A Bayesian approach to find Pareto optima in multiobjective programming problems using Sequential Monte Carlo algorithms

Mike G. Tsionas*

12th May 2017

Abstract

In this paper we consider a new approach to multicriteria decision making problems. Such problems are, usually, cast into a Pareto framework where the objective functions are aggregated into a single one using certain weights. The problem is embedded into a statistical framework by adopting a posterior distribution for both the decision variables and the Pareto weights. This embedding dates back to Pinkus (1968) but in this work we operationalize the concept further. We propose a Metropolis-Hastings and a Sequential Monte Carlo (SMC) to trace out the entire Pareto frontier and / or find the global optimum of the problem. We apply the new techniques to a multicriteria portfolio decision making problem proposed in Xidonas et al. (2010) and to a test problem proposed by Qu et al. (2013). The good performance of new techniques suggests that SMC and other algorithms, like the classical Metropolis-Hastings algorithm, can be used profitably in the context of multicriteria decision making problems to trace out the Pareto frontier and / or find a global optimum. Most importantly SMC can be considered as an off-the-shelf technique to solve arbitrary multicriteria decision making problems routinely and efficiently.

Key Words: Economics; Multicriteria Decision Making; Sequential Monte Carlo; Global Optimization; Portfolio Analysis.

Acknowledgements: The author is indebted to three anonymous reviewers and the Editor Ben Lev for useful comments on an earlier version.

*Lancaster University Management School, LA1 4YX, U.K & Athens University of Economics and Business, Greece.

1 Introduction

Multiobjective programming is an attractive method when we have multiple conflicting objectives (see, Das & Dennis, 1998; Handi, Kell, & Knowles, 2007). In fact, portfolio analysis, for example, a well known tool in operations research can be cast in terms of multiobjective programming instead of the classical Markowitz formulation of the problem. The nature of the problem has been emphasized as multicriteria diecision making by many researchers in the field (Mavrotas et al. 2008; Xidonas and Psarras 2009; Xidonas et al. 2009a, b, c, d; Steuer et al. 2005, 2006a, b, 2007a, b; Zopounidis and Doumpos 2002; Zopounidis 1999; Hurson and Zopounidis 1993, 1995, 1997; Spronk and Hallerbach 1997; Zeleny 1977, 1981, 1982; Colson and Zeleny 1979, 1980).

One may proceed in two ways. One is the so called scalarization approach in which a single objective is fomrulated and corresponding Pareto optima are found (Das & Dennis, 1998). The alternative is to solve the first order conditions for Pareto optimality in multiobjective programming (see Fliege, Grana drummond, & Svaiter, 2009; Vieira, Takahashi, & Saldnha, 2012). Qu et al. (2011) present a quasi-Newton method for smooth problems. Qu, Goh, and Liang (2013) and Neto, Silva, Ferreira, and Lopes (2013) consider methods for non-smooth objectives. Qu et al. (2014) propose an extension of the quasi-Newton method in Qu et al. (2011) to the nonsmooth case and present a new algorithm to compute the critical point of nonsmooth multiobjective programming under non-convexity. As they notice: "At each iteration, the descent direction is obtained by solving a linear programming subproblem with convex quadratic constraints. This subproblem improves the performance by constraining the descent direction norms with an small positive scalar which can control the descent direction approaching zero". Other approaches are explored in Angilella et al. (2016), Chica et al. (2016), Cardoso et al. (2016), Kadziński et al. (2017), Mavrotas et al. (2015), Paul et al. (2017), Teresinha Arns Steiner et al. (2016) and Tsai and Chen (2017).

In this paper we propose a new approach to multiobjective programming by considering an equivalent posterior disrtibution for the decision variables *and* the Pareto weights, which are often unknown. Although scalarization involves some complications even when the individual objectives are simple (Donoso & Fabregat, 2007) algorithms such as the Metropolis-Hastings and Sequential Monte Carlo (SMC) can explore the posterior and yield the global optimum using simulation techniques. Related to our work is the paper by Zhou and Chen (2013) who proposed SMC in the context of global optimization of scalar objective functions. Zhou and Chen (2013) showed that their SMC is more preferable than the multi-start simulated annealing (SA) method when the sample size is sufficiently large. As they write: "We carried out numerical experiments on several benchmark problems. The numerical results show that SMC-SA is a great improvement of the standard SA on all the test problems; SMC-SA outperforms multi-start SA and CE on badly-scaled problems and problems with a small number of local optima; the CE method works better on well-scaled problems with a large number of local optima. We also compared the performance of SMC-SA and multi-start SA as the sample size varies, and the results verified our analytical results." Our SMC is different compared to Zhou and Chen (2013) and the method can be applied to multicriteria decision making unlike Zhou and Chen (2013).

We present the model in Section 2. In Section 3 we provide details for our numerical techniques based on the Metropolis-Hastings and SMC. In Section 4 we present an empirical application to portfolio analysis, previously analyzed by Xidonas et al. (2010).

2 The model

Suppose we have the multiobjective programming problem where we have multiple conflicting objectives. Following Qu et al. (2014) suppose we have a set of objective functions $F(x) = \underline{F}(x) + \underline{U}$, where $x \in X \subseteq \mathbb{R}^k$, $\underline{U}$ represents noise, and:

$$\begin{aligned} F_1(x) &= \underline{F}_1(x) + U_1, \\ F_2(x) &= \underline{F}_2(x) + U_2, \\ &\cdots \\ F_n(x) &= \underline{F}_n(x) + U_n, \end{aligned} \tag{1}$$

where $F = (F_1, ..., F_n)' \in \mathbb{R}^n$, $\underline{F} = (\underline{F}_1, ..., \underline{F}_n)' \in \mathbb{R}^n$, and $U = (\underline{U}_1, ..., \underline{U}_n)' \in \mathbb{R}^n$. The objective is to solve the problem:

$$\min_{x \in X \subseteq \mathbb{R}^k} F(x). \tag{2}$$

As in Qu et al. (2014) we settle for global Pareto optimality, meaning that x^* is a solution if and only if there does not exist $x \in X$ and $F(x) \leq F(x^*)$, $F(x) \neq F(x^*)$.

In multicriteria decision making the problem is:

$$\min_{x \in X} \sum_{i=1}^{n} \alpha_i F_i(x), \tag{3}$$

for a certain vector of **Pareto weights** $\alpha = (\alpha_1, ..., \alpha_n)'$ which belong to the unit simplex, $S = \{\alpha \in \mathbb{R}^n : \alpha_i \geq 0,\ i = 1, ..., n,\ \sum_{i=1}^{n} \alpha_i = 1\}$. In this problem, we can get a solution for any given set of αs although it is much better if we let the "data" F speak for themselves about these weights. Indeed, suppose we have a prior, $p(\alpha)$, about the weights, possibly uniform over S and, possibly, also a prior $p(x)$ defined over X. Moreover, if we solve (3) for a range of values of $\alpha \in S$ we can trace out the Pareto frontier.

As $\underline{F}(x)$ is unobserved, we cannot take advantage of (1) by using, for example joint normality of Us to formulate a statistical problem. Therefore, we proceed as follows. Problem (3) is equivalent to finding the mean of the following posterior distribution:

$$p(x, \alpha | h, F) = \frac{\exp\{-h \sum \alpha_i F_i(x)\} p(x) p(\alpha)}{\int_{X \times S} \exp\{-h \sum \alpha_i F_i(\chi)\} p(\chi) p(\alpha) d\chi d\alpha}, \tag{4}$$

for a given positive constant h. We re-emphasize that we can condition on the αs. Letting $\theta = (x', \alpha')' \in X \times S$, the posterior mean is:

$$\overline{\theta} = \frac{\int_{X \times S} \theta \cdot \exp\{-h \sum \alpha_i F_i(x)\} p(\alpha) d\theta}{\int_{X \times S} \exp\{-h \sum_{i=1}^{n} a_i F_i(\chi)\} p(\chi) p(a) d\theta}. \tag{5}$$

This result goes back to Pinkus (1968) and we know that h must be "small". If we consider the kernel posterior:

$$p(x, \alpha, h | F) \propto \exp\left\{-h \sum_{i=1}^{n} \alpha_i F_i(x)\right\} p(x) p(\alpha) p(h), \tag{6}$$

for a certain prior $p(h)$ then h becomes part of the parameter vector. For example, we can use a *gamma* prior of the form:

$$p(h) \propto h^{a-1} \exp\{-bh\},\ h > 0,\ a, b > 0, \tag{7}$$

where the parameters a and b can be chosen so that the prior mean $E(h) = \frac{a}{b}$ is small and the prior variance $Var(h) = \frac{E(h)}{b}$ is also small. For example, we can set $a = 0.01$ and $b = \frac{a}{100}$. In this way, we do not have to worry about different values of h, although it might be useful to examine sensitivity. Integrating analytically h out of (6) using (7) we obtain:

$$p(\theta | F) \propto \left\{b + \sum_{i=1}^{n} \alpha_i F_i(x)\right\}^{-a}. \tag{8}$$

Further analytical integration with respect to x or α is not possible. Therefore, the posterior mean has to be computed numerically.

3 Numerical techniques

3.1 Independence Metropolis algorithm

One way to do so is to employ Markov Chain Monte Carlo (MCMC) techniques with the posterior kernel in (8). This depends on prior parameters a and b and not on h explicitly.

As the number of "observations", n, will, typically, be smaller than the number of parameters, which is $2k-1$ (k values of x and $k-1$ values in α) the prior must take care of whatever information we have about the problem. For example, it is likely that many of the elements of α are zero so that we do not care about certain functions. Alternatively, the xs may be expected to be concentrated in a certain subset of X. As an example, we can enforce a LASSO prior (least absolute shrinkage and selection operator, Tibshirani, 1997) on all elements of x and α if we expect some sparsity. Alternatively we may choose $\theta = (x', \alpha') \sim N(\underline{\theta}, \omega^2 I)$ for some parameter $\omega > 0$ and $\underline{\theta}$ is adjusted so that most values of θ are in $X \times S$.

As a MCMC method for inference we can use a Metropolis-Hastings sampler. Suppose $q(\theta)$ is a certain proposal or importance density from which random drawings can be generated easily and we wish to generate a large sample $\{\theta^{(s)}, s = 1, ..., S\}$ that converges in distribution to the distribution whose kernel is given by (8). Let a candidate draw from $q(\theta)$ be θ^c and we currently have $\theta^{(s)}$. Then, we set $\theta^{(s+1)} = \theta^c$ with probability $\min\left\{1, \frac{p(\theta^c|F)/q(\theta^c)}{p(\theta^{(s)}|F)/q(\theta^{(s)})}\right\}$, else we set $\theta^{(s+1)} = \theta^{(s)}$. Then the posterior mean in (5) can be estimated as: $\overline{\theta} \simeq S^{-1} \sum_{s=1}^{S} \theta^{(s)}$. This includes both an approximation to the solution of problem (3) plus estimates of the Pareto weights α.

3.2 Sequential Monte Carlo

Sometimes, MCMC can be slow to converge and, therefore, it is prevented from exploring fully the posterior, which means that sub-optimal values of the solution may be found. As an alternative, we can use Sequential Monte Carlo (SMC) techniques also known as particle filtering (PF). The version of SMC we use here was proposed by Durham and Geweke and seems to perform excellently in applications.

Chopin (2002) proposed a sequential PF for static models. Given a target posterior $p(\theta|Y) := p(\theta|Y_{1:T})$ a particle system is a sequence $\{\theta_j, w_j\}$ such that $E(h(\theta)|Y) := \int h(\theta)p(\theta|Y)d\theta \cong \lim_{J\to\infty} \frac{\sum_{j=1}^{J} w_j h(\theta_j)}{\sum_{j=1}^{J} w_j}$, almost surely, for any measurable function h, provided the expectation exists. We consider the sequence of posterior distributions $p_t := p(\theta|Y_t)$. The PF algorithm is as follows.

Step 1. Reweight: update the weights $w_j \leftarrow w_j \frac{p_{t+1}(\theta_j)}{p_t(\theta_j)}, j = 1, ..., J$.

Step 2: Resampling: resample $\{\theta_j, w_j\}_{j=1}^{H} \to \{\theta_j^r, 1\}_{j=1}^{J}$.

Step 3. Move: draw $\theta_j^m \sim K_{t+1}(\theta_j^r), j = 1, ..., J$, where K_{t+1} is any transition kernel whose stationary distribution is p_{t+1}.

Step 4. Loop: $t \leftarrow t+1, \{\theta_j, w_j\}_{j=1}^{J} \leftarrow \{\theta_j^m, 1\}_{j=1}^{J}$ and return to Step 1.

Chopin (2002) recommends the independence Metropolis algorithm to select the kernel, which requires a source distribution. A possible choice, if we sampled from p_n $(n < T)$, with respect to p_{n+s} is $\mathcal{N}(\hat{E}_{n+s}, \hat{V}_{n+s})$ where

$$\hat{E}_{n+s} = \frac{\sum_{j=1}^{J} w_j \theta_j}{\sum_{j=1}^{J} w_j}, \; \hat{V}_{n+s} = \frac{\sum_{j=1}^{J} w_j \left(\theta_j - E_{n+p}\right)\left(\theta_j - E_{n+p}\right)'}{\sum_{j=1}^{J} w_j}.$$

The strategy can be parallelized easily. If K processors are available, we can partition the particle system into K subsets, say $(S_k, k = 1, ..., K)$, and implement computations for particles of S_k in processor k. The algorithm can deal with new data at a nearly geometric rate and therefore the frequency of exhanging information between processors (after reweighting) decreases at a rate exponential to n, which is highly efficient.

Resampling according to $\theta_j^m \sim K_t(\theta_j^r, .)$ reduces particle degeneracy (Gilks and Berzuini, 2001) since identical replicates of a single particle are replaced by new ones without altering the stationary distribution. For this application using $J = 2^{12}$ particles gave a mean squared error in posterior means of 10^{-5} over 100 runs.

Chopin (2004) introduces a variation of MSC in which the observation dates at which each cycle terminates (say $t_1, ..., t_L$) and the parameters involved in specifying the Metropolis updates (say $\lambda_1, ..., \lambda_L$) are specified. Therefore, $0 = t_0 < t_1 < ... < t_L = T$ and we have the following scheme (we rely heavily on Durham and Geweke, 2013).

Step 1. Initialize $l = 0$ and $\theta_{jn}^{(l)} \sim p(\theta)$, $j \in \mathcal{J}, n \in \mathcal{N}$.

Step 2. For $l = 1, ..., L$:

(a) Correction phase:

(i) $w_{jn}(t_{l-1}) = 1$, $j \in \mathcal{J}, n \in \mathcal{N}$

(ii) For $s = t_{l-1} + 1, ..., t_l$

$$w_{jn}(s) = w_{jn}(s-1)p(y_s|y_{1:s-1}, \theta_{jn}^{(l-1)}),\ j \in \mathcal{J}, n \in \mathcal{N}.$$

(iii) $w_{jn}^{(l-1)} := w_{jn}(t_l)$, $j \in \mathcal{J}, n \in \mathcal{N}$.

(b) Selection phase, applied independently to each group $j \in \mathcal{J}$: Using multinomial or residual sampling based on $\left\{w_{jn}^{(l)}, n \in \mathcal{N}\right\}$, select

$$\{\theta_{jn}^{(l,0)}, n \in N\}$$

from $\{\theta_{jn}^{(l-1)}, n \in N\}$.

(c) Mutation phase, applied independently across $j \in \mathcal{J}, n \in \mathcal{N}$:

$$\theta_{jn}^{(l)} \sim p(\theta|y_{1:t}, \theta_{jn}^{(0)}, \lambda_l) \tag{9}$$

where the drawings are independent and the pdf above satisfies the invariance condition:

$$\int_{\Theta} p(\theta|y_{1:t_l}, \theta^*, \lambda_l)p(\theta^*|y_{1:t_l})d\nu(\theta^*) = p(\theta|y_{1:t_l}). \tag{10}$$

Step 3. $\theta_{jn} := \theta_{jn}^{(l)}$, $j \in \mathcal{J}, n \in \mathcal{N}$.

At the end of every cycle, the particles $\theta_{jn}^{(l)}$ have the same distribution $p(\theta|y_{1:t_l})$. The amount of dependence within each group depends upon the success of the Mutation phase which avoids degeneracy.

The SMC-PF algorithm produces draws from the posterior distribution of the parameters. As such, there is no formal "stopping criterion". In the statistical literature it is common to use 15,000 iterations the first 5,000 of which are discarded to mitigate possible start-up effects. Convergence to the posterior can be tested using, for example, Geweke's (1992) diagnostic. If convergence is rejected, one has to take more iterations and re-examine using the same diagnostics.

4 Empirical application

We follow closely Xidonas, Mavrotas and Psarras (2010) who develop a multi-objective mixed integer programming problem for stock selection in the Athens Stock Exchange, 66 stocks, weekly data from January 2004 to June 2007, T = 183. We use the same criteria as in Xidonas, Mavrotas and Psarras (2010), namely:

i) Capital return per share (+): $R_t = \frac{P_t - P_{t-1} + D_t}{P_{t-1}}$, where P_t is stock price and D_t is dividend yield.

ii) Relative dividend yield (+): Security's dividend yield/Subsector's dividend yield, where Dividend yield of a security = Dividend in period t/Share price closed in period t. See Hurson and Zopounidis (1995, 1997) and Zopounidis et al. (1998).

iii) Mean Absolute Deviation (-): $MAD_p = T^{-1}\sum_{t=1}^{T}|r_{pt} - E(r_p)| = T^{-1}\sum_{t=1}^{T}|\sum_{i=1}^{n} w_i[r_{it} - E(r_i)]|$, where n is the number of stocks, $E(r_i)$ is the expected capital return on stock i, $E(r_p)$ is the expected capital return of the portfolio and w_i is the amount invested in stock i.

iv) Beta coefficient (-): $cov(R_i, R_m)/var(R_m)$ where R_i is the return of stock i and R_m is the return of the market portfolio. This is a well known measure of risk.

v) Relative price-earnings ratio (-): Security's P/E/Subsector's P/E, where P/E = Share price in the stock market in period t / Earnings per share in period t.

vi) Marketability (+): Number of transactions of shares of a company during period t / Total number of shares of a company during period t (see Hurson and Zopounidis, 1995, 1997; Zopounidis et al., 1998).

The objective functions are as follows:

i) Maximize portfolio's return:

$$\max z_1 = \sum_{i=1}^{n} r_i X_i, \tag{11}$$

where r_i is the return of stock i and X_is are the decision variables.

ii) Maximize portfolio's dividend yield:

$$\max z_2 = \sum_{i=1}^{n} rd_i X_i, \tag{12}$$

where rd_i is the relative dividend's yield for stock i.

iii) Minimize portfolio's MAD:

$$\max z_3 = -T^{-1} \sum_{t=1}^{T} |\sum_{i=1}^{n} X_i (r_{it} - E(r_i))|. \tag{13}$$

We keep this constraint in its *nonlinear* form instead of using the Konno and Yamazaki (1991) transformation as in Xidonas et al. (2010). IN Xidonas et al. (2010) the cost of the linear approximation is the addition of 183 continuous variables and 366 constraints.

iv) Minimize portfolio's beta coefficient:

$$\max z_4 = -\sum_{i=1}^{n} b_i X_i, \tag{14}$$

where b_i is the beta coefficient for stock i.

v) Minimize portfolio's relative P/E ratio:

$$\max z_5 = -\sum_{i=1}^{n} rpe_i X_i, \tag{15}$$

where rpe_i is the relative P/E ratio for stock i.

vi) Maximize portfolio's marketability:

$$\max z_6 = \sum_{i=1}^{n} m_i X_i, \tag{16}$$

where m_i is the marketability index for stock i.

We have the following constraints:

$$\sum_{i=1}^{n} X_i = 1, \tag{17}$$

implying that all available capital is invested. We impose the constraint that we have at least 5 and at most 15 stocks in the portfolio:

$$5 \leq \sum_{i=1}^{n} B_i \leq 15, \tag{18}$$

where $B_i = 1$ if stock i is in the portfolio and 0 otherwise.

The maximal share of each stock cannot be more than 20%:

$$X_i - 0.2B_i \leq 0, \; i = 1, ..., n. \tag{19}$$

The minimum share must be 0.1%:

$$X_i - 0.001B_i \geq 0,\ i = 1, ..., n. \tag{20}$$

Upper bound for investment in stocks with negative average return (sectors 5, 11, 16):

$$\sum_{i \in S_1} X_i \leq 0.05. \tag{21}$$

Lower bound of investment in specific securities (sectors 1, 10, 61):

$$X_i \geq 0.05,\ i \in S_2. \tag{22}$$

Lower bound for the investment amount in securities with beta less than one (35 securities):

$$\sum_{i \in S_3} X_i \geq 0.5. \tag{23}$$

Lower bound for the investment amount in securities with high capitalization (41 securities):

$$\sum_{i \in S_4} X_i \geq 0.65. \tag{24}$$

All data are available in Table 2 of Xidonas et al. (2010). For the Metropolis algorithm we have used 15,000 iterations the first 5,000 of which are discarded to mitigate possible start-up and convergence effects. For the SMC algorithm we used the same configurations with 2^{16} particles. The results using 2^{14} or 2^{18} particles were identical. All constraints are treated using rejection sampling and sampling the binary variables is performed explicitly from their respective conditional posterior distributions. For each B_i its conditional posterior takes only two values, say f_{i0} and f_{i1}, corresponding to 0 and 1 respectively. After normalization the posterior conditional cdf takes values $\frac{f_{i0}}{f_{i0}+f_{i1}}$ and 1. Sampling random numbers from this distribution is, of course, trivial.

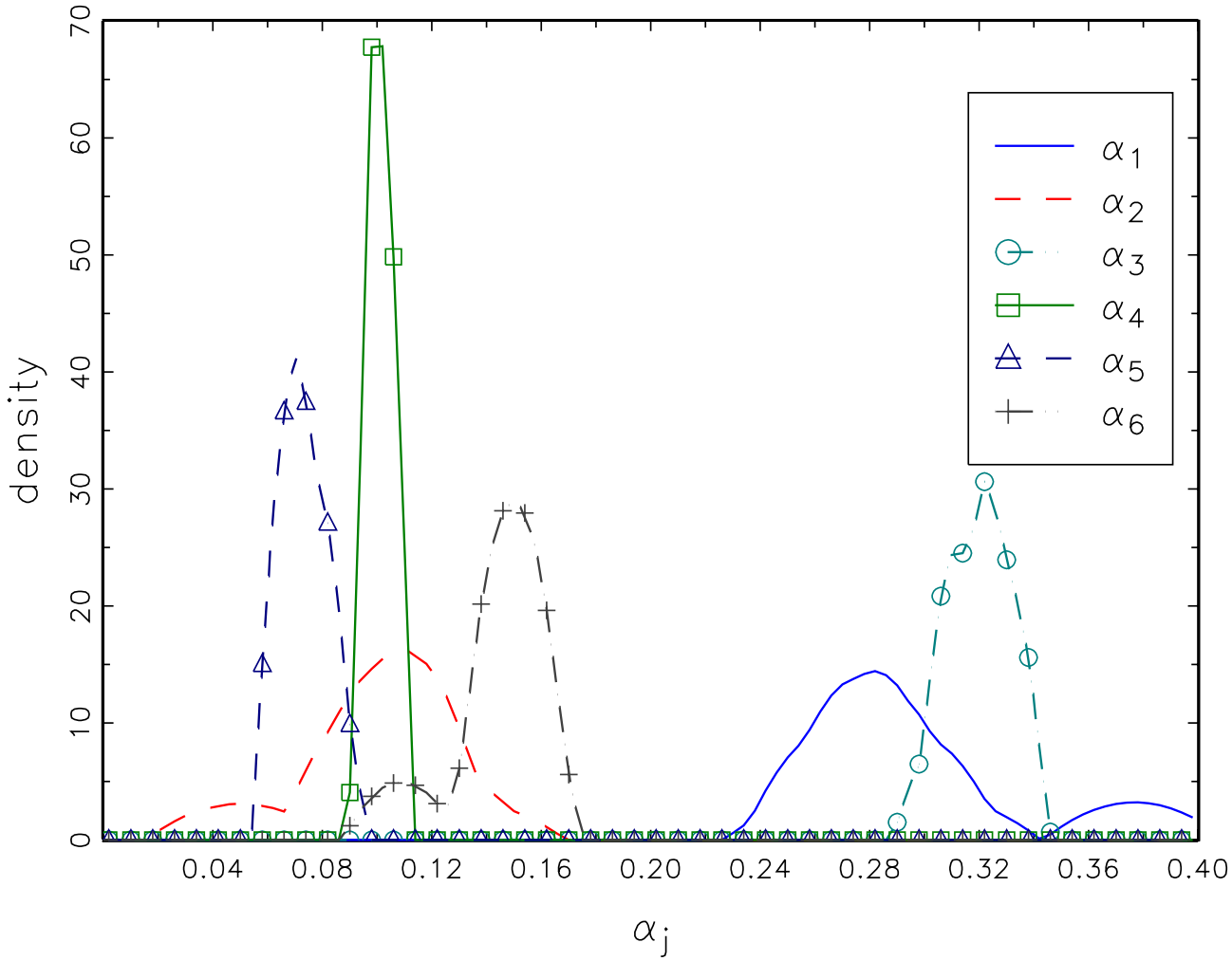
The output from the two algorithms is quite similar. In Table 1 below we summarize posterior moments for the Pareto weights, α.

Table 1. Posterior moments for the Pareto weights, α.

	post. mean	post s.d.
objective 1	0.281	0.045
objective 2	0.102	0.023
objective 3	0.317	0.014
objective 4	0.103	0.005
objective 5	0.072	0.013
objective 6	0.125	0.017

The marginal posteriors are reported in Figure 1. Evidently, the marginal posterior distributions are highly nonnormal and show preference over objectives 3 and 1 which account jointly for almost 40%. The marginal posterior of $\alpha_1$is clearly bimodal with modes around 28% and 37% so the first two objectives can account up to nearly 60%. The other objectives receive non-trivial weights, for example objective 6 receives weight 12.5% followed by objectives 4 and 2 (10.3% and 10.2% respectively). Finally, objective 5 receives only 7.2% on the average, although from Figure 1 it is evident that this can be between 4% and 10% with high posterior probability.

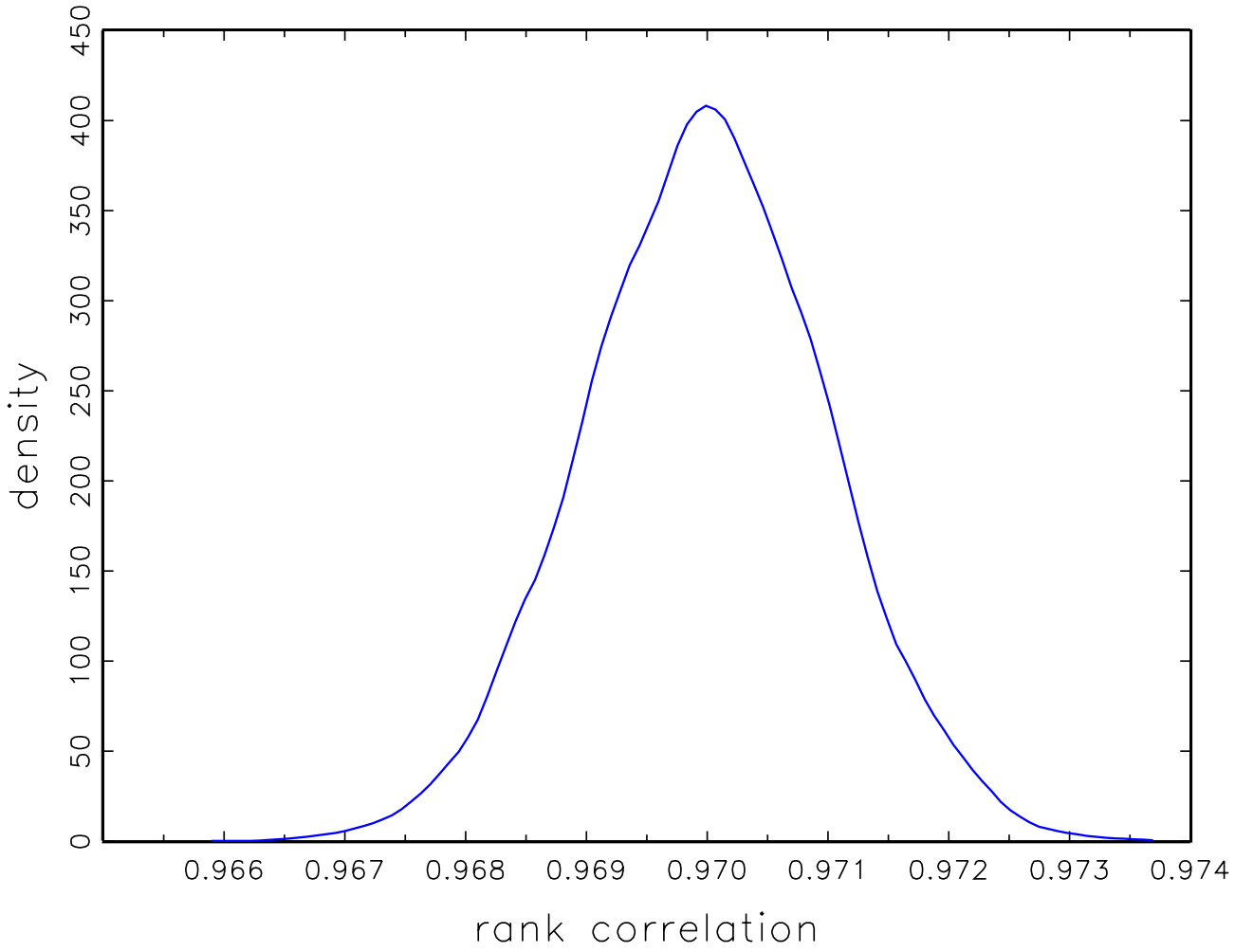
Figure 1. Marginal posterior distributions of Pareto weights



To understand how close are our results to those in portfolio 153 in Xidonas et al. (2010), the maximum absolute difference is less than 0.01% and the rank correlation of X_is is 98%. Portfolio 153 was found to "combine satisfactory rate of return with satisfactory relative dividend yield, relative P/E and MAD, in comparison to the other portfolios." according to experts they consulted. Therefore, the posterior-mean-solution replicates successfully the best portfolio which was, in fact, generated from 268 different portfolia in Xidonas et al. (2010) using a quite different methodology, based on successive filtering of the original Pareto solutions.

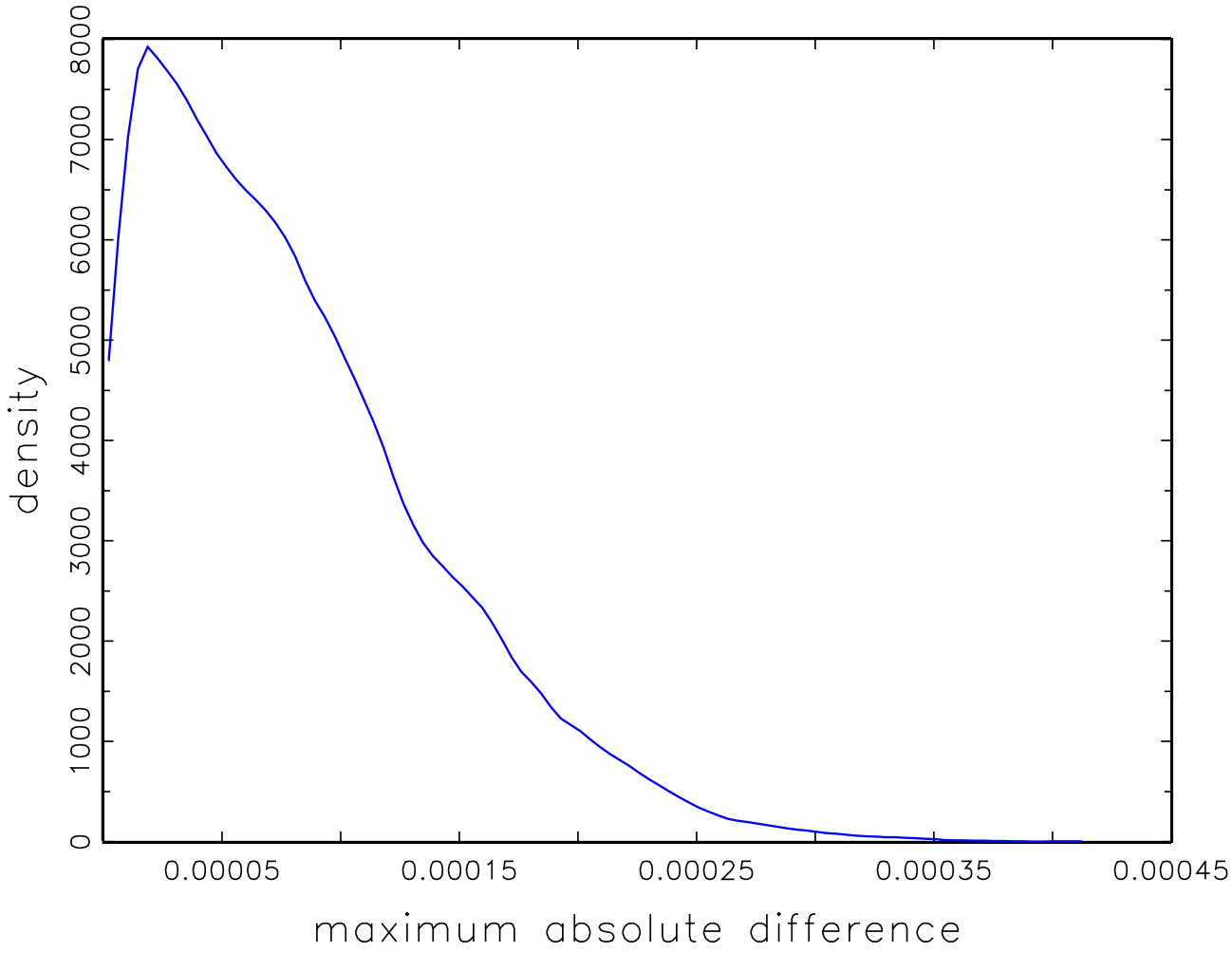
To refine our understarding of the closeness of the solution to portfolio 153 in Xidonas et al. (2010) we perform the following experiment to embed portfolio 153 in a stochastic setting. We set equal all Pareto weights and treat them as constant. In turn, we craft a prior which pre-assigns the B_is equal to those in portfolio 153. The prior is normal for the X_is (conditional on the B_is) with equal means and standard deviations all equal to 1. The standard deviations are quite large, yet our Metropolis and SMC algorithms converge quickly to posterior means which are very close to the values of portfolio 153 in Xidonas et al. (2010). After convergence (which takes 5,000 replications) we have two large sequences: One from our SMC (see Table 1 and Figure 1) and another from SMC conditional on the B_is in portfolio 153 in Xidonas et al. (2010) which we will call the "Xidonas et al. (2010) SMC sequence". Each sequence consists of 10,000 replications of $\{X_i, i = 1, ..., 66\}$. For each one of the 10,000 replications we can evaluate the rank correlation between our own SMC and the Xidonas et al. (2010) SMC sequence. The results are reported in graphical form in Figure 2. In Figure 3, we report the maximum absolute difference of the elements of X_is between our own SMC and the Xidonas et al. (2010) SMC sequence.

Figure 2. Rank correlations of X_is between SMC and Xidonas et al. (2010) SMC sequence



From Figure 2, it is evident that SMC replicates almost perfectly the global optimum in Xidonas et al. (2010) in all iterations of the two SMC schemes that we considered.

Figure 3. maximum absolute difference of the elements of X_is between and the Xidonas et al. (2010) SMC.



The maximum absolute difference of the elements of X_is between our own SMC and the Xidonas et al. (2010) SMC sequence turn out to be quite trivial as we can see in Figure 4, where the grand maximum is not larger than 0.0005 and less than 0.00015 in terms of its median, across all SMC replications. In terms of timing Xidonas et al. (2010) report 249 s in a Pentium Core 2 Duo 2.0 GHz. We have used High End Computing (HEC). The combined facility offers over 5,000 CPU cores, 23 TB aggregate of memory, 70TB of high performance filestore and 1.5PB of medium performance filestore. A number of nodes offer Nvidia GPU cards, which suport CUDA and OpenCL applications. Xidonas et al. (2010) a mixed-integer programming formulation (Exler et al., 2010) takes about 7.7 s, Metropolis-Hastings 11.12 s and SMC 22.13 s. The timings, of course do not account for the fact that a three-pass filter is required in Xidonas et al. (2010) after solving the problem to find the optimal solution, and requires expert opinion to judge different portfolios. The timings of the Metropolis-Hastings and SMC are trivial given the size and difficulty of the problem.

5 Further Validation

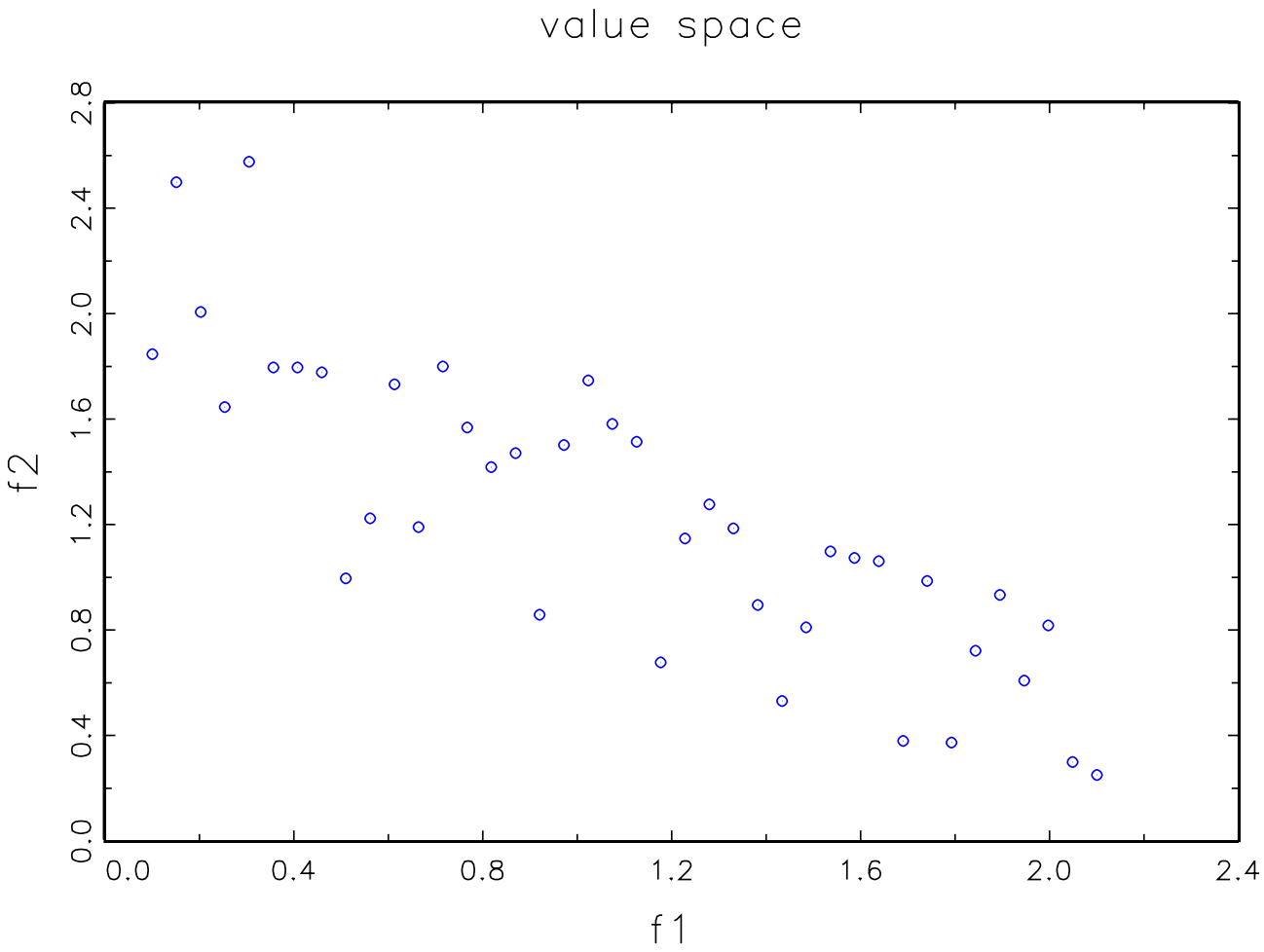
As extra validation we use the example presented in Qu et al. (2014):

$$
\begin{gathered}
F(x) = (F_1(x), F_2(x)) = (\max\{F_{11}(x),\ F_{12}\{x\}, \max\{F_{21}(x),\ F_{22}(x)) \\
\text{where} \\
F_{11}(x) = -1 + 8x_1 + 8x_2 - 32x_1x_2, \\
F_{12}(x) = 3.6 - 12x_1 - 4x_3 + 4x_1x_3 + 10x_1^2 + 2x_3^2, \\
F_{21}(x) = x_1 + x_2^2 + x_3^2 + \sin^2(x_1 + x_2 + x_3), \\
F_{22}(x) = \cos(x_3)(0.1 + x_2)\left\{\exp\left(-\tfrac{x_1}{0.1+x_3}\right)\right\}, \\
-10^{-4} < x_i < 1 + 10^{-4},\ i = 1, 2, 3.
\end{gathered}
\tag{25}
$$

In Qu et al. (2014) the weights assigned to two objectives F_1 andF_2 are $\frac{1}{2}$ and $\frac{1}{2}$ respectively. They used the the non-smooth trust region method proposed by Qu, Goh, and Liang (2013). We solve the same problem using different pre-assigned Pareto weights to find different optima which we present in Figure 4 which is the analogue of Figure 1 in Qu et al. (2014). We apply the SMC algorithm using 15,000 iterations the first 5,000 of which are discarded. This configuration is probably excessive as the same results were obtained using 2,200 iterations discarding the first 200. We use 10^7 particles although the results were the same when we used only 10^4 particles.

Figure 4. Value space.

Value space for 40 different pre-assigned Pareto weights. The circles indicate the "critical points" (posterior means) we found using SMC.



In Table 2 we provide timings and a comparison with Qu et al. (2013).

Table 2. Timings (average CPU) of different algorithms

Qu et al. (2013)[(a)]	1.000
subgradient[(a)]	1.209
scalarization[(a)]	1.025
SMC, expensive choices[(b)]	3.212
SMC, cheap choices[(c)]	2.503

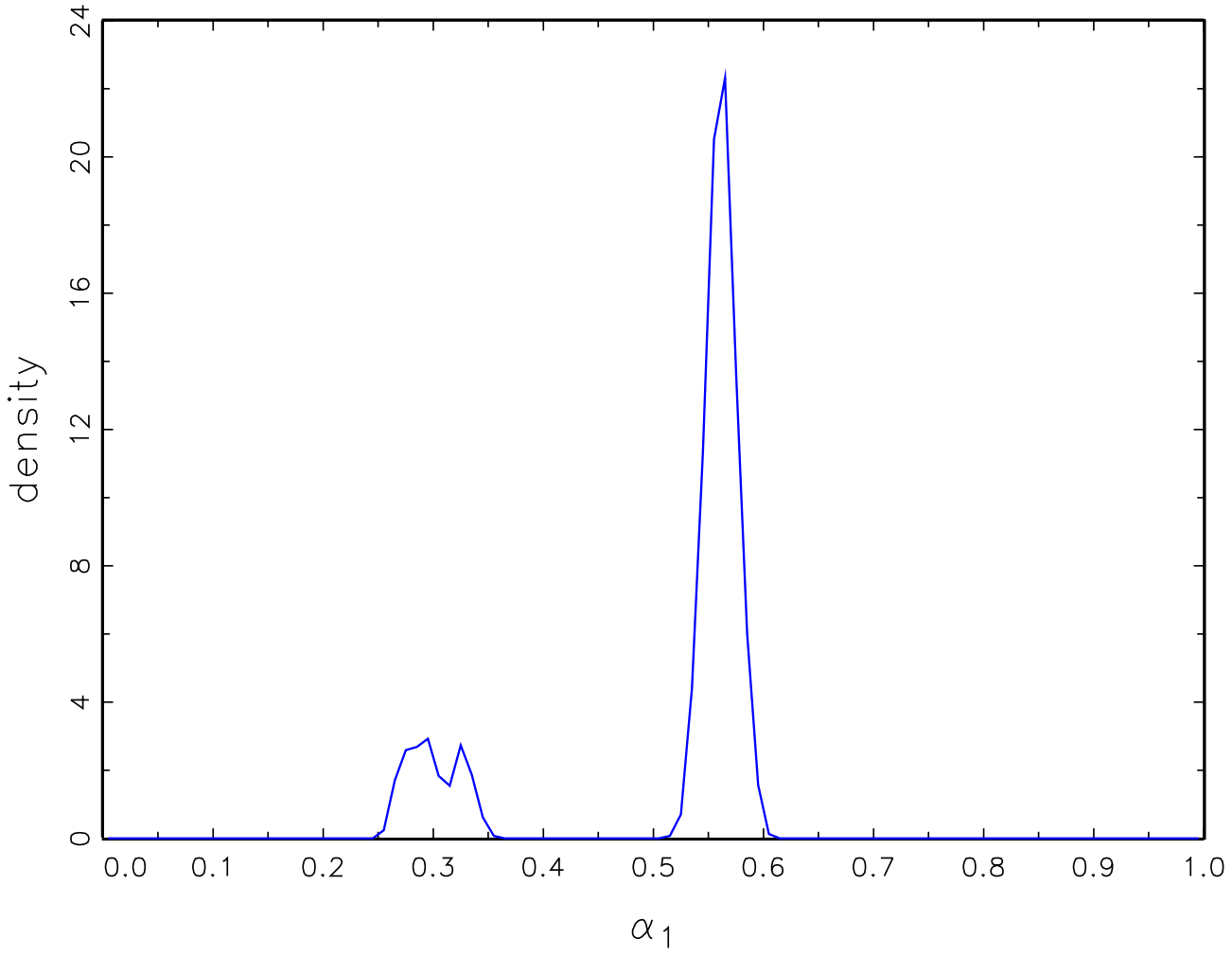
Notes: (a) Taken from Qu et al. (2013) and normalizing to 1.000 their timing which is 0.0234 CPU s. (b) The SMC algorithm using 15,000 iterations the first 5,000 of which are discarded, using 10^7 particles. (c) The SMC algorithm using 2,200 iterations the first 200 of which are discarded, using 10^4 particles.

From Table 2 it is evident that timings of SMC are competitive to Qu et al. (2013) or other quasi-Newton methods although higher: This is a well known drawback of Monte Carlo-based methods. However, timings still remain trivial from the point of view of the user due to parallelization of SMC. An advantage of our technique is that we can vary systematically the Pareto weights to trace out all Pareto optima, a task that would be quite difficult in Qu et al. (2013).

Finally, we take up global optimization in problem (25). The marginal posterior distribution of the first Parteto weight,α_1, is reported in Figure 5. The distribution is bimodal with a mode near 0.3 and another near 0.52. As our priors are flat, the

bimodality is likely due to the tension among the different objective functions in (25). It arises also in the portfolio problem that we examined in the previous section and it seems unavoidable in multiobjective programming problems where the objectives point, naturally, to very different solutions.

Figure 5. Marginal posterior distribution of the first Pareto weight,α_1.



6 Concluding remarks

In this work we developed a statistical framework for multicriteria decision making problems by embedding them in a class of Bayesian posterior distributions. The idea of using posterior distributions in optimization dates back to Pinkus (1968). In this paper, we relaxed the assumption that we need a certain parameter h so that the posterior mean converges to the global optimum. Our new approach to multiobjective programming considers an equivalent posterior disrtibution for the decision variables *and* the Pareto weights jointly, which are often unknown. Therefore, statistical inferences can be performed for the decision variables and the Pareto weights jointly. Of course the Pareto weights can be fixed in advance and, as a result, we can trace out the entire Pareto frontier, a task that is difficult when gradient-based or mixed integer programming - based methods are used. We applied the new technique to a portfolio problem previously analyzed by Xidonas et al. (2010) and a test problem proposed by Qu et al. (2013). The performance of the new approach yields similar if not identical results and timings are competitive although higher: This is expected for any Monte Carlo approach to the problem, for example simulated annealing (Zhou and Chen, 2013).

The new technique can be considered as an extension of Zhou and Chen (2013) to the multicriteria decision making. The important features of the technique are: i) The Pinkus (1968) parameter h is no longer needed if we adopt a conditionally conjugate prior for this parameter. ii) Posterior statistical inference for both the deicision variables and the Pareto weights is possible. iii) Fixing the Pareto weights and tracing out the entire Pareto frontier can be performed in a systematic way without the need for multistart gradient-based or mixed integer programming algorithms. iv) Binary or discrete variables can be handled in a natural way as we showed in connection to the multicriteria portfolio problem of Xidonas et al. (2013).

The good performance of new techniques suggests that SMC and other algorithms, like the classical Metropolis-Hastings algorithm, can be used profitably in the context of multicriteria decision making problems to trace out the Pareto frontier and / or find a global optimum. Most importantly SMC can be considered as an off-the-shelf technique to solve arbitrary multicriteria decision making problems routinely and efficiently. One may argue that the proposed approach is working in an uncontrollable way and may produce arbitrary results, because here the decision maker is replaced by the data and the data may change over time. However, by relying on the data we provide an objective approach to the problem. Of course, if a decision maker is available, she can assign the weights and a Pareto solution can be found.

References

Angilella Salvatore Corrente S., , S. Greco, R. Słowiński (2017). Robust Ordinal Regression and Stochastic Multiobjective Acceptability Analysis in multiple criteria hierarchy process for the Choquet integral preference model. Omega 63, 154-169.

Cardoso, T., M. Duarte Oliveira, A. Barbosa-Póvoa, S. Nickel (2016). Moving towards an equitable long-term care network: A multi-objective and multi-period planning approach. Omega 58, 69-85.

Chica, M., J. Bautista, Ó. Cordón, S. Damas (2016). A multiobjective model and evolutionary algorithms for robust time and space assembly line balancing under uncertain demand. Omega 58, 55-68.

Chopin, N. (2002). A sequential particle filter method for static models, Biometrika 89 (3) 539-551.

Chopin, N. (2004). Central limit theorem for sequential Monte Carlo methods and its application to Bayesian inference. Annals of Statistics 32, 2385-2411.

Colson, G., Zeleny, M. (1979). Uncertain Prospects Ranking and Portfolio Analysis Under the Condition of Partial Information Mathematical Systems in Economics, vol. 44. Verlag Anton Hain, Maisenheim.

Colson, G., Zeleny, M. (1980). Multicriterion concept of risk under incomplete information. Computers and Operations Research 7(1/2), 125–143.

Das, I., & Dennis, J. E. (1998). Normal-boundary intersection: A new method for generating the Pareto surface in nonlinear multicriteria optimization problem. SIAM Journal on Optimization, 8(3), 931–967.

Donoso, Y., & Fabregat, R. (2007). Multiobjective optimization in computer networks using Metaheuristics, Auerbach, New York.

Durham, G., and J. Geweke, 2013, Adaptive sequential posterior simulators for massively parallel computing environments. working paper.

Exler, O., T. Lehmann, K. Schittkowski (2010). MINLPB4: A Fortran Subroutine for Mixed-Integer Nonlinear Optimization by Branch-and-Bound. Department of Computer Science University of Bayreuth D-95440 Bayreuth. http://www.ai7.uni-bayreuth.de/MINLPB4.pdf

Fliege, L., Grana drummond, L. M., & Svaiter, B. F. (2009). Newton's method for multiobjective optimization. SIAM Journal on Optimization (2), 602–626.

Geweke, J. (1992). Evaluating the Accuracy of Sampling-Based Approaches to the Calculation of Posterior Moments. In Bayesian Statistics 4 (eds. J.M. Bernardo, J. Berger, A.P. Dawid and A.F.M. Smith), Oxford: Oxford University Press, 169-193.

Gilks, W.R., and C. Berzuini, 2001 Following a moving target: Monte Carlo inference for dynamic Bayesian models, Journal of the Royal Statistical Society B 63, 127-146.

Handi, J., Kell, D. B., & Knowles, J. (2007). Multiobjective optimization in bioinformatics and computational biology. IEEE/ACM Transactions on Computational Biology and Bioinformatics, 4(2), 279–290.

Hurson, C., Zopounidis, C. (1993). Return, risk measures and multicriteria decision support for portfolio selection. In: Papathanassiou, B., Paparrizos, K. (eds.) Proceedings of the 2nd Balkan conference on operational research, pp. 343–357.

Hurson, C., Zopounidis, C. (1995) On the use of multicriteria decision aid methods to portfolio selection. Journal of Euro-Asian Management 1(2), 69–94.

Hurson, C., Zopounidis, C. (1997) Gestion de portefeuille et analyse multicritere. Economica, Paris.

Kadziński, M., T. Tervonen, M. K. Tomczyk, R. Dekker (2017). Evaluation of multi-objective optimization approaches for solving green supply chain design problems. Omega 68, 168-184.

Konno, H., Yamazaki, H. (1991). Mean-absolute deviation portfolio optimization model and its application to Tokyo stock market. Management Science 37(5), 519–531.

Mavrotas, G., J.R. Figueira, E. Siskos (2015). Robustness analysis methodology for multi-objective combinatorial optimization problems and application to project selection. Omega 52, 142-155.

Mavrotas, G., Xidonas, P., Psarras, J. (2008). An integrated multiple criteria methodology for supporting common stock portfolio selection decisions. In: Lahdelma, R.,Miettinen, K., Salminen, P., Salo,A. (eds.) Proceedings of the 67th meeting of the European working group on multiple criteria decision aiding. Rovaniemi, Finland, pp. 56–71.

Neto, J. X. D. C., Silva, G. J. P. D., Ferreira, O. P., & Lopes, J. O. (2013). A subgradient method for multiobjective optimization. Computational Optimization and Applications, 54, 461–472.

Paul, N.R., B. J. Lunday, S. G. Nurre (2017). A multiobjective, maximal conditional covering location problem applied to the relocation of hierarchical emergency response facilities. Omega 66, 147-158.

Pinkus, M. (1968). A Closed-Form Solution of Certain Programming Problems, Operations Research 16: 690-694.

Qu, S. J., Goh, M., & Chan, F. T. S. (2011). Quasi-Newton methods for solving multiobjective optimization. Operations Research Letters, 39, 397–399.

Qu, S. J., Goh, M., & Liang, B. (2013). Trust region methods for solving multiobjective optimisation. Optimization Methods and Software, 28, 796–811.

Qu, S., C. Liu, M. Goh, Y. Li, Y. Ji (2014). Nonsmooth multiobjective programming with quasi-Newton methods. European Journal of Operational Research 235, 503-510.

Spronk, J., Hallerbach, W.G. (1997). Financial modelling: where to go? With an illustration for portfolio management. European Journal of Operational Research 99(1), 113–127.

Teresinha Arns Steiner, M., D. Datta, P.J. Steiner Neto, C.T. Scarpin, J.R. Figueira (2015). Multi-objective optimization in partitioning the healthcare system of Parana State in Brazil. Omega 52, 53-64.

Steuer, R.E., Qi, Y., Hirschberger, M. (2005) Multiple objectives in portfolio selection. Journal of Financial Decision Making 1(1), 11–26.

Steuer, R.E., Qi, Y., Hirschberger, M. (2006a) Developments in multi-attribute portfolio selection. In: Trzaskalik, T. Multiple Criteria Decision Making '05, pp. 251–262. Karol Adamiecki Press, Katowice Poland.

Steuer, R.E., Qi, Y., Hirschberger,M. (2006b) Portfolio optimization: new capabilities and future methods. Zeitschrift für Betriebswirtschaft 76(2), 199–219.

Tibshirani, R. (1997). The lasso Method for Variable Selection in the Cox Model. Statistics in Medicine, Vol. 16, 385—395.

Tsai, S.C., and S.T. Chen (2017). A simulation-based multi-objective optimization framework: A case study on inventory management. Omega 70, 148-159.

Vieira, D. A. G., Takahashi, R. H. C., & Saldnha, R. R. (2012). Multicriteria optimization with a multiobjective golden section line search. Mathematical Programming, Series A, 131, 131–161.

Xidonas, P., G. Mavrotas, J. Psarras (2010). Equity portfolio construction and selection using multiobjective mathematical programming. Journal of Global Optimization 47, 185-209.

Xidonas, P., Psarras, J. (2009). Equity portfolio management within the MCDM frame: A literature review. Int. J. Banking Acc. Finance 1 (3), 285–309.

Xidonas, P., Askounis, D., Psarras, J. (2009a) Common stock portfolio selection: a multiple criteria decision making methodology and an application on the Athens stock exchange. Oper. Res. Int. J. 9(1), 55–79.

Xidonas, P., Mavrotas, G., Psarras, J. (2009b). A multicriteria methodology for equity selection using financial analysis. Comput. Oper. Res. 36(12), 3187–3203

Xidonas, P.,Mavrotas, G., Psarras, J. (2009c) A multiple criteria decision making approach for the selection of stocks. J. Oper. Res. Soc. Palgrave doi:10.1057/jors.2009.74

Xidonas P., Mavrotas G., Psarras J. (2009d) Portfolio construction on the Athens stock exchange: a multiobjective optimization approach. Optimization, doi:10.1080/02331930903085375

Zeleny, M.(1977). Multidimensional measure of risk: The prospect ranking vector. In: Zionts, S. (ed.) Multiple Criteria Problem Solving pp. 529–548. Springer, Heidelberg.

Zeleny, M. (1981). Satisficing optimization, and risk in portfolio selection. In: Derkindreen, F.G.H., Crum, R.L. (eds.). Readings in Strategies for Corporate Investment, pp. 200–219. Pitman Publishing, Boston.

Zeleny, M. (1982). Multiple Criteria Decision Making. McGraw Hill, New York.

Zhou, E., X. Chen (2013). Sequential Monte Carlo simulated annealing. Journal of Global Optimization 55 (1), 101-124.

Zopounidis, C. (1999). Multicriteria decision aid in financial management. European Journal of Operational Research 119(2), 404–415.

Zopounidis, C., Despotis, D.K., Kamaratou, I. (1998). Portfolio selection using the ADELAIS multiobjective linear programming system. Computational Economics 11/3, 189–204.

Zopounidis, C., Doumpos, M. (2002). Multicriteria decision aid in financial decision making: methodologies and literature review. Journal of Multicriteria Decision Analysis 11, 167–186.