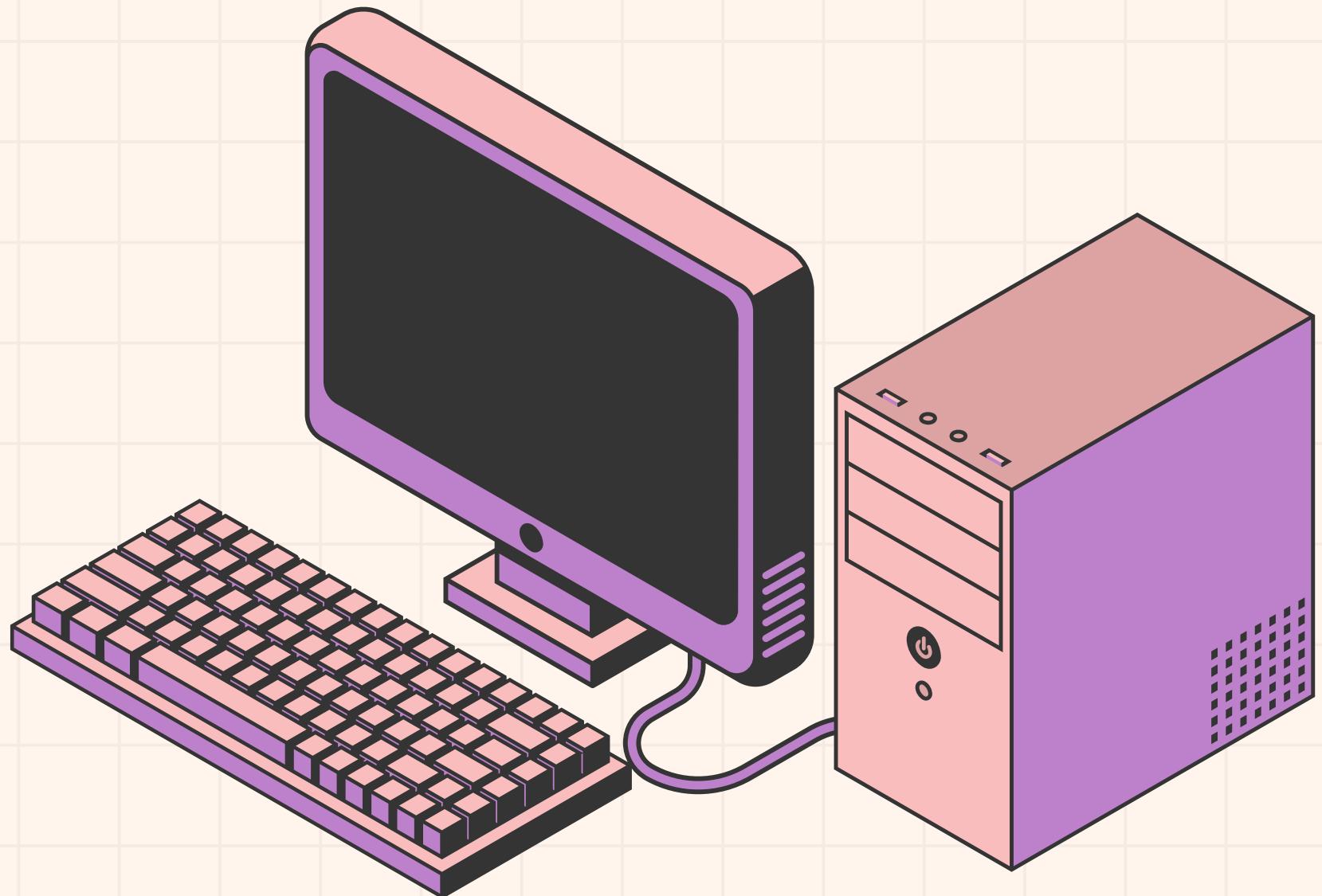


PROJET IA

SUJET 16 : COLORATION DE GRAPHE ET DE CARTE



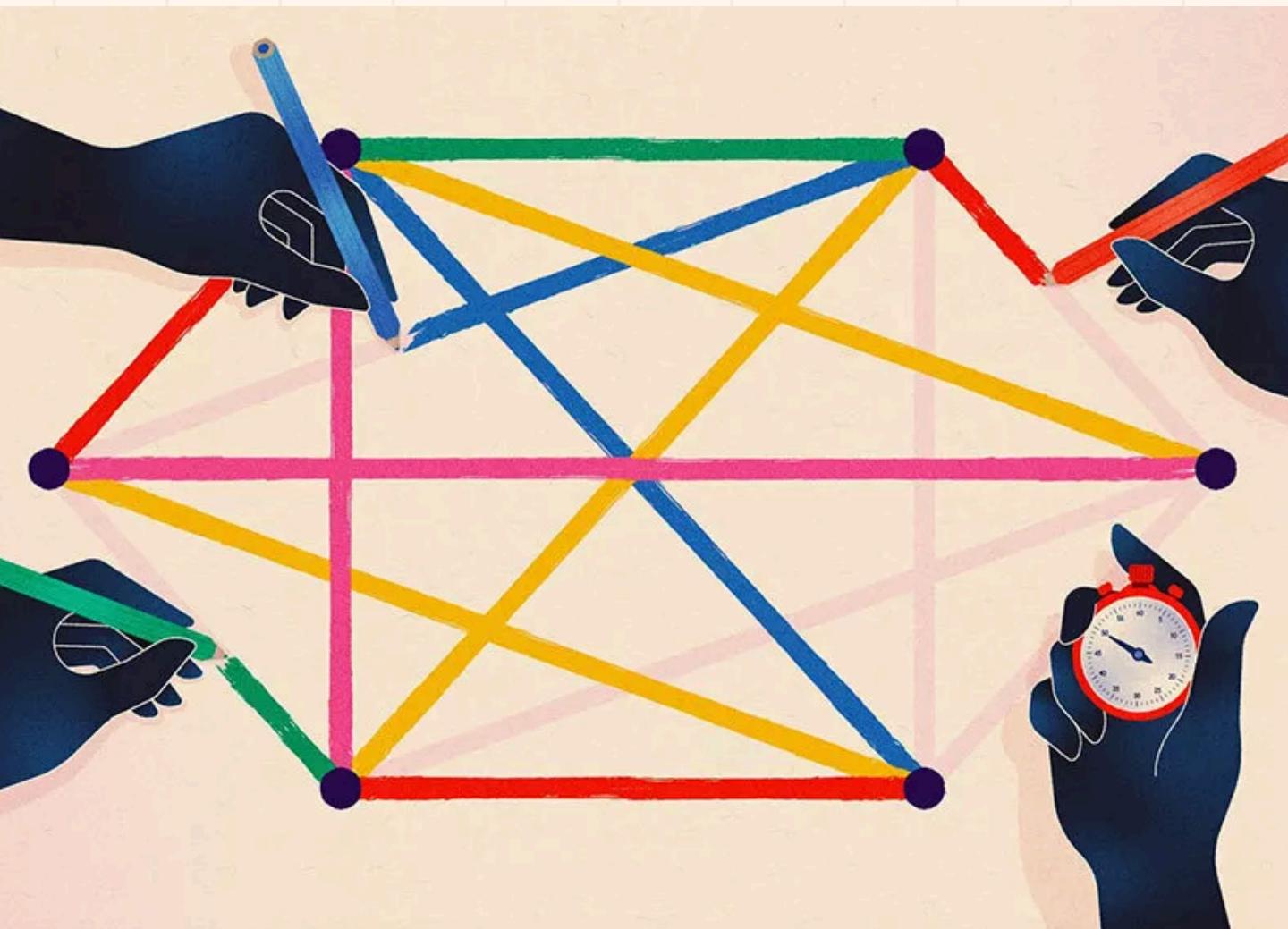
Elsa BODENAN

Shaili TUIL

Groupe 1 Finance

PROBLÈME DE COLORATION DE GRAPHE

- Attribuer une couleur à chaque nœud
- Deux nœuds voisins ≠ même couleur
- Objectif: utiliser le moins de couleurs possible



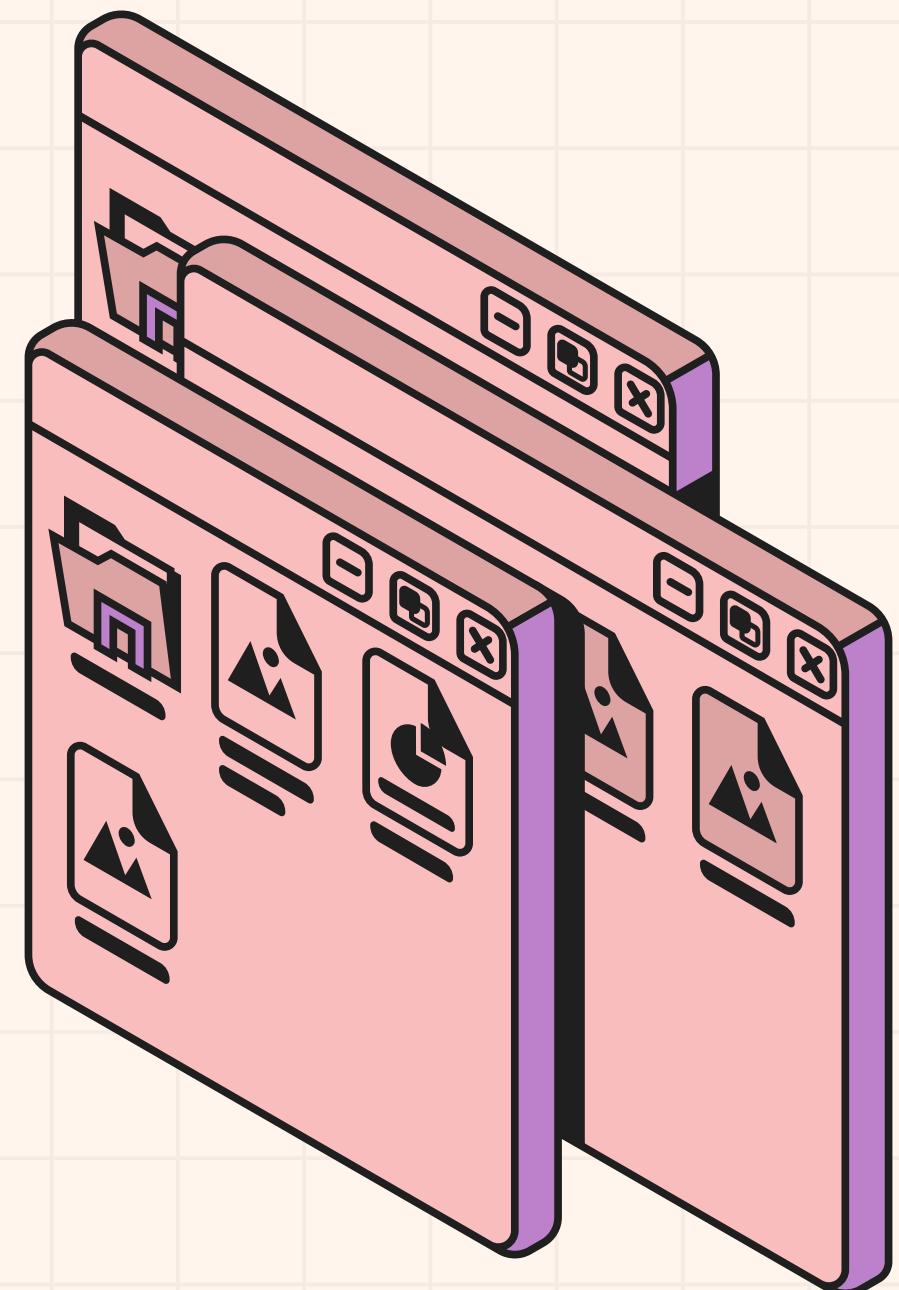
OBJECTIFS DU PROJET

- Résoudre la coloration de graphes et de cartes
- Deux versions du problème :
 - k-coloration (k fixé)
 - recherche du minimum k^*
- Créer un programme :
 - facile à utiliser
 - avec visualisations
 - avec export des résultats



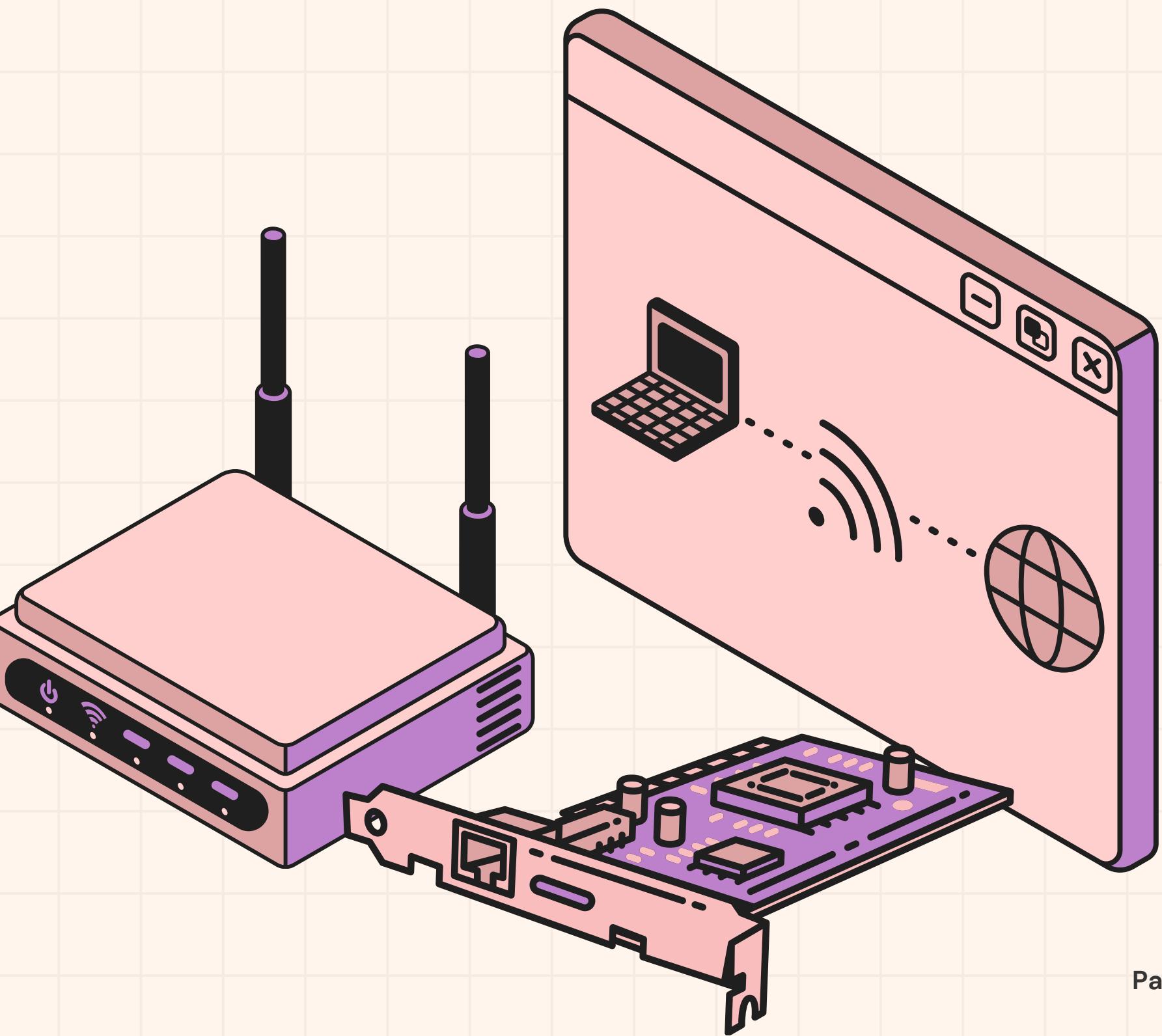
CONTEXTE THÉORIQUE

- La coloration de graphe est un problème NP-difficile
- Modélisation en programmation par contraintes
- Application directe à la coloration de cartes
- Théorème des 4 couleurs (cartes planaires)

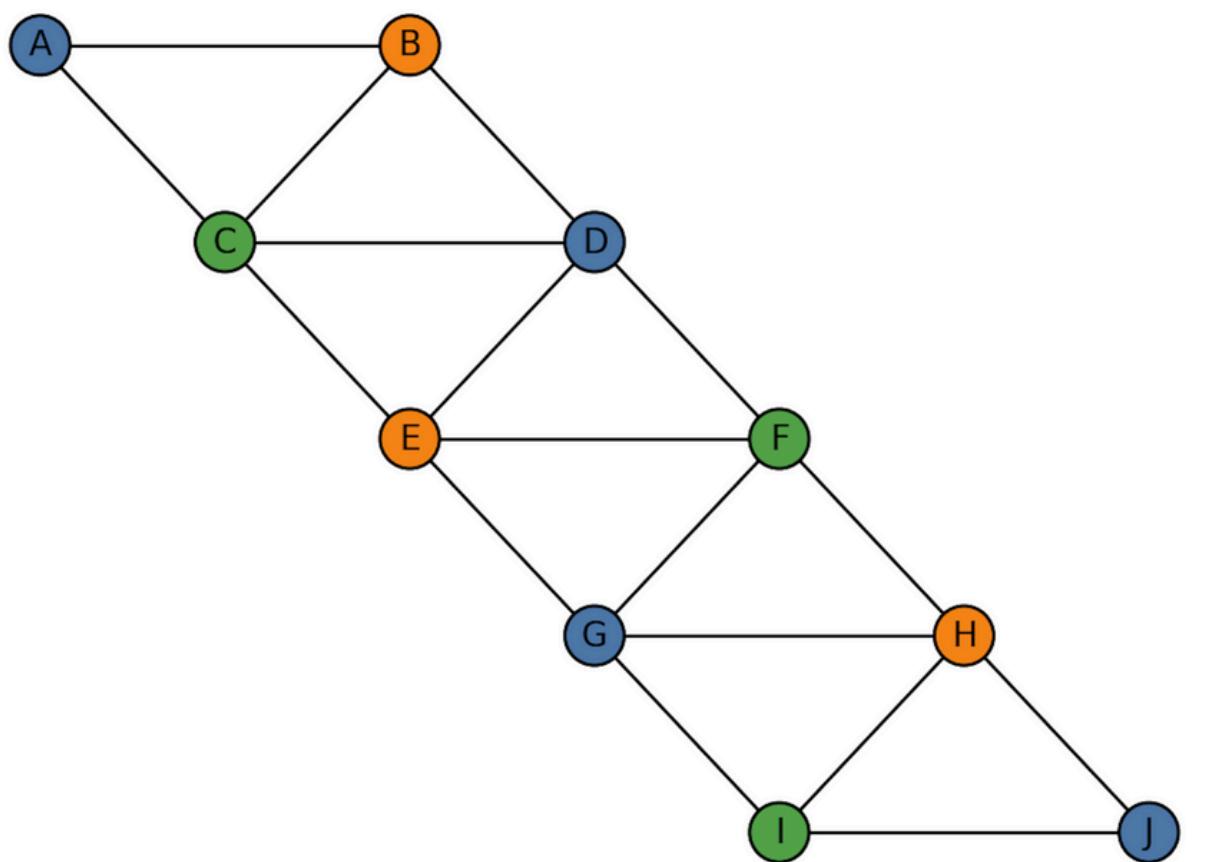


MÉTHODE IMPLÉMENTÉES

- Heuristiques (rapides):
 - Greedy
 - DSATUR
- Méthodes exactes:
 - CP-SAT avec OR-Tools



map_like | greedy | colors=3 | valid=True

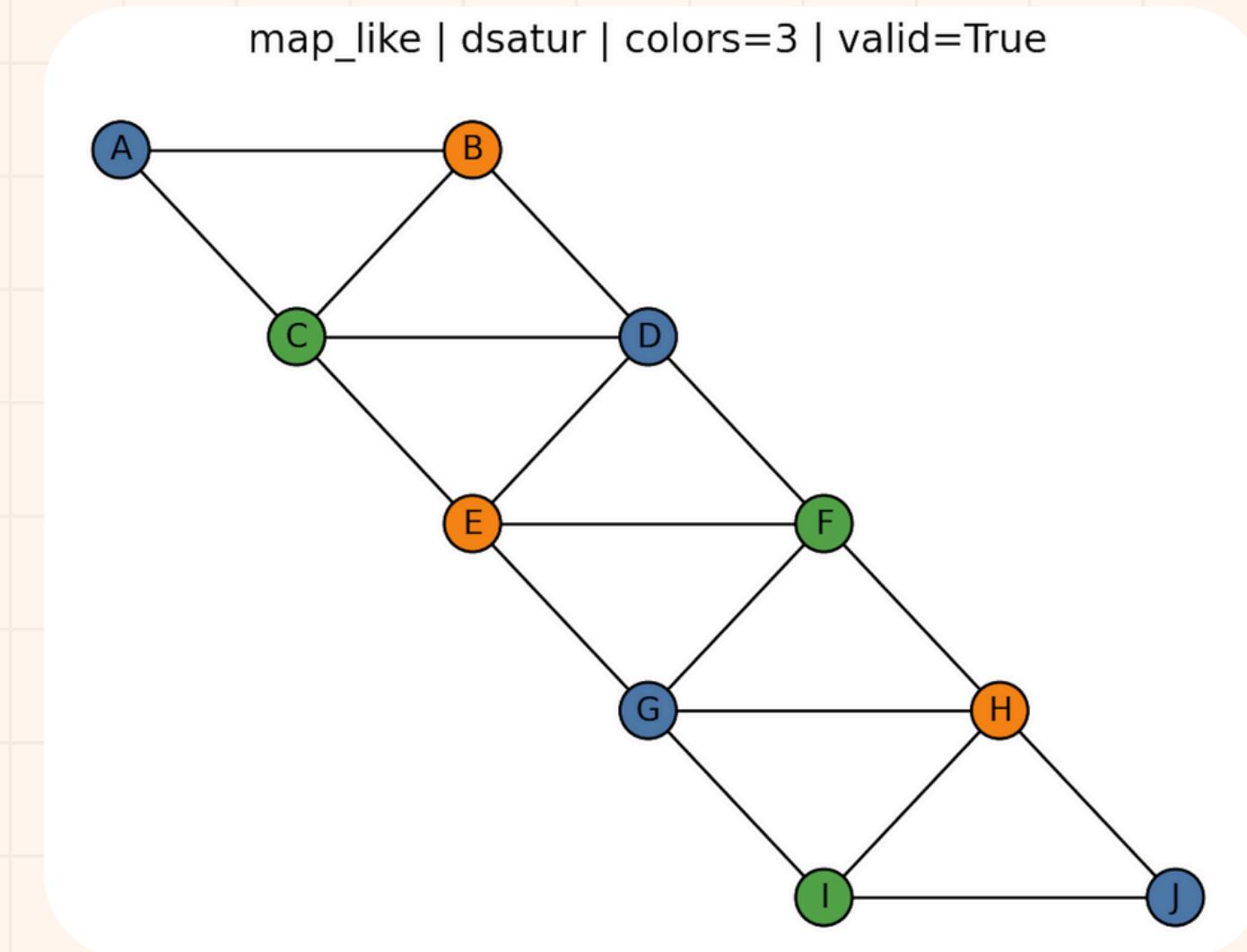


HEURISTIQUE GREEDY

- Choisit la plus petite couleur disponible
- Très rapide
- Dépend fortement de l'ordre des nœuds
- Sert de baseline

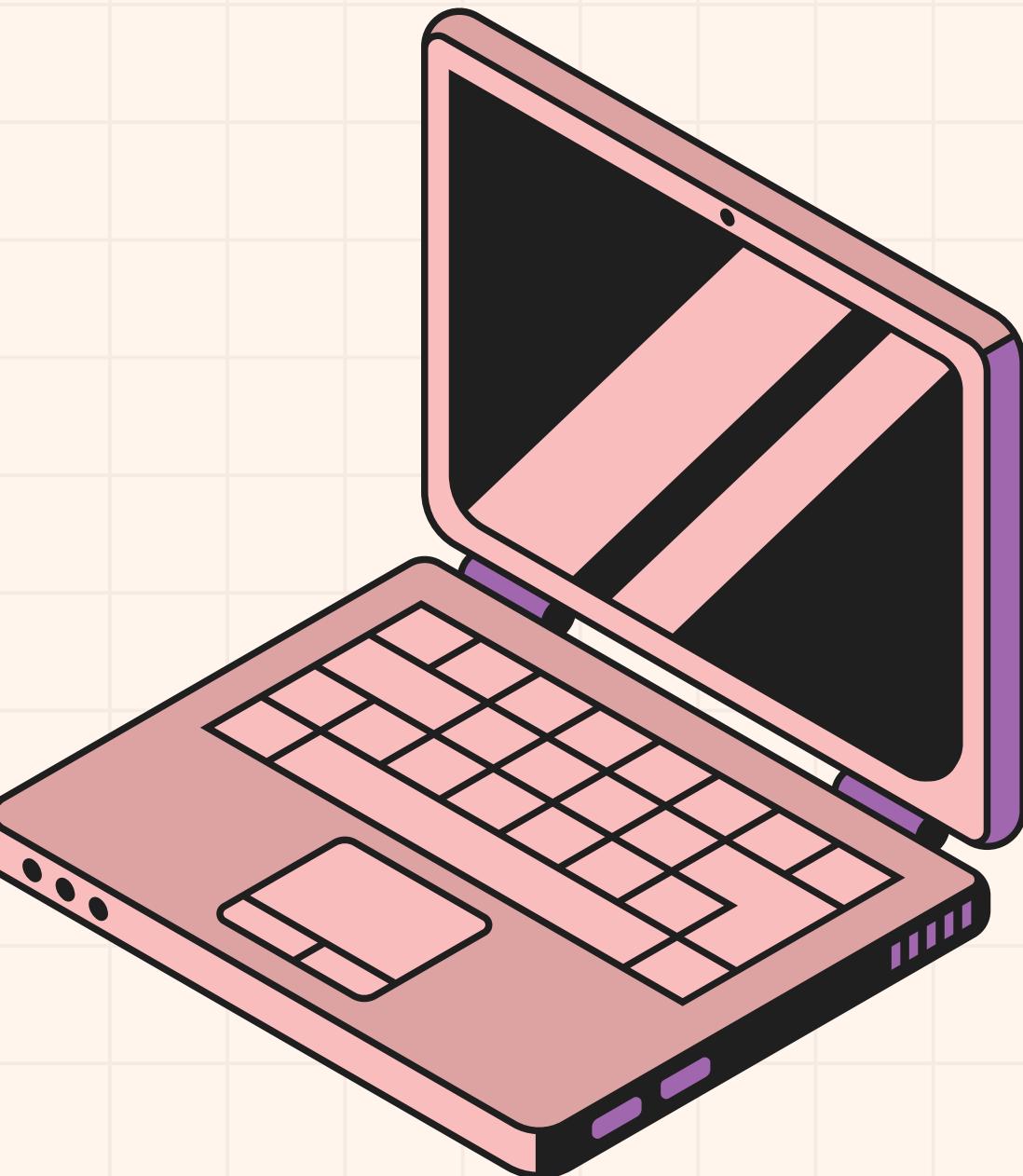
HEURISTIQUE DSATUR

- Choisit le nœud le plus constraint
- Réduit le nombre de couleurs
- Toujours très rapide
- Bonne borne supérieure pour CP-SAT



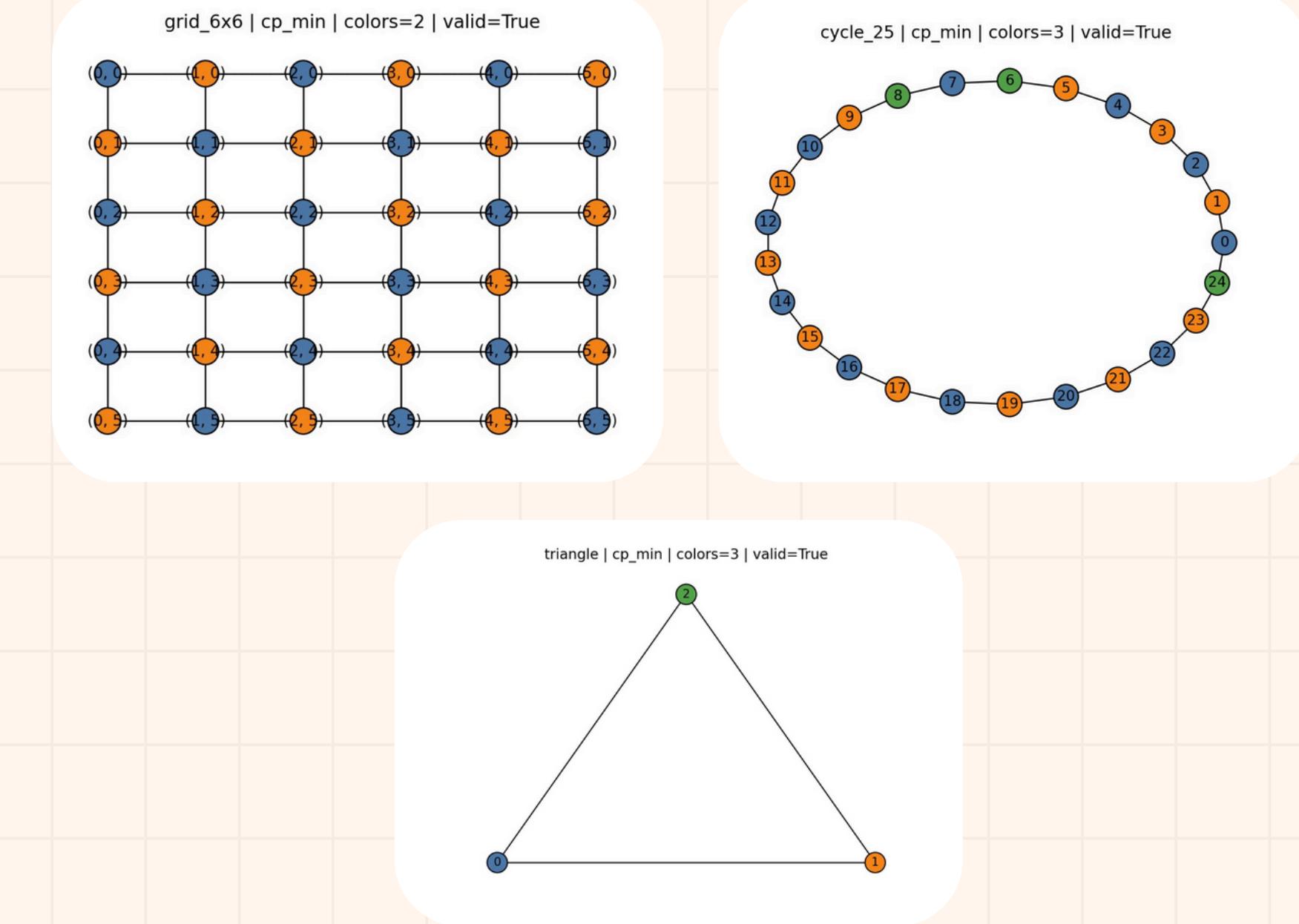
CP-SAT (OR-TOOLS)

- Variable = couleur de chaque nœud
- Contraintes : voisins \neq même couleur
- Deux modes :
 - cp_k : k fixé
 - cp_min : recherche du minimum k*



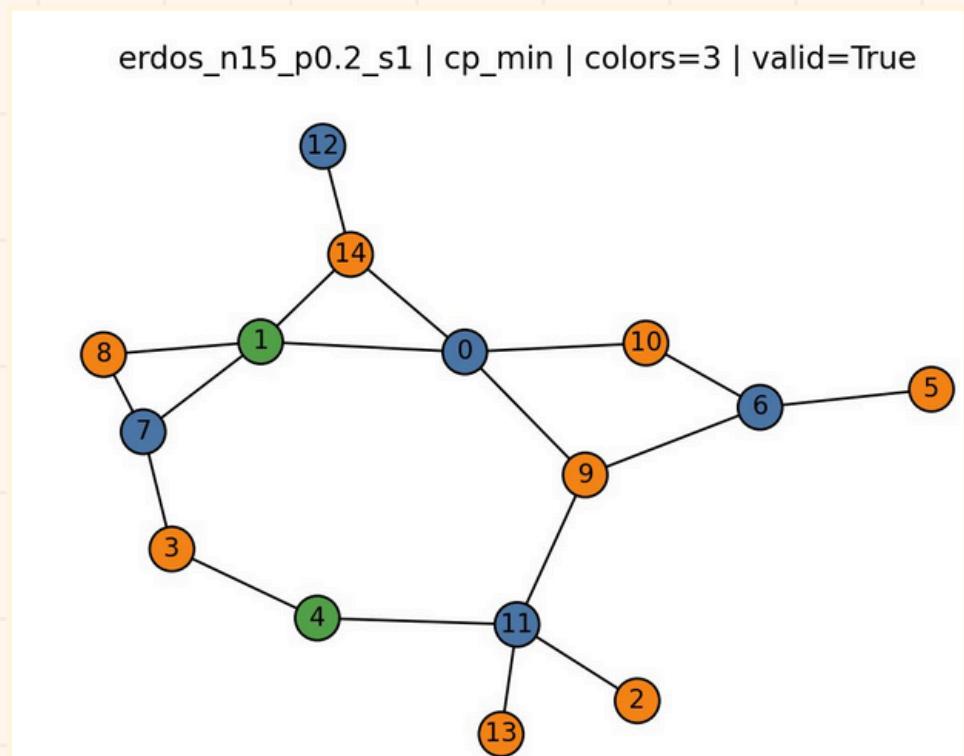
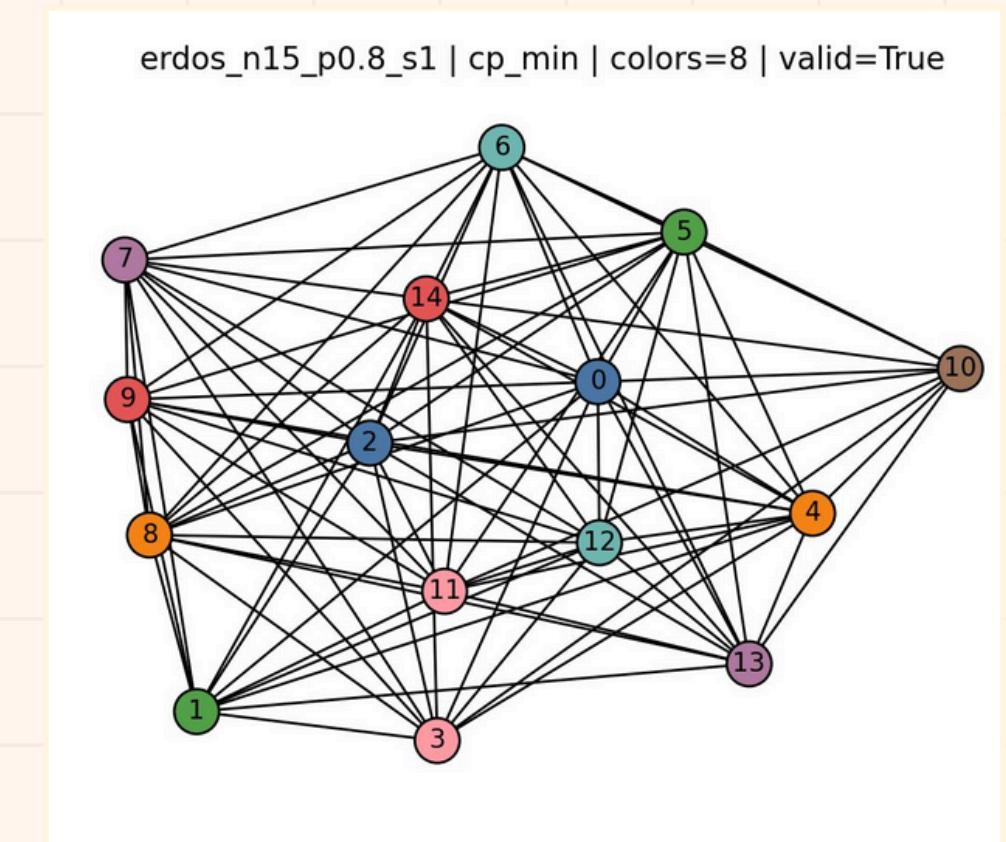
CP_MIN: RECHERCHE OPTIMALE

- Utilisation de bornes :
 - LB : clique
 - UB : DSATUR
- Recherche uniquement entre LB et UB
- Résultat certifié optimal



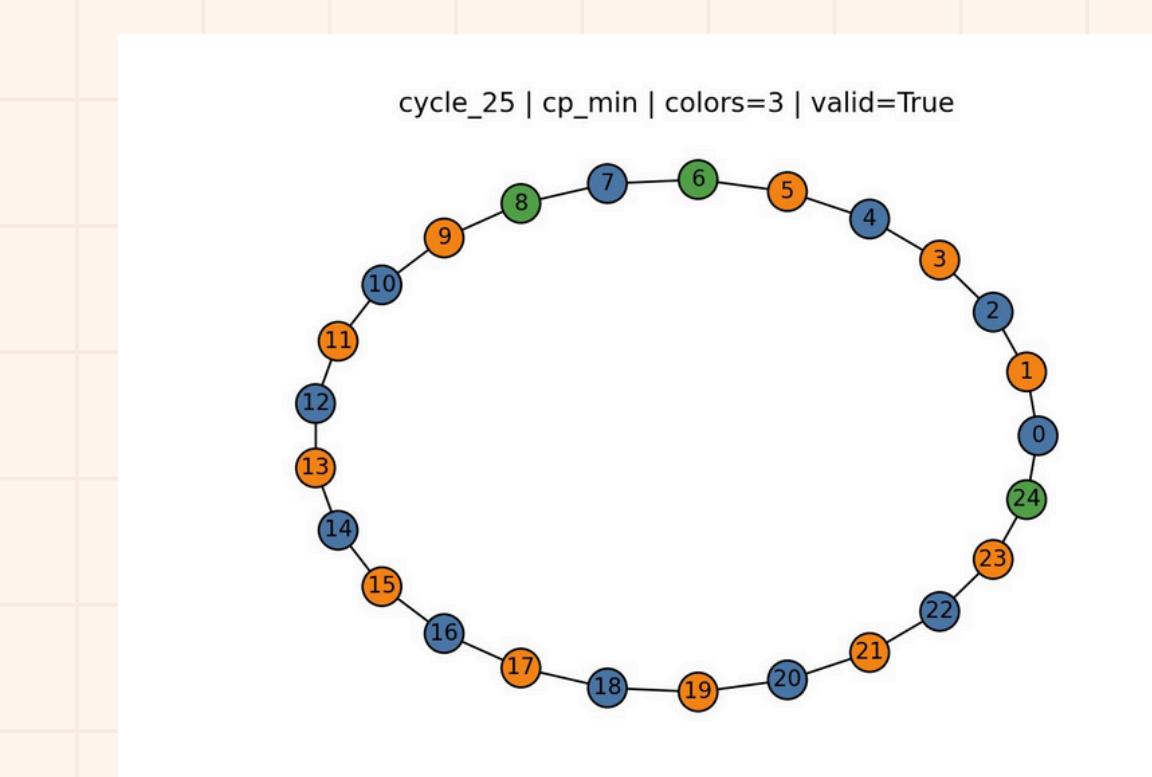
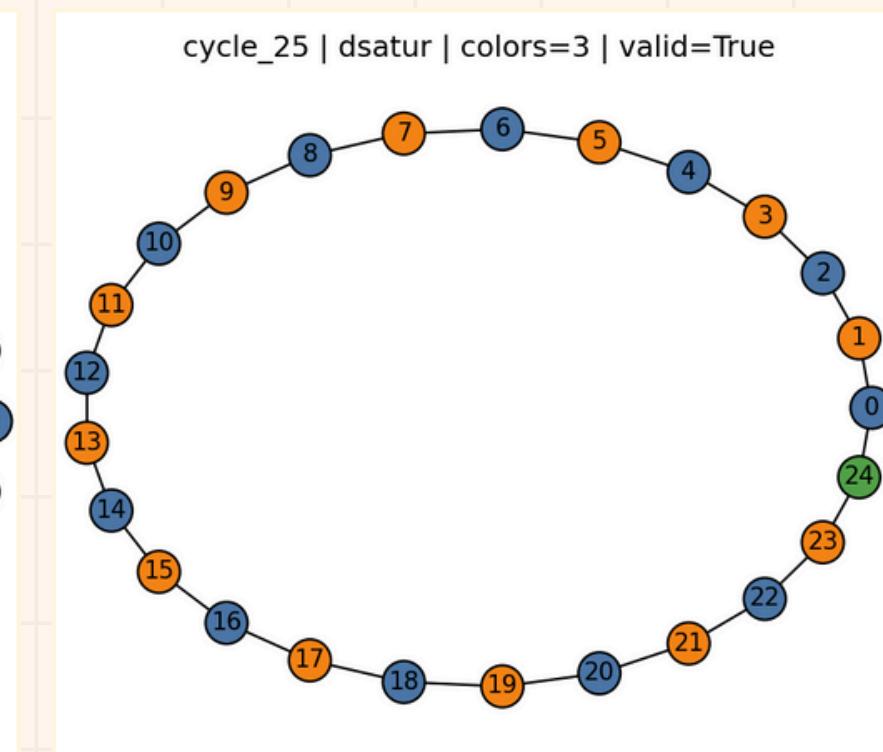
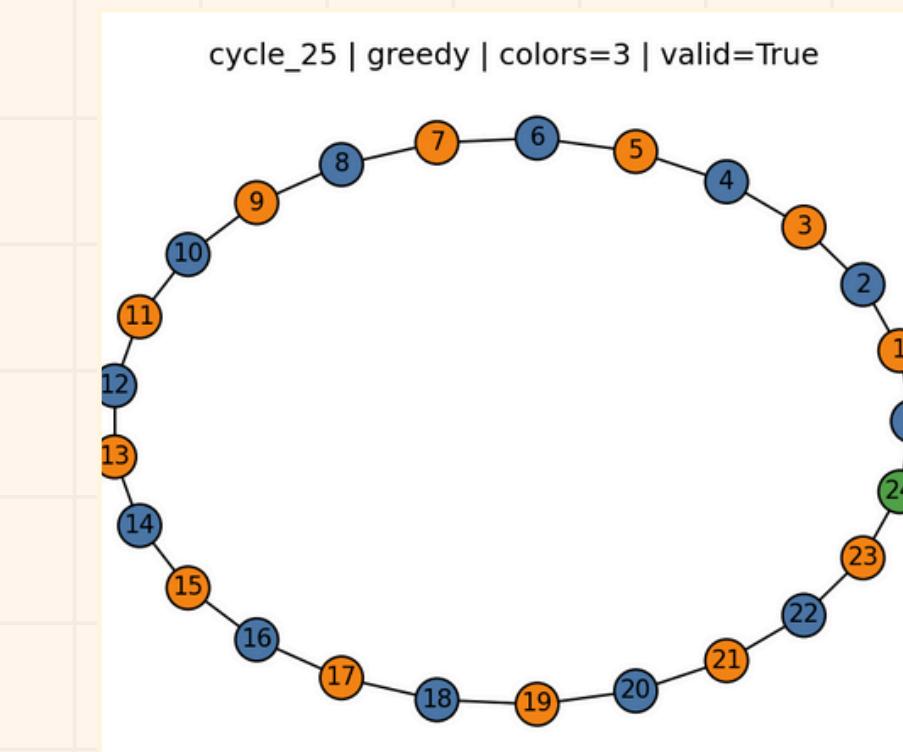
EXPÉRIMENTATIONS

- Instances testées :
 - map_like
 - triangle
 - cycles (pair / impair)
 - grilles
 - graphes Erdős-Rényi
- Influence de :
 - n (taille)
 - p (densité)
 - seed (structure)



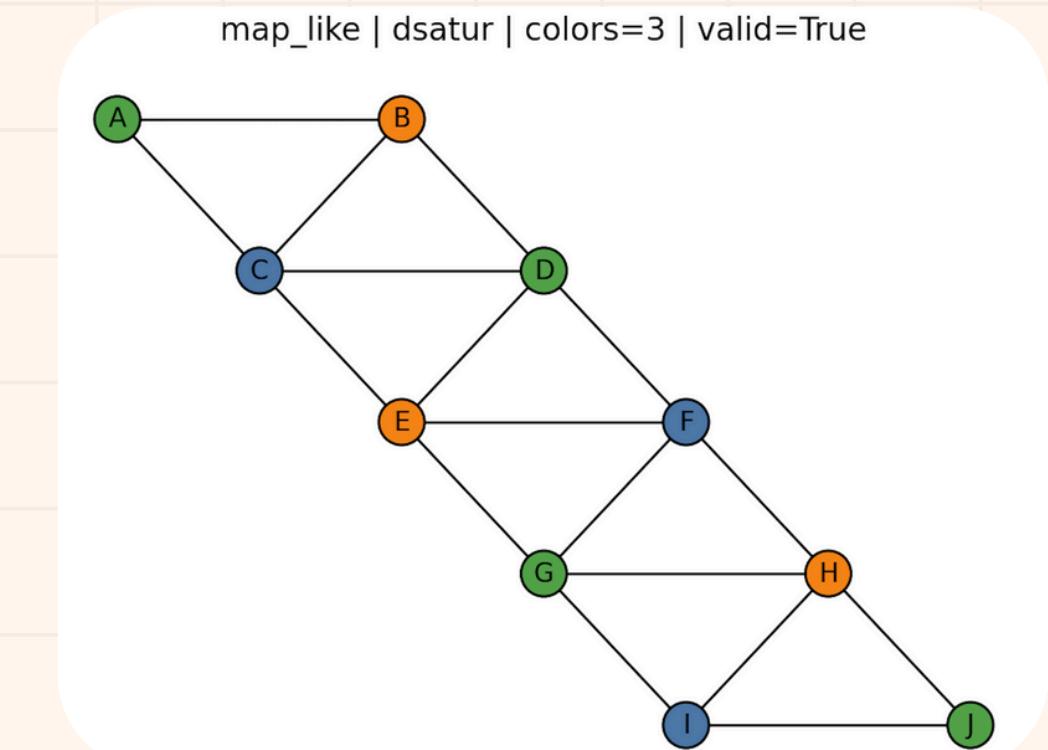
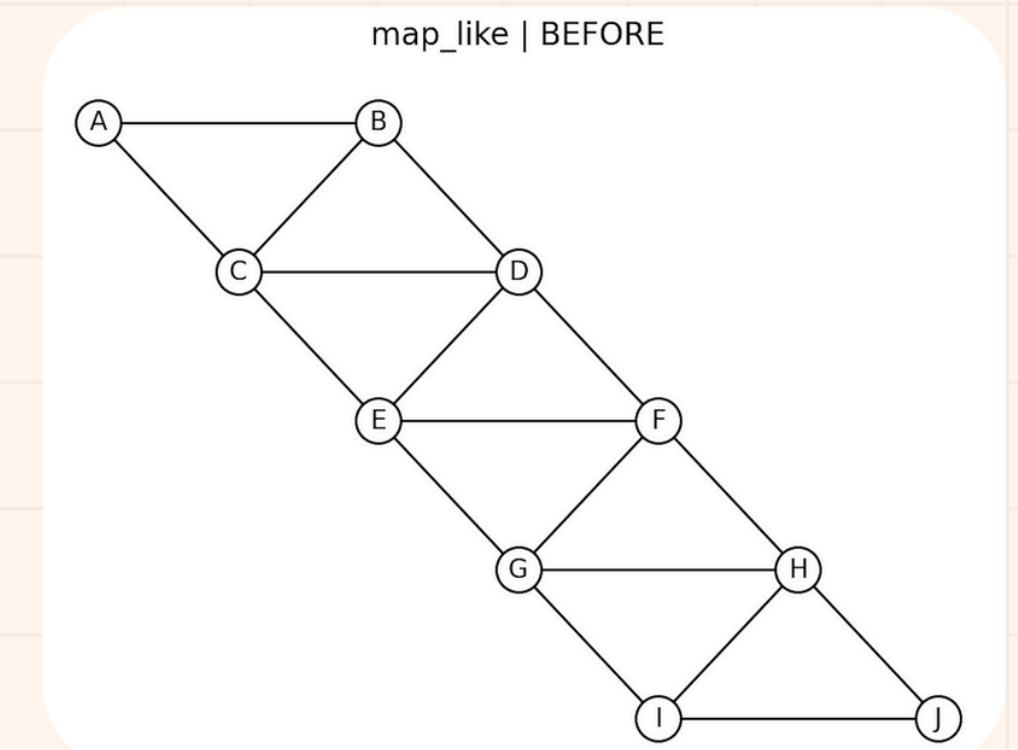
COMPARAISON

- Greedy :
 - Très rapide
 - Pas optimal
- DSATUR :
 - Bon compromis
- cp_min :
 - Plus lent
 - Optimal garanti



BONUS & FONCTIONNALITÉS

- Mode compare
- Mode benchmark (CSV)
- Visualisation avant / après
- Mode interactif & non interactif



CONCLUSION

- Problème complexe mais bien modélisable
- Heuristiques utiles pour la rapidité
- CP-SAT permet de certifier l'optimalité
- Projet complet : algorithmes + visualisation + outils

