

Challenge d'optimisation 2009

AG41

Planification de tournées de véhicules

Le challenge

Qu'es ce que c'est ?

- Projet en équipe de 3
- Développer un algorithme d'optimisation sur un problème réel et l'implémenter informatiquement afin de trouver une bonne solution
- Challenge entre les étudiants

4^{ème} édition :

- 2006 : emploi du temps de l'UTBM
- 2007 : emploi du temps de l'UTBM
- 2008 : allocation de fréquence dans les réseaux Wi-Fi
- 2009 : tournées de véhicules

Problématique - Contexte

Planification des tournées de véhicules

Comment une entreprise peut livrer au moindre coût ses clients ?

- Elle possède un certains nombres de véhicules.
- Ces clients sont répartis dans différentes villes.

L'objectif est de minimiser le trajet de l'ensemble des véhicules.

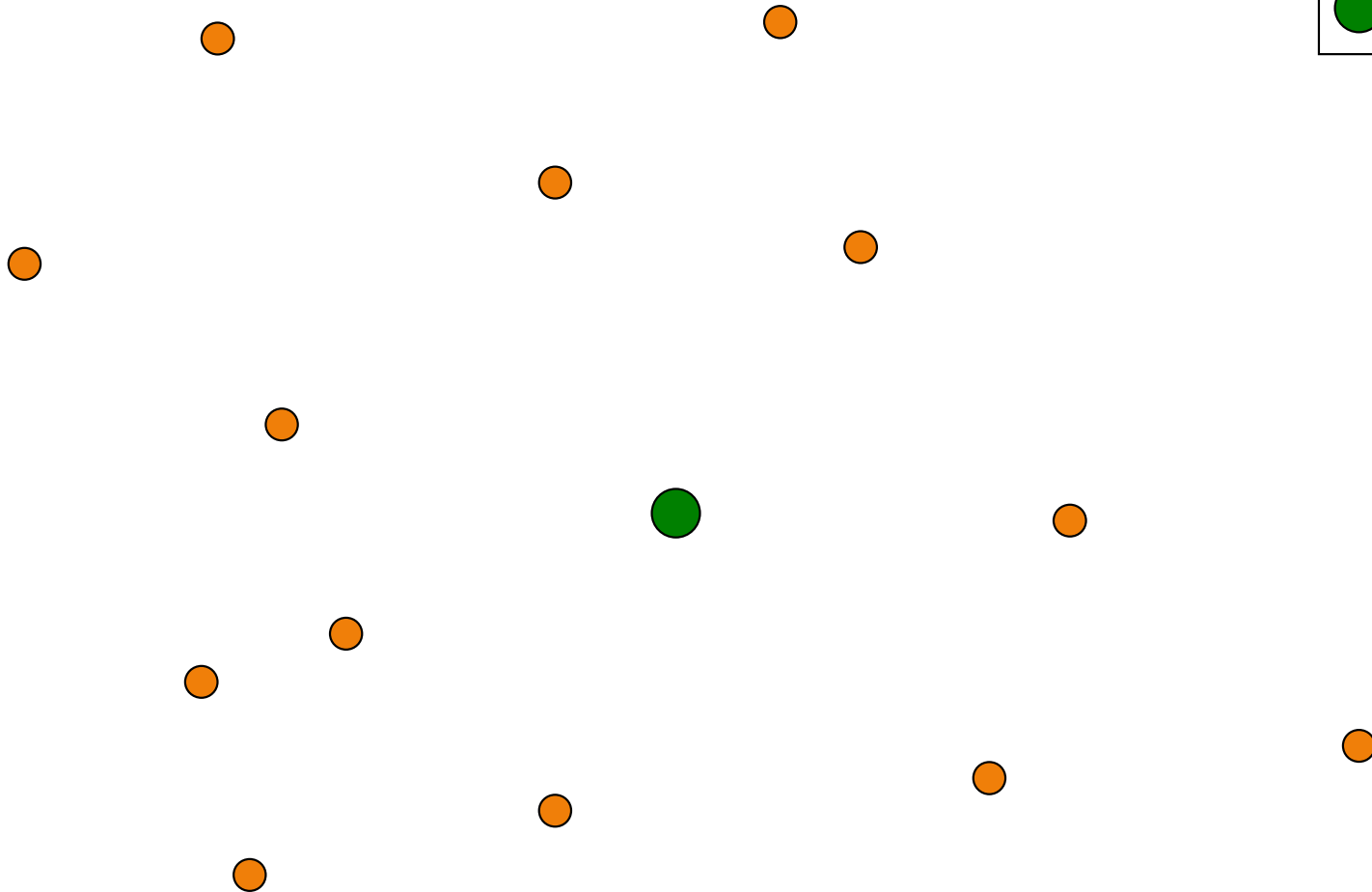
Exemples :

- Logistique : approvisionnements / transport inter-usines / transports de distribution
- Distribution de courrier postal
- Ramassage scolaires
- Lignes de bus
- Ramassage des ordures

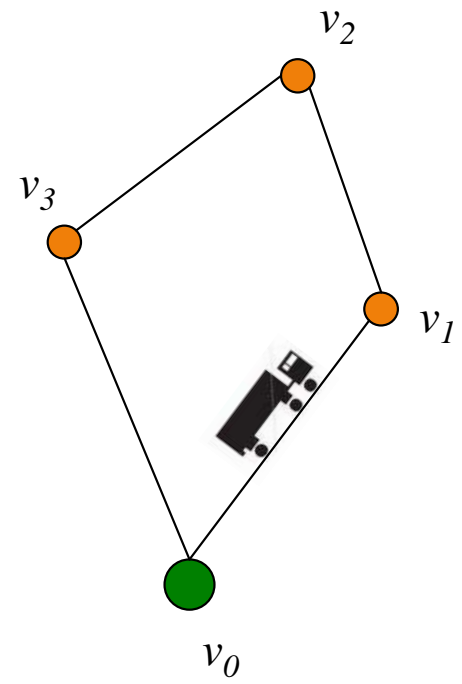
Transport routier, maritime ou aérien

Exemple

- clients
- entrepôt



Exemple



On définit une tournée, c'est-à-dire une route

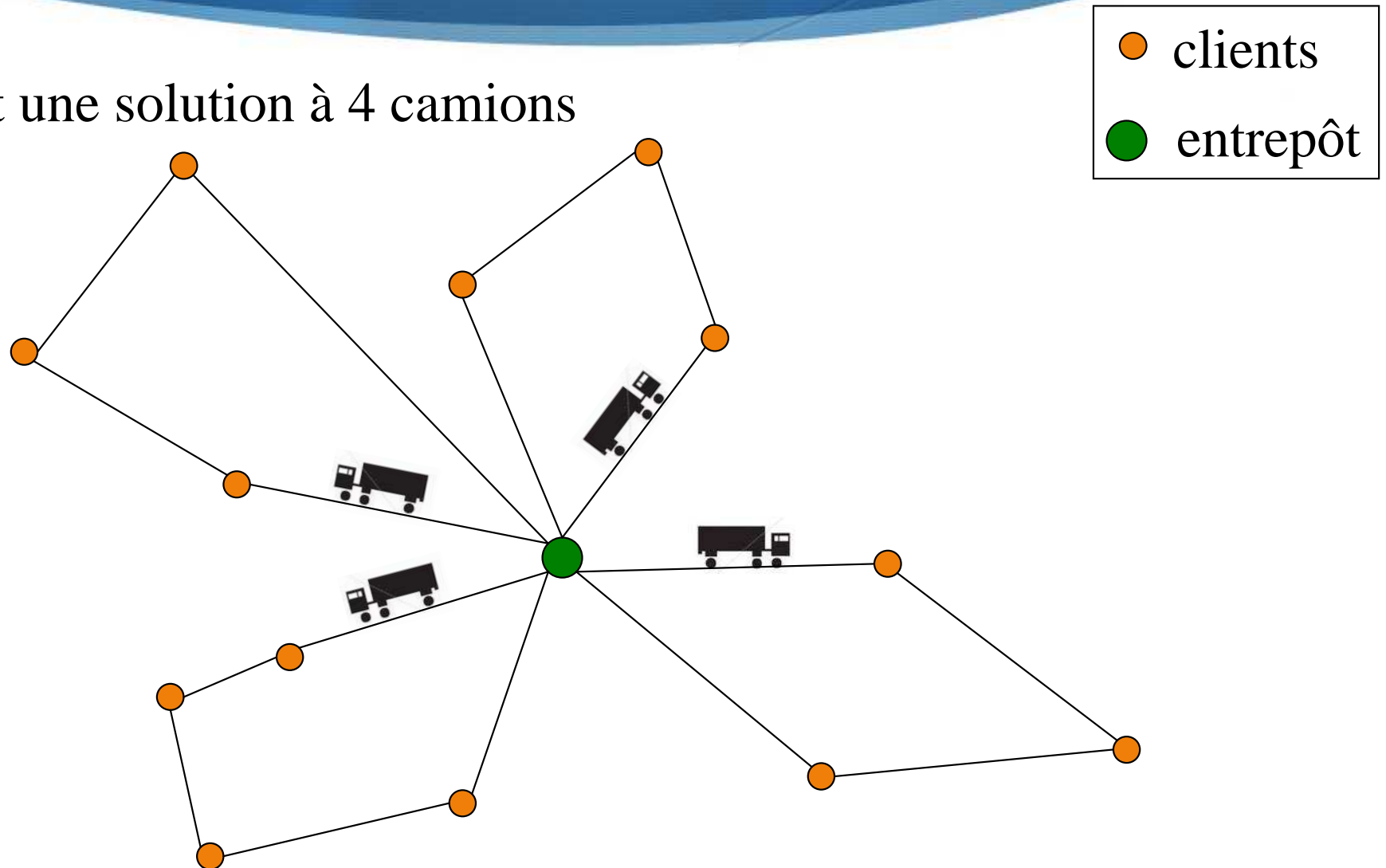
une route R est défini par une suite de clients

$$R = (v_0 ; v_1 ; v_2 ; v_3 ; v_0)$$

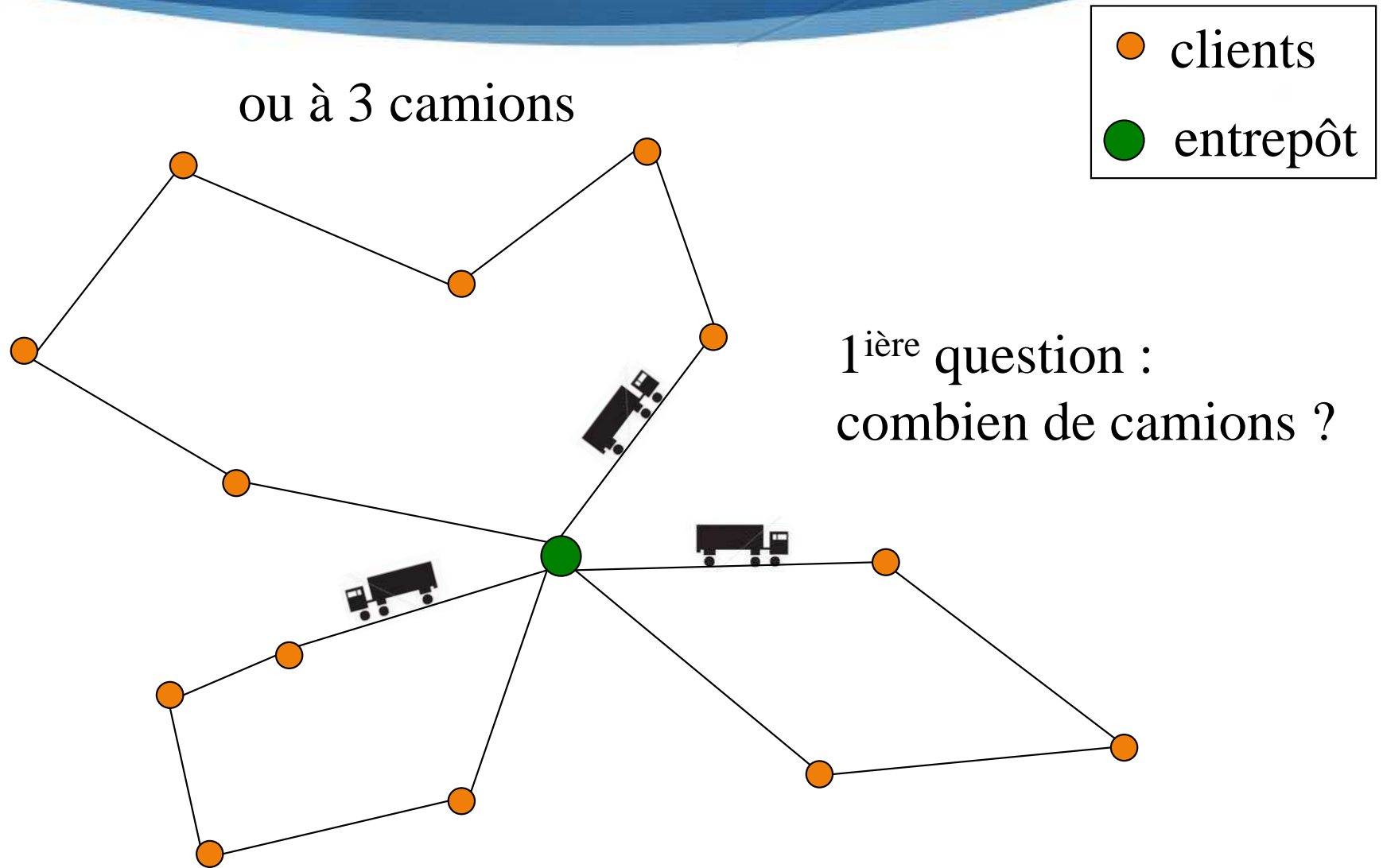
$$\text{longueur}(R) = \sum_{(v_i ; v_j) \in R} c_{ij}$$

Exemple

On définit une solution à 4 camions

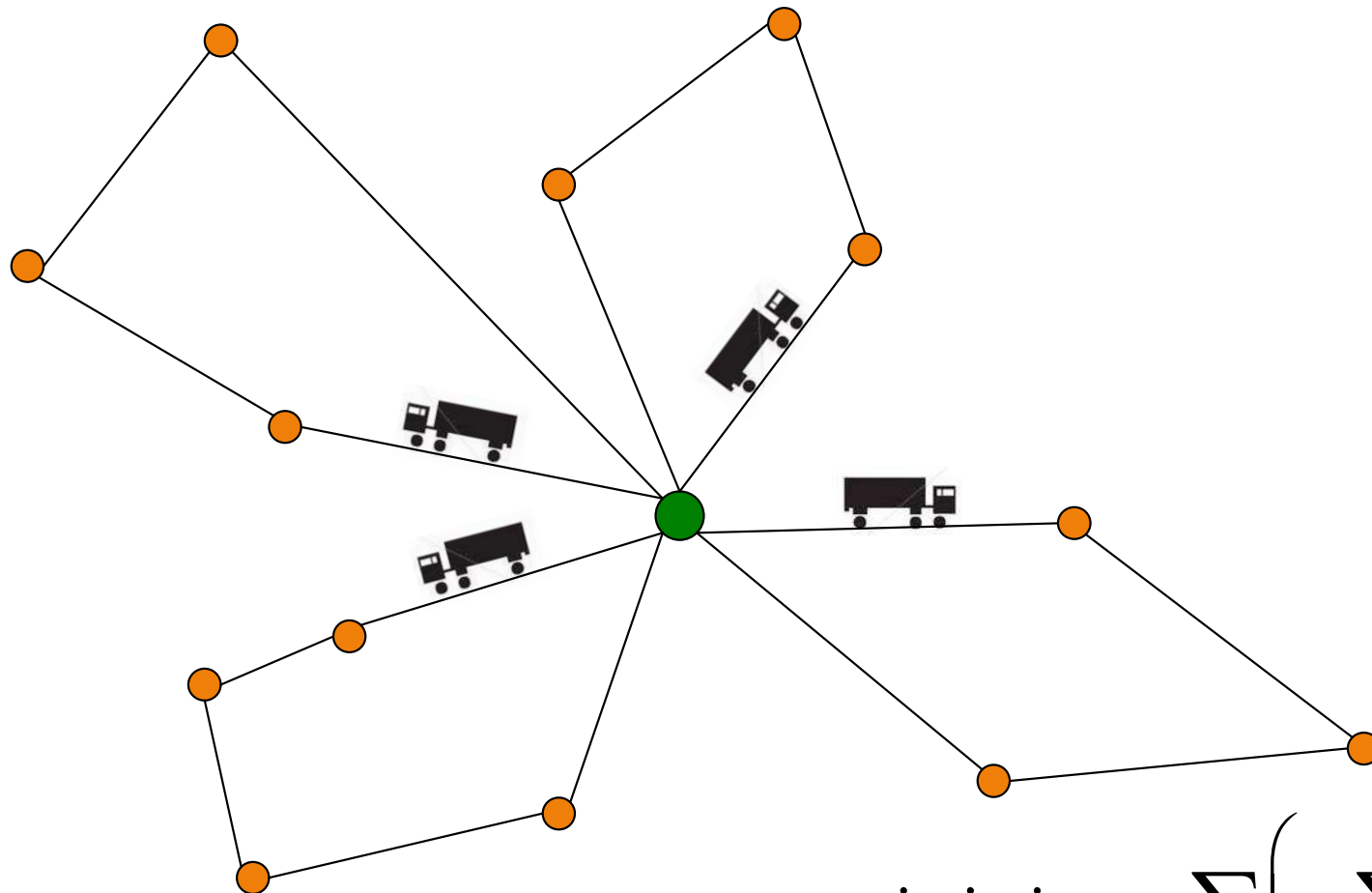


Exemple



Exemple – l'objectif

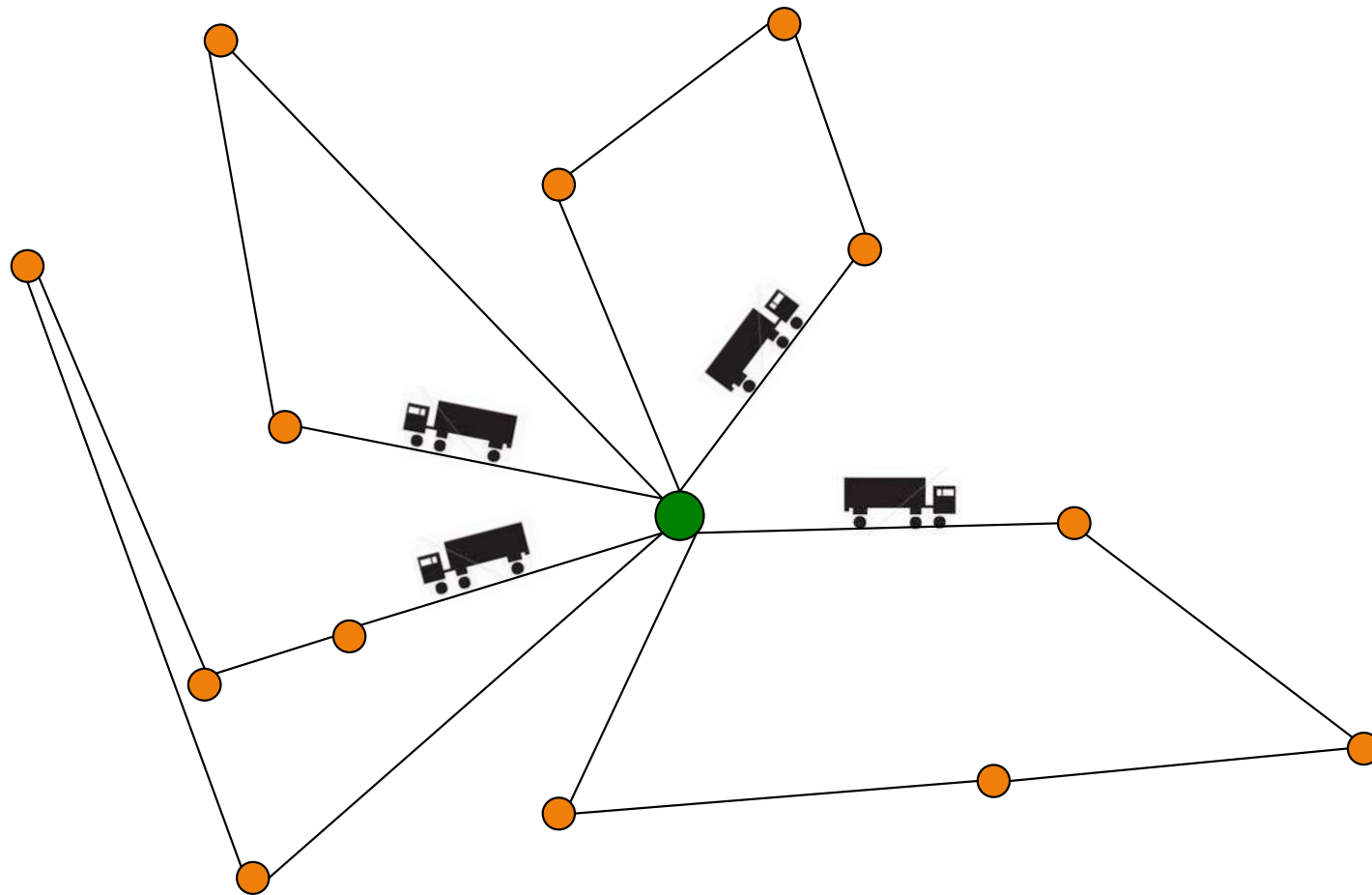
Objectif : trouver une solution qui minimise le coût des trajets ?



C'est-à-dire la somme des arrêtes

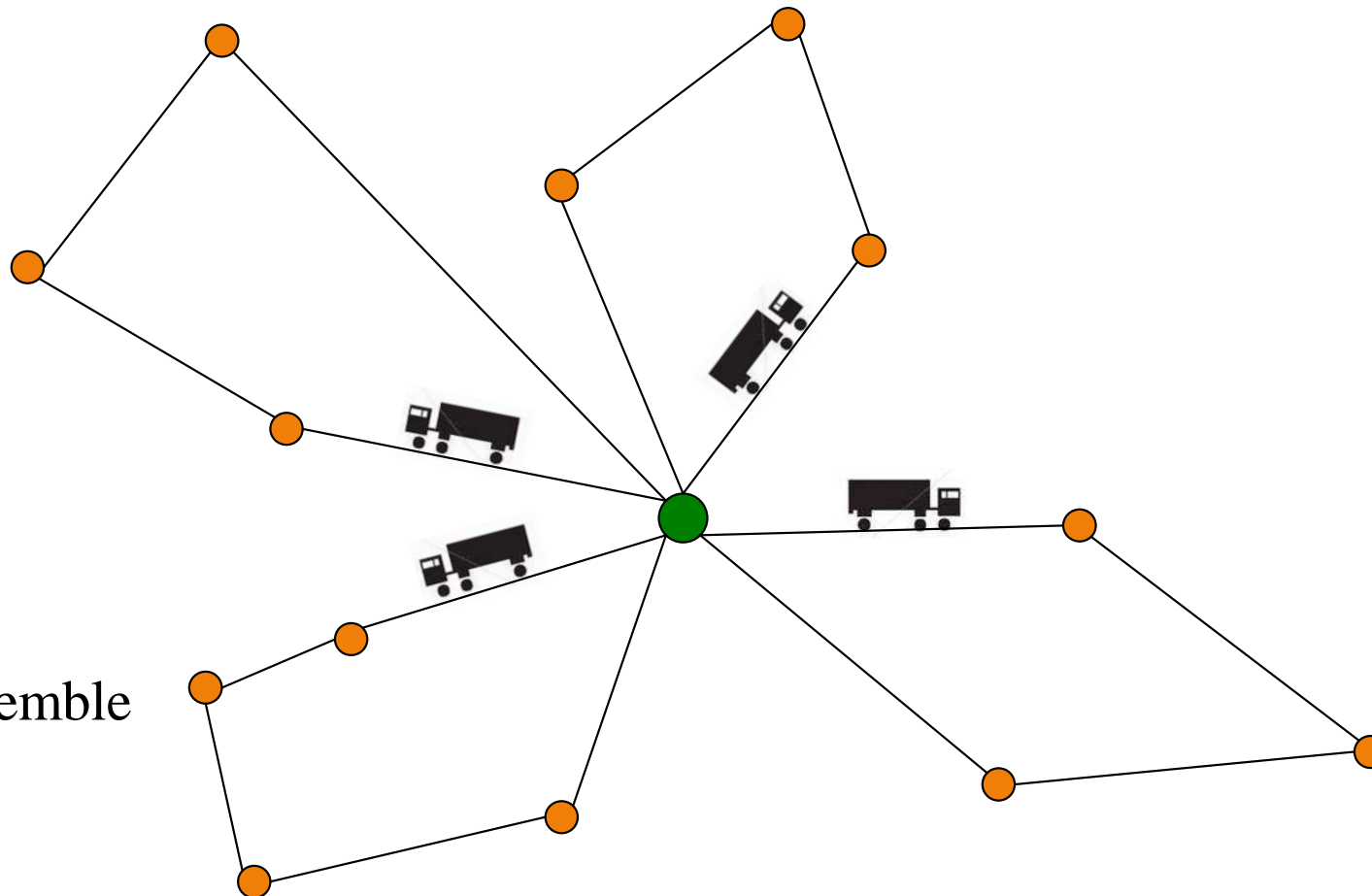
$$\text{minimiser } \sum_k \left(\sum_{(v_i, v_j) \in R_k} c_{ij} \right)$$

Exemple – l'objectif



Exemple – les contraintes

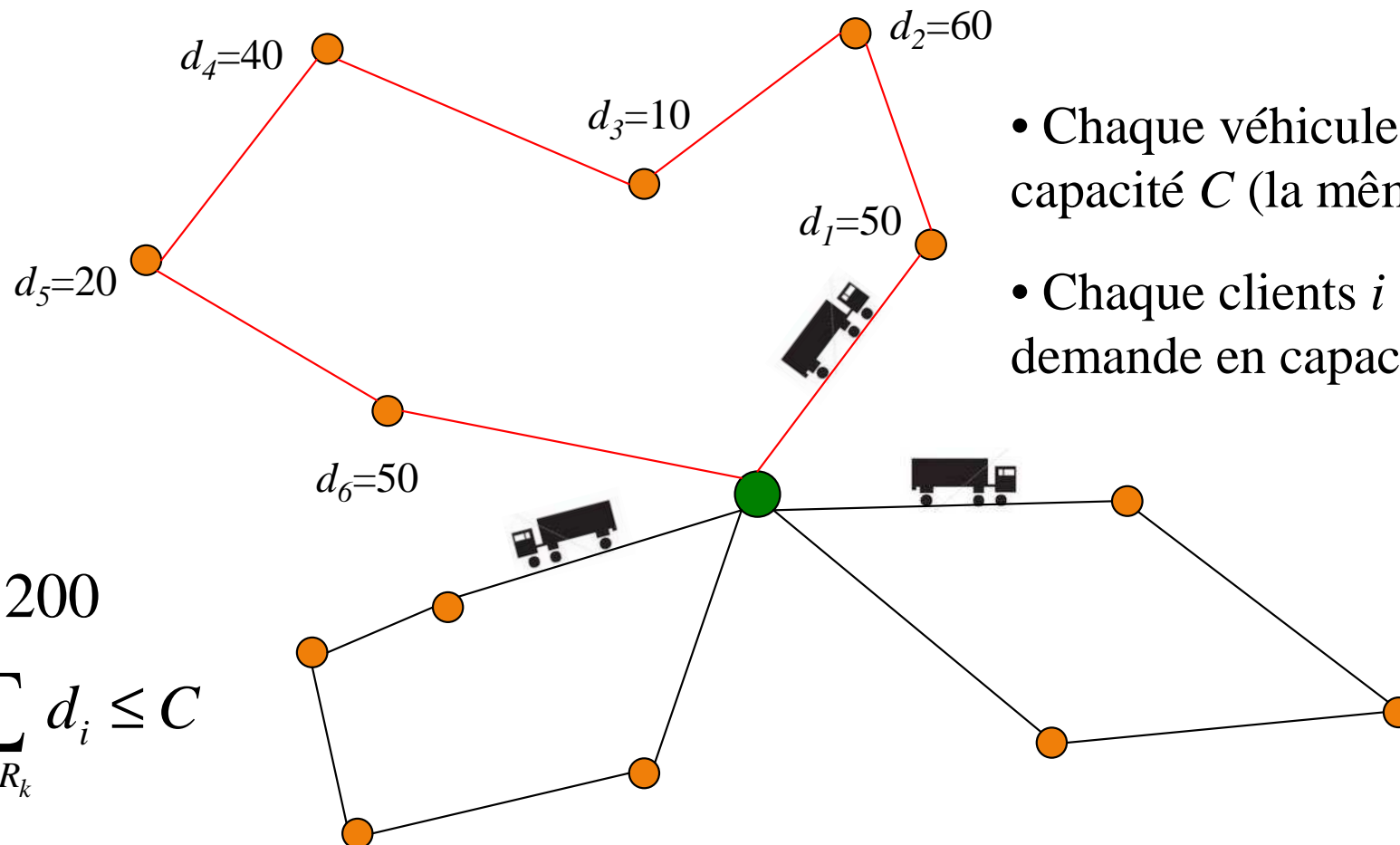
1^{ière} contrainte : Passer par tous les clients



$$\bigcup_k R_k = V$$

V étant l'ensemble
des clients

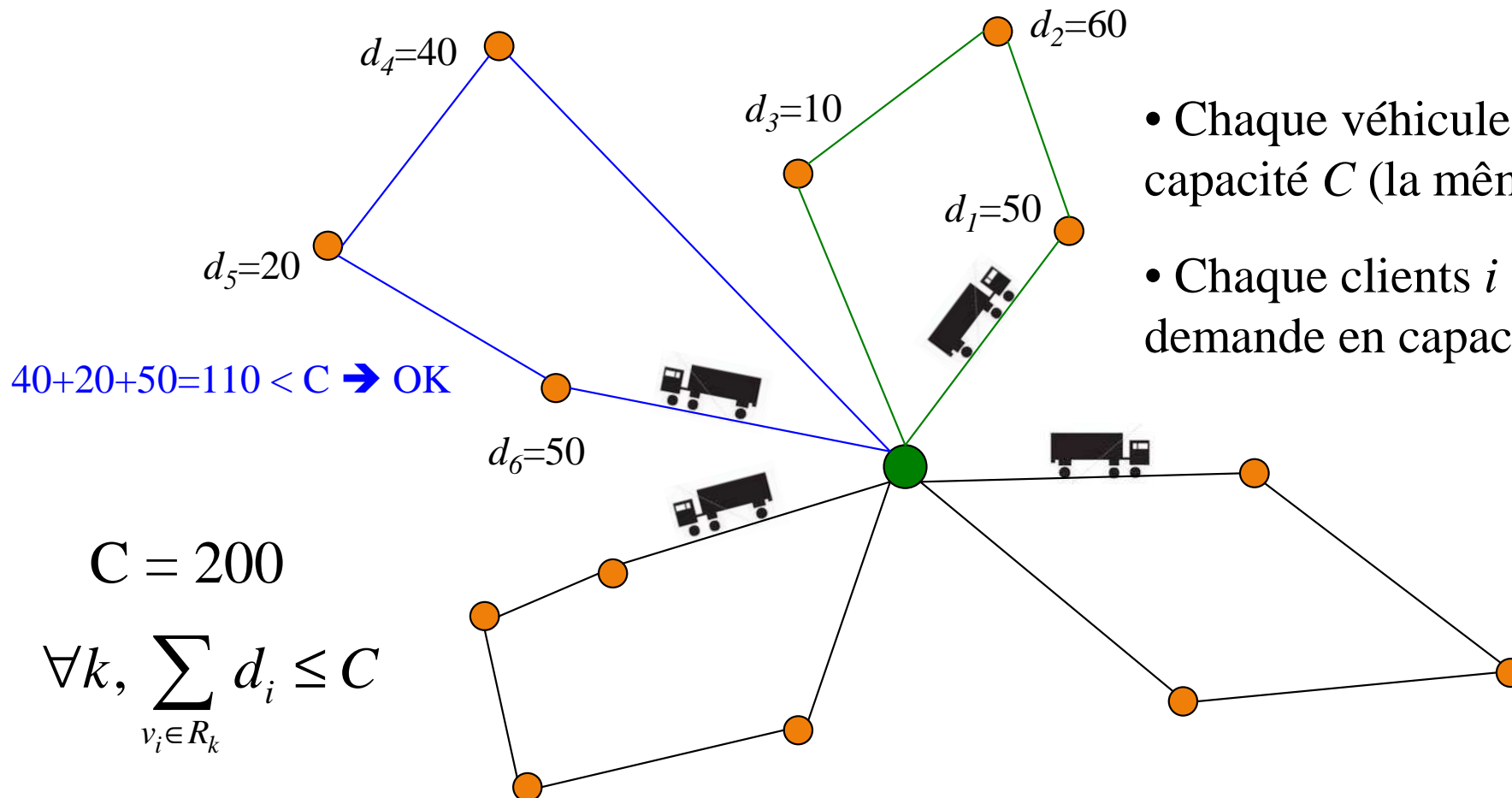
2^{ème} contrainte : Respecter la capacité des véhicules



$50+20+60+10+40+20+50=230 > C \rightarrow$ problème : solution non réalisable

2^{ème} contrainte : Respecter la capacité des véhicules

50+60+10=120 < C → OK



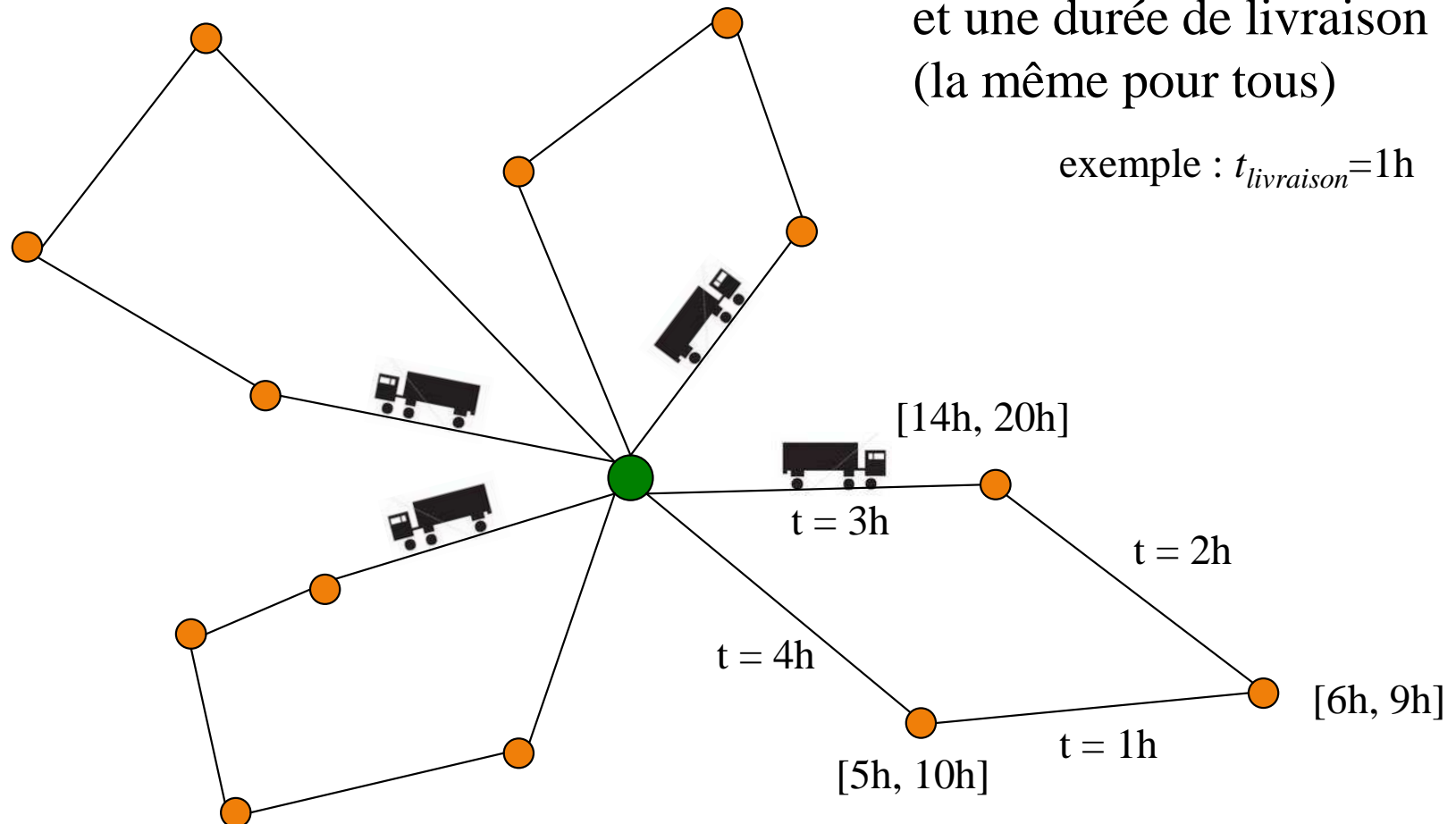
- Chaque véhicules à une capacité C (la même pour tous)
- Chaque clients i à une demande en capacité d_i

3^{ème} contrainte : Respecter les horaires d'ouverture

Chaque clients i à une heure d'ouverture (fenêtre de temps pour la livraison)

et une durée de livraison
(la même pour tous)

exemple : $t_{livraison}=1h$



3^{ème} contrainte : Respecter les horaires d'ouverture

Chaque clients i à une heure d'ouverture (fenêtre de temps pour la livraison)

et une durée de livraison
(la même pour tous)

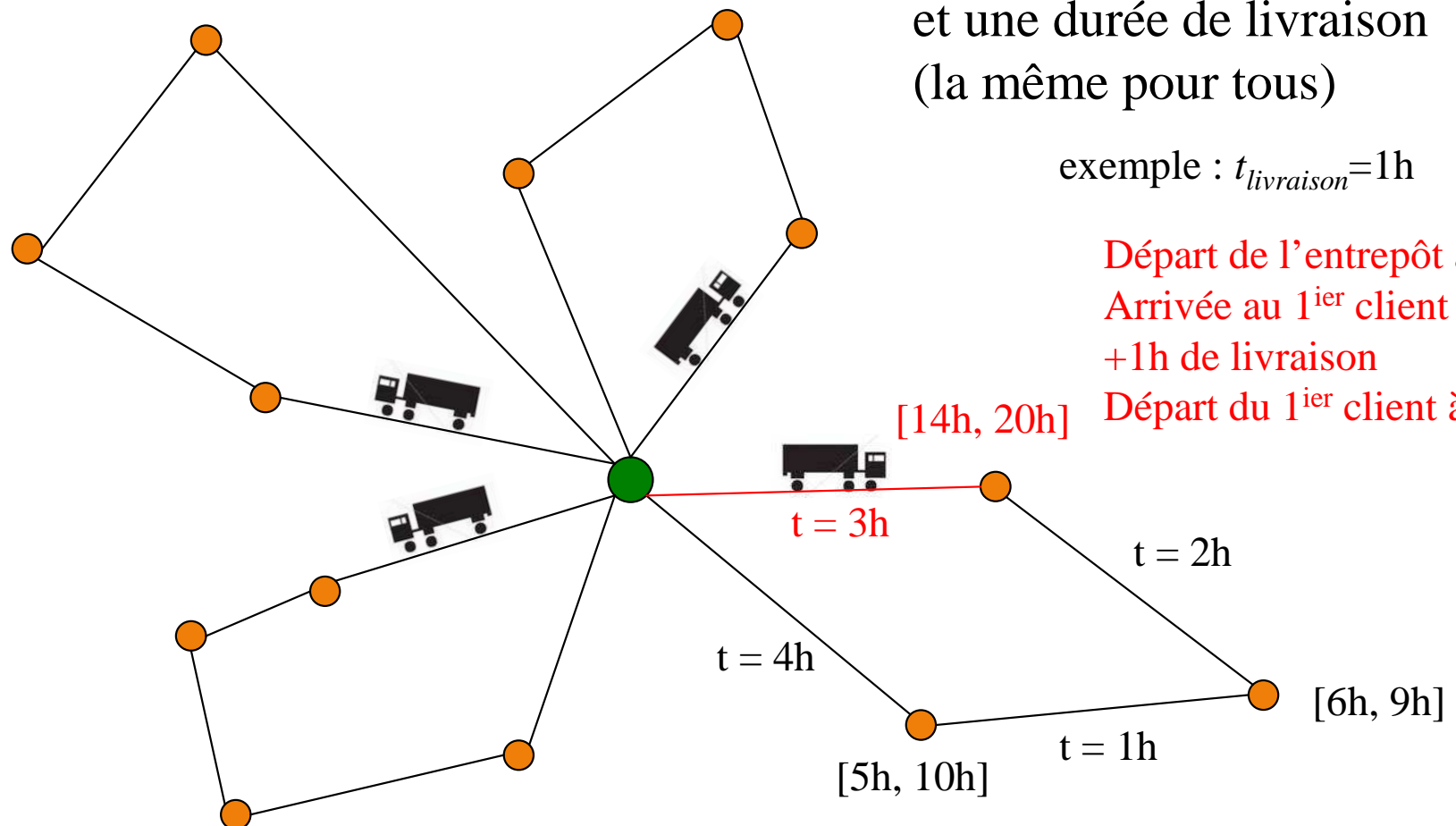
exemple : $t_{livraison}=1h$

Départ de l'entrepôt à 11h

Arrivée au 1^{ier} client à 14h

+1h de livraison

Départ du 1^{ier} client à 15h



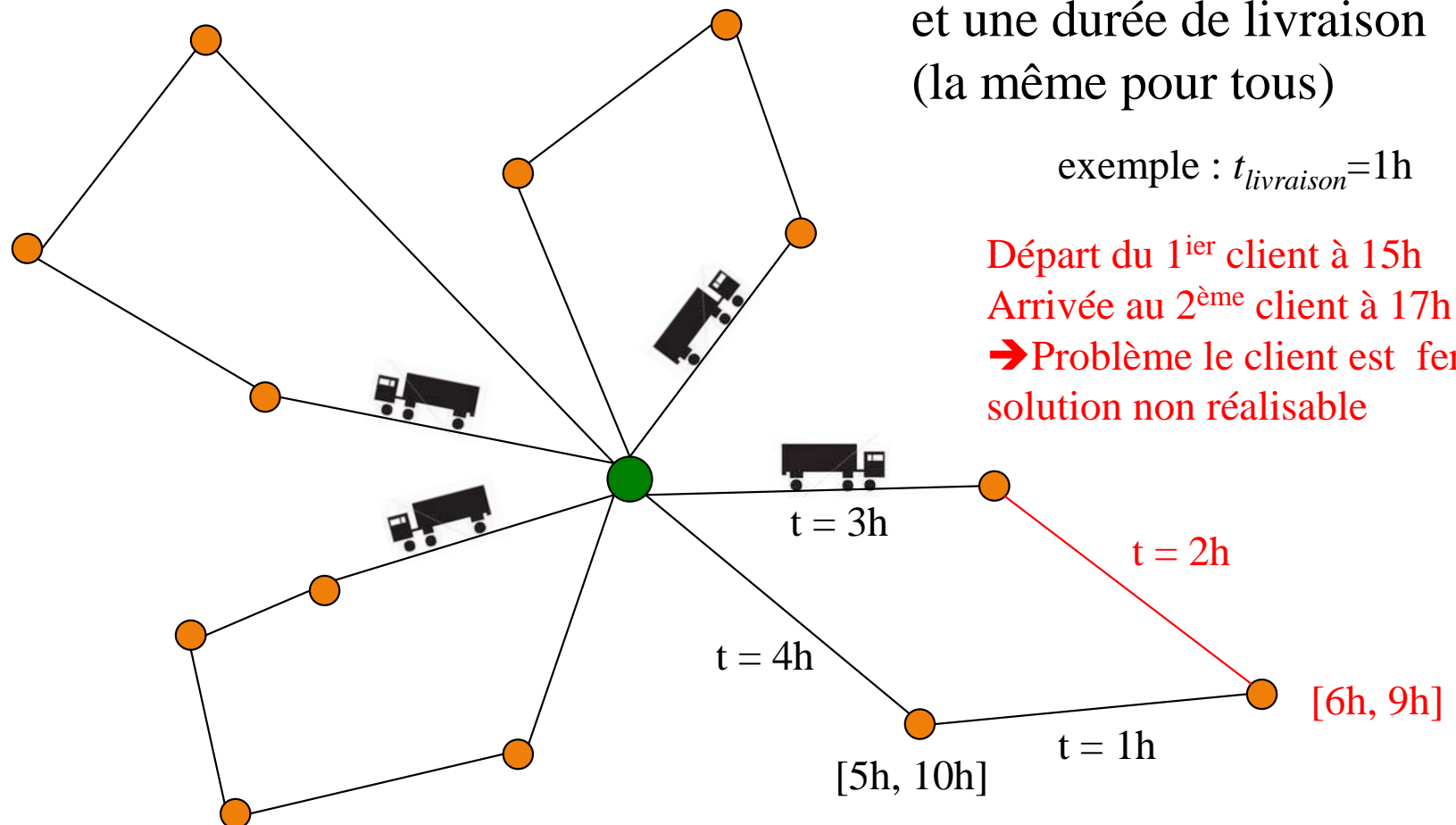
3^{ème} contrainte : Respecter les horaires d'ouverture

Chaque clients i à une heure d'ouverture (fenêtre de temps pour la livraison)

et une durée de livraison
(la même pour tous)

exemple : $t_{livraison}=1h$

Départ du 1^{er} client à 15h
Arrivée au 2^{ème} client à 17h
→ Problème le client est fermé :
solution non réalisable



Problème

Trouver R_k qui minimise $\sum_k \left(\sum_{(v_i; v_j) \in R_k} c_{ij} \right)$

sous les contraintes :

1. $\bigcup_k R_k = V$: passer par tous les clients

2. $\forall k, \sum_{v_i \in R_k} d_i \leq C$: respecter la capacité des véhicules

3. respecter les horaires d'ouvertures : $\forall i, p_i \in [o_i; f_i]$

$$\forall k, \forall (v_i; v_j) \in R_k, p_j \geq p_i + t_{livraison} + c_{ij}$$

Problème

Trouver R_k qui minimise $\sum_k \left(\sum_{(v_i; v_j) \in R_k} c_{ij} \right)$

relâcher la contrainte

sous les contraintes :

1. $\bigcup_k R_k = V$: passer par tous les clients
2. $\forall k, \sum_{v_i \in R_k} d_i \leq C$: respecter la capacité des véhicules

3. respecter les horaires d'ouvertures : $\forall i, p_i \in [o_i; f_i]$
 $\forall k, \forall (v_i; v_j) \in R_k, p_j \geq p_i + t_{livraison} + c_{ij}$

Problème

Trouver R_k qui minimise $\sum_k \left(\sum_{(v_i; v_j) \in R_k} c_{ij} \right) + \text{pénalité} \times \sum_i \max(0; p_i - f_i)$

sous les contraintes :

1. $\bigcup_k R_k = V$: passer par tous les clients
2. $\forall k, \sum_{v_i \in R_k} d_i \leq C$: respecter la capacité des véhicules

$$\forall k, \forall (v_i; v_j) \in R_k, \quad p_j = \max(o_i; p_i + t_{livraison} + c_{ij})$$

Règles

- Equipe de 3 étudiants
- Utiliser les heures de TP
- Avant implémentation : proposition technique (structures de données, méthode d'optimisation,...)
- Echanger les résultats des scénarios intermédiaires
- Langages de programmation : Java ou C++
- Fournir un fichier exécutable Windows ou Linux (Java ou C++)
`optim.exe probleme-ii.txt solution.txt temps`
- Evaluation sur la base de trois scénarios finaux (connus qu'une semaine avant l'évaluation).
- Pas de multiprocesseurs
- Littérature
- Prix décerné à la meilleure équipe

Planning : 6 phases

1. Aujourd'hui : Description (dans la semaine description détaillée)
2. Fin avril : Jeux de travail et formation des groupes
valide les formats d'entrée
3. Début mai : Evalueur
valide les formats de sortie
4. Début juin : Présentation méthode utilisée
10min + 5min, 13pts
5. Mi-juin : Jeux de test d'évaluation
6. 1 semaine plus tard : Envoie des exécutables + évaluation
7pts

alexandre.gondran@utbm.fr

Laboratoire SeT, bâtiment D206

03 84 58 38 14

Moodle de l'UTBM :

<https://moodle.utbm.fr/> → AG41

<http://alexandre.gondran.free.fr> → Teaching

Mohammad Dib et Jun Hu

Merci de votre attention