

PRÉSENTATION DU PROJET OPTIM

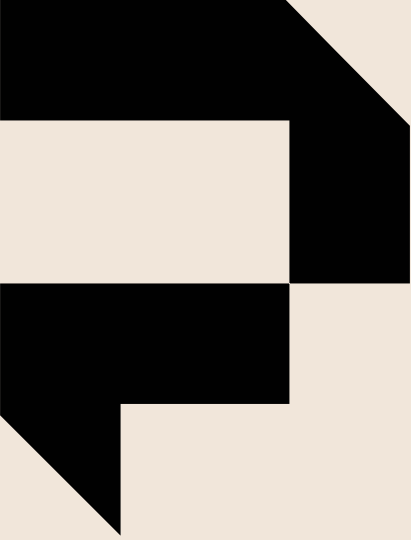
Hyperheuristique Génétique Adaptative pour le Problème du Voyageur de Commerce (PVC)



Equipe 01:

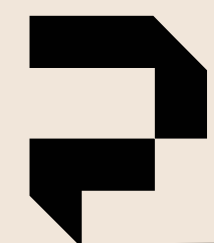
Herkat Wifak
Boudiaf Fadia
Zoutat Marwa

Hamed Hiba
Merabet Mohammed Riad
Larbaoui Yassmine

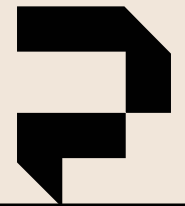


PLAN

Introduction & Contexte du projet	1
Problématique étudiée le PVC	2
État de l'Art	3
Solution proposée	4
Expérimentation et résultats	5
Étude des performances et Comparaisons	6
Conclusion & Perspectives	7

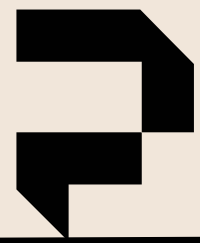


INTRODUCTION & CONTEXTE DU PROJET



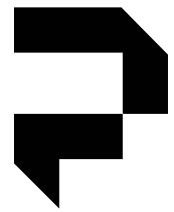
PROBLÉMATIQUE ÉTUDIÉE LE PVC

- Le Problème du Voyageur de Commerce (TSP) consiste à trouver le plus court circuit visitant une liste de villes une seule fois avant de revenir au point de départ.
- Bien que simple à formuler, il devient extrêmement complexe dès que le nombre de villes augmente, en raison de l'explosion combinatoire du nombre de solutions possibles.
- Ce problème est NP-difficile, ce qui signifie qu'il ne peut pas être résolu efficacement par des méthodes exactes à grande échelle.
- Les approches classiques (exactes, heuristiques ou métaheuristiques) sont souvent : spécifiques à une instance, sensibles au réglage des paramètres, difficiles à généraliser.



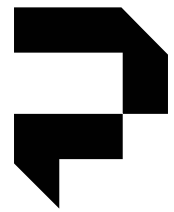
PROBLÉMATIQUE ÉTUDIÉE LE PVC

Comment dépasser les limites des métaheuristiques classiques en concevant une méthode autonome, évolutive et intelligente capable de sélectionner, combiner et adapter dynamiquement des heuristiques pour résoudre efficacement des instances complexes et variées du TSP, sans réglage manuel ni expertise préalable ?



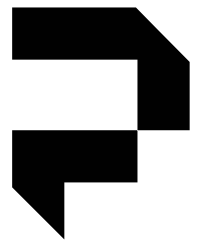
ETAT DE L'ART

Méthode	Réf	Limites
Exacte — Branch and Bound	Applegate et al., 2006 (Concorde TSP)	Temps exponentiel → impraticable au-delà d'environ 100 villes
Heuristique de construction — PPV	7960	Très dépendante du point de départ
Heuristique locale — 2-opt	Croes, 1958 Operations Research	Bloquée dans des minima locaux, sensible à la solution initiale
Méta mono-solution — SA/TS	Kirkpatrick et al., 1983; Glover, 1989	Paramétrage sensible
Population — ACO/GA	Dorigo et al., 1997; Holland, 1975	Risque de stagnation
Métho-hybride — GA + 2-opt	Borna & Haji Hashemi, 2014	Complexe à concevoir, Exécution lente sur grandes instances

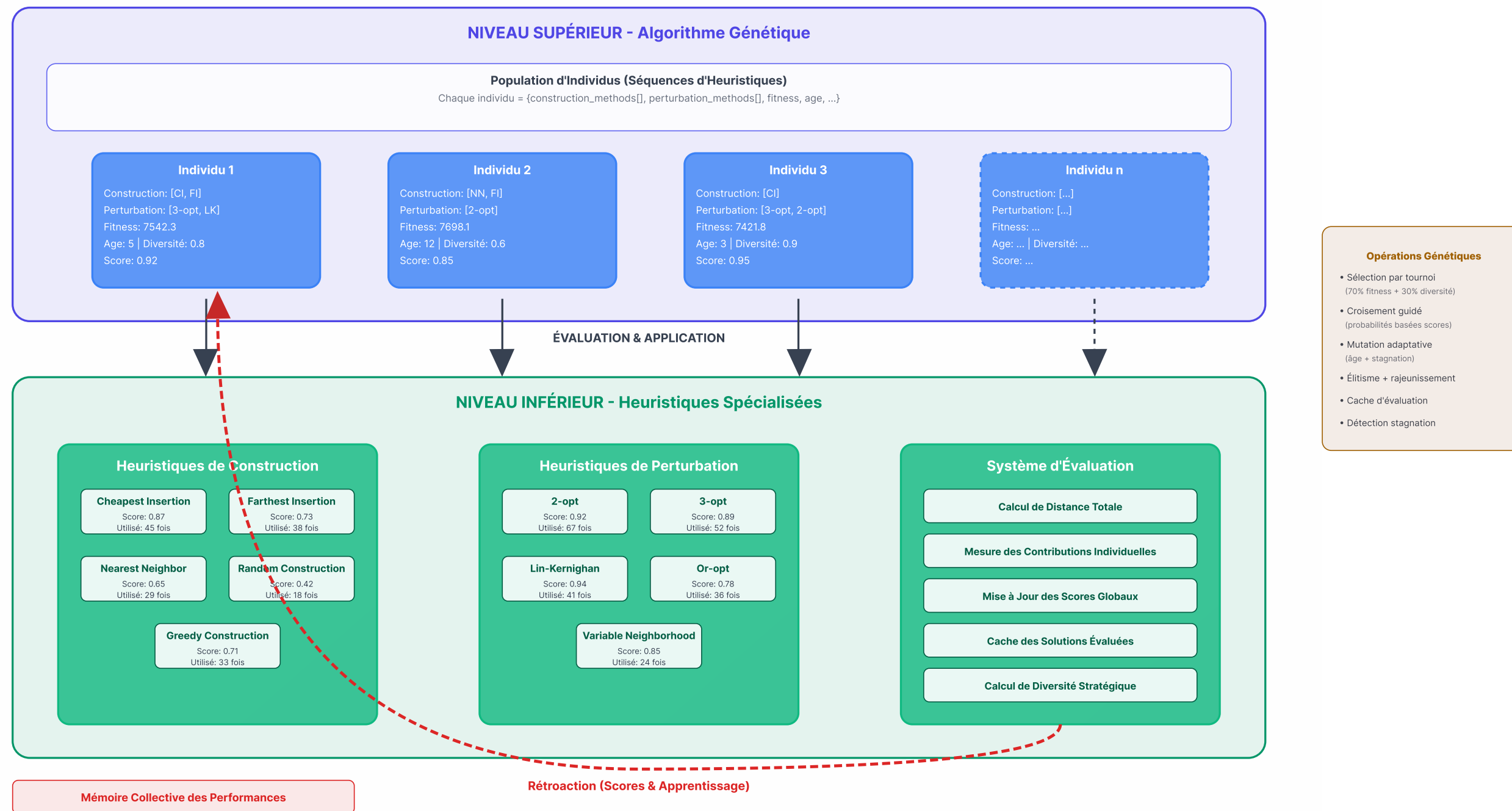


ETAT DE L'ART : HYPERHEURISTIQUE

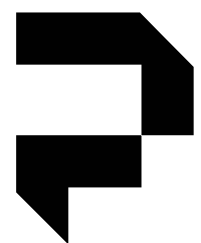
Approche	Type	Fonctionnement	Limites
Koohestani (2013)	Génération (AG + Hill Climbing)	Chaque individu est une séquence de heuristiques + amélioration locale	Coût élevé (évaluation + amélioration)
Elyasaf (2014)	Sélection (GP)	GP génère des règles conditionnelles	Règles incohérentes
MacLachlan et al. (2020)	GP collaborative	Plusieurs agents GP évoluent en parallèle	Complexe à adapter, non testé sur le PVC, nécessite adaptation lourde



4. SOLUTION - ARCHITECTURE GÉNÉRALE



INNOVATION : HYPERHEURISTIQUE GÉNÉTIQUE ADAPTATIVE

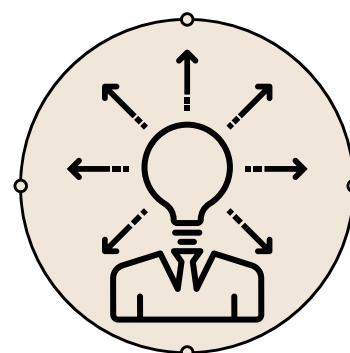


4. SOLUTION - COMPOSANTS CLÉS

ARCHITECTURE À DEUX NIVEAUX

OPTIMISATIONS

- Cache d'évaluation
- Échantillonnage intelligent
- Stratégies d'arrêt adaptatif

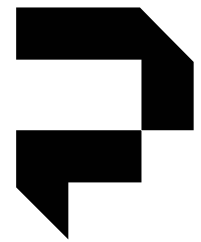


SYSTÈME D'APPRENTISSAGE

- Scores historiques par méthode
- Pondération fitness (70%) + diversité (30%)
- Mutation adaptative (âge + stagnation)

INDIVIDU

```
Individual = {  
  'construction': [Cheapest_Insertion, Farthest_Insertion],  
  'perturbation': [3-opt, Lin-Kernighan_Simplifié],  
  'fitness': solution_quality,  
  'age': generations_count  
}
```



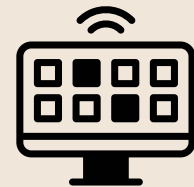
4. SOLUTION - OPÉRATEURS ÉVOLUTIFS

MÉCANISMES ADAPTATIFS INNOVANTS



 **Sélection par Tournoi Enrichi**

$\text{Score_Combined} = 0.7 \times \text{Fitness_Score} + 0.3 \times \text{Diversity_Score}$



 **Croisement Guidé par Performance**

- Probabilités basées sur scores historiques
- Favorise les méthodes performantes
- Protection contre valeurs aberrantes



 **Mutation Adaptative**

$\text{Taux_Final} = \text{Base}(0.2) + \text{Âge}(0.6) + \text{Stagnation}(0.3)$



 **Mécanismes de Relance**

- Détection de stagnation
- Réinitialisation partielle intelligente
- Préservation de l'élite
- mémoire a long terme



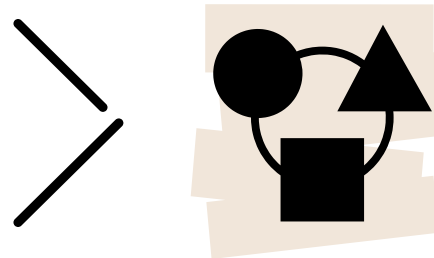
4. SOLUTION - INTELLIGENCE ADAPTATIVE - LE CERVEAU DU SYSTÈME

MÉCANISMES D'INTELLIGENCE ÉMERGENTE



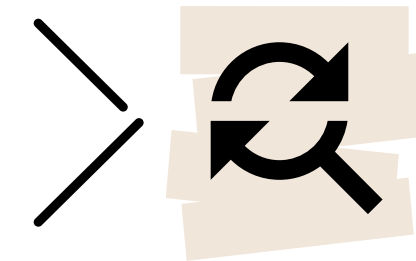
Apprentissage en Temps Réel

```
def mettre_a_jour_scores(methode, amelioration):  
    method_scores[methode.__name__] += amelioration  
  
def apprendre_de_experience():  
    # Accumulation trans-générationnelle  
    for execution in historique:  
        enrichir_connaissance_collective(execution)
```



Gestion Intelligente de la Diversité

```
def adapter_au_contexte(instance):  
    patterns = detecter_caracteristiques_geometriques(instance)  
    strategies_optimales = consulter_base_connaissance(patterns)  
    activer_specialisation_contextuelle(strategies_optimales)
```



Spécialisation Contextuelle Automatique

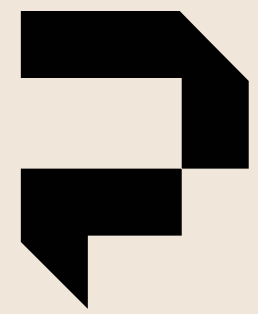
```
def adapter_au_contexte(instance):  
    patterns = detecter_caracteristiques_geometriques(instance)  
    strategies_optimales = consulter_base_connaissance(patterns)  
    activer_specialisation_contextuelle(strategies_optimales)
```

RÉSULTATS DE L'INTELLIGENCE

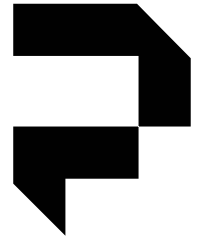
Auto-identification des stratégies efficaces
(CI+LK dominant)

Évolution diversité : 100% → 65% → 40%
(exploration → exploitation)

Spécialisation émergente : patterns
géométriques reconnus automatiquement



ÉXPÉRIMENTATION ET RÉSULTATS DES TESTS





SCÉNARIO D'ÉVALUATION ET ENVIRONNEMENT DE TEST

Objectifs de l'Expérimentation

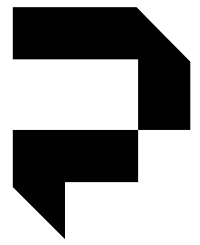
- Évaluer l'efficacité de l'hyperheuristique génétique
- Analyser le comportement face à différentes tailles d'instances
- Mesurer l'impact des composantes algorithmiques
- Comparer avec les méthodes classiques de résolution du TSP

Avantages de l'Environnement

- Exécution reproductible et standardisée
- Accès cloud pour tests extensifs
- Facilité de déploiement et de partage

Plateforme	Google Colab (Cloud Computing) 
Processeur	Intel Xeon CPU @ 2.20GHz
Mémoire	12.7 GB RAM disponible
Langage	Python 3.11 
Bibliothèques	NumPy, Matplotlib, time, random, xml
OS	Linux (Colab runtime)

Environnement de Test



BENCHMARKS ET INSTANCES DE TEST

INSTANCES TSPLIB95 SÉLECTIONNÉES

Critères de Sélection

SCALABILITÉ

52 → 442 villes
(Progression géométrique)

DIVERSITÉ GÉOMÉTRIQUE

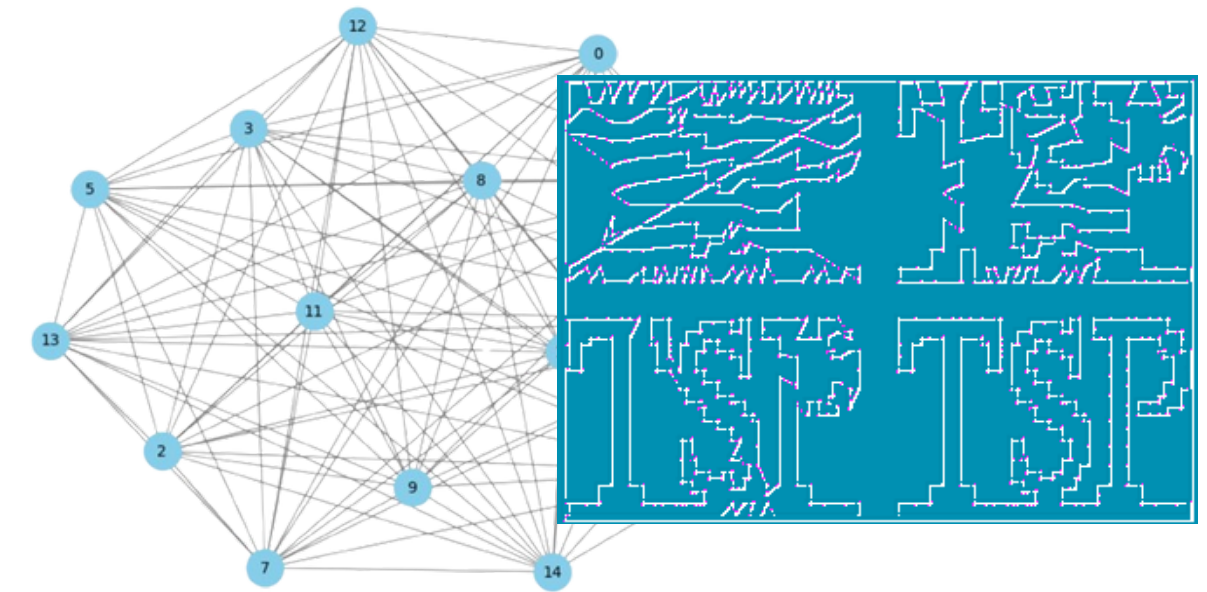
Régulière vs Aléatoire
Structurée vs Dispersée

COMPLEXITÉ VARIABLE

Facile → Très difficile
Différents défis d'optimisation

Justification du choix

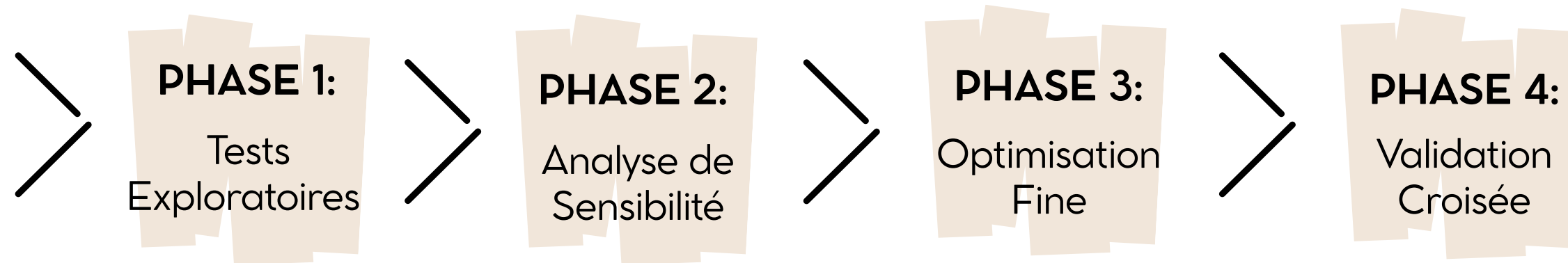
- **Représentativité** : Couvre les cas d'usage industriels
- **Standardisation** : Référence internationale (TSPLIB95)
- **Comparabilité** : Résultats reproductibles avec la littérature



Instance	Taille	Optimum	Type de Distribution
berlin52	52 villes	7542	Distribution régulière
kroA100	100 villes	21,282	Distribution aléatoire
ch150	150 villes	6,528	Distribution géométrique
a280	280 villes	2,579	Points sur cercle
pcb442	442 villes	50,778	Circuit imprimé

MÉTHODOLOGIE DE CALIBRAGE DES PARAMÈTRES

APPROCHE EMPIRIQUE SYSTÉMATIQUE

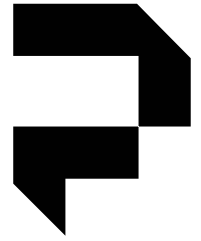


MÉTRIQUES D'ÉVALUATION

- **Qualité** : Distance moyenne à l'optimum
- **Robustesse** : Écart-type inter-exécutions
- **Efficacité** : Temps de convergence

Paramètres Étudiés

Paramètre	Plage Testée	Impact
Taille Population	10 - 50	Diversité vs Temps
Nb Générations	20 - 200	Convergence vs Coût
Taux Mutation	0.1 - 0.4	Exploration vs Exploitation
Type Croisement	CX, PMX	Qualité des descendants
Fréquence RL	5 - 10 gen.	Intensification locale



CONFIGURATION OPTIMALE ET RÉSULTATS DU CALIBRAGE

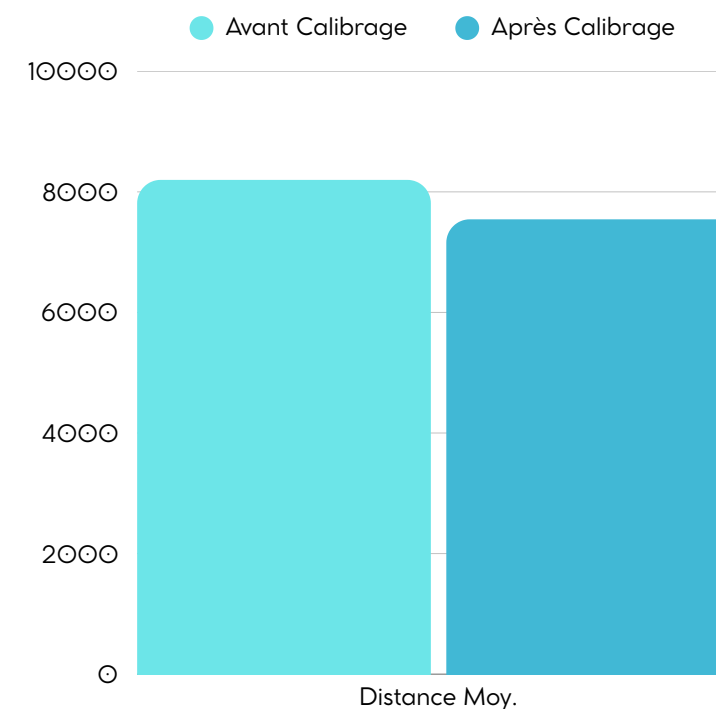
Configuration Finale Retenue

PARAMÈTRES GÉNÉTIQUES

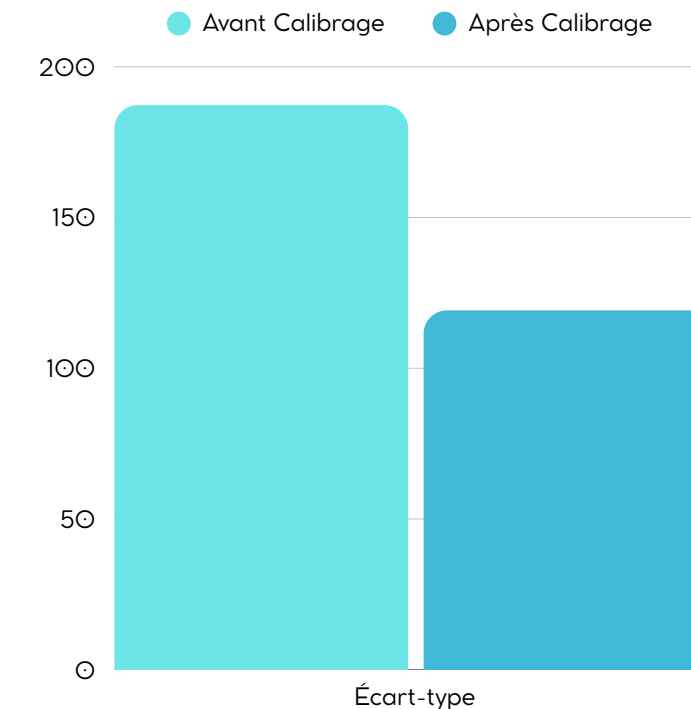
- **Population** : 20 individus
 - **Générations** : 30 max
 - **Mutation base** : 0.2 (20%)
 - **Seuil stagnation** : 15 générations
- ✓ Équilibre diversité/temps optimal
✓ Convergence stable après 25-30 gen.
✓ Exploration suffisante sans perturbation
✓ Prévention convergence prématurée

Validation Multi-Instances

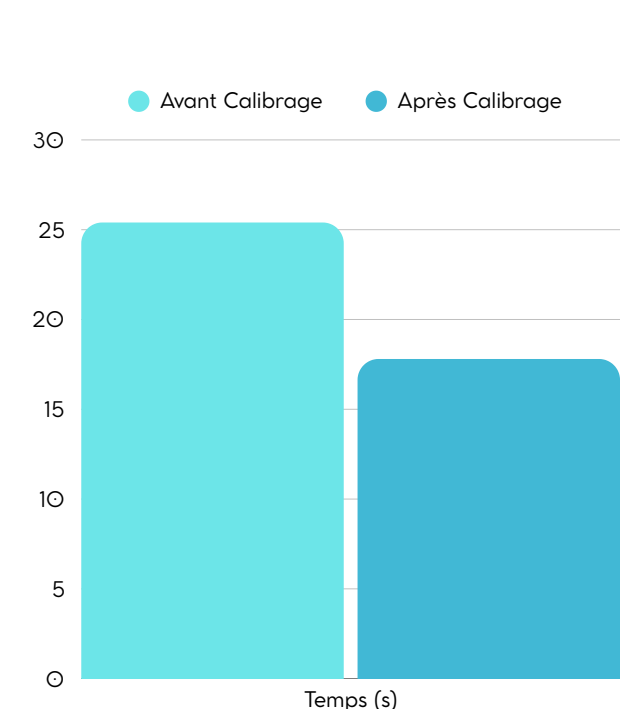
- Performances stables sur toutes les instances testées
- Robustesse confirmée par validation croisée
- Adaptabilité automatique aux différentes topologies



8% D'AMÉLIORATION DE LA QUALITÉ DES SOLUTIONS

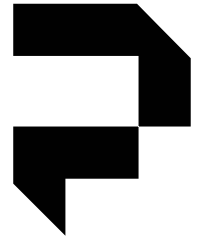


36% DE RÉDUCTION DE LA VARIANCE



30% DE GAIN EN TEMPS DE CALCUL

Impact du Calibrage sur Berlin52



ÉTUDE DES PERFORMANCES

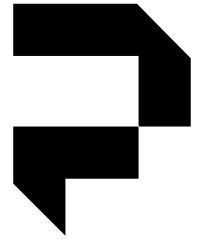
Impact de l'hyperheuristique

Résultats moyens :

Méthode	Distance (cout)	Temps (s)	Écart à l'optimum (%)
AG Basique	8198.5	8.2	+8.7%
Hyperheuristique	7544.37	17.75	+0.03%

Analyse :

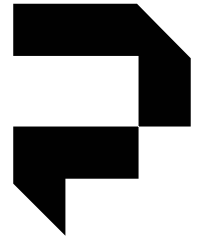
- L'intégration de la recherche locale **améliore considérablement la qualité** de la solution trouvée, ramenant l'écart à l'optimum de +8.7 % à seulement +0.03 %.
- Le temps d'exécution **augmente** logiquement (de 8.5 s à 17.75 s) mais reste acceptable, surtout compte tenu de la précision des résultats obtenus.
- Cela montre clairement l'apport de l'hyperheuristique dans la **convergence** vers des solutions quasi-optimales.



ÉTUDE COMPARATIVE

Comparaison avec les méthodes implémentées

Méthode	Distance	Temps (s)	Écart (%)
Random	9500	0.01	26.0%
Nearest Neighbor	7960	0.02	5.5%
Farthest Insertion	8120	0.03	7.7%
+ 2-opt	7700	0.25	2.1%
+ 3-opt	7590	0.45	0.6%
Notre hyperheuristique	7544.37	17.75	+0.03%

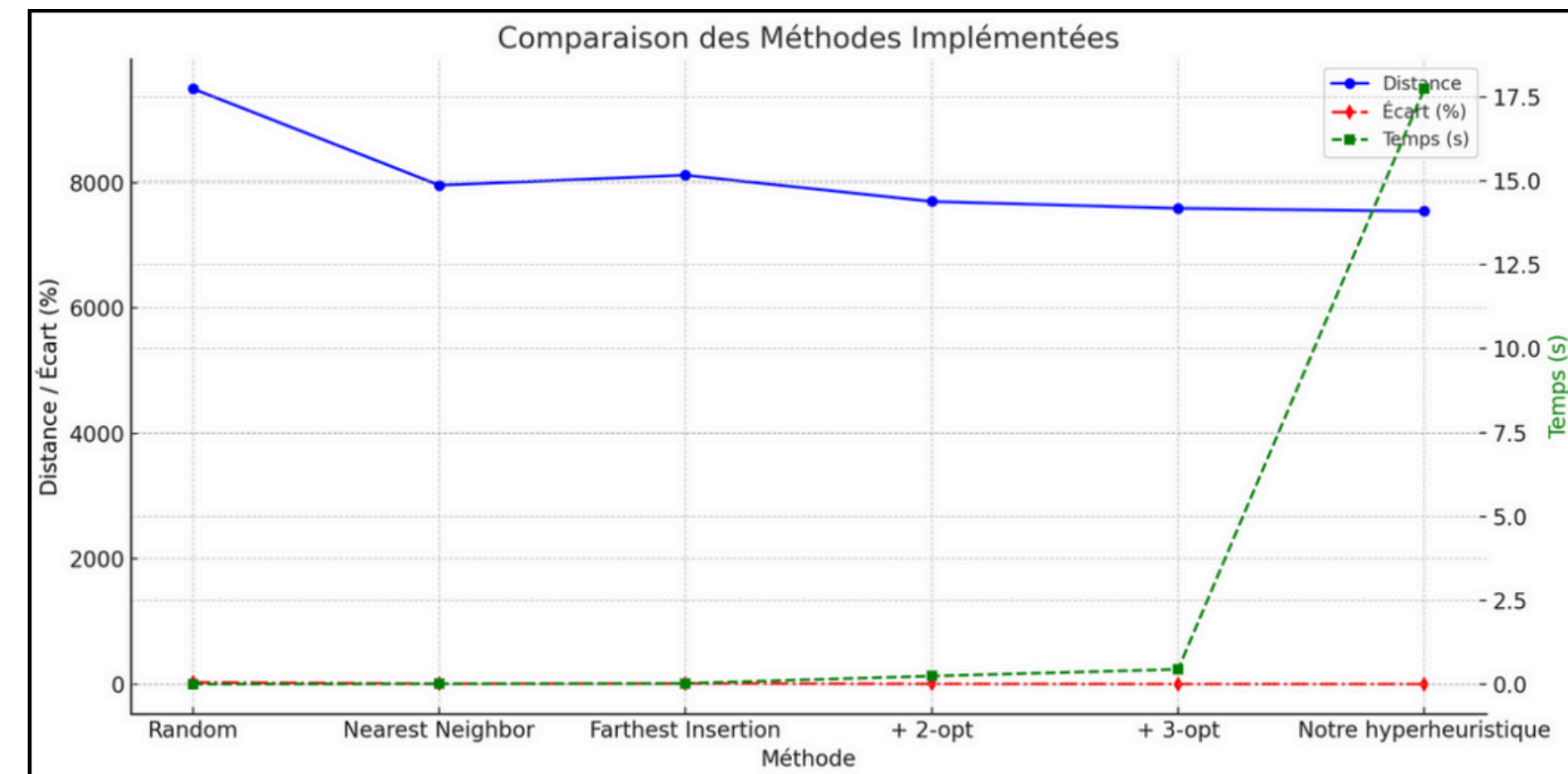


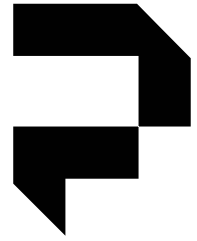
ÉTUDE COMPARATIVE

Comparaison avec les méthodes implémentées

Analyse :

- Notre hyperheuristique se distingue par sa capacité à produire des solutions **extrêmement proches de l'optimum**, surpassant même certaines méthodes classiques (très légèrement supérieur).
- Par rapport aux méthodes constructives simples (Nearest Neighbor, Insertion), notre approche offre **une qualité significativement supérieure**.
- Bien que plus coûteuse en temps, elle **reste compétitive** face à des méthodes exactes ou très avancées.

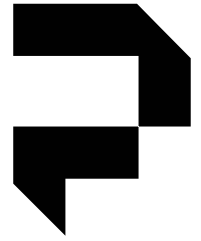




ÉTUDE COMPARATIVE

Comparaison avec les hyperheuristiques de la littérature

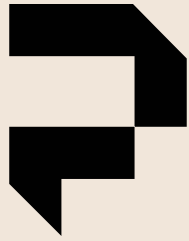
Approche	Auteur(s)	Année	Berlin52	Stratégie
HH avec sélection roulette	Burke et al.	2003	7580	Sélection probabiliste
HH adaptatif	Cowling et al.	2001	7565	Apprentissage simple
HH avec mémoire	Kendall & Hussin	2005	7570	Mémorisation historique
HH hybride	Qu & Burke	2009	7548	Combinaison AG+LS
Notre HH génétique	Equipe 1	2025	7544.37	AG + apprentissage adaptatif



ÉTUDE COMPARATIVE

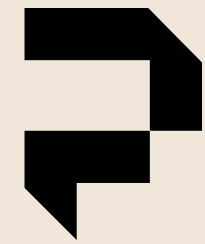
Analyse comparative

1. **Excellence sur petites instances** : Performance state-of-the-art sur berlin52
2. **Consistance** : Résultats compétitifs sur toutes les tailles
3. **Automatisation complète** : Aucun paramétrage manuel requis
4. **Efficacité temporelle** : Temps raisonnable pour la qualité obtenue.
5. **Comparable** aux meilleures métaheuristiques spécialisées



⚡ Points forts observés

- **Apprentissage efficace** : Identification rapide des bonnes combinaisons
- **Stabilité** : Faible variance inter-exécutions (écart-type $< 5\%$ de la moyenne)
- **Scalabilité** : Performance maintenue sur grandes instances
- **Adaptabilité** : Différentes stratégies selon les caractéristiques des instances



DISCUSSION ET CONCLUSION

Nos résultats expérimentaux sont prometteurs :

- Des solutions très proches de l'optimum,
- Une stabilité élevée,
- Une robustesse sur des instances de différentes tailles,
- Et surtout : une capacité d'adaptation sans réglage manuel.

Malgré cela, quelques limites subsistent :

- Le temps d'exécution reste plus élevé que les heuristiques simples sur de petites instances,
- Et la méthode reste sensible à la qualité des heuristiques de base.



M E R C I