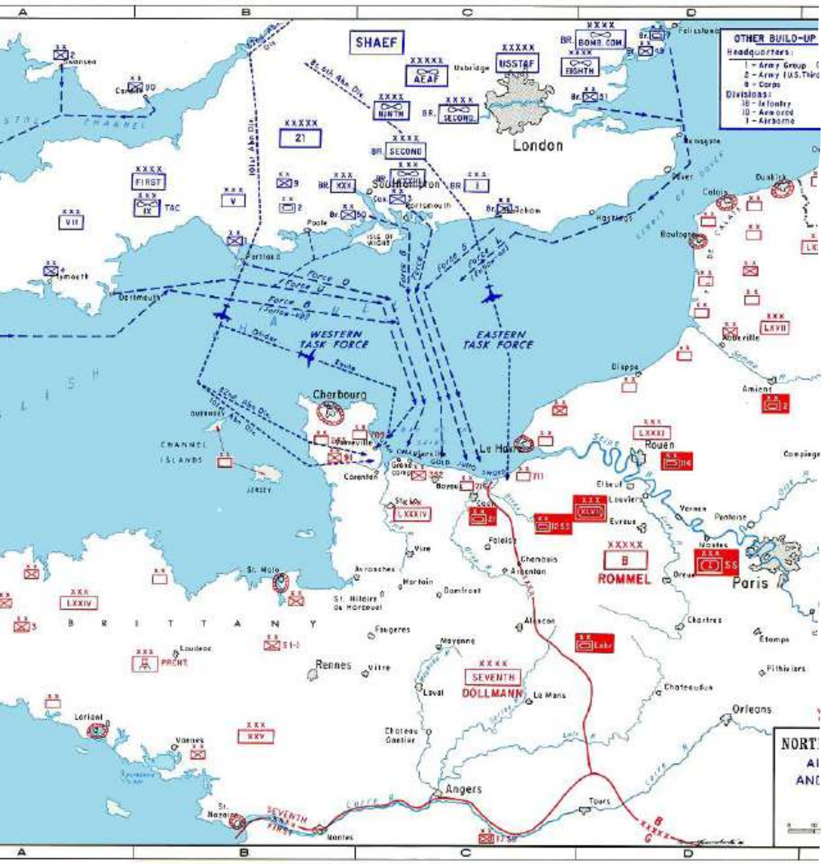
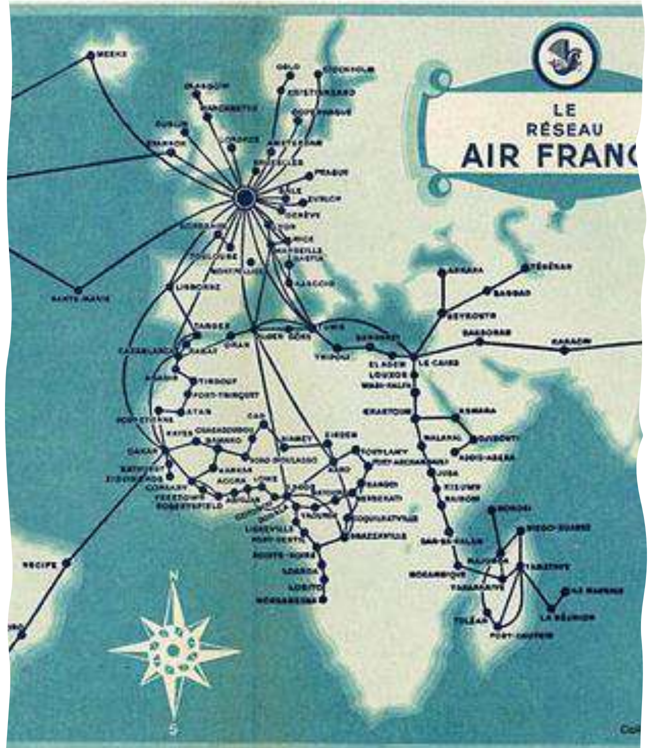
Décarboner et optimiser les processus industriels : enjeux scientifiques en recherche opérationnelle

Jeudi 3 octobre 2024 Ecole des Ponts INS2I

Axel Parmentier





Recherche Opérationnelle

Allocation optimale des ressources

De la planification du débarquement allié

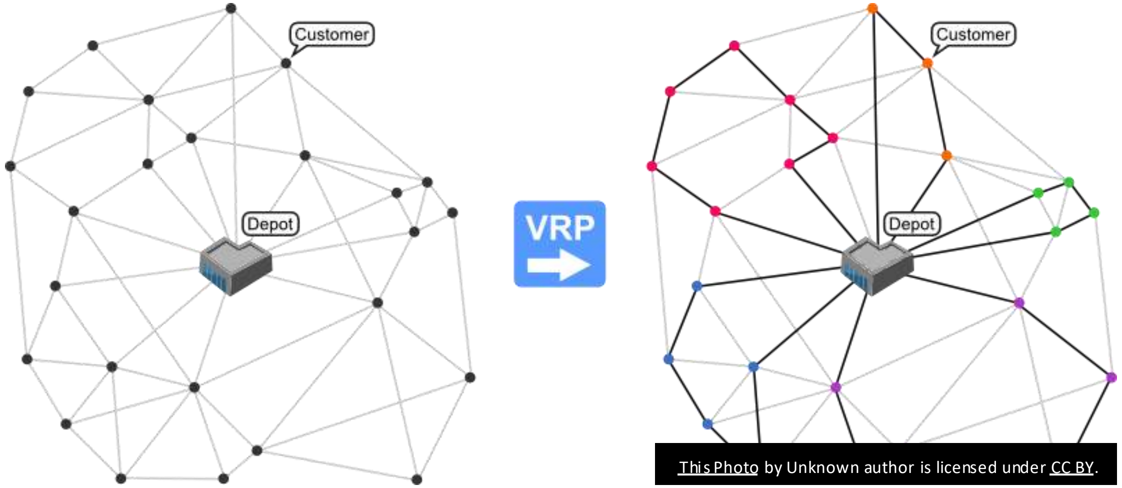
À l'optimisation des industries de réseaux (1960+)

À toutes les industries (données, numérisation)

Problèmes de tournées

- Transports

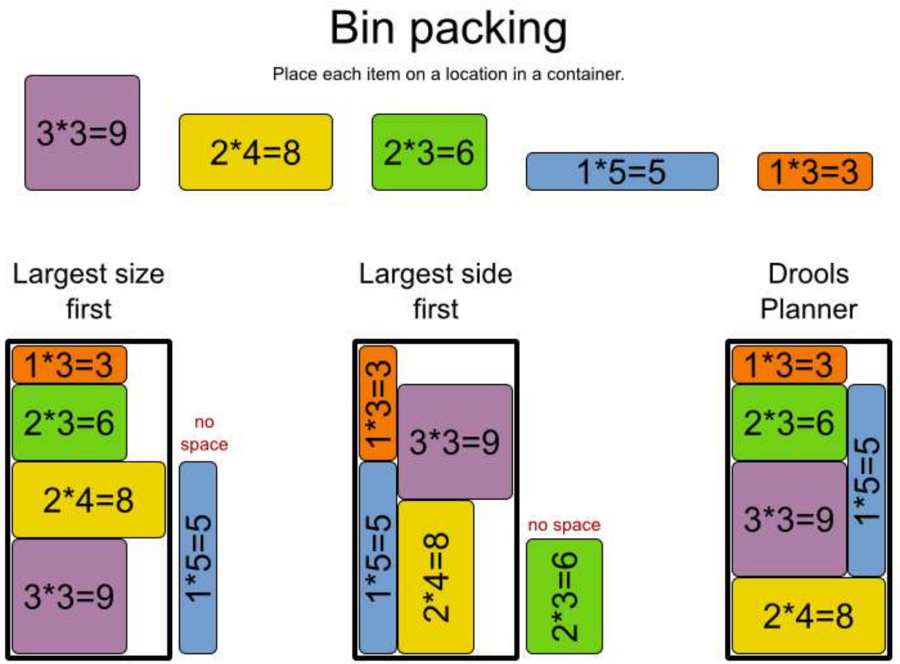
- Supply chain
  - Livraison
  - Gestion d'entrepôts
  - Transports de marchandises



Remplissage de conteneurs

Camions, bateaux, trains, porte-conteneurs

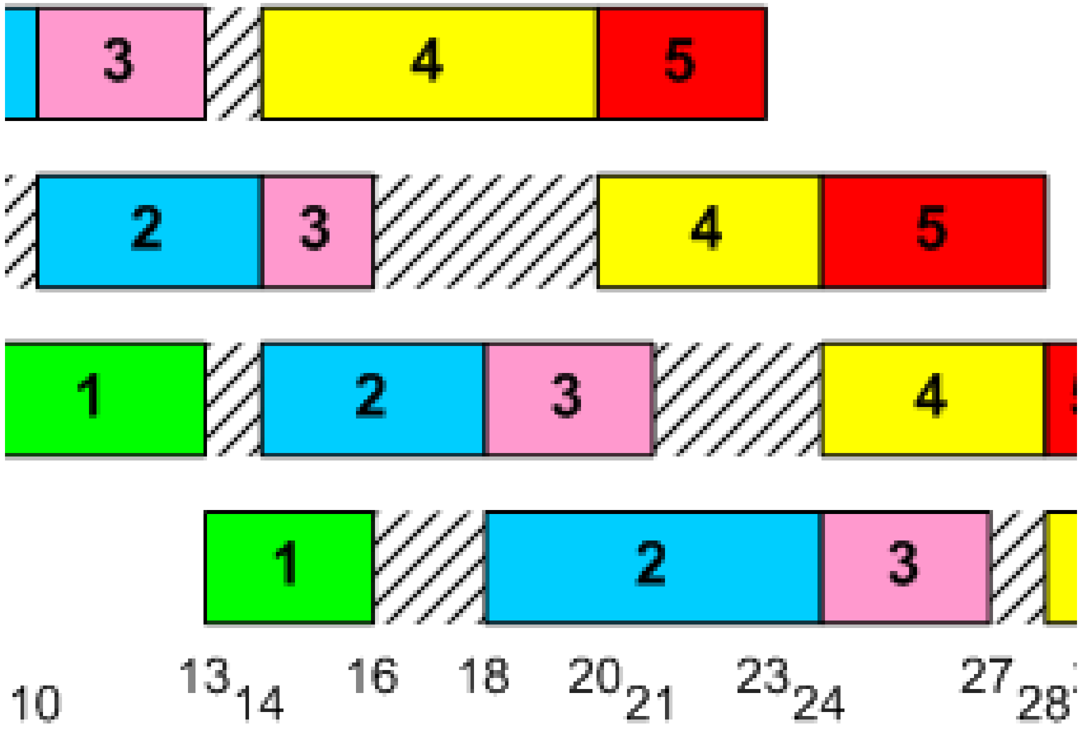
Découpes (pare-brises, patrons, etc.)



Ordonnancement

Planification de tâches

- Atelier
- Processeur
- Chantier

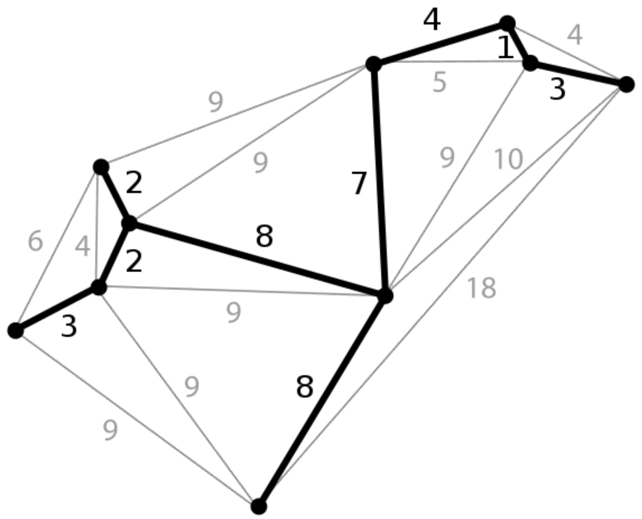


Optimisation sur les réseaux

Eau, électricité, transport, communication

Gestion optimale du réseau

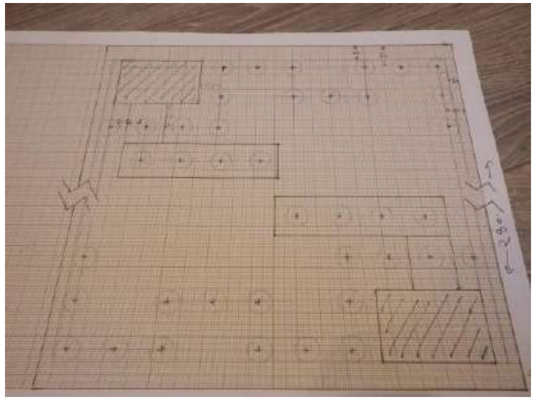
Design du réseau

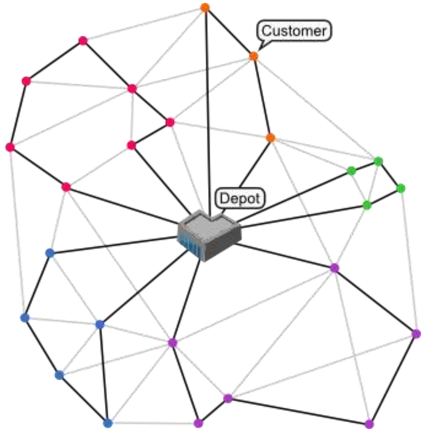


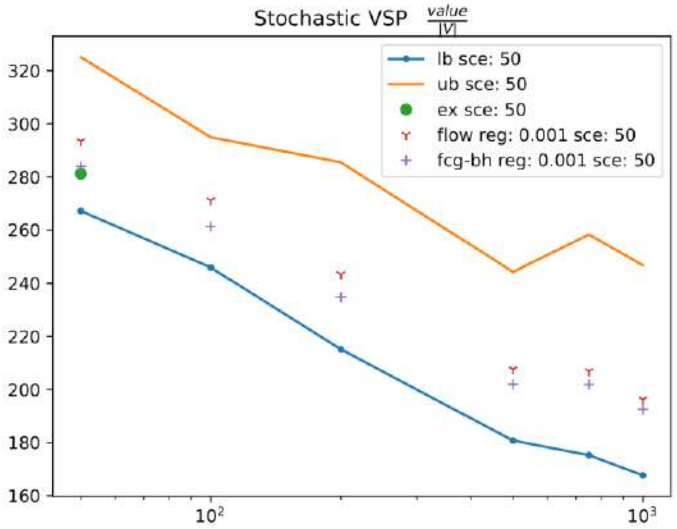
Les clefs du succès

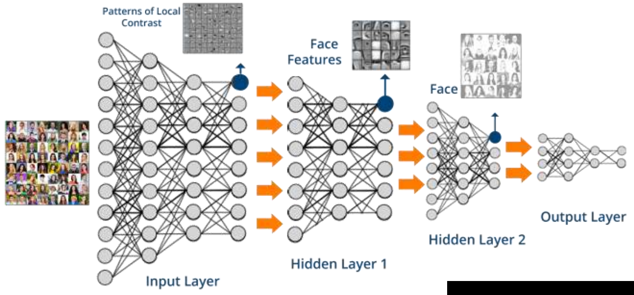
1. Modélisation du bon process
2. Passage à l'échelle de l'optimisation combinatoire
3. Approches tirées par les données

Couts marginaux décroissants



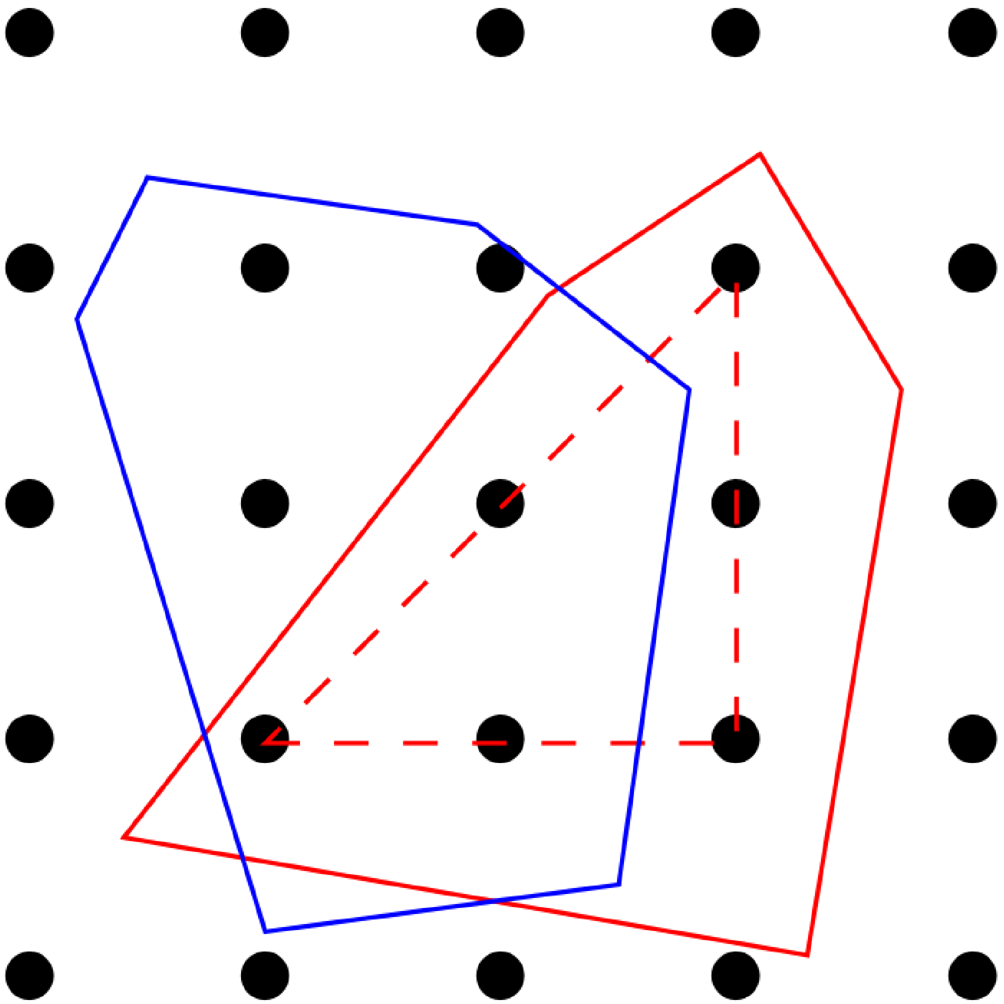






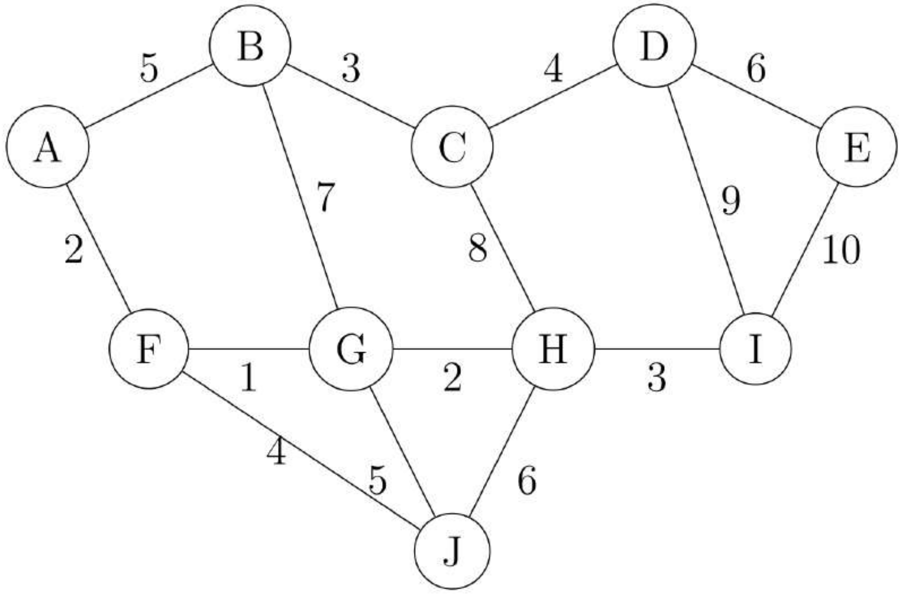
Optimisation combinatoire

Outils mathématiques



Optimisation combinatoire et graphes

$$\min_{x \in \mathcal{X}} f(x)$$



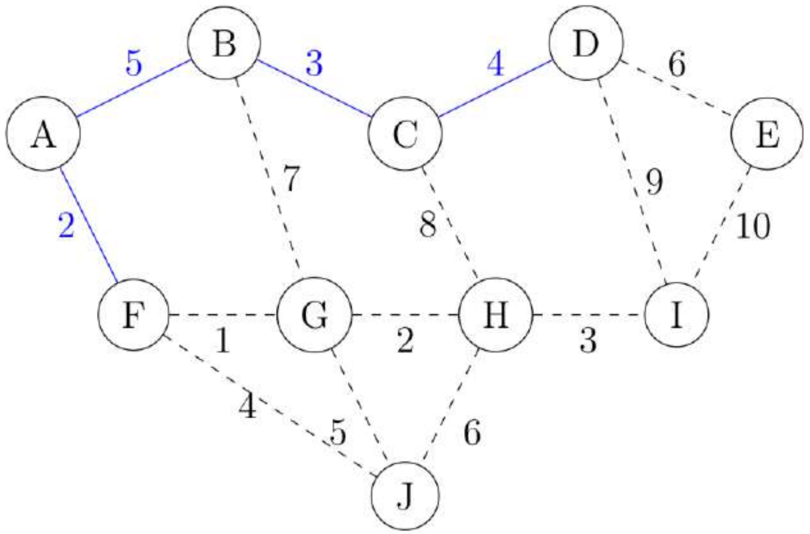
Variables *x* entières

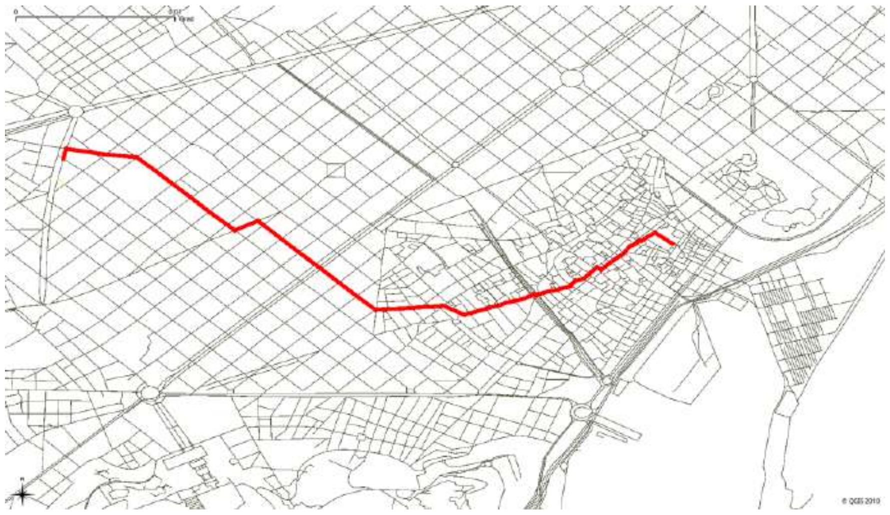
Ensemble des solutions admissibles fini mais combinatoirement grand

Algorithme pour trouver une solution

Problème de plus court chemin

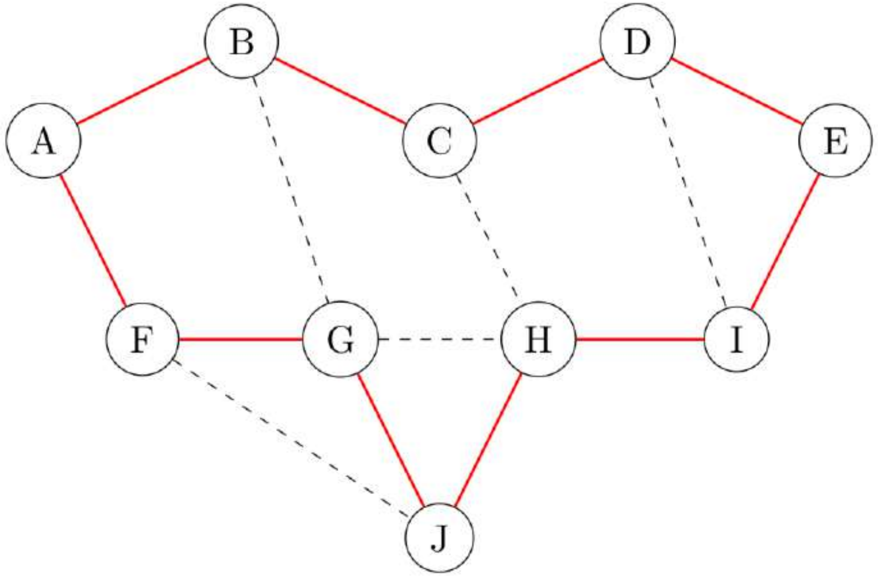
$$\min_{x \in \mathcal{X}} f(x)$$

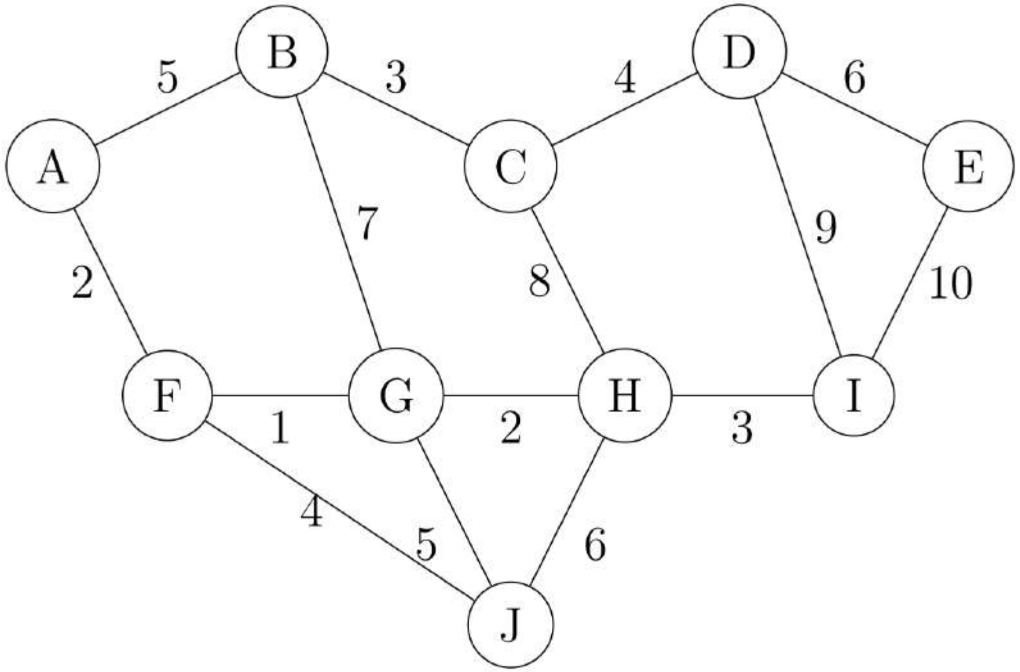




Complexité en O(a + n log(n))

Problème de voyageur de commerce





Pas d'algorithme polynomial

Complexité

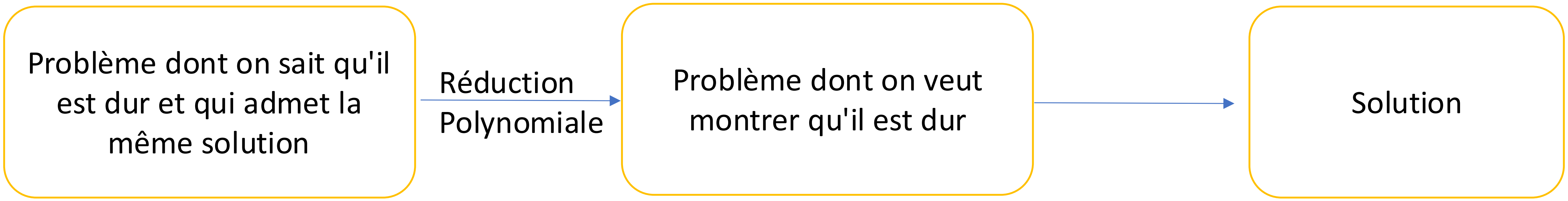
	Size n			
Complexité	10	20	50	60
n	$0,01$ µs	$0,02$ µs	$0,05$ µs	$0,06$ µs
n^2	$0,1$ µs	$0,4$ µs	$2,5$ µs	$3,6$ µs
n^3	1 µs	8 µs	125 µs	216 µs
n^5	$0,1$ ms	$3,2$ ms	$312,5$ ms	$777,6$ ms
2^n	~ 1 µs	~ 1 ms	~ 13 jours	~ 36.5 ans

Accélération ordinateurs : peu d'impact sur la taille des instances résolues

Complexité	Ordinateur actuel	Ordinateur $100 \times$ plus rapide	Ordinateur $1000 \times$ plus rapide
n	N_1	$100N_1$	$1000N_1$
n^2	N_2	$10N_2$	$31.6N_2$
n^3	N_3	$4.64N_3$	$10N_3$
n^5	N_4	$2.5N_4$	$3.98N_4$
2^n	N_5	$N_5 + 6.64$	$N_5 + 9.97$
3^n	N_6	$N_6 + 4.19$	$N_6 + 6.29$

Problème NP complet

Comment montrer que l'on ne peut **pas** trouver d'algorithme polynomial pour un problème



Algorithmes plus efficaces pour le voyageur de commerce

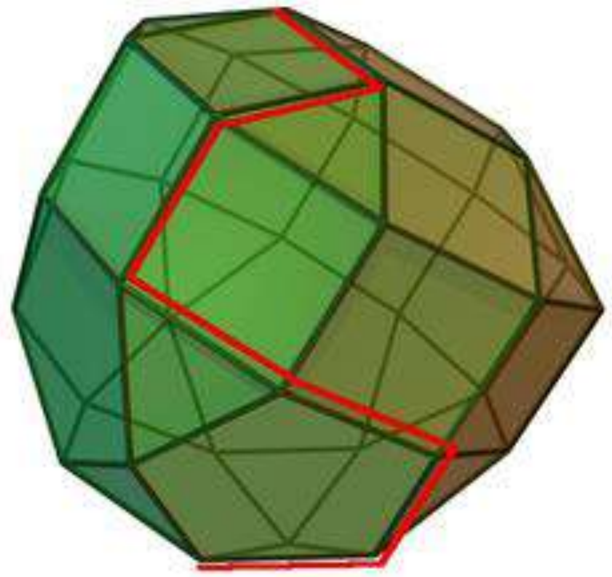
Cela ne veut pas dire que l'on n'aura pas d'algorithme efficace en pratique

1954	Dantzig, Fulkerson, Johnson	49 cities
1971	Held, Karp	64 cities
1975	Camerini, Fratta, Maffioli	67 cities
1977	Grötschel	120 cities
1980	Crowder, Padberg	318 cities
1987	Padberg, Rinaldi	532 cities
1987	Grötschel, Holland	666 cities
1987	Padberg, Rinaldi	2′392 cities
1994	Applegate, Bixby, Chvátal, Cook	7′397 cities
1998	Applegate, Bixby, Chvátal, Cook	13′509 cities
2001	Applegate, Bixby, Chvátal, Cook	15′112 cities
2004	Applegate, Bixby, Chvátal, Cook, Helsgaun	24′978 cities

Aujourd'hui : solution quasi-optimales sur un problème à 1,69 milliard d'étoiles

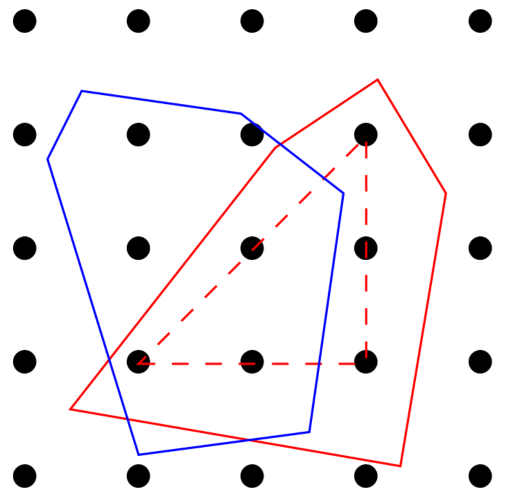
Programmation linéaire

$$\min c^\top x$$
$$\text{s.t. } Ax \leq b$$



Programmation linéaire en nombres entiers

$$\min c^\top x$$
$$\text{s.t. } Ax \leq b$$
$$x \in \mathbb{Z}^p \times \mathbb{R}^{n-p}$$

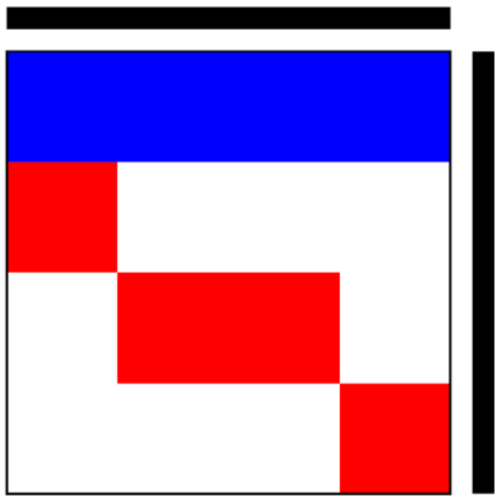


Le relaché linéaire permet de trouver des bornes inférieures

Entre 1990 et 2015, accélération d'un facteur

- 780 mille des algorithmes
- 450 milliard en prenant en compte l'accélération des ordinateurs

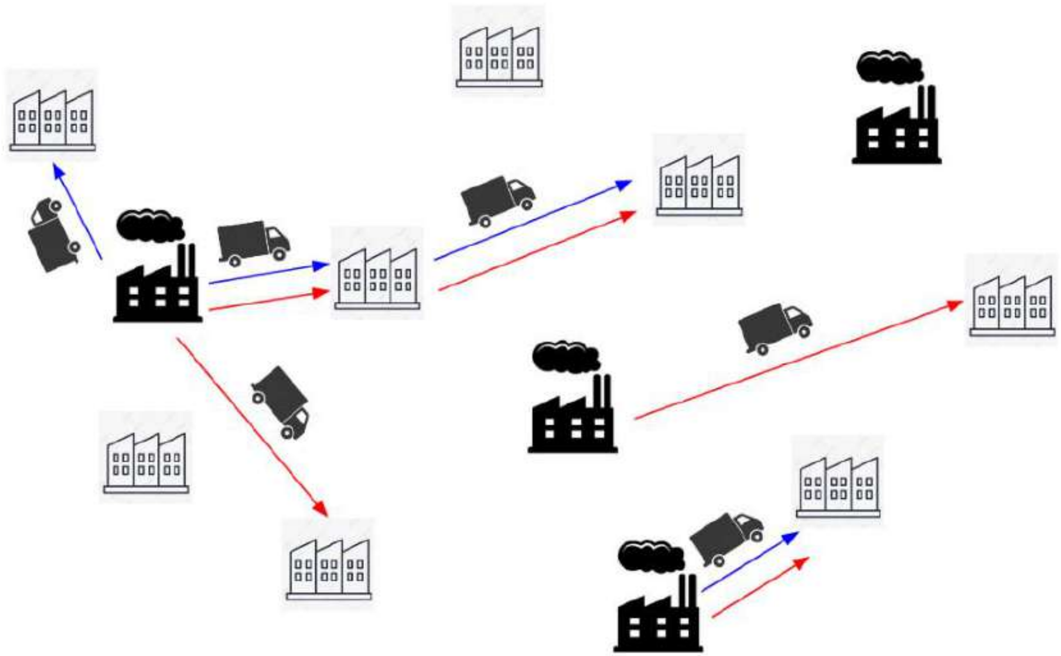
Résolution de problèmes avec ~ 1 million de contraintes et de variables



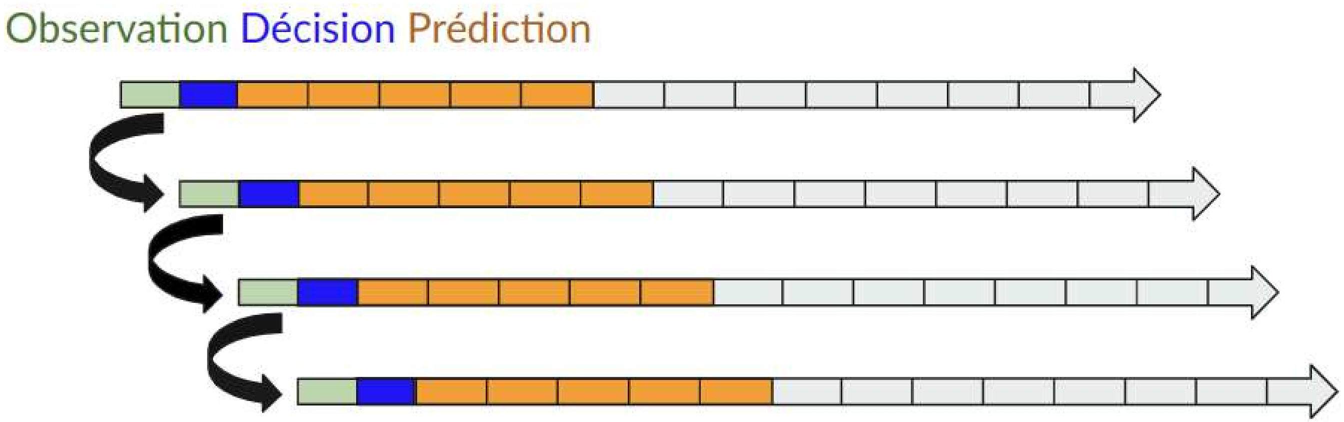
Un exemple de projet de RO

Dispatch Renault

Dispatch chez Renault

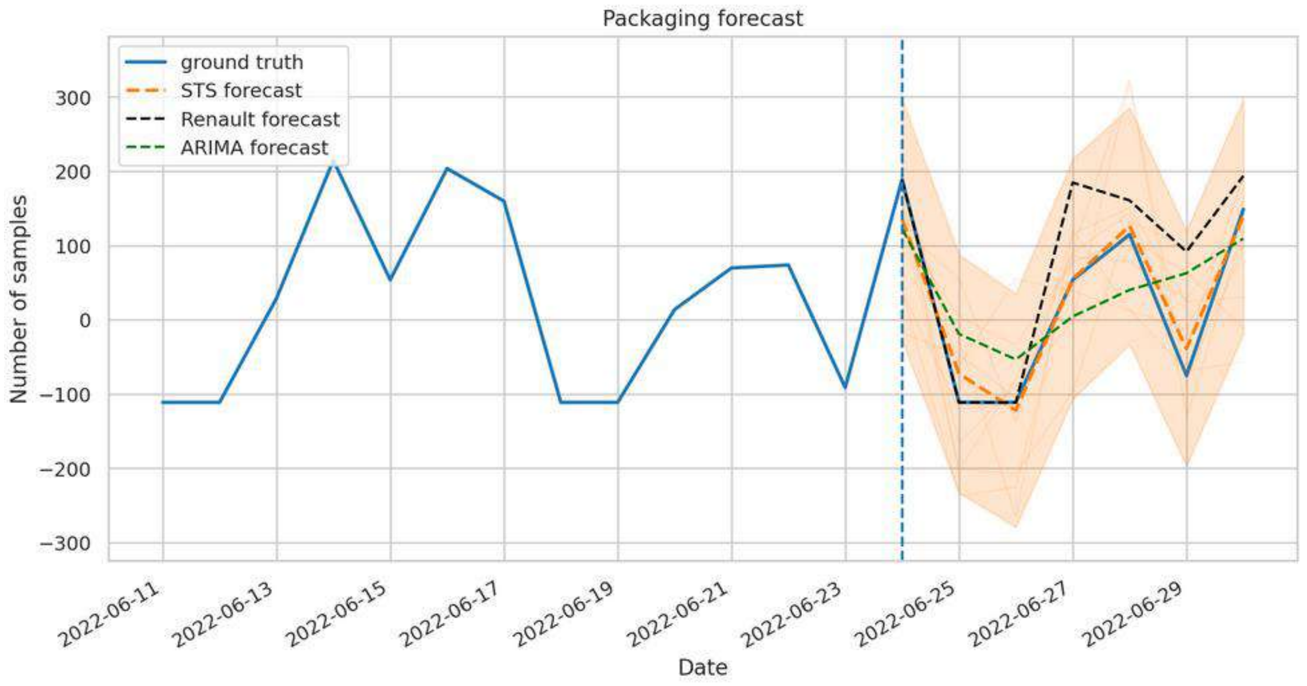


Aléa et dynamique de décision



Etat : routes non terminées, inventaires

Politique : routes à commencer aujourd'hui



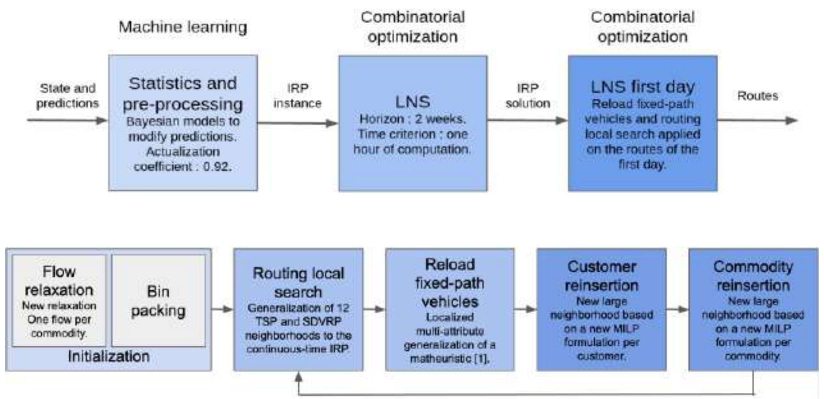
Problème stochastique multi-étapes

Etat : routes non terminées, inventaires
Politique : routes à commencer aujourd'hui

$$\min_{\text{policy}} \mathbb{E}\left[\sum_t \left(\begin{array}{l} \quad \text{routes' cost} \\ + \text{depots' inventory cost} \\ + \text{customers' inventory cost} \\ + \text{customers' shortage cost} \\ + \text{infeasibility penalization} \end{array}\right)_t \middle| \begin{array}{l} \text{predictions} \\ \text{available} \end{array}\right] \qquad \text{(SIRP)}$$

$$\text{s.t.} \left| \begin{array}{l} \text{Policy is feasible given available information.} \\ \text{Policy is non-anticipative.} \\ \text{Inventory dynamics are respected.} \\ \text{Inventory dynamics and routes are coherent.} \end{array}\right.$$

Algorithme en **production** &
publication scientifique

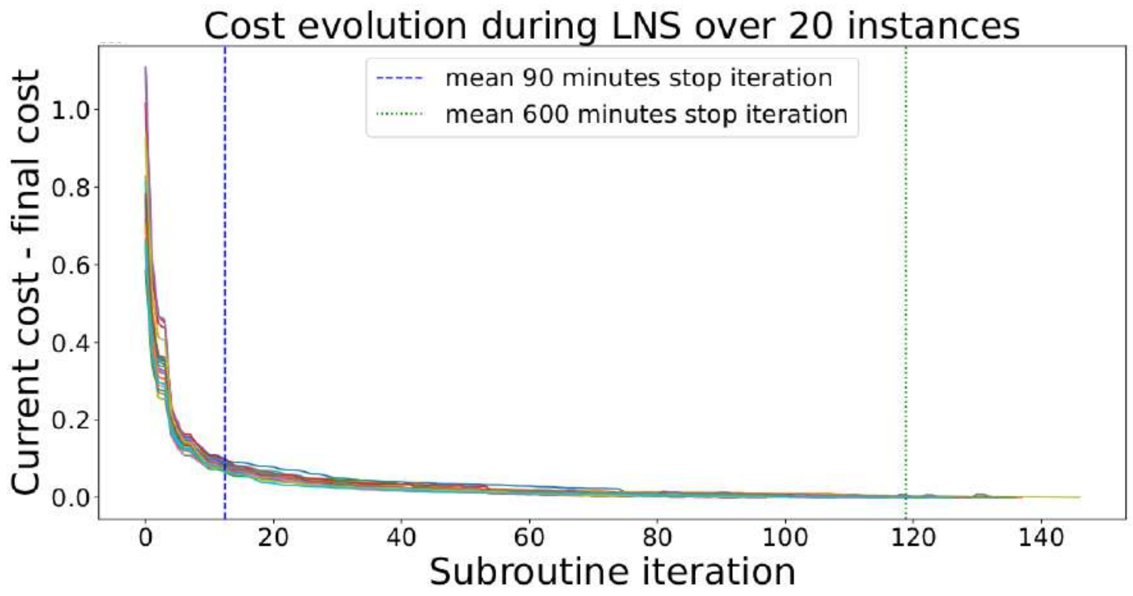


Gains environnementaux et économiques

Réduction de l'**impact carbone** :

- Moins de tonnes.kilomètres
- Moins de carton utilisé en substitu d'emballage vide

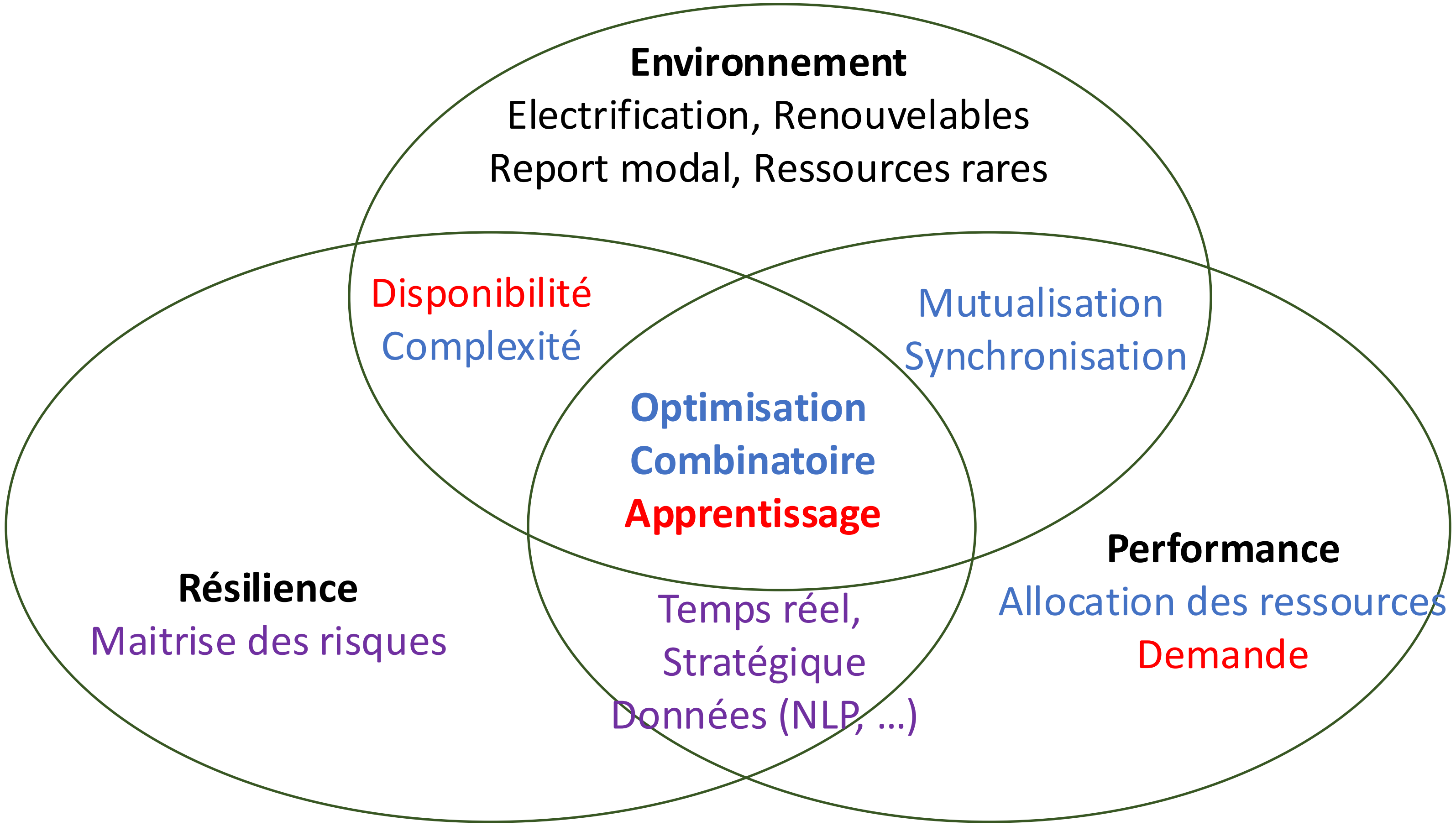
Coûts	Ancien / nouveau	Nouveau / Nouveau + ML
Transport	12%	1.5%
Pénurie d'emballages	43%	6.4%
Global	20%	4.6%



Perspectives

Académiques en industrielles

RO, apprentissage et transition écologique



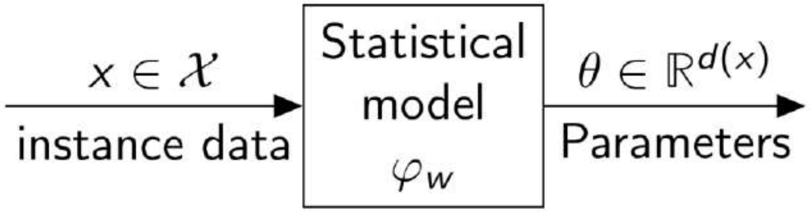
Faire plus, en consommant moins, à technologie constante

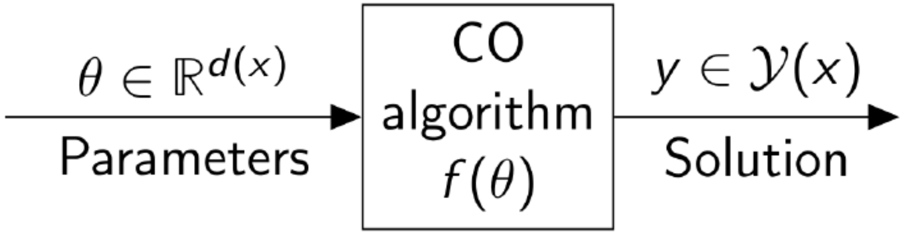
Maitriser la complexité des énergies renouvelables

Transition numérique au service de la transition écologique et de la performance industrielle

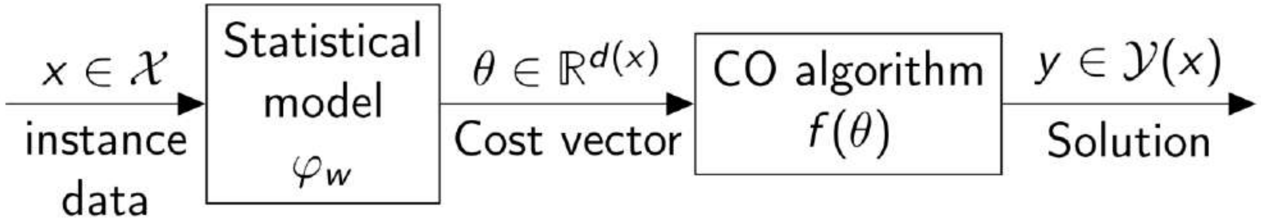
Ne pas dissocier optimisation et statistiques

Approche classique : estimation séparée de l'optimisation





Performance accrue si l'application est prise en compte



Optimisation tirée par les données

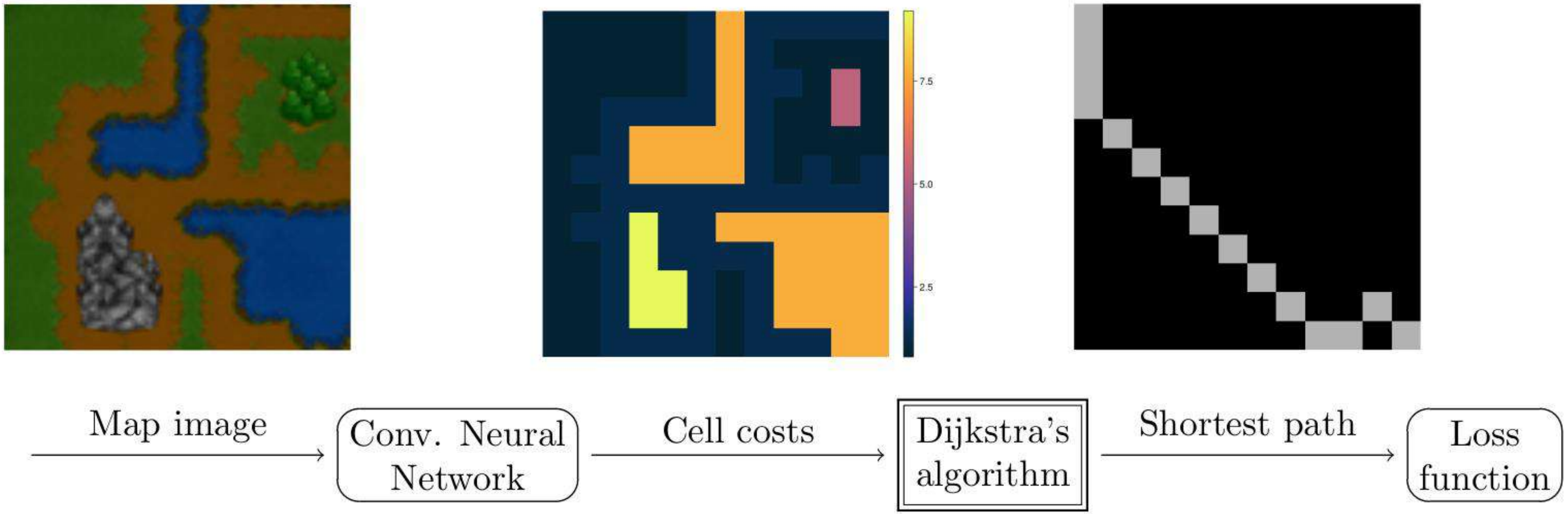
Etant donné une distribution jointe sur un contexte *x* et un aléa ξ,
trouver une politique

$$\delta : x \in \mathcal{X} \mapsto y \in \mathcal{Y}(x)$$

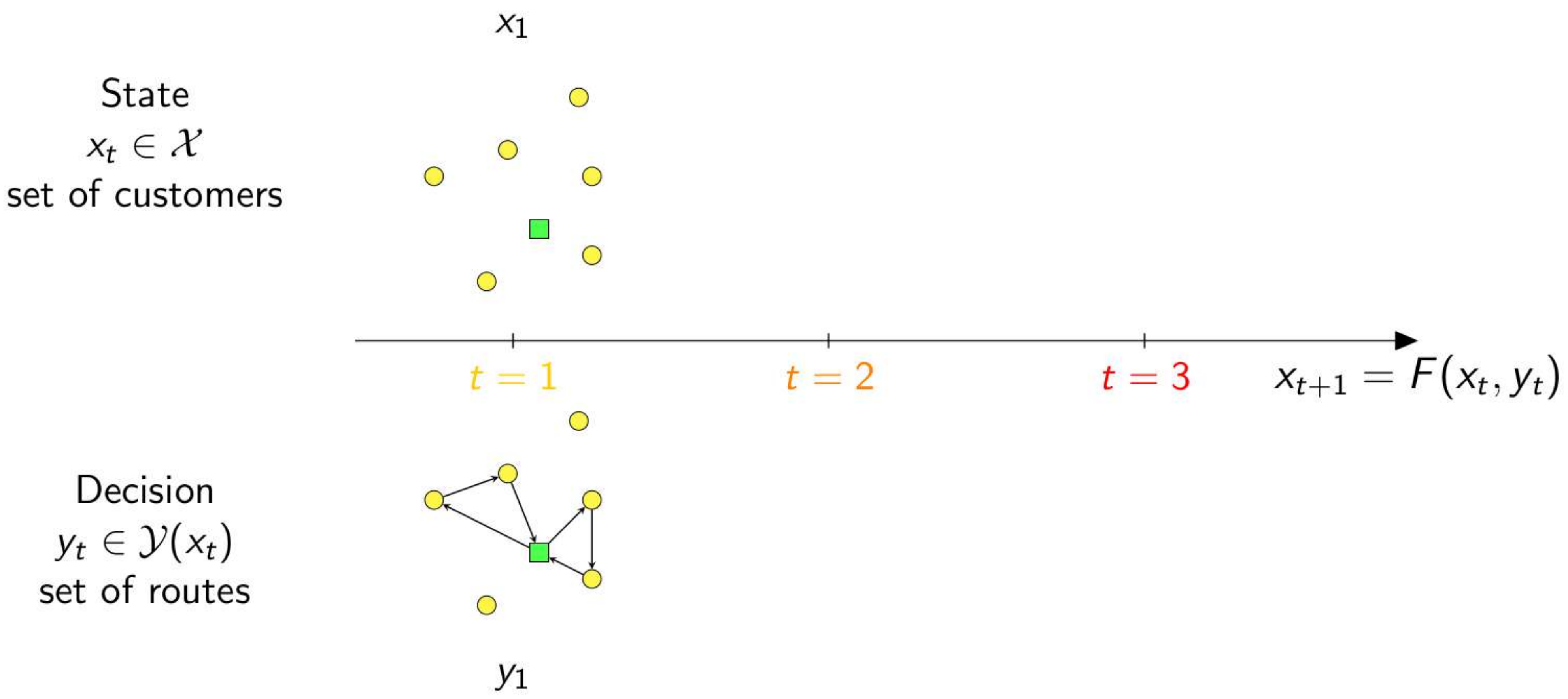
qui minimise

$$\min_{\delta} \mathbb{E}\left[c\left(x, \delta(x), \xi\right)\right]$$

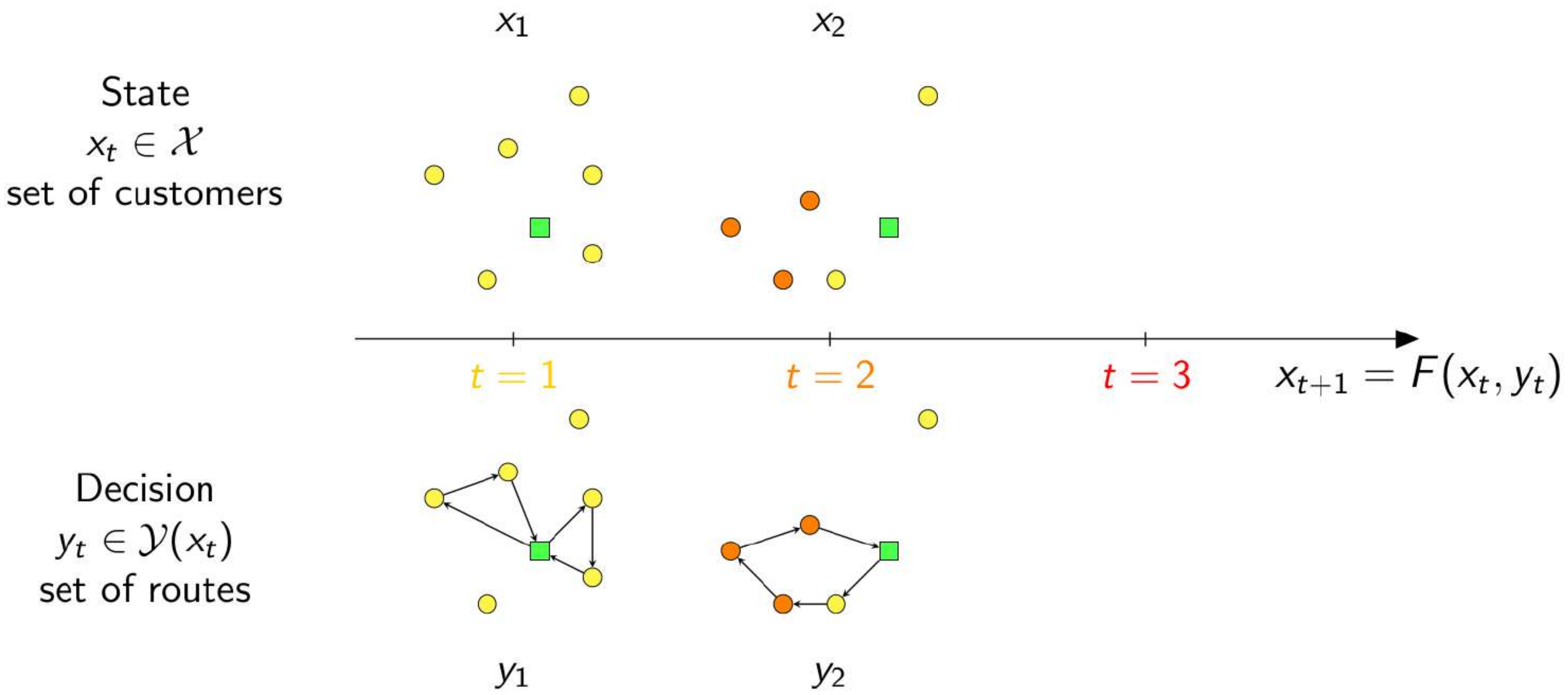
Optimisation tirée par les données



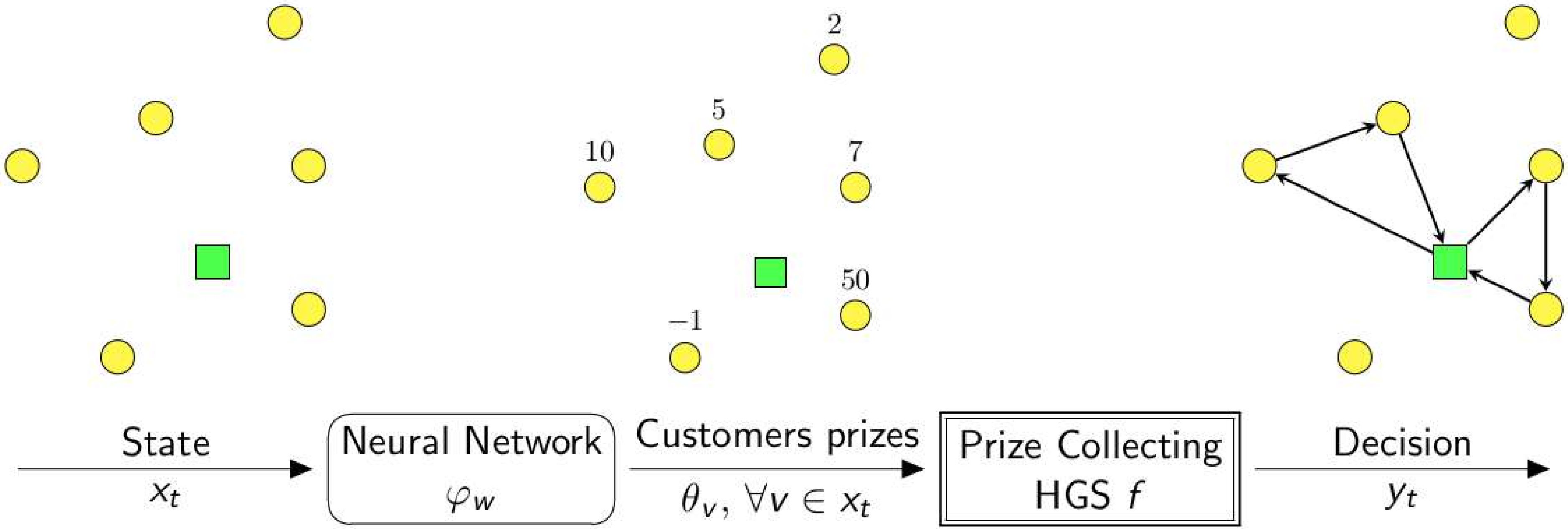
Problème dynamique



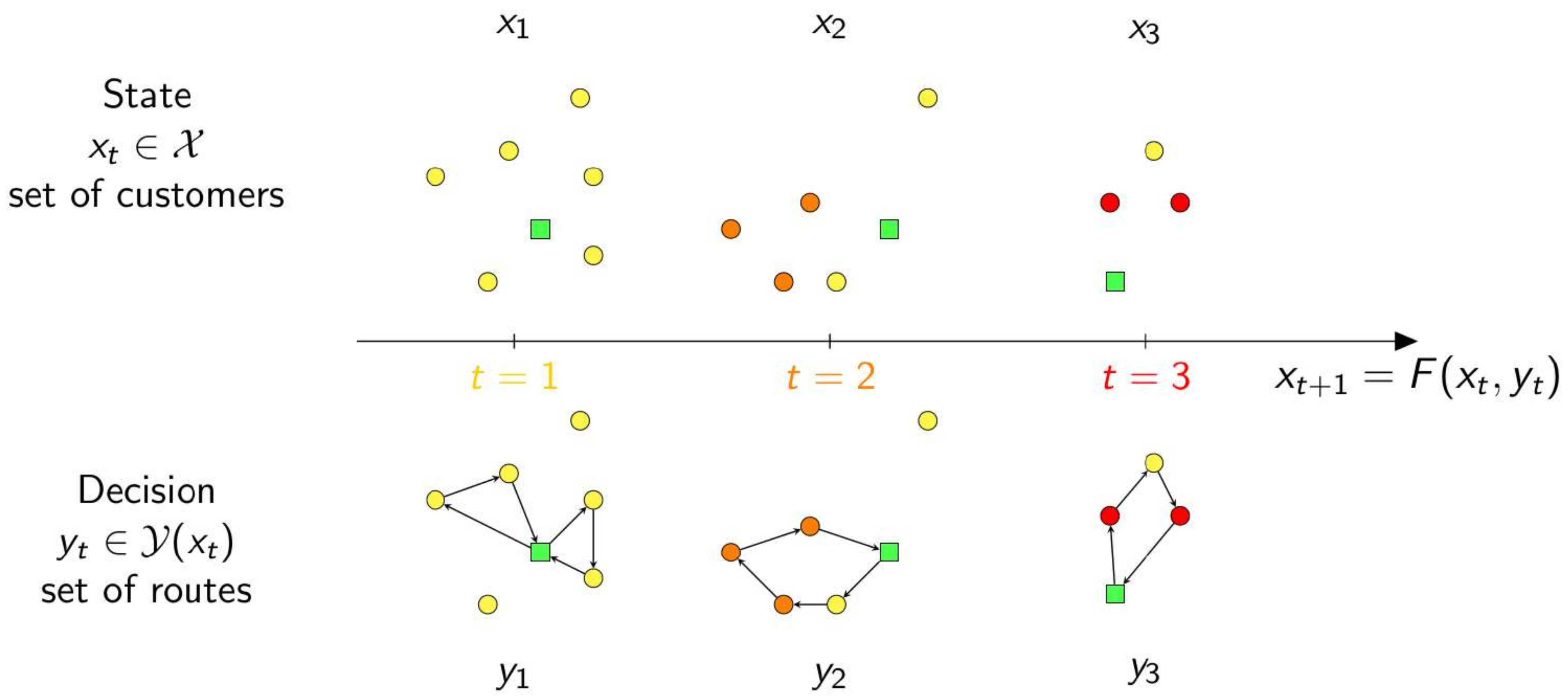
Problème dynamique



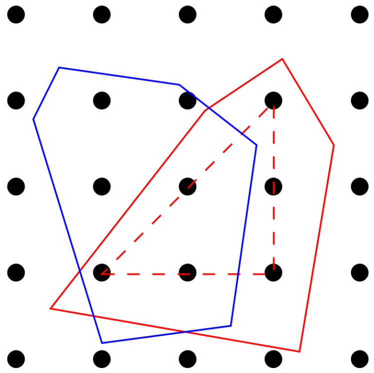
Politique pour un problème dynamique

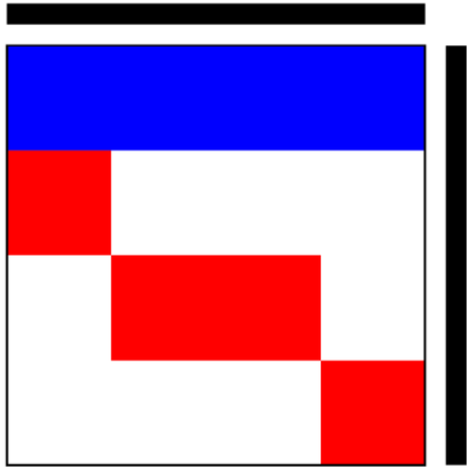


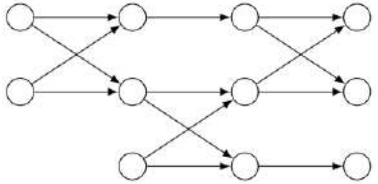
Problème dynamique



Enjeux mathématiques

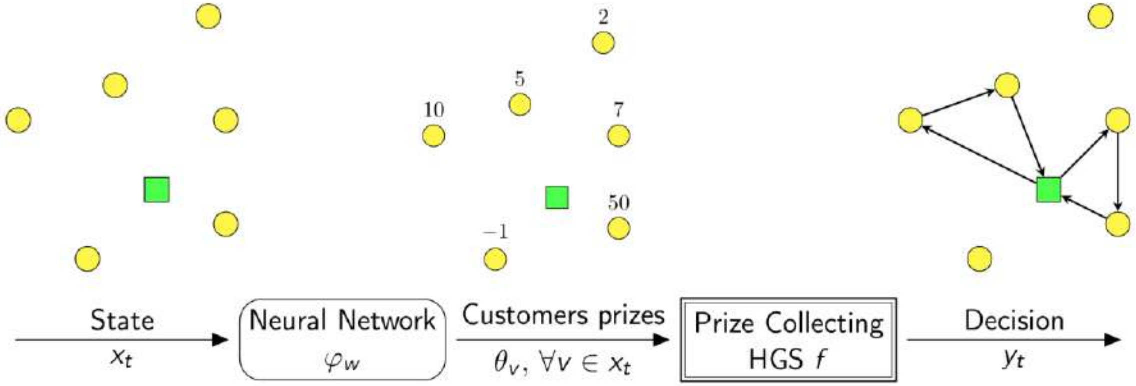






Optimisation combinatoire

Faire passer à l'échelle les algorithmes sur les problèmes d'optimisation des process industriels tels qu'ils se posent aujourd'hui

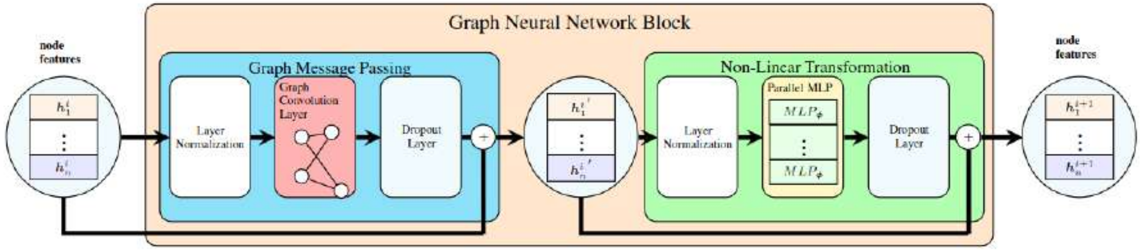


CO augmented Machine Learning

Prendre en compte toutes les données disponibles pour l'optimisation en temps réel ou stratégique

Apprentissage sur graphe

Assimilation de données (dont textuelles) dans des modèles de process industriels



Merci