Projet de CoCoMA

Mathieu Berthellemy, Élodie Cayez
& Marie-Morgane Paumard

1.
Introduction

Tenants et aboutissants du sujet

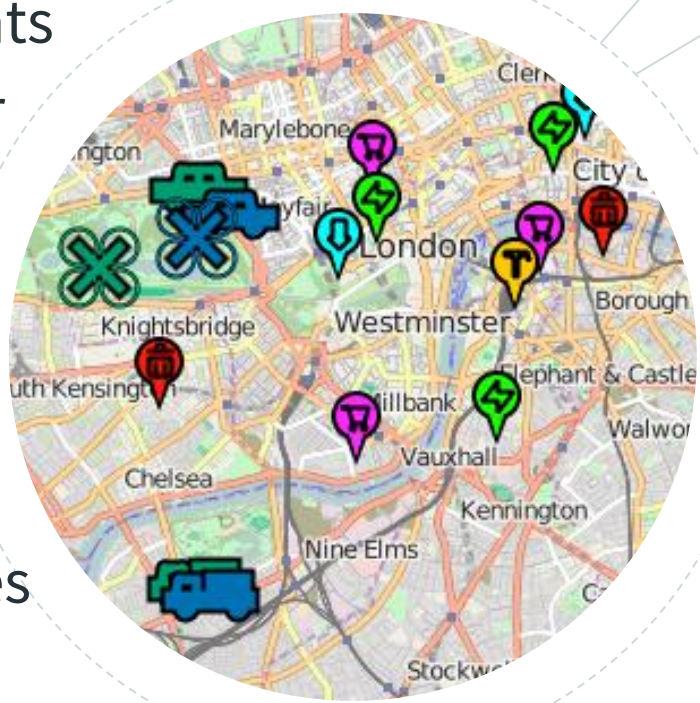
Aperçu du projet

Consignes du projet multi-agents

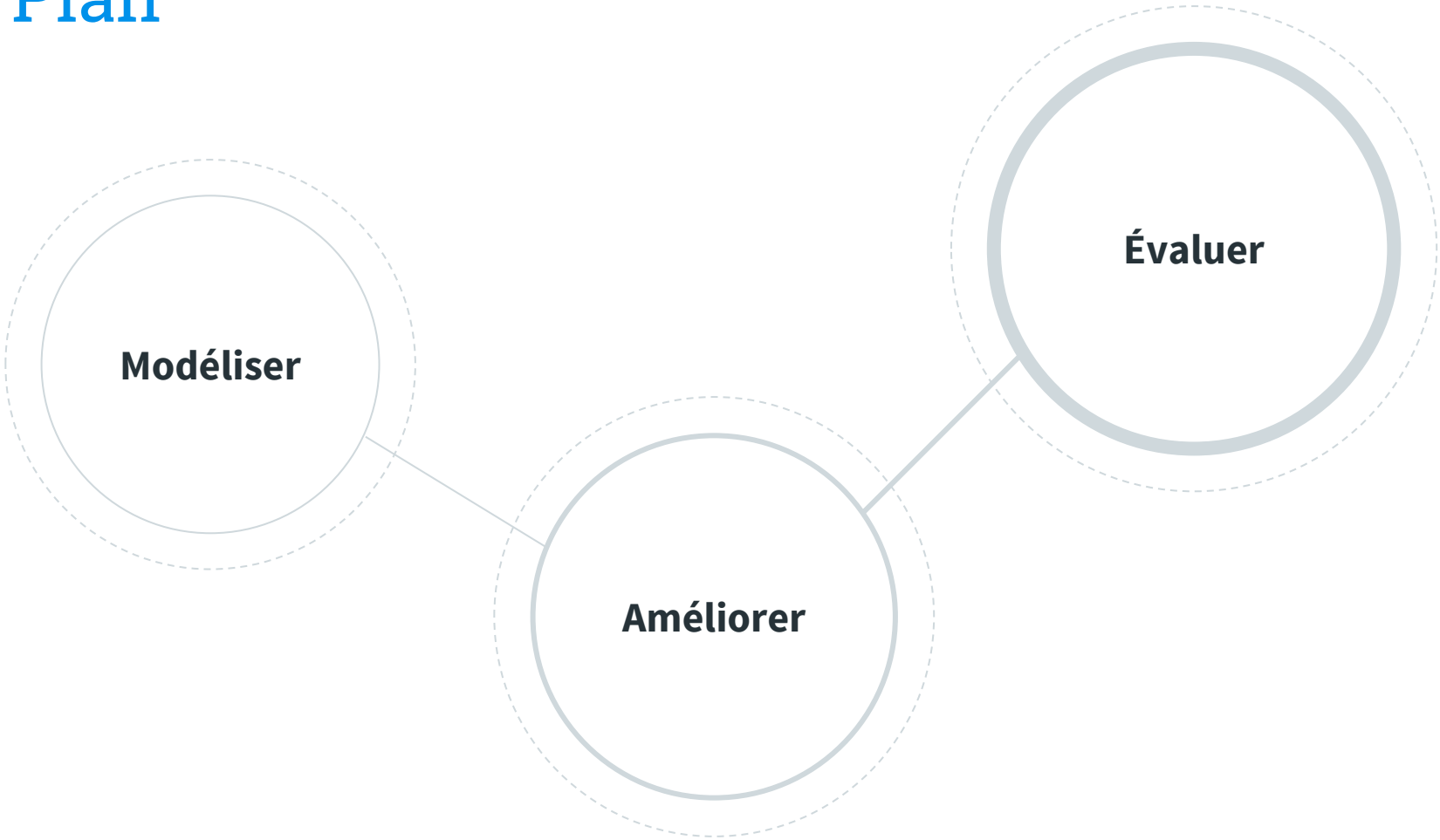
- Équipes de **4** agents différents
- Acquérir – Fabriquer – Livrer
- Maximiser les recettes

Problèmes principaux

- **Accepter** une tâche
- **Coordination** des agents
- **Distribution** des sous-tâches



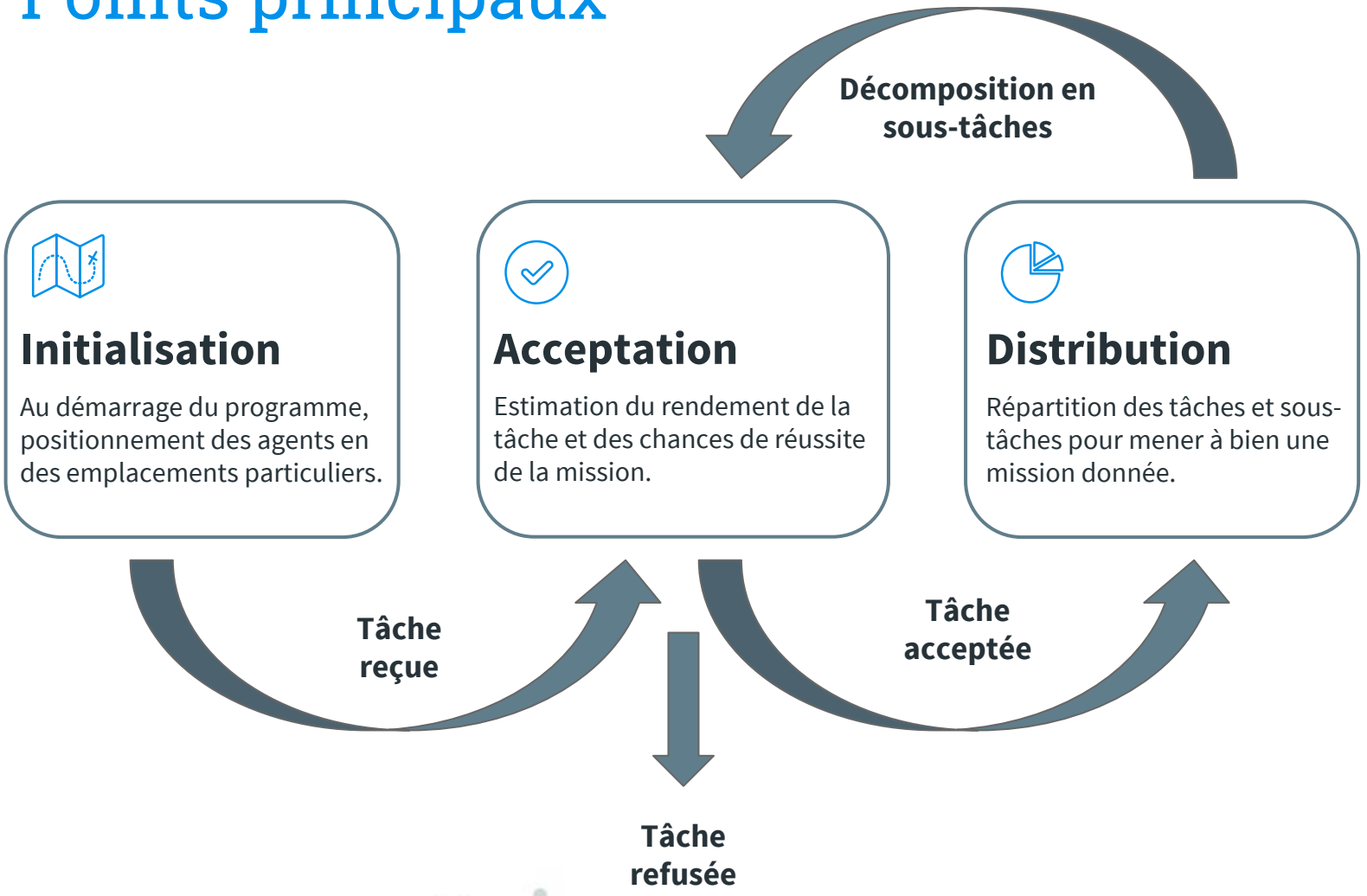
Plan



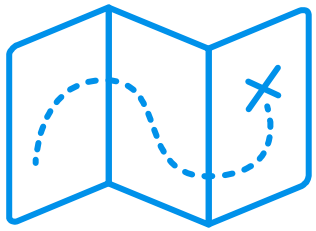
2.
Modèle de base

Coordination, acceptation et distribution des tâches

Points principaux



Intérêt de l'initialisation



Seules sont connues les positions lorsque les agents sont à des endroits spécifiques.

Méthodes d'initialisation

Arbitraire

Les agents sont envoyés vers un emplacement choisi arbitrairement.

Méthode choisie

- Introduit une diversité des positions.
- Expérimentalement plus satisfaisante.

Déterminée

Les agents sont tous envoyés vers un emplacement précis (i.e. au centre de la carte).

Communication

Déplacement arbitraire jusqu'à destination

Envoi d'un message aux *n-1* autres agents indiquant la position

Attente des *n-1* messages réponses

Acceptation d' une tâche

Recettes ≥ Coûts

Estimation des coûts

- Livraison : T
- Achat : $T+\lambda \times p \times n$
- Fabrication : $T+\lambda \times c$
- Recette : $\lambda \times r$

λ est le ratio entre les frais monétaires et le temps moyen de parcours T.

Rendement

On note *R* le rendement (recette - coûts de livraison, d'achat et/ou de fabrication).

Une tâche est acceptée si et seulement si son rendement est strictement positif.

Valeur de λ

$$\lambda = 10$$

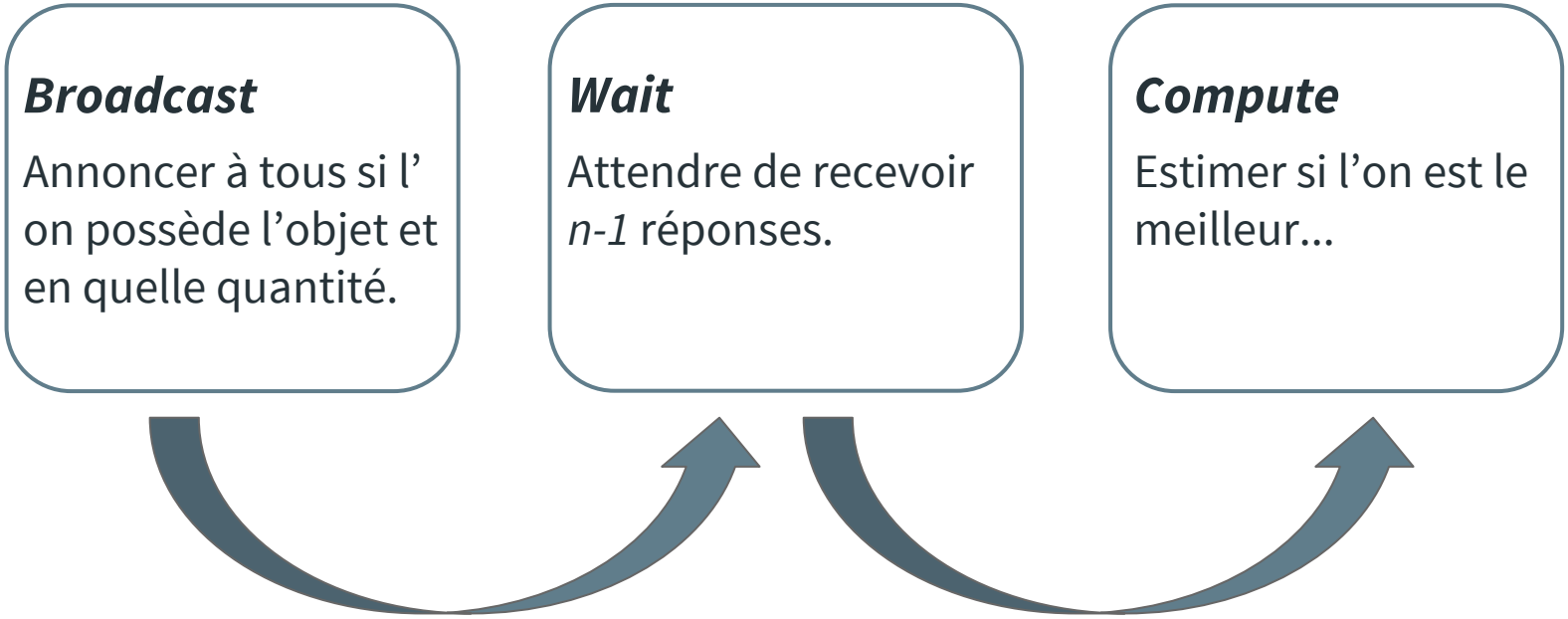
λ est borné :

- Si trop faible, refus systématique ;
- Si trop élevé, acceptation déraisonnable.

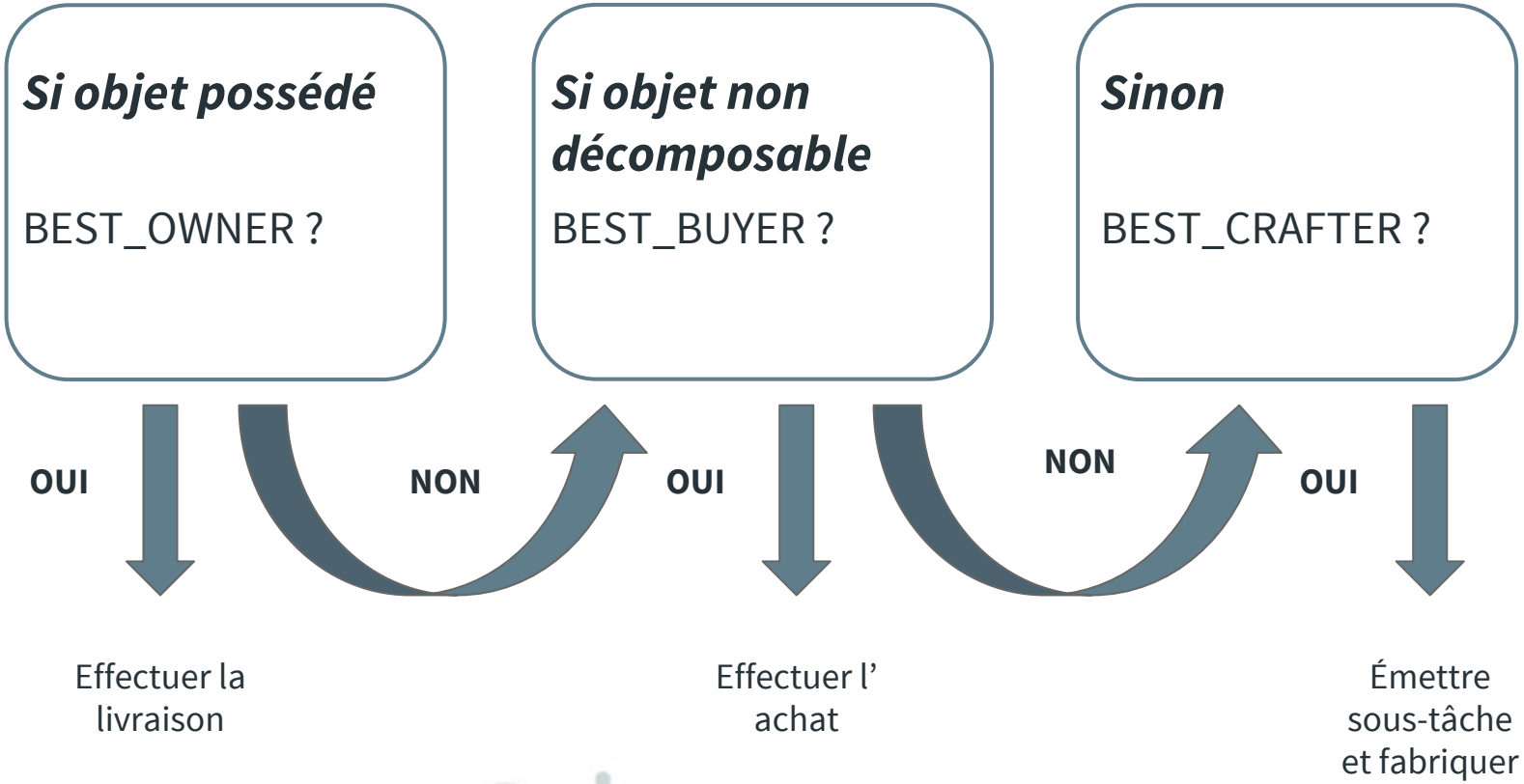
Distribution d'une tâche

Comment désigner le meilleur agent possible ?

Points principaux

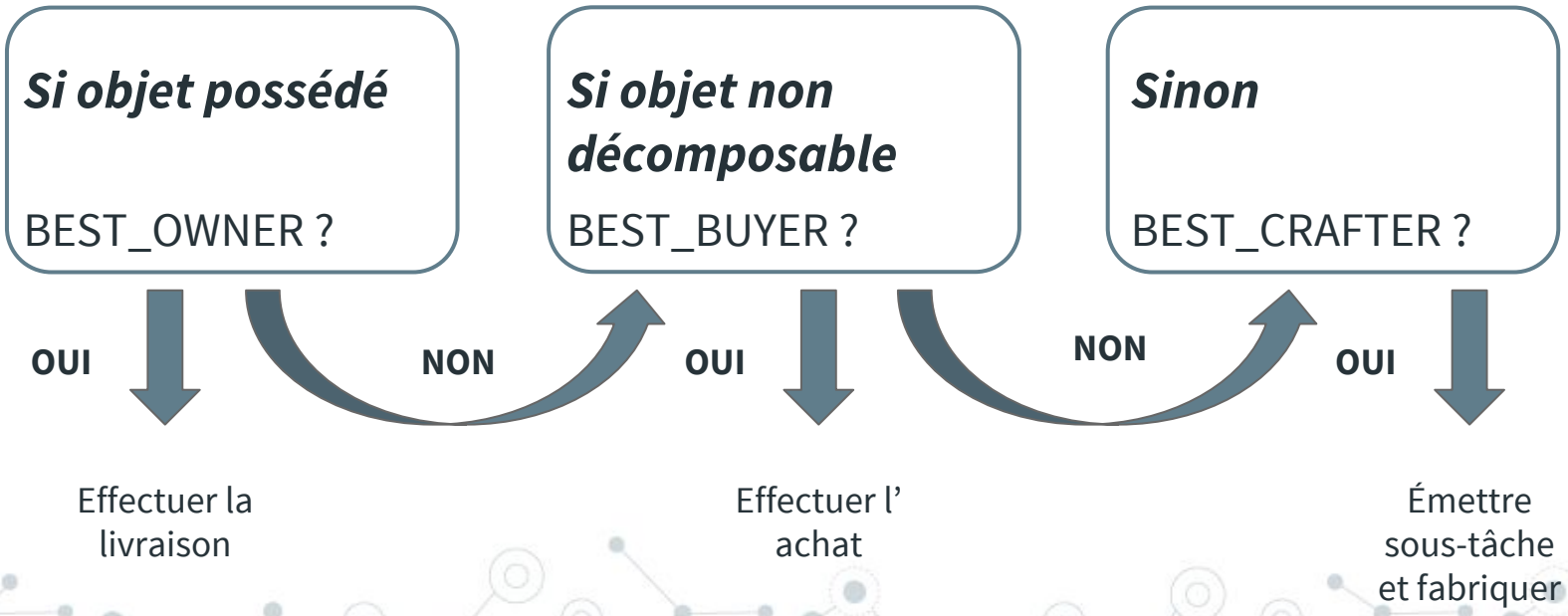


Fonctions de coût



Fonctions de coût : remarques

- Unicité du meilleur agent
- Résistance aux pannes
- Terminaison



Fonction BEST_OWNER

Pour chaque agent disposant de suffisamment d'objets, on calcule BEST_OWNER selon les termes :

- Distance au lieu de livraison.

Fonction BEST_BUYER

On calcule BEST_BUYER selon les termes :

- Distance aux magasins ;
- Distance magasin-livraison.

Fonction BEST_CRAFTER

Pour chaque agent disposant des compétences nécessaires, on calcule BEST_CRAFTER selon les termes :

- Distance aux ateliers ;
- Distance atelier-livraison.

3.
Améliorations

Détails et améliorations
de notre algorithme

Sûreté

Des agents et des canaux de communication

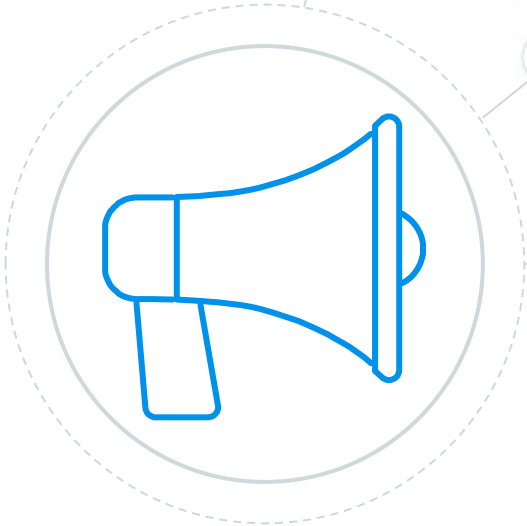
FIFO

First In First Out

Asynchronisme

Des canaux de communication

Améliorer les échanges



- Topologie en anneaux
  - Système de jetons

Ressources & engagement

Indisponibilité des ressources réservées

Gestion des tâches et priorités

- Refus des tâches secondaires lors de l’initialisation ;
- Tâches effectuées par ordre de réception.

4.
Évaluation des performances

Nombre cool

Meilleur résultat obtenu sur la tâche

Analyse, remarques

5.

Conclusion & Perspectives

Résumé et ouverture

Réalisations

Algorithme décentralisé permettant de résoudre les tâches proposées via :

- Gestion des communications
- Génération de plans
- Prévention des inter-blocages

Perspectives : contraintes

- Capacité
- Batterie
- Assistance

Perspectives : améliorations

- Fonctions de coût
- Gestion des tâches
- Synchronisation

6. Démonstration

Merci !

Avez-vous des questions ?