



Réglage de paramètres de la physique d'ARPEGE PNT par une approche statistique

Ateliers de Modélisation de l'Atmosphère

L. Descamps - 12 Mars 2019

Détails de la démarche adoptée

- ▶ Définir une liste de N paramètres et leur plage de variation
- ▶ Fabriquer M jeux de paramètres qui échantillonnent l'hyper-espace de dimension N
- ▶ Effectuer M prévisions ARPEGE et les évaluer à l'aide de différentes métriques
- ▶ Construire des émulateurs associés à chaque métrique
- ▶ Chercher le sous-espace des paramètres dans lequel les métriques sont optimisées

Liste de paramètres de la physique d'ARPEGE et échantillonnage de l'hyper-espace de dimension N

- ▶ $N = 23$ paramètres fournis par l'équipe Processus Physiques
 - ▶ 6 liés à la **microphysique**, 4 liés à la **turbulence**, 9 liés à la **convection profonde**, 2 liés au **Gravity Wave Drag**, 2 liés au **schéma de surface**
- ▶ A chaque paramètre est associée une plage de variation
- ▶ Echantillonnage par Hypercube Latin pour optimiser le tirage au sort de $M = 300$ jeux de 23 paramètres

Réalisation de M + 1 prévisions et évaluation

- ▶ Réalisation des **301 prévisions ARPEGE** (T798 c2.4, 90 niveaux) à 108h d'échéance sur une période de 2 mois de l'hiver 2017 (1 jour sur 2 soit 32 jours)
- ▶ **Evaluation avec 3 métriques (EQM, EAM, BIAIS)** pour plusieurs échéances, domaines et variables. Scores aux analyses IFS (sauf cumul de pluie et force du vent à 10m évalués aux observations)

Algorithme d'optimisation

K₀ = 301

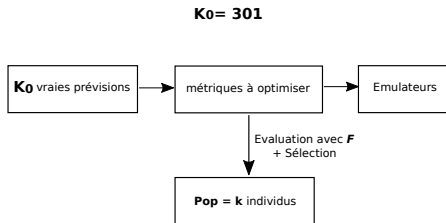
K₀ vraies prévisions

Algorithme d'optimisation

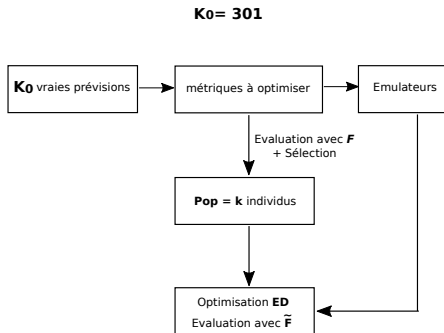
K₀ = 301



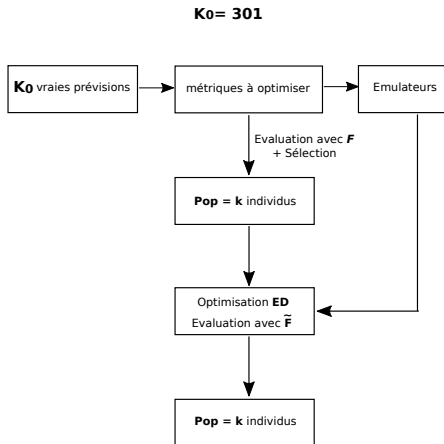
Algorithme d'optimisation



Algorithme d'optimisation

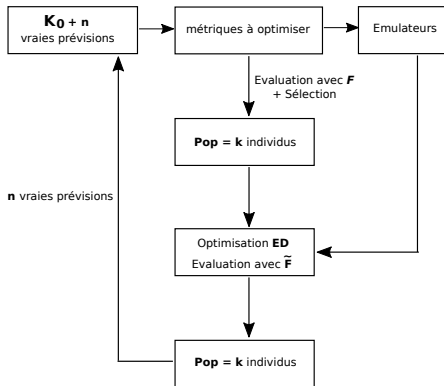


Algorithme d'optimisation



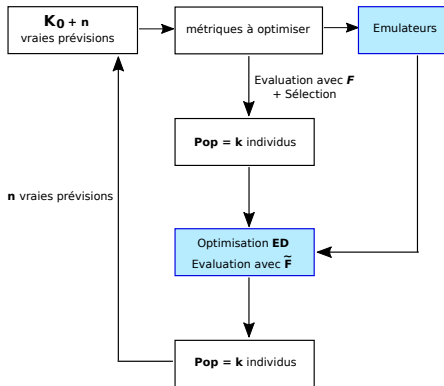
Algorithme d'optimisation

Processus renouvelé p fois avec $K_0 = 301$



Algorithme d'optimisation

Processus renouvelé p fois avec $K_0 = 301$

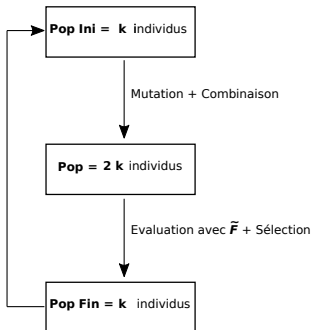


Un émulateur pour chaque métrique à optimiser

- ▶ Utiliser les résultats des évaluations des 301 prévisions comme apprentissage pour construire des émulateurs
- ▶ **Emulateur : modèle statistique à coût faible** qui remplace le modèle « physique » pour permettre un grand nombre de simulations
- ▶ Représenter à l'aide de l'émulateur la relation valeurs des 23 paramètres \longrightarrow valeur du score
- ▶ Choix d'un **émulateur de type Processus Gaussien** (covariance de type Matérn-5/2)

Algorithme d'optimisation à Evolution Différentielle

Algorithme d'optimisation ED

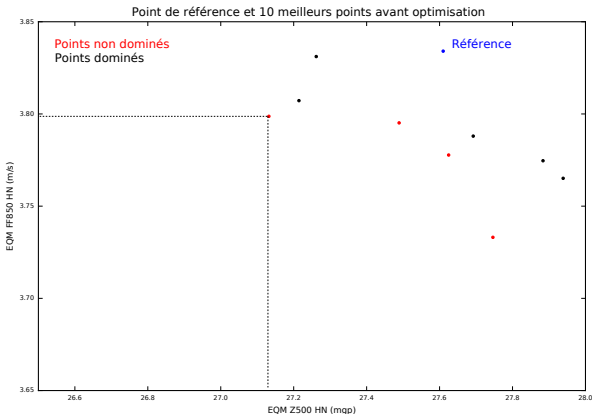


- ▶ Processus cyclé pour obtenir la m -ième génération
- ▶ \tilde{F} calculée à l'aide des émulateurs

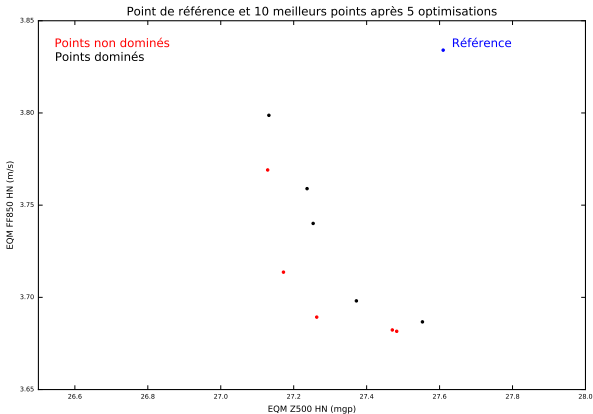
Exemple en 2 dimensions

- ▶ **2 métriques à optimiser** : f_1 : EQM de Z500 sur HN à 78h et f_2 : EQM de FF850 sur HN à 78h
- ▶ Fonction d'**évaluation** : $F(X_i)$ nombre de points qui dominant X_i **au sens de Pareto**
- ▶ Au sens de Pareto X_j domine X_i **SSI**
 $f_1(X_j) < f_1(X_i)$ et $f_2(X_j) < f_2(X_i)$
- ▶ Population de **k=100 individus**, **p=20 optimisations** réalisées avec à chaque fois **m=250 générations** pour l'algorithme ED, **n=5 prévisions** effectuées en sortie

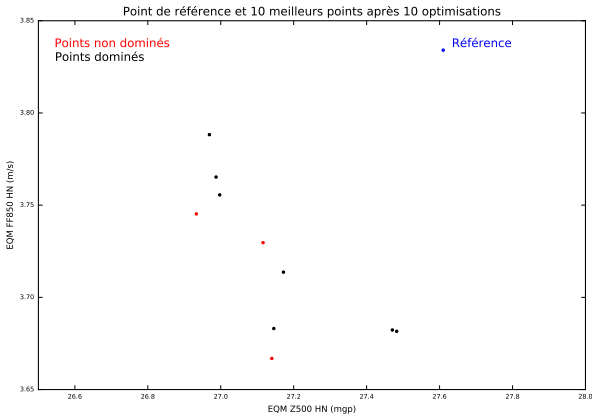
Exemple en 2 dimensions : avance du pseudo-front de pareto



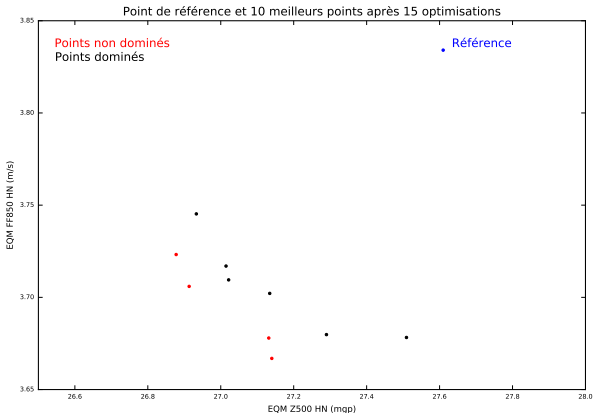
Exemple en 2 dimensions : avance du pseudo-front de pareto



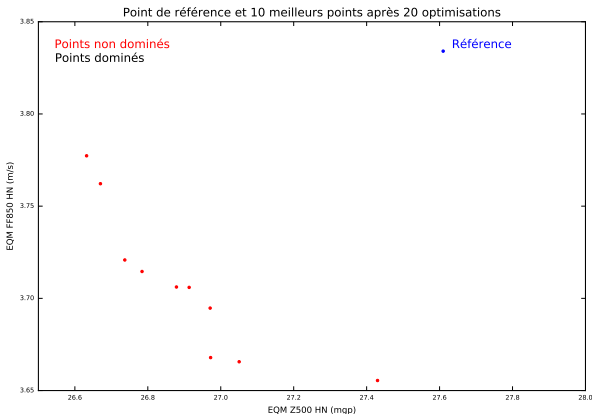
Exemple en 2 dimensions : avance du pseudo-front de pareto



Exemple en 2 dimensions : avance du pseudo-front de pareto



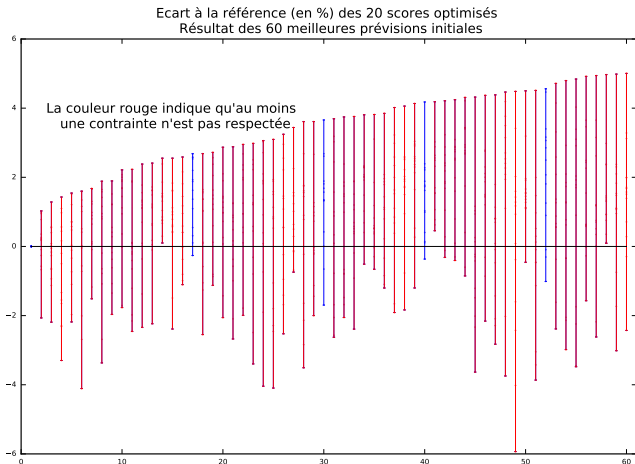
Exemple en 2 dimensions : avance du pseudo-front de pareto



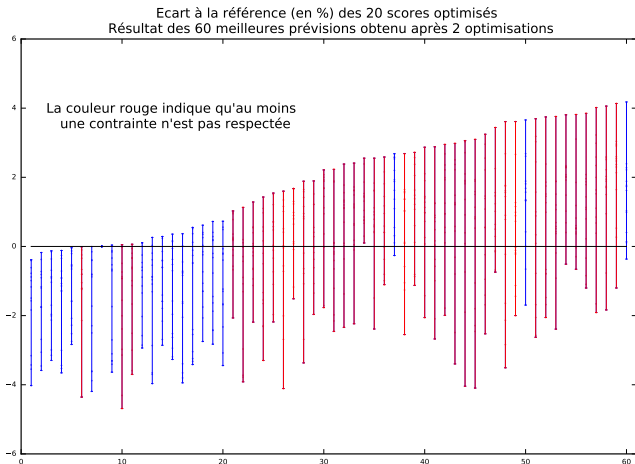
Optimisation multiobjectif avec contraintes

- ▶ **14 critères à optimiser** : rapport à la référence des **EQM** de Z500, T850, T500, T300, FF850, FF300, HU700 sur GL à 30h et 78h
- ▶ **6 contraintes** liées aux écarts à la référence des biais de T850 et T300
- ▶ Fonction d'évaluation : **F** basée sur le **classement dans la domination** (Angantyr et al. 2003) et la **L-optimalité** (Zou et al. 2008)
- ▶ Population de **k=100 individus**, **p=6 optimisations** réalisées avec à chaque fois **m=250 générations** pour l'algorithme ED, **n=10 prévisions** effectuées en sortie

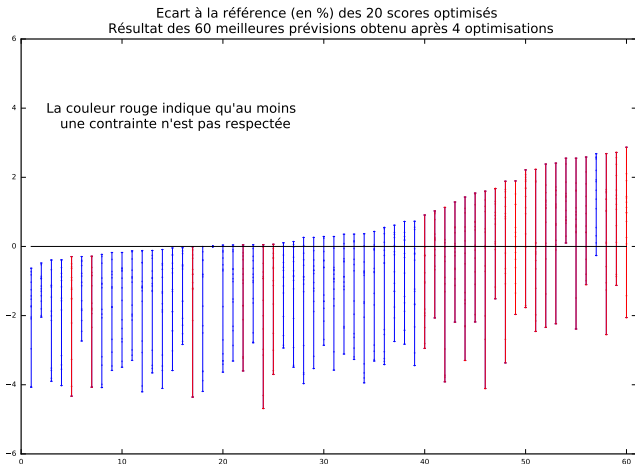
Optimisation multiobjectif avec contraintes : résultats



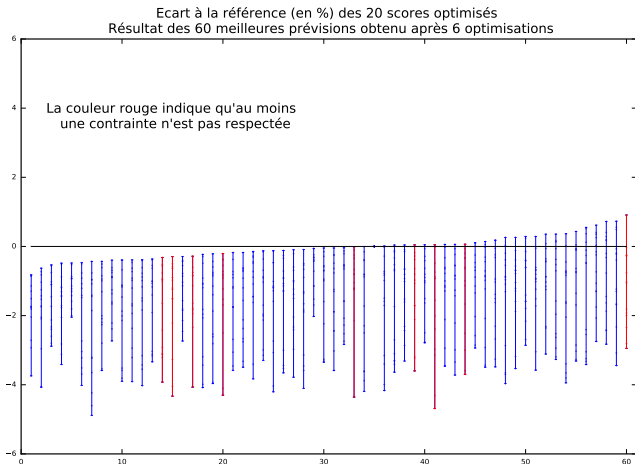
Optimisation multiobjectif avec contraintes : résultats



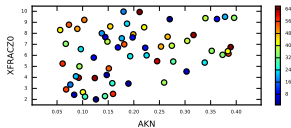
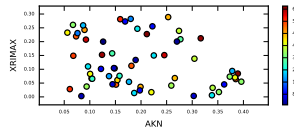
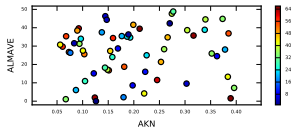
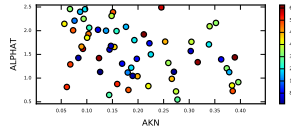
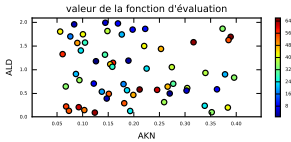
Optimisation multiobjectif avec contraintes : résultats



Optimisation multiobjectif avec contraintes : résultats

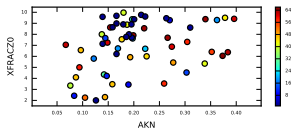
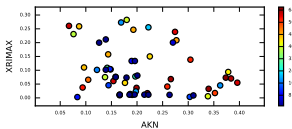
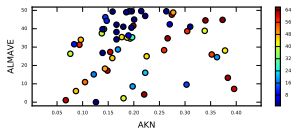
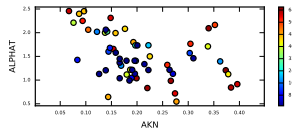
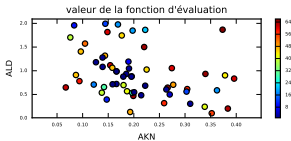


Optimisation multiobjectif avec contraintes : résultats



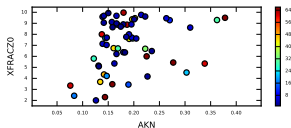
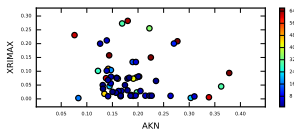
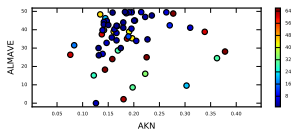
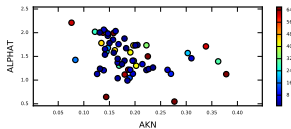
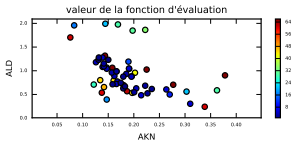
Répartition initiale des 60 meilleurs points

Optimisation multiobjectif avec contraintes : résultats



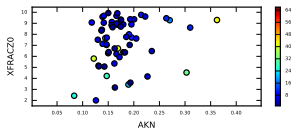
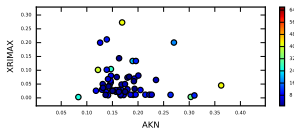
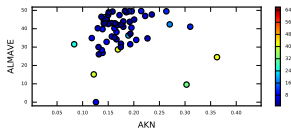
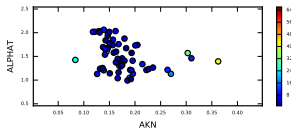
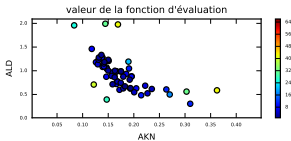
Répartition des 60 meilleurs points
après 2 optimisations

Optimisation multiobjectif avec contraintes : résultats



Répartition des 60 meilleurs points
après 4 optimisations

Optimisation multiobjectif avec contraintes : résultats



Répartition des 60 meilleurs points
après 6 optimisations

Conclusions

- ▶ Le réglage automatique de paramètres montre un potentiel d'amélioration non-négligeable même en grande dimension
- ▶ Les améliorations obtenues semblent se retrouver en testant d'autres périodes et/ou avec d'autres configurations d'ARPEGE
- ▶ Le choix des scores à optimiser est crucial, comment choisir les bonnes métriques en PNT ?
- ▶ L'approche nécessite une compréhension des résultats obtenus (discussion avec des physiciens/spécialistes du modèle) avant application

Perspectives

- ▶ Appliquer ce type de méthode aux variables diagnostiquées en sortie de modèle : rafales, visibilité, ...
- ▶ Appliquer ce type d'approche pour la mise en place de méthodes de perturbations de paramètres en prévision d'ensemble

Références

- ▶ A. Angantyr ; J. Andersson ; J-O. Aidanpaa (2003). Constrained optimization based on a multiobjective evolutionary algorithm. The 2003 Congress on Evolutionary Computation
- ▶ X. Zou ; Y. Chen ; M. Liu ; L. Kang (2008). A New Evolutionary Algorithm for Solving Many-Objective Optimization Problems. IEEE Transactions on Systems, Man and Cybernetics V30 n5

Optimisation multiobjectif avec contraintes : détails de F

$$F(X_j) = \frac{k_a}{k} r_1(X_j) + \frac{k - k_a}{k} r_2(X_j)$$

- ▶ k : taille de la population, k_a : nombre d'individus acceptables, r_1 rang sur les critères, r_2 rang sur les contraintes
- ▶ r_1 : Classement en fonction du nombre de critères améliorés. Si égalité, on départage en fonction de la valeur du plus mauvais critère
- ▶ r_2 : Classement en fonction du nombre de contraintes respectées