



Algorithmes mémétiques pour l'optimisation combinatoire

Marc Schoenauer

Projet TAO – INRIA Futurs et LRI

<http://tao.lri.fr/>

Ecole d'été JET, le 30 mai 2007



Optimisation combinatoire

- Problèmes NP-difficiles Taille de problème exponentielle
- Typiquement, on cherche une permutation de $[1, N]$
- **Exemples:** le Voyageur de Commerce (TSP), la coloration de graphes, les problèmes de satisfaction de contraintes, . . .
et de nombreux problèmes réels

Les AEs standards ne sont **pas compétitifs**

Handbook of Evolutionary Computation 94-97

Un article visionnaire :

J. J. Grefenstette. Incorporating problem specific knowledge in genetic algorithms. In Davis L., editor, *Genetic Algorithms and Simulated Annealing*, pages 42-60. Morgan Kaufmann, (1987).



Exemple historique

Le TSP évolutionnaire

- Croisement spécifique
Grefenstette, 87 (20%), Suh & van Gucht, 89 (1.7 puis 1.3 %, 100 villes)
- *Genetic Edge Recombination*
Matthias & Whitley, 92 (3.8%), Tang & Leung, 94 (0.95%, 532 villes)
- Changements de représentation
Radcliffe, 91-05
- Lamarckisme via techniques de RO
Mülenbein, 91, modifié par Ulder, 91 (0.17%, 532 villes)
- Genetic Local Search
Merz & Freisleben, 96 (0%, 15 000 villes)
- Inver-Over operator
and EAX crossover
Tao & Michalewicz, 98
Tagata, 97-06 (0%, 24 000 villes)



Plan

- Introduction le problème du voyageur de commerce – TSP
- Générique vs spécifique
- Opérateurs spécifiques sur le TSP
- Hybridation : algorithmes mémétiques
- Représentations indirectes
- Quelques exemples pas le TSP
 - Horaires de trains
 - Planification temporelle



Évolution de permutations

Croisements

- **UOX** Uniform Order-based Crossover Davis, 85
uniformément
Copier certaines positions du parent 1
Compléter dans l'ordre du parent 2
- **PMX** Partially Matched Crossover Goldberg & Lingle, 85
Choisir une sous-chaîne du parent 1
Modifier minimalement parent 2 pour le faire apparaître

Mutations

- **Transposition** : échanger 2 valeurs. Éventuellement avec un rayon limité à répéter N fois Radcliffe's BMM
- **Inversion** : choisir 2 positions et inverser la sous-chaîne
- **Scramble** : choisir 2 positions et mélanger la sous-chaîne



Autres représentations pour le TSP

Radcliffe, 91-95

- **Permutation** : (1234)
Dégénérescence : choix du pt de départ (facile à réparer),
et du sens (en général négligé).
Redondance: la dernière ville.
- **Arc non orientés** : (12,23,34,14).
Pas de dégénérescence, redondance : le dernier.
- **Arc orientés** : ($\vec{12}$, $\vec{23}$, $\vec{34}$, $\vec{41}$)
Dégénérescence : sens de parcours, redondance : le dernier.
- **Coins** : (1(24),2(13),3(24),4(13))
Pas de dégénérescence, redondance importante.



Représentations pour le TSP (2)

Opérateurs indépendants de la représentation

Croisements

- Échange de “gènes” 1-pt, uniforme, ...
- Compléments aléatoire en respectant les contraintes
- ... ou réparation

Binomial Minimal Mutation

- Répéter $B(n, p)$ fois une mutation minimale e.g. transposition

Corrélation

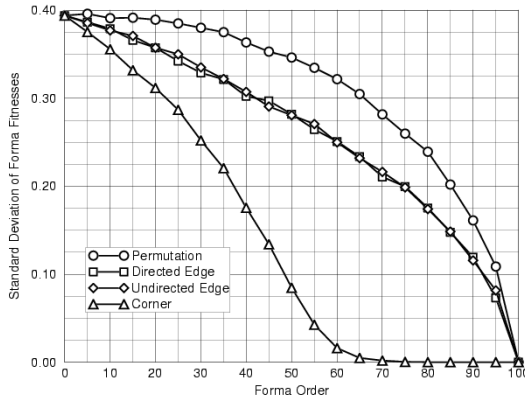
- Schémas: avoir certains gènes communs
- Variance de la fitness fn de la précision

Quotienter par la redondance (coins)

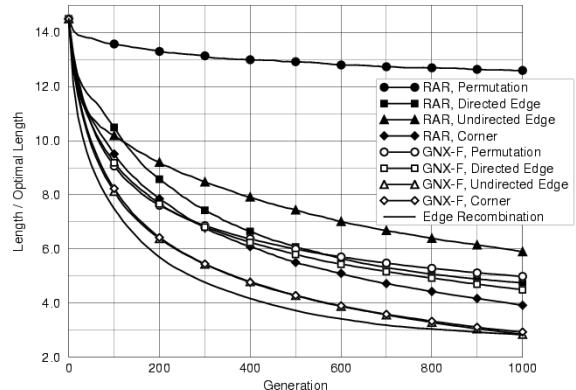


Corrélation

Résultats expérimentaux



Ecart-type de la performance en fonction de la précision des formae



Performances des algorithmes génétiques correspondants

Conclusions

- Bon accord Variance – Performance
- Genetic Edge Recombination plus performant



Plan

- Introduction le problème du voyageur de commerce – TSP
- Générique vs spécifique
- **Opérateurs spécifiques** sur le TSP
- Hybridation : algorithmes mémétiques
- Représentations indirectes
- Quelques exemples pas le TSP
 - Horaires de trains
 - Planification temporelle



Genetic Edge Recombination

Whitley et al., 89

- Construire la carte des liens

Pour chaque ville, liste des villes liées dans au moins 1 parent

- Choisir une ville de départ

Random, ou d'après 3

1. Enlever la ville courante de la carte des liens

2. Si la ville courante n'a plus de ville dans sa liste,

- S'il reste des villes non visitées, en choisir une au hasard.

Goto 1

- Sinon, STOP

3. Choisir dans la liste de la ville courante celle qui reste liée au minimum de villes

Random sur les ex-aequo

Goto 1

Le tour est complet, et possède de nombreux arcs des parents



Inver-over operator

Tao & Michalewicz, 98

Algorithme :

- Pour chaque $T = (v_0, \dots, v_n)$ dans la population
- Soit v choisie uniformément dans T
- Tant que $v' \neq Succ_T(v)$
 - Avec proba p
 - Choisir $v' \neq v$ uniformément dans T
 - Inverser $v - v'$
 - Avec proba $1 - p$
 - Choisir T' uniformément dans la population
 - $v' = Succ_{T'}(v)$
 - Inverser $v - v'$
- Remplacer T par son enfant s'il est meilleur

typiquement 0.02

dans T'
dans T !



Inver-over operator

Performances

Avantages

- 3 paramètres seulement
 - Taille population 100
 - Probabilité d'inversion 0.02
 - Critère d'arrêt 10 boucles sans amélioration
- Une "évaluation" = 4 opérations 2 arrêtes seulement sont modifiées

Résultats

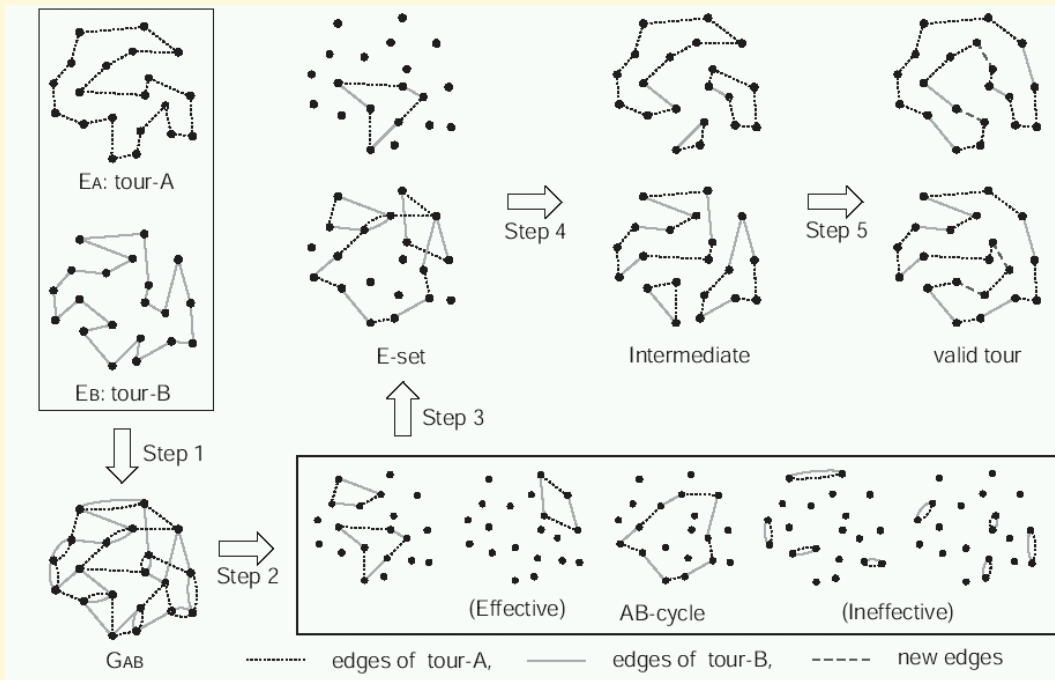
- Très rapide et très performant pour ≤ 500 villes
- Rapide et performant jusqu'à 10000 villes
- Seulement 1000 fois plus lent que Lin-Kernighan

Mais n'a pas tenu ses promesses ...



Edge Assembly Crossover

Y. Nagata, 97-06



Y. Nagata, PPSN 2006



Edge Assembly Crossover (2)

Étape 1 : Points de départs aléatoire, arêtes de A et B en alternance

Étape 4 : Ôter du parent les arrêtes des sous-tours qui lui appartiennent

Étape 5 : Algorithme glouton par ordre de taille de cycle croissante

Stratégies pour l'étape 3

- **Rand** : choix aléatoires uniformes de AB-cycles
- **1AB** : un unique tour est utilisé
- **Block** :
 - Choisir le plus grand AB-cycle C_0
 - Construite la solution intermédiaire
 - Pour chaque sous-tour $k > 1$, choisir un AB-cycle C_k qui le touche
 - Refaire l'étape 3 avec tous les C_i

2006

étape 4



Edge Assembly Crossover : sélection

Préservation de la diversité : Sélection environnementale particulière
Les N parents sont remplacés par N enfants

- Faire N fois
- Pour chaque paire de parents
- Créer N_e enfants par EAX
- Le meilleur des N_e enfants et des 2 parents survit



Edge Assembly Crossover : résultats

- Avec 1AB puis Rand si stagnation Nagata & Kobayashi, 1997
 - $N = 300$, $N_e = 30$
 - 0% sur 3038 villes
 - en 2h30 (Intel 200MHz)
 - mais perd en efficacité pour des plus grandes tailles
- Avec 1AB puis Block Nagata, 2006
 - $N = 300$, $N_e = 30$
 - 0% sur 24978 villes, en 45mn (Itanium 1.3 GHz)
 - Meilleurs résultats connus, pour 8079 (2mn), 14773 (6mn) et 33708 (1h25) villes !



Plan

- Introduction le problème du voyageur de commercer – TSP
- Générique vs spécifique
- Opérateurs spécifiques sur le TSP
- **Hybridation : algorithmes mémétiques**
- Représentations indirectes
- Quelques exemples pas le TSP
 - Horaires de trains
 - Planification temporelle



Algorithme mémétique pour le TSP

Genetic Local Search – Freisleben & Merz - 96

- Initialisation gloutonne Départ aléatoire, puis ville la plus proche
- Optimisation locale (Lin-Kernighan, 3-opt)
Algorithme *mémétique* (population d'optima locaux)
- *Distance Preserving Crossover*

$$d(enfant, parent1) \approx d(enfant, parent2) \approx d(parent1, parent2)$$
- Mutation “4-arcs” Pour quitter l'optimum local

Au niveau de l'état de l'art (jusqu'à 10000-15000 villes)

Mais , sur f11400:	Plus proche voisin	20%	12s
	Lin-Kernighan	2-3%	68s
	<i>Genetic Local Search</i>	0%	57h



Algorithmes mémétiques

Autres succès

- **Coloration de graphes** Dorne & Hao 97-98
 - Croisement spécifique idem ER, EAX, ...
 - Mutation inspiré de la Recherche Tabou
 - Meilleurs résultats connus (500 à 1000 noeuds).
- **Assignment quadratique** Merz & Freisleben 1997
 - *Distance Preserving Crossover*
 - Mutation par inversion
 - Algorithme mémétique – un 2-opt dégradé
 - Meilleurs résultats connus (30-256 emplacements).



Heuristique constructive

- E.g. gloutonne pour GLS

Inoculation de bonnes solutions

- Obtenues par d'autres méthodes
- Faut-il les perturber ?
- Biais la recherche
 - Radcliffe et Surry, 94 Dimensionnement d'un réseau de gaz
Meilleur en moyenne à partir d'une population inoculée
Meilleure solution à partir d'une population aléatoire
 - Sourd et MS, 98 Renumérotation de maillages
Semet et MS, 2005 Re-ordonnancement de trains
Meilleurs résultats à partir d'une population biaisée



Algorithmes mémétiques

Optimisation locale

Quand, qui, comment ?

- Comme opérateur de mutation
 - à certains individus seulement
 - éventuellement partiellement
 - Dépend du nombre et de la répartition des optima locaux
- Systématiquement à chaque apparition d'un nouvel individu
 - On travaille dans l'espace des optima locaux

Lamarck vs Baldwin

- **Lamarck** : le résultat de l'optimisation locale remplace l'individu de départ
- **Baldwin** : l'individu est inchangé, mais sa fitness est celle du résultat de l'optimisation
Hinton & Nowlan, 87



Algorithmes mémétiques

Conclusion

Opérateurs de variation

- Opérateurs locaux différents des opérateurs de l'optimiseur local
- Opérateurs spécifiques e.g. ER, EAX, ...
Nécessite de la connaissance experte

Conclusion

- Le meilleur de plusieurs mondes
- Gains de plusieurs ordres de grandeurs en qualité et en coût

État de l'art sur de nombreux benchmarks, **mais** ...



Plan

- Introduction le problème du voyageur de commercer – TSP
- Générique vs spécifique
- Opérateurs spécifiques sur le TSP
- Hybridation : algorithmes mémétiques
- **Représentations indirectes**
- Quelques exemples pas le TSP
 - Horaires de trains
 - Planification temporelle



Hybridation ... dès la représentation

Les problèmes réels (emplois du temps, tournées, planification de tâches, ...) sont **fortement contraints** :

écrire un opérateur spécifique \equiv résoudre un problème d'optimisation

Représentation directe : génotype \equiv phénotype

- Pour chaque {salle, livreur, machine}
- liste des {cours, clients, tâche} avec horaire

Les contraintes

- Dans les opérateurs de variation
- A posteriori, par réparation

Problème ... d'optimisation !

Peut biaiser la recherche



Hybridation ... dès la représentation (2)

Représentation indirecte

On dispose de règles heuristiques (e.g. “scheduler” glouton)
qui respectent les contraintes

- **Génotype** = permutation des divers {cours, clients, tâche}
- **Phénotype** = résultat de l’application des règles au génotype
- + Opérateurs “classiques” sur un espace de permutations
- + Choix de règles → connaissance du domaine
- Morphogenèse coûteuse
- Causalité faible



Plan

- Introduction le problème du voyageur de commerce – TSP
- Générique vs spécifique
- Opérateurs spécifiques sur le TSP
- Hybridation : algorithmes mémétiques
- Représentations indirectes
- Quelques exemples pas le TSP
 - Horaires de trains
 - Planification temporelle



“Vraies” applications

- Rotations d’équipages Levine 97
- Emplois du temps Paechter & al. 98
- Ordonnancement d’ateliers Hart & Ross 98, 2000
- Assignation de fréquences Crisan & al. 97, Dorne 98-99
- Contrôle du trafic aérien Alliot, Delahaye & al. 95-00
- Réseaux de téléphonie mobile ARNO project, 00
- Routage de réseaux complexes Knowles & Corne, 98-00
- Ramassage/distribution de colis Chronopost International, 2002

Un exemple: Re-ordonnancement de trains

SNCF, 2006



(Re)ordonnement de trains

Y. Semet 2006, coll. SNCF

- Re-planifier les horaires des trains après un incident
- En respectant les contraintes
 - Sécurité E spacements temporels et géométriques
 - Correspondances
 - ...
- Degrés de liberté : horaire de chaque train dans chaque gare

État de l'art : CPLEX (c) Ilog

Programmation Linéaire en Nombres Entiers

Des millions de variables dans les cas réalistes



Représentation

Génotype : une permutation de tous les trains à re-programmer

Ce n'est pas un TSP !

Morphogenèse : un scheduler

- Glouton Place chaque train le plus tôt possible
- mais en respectant les contraintes Connaissances métier

Question : est-ce que toutes les solutions sont représentables ?

Réponse : Certainement pas ...



Opérateurs de variation

Recherche parmi les opérateurs usuels

Par essais-erreurs

- Pas de croisement utile

Uniform Order-based Crossover, Partially Matched Crossover, ...

- Aucune mutation sophistiquée

inversion, ...

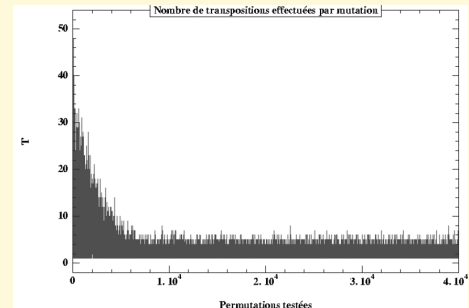
- MM = inversion de 2 trains

- Avec un "rayon" limité pour limiter les grands changements

Application de la mutation

- Suivant une loi binomiale
- de moyenne T
- décroissante

e.g. de 30 à 3





Initialisation

- Les permutations aléatoires ont une performance lamentable
 - C'est normal, mais ... on connaît une solution pas trop mauvaise
- Les horaires avant incident

Procédure

- Reconstruire le génotype de la bonne solution connue
- Car on ne connaît que son phénotype
- Initialiser la population avec des perturbations de cette solution
 - Mass Mutation : la bonne solution est mutée $k = 3$ fois
 - 3 couches, sur lesquelles on fait 0, 10 et 500 mutations



Résultats

Sur des instances réelles (incident simulé !)

541 trains en Île de France

Conditions expérimentales

- (10+70)-ES
- Init: Mass Mutation ou 3 couches
- T (nombre d'inversions de la mutation) fixe ou décroissant
- Test de Wilcoxon non signé à 99% de confiance

Tendances

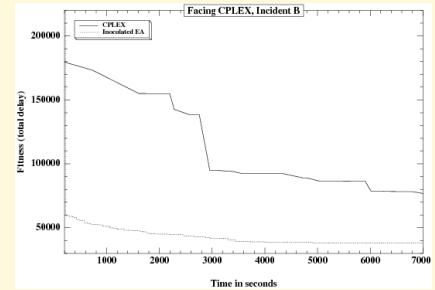
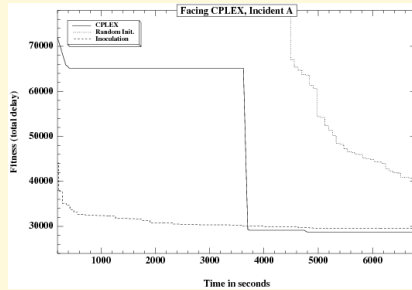
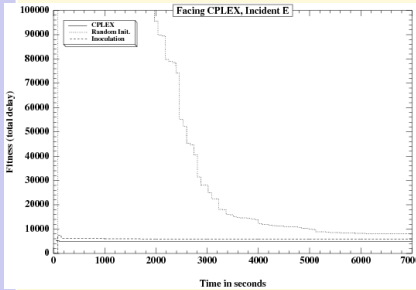
- A T fixé, MM est le meilleur choix
- Pour T décroissant, MM et 3-couches sont compétitifs



EA vs CPLEX

État de l'art déterministe : CPLEX (c) Ilog

541 trains \equiv 1M variables, 3M contraintes



Incident facile
CPLEX imbattable

Incident difficile
CPLEX rattrape
en 2 heures

Incident très difficile
CPLEX rattrape
en 6 heures

... et sur les incidents très très difficile, aucun algorithme ne progresse



Conclusion ordonnancement de trains

- Tout le savoir *métier* est dans le scheduler

Mise au point fastidieuse

- Opérateur de variation le plus efficace: le plus simple !

- ... mais savamment manié

- (10+70)-ES
- Appliqué plusieurs fois,
- Loi binomiale de moyenne décroissante

- Initialisation : clé du succès

- CPLEX est battu

mais il n'a perdu qu'une bataille :-)



Optimisation combinatoire : conclusions

- No Free Lunch Le générique ne marche pas
- Hybridation à tous les étages
 - Représentation Représentations indirectes
 - Opérateurs Problèmes d'optimisation à eux tous seuls
 - Initialisation Inoculation de bonnes solutions
 - Lamarckisme (ou Baldwinisme) OR ou autres métaheuristiques
 - Découpage séquentiel

Résultats à la hauteur de l'investissement :-)