

## Chapitre 4

# Algorithme Pca-Mads

Afin d'améliorer les performances de l'algorithme MADS, on cherche à définir une nouvelle stratégie pour l'étape de recherche. Celle-ci consiste à utiliser une méthode d'analyse de sensibilité afin de définir un nouveau problème d'optimisation de plus petite taille. Cela a déjà été fait dans certains travaux comme décrit dans la section 3.4. A la différence des méthodes présentées dans cette section, on cherche à fixer des combinaisons de variables et pas seulement des variables. Pour ce faire, on se base sur l'analyse en composante principale, décrite à la section 2.6, pour construire un problème de plus petite dimension. Nous appelons cet algorithme PCA-MADS.

L'analyse et la construction du problème en petite dimension sont décrites dans les section 4.1, 4.2 et 4.3. La section 4.4 décrit le nouvel algorithme proposé tandis que la section 4.6 reprend les paramètres principaux de celui-ci. L'influence de ses paramètres sera analysée au chapitre 5.

### 4.1 Analyse en composante principale dans Pca-Mads

L'analyse en composante principale est une méthode statistique qui permet, à partir de  $(n + 1)$  variables aléatoires corrélées, de définir  $(n + 1)$  variables non corrélées. Celles-ci sont des combinaisons linéaires des variables originales. En pratique, la première composante principale correspond à la direction qui reprend la plus grande variabilité des variables aléatoires. La deuxième composante principale est la direction qui reprend le plus de variabilité tout en étant orthogonale à la première, et ainsi de suite.

Dans le cadre de l'optimisation de boîtes noires, on veut exploiter l'ensemble des évaluations déjà effectuées, contenues dans la *cache*, pour déterminer des directions intéressantes. On suppose donc que l'on a  $(n + 1)$  variables aléatoires, notées  $X$ , qui correspondent aux  $n$  variables du problème d'optimisation et à la valeur correspondante à la fonction objectif. A partir des évaluations déjà effectuées, on peut estimer une matrice de covariance et effectuer une analyse en composante principale à partir de cette matrice. Cette analyse va donc nous fournir  $(n + 1)$  directions que l'on peut écrire dans une matrice  $M_+ \in \mathbb{R}^{(n+1) \times (n+1)}$ . Les nouvelles variables non corrélées  $\bar{X} \in \mathbb{R}^{n+1}$  sont obtenues grâce à la transformation des variables originales  $X$  et les directions obtenues par l'analyse :

$$\bar{X} = M_+^\top X. \quad (4.1)$$

L'analyse en composante principale est utilisée pour repérer des corrélations entre des variables à partir d'un nuage de point. Or, nous voulons l'utiliser afin de déterminer des directions en dimension  $n$  selon lesquelles la fonction objectif a le plus de variabilité. Nous pouvons ordonner ces directions en fonction de leur alignement avec la dimension de l'objectif. De cette manière, les premières directions sont celles qui influent le plus sur l'objectif.

On définit une matrice  $M \in \mathbb{R}^{n \times n}$  qui correspond aux  $n$  premières lignes des  $n$  premières colonnes de la matrice  $M_+$  réordonnées. Celles-ci donnent les directions de  $\mathbb{R}^n$  qui ont le plus d'impact sur l'objectif. On néglige également la direction ayant le moins d'impact afin d'obtenir une matrice carrée. On a donc une transformation qui permet de transformer un vecteur  $x \in \mathbb{R}^n$  en un vecteur

$$\bar{x} = M^\top x \quad (4.2)$$

où les premières variables ont plus d'impact sur l'objectif que les dernières.

Dans le cadre de l'optimisation, on cherche à réduire la dimension de l'espace de recherche en ne gardant que des dimensions qui ont une grande influence sur l'objectif. On peut donc définir des matrices de projection  $P \in \mathbb{R}^{n \times p}$  et  $Q \in \mathbb{R}^{n \times (n-p)}$  telles que

$$M = [P \quad Q]. \quad (4.3)$$

La matrice  $P$  contient les  $p$  directions qui ont le plus d'impact sur l'objectif. D'une manière similaire à (4.2), on peut définir les vecteurs  $y \in \mathbb{R}^p$  et  $z \in \mathbb{R}^{n-p}$  comme

$$\bar{x} = \begin{bmatrix} y \\ z \end{bmatrix} = \begin{bmatrix} P^\top \\ Q^\top \end{bmatrix} x \quad (4.4)$$

**Exemple 4.1** Par exemple, considérons la fonction  $f_1(x_1, x_2) = 10x_1 + x_1x_2$  et le nuage de points en deux dimensions  $\{(-5, -5); (-5, 5); (5, -5); (5, 5); (0, 5); (5, 0); (1, 5); (5, 1); (-1, 5); (-5, 1)\}$ . L'ensemble des points évalués est repris dans le tableau 4.1. A partir de ce nuage de points, on peut construire une matrice de covariance  $S$ ,

$$S = \begin{bmatrix} 19.389 & -0.944 & 200.556 \\ -0.944 & 16.456 & -9.444 \\ 200.556 & -9.444 & 2361.111 \end{bmatrix}.$$

Les vecteurs propres de cette matrice de covariance sont les composantes principales du nuage de points. Ceux-ci sont les colonnes de la matrice  $M_+$  ci-dessous,

$$M_+ = \begin{bmatrix} -0.9964 & -0.0096 & 0.0847 \\ -0.0100 & 0.9999 & -0.0040 \\ 0.0847 & 0.0049 & 0.9964 \end{bmatrix}.$$

Dans notre situation, on cherche à minimiser la fonction objectif, on s'intéresse aux composantes principales étant le plus alignées sur la direction de l'objectif, i.e. le vecteur ayant la plus grande  $(n+1)^e$  composante en valeur absolue. On peut réordonner les colonnes de la matrice  $M_+$  selon leur valeur dans la  $(n+1)^e$  ligne, par ordre décroissant. On obtient

$$M'_+ = \begin{bmatrix} 0.0847 & -0.9964 & -0.0096 \\ -0.0049 & -0.0100 & 0.9999 \\ 0.9964 & 0.0847 & 0.0049 \end{bmatrix}.$$

Puisqu'on cherche des directions dans  $\mathbb{R}^2$ , la matrice  $M_+$  sera amputée de sa dernière ligne et de sa dernière colonne pour construire une matrice  $M'$ ,

$$M' = \begin{bmatrix} 0.0847 & -0.9964 \\ -0.0049 & -0.0100 \end{bmatrix}.$$

Les colonnes de cette matrice  $M'$  forment les directions de  $\mathbb{R}^n$  qui nous intéressent. On peut donc les normer pour construire une matrice  $M$ ,

$$M = \begin{bmatrix} 0.9983 & -0.9999 \\ -0.0578 & -0.0101 \end{bmatrix}.$$

A partir de cette matrice  $M$ , on peut définir les matrices de projection  $P$  et  $Q$  de l'équation (4.3), en fonction de la dimension  $p$  qui nous intéresse. Dans ce cas-ci, on peut noter que la direction  $(0.9983, -0.0578)^\top$  indique que la variable  $x_1$  a plus d'importance que la variable  $x_2$  sur l'objectif  $f_1(x_1, x_2) = 10x_1 + x_1x_2$ .

TABLE 4.1 – Ensemble d'évaluations pour les fonctions  $f_1(x_1, x_2) = 10x_1 + x_1x_2$  et  $f_2(x_1, x_2) = x_1 + x_2$

$x$	$f_1(x)$	$f_2(x)$
(-5, -5)	-25	-10
(-5, 5)	-75	0
(5, -5)	25	0
(5, 5)	75	10
(0, 5)	0	5
(5, 0)	50	5
(1, 5)	15	6
(5, 1)	55	6
(-1, 5)	-15	4
(-5, 1)	-55	-4

**Exemple 4.2** *Considérons la fonction  $f_2(x_1, x_2) = x_1 + x_2$  et le même ensemble de points que l'exemple précédent. De la même manière, on peut calculer les composantes principales du nuage de points, cela donne*

$$M_+ = \begin{bmatrix} 0.5774 & 0.6729 & 0.4625 \\ 0.5774 & -0.7370 & 0.3515 \\ -0.5774 & -0.0641 & 0.8140 \end{bmatrix}.$$

*On peut réordonner les colonnes de la matrice en fonction de la dernière ligne,*

$$M'_+ = \begin{bmatrix} 0.4625 & 0.5774 & 0.6729 \\ 0.3515 & 0.5774 & -0.7370 \\ 0.8140 & -0.5774 & -0.0641 \end{bmatrix}.$$

*Ensuite, en tronquant la dernière ligne et la dernière colonne, on obtient deux vecteurs que l'on peut normer, ce qui nous donne*

$$M = \begin{bmatrix} 0.7962 & 0.7071 \\ 0.6051 & 0.7071 \end{bmatrix}.$$

*Dans ce cas, les directions proposées sont assez proches de  $(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})^\top$ , ce qui correspond à la direction attendue.*

## 4.2 Changement de variables

Le sous-problème d'optimisation est défini à partir des variables  $y \in \mathbb{R}^p$ . Celles-ci devraient correspondre aux  $p < n$  combinaison linéaire des variables  $x$  qui ont le plus d'impact sur l'objectif. Puisque la dimension  $p$  est plus petite que la dimension  $n$ , plusieurs points en dimension  $n$  peuvent être projetés sur un même point  $y$  en dimension  $p$ .

Lors de la recherche dans le sous-espace, on évalue des points de dimension  $p$ . A partir de ces points, il faut pouvoir retrouver les points correspondants en dimension  $n$  afin d'évaluer la boîte noire. On propose ici deux solutions afin de pouvoir construire un point en dimension  $n$  correspondant à  $y \in \mathbb{R}^p$ .

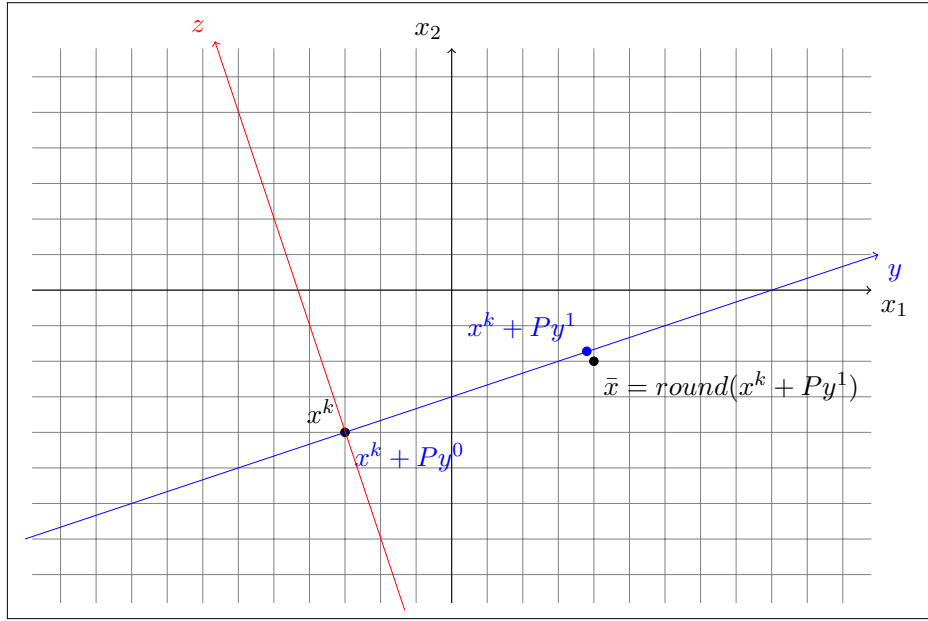


FIGURE 4.1 – Illustration de la transformation (4.5) en deux dimensions

La première solution consiste à compléter les informations manquantes par des informations que l'on a gardé. Lorsqu'on lance une instance de MADS sur le sous-problème, on lui fournit un point de départ. Celui-ci va être défini comme l'origine du sous-espace de recherche. On veut que ce point de départ corresponde au meilleur point trouvé jusqu'à présent  $x^k \in \mathbb{R}^n$ . Lorsqu'on veut évaluer un point  $y \in \mathbb{R}^p$ , on pourra évaluer le point

$$x = x^k + Py. \quad (4.5)$$

Puisque la matrice  $P$  est de dimension  $n \times p$ , le point  $x^k + Py$  est bien de dimension  $n$ . De plus, lorsque  $y = 0$ , on retrouve effectivement le point  $x^k$ . Des illustrations de cette transformation sont présentées aux figures 4.1 et 4.2.

La deuxième méthode se base sur la pseudo-inverse. Soit une matrice  $A \in \mathbb{R}^{m \times n}$ , avec  $m \leq n$ . La pseudo-inverse de Moore-Penrose [50] de  $A$  est une matrice  $A^\dagger \in \mathbb{R}^{n \times m}$  telle que

$$AA^\dagger A = A; \quad (4.6)$$

$$A^\dagger AA^\dagger = A^\dagger; \quad (4.7)$$

$$(AA^\dagger)^\top = AA^\dagger; \quad (4.8)$$

$$(A^\dagger A)^\top = A^\dagger A. \quad (4.9)$$

On note que si  $A$  a des colonnes linéairement indépendantes, alors  $A^\dagger = (A^\top A)^{-1} A^\top$ . On note  $(P^\top)^\dagger$  la pseudo-inverse de Moore-Penrose de la transposée de la matrice  $P$ . Si cette matrice est réelle et a des colonnes linéairement indépendantes, alors

$$(P^\top)^\dagger = (PP^\top)^{-1} P. \quad (4.10)$$

On a la relation  $y = P^\top x$ , comme défini dans l'équation (4.4). En multipliant la relation par la pseudo-inverse de  $P^\top$ , on obtient

$$x = (P^\top)^\dagger y. \quad (4.11)$$

Cette matrice pseudo-inverse peut donc être utilisée pour passer d'un espace à l'autre. Afin que l'origine du sous-espace corresponde à la meilleure solution trouvée  $x^k$ , on utilise la transformation

$$x = x^k + (P^\top)^\dagger y. \quad (4.12)$$

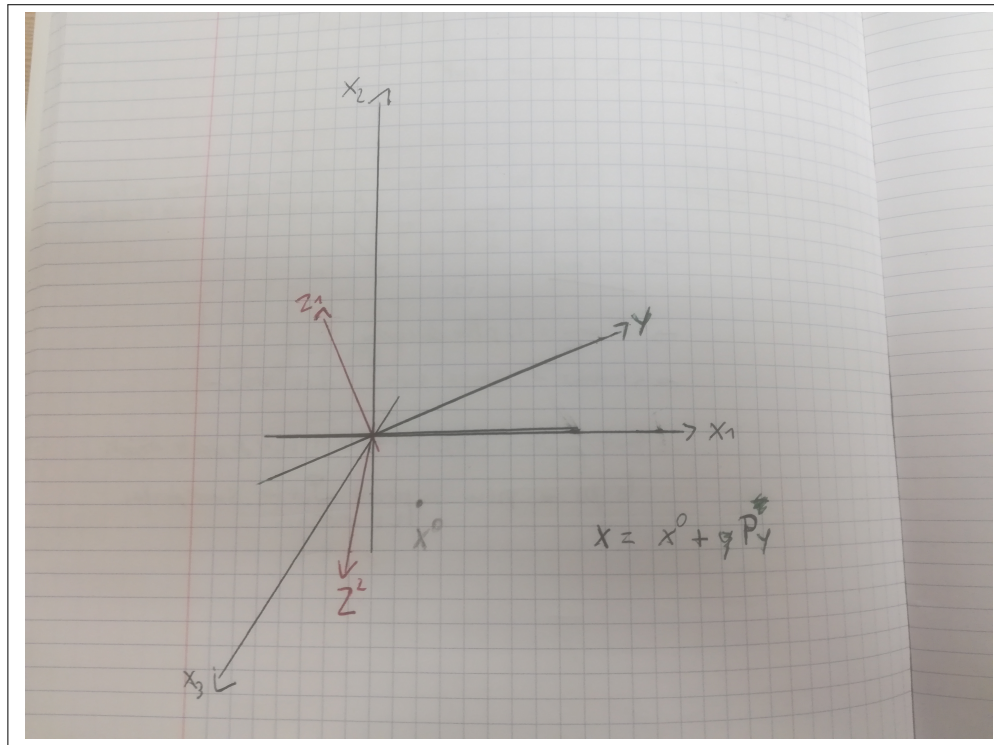


FIGURE 4.2 – Illustration de la transformation (4.5) en trois dimensions

### 4.3 Evaluation de la boîte noire

Dans le cadre de l'algorithme MADS, tous les points sont évalués sur un treillis. Dans l'algorithme PCA-MADS proposé, des points en dimension  $p$  sont projetés sur des points en dimension  $n$  afin d'être évalués. Pour préserver les résultats de convergence de l'algorithme original, ceux-ci seront alors arrondis afin d'appartenir au treillis en  $n$  dimension. Les points évalués n'appartiendront pas toujours au sous-espace défini et cela permettra de récupérer un peu d'information qui aurait été perdue sinon.

On considère que le treillis de l'algorithme MADS est défini par la matrice  $D = [I_n - I_n]$  où  $I_n$  est la matrice identité en  $n$  dimensions. Soit  $\Delta_{min}^m$  la valeur du plus petit paramètre de taille du treillis au cours de l'exécution de l'algorithme. Alors un point  $x \in \mathbb{R}^n$  sera projeté sur le point

$$\bar{x} = x^k + \Delta_{min}^m \times \text{round} \left( \frac{x - x^k}{\Delta_{min}^m} \right), \quad (4.13)$$

où  $x^k$  est un point sur le treillis et l'opérateur  $\text{round}(\cdot)$  est tel que  $\text{round}(0.6) = 1$  et  $\text{round}(-0.6) = -1$ .

Lors de l'optimisation du problème en dimension  $p$ , un point  $y \in \mathbb{R}^p$  est évalué de la façon suivante. Dans un premier temps, celui-ci est projeté en dimension  $n$  en utilisant l'une des méthodes (4.5) ou (4.12). Cela permet de définir un point  $x \in \mathbb{R}^n$  correspondant au point  $y \in \mathbb{R}^p$  à évaluer. Ensuite, le point  $x \in \mathbb{R}^n$  est projeté sur un point  $\bar{x}$  du treillis de MADS en dimension  $n$  selon (4.13). La boîte noire est évaluée au point  $\bar{x}$  et sa sortie est utilisée dans l'optimisation du sous-problème comme si cette valeur correspondait à l'évaluation de la boîte noire réduite au point  $y \in \mathbb{R}^p$ .

### 4.4 Algorithme Pca-Mads

Le cadre algorithmique de MADS laisse la possibilité d'ajouter une étape de recherche qui peut évaluer un nombre fini de points sur le treillis.

L'algorithme PCA-MADS propose d'optimiser un problème en dimension réduite comme étape de recherche. La construction de ce problème se base sur une analyse en composantes principales et évalue des points sur le treillis. En donnant un budget fini pour l'optimisation de ce problème en

petite dimension et en évaluant la boîte noire sur le treillis, cette étape de recherche satisfait donc bien les conditions d’une étape de recherche dans MADS. Idéalement, on voudrait pouvoir poursuivre la recherche uniquement dans un problème en petite dimension. Ceci est possible tant que la recherche en petite dimension trouve des solutions plus intéressantes que celles déjà connues. Si la recherche n’améliore pas la solution courante, alors on effectue une étape de sonde en grande dimension, selon l’algorithme MADS classique.

Une description de l’algorithme PCA-MADS est proposée à l’algorithme 10.

---

**Algorithme 10** PCA-MADS

---

0. Initialisation
  1. Evaluer  $N$  points sur le treillis
  2. Boucle :
    - 2.1 Définir un sous-problème de dimension  $p$  à partir des  $N$  derniers points évalués et d’une analyse en composante principale ;
    - 2.2 Lancer une instance de MADS sur le sous-problème avec un budget  $B$  fini ;
    - 2.3 Si aucun point améliorant la solution n’a été trouvé, effectuer une étape de sonde en  $n$  dimension ;
    - 2.4 Evaluer les critères de terminaison de l’algorithme en dimension  $n$  et mettre à jour les paramètres en conséquence
- 

## 4.5 Analyse de convergence

L’algorithme PCA-MADS proposé fait partie de la classe d’algorithmes MADS décrits à la section 3.3.3. Son analyse de convergence sera donc similaire à l’analyse de convergence de cette classe d’algorithmes.

Dans un premier temps, il faut qu’à chaque itération de l’algorithme PCA-MADS, deux étapes successives peuvent être effectuées. La première est l’étape de recherche, qui consiste en la construction d’un sous-problème et l’optimisation de celui-ci avec un certain budget d’évaluations fini. La deuxième est l’étape de sonde qui n’est appliquée que si l’étape de recherche n’a pas généré de nouvelle solution améliorante. De plus, aussi à l’étape de recherche qu’à l’étape de sonde, tous les points évalués appartiennent à une structure finie appelée treillis, définie à la définition 3.1.

En appliquant les règles de mise à jour des paramètres de taille du treillis  $\delta^k$  et de taille de cadre  $\Delta^k$ ,

$$\begin{aligned} \delta^k &= \min(\Delta^k, (\Delta^k)^2), \\ \Delta^{k+1} &= \tau^{\omega_k} \Delta^k, \\ \text{pour } \omega_k &\in \begin{cases} \{0, 1, 2, \dots, \omega^+\} & \text{si un point améliorant la solution est trouvé,} \\ \{\omega^-, \omega^- - 1, \dots, -1\} & \text{sinon,} \end{cases} \end{aligned}$$

le maillage du treillis est raffiné uniquement lors des itérations infructueuses, *i.e.* les itérations où aucun point améliorant la solution n’a été trouvé.

Pour analyser la convergence de l’algorithme, on s’intéresse à la situation où le nombre d’itérations tend vers l’infini et l’ensemble de niveau de la fonction objectif  $\{x \in \mathbb{R}^n : f(x) \leq f(x_0)\}$  forme un compact. Puisque tous les points évalués appartiennent au treillis et que ce dernier est une structure finie, il n’y a qu’un nombre fini d’itérations réussies, *i.e.* une nouvelle solution améliorante a été générée. Donc, il y a un nombre infini d’itérations infructueuses. On peut en déduire une limite quant à la taille du cadre de sonde,

$$\liminf_{k \rightarrow \infty} \Delta^k = 0,$$

et puisque  $\delta^k \leq \Delta^k$ ,

$$\liminf_{k \rightarrow \infty} \delta^k = 0.$$

Par un raisonnement similaire que pour l'analyse de convergence de GPS à la section 3.3.2, il existe une sous-suite d'itérations infructueuses  $\{k_i\}$  et un point  $x^*$  tels

$$\lim_{i \rightarrow \infty} \delta^{k_i} = 0, \quad \lim_{i \rightarrow \infty} x^{k_i} = x^*.$$

Cela implique que l'algorithme raffine le treillis autour de la solution courante  $x^*$  et en supposant que la fonction est localement Lipschitz autour de  $x^*$ , on en conclut que

$$f^\circ(x^*; d) \geq 0, \quad \forall d \in \mathbb{R}^n.$$

Les résultats de convergence sont donc similaires à la classe d'algorithme MADS originale.

## 4.6 Paramètres

Le cadre algorithmique de MADS laisse certaines libertés quant à la valeur de certains paramètres, ainsi que le présence de certaines étapes. La variante proposée PCA-MADS profite également de ces libertés. On reprend une liste des paramètres et choix à faire pour un algorithme MADS, ainsi que ceux supplémentaires issus de la variante PCA-MADS.

Les paramètres et les choix d'implémentation pour l'algorithme MADS général sont les suivants.

1. Initialisation :
  - paramètre de taille de cadre initial  $\Delta^0$ ,
  - matrice générateur positif  $D$ ,
  - paramètres d'ajustement du treillis  $\tau$ ,  $\omega^+$  et  $\omega^-$ ,
  - tolérance d'arrêt  $\epsilon$
2. Etape de recherche : nombre de points à évaluer et façon de les générer ;
3. Etape de sonde : nombre de directions ( $(n + 1)$  ou  $2n$  directions) et façon de les générer.

Lors de l'initialisation, des valeurs des paramètres sont proposées dans [8]. Pour l'étape de sonde, plusieurs implémentations sont proposées dans [8, 2, 62]. La façon de générer des points lors de l'étape de recherche repose sur une analyse en composante principale et une méthode d'optimisation. Les paramètres liés à cette étape sont repris dans la liste suivante :

- budget d'évaluations pour l'optimisation du sous-problème  $B$  ;
- choix de la dimension du sous-problème  $p$  ;
- nombre de points nécessaires pour effectuer l'étape de recherche  $N_{min}^{search}$  et la possibilité d'en générer pour compléter la *cache* ;
- nombre de points utilisés dans l'analyse en composante principale  $N_{pca}^{search}$  et la façon de les sélectionner ;
- méthode de construction du sous-problème à partir de l'analyse en composante principale et la façon d'évaluer la boîte noire ;
- paramètres inhérents à la méthode d'optimisation du sous-problème.

Une méthode de construction du sous-problème et d'évaluations de la boîte a été présentée aux sections précédentes.