HAL
open science

Métaheuristiques pour le flow-shop de permutation bi-objectif stochastique

Arnaud Liefooghe, Laetitia Jourdan, Matthieu Basseur, El-Ghazali Talbi

To cite this version:

Arnaud Liefooghe, Laetitia Jourdan, Matthieu Basseur, El-Ghazali Talbi. Métaheuristiques pour le flow-shop de permutation bi-objectif stochastique. Revue des Sciences et Technologies de l'Information - Série RIA : Revue d'Intelligence Artificielle, 2008, 22 (2), pp.183–208. inria-00269981

HAL Id: inria-00269981

https://inria.hal.science/inria-00269981

Submitted on 3 Apr 2008

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Métaheuristiques pour le flow-shop de permutation bi-objectif stochastique

Arnaud Liefooghe* — **Laetitia Jourdan*** — **Matthieu Basseur**** — **El-Ghazali Talbi ***

** LIFL – INRIA Lille-Nord Europe, Équipe-projet Dolphin*
Parc scientifique de la Haute-Borne, 40 av. Halley, 59650 Villeneuve d'Ascq, France
{Arnaud.Liefooghe, Laetitia.Jourdan, El-Ghazali.Talbi}@lifl.fr

*** LERIA, UFR Sciences*
2 Boulevard Lavoisier, 49045 Angers cedex 01, France
Matthieu.Basseur@univ-angers.fr

RÉSUMÉ. *Bien que les algorithmes évolutionnaires soient couramment utilisés pour résoudre des problèmes multi-objectifs d'une part, et stochastiques d'autre part, très peu de travaux ont été menés sur ces deux aspects simultanément. Par exemple, les problèmes d'ordonnancement sont habituellement traités sous une forme mono-objectif déterministe, alors qu'ils sont clairement multi-objectifs et qu'ils sont soumis à de nombreux facteurs d'incertitude. Dans cet article, nous présentons différentes approches pour résoudre des problèmes d'optimisation multi-objectif stochastiques et les appliquons à un problème d'ordonnancement de type flow-shop de permutation bi-objectif avec durées d'exécution aléatoires.*

ABSTRACT. *Although evolutionary algorithms are commonly used for solving multi-objective problems on the one hand and stochastic problems on the other hand, very few studies have investigated these two aspects simultaneously. For instance, scheduling problems are usually tackled in a single-objective deterministic form, whereas they are clearly multi-objective and they are subject to a wide range of uncertainty. In this paper, we present different approaches to solve stochastic multi-objective optimization problems and apply them to a bi-objective permutation flow-shop scheduling problem with random processing times.*

MOTS-CLÉS : *Métaheuristiques, algorithmes évolutionnaires, optimisation multi-objectif, incertitude, problèmes d'ordonnancement, durées d'exécution aléatoires*

KEYWORDS: *Metaheuristics, evolutionary algorithms, multi-objective optimization, uncertainty, scheduling problems, random processing times*

1. Introduction

Durant les vingt dernières années, beaucoup de travaux se sont penchés sur les problèmes d'optimisation multi-objectif, dont le but est de trouver un ensemble de solutions offrant différents compromis entre les critères considérés. Résoudre ce type de problème de façon exacte est souvent impossible en raison de la difficulté intrinsèque du problème ou du grand nombre d'optima. Ainsi, des méthodes approximatives telles que les métaheuristiques, et particulièrement les algorithmes évolutionnaires, sont couramment utilisées afin de trouver une bonne approximation de cet ensemble de solutions de compromis (Deb, 2001; Coello Coello *et al.*, 2002). Un algorithme évolutionnaire est une méthode d'optimisation stochastique itérative inspirée par l'évolution biologique des espèces. Une solution réalisable pour le problème étudié est vue comme un individu soumis à l'évolution. Un ensemble d'individus interviennent au sein de l'algorithme et constituent une population. Celle-ci s'améliore au fur et à mesure de générations jusqu'à la vérification d'un critère d'arrêt. À chaque génération, une succession d'opérateurs génétiques est appliquée aux individus afin de constituer la population de la génération suivante.

Par ailleurs, une grande partie des problèmes d'optimisation réels sont soumis à des incertitudes devant être prises en compte, tant au niveau de la (ou des) fonction(s) objectif qu'au niveau des paramètres environnementaux ou des variables de décision. Comme le montre l'état de l'art proposé par Jin et Branke (Jin *et al.*, 2005), beaucoup de méthodes de résolution dédiées aux problèmes d'optimisation sous incertitude sont également des algorithmes évolutionnaires. Malgré l'importance de l'optimisation multi-objectif et de l'optimisation sous incertitude, très peu de travaux consacrés à l'optimisation multi-objectif sous incertitude existent à ce jour. En outre, bien qu'elles soient généralement adaptables au cas combinatoire, les études existantes n'ont été expérimentées que sur des problèmes continus ou sur des fonctions de test. Or, il n'est pas évident que les performances soient identiques pour des problèmes combinatoires. Par exemple, les problèmes d'ordonnancement sont des problèmes combinatoires habituellement étudiés sous une forme mono-critère déterministe. Pourtant, bien qu'il existe de nombreux problèmes d'ordonnancement multi-objectifs d'une part (T'kindt *et al.*, 2002), et stochastiques d'autre part (Billaut *et al.*, 2005), aucune étude alliant ces deux aspects simultanément existe à ce jour.

Dans cet article, nous présentons divers modèles de représentation de l'incertitude pour un problème d'ordonnancement de type flow-shop de permutation. Nous proposons également plusieurs méthodes d'optimisation multi-objectif permettant de manipuler un type quelconque de distribution de probabilité, que ce soit sur les paramètres ou sur les fonctions objectif. Ces méthodes sont des algorithmes évolutionnaires multi-objectifs basés sur un indicateur de qualité proposés par Zitzler et Künzli (Zitzler *et al.*, 2004) pour le cas déterministe. Ces algorithmes sont alors appliqués à un problème de flow-shop de permutation bi-objectif avec durées d'exécution incertaines. À notre connaissance, c'est la première fois qu'un tel problème est étudié sous une forme multi-critère. Le reste de l'article est organisé comme suit. La section 2 est consacrée à l'optimisation multi-objectif et à l'optimisation multi-objectif sous incertitude. Dans

la section 3, un problème de flow-shop bi-objectif avec temps d'exécution stochastiques est formulé. La section 4 présente trois algorithmes évolutionnaires dédiés à la résolution de problèmes d'optimisation multi-objectif stochastiques et leur application sur le problème d'ordonnancement étudié ici. La section 5 donne des résultats expérimentaux. Enfin, la dernière section conclut le document et donne quelques perspectives de recherche dans ce domaine.

2. Préliminaires

Cette section donne les notions, définitions et notations fondamentales pour l'optimisation multi-objectif. Puis, nous nous concentrons plus particulièrement sur le cas de l'optimisation multi-objectif basé sur un indicateur de qualité, et présentons les principaux travaux dédiés à la prise en compte de l'incertitude en optimisation évolutionnaire.

2.1. *Optimisation multi-objectif*

L'*optimisation multi-objectif* vise à optimiser plusieurs composants d'un vecteur de fonctions coût, chaque composant correspondant à un objectif. Le processus d'optimisation cherche donc à minimiser ou à maximiser plusieurs critères simultanément. La difficulté réside dans le fait que ces critères sont souvent antagonistes, *i.e.* lorsque l'un d'eux est amélioré, les autres sont généralement dégradés. Un problème d'optimisation multi-objectif (*Multi-objective Optimization Problem*, ou *MOP*) peut être formulé ainsi :

$$(MOP) = \begin{cases} \text{“min”} & f(x) = (f_1(x), f_2(x), \ldots, f_n(x)) \\ \text{tel que} & x \in X \end{cases} \qquad [1]$$

où $f_1, f_2, \ldots, f_n$ sont les $n \geq 2$ fonctions objectif à optimiser, $x = (x_1, x_2, \ldots, x_k)$ est un vecteur de k variables de décision, et X représente l'ensemble des solutions réalisables dans l'*espace décisionnel*. Dans le cas combinatoire, un vecteur de décision $x \in X$ a un nombre fini de valeurs possibles. Soit Z l'ensemble des points réalisables dans l'*espace des objectifs*. À chaque vecteur de décision $x \in X$ est affecté un vecteur objectif $z \in Z$ basé sur le vecteur de fonction coût $f : X \rightarrow Z$ avec $z = z_1, z_2, \ldots, z_n = f(x) = f_1(x), f_2(x), \ldots, f_n(x)$. Notez que tout au long de l'article nous considérons que les valeurs objectif sont normalisées. Pour ce faire, chacune des fonctions objectif est remplacée par une fonction normalisée correspondante dont les valeurs sont comprises dans l'intervalle $[0, 1]$. Sans perte de généralité, nous supposons ici que $Z \subseteq \mathbb{R}^n$ et que les n fonctions objectif sont à minimiser.

Un vecteur objectif $z \in Z$ *domine faiblement* un vecteur objectif $z' \in Z$ si et seulement si $\forall i \in [1..n], z_i \leq z'_i$. On notera alors $z \succeq z'$. Un vecteur objectif $z \in Z$ *domine* un vecteur objectif $z' \in Z$ si et seulement si $z \succeq z'$ et $\exists j \in [1..n]$ tel que $z_j < z'_j$. On notera alors $z \succ z'$. Une solution $x \in X$ est *Pareto optimale* s'il n'existe

pas de solution $x' \in X$ telle que $f(x') \succ f(x)$. Le but de l'optimisation est de trouver l'ensemble des solutions Pareto optimale, ou ensemble *Pareto optimal* $\mathcal{PO}$. Cet ensemble est défini ainsi :

$$\mathcal{PO} = \{x \in X | \nexists x' \in X, f(x') \succ f(x)\} \quad [2]$$

Générer la totalité des solutions Pareto optimales est en général impossible (en raison du temps de calcul et de la complexité du problème). Les algorithmes évolutionnaires sont couramment utilisés pour trouver une approximation de cet ensemble qui soit de bonne qualité en termes de convergence et de diversité. Le lecteur se référera à (Deb, 2001; Coello Coello *et al.*, 2002) pour plus de détails sur l'optimisation évolutionnaire multi-objectif.

2.2. *Optimisation basée sur un indicateur de qualité*

Il existe différentes interprétations de ce qu'est une bonne approximation de l'ensemble Pareto optimal. La définition de qualité d'approximation dépend fortement des préférences du décideur et du scénario de l'optimisation. En se basant sur cette observation, Zitzler et Künzli (Zitzler *et al.*, 2004) proposent de définir le but de l'optimisation à l'aide d'un indicateur binaire de qualité $I : \Omega \times \Omega \rightarrow \mathbb{R}$ (où Ω représente l'ensemble de tous les ensembles possibles de solutions). La valeur $I(A, B)$ quantifie la différence de qualité entre deux ensembles A et $B \in \Omega$. Maintenant, si R dénote un ensemble de référence (qui peut être $\mathcal{PO}$ ou n'importe quel autre ensemble), le but global de l'optimisation peut être formulé comme suit :

$$\text{argmin}_{A \in \Omega}\ I(A, R) \quad [3]$$

Considérons une population P comme un ensemble de solutions de X. La fonction de $fitness$ d'un algorithme évolutionnaire multi-objectif mesure l'utilité d'une solution $x \in P$ par rapport au but de l'optimisation, ici donné par l'équation 3. Comme proposé dans l'algorithme IBEA (Indicator-Based Evolutionary Algorithm) (Zitzler *et al.*, 2004), nous définissons la valeur de $fitness$ d'une solution $x \in P$ comme la mesure de la perte de qualité de la population P si x en était supprimée :

$$fitness(x) = I(x, P \setminus \{x\}) \quad [4]$$

L'indicateur I, qui peut être choisi en fonction des préférences du décideur, est donc directement utilisé pour quantifier la qualité d'une solution durant les processus de sélection de l'algorithme évolutionnaire. Un des avantages de l'optimisation basée sur un indicateur de qualité est que l'indicateur utilisé peut prendre en compte la notion de diversité. Ainsi, aucun mécanisme de diversification supplémentaire n'est généralement nécessaire au sein de l'algorithme évolutionnaire.

Prenons l'exemple de l'indicateur ϵ-additif ($I_{\epsilon+}$) proposé dans (Zitzler *et al.*, 2003; Zitzler *et al.*, 2004). Comme illustré sur la figure 1, $I_{\epsilon+}$ mesure la distance minimale

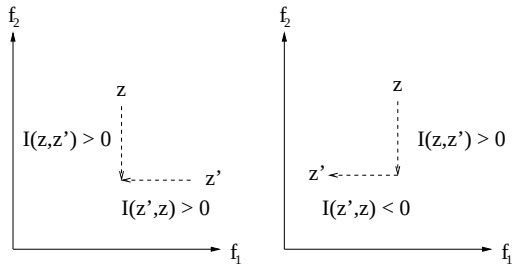

Figure 1. *Illustration de l'indicateur ϵ-additif ($I_{\epsilon+}$)*

par laquelle un point $z \in Z$ doit ou peut être translaté dans l'espace objectif pour dominer faiblement un autre point $z' \in Z$. Pour un problème où les n objectifs sont à minimiser, il peut être défini comme suit :

$$I_{\epsilon+}(z, z') = \max_{i \in \{1,\ldots,n\}} (z_i - z'_i) \qquad [5]$$

Ainsi, dans le cas déterministe, pour évaluer la qualité d'une solution $x \in X$ par rapport à une solution $x' \in X$ à l'aide de $I_{\epsilon+}$, il suffit d'utiliser le vecteur objectif associé à chacune des solutions (nous verrons que ce sera moins trivial pour le cas stochastique) :

$$I_{\epsilon+}(x, x') = I_{\epsilon+}(f(x), f(x')) \qquad [6]$$

Ensuite, pour mesurer la qualité d'une solution $x \in X$ par rapport à une population P à l'aide de $I_{\epsilon+}$ (et donc calculer la valeur de $fitness$ de x), plusieurs stratégies existent pour le cas déterministe. Un exemple couramment employé (Zitzler *et al.*, 2004) est donné dans l'équation suivante, où κ est un facteur de redimensionnement.

$$I_{\epsilon+}(x, P \setminus \{x\}) = \sum_{x' \in P \setminus \{x\}} -e^{-I_{\epsilon+}(x',x)/\kappa} \qquad [7]$$

Cette approche additive a tendance à amplifier l'influence des solutions non dominées sur les autres solutions. D'autres exemples d'indicateurs de qualité pouvant être utilisés au sein d'IBEA sont donnés dans (Hansen *et al.*, 1998; Zitzler *et al.*, 2003; Zitzler *et al.*, 2004).

2.3. *Optimisation évolutionaire en environnement incertain*

Dans leur état de l'art sur les méthodes évolutionnaires dédiées à l'optimisation sous incertitude, Jin et Branke (Jin *et al.*, 2005) classifient les types d'incertitude en quatre catégories : (i) fonction objectif bruitée, (ii) variables ou paramètres environnementaux soumis à des perturbations ou changeant après le processus d'optimisation, (iii) fonction objectif approximée, (iv) fonction objectif évoluant au cours

du temps. Les auteurs remarquent que peu d'études ont été réalisées sur l'optimisation multi-objectif en environnement incertain. Ainsi, Hughes (Hughes, 2001) et Teich (Teich, 2001) ont suggéré d'étendre le concept de dominance Pareto pour le cas stochastique en remplaçant le rang d'une solution par sa probabilité d'être dominée ; mais chacune de ces deux études émet une hypothèse sur la distribution de probabilité suivie par les fonctions objectif. Dans (Babbar *et al.*, 2003), les auteurs présentent une autre méthode de ranking, basée sur une valeur moyenne par objectif et sur la variance de ces valeurs, calculées à partir d'un échantillon d'évaluations. De même, Deb et Gupta (Deb *et al.*, 2005) proposent d'appliquer des méthodes classiques d'optimisation multi-objectif en utilisant une valeur moyenne par objectif. Plus récemment, Goh et Tan ont étudié l'impact de fonctions objectif bruitées sur des algorithmes évolutionnaires multi-objectifs et ont proposé plusieurs dispositifs pour manipuler le bruit (Goh *et al.*, 2007). Enfin, Basseur et Zitzler (Basseur *et al.*, 2006) ont récemment étendu le concept d'optimisation par indicateur de qualité afin de prendre l'incertitude en considération. Les méthodes proposées dans la littérature ont majoritairement été testées sur des problèmes continus et où l'incertitude se trouve sur les fonctions objectif. Ainsi, à notre connaissance, c'est la première fois qu'un problème d'optimisation combinatoire multi-objectif ayant des paramètres environnementaux stochastiques est étudié. Néanmoins, tous les concepts introduits dans cet article sont applicables au cas continu et à tout type d'incertitude, que ce soit sur les paramètres ou sur les fonctions objectif.

3. Le problème d'ordonnancement de type flow-shop de permutation bi-objectif avec durées d'exécutions stochastiques

Le *flow-shop* est l'un des problèmes d'ordonnancement les plus étudiés de la littérature. La majorité des travaux qui lui sont consacrés le considère sous une forme déterministe mono-objectif et vise généralement à minimiser la date de fin de l'ordonnancement (le *makespan*). Bien que le flow-shop soit largement étudié sous une forme stochastique d'une part (Billaut *et al.*, 2005) ou multi-objectif d'autre part (T'kindt *et al.*, 2002), aucune étude alliant ces deux aspects simultanément n'a été menée à l'heure actuelle. Dans cette section, nous présentons un modèle bi-objectif déterministe du flow-shop. Puis, nous nous intéressons aux différentes sources d'incertitude auxquelles il peut être soumis. Nous nous focalisons ensuite sur le flow-shop avec temps d'exécution incertains et proposons des modèles stochastiques pour ce problème. Bien que ce travail se concentre sur le flow-shop, il est généralisable à tout type de problèmes stochastiques et aisément adaptable à d'autres problèmes d'ordonnancement.

3.1. *Modèle déterministe*

Le flow-shop consiste à ordonnancer N jobs $J_1, J_2, \ldots, J_N$ sur M machines $M_1, M_2, \ldots, M_M$. Une machine ne peut traiter qu'un seul job à la fois. Les machines

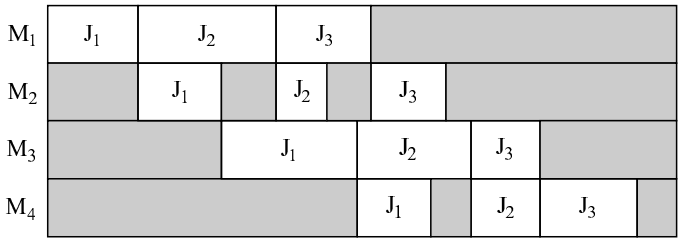

Figure 2. *Exemple d'ordonnancement pour un flow-shop de permutation à 3 jobs et 4 machines*

sont donc ici comparables à une chaîne de production. Un job J_i est composé de M tâches consécutives $t_{i1}, t_{i2}, \ldots, t_{iM}$, où t_{ij} est la $j$ième tâche du job J_i, nécessitant la machine M_j. À chaque tâche t_{ij} est associé une durée d'exécution p_{ij}, et à chaque job J_i est associée une date due d_i (la date de fin souhaitée du job). Comme illustré dans la figure 2, nous nous focalisons ici sur le flow-shop de permutation, où la séquence des jobs est identique et unidirectionnelle sur chaque machine (*i.e.* les jobs doivent tous être exécutés dans le même ordre $M_1, M_2, \ldots, M_M$). Ainsi, pour une instance de N jobs, il existe $N!$ solutions réalisables dans l'espace de décision. Beaucoup de critères peuvent être envisagés lors de l'ordonnancement de tâches sur plusieurs machines. Le flow-shop considéré ici consiste à minimiser à la fois le makespan (C_{max}) et le retard total ($\overline{T}$), qui sont deux des objectifs les plus étudiés de la littérature. Pour chaque tâche t_{ij} ordonnancée à un temps s_{ij}, les deux fonctions objectif peuvent être calculées comme suit :

$$C_{max} = \max_{i \in \{1,\ldots,N\}} [s_{iM} + p_{iM}] \qquad [8]$$

$$\overline{T} = \sum_{i=1}^{N} [\max(0, s_{iM} + p_{iM} - d_i)] \qquad [9]$$

D'après la notation établie par Graham et al. (Graham *et al.*, 1979), le problème d'ordonnancement de type *flow-shop de permutation bi-objectif* étudié ici est symbolisé par $F/perm, d_i/(C_{max}, \overline{T})$. Le lecteur est invité à consulter (Nagar *et al.*, 1995; T'kindt *et al.*, 2002; Landa Silva *et al.*, 2004) pour plus d'informations sur les prolèmes d'ordonnancement multicritère.

3.2. *Sources d'incertitude*

Dans les situations réelles d'ordonnancement, l'incertitude provient principalement des variables d'environnement et peut donc être classée dans la seconde catégorie de la classification proposée par Jin and Branke (Jin *et al.*, 2005) ; *i.e.* variables ou

paramètres environnementaux soumis à des perturbations. En effet, rares sont les paramètres associés aux problèmes d'ordonnancement qui sont dépourvus d'incertitude. Les solutions sont sensibles à ces perturbations, et il s'avère donc indispensable de tenir compte des "zones d'ignorances" et des "à-peu-près" que laissent entrevoir certains paramètres (Roy, 2005). Cette incertitude peut provenir de différentes sources, telles que des variations sur les dates dues, des pannes de machines, l'ajout ou la suppression inattendue de jobs, des temps d'exécution variables, etc. D'après la littérature du domaine, il apparaît que pour le flow-shop de permutation considéré ici, l'incertitude est susceptible de provenir essentiellement des dates dues et des temps d'exécution.

Tout d'abord, dans le modèle déterministe, la date due d'un job J_i est donnée par une constante d_i. Pourtant, déterminer ce nombre sans ambiguïté paraît difficile. Il semblerait donc plus naturel de donner un intervalle $[d_i^1, d_i^2]$ durant lequel la satisfaction pour la complétion du job J_i décroît. De plus, une date de fin d_i peut subir des modifications dynamiques du fait qu'un job sans trop d'importance à un moment peut devenir plus important à un autre moment (et vice versa). Par ailleurs, un temps d'exécution peut varier d'une exécution à une autre du fait d'événements extérieurs. Ainsi, le temps d'exécution p_{ij} associé à une tâche t_{ij} est rarement constant dans la réalité. Il paraît donc évident qu'aucun paramètre ne peut être vu comme une donnée exacte et précise et que des modèles non déterministes doivent être élaborés pour résoudre un problème d'ordonnancement. Pour cela, nous décidons d'adopter une approche proactive où les temps d'exécutions sont considérés comme incertains et sont modélisés à l'aide de variables aléatoires.

3.3. *Modèles stochastiques*

À notre connaissance, le problème du flow-shop stochastique n'a jamais été étudié de façon multicritère. Pourtant, dans un cas réel, dès qu'un historique des temps d'exécution précis et valide des différentes tâches est disponible, il est relativement aisé d'obtenir la distribution de probabilité associée aux données et, ainsi, la loi de probabilité correspondante. Suite à une analyse, nous proposons ici quatre distributions de probabilité génériques pouvant être suivies par une durée de traitement aléatoire. Bien sûr, une analyse statistique rigoureuse, basée sur des données réelles, est impérative pour déterminer la loi de distribution exacte associée à un temps d'exécution pour un problème concret d'ordonnancement. Un état de l'art sur le problème du flow-shop monocritère avec durées d'exécution aléatoires est proposé dans (Gourgand *et al.*, 2005).

3.3.1. *Loi uniforme*

Tout d'abord, il se peut qu'un temps d'exécution p_{ij} soit compris entre deux valeurs a et b, la répartition étant uniforme entre ces deux valeurs (voir figure 3). p_{ij} suit alors une *loi uniforme* sur l'intervalle $[a, b]$ et admet pour densité :

$$f(x) = \begin{cases} \frac{1}{b-a} & \text{si } x \in [a, b] \\ 0 & \text{sinon} \end{cases} \qquad [10]$$

Bien que peu d'exemples de temps d'exécution suivant une loi uniforme aient été trouvés au sein d'applications réelles, ce modèle a été conservé afin de fournir un modèle simplifié de la réalité. Il a, par exemple, été utilisé dans (Kouvelis *et al.*, 2000; Gourgand *et al.*, 2005). Le seul exemple effectif envisagé est celui où la durée d'un traitement est dépendante du placement initial de la machine sur laquelle il est réalisé.

3.3.2. *Loi exponentielle*

Une durée de traitement p_{ij} est susceptible de suivre une loi exponentielle $\mathcal{E}(\lambda, a)$, où λ et a sont deux paramètres positifs (voir figure 3). Sa densité de probabilité est alors :

$$f(x) = \begin{cases} \lambda\, e^{-\lambda(x-a)} & \text{si } x \geq a \\ 0 & \text{sinon} \end{cases} \qquad [11]$$

Les distributions exponentielles sont couramment utilisées pour modéliser les évènements aléatoires qui risquent de se produire avec incertitude. C'est typiquement le cas en présence d'*aléas* ; c'est-à-dire lorsqu'une valeur est associée à un paramètre, mais que celle-ci est susceptible d'être modifiée par certains évènements inattendus (Sanlaville, 2005). Ce type d'incertitude est constaté dans les cas de panne, ou, plus généralement, dès lors qu'une machine nécessite une intervention humaine : réparation, entretien, recharge de composants nécessaires à son bon fonctionnement, réinstallation d'un objet mal placé, etc. La durée du traitement s'écoule alors jusqu'à la fin de l'intervention (ou jusqu'à la réparation de la machine endommagée). Des temps de traitement ont été modélisés à l'aide d'une loi exponentielle dans (Makino, 1965; Talwar, 1967; Cunningham *et al.*, 1973; Ku *et al.*, 1986; Gourgand *et al.*, 2005).

3.3.3. *Loi normale*

Il s'avère également possible qu'un temps d'exécution p_{ij} suive une *loi normale* $\mathcal{N}(\mu, \sigma)$, où μ est l'Espérance et $\sigma^2 > 0$ la Variance de p_{ij} (voir figure 3). Dans ce cas, sa densité de probabilité est donnée par :

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}}\, exp\Big(-\frac{1}{2}\Big(\frac{x-\mu}{\sigma}\Big)^2\Big) \qquad [12]$$

Ce genre de cas est particulièrement fréquent en présence de facteurs humains. En effet, il semble peu probable qu'un ouvrier accomplisse une certaine tâche en un temps parfaitement identique à chaque exécution. Ce dernier la réalisera plus ou moins rapidement selon divers critères tels que son degré d'attention, sa forme physique, etc. Par ailleurs, il se peut qu'un traitement dépende de facteurs inconnus, incontrôlables ou décrits de manière imprécise ou ambigue par l'analyste. Prenons l'exemple d'une usine chimique (Fortemps, 1997). Un traitement est, dans ce cas, une réaction chimique entre plusieurs composants. Or la durée d'une réaction chimique n'est pas fixe, elle peut varier selon la température, la pression, la qualité des composants, et d'autres paramètres dont certains peuvent être complètement inconnus. Il est donc impossible de vouloir contrôler toutes ces dépendances. Les temps d'exécution observés varient alors selon une loi normale. Cette modélisation a, par exemple, été utilisée dans (Wang *et al.*, 2005; Gourgand *et al.*, 2005).

3.3.4. *Loi log-normale*

Une variable aléatoire x suit une *loi log-normale* de paramètres (μ, σ) si $\log x$ suit la loi $\mathcal{N}(\mu, \sigma)$ (voir figure 3). Sa densité de probabilité est alors :

$$f(x) = \begin{cases} \frac{1}{x\sigma\sqrt{2\pi}} \frac{1}{x} exp(-\frac{1}{2}(\frac{\log x - \mu}{\sigma})^2) & \text{si } x > 0 \\ 0 & \text{sinon} \end{cases} \quad [13]$$

La distribution log-normale est couramment utilisée pour modéliser l'influence de variables environnementales incontrôlables. Dans notre cas, un temps d'exécution modélisé à l'aide d'une loi log-normale prendra en compte la totalité des incertitudes observées simultanément (panne, présence de facteurs humains, paramètres inconnus ou incontrôlables). Par exemple, cette modélisation a été utilisée dans (Maddox *et al.*, 1996; Dauzère-Pérès *et al.*, 2005).

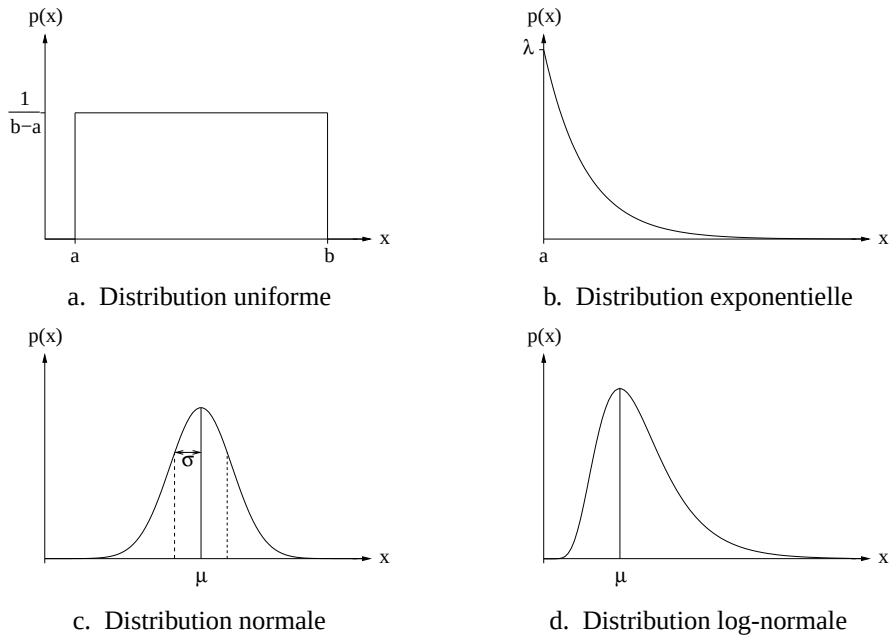

Figure 3. *Exemples de fonctions de densité de probabilité susceptibles de modéliser une durée d'exécution aléatoire : (a) loi uniforme, (b) loi exponentielle, (c) loi normale (d) loi log-normale*

4. Algorithmes évolutionnaires basés sur un indicateur de qualité pour les problèmes d'optimisation multi-objectif stochastiques

Dans cette section, nous présentons diverses extensions de l'optimisation évolutionnaire multi-objectif basée sur un indicateur de qualité, initialement proposée dans (Zitzler *et al.*, 2004) et présentée en section 2.2, pour la prise en compte de l'incertitude. Ainsi, trois algorithmes évolutionnaires dédiés à la résolution de problèmes d'optmisation multi-objectif stochastiques sont proposés. Bien sûr, ces méthodes sont génériques et peuvent donc être facilement applicables à tout type de problèmes d'optimisation, et pas uniquement au problème de flow-shop étudié dans cet article.

4.1. *Prise en compte de l'incertitude*

Dans le cas déterministe, à chaque solution x appartenant à l'ensemble des solutions réalisables de l'espace de décision X est assigné exactement un vecteur objectif z appartenant à l'ensemble des points réalisables de l'espace objectif Z sur la base du vecteur de fonctions objectif f. Ainsi, $f(x)$ définie l'évaluation réelle d'une solution $x \in X$. Au contraire, dans le cas stochastique, à chaque fois qu'une solution $x \in X$ est évaluée, le vecteur objectif résultant est différent. Donc, le vecteur de fonctions f ne calcule pas une image déterministe de X vers Z, car un ensemble potentiellement infini de vecteurs objectif est maintenant associé à une solution $x \in X$. Plus grand est le degré d'incertitude, plus large sera la variance parmi les vecteurs objectif obtenus par de multiples évaluations d'une solution. C'est pourquoi, comme proposé dans (Basseur *et al.*, 2006), à chaque solution $x \in X$, nous associons une variable aléatoire F prenant ses valeurs dans Z. $F(x)$ donne une évaluation de la solution $x \in X$ dans l'espace objectif. Dans cet article, contrairement à d'autres approches (Jin *et al.*, 2005), nous ne considérons pas qu'il existe une évaluation réelle perturbée par le bruit associée à une solution, mais un ensemble d'évaluations "probables". Nous considérons que l'évaluation réelle d'une solution est absolument indéterminée avant la fin du processus d'optimisation. Nous ne faisons aucune hypothèse sur une quelconque distribution de probabilité associée aux paramètres ou aux fonctions objectif, car l'espace d'évaluation potentiel d'une solution est généralement inconnu et peut varier pour chaque solution. De ce fait, nous tentons d'estimer l'espace d'évaluation potentiel d'une solution à l'aide d'un ensemble fini d'échantillons de ses évaluations. Pour cela, nous définissons $S(x) = \{z^{(1)}, z^{(2)}, \ldots, z^{(s)}\}$ comme un ensemble fini d'évaluations indépendantes pour une solution $x \in X$, c'est à dire un ensemble de vecteurs objectif associés à x. Ainsi, la probabilité qu'une solution $x \in X$ soit représentée par un vecteur objectif $z \in Z$ peut être estimée par la fréquence relative de z dans $S(x)$:

$$\hat{P}(F(x) = z) = \sum_{z' \in S(x), z'=z} \frac{1}{|S(x)|} \qquad [14]$$

4.2. *Algorithmes proposés*

Nous présentons ici trois algorithmes évolutionnaires dédiés à la résolution de problèmes d'optimisation multi-objectif stochastiques, basés sur trois indicateurs de qualité pouvant être utilisés au sein d'IBEA (Zitzler *et al.*, 2004) spécifiquement élaborés pour le cas stochastique. Afin de comparer deux vecteurs objectif, nous utiliserons l'indicateur ϵ-additif (Zitzler *et al.*, 2003; Zitzler *et al.*, 2004), introduit dans l'équation 5. Cet indicateur se montre particulièrement efficace sur différents types de problèmes déterministes (Zitzler *et al.*, 2004), et a obtenu de meilleurs résultats que l'indicateur binaire hypervolume proposé dans (Zitzler *et al.*, 2004) sur la version déterministe du problème abordé ici (Liefooghe, 2006).

4.2.1. *Estimation basée sur une simple évaluation*

Le premier indicateur stochastique proposé, I^1, consiste à conserver l'approche utilisée dans le cas déterministe. Une solution est évaluée une seule fois et la qualité de cette solution est estimée en utilisant cette seule évaluation (voir la figure 4). Ainsi, $|S(x)| = 1$ pour chaque solution $x \in X$. La qualité d'une solution $x \in X$ par rapport à une autre solution $x' \in X$ est mesurée ainsi :

$$I^1(x, x') = I(z^{(1)}, z'^{(1)}) \quad \text{tel que} \quad S(x) = \{z^{(1)}\}, S(x') = \{z'^{(1)}\} \qquad [15]$$

Ainsi, $\hat{P}(F(x) = z) = 1$ et $\hat{P}(F(x') = z') = 1$. Par exemple, la $I^1_{\epsilon+}$-valeur de $x \in X$ par rapport à $x' \in X$ se calcule :

$$I^1_{\epsilon+}(x, x') = I_{\epsilon+}(z^{(1)}, z'^{(1)}) \quad \text{tel que} \quad S(x) = \{z^{(1)}\}, S(x') = \{z'^{(1)}\} \quad [16]$$

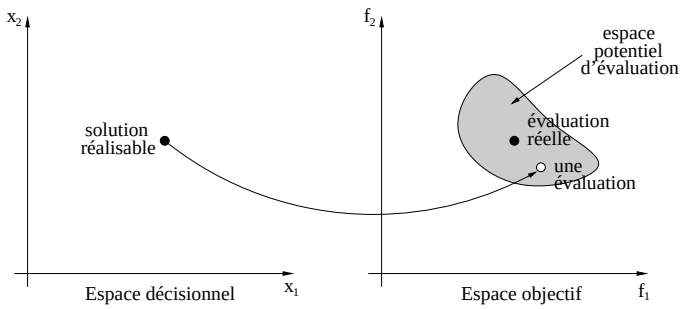

Figure 4. I^1 *: une seule évaluation est utilisée pour approximer l'évaluation d'une solution*

Comme proposé dans (Zitzler *et al.*, 2004), la I^1-valeur d'une solution $x \in X$ par rapport au reste de la population P (et donc la valeur de *fitness* de x, voir l'équation 4) est déterminée ainsi :

$$I^1(x, P \setminus \{x\}) = \sum_{x' \in P \setminus \{x\}} -e^{-I^1(x',x)/\kappa} \qquad [17]$$

où $\kappa > 0$ est un facteur de redimensionnement. À l'heure actuelle, la plupart des méthodes procèdent de cette façon, car elles considèrent que les paramètres sont constants et ne tiennent pas compte de l'incertitude. L'avantage de cette méthode est son faible coût de calcul, mais l'erreur d'estimation peut être grande, la seule évaluation utilisée n'étant pas nécessairement représentative de l'espace potentiel des évaluations d'une solution.

4.2.2. *Estimation basée sur le vecteur objectif moyen*

La seconde approche, I^{avg}, se base sur une idée communément suivie dans le cas mono-objectif et suggérée par quelques études de problèmes multi-objectifs stochastiques (comme, par exemple, dans (Babbar *et al.*, 2003; Deb *et al.*, 2005)). Comme illustré sur la figure 5, elle consiste à calculer un vecteur objectif moyen z^{avg} à partir des évaluations $S(x)$ d'une solution $x \in X$:

$$z^{avg} = \begin{bmatrix} z_1^{avg} \\ z_2^{avg} \\ \vdots \\ z_n^{avg} \end{bmatrix} \quad \text{tel que} \quad z_i^{avg} = \frac{1}{|S(x)|} \sum_{z \in S(x)} z_i, \ \forall i \in 1, 2, \ldots, n \qquad [18]$$

Ensuite, une approche déterministe classique est appliquée sur le vecteur objectif moyen associé à chaque solution $x \in X$ telle que dans l'équation suivante, où z^{avg}

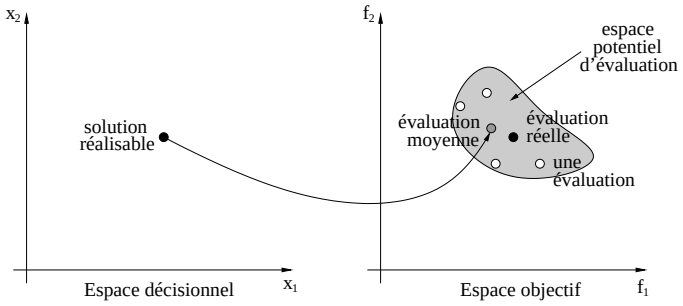

Figure 5. I^{avg} *: la moyenne des valeurs des évaluations est utilisé pour approximer la qualité d'une solution*

est le vecteur objectif moyen associé à x, et z'^{avg} est le vecteur objectif moyen associé à x'.

$$I^{avg}(x, x') = I(z^{avg}, z'^{avg}) \qquad [19]$$

Ainsi, de même que précédemment, la I^{avg}-valeur d'une solution $x \in X$ par rapport au reste de la population P peut se formuler ainsi :

$$I^{avg}(x, P \setminus \{x\}) = \sum_{x' \in P \setminus \{x\}} -e^{-I^{avg}(x',x)/\kappa} \qquad [20]$$

Cette méthode possède également l'avantage d'avoir un faible coût de calcul si l'évaluation d'une solution dans l'espace objectif n'est pas trop coûteuse (ce qui est le cas pour notre problème). Pourtant, des pertes d'information peuvent survenir en utilisant cette estimation moyenne, telles que la distribution potentielle des points dans l'espace objectif.

4.2.3. *Estimation probabiliste*

Le troisième et dernier indicateur stochastique consiste à estimer la qualité d'une solution de façon probabiliste. Ici, nous considérons qu'une probabilité de distribution arbitraire est associée à chaque solution dans l'espace objectif (voir la figure 6). Cette approche, dénotée $I^{\widehat{eiv}}$, réside principalement en une modification de la procédure d'affectation des performances telle que proposée dans (Basseur *et al.*, 2006) pour le cas spécifique de l'indicateur ϵ-additif.

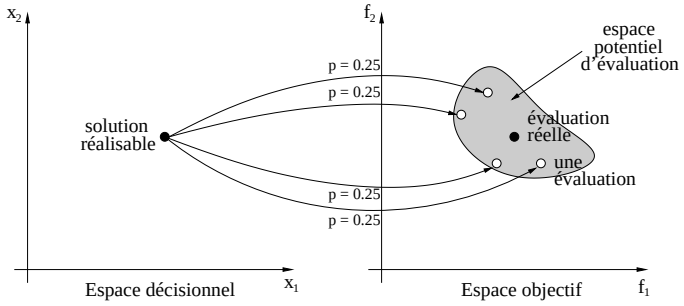

Figure 6. $I^{\widehat{eiv}}$ *: la qualité d'une solution est estimée de façon probabiliste, un indicateur de qualité étant associé pour chaque évaluation*

Tout d'abord, pour chaque évaluation d'une solution, nous allons calculer une estimation de l'espérance de sa $I_{\epsilon+}$-valeur par rapport à la population P. Soit une évaluation $z^\star$ telle que $z^\star \in S(x^\star)$ et $x^\star \in P$. Nous allons considérer toutes les paires (x, z) telles que $x \in P \setminus \{x^\star\}$ et $z \in S(x)$, et les trier dans l'ordre croissant selon les valeurs $I_{\epsilon+}(z^\star, z)$. Supposons que l'ordre obtenu soit $(x^1, z^1), (x^2, z^2), \ldots, (x^l, z^l)$.

Alors, l'estimation de l'espérance mathématique de la $I_{\epsilon+}$-valeur de l'évaluation $z^\star$ par rapport à la population P est définie ainsi :

$$\begin{aligned}\hat{E}(I_{\epsilon+}(z^\star, F(P))) &= I_{\epsilon+}(z^\star, z_1) \,.\, \hat{P}(F(x_1) = z_1) \\ &+ I_{\epsilon+}(z^\star, z_2) \,.\, \hat{P}(F(x_2) = z_2 \,|\, F(x_1) \neq z_1) \\ &+ \ldots \\ &+ I_{\epsilon+}(z^\star, z_l) \,.\, \hat{P}(F(x_l) = z_l \,|\, \forall_{1 \leq i < l}\, F(x_i) \neq z_i)\end{aligned} \quad [21]$$

Notez qu'il n'est pas nécessaire de calculer les l sommes et que le temps de calcul peut donc être réduit. Ensuite, la $I^{\widehat{eiv}}$-valeur d'une solution $x \in X$ par rapport à la population P est calculée comme suit :

$$I^{\widehat{eiv}}(x, P \setminus \{x\}) = \frac{1}{|S(x)|} \sum_{z \in S(x)} \hat{E}(I(z, F(P \setminus \{x\})) \quad [22]$$

La complexité en temps du calcul de la $I_{\epsilon+}^{\widehat{eiv}}$-valeur d'une solution est de l'ordre de $\mathcal{O}(n(|P|.s)^2 \log(|P|.s))$, où $|P|$ est la taille de la population, n est le nombre d'objectifs et s est le nombre d'évaluations par solution.

4.3. *Implémentation*

Pour développer les algorithmes découlant des indicateurs stochastiques présentés dans la section précédente (que nous allons dénoter $IBEA^1$, $IBEA^{avg}$ et $IBEA^{\widehat{eiv}}$), nous avons utilisé une plate-forme dédiée à la conception de métaheuristiques pour l'optimisation multi-objectif : ParadisEO-MOEO[1] (Liefooghe *et al.*, 2007). Tout d'abord, il a été nécessaire d'étendre cette plate-forme en définissant une notion de vecteur objectif pour les problèmes stochastiques. Ensuite, nous avons développé les méthodes proposées en suivant le même schéma que pour les méthodes déterministes déjà implémentées. Tous ces nouveaux concepts seront bientôt disponibles au sein de la plate-forme ParadisEO-MOEO. Les trois approches diffèrent les unes des autres de par l'indicateur de qualité utilisé. Les points communs des trois algorithmes sont :

– *représentation* : permutation de taille N (où N est le nombre de jobs) ;

– *initialisation de la population* : individus générés aléatoirement ;

– *sélection pour la reproduction* : tournoi binaire déterministe ;

– *opérateur de croisement* : deux points (Ishibuchi *et al.*, 1998) (voir Figure 7) ;

– *opérateur de Mutation* : insertion (Ishibuchi *et al.*, 1998) (voir Figure 8) ;

– *sélection pour le remplacement* : élitiste (après avoir supprimer les doublons) ;

– *critère d'arrêt* : nombre de générations.

1. ParadisEO est disponible sur le web à

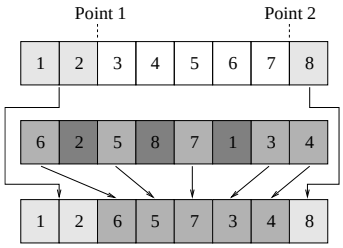

Figure 7. *Opérateur de croisement*

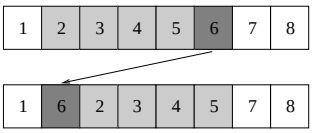

Figure 8. *Opérateur de mutation*

5. Résultats

5.1. *Jeux de données*

Pour tester nos algorithmes, nous proposons différents jeux de données[2] construits à partir des instances proposées par Taillard pour le flow-shop mono-objectif (Taillard, 1993).

5.1.1. *Jeux de données déterministes bi-objectif*

Tout d'abord, nous avons étendu les instances de Taillard (Taillard, 1993) pour le cas multi-objectif déterministe en ajoutant une date due à chacun des jobs. Ces dates dues ont été générées de façon aléatoire entre $\overline{p} \times M$ et $\overline{p} \times (N + M - 1)$, où M représente le nombre de machines, N le nombre de jobs et $\overline{p}$ la moyenne des temps d'exécution préalablement générés pour l'instance considérée. Ainsi, une date due sera comprise entre la date de fin moyenne du premier job ordonnancé et la date de fin moyenne du dernier job ordonnancé. Des jeux de données sont proposés pour des problèmes de $N = 20$, 30, 50, 70, 100, 150, 200, 300 et 500 jobs et $M = 5$, 10, 15 et 20 machines, avec 10 instances différentes par taille $(N \times M)$. Une instance nommée N_M_i représente la $i$ème instance composée de N jobs et M machines.

5.1.2. *Jeux de données stochastiques*

Afin de générer de l'incertitude sur une instance déterministe initiale, les quatre distributions de probabilité proposées à la section 3.3 peuvent être appliquées sur les données initiales à l'aide d'un fichier de configuration. Nous avons choisi de permettre la configuration de cette incertitude sur les machines uniquement, en spécifiant, pour chacune d'elles, une distribution de probabilité et ses paramètres ou des proportions liées à sa tendance centrale. Ainsi, à chaque fois que de l'incertitude sera générée sur une instance déterministe, les durées d'exécution obtenues seront différentes ; ceci afin de refléter au mieux la réalité du monde industriel.

2. Les jeux de données pour le cas déterministe et le cas stochastique sont disponibles sur le web à

5.2. *Tests réalisés*

Afin de réaliser nos expérimentations, nous avons généré de l'incertitude sur les temps d'exécution des jeux de données bi-objectif déterministes. Pour cela, nous avons utilisé les modèles stochastiques définis précédemment. Ainsi, pour une instance donnée, nous avons créé 10 évaluations indépendantes dont les temps d'exécution p_{ij} suivent des distributions uniforme, normale, exponentielle ou log-normale de la façon suivante :

– loi uniforme : $p_{ij} \sim \mathcal{U}(a = 0.85 \times p_{ij}, b = 1.15 \times p_{ij})$;

– loi normale : $p_{ij} \sim \mathcal{N}(\mu = p_{ij}, \sigma = 0.15 \times p_{ij})$;

– loi exponentielle : $p_{ij} \sim \mathcal{E}(a = p_{ij}, \lambda = \frac{1}{0.15 \times p_{ij}})$;

– loi log-normale : $p_{ij} \sim log\text{-}\mathcal{N}(\mu = \log p_{ij}, \sigma = 0.15 \times \log p_{ij})$;

– lois variables : la loi associée aux durées d'exécution diffère sur chacune des machines.

La taille de la population N a été fixée à 50 individus. Pour chaque type de distribution de probabilité, 10 exécutions par instance et par algorithme ont été réalisées, avec 10 évaluations différentes par solution (sauf pour $IBEA^1$ où seule la première évaluation est utilisée). Les différentes méthodes testées utilisent la même population initiale et le même nombre de génération (5 000 générations pour les instances à 20 jobs et 5 machines, et 10 000 générations pour les instances à 20 jobs et 10 machines). Les probabilités d'application de l'opérateur de croisement et de l'opérateur de mutation ont été fixées à 0.05 et 1 respectivement. Le facteur de redimensionnement κ, nécessaire dans $IBEA^1$ et $IBEA^{avg}$, est fixé à $0,05$.

5.3. *Evaluation de performance*

À notre connaissance, il n'existe pas de protocole pleinement adapté à l'évaluation des performances de méthodes de recherche pour les problèmes d'optimisation multi-objectif stochastiques. Par conséquent, nous choisissons de réévaluer l'ensemble final de solutions de chacun des algorithmes sur le jeu de données de référence (celui qui a servi à l'élaboration des jeux de données stochastiques), et de considérer cette évaluation comme étant l'évaluation réelle des solutions. Ensuite, nous ne conservons que les points non dominés sur l'ensemble de vecteurs objectif obtenus par chaque méthode sur cette évaluation. Ainsi, il nous est possible d'utiliser des métriques classiques, employées dans le cas déterministe, pour mesurer la qualité des approximations des fronts Pareto obtenus. Dans le cas multi-objectif, la comparaison de résultats est plus délicate que dans le cas mono-objectif, le résultat étant constitué d'un ensemble de points non dominés, donc non comparables entre eux. Il s'avère donc tout aussi important de mesurer la qualité de ces points en termes de convergence qu'en termes de diversité. Pour cela, nous utiliserons la métrique *Contribution*, dédiée à l'évaluation de la convergence, ainsi que la métrique *Hypervolume*, dédiée à la fois à l'évaluation de la convergence et de la diversité.

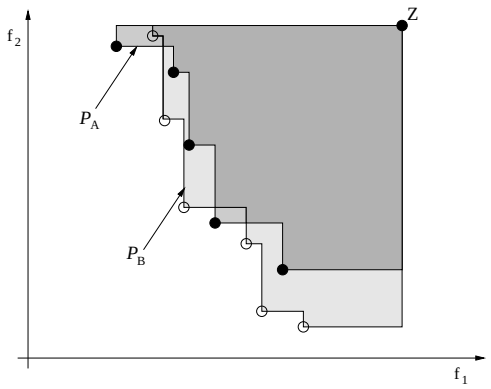

Figure 9. *Illustration de la métrique Hypervolume : calcul des aires de dominance d'ensembles de solutions Pareto par rapport à un point de référence Z*

La métrique *Contribution* (Meunier *et al.*, 2000) permet d'avoir une idée de l'apport d'un algorithme par rapport à un autre. Elle se base sur la comparaison de deux fronts Pareto approchés $\mathcal{P}_A$ et $\mathcal{P}_B$ (respectivement obtenus par deux algorithmes A et B). Soit $\mathcal{P}_{AB}$ l'ensemble des points non dominés de $\mathcal{P}_A \cup \mathcal{P}_B$. La valeur $Contribution(A, B)$ (respectivement $Contribution(B, A)$) mesure la proportion de l'ensemble $\mathcal{P}_{AB}$ représentée par les points de $\mathcal{P}_A$ (respectivement $\mathcal{P}_B$). Ainsi, si les deux algorithmes A et B produisent les mêmes approximations Pareto, $Contribution(A, B) = Contribution(B, A) = 1/2$. Si tous les vecteurs de A sont dominés par des vecteurs de B, $Contribution(A, B) = 0$. Et, de manière générale, $Contribution(A, B) + Contribution(B, A) = 1$. La métrique *Hypervolume* (Zitzler *et al.*, 1999) calcule l'hypervolume de la région multidimensionnelle fermée par un front Pareto approché $\mathcal{P}_A$ et un point de référence Z (voir figure 9). Cet hypervolume représente la taille de l'espace objectif dominé par $\mathcal{P}_A$. Le point de référence que nous utiliserons pour comparer les résultats observés sur une instance donnée sera constitué des valeurs maximales obtenues lors de nos expérimentations pour chacun des critères sur cette instance, multipliées par 1.1 (ce qui permet de tenir compte des points extrêmes sans leur donner trop d'importance).

Par ailleurs, afin de comparer de façon significative les algorithmes proposés, nous avons choisi de réaliser un test d'hypothèse de *Wilcoxon* (Wilcoxon, 1945) pour chaque instance et chaque type de distribution de probabilité. Dans chacune des tables de résultats de la section suivante, la colonne 'T' indique le résultat du test statistique pour une p-valeur inférieure à 5%, *i.e.* :

– selon la métrique considérée, les résultats de l'algorithme situé à la ligne spécifiée sont significativement meilleurs que ceux de l'algorithme situé à la colonne spécifiée $(+)$;

– selon la métrique considérée, les résultats de l'algorithme situé à la ligne spécifiée sont significativement moins bons que ceux de l'algorithme situé à la colonne spécifiée ($-$) ;

– selon la métrique considérée, il n'y a pas de différence significative entre les résultats de l'algorithme situé à la ligne spécifiée et ceux de l'algorithme situé à la colonne spécifiée ($\equiv$).

5.4. *Résultats et discussion*

Les résultats obtenus par les algorithmes proposés pour des durées d'exécution suivant une loi uniforme, une loi exponentielle, une loi normale, une loi log-normale ou des lois variables sont respectivement présentés dans les tableaux 1, 2, 3, 4 et 5. Le nombre moyen de points non dominés obtenus par les algorithmes suite à la rééavluation des solutions sur l'instance de référence est donné dans le tableau 6.

De manière générale, selon le protocole d'évaluation des performances que nous avons utilisé, $IBEA^{avg}$ est majoritairement plus performant et n'est jamais moins performant qu'$IBEA^{1}$ et $IBEA^{\widehat{eiv}}$ sur l'ensemble des instances testées. Seul $IBEA^{1}$ se montre significativement plus efficace, d'après la métrique Hypervolume, sur la première instance testée dont les durées de traitement suivent une loi log-normale (voir le tableau 4). Étant donné qu'$IBEA^{avg}$ utilise des valeurs objectif moyennes, la supériorité de cette méthode peut facilement se justifier pour des probabilités dont la tendance centrale est la moyenne, comme c'est par exemple le cas pour une distribution uniforme ou normale. Néanmoins, n'oublions pas que l'incertitude est située ici sur les paramètres et non sur les fonctions objectif. Ce qui peut signifier que, pour l'ensemble des lois de probabilité considérées sur les durées d'exécution, l'incertitude engendrée sur la valeur des objectifs pourrait se réduire à une loi de probabilité dont la tendance centrale est la moyenne.

Comme indiqué sur les tableaux 1 et 3, pour des temps de traitement normalement et uniformément distribués, $IBEA^{1}$ s'avère plus efficace qu'$IBEA^{\widehat{eiv}}$ selon la métrique Hypervolume. Cependant, selon la métrique Contribution, il n'existe pas de différence significative entre ces deux algorithmes, à part pour une distribution uniforme où $IBEA^{\widehat{eiv}}$ surpasse $IBEA^{1}$ sur la dernière instance (voir le tableau 1). Pour des durées exponentiellement distribuées, il n'y a pas de différence significative entre ces deux méthodes (voir le tableau 2). Les seules exceptions sont la seconde instance où $IBEA^{1}$ est significativement plus performant selon la métrique Hypervolume, et la dernière instance où $IBEA^{\widehat{eiv}}$ est significativement plus performant selon la métrique Contribution. Enfin, comme présenté sur les tableaux 4 et 5, la métrique Contribution indique qu'il n'existe pas de différence significative entre $IBEA^{\widehat{eiv}}$ et $IBEA^{1}$ pour des temps de traitement distribués selon une loi log-normale ou de façon variable. Au contraire, selon la métrique Hypervolume, $IBEA^{\widehat{eiv}}$ se montre généralement plus efficace qu'$IBEA^{1}$ pour la loi log-normale (excepté sur la première instance), alors

qu'il est plus performant uniquement sur la dernière instance pour des durées distribuées de façon variable.

La faible efficacité d'$IBEA^{\widehat{eiv}}$ peut en partie s'expliquer par un manque de diversité des solutions obtenues dans l'espace de décision. Ainsi, lors de la réévaluation de cet ensemble de solutions sur l'instance de référence, le front obtenu s'avère généralement peu diversifié en raison de la spécificité de notre problème (voir le tableau 6), et ne peut donc pas rivaliser avec ceux des deux autres méthodes proposées en termes de diversité. Dans (Basseur *et al.*, 2006), ce problème du nombre de points non dominés n'apparaît pas, le protocole d'évaluation des performances étant différent. Ainsi, il pourrait se montrer intéressant de mettre en place un mécanisme de diversification au sein d'$IBEA^{\widehat{eiv}}$ afin d'augmenter la qualité des ensembles produits. Néanmoins, les résultas expérimentaux sont tous discutables, le protocole d'évaluation des performances n'étant qu'imparfaitement adapté aux problèmes d'optimisation multi-objectif stochastiques.

		Contribution				Hypervolume			
		$IBEA^1$		$IBEA^{avg}$		$IBEA^1$		$IBEA^{avg}$	
		p-valeur	T	p-valeur	T	p-valeur	T	p-valeur	T
20_05_01	$IBEA^{avg}$	$> 5\,\%$	$\equiv$			0.002	$+$		
	$IBEA^{\widehat{eiv}}$	$> 5\,\%$	$\equiv$	0.009	$-$	0.002	$-$	0.001	$-$
20_05_02	$IBEA^{avg}$	0.013	$+$			0.002	$+$		
	$IBEA^{\widehat{eiv}}$	$> 5\,\%$	$\equiv$	0.006	$-$	0.005	$-$	0.001	$-$
20_10_01	$IBEA^{avg}$	0.002	$+$			0.001	$+$		
	$IBEA^{\widehat{eiv}}$	0.005	$+$	0.002	$-$	0.001	$-$	0.001	$-$

Tableau 1. *Comparaison des valeurs de métriques Contribution et Hypervolume obtenues par $IBEA^1$, $IBEA^{avg}$ et $IBEA^{\widehat{eiv}}$ pour des durées d'exécution suivant une* ***loi uniforme*** *à l'aide d'un test statistique de Wilcoxon*

		Contribution				Hypervolume			
		$IBEA^1$		$IBEA^{avg}$		$IBEA^1$		$IBEA^{avg}$	
		p-valeur	T	p-valeur	T	p-valeur	T	p-valeur	T
20_05_01	$IBEA^{avg}$	$> 5\,\%$	$\equiv$			0.001	$+$		
	$IBEA^{\widehat{eiv}}$	$> 5\,\%$	$\equiv$	$> 5\,\%$	$\equiv$	$> 5\,\%$	$\equiv$	0.033	$-$
20_05_02	$IBEA^{avg}$	$> 5\,\%$	$\equiv$			$> 5\,\%$	$\equiv$		
	$IBEA^{\widehat{eiv}}$	$> 5\,\%$	$\equiv$	$> 5\,\%$	$\equiv$	0.002	$-$	0.001	$-$
20_10_01	$IBEA^{avg}$	0.045	$+$			0.001	$+$		
	$IBEA^{\widehat{eiv}}$	0.036	$+$	$> 5\,\%$	$\equiv$	$> 5\,\%$	$\equiv$	0.005	$-$

Tableau 2. *Comparaison des valeurs de métriques Contribution et Hypervolume obtenues par $IBEA^1$, $IBEA^{avg}$ et $IBEA^{\widehat{eiv}}$ pour des durées d'exécution suivant une* ***loi exponentielle*** *à l'aide d'un test statistique de Wilcoxon*

		Contribution				Hypervolume			
		$IBEA^1$		$IBEA^{avg}$		$IBEA^1$		$IBEA^{avg}$	
		p-valeur	T	p-valeur	T	p-valeur	T	p-valeur	T
20_05_01	$IBEA^{avg}$	$> 5\,\%$	$\equiv$			0.002	$+$		
	$IBEA^{\widehat{eiv}}$	$> 5\,\%$	$\equiv$	$> 5\,\%$	$\equiv$	0.021	$-$	0.001	$-$
20_05_02	$IBEA^{avg}$	$> 5\,\%$	$\equiv$			0.001	$+$		
	$IBEA^{\widehat{eiv}}$	$> 5\,\%$	$\equiv$	$> 5\,\%$	$\equiv$	0.002	$-$	0.001	$-$
20_10_01	$IBEA^{avg}$	0.004	$+$			0.002	$+$		
	$IBEA^{\widehat{eiv}}$	$> 5\,\%$	$\equiv$	0.004	$+$	0.001	$-$	0.001	$-$

Tableau 3. *Comparaison des valeurs de métriques Contribution et Hypervolume obtenues par $IBEA^1$, $IBEA^{avg}$ et $IBEA^{\widehat{eiv}}$ pour des durées d'exécution suivant une* ***loi normale*** *à l'aide d'un test statistique de Wilcoxon*

		Contribution				Hypervolume			
		$IBEA^1$		$IBEA^{avg}$		$IBEA^1$		$IBEA^{avg}$	
		p-valeur	T	p-valeur	T	p-valeur	T	p-valeur	T
20_05_01	$IBEA^{avg}$	$> 5\,\%$	$\equiv$			0.032	$-$		
	$IBEA^{\widehat{eiv}}$	$> 5\,\%$	$\equiv$	$> 5\,\%$	$\equiv$	0.001	$-$	0.001	$-$
20_05_02	$IBEA^{avg}$	$> 5\,\%$	$\equiv$			0.001	$+$		
	$IBEA^{\widehat{eiv}}$	$> 5\,\%$	$\equiv$	$> 5\,\%$	$\equiv$	0.001	$+$	0.001	$-$
20_10_01	$IBEA^{avg}$	$> 5\,\%$	$\equiv$			0.001	$+$		
	$IBEA^{\widehat{eiv}}$	$> 5\,\%$	$\equiv$	$> 5\,\%$	$\equiv$	0.001	$+$	0.020	$-$

Tableau 4. *Comparaison des valeurs de métriques Contribution et Hypervolume obtenues par $IBEA^1$, $IBEA^{avg}$ et $IBEA^{\widehat{eiv}}$ pour des durées d'exécution suivant une* ***loi log-normale*** *à l'aide d'un test statistique de Wilcoxon*

		Contribution				Hypervolume			
		$IBEA^1$		$IBEA^{avg}$		$IBEA^1$		$IBEA^{avg}$	
		p-valeur	T	p-valeur	T	p-valeur	T	p-valeur	T
20_05_01	$IBEA^{avg}$	$> 5\,\%$	$\equiv$			0.001	$+$		
	$IBEA^{\widehat{eiv}}$	$> 5\,\%$	$\equiv$	$> 5\,\%$	$\equiv$	0.019	$-$	0.001	$-$
20_05_02	$IBEA^{avg}$	$> 5\,\%$	$\equiv$			0.014	$+$		
	$IBEA^{\widehat{eiv}}$	$> 5\,\%$	$\equiv$	$> 5\,\%$	$\equiv$	$> 5\,\%$	$\equiv$	0.010	$-$
20_10_01	$IBEA^{avg}$	$> 5\,\%$	$\equiv$			0.001	$+$		
	$IBEA^{\widehat{eiv}}$	$> 5\,\%$	$\equiv$	$> 5\,\%$	$\equiv$	0.003	$+$	$> 5\,\%$	$\equiv$

Tableau 5. *Comparaison des valeurs de métriques Contribution et Hypervolume obtenues par $IBEA^1$, $IBEA^{avg}$ et $IBEA^{\widehat{eiv}}$ pour des durées d'exécution distribués de façon* ***variable*** *à l'aide d'un test statistique de Wilcoxon*

		$IBEA^1$	$IBEA^{avg}$	$IBEA^{\widehat{eiv}}$
20_05_01	loi uniforme	3.7	3.1	1.0
	loi exponentielle	1.1	2.6	1.0
	loi normale	5.8	3.6	1.0
	loi log-normale	1.9	1.8	1.6
	lois variables	2.0	2.5	1.0
20_05_02	loi uniforme	3.7	3.5	1.0
	loi exponentielle	2.8	2.2	1.0
	loi normale	2.4	2.8	1.1
	loi log-normale	1.0	3.9	1.3
	lois variables	2.7	2.9	2.0
20_10_02	loi uniforme	10.4	12.4	2.6
	loi exponentielle	4.5	7.2	3.4
	loi normale	6.8	7.0	2.6
	loi log-normale	3.5	2.0	1.8
	lois variables	6.2	4.8	2.2

Tableau 6. *Nombre moyen de points non dominés obtenus par $IBEA^1$, $IBEA^{avg}$ et $IBEA^{\widehat{eiv}}$ suite à la rééavluation des solutions sur l'instance de référence*

6. Conclusion et perspectives

Dans cet article, plusieurs métaheuristiques dédiées à la résolution de problèmes d'optimisation multi-objectif stochastiques ont été présentées. Dans un premier temps, nous avons vu que les problèmes concrets d'ordonnancement se devaient couramment de traiter plusieurs critères simultanément et étaient généralement soumis à un grand nombre d'incertitudes. Des modèles non déterministes ont été proposés pour un problème d'ordonnancement bi-objectif de type flow-shop de permutation avec durées d'exécution aléatoires. Par ailleurs, des jeux de données ont été construits pour le problème considéré ; dans un premier temps pour le cas multi-objectif, ensuite étendus au cas stochastique. D'un point de vue algorithmique, la contribution du travail réside en la proposition d'indicateurs de qualité spécifiquement élaborés pour la prise en compte de tout type d'incertitude, et pouvant être utilisés au sein de l'algorithme évolutionnaire basé sur un indicateur de qualité IBEA (Indicator-Based Evolutionary Algorithm) proposé par Zitzler et Künzli (Zitzler *et al.*, 2004). Trois indicateurs stochastiques (I^1, I^{avg} et $I^{\widehat{eiv}}$) ont été introduits, donnant lieu à trois algorithmes évolutionnaires (respectivement $IBEA^1$, $IBEA^{avg}$ et $IBEA^{\widehat{eiv}}$). Notez que tous ces algorithmes seront bientôt intégrés au sein de la plateforme ParadisEO-MOEO. Ces indicateurs utilisent tous l'indicateur ϵ-additif pour comparer des vecteurs objectif entre eux. I^1 préserve l'approche utilisée dans le cas déterministe. Une solution est évaluée une seule fois et sa qualité est estimée en utilisant cette simple évaluation. I^{avg} consiste à utiliser un vecteur objectif moyen pour estimer la qualité d'une solution. Enfin, $I^{\widehat{eiv}}$ estime la qualité d'une solution de manière probabiliste telle que proposé dans (Basseur *et al.*, 2006). Nous avons ensuite expérimenté nos algorithmes sur le flow-shop de permutation. Nous avons vu qu'$IBEA^{avg}$ était globalement plus

efficace que les deux autres méthodes de résolution sur les instances que nous avons testées. Les résultats obtenus par $IBEA^{\widehat{eiv}}$ sont un peu surprenants. D'un point de vue théorique, cet algorithme semble en effet celui qui estime la qualité d'une solution de la façon la plus représentative, ce qui n'apparaît pas du tout expérimentalement. Ceci semble être expliqué par un manque de diversité des solutions de l'ensemble final obtenu par $IBEA^{\widehat{eiv}}$. Néanmoins, nous avons vu que le protocole utilisé pour l'évaluation des performances, bien que d'une grande simplicité, était insuffisant pour analyser équitablement les résultats obtenus dans le cas stochastique.

Différentes perspectives apparaissent suite à ce travail. Tout d'abord, d'autres sources d'incertitude que les durées d'exécution, telles que les dates dues, existent pour le problème d'ordonnancement étudié ici ; il serait donc intéressant de les considérer également. De plus, des approches différentes de l'approche stochastique peuvent permettre de représenter l'incertitude sur les durées d'exécution. Ainsi, des approches floues ont déjà été utilisées pour modéliser ce type d'incertitude pour des problèmes d'ordonnancement mono-objectif. Ensuite, beaucoup de problèmes d'ordonnancement à plus de deux objectifs existent et pourraient être expérimentés. Étudier le paysage des jeux de données que nous avons proposés pourrait également se montrer utile ; ceci afin d'expliquer plus précisément le comportement des algorithmes et d'en tirer partie lors du processus d'optimisation. Par ailleurs, d'autres indicateurs de qualité à utiliser au sein d'IBEA peuvent être imaginés pour résoudre des problèmes d'optimisation multi-objectif stochastiques. Les comparer aux indicateurs proposés dans cet article serait donc intéressant. De même, les concepts proposés ici peuvent facilement être applicables à d'autres types de métaheuristiques, telle que la recherche locale basée sur indicateur de qualité proposée dans (Basseur *et al.*, 2007). Ceci pourrait, par exemple, donner lieu à une hybridation entre plusieurs méthodes de résolution, et pourrait considérablement améliorer les résultats. Cependant, avant toute autre chose, il semble indispensable de réfléchir à un protocole d'évaluation pleinement adapté au cas multi-objectif stochastique, ceci afin de comparer les méthodes de résolution de manière plus impartiale.

7. Bibliographie

Babbar M., Lakshmikantha A., Goldberg D. E., « A Modified NSGA-II to Solve Noisy Multiobjective Problems », *Genetic and Evolutionary Computation Conference (GECCO 2003)*, vol. 2723 of *Lecture Notes in Computer Science*, Springer-Verlag, Chicago, USA, p. 21-27, 2003.

Basseur M., Burke E. K., « Indicator-Based Multi-Objective Local Search », *IEEE Congress on Evolutionary Computation (CEC 2007)*, Singapore, p. 3100-3107, 2007.

Basseur M., Zitzler E., « Handling Uncertainty in Indicator-Based Multiobjective Optimization », *International Journal of Computational Intelligence Research*, vol. 2, n° 3, p. 255-272, 2006.

Billaut J.-C., Moukrim A., Sanlaville E. (eds), *Flexibilité et robustesse en ordonnancement*, Hermès Science, Paris, France, 2005.

Coello Coello C. A., Van Veldhuizen D. A., Lamont G. B., *Evolutionary Algorithms for Solving Multi-Objective Optimization Problems*, Kluwer Academic Publishers, 2002.

Cunningham A. A., Dutta S. K., « Scheduling jobs with exponentially distributed processing times on two machines of a flow shop », *Naval Research Logistics Quarterly*, vol. 16, p. 69-81, 1973.

Dauzère-Pérès S., Castagliola P., Lahlou C., « Niveau de service en ordonnancement stochastique », *in* J.-C. Billaut, A. Moukrim, E. Sanlaville (eds), *Flexibilité et robustesse en ordonnancement*, Hermès Science, Paris, France, chapter 5, p. 97-113, 2005.

Deb K., *Multi-Objective Optimization using Evolutionary Algorithms*, Wiley, 2001.

Deb K., Gupta H., « Searching for Robust Pareto-Optimal Solutions in Multi-Objective Optimization », *in* C. A. C. Coello, A. H. Aguirre, E. Zitzler (eds), *Evolutionary Multi-Criterion Optimization, third International Conference (EMO 2005)*, vol. 3410 of *Lecture Notes in Computer Science*, Springer-Verlag, Guanajuato, Mexico, p. 150-164, 2005.

Fortemps P., « Jobshop Scheduling with Imprecise Durations : A Fuzzy Approach », *IEEE Transactions On Fuzzy System*, vol. 5, n° 4, p. 557-569, 1997.

Goh C. K., Tan K. C., « An Investigation on Noisy Environments in Evolutionary Multiobjective Optimization », *IEEE Transactions on Evolutionary Computation*, vol. 11, n° 3, p. 354-381, 2007.

Gourgand M., Grangeon N., Norre S., « Modèle du flow-shop de permutation stochastique », *in* J.-C. Billaut, A. Moukrim, E. Sanlaville (eds), *Flexibilité et robustesse en ordonnancement*, Hermès Science, Paris, France, chapter 7, p. 135-161, 2005.

Graham R. L., Lawler E. L., Lenstra J. K., Rinnooy Kan A. H. G., « Optimization and Approximation in Deterministic Sequencing and Scheduling : A Survey », *Annals of Discrete Mathematics*, vol. 5, p. 287-326, 1979.

Hansen M. P., Jaszkiewicz A., Evaluating the quality of approximations of the non-dominated set, Technical Report n° IMM-REP-1998-7, Institute of Mathematical Modeling, Technical University of Denmark, 1998.

Hughes E. J., « Evolutionary Multi-Objective Ranking with Uncertainty and Noise », *in* E. Zitzler, K. Deb, L. Thiele, C. A. Coello Coello, D. Corne (eds), *Evolutionary Multi-Criterion Optimization, First International Conference (EMO 2001)*, vol. 1993 of *Lecture Notes in Computer Science*, Springer-Verlag, London, UK, p. 329-343, 2001.

Ishibuchi H., Murata T., « A Multi-Objective Genetic Local Search Algorithm and Its Application to Flowshop Scheduling », *IEEE Transactions on Systems, Man and Cybernetics*, vol. 28, p. 392-403, 1998.

Jin Y., Branke J., « Evolutionary Optimization in Uncertain Environments - A Survey », *IEEE Transactions on Evolutionary Computation*, vol. 9, n° 3, p. 303-317, 2005.

Kouvelis P., Daniels R. L., Vairaktarakis G., « Robust scheduling of a two-machine flow shop with uncertain processing times », *IIE Transactions*, vol. 32, n° 5, p. 421-432, 2000.

Ku P. S., Niu S. C., « On Johnson's Two-Machine Flow Shop with Random Processing Times », *Operations Research*, vol. 34, p. 130-136, 1986.

Landa Silva J. D., Burke E., Petrovic S., « An Introduction to Multiobjective Metaheuristics for Scheduling and Timetabling », *in* X. Gandibleux, M. Sevaux, K. Sörensen, V. T'kindt (eds), *Metaheuristics for Multiobjective Optimisation*, vol. 535 of *Lecture Notes in Economics and Mathematical Systems*, Springer-Verlag, p. 91-129, 2004.

Liefooghe A., « *Optimisation multi-objectif sous incertitude : Application aux problèmes d'ordonnancement de type flow-shop* », Master's thesis, Université des Sciences et Technologies de Lille, France, 2006.

Liefooghe A., Basseur M., Jourdan L., Talbi E.-G., « ParadisEO-MOEO : A framework for Evolutionary Multi-objective Optimization », *in* S. Obayashi, C. Poloni, K. Deb (eds), *Evolutionary Multi-Criterion Optimization, Fourth International Conference (EMO 2007)*, vol. 4403 of *Lecture Notes in Computer Science*, Springer-Verlag, Matsushima, Japan, p. 386-400, 2007.

Maddox M. J., Birge J. R., « Using second moment information in stochastic scheduling », *Recent advances in control and optimization of manufacturing systems*, vol. 214 of *Lecture Notes in Control and Information Sciences*, Springer, London, UK, p. 99-120, 1996.

Makino T., « On a scheduling problem », *Journal of the Operations Research Society of Japan*, vol. 8, p. 32-44, 1965.

Meunier H., Talbi E.-G., Reininger P., « A Multiobjective Genetic Algorithm for Radio Network Optimization », *IEEE Congress on Evolutionary Computation (CEC 2000)*, San Diego, USA, p. 317-324, 2000.

Nagar A., Haddock J., Heragu S., « Multiple and bicriteria scheduling : A literature survey », *European Journal of Operational Research*, vol. 81, p. 88-104, 1995.

Roy B., « À propos de robustesse en recherche opérationnelle et aide à la décision », *in* J.-C. Billaut, A. Moukrim, E. Sanlaville (eds), *Flexibilité et robustesse en ordonnancement*, Hermès Science, Paris, France, chapter 2, p. 35-50, 2005.

Sanlaville E., Ordonnancement sous conditions changeantes : Comment prendre en compte les variations, aléas, incertitudes sur les données ?, Habilitation à diriger des recherches, Université Blaise Pascal Clermont Ferrand, 2005.

Taillard E. D., « Benchmarks for Basic Scheduling Problems », *European Journal of Operational Research*, vol. 64, p. 278-285, 1993.

Talwar P. P., « A note on sequencing problems with uncertain job times », *Journal of the Operations Research Society of Japan*, vol. 9, p. 93-97, 1967.

Teich J., « Pareto-Front Exploration with Uncertain Objectives », *in* E. Zitzler, K. Deb, L. Thiele, C. A. Coello Coello, D. Corne (eds), *Evolutionary Multi-Criterion Optimization, First International Conference (EMO 2001)*, vol. 1993 of *Lecture Notes in Computer Science*, Springer-Verlag, London, UK, p. 314-328, 2001.

T'kindt V., Billaut J.-C., *Multicriteria Scheduling - Theory, Models and Algorithms*, Springer, 2002.

Wang L., Zhang L., Zheng D.-Z., « A class of hypothesis-test based genetic algorithm for flow shop scheduling with stochastic processing time », *The International Journal of Advanced Manufacturing Technology*, vol. 25, n° 11-12, p. 1157-1163, 2005.

Wilcoxon F., « Individual comparisons by ranking methods », *Biometrics*, vol. 1, p. 80-83, 1945.

Zitzler E., Künzli S., « Indicator-Based Selection in Multiobjective Search », *Proceedings of the 8th International Conference on Parallel Problem Solving from Nature (PPSN VIII)*, vol. 3242 of *Lecture Notes in Computer Science*, Springer-Verlag, Birmingham, UK, p. 832-842, 2004.

Zitzler E., Thiele L., « Multiobjective Evolutionary Algorithms : A Comparative Case Study and the Strength Pareto Approach », *IEEE Transactions on Evolutionary Computation*, vol. 3, n° 4, p. 257-271, 1999.

Zitzler E., Thiele L., Laumanns M., Foneseca C. M., Grunert da Fonseca V., « Performance Assessment of Multiobjective Optimizers : An Analysis and Review », *IEEE Transactions on Evolutionary Computation*, vol. 7, n° 2, p. 117-132, 2003.