

Optimisation multi objectif par essaim particulaire utilisant une dominance et une sélection de guide améliorées

**Gérard Dupont¹², Sébastien Adam¹, Yves Lecourtier¹,
Stephan Brunessaux²**

(1) Laboratoire d'Informatique de Traitement de l'Information et des Systèmes (LITIS)
Université de Rouen, Saint-Étienne-du-Rouvray, France

(2) EADS Defense and Systems, Information Processing and Competence Center
Val de Reuil, France

Plan de la présentation

- PSO et problèmes multi objectif
 - Formalisme : nouvelle forme de la solution
 - Conséquences de l'adaptation des PSO
- Contributions
 - Une variante de l' ε -dominance
 - Une amélioration de la gestion du guide
- Expérimentations sur des problèmes classiques
- Validation sur un cas réel
 - Présentation du problème
 - Résultats face au NSGAII

PSO et problèmes multi objectif

Optimisation multi objectif

- Minimisation de N fonctions sous M contraintes

$$\vec{f}(\vec{x}) = \{f_1(\vec{x}), \dots, f_i(\vec{x}), \dots, f_N(\vec{x})\}$$

$$avec \quad \vec{g}(\vec{x}) \geq 0$$

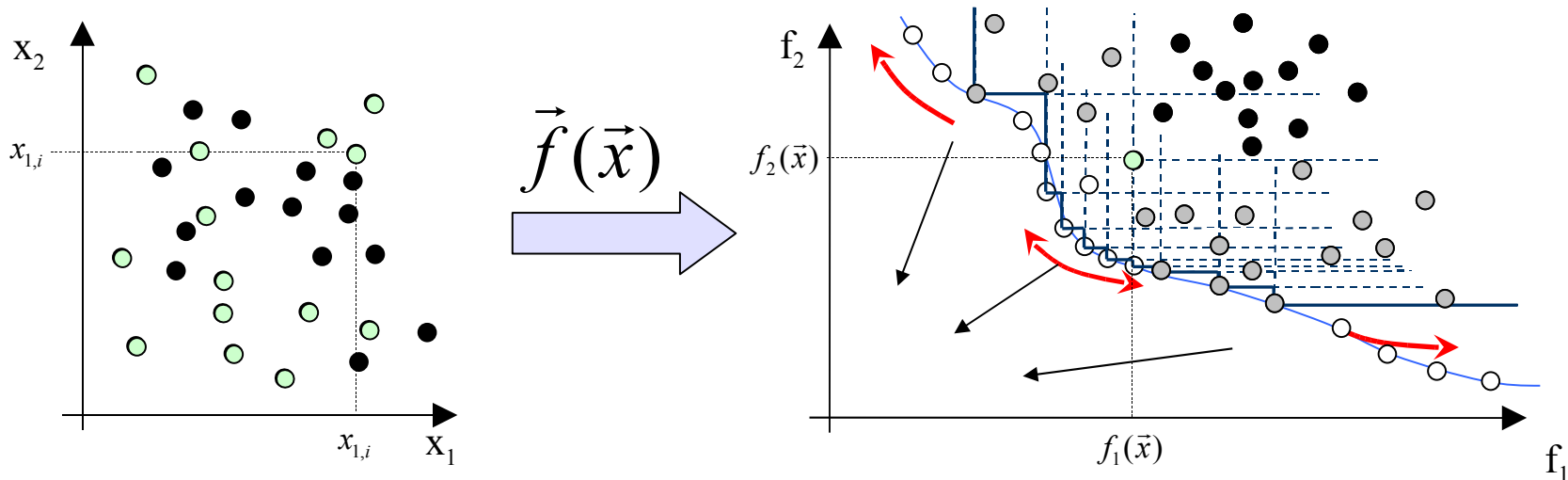
- Deux espaces de travail :

- Espace des paramètres $\rightarrow \vec{x} \in \mathbb{R}^n$
- Espace des objectifs $\rightarrow \vec{f}(\vec{x}) \in \mathbb{R}^N$

Optimisation multi objectif

- Nouvelle forme de solution : notion d'ordre partiel

Ensemble des particules « non dominées » dans l'espace des objectifs : dominance de Pareto → front de Pareto



→ Diversité et couverture de l'estimation du front de Pareto

Adaptation des essaims particulaires

$$v_{i,t+1} = \omega r_0 v_{i,t} + c_1 r_1 (p_{i,best} - x_{i,t}) + c_2 r_2 (p_{neighbor,best} - x_{i,t})$$

$$x_{i,t+1} = x_{i,t} + \chi v_{i,t+1}$$

- Disparition du concept de « meilleure » position
- Apparition d'ensemble de compromis équivalents

→ Besoin de revoir le cœur de l'algorithme

Deux conséquences successives

1. Stockage des solutions équivalentes

- Une solution commune : une archive (mémoire globale)
- Comment construire cette archive ?
- Comment éviter une augmentation incontrôlée de sa taille ?

Solutions qui optimisent la description du front par gestion du voisinage dans l'espace des objectifs :

- découpage complexe de l'espace des objectifs (clustering)
- utilisation de l' ε -dominance

Deux conséquences successives

1. Sélection de guides dans les ensembles de compromis
 - Comment choisir parmi des solutions « équivalentes » ?
 - Comment équilibrer l'exploration/exploitation ?

Solutions de sélection sur de nouveaux critères

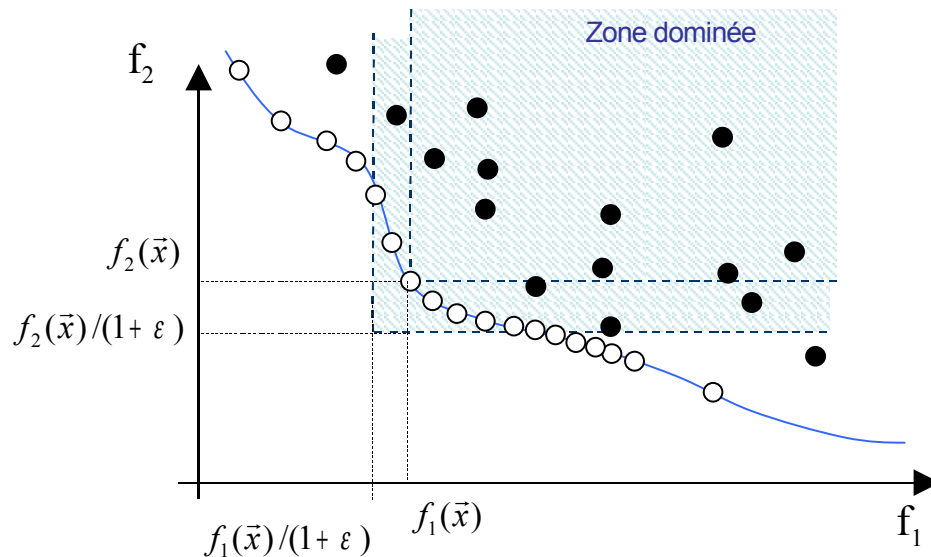
- redéfinition complexe du concept de voisinage (niching)
- favorisation de solutions pour améliorer l'estimation du front de Pareto

Contributions au MOPSO

Contributions

- Une gestion simple de voisinage dans l'espace des objectifs
 - Une amélioration de l' ϵ -dominance
- Une stratégie de gestion des guides favorisant la couverture du front
 - Sélection probabiliste basée sur une estimation de densité dans l'espace des objectifs
 - Un principe de mémorisation pour maximiser l'influence du guide

Gestion du voisinage basée sur l' ε -dominance



Principe de l' ε -dominance :

Agrandir la zone de domination pour limiter la « sur-définition » du front

→ Gestion du voisinage couplée à la notion de domination

– Avantages de l' ε -dominance :

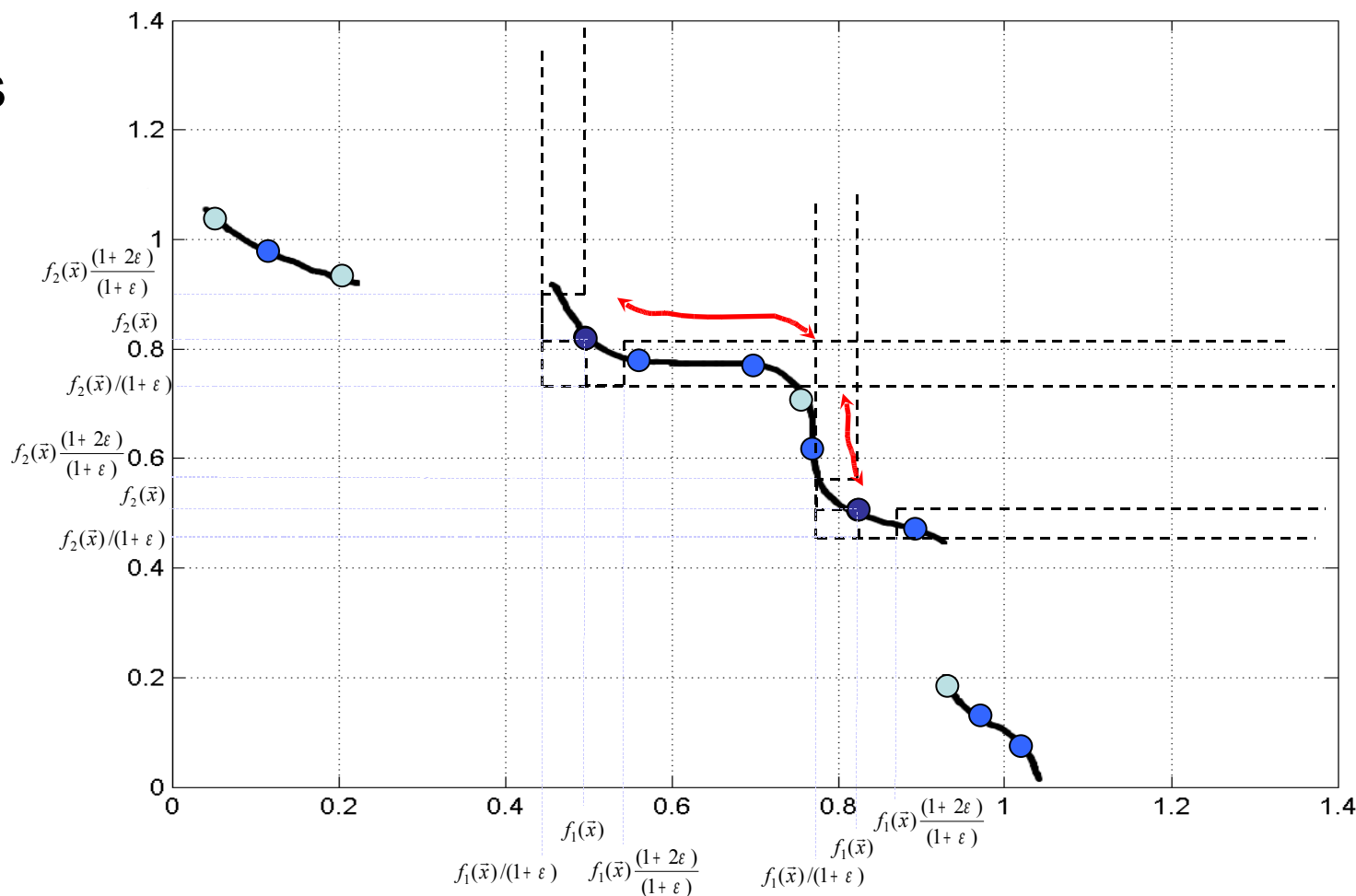
- Gestion du voisinage peu coûteuse
- Paramétrage simple (en ε relatif)

Une amélioration de l'estimation du front de Pareto

Illustration des défauts :

Une limitation de la description du front dans certaines conditions

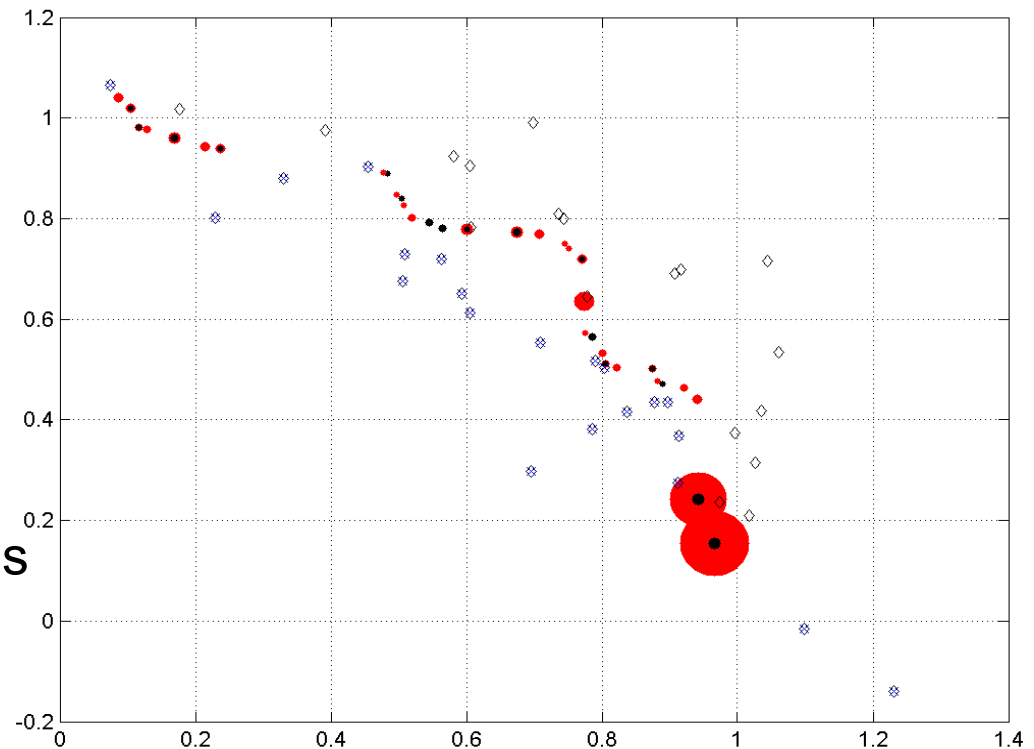
- Extremums
- Discontinuités
- Zones de faible variation d'un objectif



Une stratégie de gestion des guides

- Sélection probabiliste d'un guide dans l'archive pour chaque particule

$P(x) \sim$ estimation
de densité dans
l'espace des
objectifs favorisant
les zones peu denses
pour « combler » les trous



Valorisation de la sélection améliorée

- Problème de la volatilité du guide
 - Un guide n'influence une particule que pendant une itération

→ Correction par
une stratégie de
mémorisation
avec seuil et
contrôle

```

BEGIN
R = random(0;1)
if particle's R < memory_threshold
    AND last_guide in archive
    keep(last_guide)
else
    select(new guide in archive)
end if
END
    
```

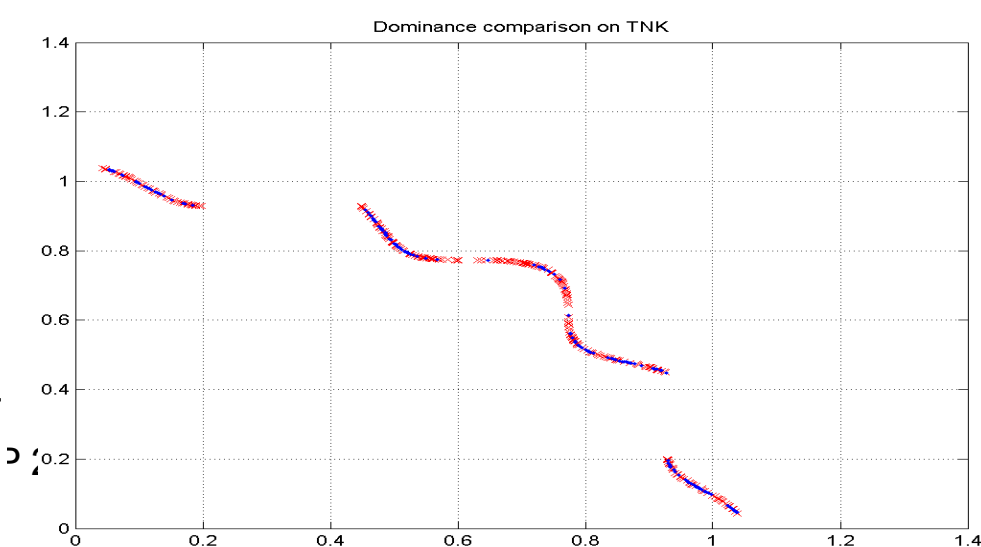
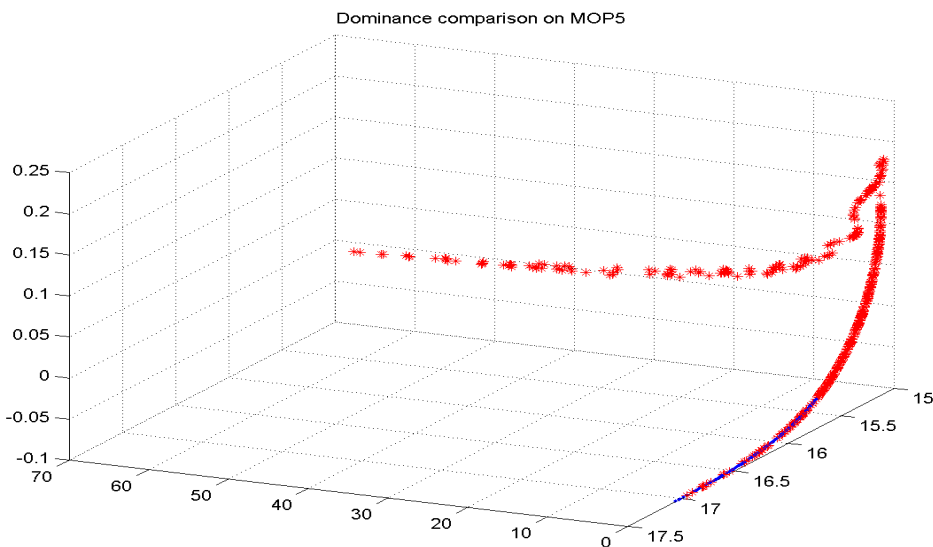
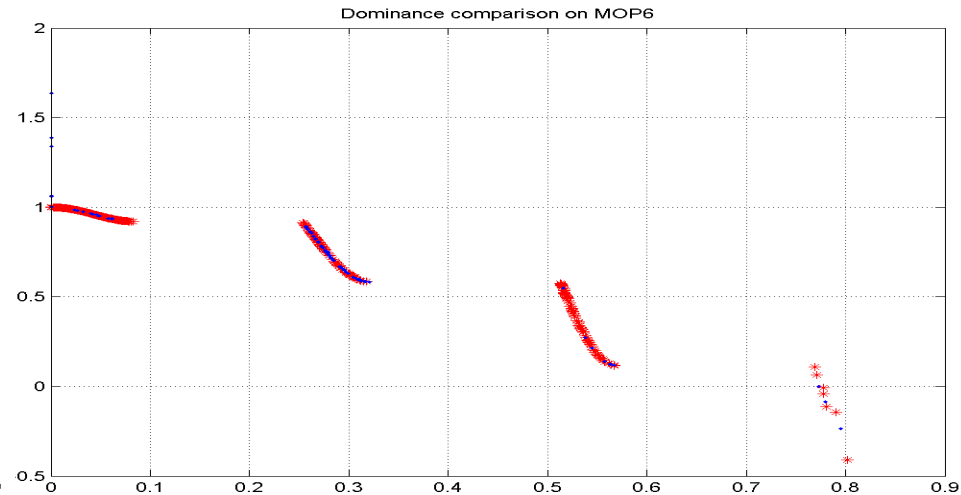
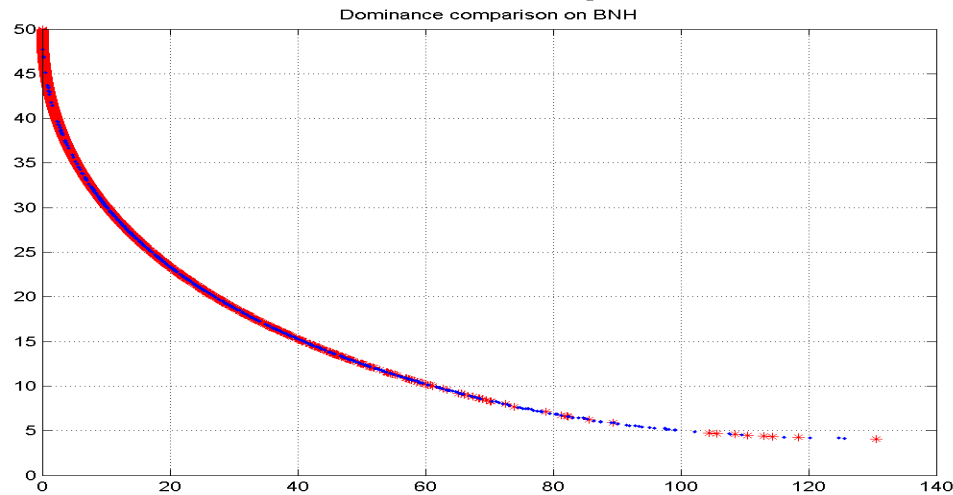
Expérimentations sur des problèmes classiques

Expérimentations

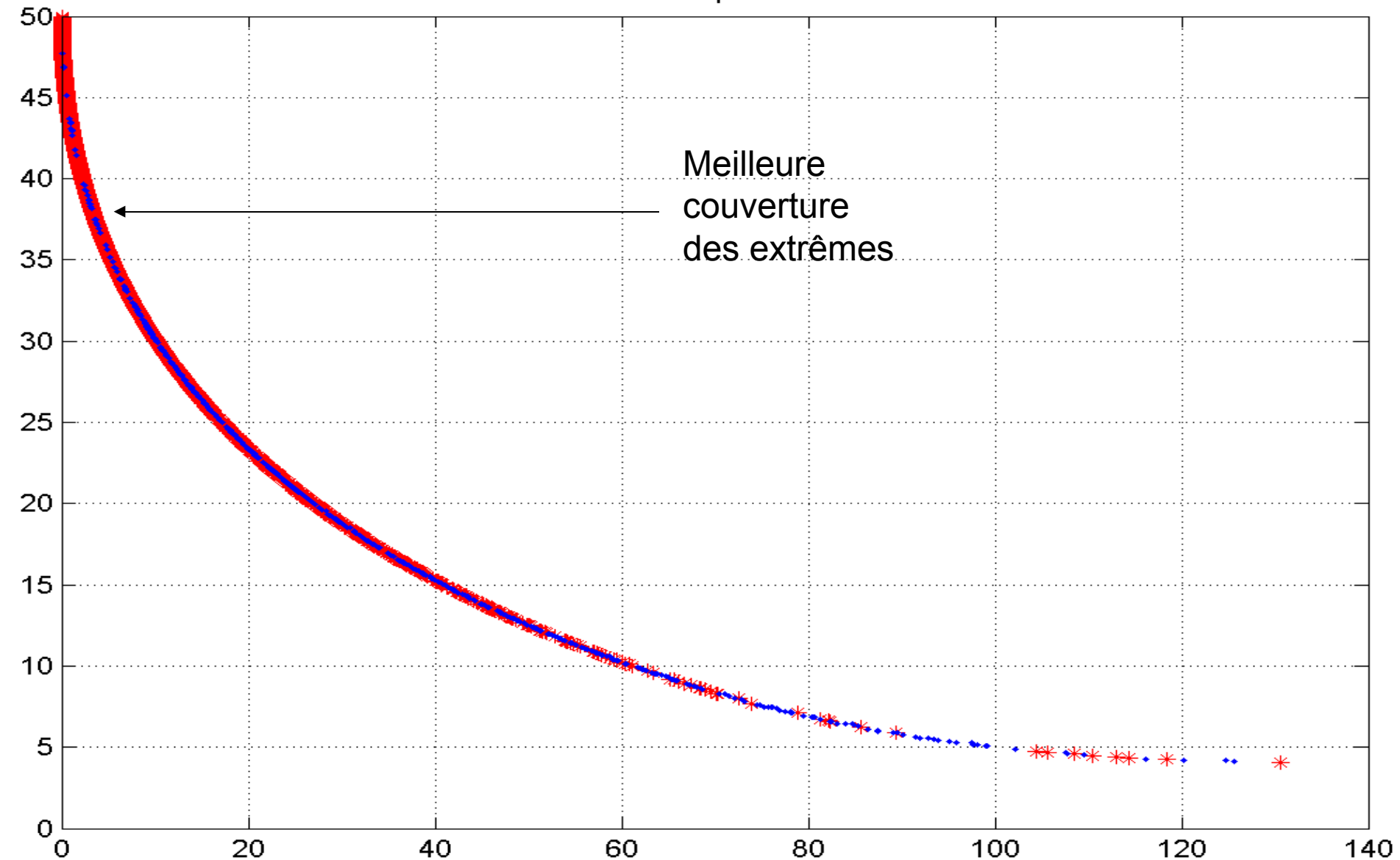
- Tests des nouvelles approches sur des problèmes jouets
 - BNH (MOPC1) : simple
 - MOP5 : multiplication des objectifs
 - MOP6 : discontinuités
 - TNK : multiplication des contraintes
- Compétition avec des MOPSO de base
- Critère d'arrêt : le nombre de calcul d'objectif
- Problème de l'aléatoire des MOPSO : estimation générale du comportement sur 100 répétitions

Résultats

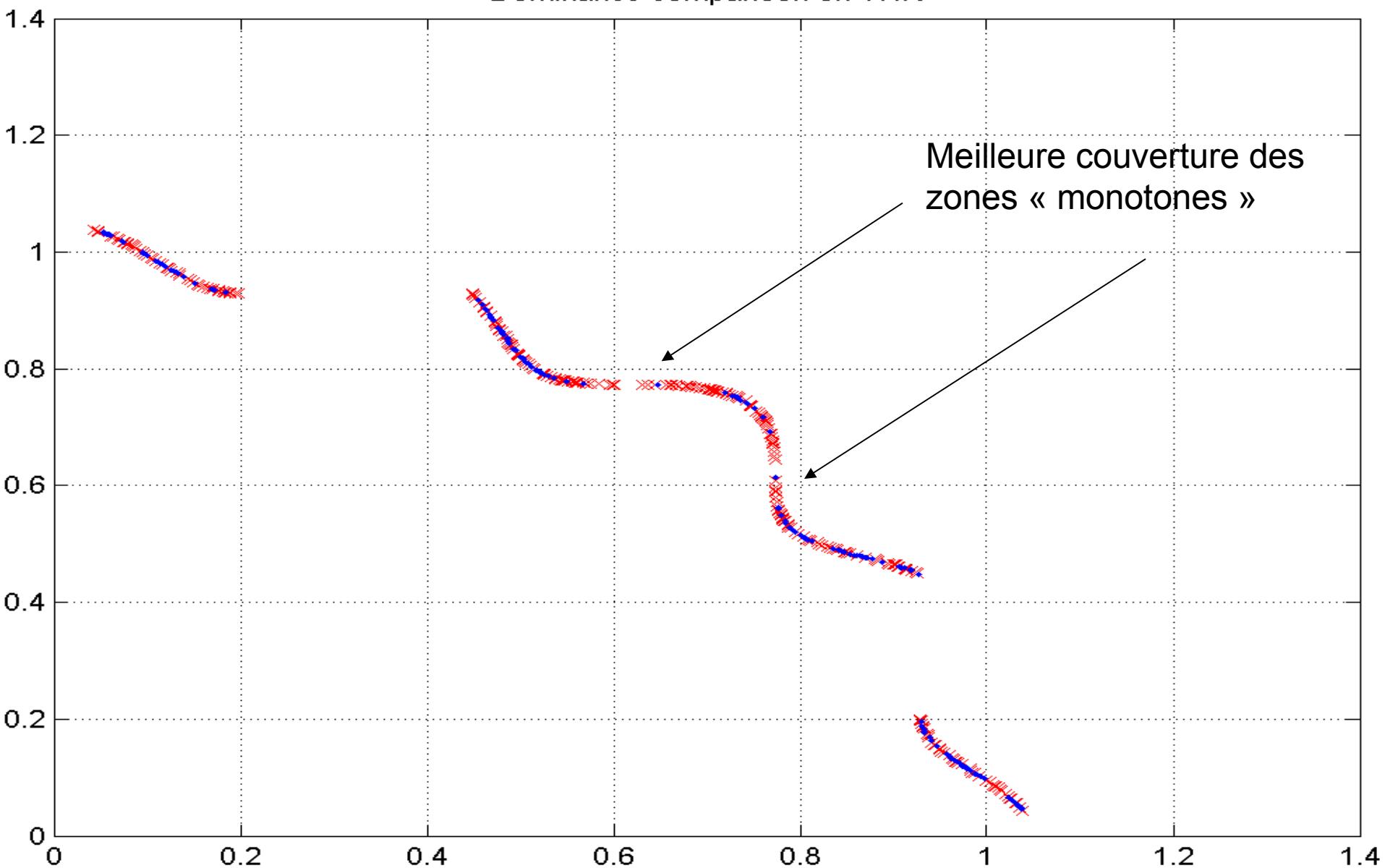
- Illustration : Epsilon dominance VS variante



Dominance comparison on BNH

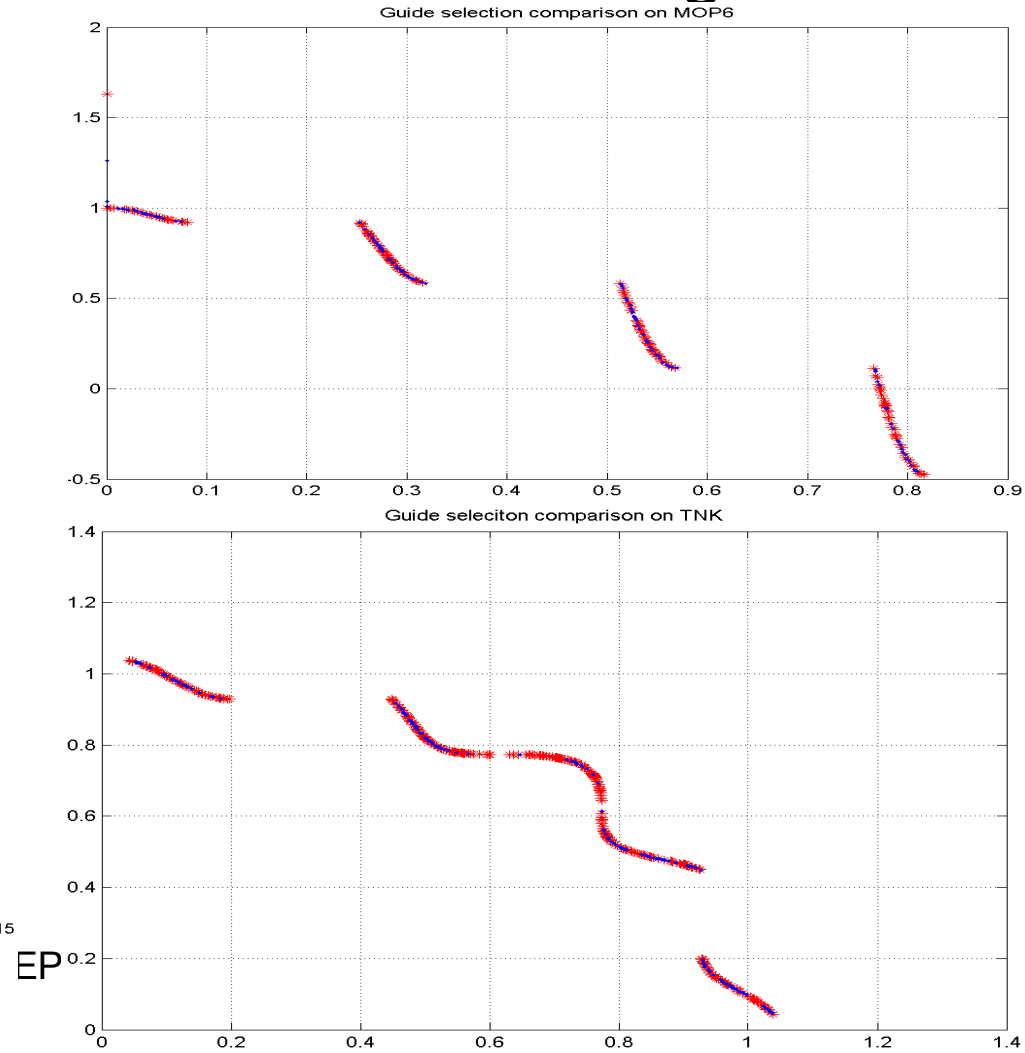
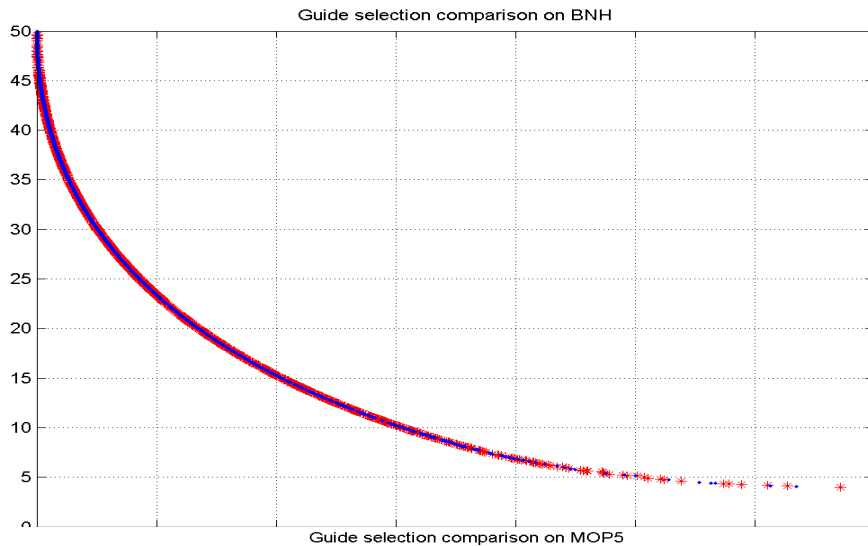


Dominance comparison on TNK

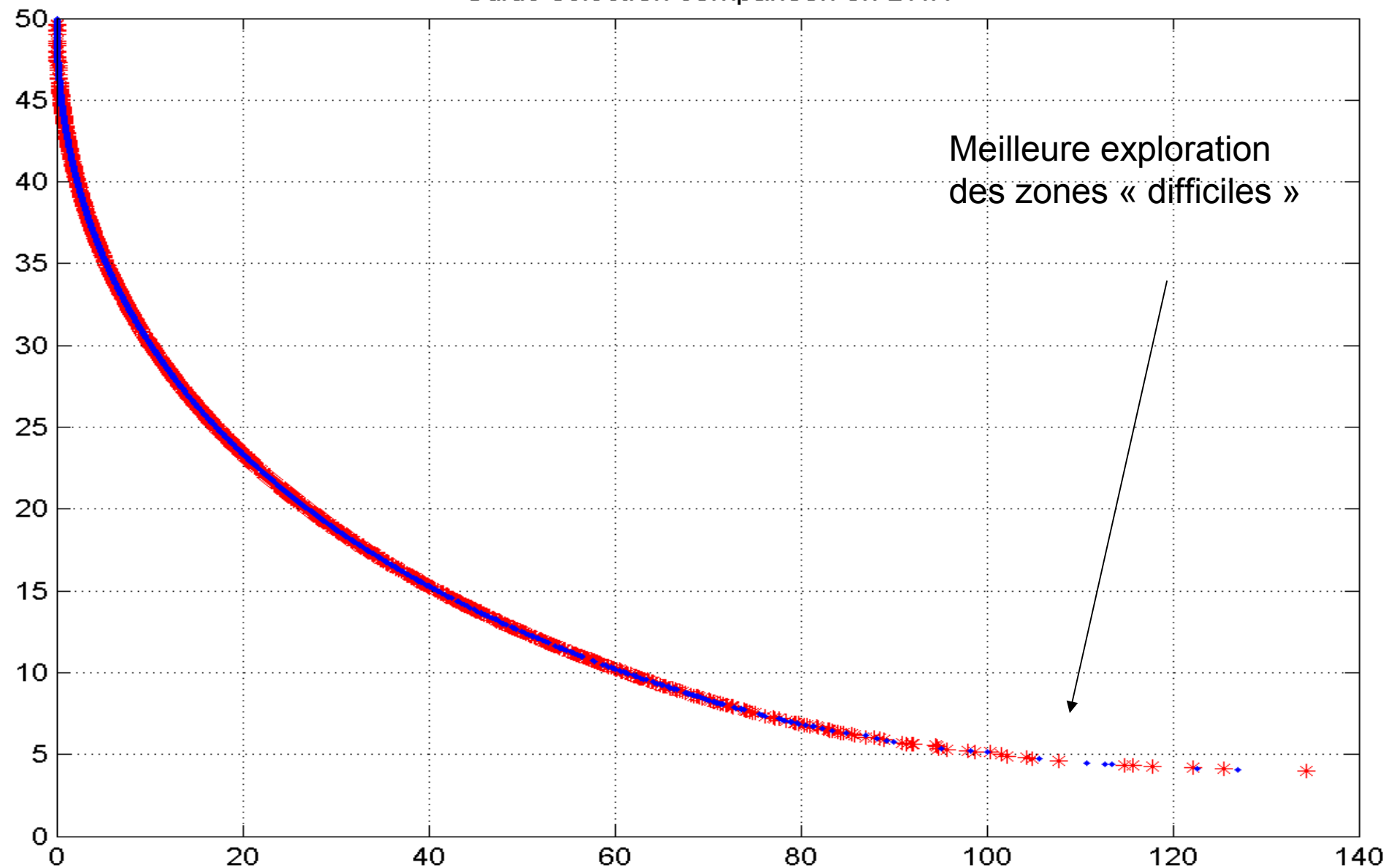


Résultats

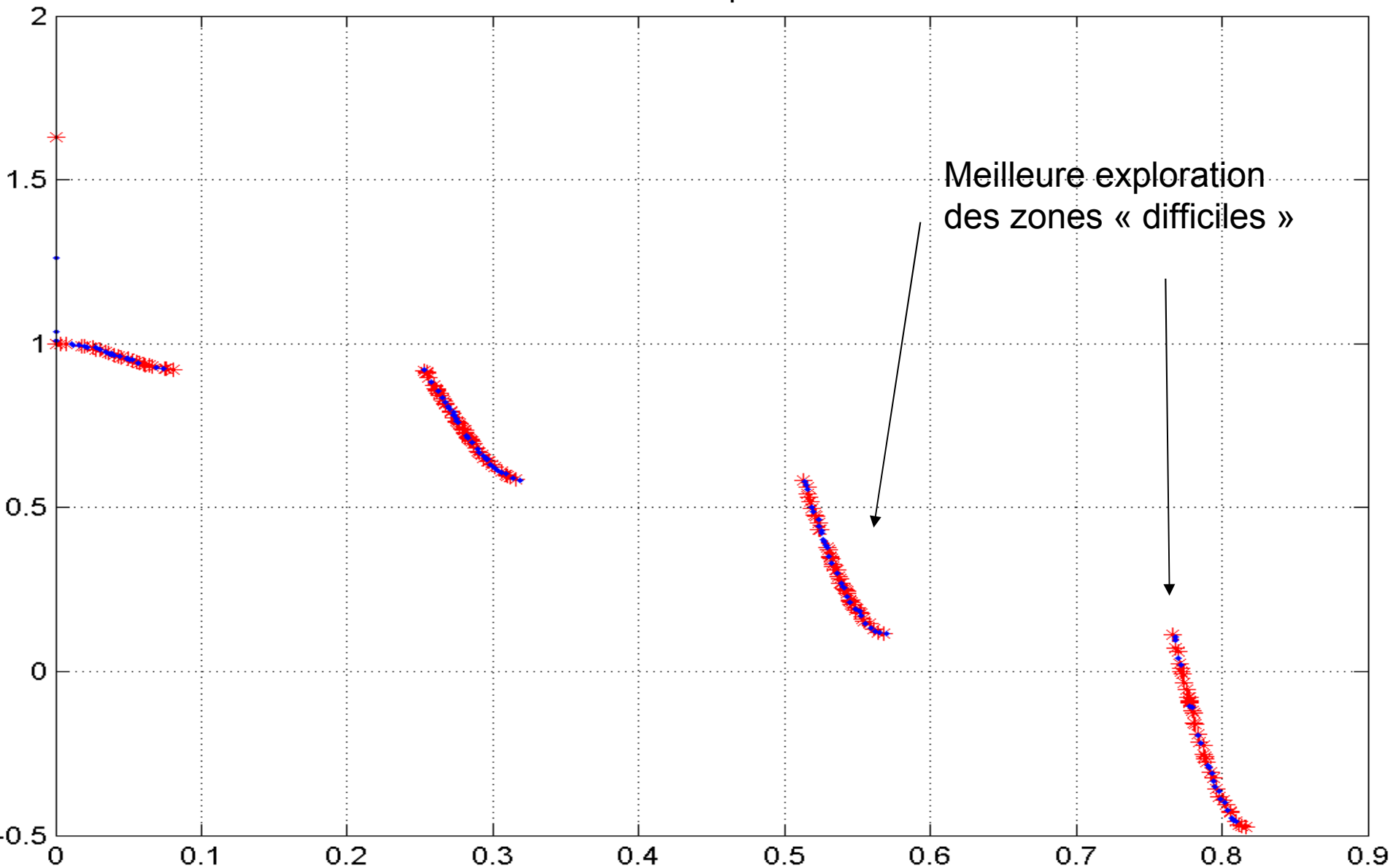
- Illustration : Amélioration de la sélection du guide



Guide selection comparison on BNH



Guide selection comparison on MOP6



Comparaison quantitative

- Métriques de comparaison
 - Taille de l'archive finale
 - Recouvrement des ensembles
 - Métrique d'espacement
 - Extension maximale

- Difficultés d'interprétations sans une analyse qualitative des fronts

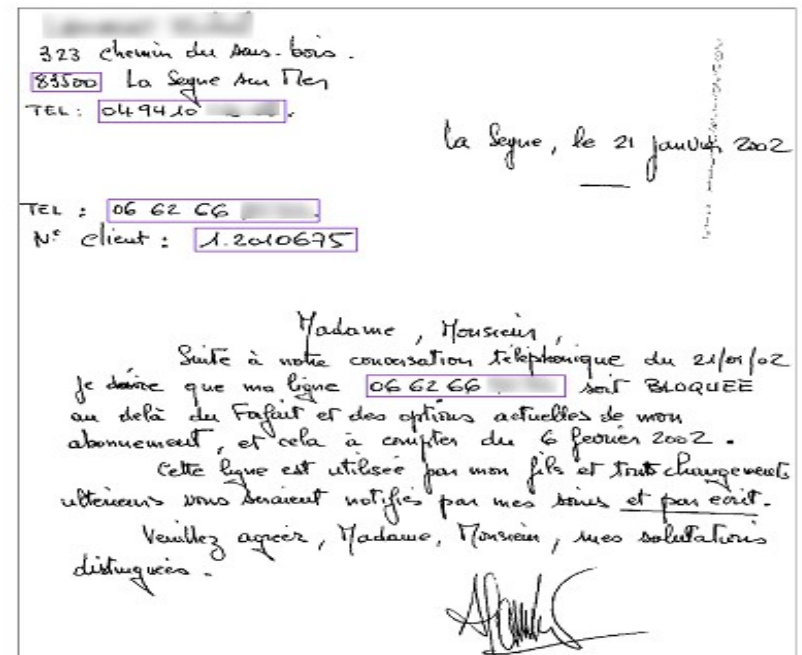
Validation sur un problème réel

Un problème réel : Analyse/tri de courrier manuscrit

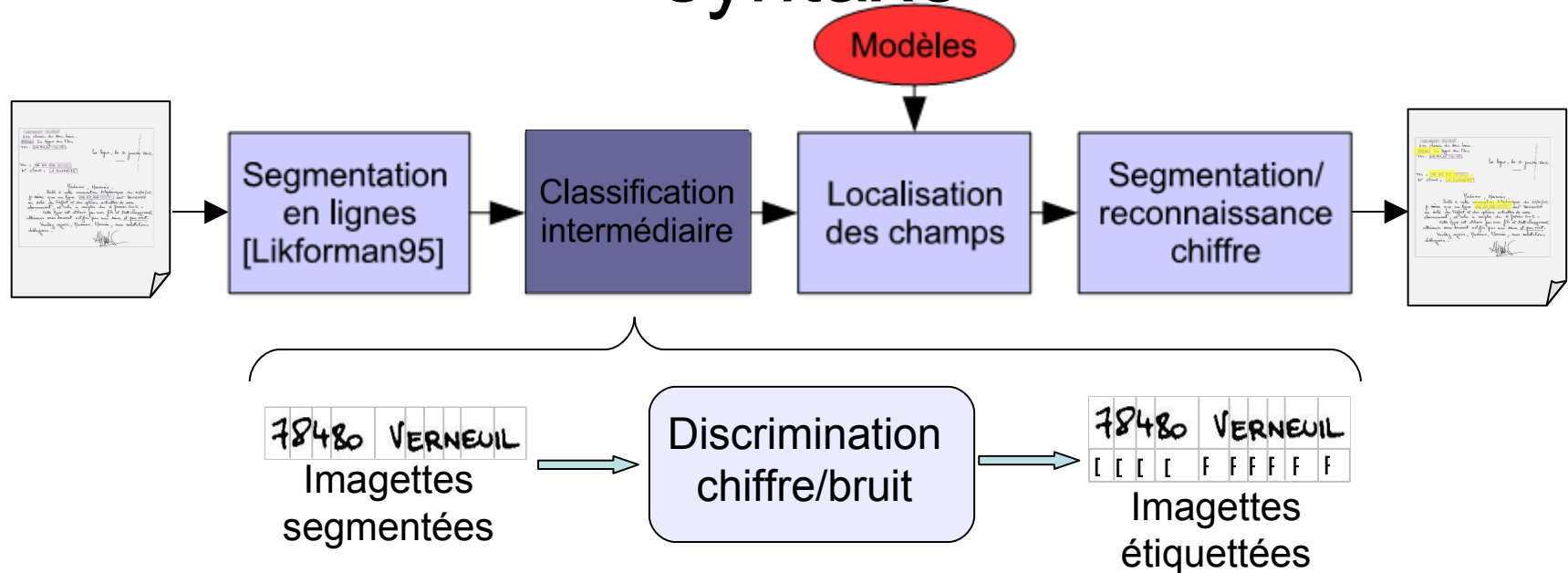
- Extraction de champs numériques pour une redirection des courriers :

- Numéro de téléphone
- Référence client
- Date
- Codes postaux

Problème de discrimination de chiffres dans des lettres manuscrites



Stratégie d'extraction guidée par la syntaxe

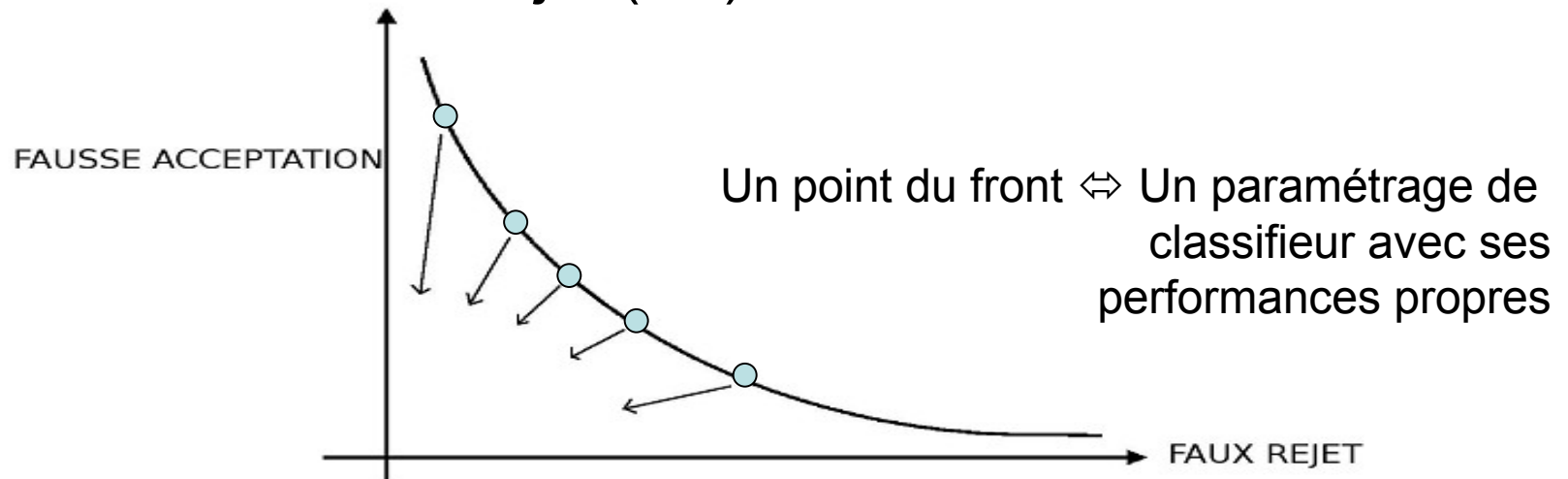


Objectifs : faciliter la localisation des champs numériques pour limiter les traitements lourds de reconnaissance aux éléments intéressants

→ Classifieur d'imagettes issues des documents

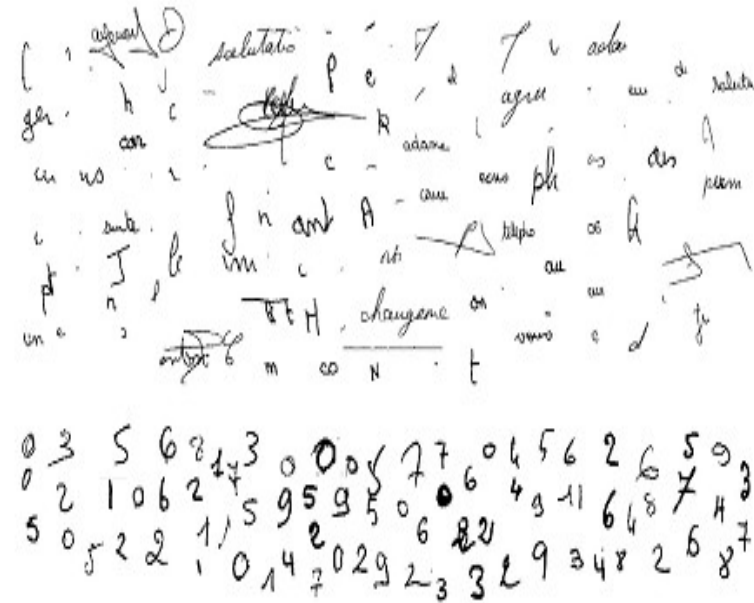
Un problème d'optimisation multi objectif

- Rejeter le plus possible pour filtrer au mieux
 - minimiser la fausse acceptation (FA)
- Éviter le rejet des formes numériques à reconnaître
 - minimiser le faux rejet (FR)



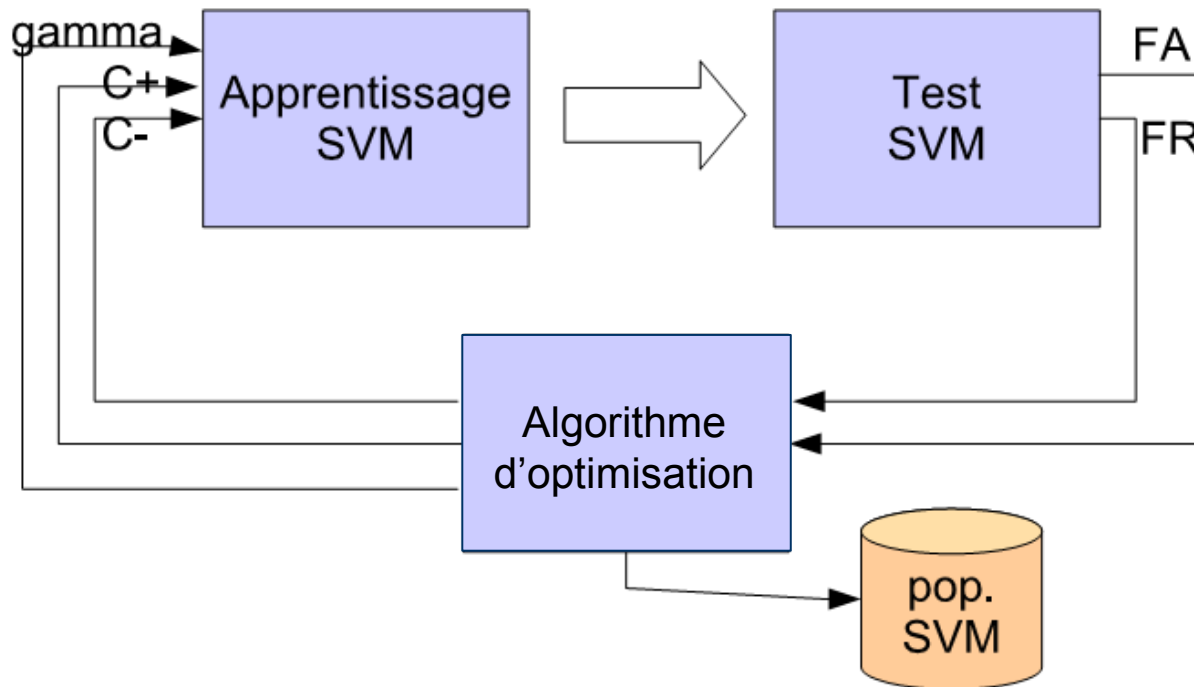
Application d'un classifieur SVM

- Intérêt des SVMs sur ce problème :
 - Adaptation au problème à 2 classes
 - Importance de la dissymétrie des coûts des erreurs sur chaque classe : adaptation des marges
- Besoin de régler les hyper paramètres du classifieur
 - C+ : marge positive
 - C- : marge négative
 - paramètres du noyau



Un choix de configuration optimale

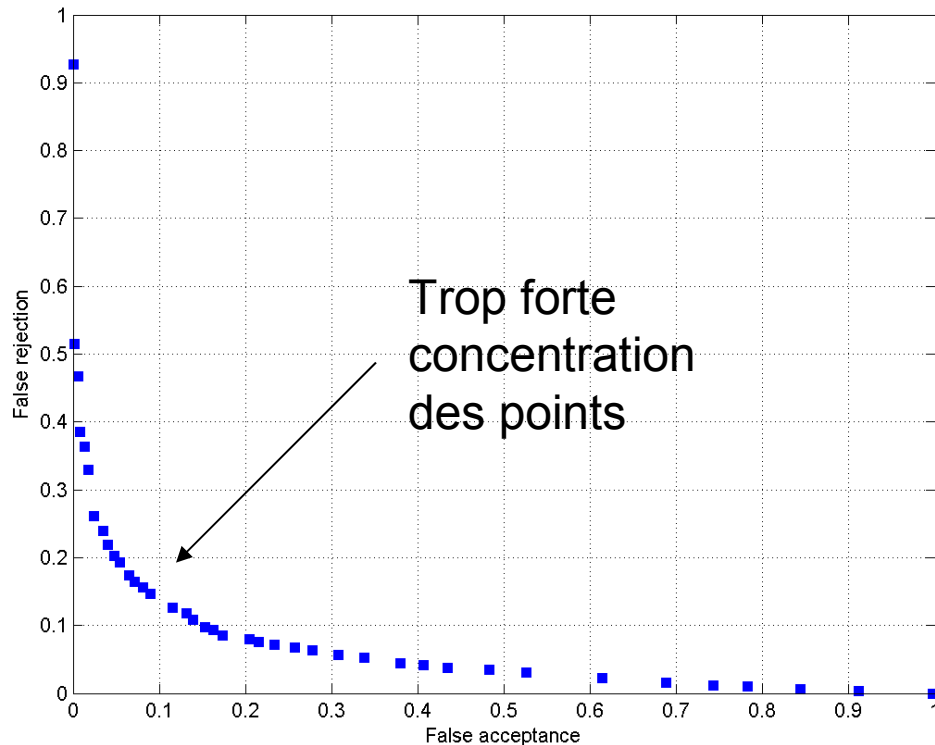
- Optimisation du SVM par MOPSO



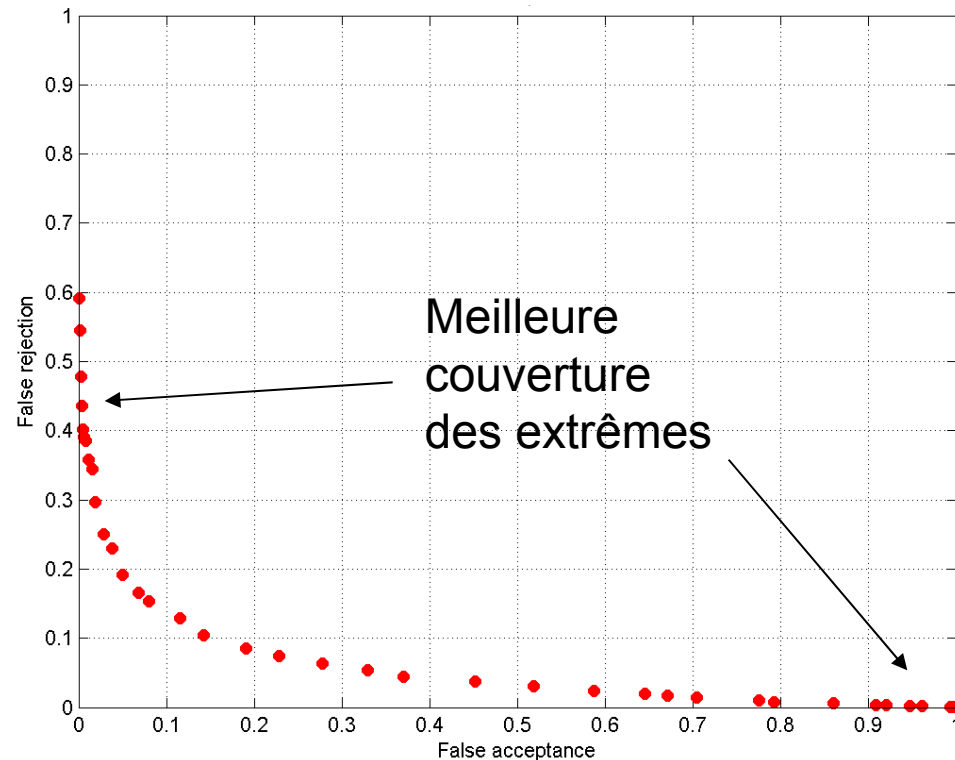
Apprentissage
d'un ensemble de
classifieurs
optimaux pour
chaque compromis
FA / FR

Résultats expérimentaux MOPSO

Sans améliorations



Avec améliorations



→ Une meilleure estimation du front

Comparaison à une approche évolutionnaire

- NSGA II (Non-Sorting Genetic Algorithm II)

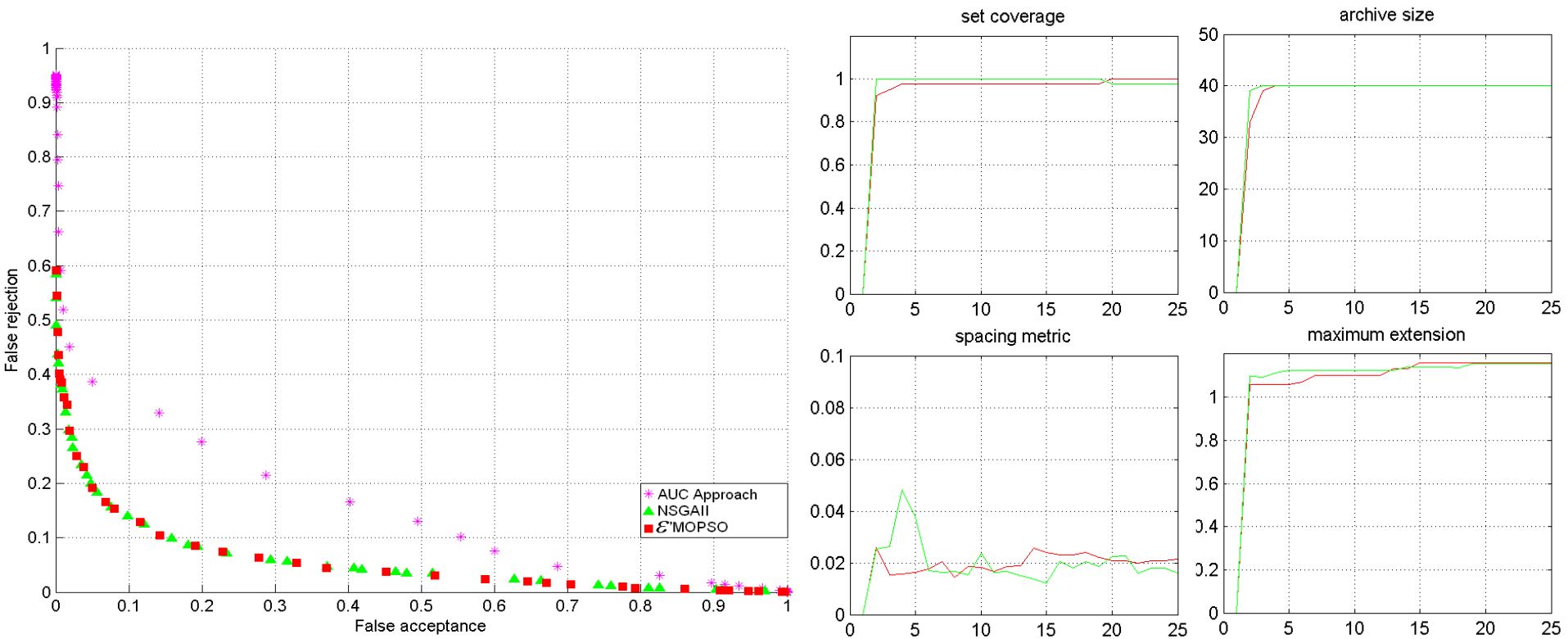
Algorithme génétique évolutionnaire

- Élitiste par rang de dominance
- Gestion du voisinage par évaluation d'une distance de surpeuplement

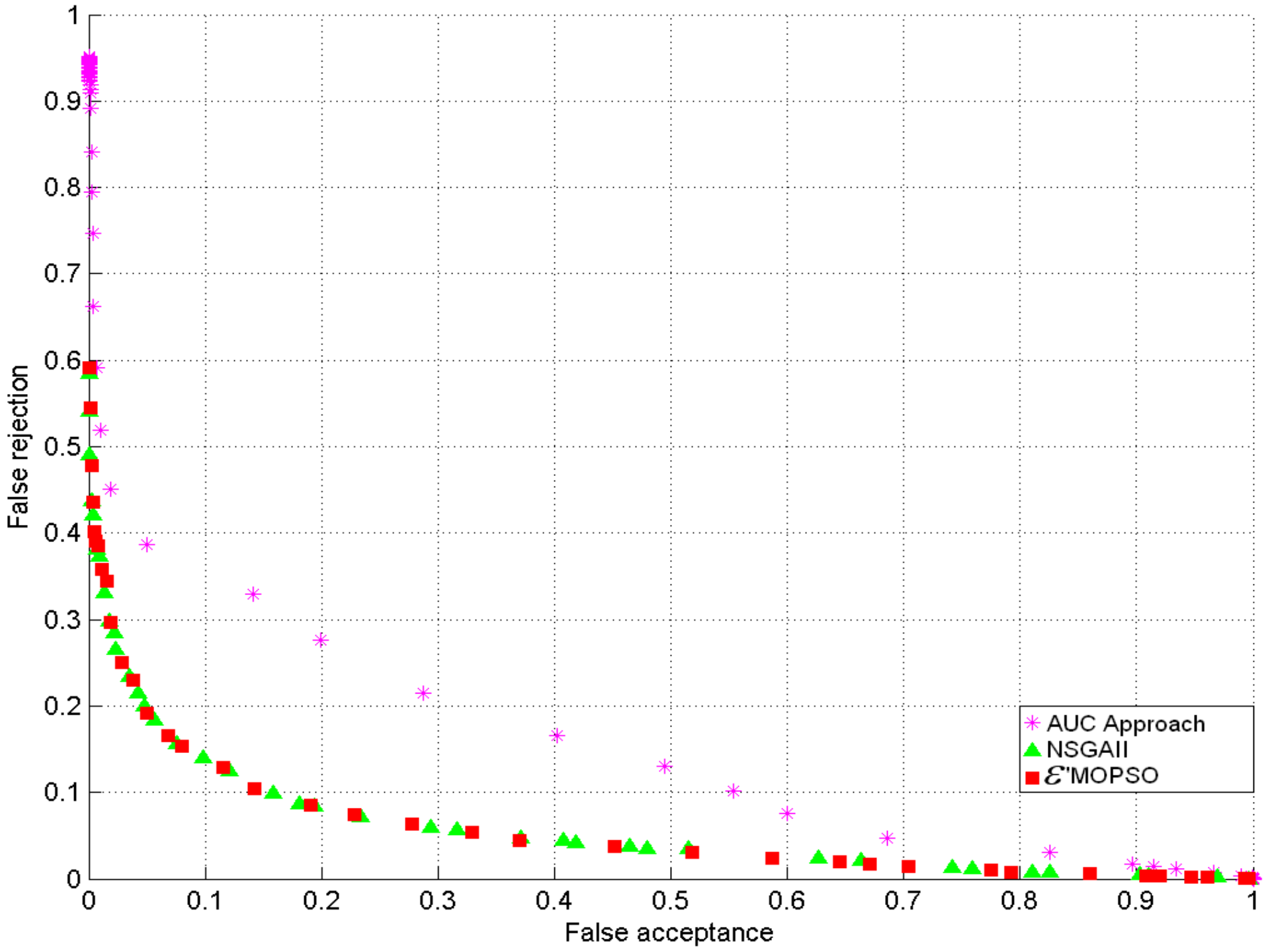
Nécessité d'adapter les MOPSO pour fixer la taille de l'archive

→ Heuristique de calcul dynamique de la valeur d' $\mathcal{E}$

Résultats expérimentaux MOPSO vs NSGAII



→ Des performances conformes à l'état de l'art



Conclusion sur l'approche

- Deux contributions sur des problèmes intrinsèques aux MOPSO
 - Une gestion simple du voisinage grâce à une variante de l'épsilon dominance
 - Une amélioration de la gestion des guides par une sélection probabiliste et une stratégie de mémorisation
- Approches validées par des expérimentations

Futurs axes de travail

- Amélioration de la gestion de la mémoire individuelle
- Élargissement de la notion de voisinage à l'espace des paramètres
- Amélioration de la gestion des extremums
 - Possibilités d'utilisation de la version absolue de l'épsilon dominance
- Autodétermination du paramètre epsilon
 - Des expérimentations empiriques prometteuses