 27 (bas) : Lundi 01/14/2013 Mercredi 01/16/2013 Vendredi 01/18/2013 Dimanche 01/20/2013

## Modèles prédictifs construits à partir de ces données

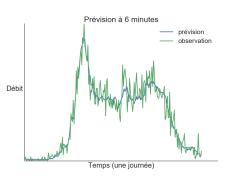
Prévisions à court terme (moins d'une heure)

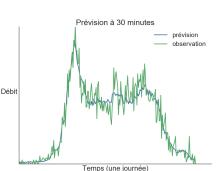
#### **Approches existantes:**

- Linéaires : Auto-régression linéaire (ARIMA)
- Non linéaires : k plus proches voisins (kNN), réseaux de neurone, ...

Possibilité de prendre en compte la dépendance spatio-temporelle

### Exemples de prévisions (avec kNN):





### Modèles prédictifs construits à partir de ces données Prévisions à long terme

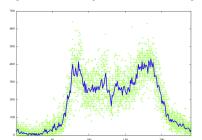
#### Clustering des journées pour chaque capteur :

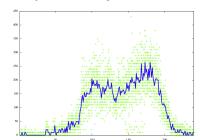
- Regrouper les journées ayant des séries temporelles similaires
- Peut être fait manuellement, ou automatiquement

#### Construction d'une série représentative pour chaque cluster :

Pour chaque pas de temps, recherche d'une valeur représentative 
→ moyenne ou médiane

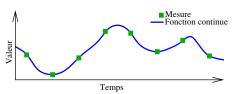
#### Exemple : Débits prévus sur une journée pour 2 tronçons





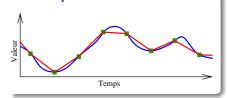
## Représentation de ces modèles prédictifs

#### Série temporelle = une mesure par pas de temps (6 minutes) :

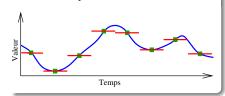


→ Modèles = une prévision par pas de temps

# Représentation par une fonction linéaire par morceaux :



# Représentation par une fonction constante par morceaux :



### Comment exploiter les prévisions pour optimiser les tournées ?

#### Au moment de préparer la tournée :

- Minimiser la durée de tournées, en tenant compte des vitesses prévues
- Tenir compte du fait que les vitesses varient dans le temps
- → Optimisation avec des données qui dépendent du temps

#### Au moment de réaliser la tournée :

- Adapter la tournée si les vitesses observées sont très différentes des vitesses prédites (évènements imprévus)
- Anticiper sur les évènements les plus probables en exploitant des statistiques sur le passé
- --- Optimisation avec des données incertaines

### Comment exploiter les prévisions pour optimiser les tournées ?

#### Au moment de préparer la tournée :

- Minimiser la durée de tournées, en tenant compte des vitesses prévues
- Tenir compte du fait que les vitesses varient dans le temps
- --- Optimisation avec des données qui dépendent du temps

#### Au moment de réaliser la tournée :

- Adapter la tournée si les vitesses observées sont très différentes des vitesses prédites (évènements imprévus)
- Anticiper sur les évènements les plus probables en exploitant des statistiques sur le passé
- → Optimisation avec des données incertaines

### Plan du cours

Quelle intelligence pour les Smart Cities?

- Qu'est-ce qu'une ville intelligente ?
- 2 Illustration : Smart Deliveries
- Optimisation avec des données qui dépendent du temps
- Optimisation avec des données incertaines
- 5 Conclusion

### Contexte du travail

#### Workpackage Smart Deliveries du projet Optimod'Lyon:

Exploiter des prévisions de vitesse qui dépendent du temps pour optimiser des tournées de livraison

#### Thèse de Penélope Aguiar Melgarejo:

- Financée par IBM et co-encadrée avec Philippe Laborie d'IBM
- Contributions:
  - Algorithmes pour exploiter des prévisions qui dépendent du temps
  - Evaluation de l'intérêt en pratique

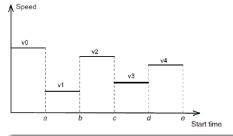
→ Calcul de la durée pour traverser un tronçon

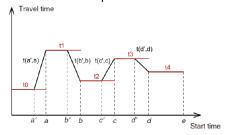
#### Calcul d'une durée dans la cas où la vitesse est constante :

durée = longueur / vitesse

#### Calcul d'une durée dans la cas où la vitesse dépend du temps :

La vitesse prévue peut changer pendant la traversée du tronçon → La durée n'est pas constante pour une fenêtre de temps donnée





Calcul des chemins les plus rapides

#### Calcul de la durée d'un chemin :

- Un chemin est une succession de tronçons
  - → Durée d'un chemin = somme des durées des tronçons
    - + durées des changements de tronçons
- Hypothèse de travail : les durées des changements sont nulles
  - --- Sous-estime les durées, surtout quand le trafic est congestionné

#### Calcul du chemin le plus rapide entre deux points :

- Le chemin le plus rapide peut changer en fonction de l'heure de départ
   Doit être calculé pour toutes les heures de départ possibles
- Les algorithmes peuvent être adaptés au cas où les données dépendent du temps sous réserve que les données soient FIFO :
  - Si  $t_i < t_j$  alors  $t_i +$  durée en partant à  $t_i \le t_j +$  durée en partant à  $t_j$

Autrement dit : on ne peut arriver plus tôt si on part plus tard

→ Calcul des chemins les plus rapides

#### Calcul de la durée d'un chemin :

- Un chemin est une succession de tronçons
  - → Durée d'un chemin = somme des durées des tronçons
     + durées des changements de tronçons
- Hypothèse de travail : les durées des changements sont nulles
   Sous-estime les durées, surtout quand le trafic est congestionné

#### Calcul du chemin le plus rapide entre deux points :

- Le chemin le plus rapide peut changer en fonction de l'heure de départ
   → Doit être calculé pour toutes les heures de départ possibles
- Les algorithmes peuvent être adaptés au cas où les données dépendent du temps sous réserve que les données soient FIFO :
  - Si  $t_i < t_j$  alors  $t_i +$  durée en partant à  $t_i \le t_j +$  durée en partant à  $t_j$

Autrement dit : on ne peut arriver plus tôt si on part plus tard

→ Modélisation mathématique du ATSP (en OPL)

h[pos[i+1]] == h[pos[i]] + duree[pos[i]][pos[i+1]]

Données en entrée : range Points = 1..nbPoints:

```
int duree[Points][Points] =...
Variables de décision (inconnues) :
dvar int pos[Points] in Points;
                                                   /* pos[i] = ième point visité */
dvar int h[Points] in T;
                                                /* h[i] = heure d'arrivée sur i */
Fonction objectif à optimiser et contraintes :
minimize h[v_{final}];
subject to {
    pos[1] == v_{init};
                                                                   /* On part de Vinit */
    h[v_{init}] == t_0;
                                                                      /* à l'heure to */
    pos[nbPoints] == V_{final};
                                                         /* et on termine sur V_{final} */
    allDifferent(pos);
                                           /* Chaque point est visité une fois */
    forall(i \in 1...nbPoints - 1){
```

### Algorithmes pour exploiter des prévisions qui dépendent du temps

→ Modélisation mathématique du TD-ATSP (en OPL)

```
Données en entrée :
range Points = 1..nbPoints;
range T = t_0 ... t_{horizon};
int duree[Points][Points][T] =...
Variables de décision (inconnues) :
dvar int pos[Points] in Points;
                                                    /* pos[i] = ième point visité */
dvar int h[Points] in T;
                                                  /* h[i] = heure d'arrivée sur i */
Fonction objectif à optimiser et contraintes :
minimize h[v_{final}];
subject to {
    pos[1] == v_{init};
                                                                     /* On part de Vinit */
    h[v_{init}] == t_0;
                                                                        /* à l'heure to */
    pos[nbPoints] == V_{final};
                                                          /* et on termine sur V_{final} */
    allDifferent(pos);
                                            /* Chaque point est visité une fois */
    forall(i \in 1...nbPoints - 1){
         h[pos[i+1]] == h[pos[i]] + duree[pos[i]][pos[i+1]][h[pos[i]]];
```

## Algorithmes pour exploiter des prévisions qui dépendent du temps

→ Recherche du plus court circuit (ATSP)

### Espace de recherche à explorer :

- Ensemble des permutations des points à visiter
- --- Le fait que les données dépendent du temps ne change rien

### Exploration de l'espace de recherche par *Branch & Bound*

- Construction d'un arbre de recherche (Branch)
- A chaque étape de la construction de l'arbre, calcul d'une borne (Bound)
  - Borne = Durée nécessaire pour visiter les points restants
  - Intelligence --- Compromis entre temps de calcul et qualité
- Calcul de Bound plus coûteux quand les données dépendent du temps
- → Besoin de développer des algorithmes dédiés pour ce type de données

# Algorithmes pour exploiter des prévisions qui dépendent du temps

→ Recherche du plus court circuit (ATSP)

#### Espace de recherche à explorer :

- Ensemble des permutations des points à visiter
- → Le fait que les données dépendent du temps ne change rien

### Exploration de l'espace de recherche par Branch & Bound

- Construction d'un arbre de recherche (Branch)
- A chaque étape de la construction de l'arbre, calcul d'une borne (Bound)
  - Borne = Durée nécessaire pour visiter les points restants
  - Intelligence → Compromis entre temps de calcul et qualité
- Calcul de Bound plus coûteux quand les données dépendent du temps
- --- Besoin de développer des algorithmes dédiés pour ce type de données

## Est-ce intéressant d'exploiter des données dépendant du temps ?

→ Constitution d'un jeu d'essai

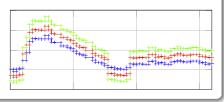
### Fabrication de tournées à partir de données réelles :

- 255 adresses de livraison provenant de tournées réelles
- Génération de tournées de tailles variables (entre 10 et 100 points)
   Choix aléatoire des points parmi les 255 adresses



### Calcul des plus courts chemins dépendant du temps :

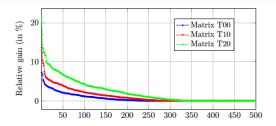
- En bleu : Durée d'un chemin = somme des durées des tronçons
   → Peu réaliste si congestions
- Augmentation des contrastes de 10% (rouge) et 20% (vert)



## Est-ce intéressant d'exploiter des données dépendant du temps ?

### Comment évaluer le gain ?

- Calcul d'une tournée optimale avec des données non dépendant du temps (durée médiane)
  - → t = durée de cette tournée quand les données dépendent du temps
- Calcul d'une tournée optimale avec des données dépendant du temps
   → t<sub>TD</sub> = durée de cette tournée
- Gain relatif =  $\frac{t t_{TD}}{t_{TD}}$



- Pour quelques instances, le gain est important (jusque 20% pour T20)
- Pour la moitié des instances il est quasi nul

## Est-ce intéressant d'exploiter des données dépendant du temps ?

--- Instances comportant des contraintes sur les heures de livraison

Nb de	Dilatation	Nb résolues		Gain TD/statique		
points	des durées	statique	TD	Moy	Min	Max
10	0%	10	20	19%	0%	95%
10	10%	9	20	21%	0%	99%
10	20%	10	20	22%	0%	99%
20	0%	10	20	13%	0%	50%
20	10%	7	20	20%	0%	50%
20	20%	4	20	8%	3%	18%
30	0%	10	20	9%	0%	61%
30	10%	5	20	17%	1%	62%
30	20%	1	20	6%	6%	6%

Il faut exploiter des données dépendant du temps pour respecter les horaires

### Plan du cours

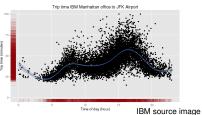
Quelle intelligence pour les Smart Cities?

- Qu'est-ce qu'une ville intelligente?
- 2 Illustration : Smart Deliveries
- Optimisation avec des données qui dépendent du temps
- Optimisation avec des données incertaines
- 5 Conclusion

### **Motivations**

#### Optimisation avec des données issues de modèles prédictifs :

- Prévisions basées sur des historiques: Valeurs moyennes ou médianes
- L'écart type peut être très variable fiables que d'autres



Que faire quand les observations sont différentes des prévisions ?

#### Contexte de ce travail:

Thèse de Michael Saint Guillain, en co-tutelle entre l'Université Catholique de Louvain (Belgique) et l'INSA de Lyon, en co-encadrement avec Yves Deville

# Problèmes d'optimisation classiques

#### Modéliser le problème = Définir :

- Les données
- Les inconnues (var. de décision) et leurs valeurs possibles (domaines)
- Les contraintes à satisfaire
- La fonction objectif à optimiser

#### Solution:

Valuation des inconnues qui satisfait les contraintes et optimise la fct objectif

# Problèmes d'optimisation classiques

#### Modéliser le problème = Définir :

- Les données
- Les inconnues (var. de décision) et leurs valeurs possibles (domaines)
- Les contraintes à satisfaire
- La fonction objectif à optimiser

#### Solution:

Valuation des inconnues qui satisfait les contraintes et optimise la fct objectif

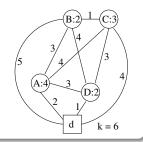
### Que faire quand données observées $\neq$ données initiales ?

- Recalculer une nouvelle solution avec les nouvelles données
- Inconvénients :
  - Le re-calcul peut être long
  - La nouvelle solution peut être bien plus mauvaise que celle qu'on aurait calculé si on avait anticipé / événements probables

# **Exemple: Vehicle Routing Problem (VRP)**

#### Données en entrée :

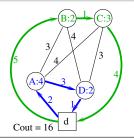
- Ensemble de points à visiter + un dépôt
- Distances (ou durées) entre points
- Demandes associées aux points
- Capacité k d'un véhicule



#### **Solution:**

Ensemble de circuits partant du dépôt tels que :

- Chaque point est visité une fois
- La somme des demandes des points d'un circuit ne dépasse pas k
- La distance (ou durée) totale est minimale



## **Exemple: Vehicle Routing Problem (VRP)**

**→ Livraisons avec plusieurs camions** 

#### Données incertaines pour le VRP :

- Ajout de nouvelles livraisons pendant la réalisation des tournées
- Durées observées ≠ prédictions à cause d'un accident
- Demande associée à un point supérieure à ce qui était prévu
- etc

### Re-calcul d'une solution alors que les tournées ont déjà commencé :

# Problèmes d'optimisation stochastiques

#### Modéliser le problème = Définir :

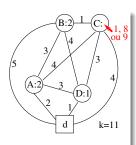
- Les données connues avec certitude
- Les données incertaines = var. aléatoires + distributions de probabilité
- Les inconnues (var. de décision) et leurs valeurs possibles (domaines)
- Les contraintes à satisfaire
- La fonction objectif à optimiser

#### **Exemple: VRP avec demandes stochastiques**

- Données certaines : points à livrer, distances, capacité des camions
- Données incertaines : demandes
- Distribution de probabilité des demandes :

• 
$$p(r_A = 2) = p(r_B = 2) = p(r_D = 1) = 1$$

• 
$$p(r_C = 1) = \frac{1}{3}, p(r_C = 8) = \frac{1}{3}, p(r_C = 9) = \frac{1}{3}$$



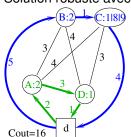
# Solution robuste d'un problème d'optimisation stochastique

### Garantit la faisabilité de la solution / une probabilité p donnée :

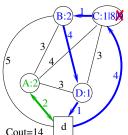
Solution = Valuation des inconnues optimisant la fonction objectif et telle que la probabilité que les contraintes soient satisfaites / réalisations des variables aléatoires soit supérieure à p

### Exemple: Solution robuste pour VRP avec demandes stochastiques

Solution robuste avec p = 1:



Solution robuste avec  $p = \frac{2}{3}$ :



# Soution flexible d'un problème d'optimisation stochastique

#### Optimalité des solutions adaptées / réalisations des var. aléatoires :

- Définir une procédure d'adaptation à la réalisation des var. aléatoires
   Procédure simple (rapide à appliquer)
- Avant le début de la réalisation des variables aléatoires (offline) :
   Calcul d'une solution a priori = Valuation des inconnues qui optimise l'espérance de la fct objectif pour la procédure d'adaptation
- A chaque fois qu'une variable aléatoire se réalise (online) :
   Application de la procédure d'adaptation

## Ex. : Procédure d'adaptation pour le VRP avec demandes stochastiques

Retour au dépôt si charge courante + demande du prochain point > k

Solution *a priori* pour k = 11:

Solution adaptée si  $r_C = 8$  ou 9 :



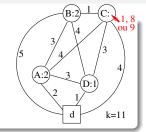


# Optimiser l'espérance vs Optimiser sur des données moyennes

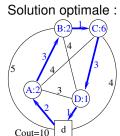
## Rappel du problème :

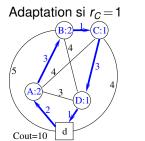
Distribution de probabilité des demandes :

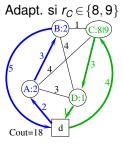
- $p(r_A = 2) = p(r_B = 2) = p(r_D = 1) = 1$
- $p(r_C = 1) = \frac{1}{3}, p(r_C = 8) = \frac{1}{3}, p(r_C = 9) = \frac{1}{3}$



# Que se passe-t-il si on optimise sur des données moyennes (i.e. $r_C = 6$ )?







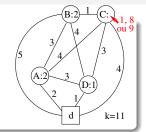
 $\rightarrow$  Espérance du coût =  $\frac{10+18+18}{3} = \frac{46}{3}$ . Peut-on faire mieux ?

# Optimiser l'espérance vs Optimiser sur des données moyennes

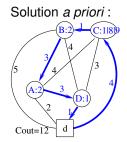
## Rappel du problème :

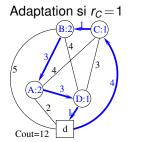
Distribution de probabilité des demandes :

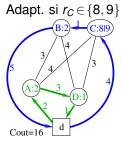
- $p(r_A = 2) = p(r_B = 2) = p(r_D = 1) = 1$
- $p(r_C = 1) = \frac{1}{3}$ ,  $p(r_C = 8) = \frac{1}{3}$ ,  $p(r_C = 9) = \frac{1}{3}$



## Optimisation de l'espérance du coût des solutions adaptées :







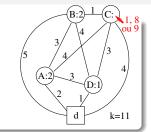
 $\rightarrow$  Espérance du coût =  $\frac{12+16+16}{3} = \frac{44}{3}$ . C'est mieux !

# Optimiser l'espérance vs Optimiser sur des données moyennes

## Rappel du problème :

Distribution de probabilité des demandes :

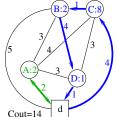
- $p(r_A = 2) = p(r_B = 2) = p(r_D = 1) = 1$
- $p(r_C = 1) = \frac{1}{3}, p(r_C = 8) = \frac{1}{3}, p(r_C = 9) = \frac{1}{3}$

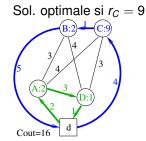


### Et si on savait prévoir le futur avec certitude ?

Sol. optimale si  $r_C = 1$  Sol. optimale si  $r_C = 8$ 

Cout=10





 $\rightarrow$  Espérance du coût =  $\frac{10+14+16}{3} = \frac{40}{3}$ . C'est encore mieux, mais on a triché!

## Est-ce difficile de trouver une solution flexible optimale?

→ Calcul de l'espérance des adaptations d'une solution a priori donnée

### Calcul par énumération des scénarios pour une solution a priori x :

- Enumérer tous les scénarios et, pour chaque scénario, calculer :
  - sa probabilité de réalisation
  - le coût de l'adaptation de x à ce scénario
- Espérance de *x* = somme des coûts pondérés par les probabilités

Problème : le nombre de scénarios est très grand (exponentiel... ou infini) Approximation par échantillonnage (Monte-Carlo Sampling)

#### Calcul direct :

Ex. : Esp. de la longueur d'un tour / proba d'apparition des points à visiter  $\rightarrow$  Soit t(i) = esp. de la longueur pour retourner du  $i^{\text{ème}}$  point au dépot

- Si i = dernier sommet, alors t(i) = cout(i, depot)
- Sinon,  $t(i) = \sum_{j \in suivants(i)} Pr(i,j) * (cout(i,j) + t(j))$ où Pr(i,j) = probabilité que j soit le premier point présent après i $\Rightarrow Pr(i,j) = (1 - p(i+1)) * (1 - p(i+2)) * \dots * (1 - p(j-1)) * p(j)$
- → Beaucoup plus rapide, mais pas toujours possible...

# Est-ce difficile de trouver une solution flexible optimale ?

→ Calcul de l'espérance des adaptations d'une solution a priori donnée

#### Calcul par énumération des scénarios pour une solution a priori x :

- Enumérer tous les scénarios et, pour chaque scénario, calculer :
  - sa probabilité de réalisation
  - le coût de l'adaptation de x à ce scénario
- Espérance de x = somme des coûts pondérés par les probabilités

Problème : le nombre de scénarios est très grand (exponentiel... ou infini) Approximation par échantillonnage (Monte-Carlo Sampling)

#### Calcul direct:

Ex. : Esp. de la longueur d'un tour / proba d'apparition des points à visiter  $\sim$  Soit t(i) = esp. de la longueur pour retourner du  $i^{\text{ème}}$  point au dépot

- Si i = dernier sommet, alors t(i) = cout(i, depot)
- Sinon,  $t(i) = \sum_{j \in suivants(i)} Pr(i,j) * (cout(i,j) + t(j))$ où Pr(i,j) = probabilité que j soit le premier point présent après i
- $\rightsquigarrow Pr(i,j) = (1-p(i+1))*(1-p(i+2))*...*(1-p(j-1))*p(j)$
- → Beaucoup plus rapide, mais pas toujours possible...

# Est-ce difficile de trouver une solution flexible optimale?

→ Calcul d'une solution a priori dont l'espérance des adaptations est optimale

### Problèmes d'optimisation $\mathcal{NP}$ -difficiles :

- Approches exactes limitées aux petits problèmes
- En pratique : Utilisation d'approches heuristiques pour trouver rapidement une solution de bonne qualité
  - Necherche locale, Algorithmes génétiques, Colonies de fourmis, etc

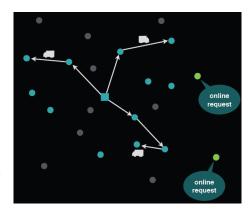
#### Exemple de résolution utilisant la recherche locale :

- Générer une solution a priori s (par ex., aléatoirement)
- Tant que limite de temps non dépassée faire :
  - Calculer l'espérance de *x* (par échantillonnage ou de façon exacte)
  - Modifier s localement :
    - → Changer les valeurs d'une ou plusieurs variables de décision
- Retourner la meilleure solution a priori trouvée

#### Stratégies possibles pour modifier localement s :

Recherche gloutonne, recuit simulé, recherche taboue, etc.

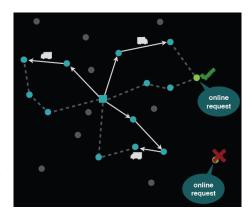
- Demandes révélées au fur et à mesure de la journée
- Connaissance stochastique : Probabilité de demande pour chaque arrêt et chaque heure
- Pour chaque demande, décider en tps réel de l'accepter ou non
   Si acceptée, alors adaptation dynamique des tournées
- Objectif : Minimiser le nombre de demandes rejetées à la fin de la journée



Evaluation expérimentale sur des instances artificielles

Diminution (entre 19% et 65%) du nombre de demandes rejetées A confirmer sur des instances réelles!

- Demandes révélées au fur et à mesure de la journée
- Connaissance stochastique : Probabilité de demande pour chaque arrêt et chaque heure
- Pour chaque demande, décider en tps réel de l'accepter ou non
   Si acceptée, alors adaptation dynamique des tournées
- Objectif: Minimiser le nombre de demandes rejetées à la fin de la journée

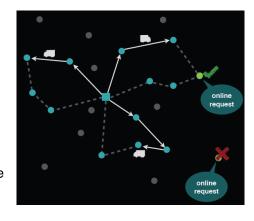


#### Evaluation expérimentale sur des instances artificielles

Diminution (entre 19% et 65%) du nombre de demandes rejetées 

A confirmer sur des instances réelles!

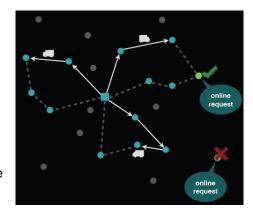
- Demandes révélées au fur et à mesure de la journée
- Connaissance stochastique : Probabilité de demande pour chaque arrêt et chaque heure
- Pour chaque demande, décider en tps réel de l'accepter ou non
   Si acceptée, alors adaptation dynamique des tournées
- Objectif : Minimiser le nombre de demandes rejetées à la fin de la journée



#### Evaluation expérimentale sur des instances artificielles

Diminution (entre 19% et 65%) du nombre de demandes rejetées A confirmer sur des instances réelles!

- Demandes révélées au fur et à mesure de la journée
- Connaissance stochastique : Probabilité de demande pour chaque arrêt et chaque heure
- Pour chaque demande, décider en tps réel de l'accepter ou non
   Si acceptée, alors adaptation dynamique des tournées
- Objectif : Minimiser le nombre de demandes rejetées à la fin de la journée



#### Evaluation expérimentale sur des instances artificielles

Diminution (entre 19% et 65%) du nombre de demandes rejetées

→ A confirmer sur des instances réelles!

### Plan du cours

Quelle intelligence pour les Smart Cities?

- Qu'est-ce qu'une ville intelligente?
- 2 Illustration : Smart Deliveries
- 3 Optimisation avec des données qui dépendent du temps
- Optimisation avec des données incertaines
- Conclusion

### Conclusion

#### Comment exploiter intelligemment les masses de données captées ?

- Analyse descriptive pour comprendre les usages
- Analyse prédictive pour prévoir les usages futurs
- Analyse prescriptive pour optimiser les usages :
  - Optimisation *Time-Dependent* pour les données temporelles
  - Optimisation Stochastique pour les données incertaines

#### Où sont les difficultés ?

- Problèmes d'optimisation difficiles
- Besoin de (toujours plus de) données
  - → Peut poser des problèmes de respect de la vie privée
- Besoin que les usagers adhèrent au système
  - ... ou que le système s'adapte aux besoins des usagers !

Domaines de recherche très actifs et pluri-disciplinaires