Improved block rearrangement algorithm

Carole Bernard*, Jinghui Chen†, Ludger Rüschendorf‡ and Steven Vanduffel§

December 21, 2024

Abstract

In the context of finding risk bounds for portfolios of risks, Puccetti and Rüschendorf (2012) introduce the rearrangement algorithm (RA) as a tool for (optimally) rearranging matrices by permuting, in each step, the elements of a given column. The RA also has applications in finance and operations research. Bernard and McLeish (2016) and Bernard et al. (2017) show that, in principle, better results can be expected by permuting the rows of randomly chosen blocks of the matrix. They label such an algorithm the block rearrangement algorithm (BRA). Various versions of BRA exist, and they mainly differ with respect to the manner in which the blocks (i.e., the submatrices) are chosen in each step.

In this paper, we aim to develop an improved version of BRA based on a dynamic choice of block sizes. That is, we seek to find the optimal sequence of block (submatrix) sizes. To achieve this, we refine the BRA by sampling the block size, r_t, at the t-th step from a Beta distribution with two parameters that evolve over the different steps. The proposed BRA Beta is designed to select large block sizes initially (similar to BRA Binomial), and then transition to smaller sizes (resembling the RA). A numerical study demonstrates that BRA Beta outperforms other variants of the RA available in the literature. For example, the improvement in variance reduction achieved by BRA Beta is double that of other BRA-based algorithms when the risks are heterogeneous and the portfolio is large (see Section 5).

Keywords: Rearrangement algorithm, Risk bounds, Aggregation risk, Variance of sum, Copula.

*Corresponding author: Carole Bernard, Department of Accounting, Law and Finance, Grenoble Ecole de Management and Department of Business at Vrije Universiteit Brussel. (email: carole.bernard@grenoble-em.com).

†Jinghui Chen, Department of Statistical Sciences at University of Toronto and Department of Mathematics and Statistics at York University, RISC Foundation. (email: jh8chen@yorku.ca).

‡Ludger Rüschendorf, Department of Stochastics at University of Freiburg. (email: ruschen@stochastik.uni-freiburg.de).

§Steven Vanduffel, Department of Business at Vrije Universiteit Brussel. (email: steven.vanduffel@vub.be).

1 Introduction

Let X_i be random variables assumed to be square integrable and with given marginal distributions F_i, $i = 1, \cdots, d$. A classic dependence problem is finding the bounds on the expectation of $\psi(X_1, \ldots, X_d)$, in which $\psi : \mathbb{R}^d \to \mathbb{R}$ is a measurable function, i.e., we aim to determine

$$m_\psi = \inf\{\mathbb{E}\psi(X_1, \ldots, X_d);\ X_j \sim F_j,\ 1 \le j \le d\}, \tag{1.1a}$$

$$M_\psi = \sup\{\mathbb{E}\psi(X_1, \ldots, X_d);\ X_j \sim F_j,\ 1 \le j \le d\}. \tag{1.1b}$$

These two problems are related to various topics in operations research, statistics, and quantitative risk management. We refer to Puccetti and Rüschendorf (2012) for applications in statistics, Embrechts and Puccetti (2010) for quantitative risk management and Hsu (1984), Jakobsons and Wang (2016) and Boudt et al. (2018) for applications in operations research. We also refer to Rüschendorf et al. (2024) for a detailed account. There are many problem instances for (1.1a) and (1.1b). For the case in which $\psi(X_1, \ldots, X_d) := f\left(\sum_{j=1}^d X_j\right)$ such that f is a convex function (risk aggregation), Wang and Wang (2011) study the problem when the distributions of X_j are completely mixable, leading to a solution of (1.1a). We refer to Embrechts and Puccetti (2006) and Puccetti and Rüschendorf (2013) for the tail risk, where $\psi(X_1, \ldots, X_d) := \mathbb{1}_{X_1+\cdots+X_d>x},\ x \in \mathbb{R}$. Puccetti and Rüschendorf (2015) put forward a numerical method to approximate (1.1a) and (1.1b) when ψ is a supermodular function. Moreover, Bernard et al. (2023) derive the analytic solutions for (1.1a) and (1.1b) when ψ is the product, i.e., when $\psi(X_1, \ldots, X_d) := \prod_{j=1}^d X_j$, and specific constraints on F_j are satisfied. However, no general analytic result exists to solve problems (1.1a) and (1.1b).

Puccetti and Rüschendorf (2012) introduce the rearrangement algorithm (RA) to approximate the bounds m_ψ and M_ψ numerically when $\psi(X_1, \cdots, X_d) := \mathbb{1}_{f(X_1,\ldots,X_d)>x},\ x \in \mathbb{R}$ and the function f is strictly increasing in each coordinate, based on equivalent formulation for the problem in Rüschendorf (1983). The RA rearranges matrices by permuting the elements of a given column at each step. Puccetti and Rüschendorf (2013) show that the RA is fast, accurate and can be used to determine the bounds on the distribution of $\sum_{j=1}^d X_j$. Bernard and McLeish (2016) and Bernard et al. (2017) demonstrate that better m_ψ and M_ψ can theoretically be achieved by permuting the rows of randomly chosen blocks of the matrix and refer it as the block rearrangement algorithm (BRA). Different versions of BRA exist, distinguished primarily by how the blocks (submatrices) are selected at each step. For example, Boudt et al. (2018) propose selecting blocks such that the variances of the row sums are as equal as possible, naming this variation the BRA with Variance Equalization (BRAVE). In contrast, Bernard and McLeish (2016) and Bernard et al. (2017) randomly determine block sizes at each step using a binomial distribution, a version referred to here as the BRA Binomial.

In this paper, we intend to study the factors that influence the convergence and accuracy of the BRA algorithm. Specifically, we focus on a key factor: the block size of the submatrix used to be rearranged in each step. Based on this analysis, we propose an enhanced version of the BRA called BRA Beta, aimed at improving both the accuracy and speed of convergence. Our experimental findings show that, in the important context of a relatively high dimensional case, BRA Beta significantly outperforms other existing variants of the rearrangement algorithm reported in the literature. In addition to the rearrangement algorithms, the simulated annealing (SA) algorithm (Kirkpatrick et al., 1983) can also solve these operations research and risk management problems. The SA algorithm is one of the simplest meta-heuristic methods to address global optimization problems. For a comprehensive introduction to the SA algorithm, we refer to Van Laarhoven et al. (1987). However, the SA algorithm is significantly more time-consuming—often thousands of times slower than the BRA Beta—particularly when dealing with large portfolios. Boudt et al. (2018) also illustrate the advantages of these BRA-based methods in operations research by using them to deal with the classic number partitioning problem in operations research. Specifically, these authors show how a well-designed BRA, i.e., BRAVE, outperforms the seminal Karmarkar-Karp differencing algorithm. In our paper, we compare the different versions of the BRA in assessing model risk uncertainty for measures such as Value-at-Risk (VaR), Tail VaR, and variance.

This paper presents a novel contribution that addresses the crucial issue in the block size (i.e. the size of the submatrix) selection for BRA. Unlike previous methods that utilize constant (time-independent) cardinality selection techniques, such as RA and BRA Binomial, the proposed BRA Beta method introduces a time-dependent approach that uses Beta distributions with dynamic parameters α_t and β_t. These time-dependent parameters govern the selection of block sizes in a flexible and adaptive manner. BRA Beta aims to improve on existing (B)RA algorithms by using a more effective and more efficient approach for block size selection.

The structure of the paper is as follows: In Section 2, we recall the theoretical considerations that lead to BRA and discuss its different versions. Section 3 investigates the factors that impact the performance of BRA. We propose in Section 4 an enhanced version of the BRA, called BRA Beta. The performance of BRA Beta is evaluated in Section 5, where we compare it with RA and several variants of BRA. In Section 6, we apply BRA Beta to approximate the upper bound on VaR for a portfolio of risks. Finally, we present our conclusions in the last section.

2 A combinatorial problem

In this section, we first recast problem (1.1) as a rearrangement problem; see also Rüschendorf et al. (2024) for more detail. We then consider a discretized problem and argue that its

solutions are approximations for the solutions of the rearrangement problem. Finally, we reformulate the BRA as an optimization technique to solve the discretized problem.

2.1 Rearrangement of functions

Let $h, g : \Omega \to \bar{\mathbb{R}}$ be integrable functions on a probability space $(\Omega, \mathcal{F}, P)$. If h and g have the same distribution function F, we say g is a rearrangement of h (see the introduction of the rearrangements of functions in Hardy et al. (1952) and the applications of them in Luxemburg (1967), Chong and Rice (1971) and Day (1972)). For functions h and g in L^1, it is possible to construct non-decreasing and non-increasing rearrangements, i.e., h^*, g^* and h_*, g_*, respectively. In particular,

$$\int h^* g_* dP \leq \int h g dP \leq \int h^* g^* dP. \tag{2.1}$$

Moreover, $\int h^* g_* dP = \int h_* g^* dP$ and $\int h^* g^* dP = \int h_* g_* dP$. Let $\mathcal{M}(F_1, \ldots, F_d)$ denote the class of all distributions with marginals F_j, $j = 1, \ldots, d$ and all possible dependence structures. The following classical lemma is recalled from Rüschendorf (1983).

Lemma 2.1 (Fréchet class)**.** *Let U be a standard uniform random variable, i.e. $U \sim U(0,1)$. Then*

$$\mathcal{M}(F_1, \ldots, F_d) = \{P^{(h_1(U), \ldots, h_d(U))};\ h_j \text{ is a rearrangement of } F_j^{-1},\ 1 \leq j \leq d\}. \tag{2.2}$$

Let ψ be a supermodular function and coordinate-wise increasing. Then, problem (1.1) can be reformulated using Lemma 2.1 by the following rearrangement problems:

$$\begin{aligned} m_\psi &= \inf\{\mathbb{E}\psi(X_1, \ldots, X_d);\ X_j \sim F_j,\ 1 \leq j \leq d\} \\ &= \inf\{\mathbb{E}\psi(f_1(U), \ldots, f_d(U));\ f_j \text{ is a rearrangement of } F_j^{-1},\ 1 \leq j \leq d\}, \qquad (2.3a) \\ M_\psi &= \sup\{\mathbb{E}\psi(X_1, \ldots, X_d);\ X_j \sim F_j,\ 1 \leq j \leq d\} \\ &= \sup\{\mathbb{E}\psi(f_1(U), \ldots, f_d(U));\ f_j \text{ is a rearrangement of } F_j^{-1},\ 1 \leq j \leq d\}. \qquad (2.3b) \end{aligned}$$

Let $\mathbf{Y} = (Y_1, \ldots, Y_d) \in \mathbb{R}^d$ and I be respective a given random vector and a subset of $\{1, \ldots, d\}$. Moreover, $I^c = \{1, \ldots, d\} \backslash I$ and $r = |I|$ denotes the cardinality of I. We assume that for any complementary subvectors $\mathbf{Y}_1 = ((Y_j)_{j \in I}) \in \mathbb{R}^r$ and $\mathbf{Y}_2 = ((Y_j)_{j \in I^c}) \in \mathbb{R}^{d-r}$, there exist measurable functions $\psi_I^r : \mathbb{R}^r \to \mathbb{R}$, $\psi_{I^c}^{d-r} : \mathbb{R}^{d-r} \to \mathbb{R}$ and $\psi_I^2 : \mathbb{R}^2 \to \mathbb{R}$ such that

$$\psi(Y_1, \ldots, Y_d) = \psi_I^2(\psi_I^r(\boldsymbol{Y}_1), \psi_{I^c}^{d-r}(\boldsymbol{Y}_2)), \quad 1 \leq r \leq \left\lfloor \frac{d}{2} \right\rfloor, \tag{2.4}$$

see Puccetti and Rüschendorf (2012, 2015) when $r = 1$. ψ_I^2 can be the sum, $\psi_I^2(Y_1, Y_2) = Y_1 + Y_2$, the product, $\psi_I^2(Y_1, Y_2) = Y_1 Y_2$, for $Y_1, Y_2 > 0$, the minimum, $\psi_I^2(Y_1, Y_2) = \min\{Y_1, Y_2\}$, and the minus maximum, $\psi_I^2(Y_1, Y_2) = -\max\{Y_1, Y_2\}$ or generally a supermodular function

of two arguments.

For the minimization and maximization problems in (2.3a) and (2.3b), one obtains the following well-known necessary conditions under the above settings (Puccetti and Rüschendorf (2015) and Puccetti and Wang (2015)).

Proposition 2.1. *If* $(f_1(U), \dots, f_d(U))$*, where* f_j *is a rearrangement of* F_j^{-1}*,* $j = 1, \dots, d$*, solves problem* (2.3a) (*resp.,* (2.3b)) *and* ψ *is a supermodular function as specified in* (2.4) *and strictly increasing in each coordinate, then for any choice of subsets* $I \subset \{1, 2, \dots, d\}$*, it holds that* $\psi_I^r((f_j(U))_{j\in I})$ *and* $\psi_{I^c}^{d-r}((f_j(U))_{j\in I^c})$ *are antimonotonic* (*resp., comonotonic*)*.*

In the following subsection, we approximate the rearrangement problems (2.3) by discretizing the marginals F_j for $j = 1, \dots, d$.

2.2 Discretized problem

The next stage to solve (1.1) (or equivalently (2.3)) involves approximating the rearrangement problems in (2.3a) and (2.3b) through discretization of the marginals, F_j, $j = 1, \dots, d$.

Let $\boldsymbol{X} = (x_1, \dots, x_d)$ where $x_j = (x_{1j}, \dots, x_{nj})^T$ denotes the j-th column, $j = 1, \dots, d$, be a $n \times d$ matrix, and I be a subset of $\{1, \dots, d\}$. Here, x_j reflects the n possible mass points of F_j, assuming that all discretized marginal support sets of F_j have the same size, n. Then $\boldsymbol{X_1}$ (resp., $\boldsymbol{X_2}$) is a $n \times r$ (resp., $n \times (d-r)$) submatrix obtained from $\boldsymbol{X}$ by selecting its r (resp., $d-r$) columns with indices in the subset I (resp., I^c). By applying the function ψ (resp., ψ_I^r and $\psi_{I^c}^{d-r}$) to $\boldsymbol{X}$ (resp., $\boldsymbol{X_1}$ and $\boldsymbol{X_2}$), the n-dimensional vector $\Psi(\boldsymbol{X})$ (resp., $\Psi^I(\boldsymbol{X_1})$ and $\Psi^{I^c}(\boldsymbol{X_2})$) can be computed. Hence,

$$\Psi(\boldsymbol{X}) = \begin{bmatrix} \psi(x_{11}, \dots, x_{1d}) \\ \vdots \\ \psi(x_{i1}, \dots, x_{id}) \\ \vdots \\ \psi(x_{n1}, \dots, x_{nd}) \end{bmatrix}, \quad \Psi^I(\boldsymbol{X_1}) = \begin{bmatrix} \psi_I^r(x_{1j})_{j\in I} \\ \vdots \\ \psi_I^r(x_{ij})_{j\in I} \\ \vdots \\ \psi_I^r(x_{nj})_{j\in I} \end{bmatrix}, \quad \Psi^{I^c}(\boldsymbol{X_2}) = \begin{bmatrix} \psi_{I^c}^{d-r}(x_{1j})_{j\in I^c} \\ \vdots \\ \psi_{I^c}^{d-r}(x_{ij})_{j\in I^c} \\ \vdots \\ \psi_{I^c}^{d-r}(x_{nj})_{j\in I^c} \end{bmatrix},$$

where $\psi_I^r(x_{ij})_{j\in I}$ (resp., $\psi_{I^c}^{d-r}(x_{ij})_{j\in I^c}$) is obtained by applying the function ψ_I^r (resp., $\psi_{I^c}^{d-r}$) to the i-th rows of $\boldsymbol{X_1}$ (resp., $\boldsymbol{X_2}$), $i = 1, \dots, n$. Using (2.4), we have

$$\psi(x_{i1}, \dots, x_{id}) = \psi_I^2(\psi_I^r(x_{ij})_{j\in I}, \psi_{I^c}^{d-r}(x_{ij})_{j\in I^c}), \quad 1 \le i \le n.$$

We denote by $\mathcal{P}(\boldsymbol{X})$ the set of all $n \times d$ matrices obtained from $\boldsymbol{X}$ by swapping elements within each column, i.e.,

$$\mathcal{P}(\boldsymbol{X}) = \left\{ \tilde{\boldsymbol{X}} = (\tilde{x}_{ij}) : \tilde{x}_{ij} = x_{\pi_j(i),j}, \pi_1, \dots, \pi_d \text{ are permutations of } \{1, \dots, n\} \right\},$$

see similar definitions in Puccetti and Rüschendorf (2012, 2015) and Puccetti and Wang (2015). Next, we study the problems of how to rearrange the columns of $\boldsymbol{X}$ to obtain matrices that minimize and maximize $\frac{1}{n}\sum_{i=1}^{n}\psi(\tilde{x}_i)$ where $\tilde{x}_i = (\tilde{x}_{i1},\dots,\tilde{x}_{id})$ and $\tilde{\boldsymbol{X}} = (\tilde{x}_1,\dots,\tilde{x}_n)^T$, $i = 1,\dots,n$. That is, we consider the problems

$$m_\psi^n = \inf\left\{\frac{1}{n}\sum_{i=1}^{n}\psi(\tilde{x}_i);\ \tilde{x}_i = (\tilde{x}_{i1},\dots,\tilde{x}_{id}), \tilde{\boldsymbol{X}} = (\tilde{x}_1,\dots,\tilde{x}_n)^T \in \mathcal{P}(\boldsymbol{X})\right\}, \tag{2.5a}$$

$$M_\psi^n = \sup\left\{\frac{1}{n}\sum_{i=1}^{n}\psi(\tilde{x}_i);\ \tilde{x}_i = (\tilde{x}_{i1},\dots,\tilde{x}_{id}), \tilde{\boldsymbol{X}} = (\tilde{x}_1,\dots,\tilde{x}_n)^T \in \mathcal{P}(\boldsymbol{X})\right\}. \tag{2.5b}$$

We write $x_1 \downarrow x_2$ (resp., $x_1 \uparrow x_2$) to indicate that $x_1^{[i]} = F_1^{-1}(u_i)$ and $x_2^{[i]} = F_2^{-1}(1-u_i)$ (resp., $x_2^{[i]} = F_2^{-1}(u_i)$) for $i = 1,\dots,n$ where u_i is sampled from $U \sim U[0,1]$, and F_1 and F_2 are respective marginal distributions of x_1 and x_2. Let

$$\mathcal{O}_\psi(\boldsymbol{X}) = \left\{\boldsymbol{X}^* \in \mathcal{P}(\boldsymbol{X}) : \Psi^I(\boldsymbol{X}_{\boldsymbol{1}}^*) \downarrow \Psi^{I^c}(\boldsymbol{X}_{\boldsymbol{2}}^*) \text{ for any choice of the subsets } I\right\}$$

be the set of those permutation matrices $\boldsymbol{X}^*$ such that $\Psi^I(\boldsymbol{X}_{\boldsymbol{1}}^*)$ and $\Psi^{I^c}(\boldsymbol{X}_{\boldsymbol{2}}^*)$ are antimonotonic. Similarly, let

$$\mathcal{Q}_\psi(\boldsymbol{X}) = \left\{\boldsymbol{X}' \in \mathcal{P}(\boldsymbol{X}) : \Psi^I(\boldsymbol{X}_{\boldsymbol{1}}') \uparrow \Psi^{I^c}(\boldsymbol{X}_{\boldsymbol{2}}') \text{ for any choice of the subsets } I\right\}$$

be the set of those permutation matrices $\boldsymbol{X}'$ such that $\Psi^I(\boldsymbol{X}_{\boldsymbol{1}}')$ and $\Psi^{I^c}(\boldsymbol{X}_{\boldsymbol{2}}')$ are comonotonic. The sets $\mathcal{O}_\psi(\boldsymbol{X})$ and $\mathcal{Q}_\psi(\boldsymbol{X})$ are two classes of possible solutions for the problems (2.5a) and (2.5b), respectively; see also Proposition 2.1.

2.3 Block rearrangement algorithm

Motivated by Proposition 2.1, we can approximate the bounds m_ψ and M_ψ in (1.1) using BRA for determining m_ψ^n and M_ψ^n in (2.5).

In Figure 1, we show the generic flowchart of the BRA for the case of m_ψ^n. We make the following comments. The subset I is selected uniformly because the homogeneous risks are employed in this section. In the case of heterogeneous risks, assigning a higher probability to some columns—expected to matter more in the optimization—could make sense. In the case of M_ψ^n, we replace the text in the fourth rectangular zone and the rhombic decision zone by "Rearrange the rows of $\tilde{\boldsymbol{X}}_1$ so that $\Psi^I(\tilde{\boldsymbol{X}}_1) \uparrow \Psi^{I^c}(\tilde{\boldsymbol{X}}_2)$" and "Is $\tilde{\boldsymbol{X}} \in \mathcal{Q}_\psi(\boldsymbol{X})$?".

Remark 2.1. The BRA in Figure 1 produces only a local minimum of m_ψ in (1.1a) and a local maximum of M_ψ in (1.1b). The algorithm terminates once the matrix $\tilde{\boldsymbol{X}} \in \mathcal{O}_\psi(\boldsymbol{X})$ or $\tilde{\boldsymbol{X}} \in \mathcal{Q}_\psi(\boldsymbol{X})$ is found. However, such matrices may not be unique. Hence, the necessary condition in Proposition 2.1 is in general not sufficient to obtain solutions that are globally optimal.

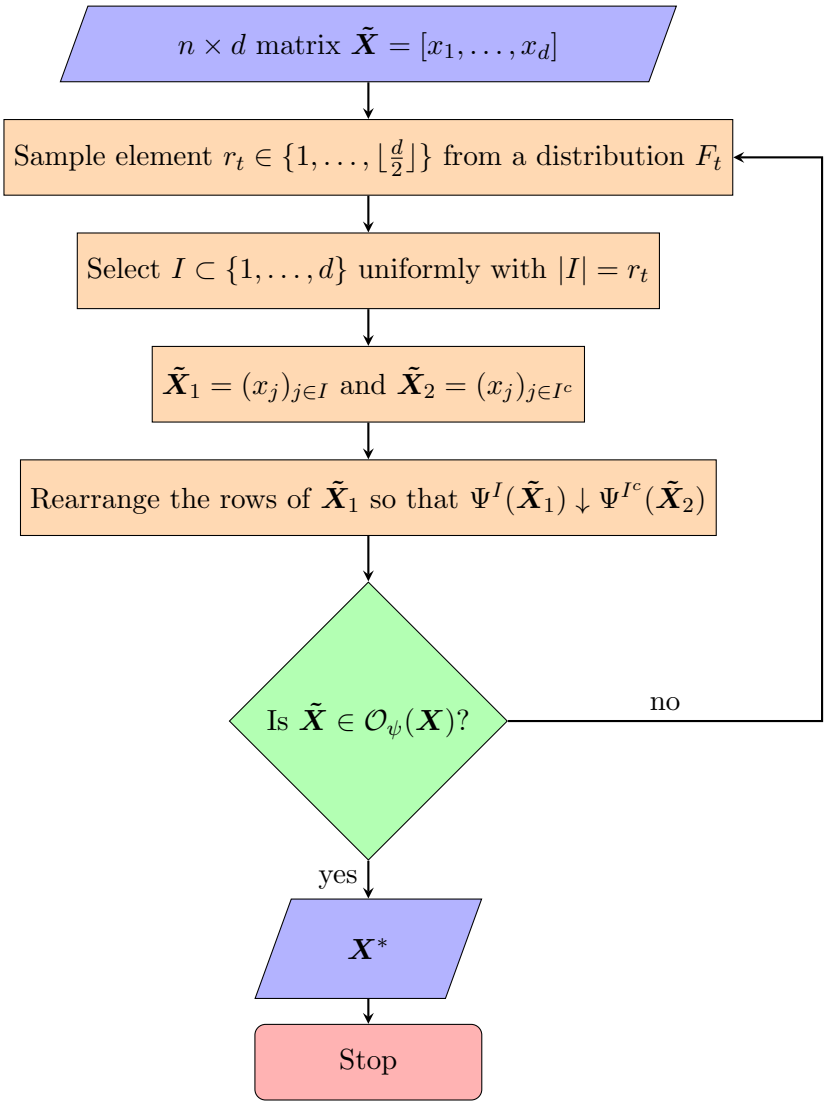

Figure 1: Flowchart for the BRA.

To answer the decision step that $\tilde{\boldsymbol{X}}$ belongs to the set $\mathcal{O}_\psi(\boldsymbol{X})$, we have to check the antimonotonicity condition for all subsets I, i.e., we need to consider $2^{d-1} - 1$ possibilities (Proposition 2.1). For a small number of columns (typically when d is less than 30), it is possible to check all possibilities. However, it is not possible in practice for large values of d. For instance, when $d = 30$, the computer needs to check more than one billion times the antimonotonicity condition for each single decision (step). Furthermore, finding an optimal solution $\boldsymbol{X}^*$ after one time rearrangement is almost impossible. Therefore, we put forward the following pseudo-code for BRA.

There are several versions of BRA, such as RA (F_t is degenerated with only one mass point with probability one: $\mathbb{P}(r_t = 1) = 1$) and BRA Binomial (F_t is $B(d, \frac{1}{2})$ distributed). We refer to Puccetti and Rüschendorf (2012) for the RA, and Bernard and McLeish (2016) and Bernard et al. (2018) for the BRA Binomial. Moreover, Boudt et al. (2018) introduce a version of the rearrangement algorithm named BRA with variance equalization (BRAVE).

BRA: Pseudo-code of the algorithm to minimize resp. maximize $\frac{1}{n}\sum_{i=1}^{n}\psi(\tilde{x}_i)$ over $\mathcal{P}(\boldsymbol{X})$. A BRA step means a single iteration within a loop from Step 4 to 8.

1 Initialize $n \times d$ matrix $\tilde{\boldsymbol{X}} = (x_1, \ldots, x_d)$ where $x_j = (x_{1j}, \ldots, x_{nj})^T$ denotes the j-th column ($j = 1, \ldots, d$);
2 Set maximum number of BRA steps T and $t = 0$;
3 **while** $t \leq T$ **do**
4 Sample element r_t from a distribution F_t with domain $\{1, 2, \ldots, \lfloor \frac{d}{2} \rfloor\}$;
5 Select uniformly a random subset $I \subset \{1, \ldots, d\}$ with $|I| = r_t$;
6 Separate two blocks (submatrices) $\tilde{\boldsymbol{X}}_1 = (x_j)_{j \in I}$ and $\tilde{\boldsymbol{X}}_2 = (x_j)_{j \in I^c}$ from $\tilde{\boldsymbol{X}}$;
7 Rearrange (swap) the rows of $\tilde{\boldsymbol{X}}_1$ so that the vector $\Psi^I(\tilde{\boldsymbol{X}}_1)$ is antimonotonic (resp., comonotonic) to $\Psi^{I^c}(\tilde{\boldsymbol{X}}_2)$ in the case of problem (2.5a) (resp., (2.5b));
8 Set $t = t + 1$;
9 **end**

In this case, they select I such that the variance of row sums across the two submatrices are as equal as possible, i.e., $I = \text{argmin}_I |\text{Cov}(\Psi^I(\tilde{\boldsymbol{X}}_1), \Psi(\tilde{\boldsymbol{X}})) - \text{Cov}(\Psi^{I^c}(\tilde{\boldsymbol{X}}_2), \Psi(\tilde{\boldsymbol{X}}))|$. In BRAVE, I is selected directly and not the result of first selecting r_t and then choosing uniformly the columns.

As a benchmark, we also study the case in which the cardinality r_t of the chosen subset I is uniform. Therefore, we propose to name this version of BRA, "BRA Unif", to indicate that each r_t is equally likely.

Definition 2.1 (BRA Unif)**.** A BRA is called BRA Unif if, in Step 4, each F_t is a discrete uniform distribution with support $\{1, 2, \ldots, \lfloor \frac{d}{2} \rfloor\}$.

We expect that the performance of BRA may be affected by three elements: (1) the cardinality of a subset I in each time step of the algorithm, i.e., the number r_t of columns or block size of $\tilde{\boldsymbol{X}}_1$; (2) how we initialize the matrix $\tilde{\boldsymbol{X}}$; (3) the maximum number of iterations T. T needs to be large enough such that when replaced by $T + k$, where $k = 50$, say, the change is "negligible". That is, there is no difference in $\frac{1}{n}\sum_{i=1}^{n}\psi(\tilde{x}_i)$ after T resp. $T + k$ steps. See also the relative stop criteria in the footnote of Algorithm 3.1 of Bernard et al. (2023).

In the remainder of the paper, we study in detail point (1) on the choice of the cardinality r_t of the chosen subset I in each step, which is determined by probability distributions F_t, $t = 0, 1, \ldots, T$, in Step 4. For the choice of the initial matrix $\tilde{\boldsymbol{X}}$ and the maximum number T of iterations, we give suggestions based on our ample numerical experience with the algorithm. The objective is to design an algorithm in such a way that it leads to a considerable improvement of block rearrangement algorithms.

3 Effect of block size

In this section, we first present some experiments to show the importance of choosing the cardinality r_t of the subset I in each t-th step of the algorithm.

3.1 Preliminary setting

We assume that $\psi(y_1, \ldots, y_d) = (\sum_{j=1}^{d} y_j)^2$. For this choice, equation (2.4) holds, that is $\sum_{j=1}^{d} y_j = \sum_{j \in I} y_j + \sum_{j \in I^c} y_j$. This assumption corresponds to the problem of minimizing the variance of a sum of random variables with fixed marginal distributions (see Embrechts and Puccetti (2010) and Wang and Wang (2011)). We omit the maximization problem M_ψ^n because, under this assumption, a unique solution is given by the comonotonic vector since ψ is a supermodular function. Therefore, the minimization problem is now recast as

$$m_\psi^n = \inf \left\{ \frac{1}{n} \sum_{i=1}^{n} \left(\sum_{j=1}^{d} \tilde{x}_{ij} - \frac{1}{n} \sum_{i=1}^{n} \left(\sum_{j=1}^{d} \tilde{x}_{ij} \right) \right)^2 ; (\tilde{x}_{ij}) \in \mathcal{P}(\boldsymbol{X}) \right\}. \tag{3.1}$$

To compare the performance of the algorithms in relation to the manner in which r_t, $t = 1, \ldots, T$, is chosen, we compute across k experiments the average log variance of the row sums instead of the variance because the variance decreases exponentially fast at the beginning for all versions of BRA (Boudt et al., 2018).

Definition 3.1 (Average Log Variance of Row Sums)**.** The average log variance of the row sums across k experiments after t BRA steps denoted by δ_t is defined as

$$\delta_t = \frac{1}{k} \sum_{l=1}^{k} \left\{ \log \left[\frac{1}{n} \sum_{i=1}^{n} \left(\sum_{j=1}^{d} x_{ij}^t - \frac{1}{n} \sum_{i=1}^{n} \left(\sum_{j=1}^{d} x_{ij}^t \right) \right)^2 \right] \right\}_l \tag{3.2}$$

where $t \in \{0, 1, \ldots, T\}$ and x_{ij}^t is the element of matrix $\tilde{\boldsymbol{X}}$ in i^{th} row and j^{th} column after t BRA steps.

Note that using the average log variance, we get a smoother and more robust experiment result from BRA. Moreover, δ_0 is defined as the log variance of the row sums of the initial input matrix $\tilde{\boldsymbol{X}}$.

Unless we specify it otherwise in this paper, the initial matrix $\tilde{\boldsymbol{X}}$ used in the algorithm is comonotonic. That is, the $\tilde{\boldsymbol{X}}$ consists of columns $x_j = (F_j^{-1}(u_1), F_j^{-1}(u_2), \ldots, F_j^{-1}(u_n))^T$, $j = 1, 2, \ldots, d$, where $u_i = \frac{i}{n+1}$ for $i = 1, 2, \ldots, n$. We use the comonotonic $\tilde{\boldsymbol{X}}$ because by starting with this initial matrix it is our experience that smaller δ_T values can typically be obtained than starting with other matrices. This procedure is verified after numerous

experiments for various initial matrices. For example in Figure 2, we compare the δ_T for this comonotonic choice with the δ_T one would obtain by starting with an "independent" $\tilde{\boldsymbol{X}}$, that is when the initial matrix consists of entries $x_{ij} = F_j^{-1}(u_{ij})$, $i = 1, 2, \ldots, n$ and $j = 1, 2, \ldots, d$, where u_{ij}, are $n \times d$ simulated values from independent standard uniform distributed random variables.

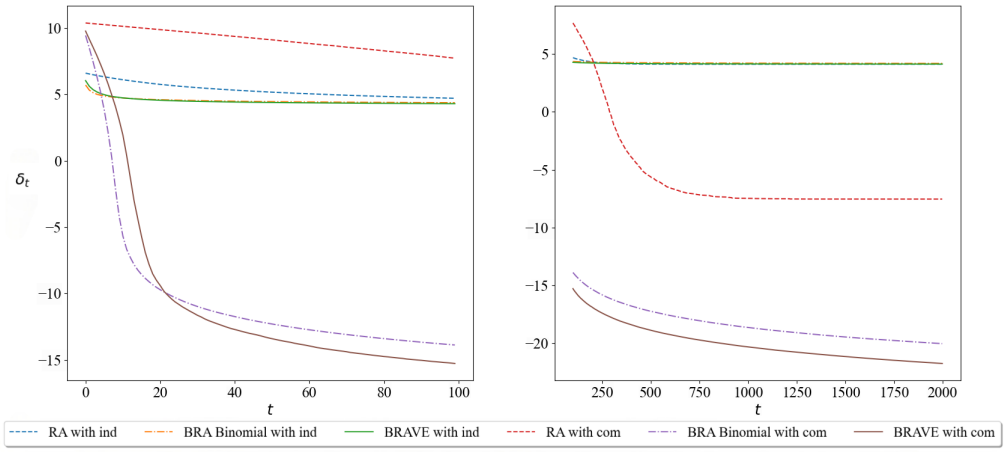

Figure 2: **Pareto Risks:** δ_t in (3.2) with $k = 100$, $T = 2000$, $n = 1000$ and $d = 100$. The left (resp., right) graph shows δ_t for $t \leq 100$ (resp., $t > 100$). The dashed blue, dash-dotted orange and solid green lines represent the evolution of δ_t by implementing the respective RA, BRA Binomial and BRA Unif, starting from the independent $\tilde{\boldsymbol{X}}$, while the corresponding red, purple and brown lines show the evolution of δ_t, starting from the comonotonic $\tilde{\boldsymbol{X}}$. The tail indexes of the Pareto distributions are 2.

In the following subsection, we run experiments to show the importance of appropriately choosing r_t, $t = 1, \ldots, T$.

3.2 Cardinality of I in BRA

The BRA Binomial from Bernard and McLeish (2016) selects in each t-th step I uniformly out of $2^{d-1} - 1$ possible pairs of subsets $I, I^c \subset \{1, 2, \ldots, d\}$, where $|I| \leq |I^c|$. Thus, in this situation, the value for r_t is not uniformly sampled in $\{1, 2, \ldots, \lfloor \frac{d}{2} \rfloor\}$. For instance, $\mathbb{P}(r_t = 1) = \frac{d}{2^{d-1}-1}$ since there are only d sets with the cardinality equal to 1. In fact, out of the $2^{d-1} - 1$ possible pairs of subsets I and I^c, there are many more I such that r_t is close to $\lfloor \frac{d}{2} \rfloor$ than that $r_t = 1$ or 2. Thus, in the BRA Binomial, many steps will be done with blocks of similar size rather than with a tiny block and one big block (as in the RA). The subset I always has only one element in the case of RA, $\mathbb{P}(r_t = 1) = 1$. That is, $\tilde{\boldsymbol{X}}_1$ (resp., $\tilde{\boldsymbol{X}}_2$) is a $n \times 1$ (resp., $n \times (d-1)$) matrix. However, in the case of BRA Unif, $\mathbb{P}(r_t = i) = \frac{1}{\lfloor \frac{d}{2} \rfloor}$

for all $i \in \{1, 2, ..., \lfloor \frac{d}{2} \rfloor\}$.

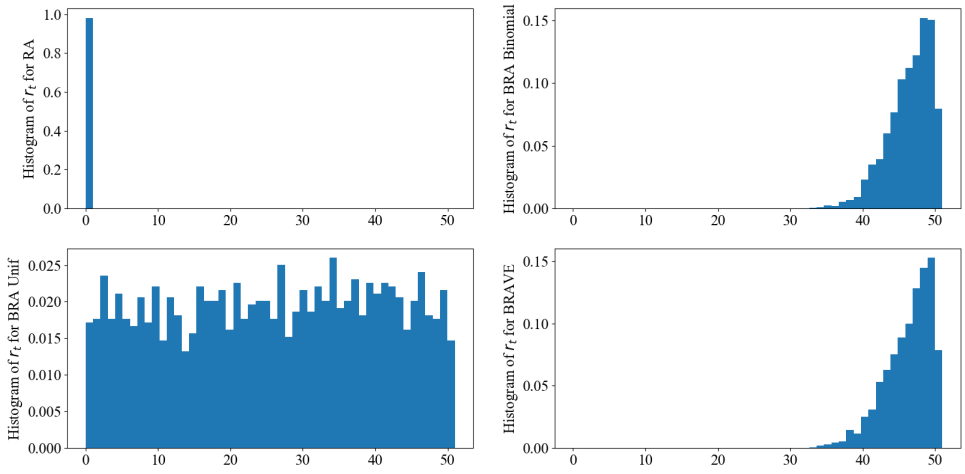

Figure 3: The histograms of the cardinality r_t of the subset I when implementing four types of BRA: RA, BRA Binomial, BRA Unif and BRAVE with $T = 2000$, $n = 1000$ and $d = 100$.

Figure 3 presents the histograms of the cardinality r_t, $t = 1, \ldots, T$ when implementing four types of BRA: RA, BRA Binomial, BRA Unif and BRAVE. We set $T = 2000$, $n = 1000$ (the number of rows) and $d = 100$ (the number of columns) in the BRA. These four plots verify our previous comments on the choice of r_t. In particular, the histograms of the BRA Binomial and the BRAVE are similar. Thus, we expect the effects of variance reduction for these two types of BRA to be similar.

To examine the impact of block size r_t, we analyze the δ_t plots obtained from the implementation of three distinct BRAs: RA, BRA Binomial and BRA Unif. These algorithms are applied to an initial matrix $\tilde{X}$, which is sampled from uniform or Pareto distributions with tail indexes of 2. The same choices have been made in Bernard and McLeish (2016). Our study focuses on understanding the characteristics of these BRA types. We carry out experiments by running the BRA with $d \in \{100, 500\}$ (number of columns), $n = 1000$ (number of rows) and $k = 100$ (number of repeating experiments), with each experiment consisting of 2000 BRA steps ($T = 2000$).

Numerical results (δ_t) are displayed in Figures 4 and 5. On the x-axis, we display the BRA steps; on the y-axis, we represent the magnitude of the average log variance of the row sums (as in Definition 3.1). By analyzing Figure 4, which depicts the δ_t with uniformly distributed variables, and Figure 5, which shows the δ_t with Pareto distributed variables, we can observe that the BRA Binomial and BRA Unif lead to a higher variance reduction in the initial steps. However, the RA exhibits a much higher variance reduction after approximately 25 steps for uniformly distributed variables and 100 steps for Pareto distributed variables. The BRA Unif method outperforms the BRA Binomial method in all

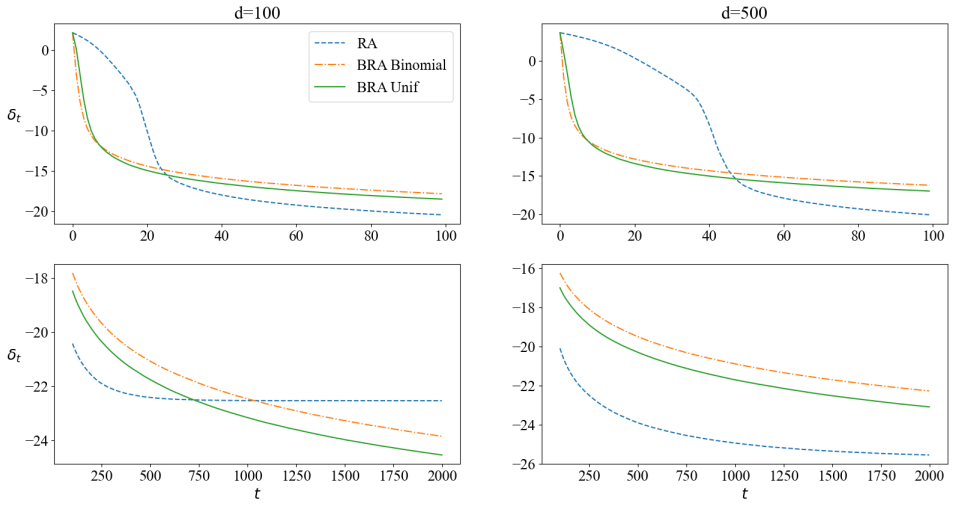

Figure 4: **Uniform Risks:** δ_t in (3.2) with $k = 100$, $T = 2000$ and $n = 1000$. The left (resp., right) graphs show plots when $d = 100$ (resp., $d = 500$). The top figures display the δ_t during the first 100 steps, while the bottom display the δ_t after 100 steps. The dashed blue, dash-dotted orange and solid green lines represent the evolution of δ_t by implementing the respective RA, BRA Binomial and BRA Unif.

tested scenarios, indicating that selecting I uniformly out of the $2^{d-1} - 1$ possible subsets as done in the BRA Binomial is not the optimal choice for selecting r_t, $t = 1, \ldots, T$. We also observe that RA is sometimes the best of all algorithms discussed so far.

Hence, our observations suggest the need for a BRA design that further improves BRA Unif and RA. That is, we propose a BRA where the first steps are performed with larger block sizes while the last steps are performed with smaller ones. This approach would result in an algorithm that behaves similarly to the BRA Binomial at the beginning, and more like the RA towards the end, achieving better overall performance. We will design a BRA taking these features into account in the next section.

4 Improved BRA

In this section, we propose a better method to select the r_t when running the BRA to get the best convergence and accuracy.

4.1 BRA Beta

Let $\{r_1, \ldots, r_T\}$ be a sequence of the cardinality of the subset I. That is, r_t is the number of columns of $\tilde{\boldsymbol{X}}_1$ for the t^{th} BRA step. Note that $t \in \{1, \ldots, T\}$, and T is the maximum

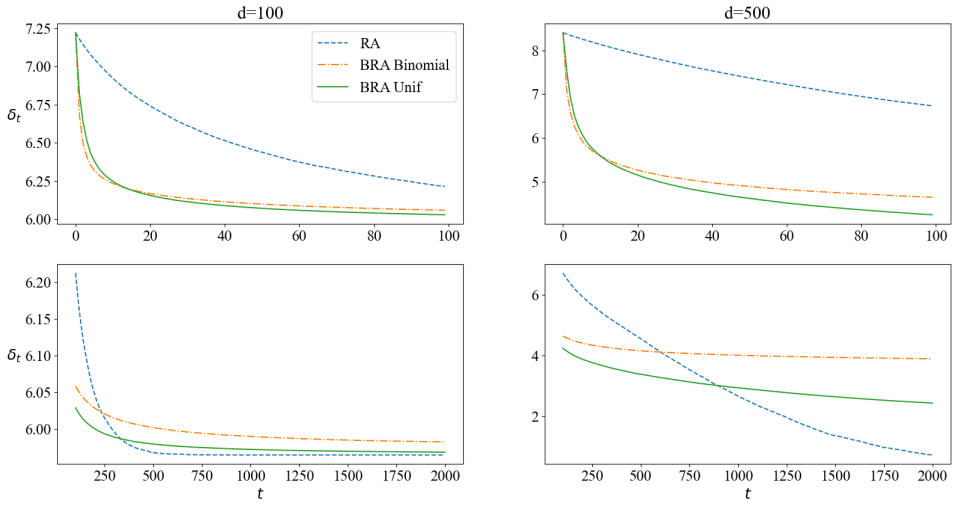

Figure 5: **Pareto Risks:** δ_t in (3.2) with $k = 100$, $T = 2000$ and $n = 1000$. The left (resp., right) graphs show plots when $d = 100$ (resp., $d = 500$). The top figures display the δ_t during the first 100 steps, while the bottom display after 100 steps. The dashed blue, dash-dotted orange and solid green lines represent the evolution of δ_t by implementing the respective RA, BRA Binomial and BRA Unif. The tail indexes of the Pareto distributions are 2.

number of BRA steps in the BRA. In particular, r_t is sampled from a random variable R_t, which is equal to one plus the floor of the product of $\frac{d}{2}$ and a Beta distributed random variable B_t with parameters α_t and β_t. The discrete distribution is then scaled to have support in $\{1, 2, \ldots, \lfloor \frac{d}{2} \rfloor + 1\}$. That is,

$$R_t = \left\lfloor \frac{d}{2} B_t \right\rfloor + 1,$$

where $B_t \sim \text{Beta}(\alpha_t, \beta_t)$. Therefore, the probability mass function of R_t under the setting is

$$\begin{aligned}\mathbb{P}(R_t = r_t) &= \mathbb{P}\left(\left\lfloor \frac{d}{2} B_t \right\rfloor + 1 = r_t\right) = \mathbb{P}\left(r_t - 1 \leq \frac{d}{2} B_t < r_t\right) \\ &= \mathbb{P}\left(\frac{2(r_t - 1)}{d} \leq B_t < \frac{2r_t}{d}\right) = \frac{\Gamma(\alpha_t + \beta_t) \int_a^b t^{\alpha_t - 1}(1-t)^{\beta_t - 1} dt}{\Gamma(\alpha_t)\Gamma(\beta_t)},\end{aligned}$$

where $a = \frac{2(r_t - 1)}{d}$, $b = \frac{2r_t}{d}$, and $\Gamma(x)$ is the gamma function. Thus, we can control r_t by appropriately choosing α_t and β_t. To do so, we use the following parametrization of the

Beta distribution with two extra parameters, A and B,

$$\begin{aligned}\alpha_t &= A - \left(\frac{t-1}{T-1}\right)^{\frac{1}{B}}(A-1),\\ \beta_t &= 1 + \left(\frac{t-1}{T-1}\right)^{\frac{1}{B}}(A-1).\end{aligned} \tag{4.1}$$

Using (4.1) and proper parameters A and B, we can design a BRA that behaves similarly to the BRA Binomial at the initial steps and more like the RA towards the end. Note that the mean of a beta distribution with parameters α_t and β_t is given by $\frac{\alpha_t}{\alpha_t+\beta_t}$. When $A = B = 1$, the Beta distribution is uniform; thus r_t is equally likely among $\{1, 2, \ldots, \lfloor \frac{d}{2} \rfloor + 1\}$. That

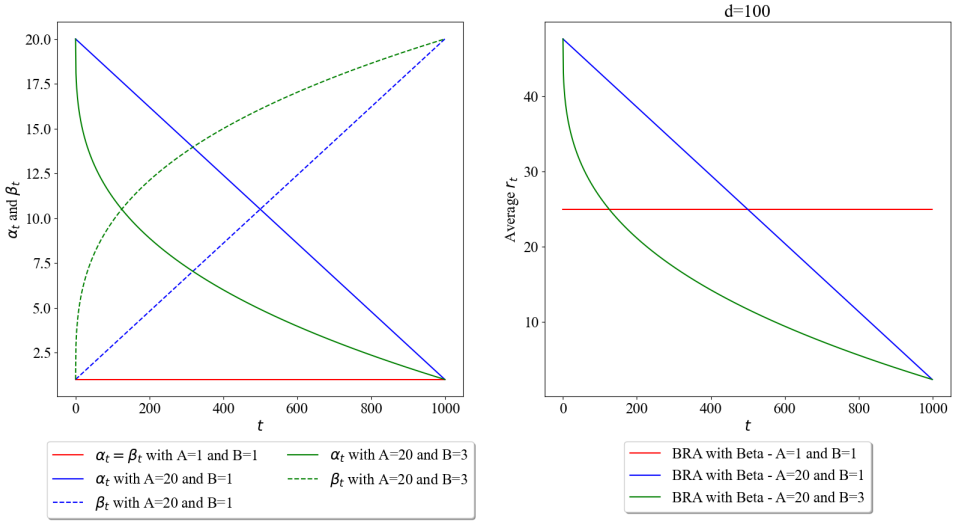

Figure 6: Parameters of the Beta distribution in terms of t. The left panel shows the parameters α_t and β_t for $t \in \{1, 2, \ldots, 1000\}$ and some examples of A and B, while the right shows the average r_t from the corresponding Beta distribution.

is, we obtain the BRA Unif. When $A \neq 1$ and $B = 1$, we have a linear decrease in the average block size used in the BRA over time, as displayed in the right panel of Figure 6. Furthermore, when $A \neq 1$ and $B > 1$, we have a convex curve decrease of average r_t. The parameters A and B control the switching speed of the algorithm from behaving BRA Binomial to RA. The choice of α_t and β_t over time t implies that the BRA Beta switches from the behavior of BRA Binomial at $t = 1$ to the behavior of RA at $t = T$. As an indication, note that for $t = 1$, $\mathbb{E}[R_1] = \sum_{r_1=1}^{\lfloor \frac{d}{2} \rfloor + 1} r_1 \mathbb{P}(R_1 = r_1) \approx \lfloor \frac{d}{2} \rfloor + 1$ (BRA Binomial), while for $t = T$, $\mathbb{E}[R_T] = \sum_{r_T=1}^{\lfloor \frac{d}{2} \rfloor + 1} r_T \mathbb{P}(R_T = r_T) \approx 1$ (RA). In the left panel of Figure 6, we also illustrate the choices of parameters α_t and β_t of the Beta distribution as a function of t for various values of A and B.

Definition 4.1 (BRA Beta)**.** A BRA is called BRA Beta if, in Step 4, F_t is the distribution where a random variable, $X_t \sim F_t$, takes integer parts of numbers sampled from Beta(α_t, β_t).

The parameters α_t and β_t are specified in (4.1).

In the subsequent Figures 7 and 8, we run BRA Beta with some choices of A and B to illustrate the effect of choosing r_t appropriately. Figure 7 shows the evolution of δ_t starting from a $\tilde{\boldsymbol{X}}$ with standard uniform marginals, while Figure 8 illustrates that with Pareto marginals, the tail index is two. Note that $A = B = 1$ corresponds to the uniform distribution of the block sizes and is reported as "BRA Unif" in Figures 4 to 5. Figures 7 and 8 provide evidence that the choice of r_t in BRA plays an important role, especially when one wants to achieve quick convergence. We point out that running these calculations requires only a few minutes on a standard laptop.

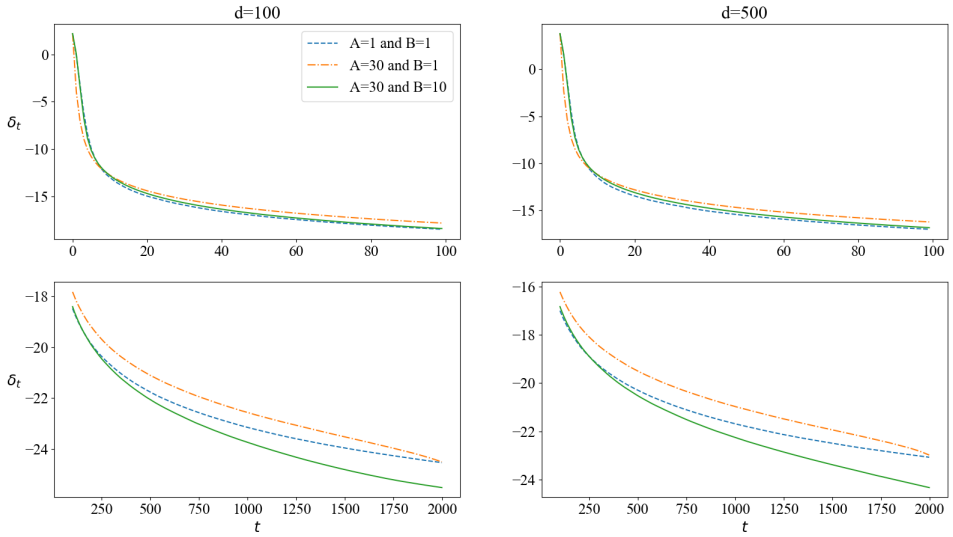

Figure 7: **Uniform risks:** δ_t in (3.2) with $k = 100$, $T = 2000$ and $n = 1000$. The marginal distributions are standard uniform. The left (resp., right) graphs show plots when $d = 100$ (resp., $d = 500$). The top figures display the δ_t during the first 100 steps, while the bottom display after 100 steps. The dashed blue (resp., dash-dotted orange, and solid green) lines illustrate δ_t when implementing the BRA Beta with $A = 1$ and $B = 1$ (resp., $A = 30$ and $B = 1$, and $A = 30$ and $B = 10$).

4.2 Best choices of A and B

To help us analyze the results of BRA Beta, we use a new data visualization method–grid heatmap in the following. It displays magnitude (δ_T in our case) as colour in a two-dimensional matrix, with one dimension representing A and another representing B. Smaller δ_T are represented by deeper or darker colour squares and larger δ_T by lighter colour squares. See Wilkinson and Friendly (2009) for the definition and history of the heatmap.

In Figure 9, the left (resp., the right) figures show the heatmaps of δ_T with $k = 100$,

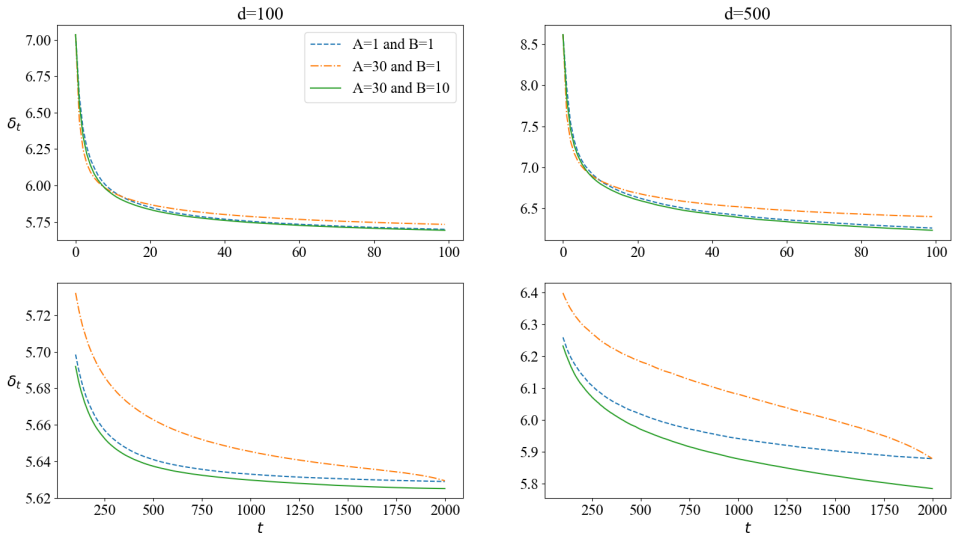

Figure 8: **Pareto risks:** δ_t in (3.2) with $k = 100$, $T = 2000$, and $n = 1000$. The left (resp., right) graphs show plots when $d = 100$ (resp., $d = 500$). The top figures display the δ_t during the first 100 steps, while the bottom display after 100 steps. The dashed blue (resp., dash-dotted orange, and solid green) lines illustrate δ_t when implementing the BRA Beta with $A = 1$ and $B = 1$ (resp., $A = 30$ and $B = 1$, and $A = 30$ and $B = 10$). The tail indexes of the Pareto distributions are 2.

$T = 2000$, $n = 1000$ and $d = 100$ (resp., $d = 500$). The initial matrix is the $\tilde{\boldsymbol{X}}$ with standard uniform marginals. A and B vary in the range of $[1, 10]$ with interval 1 in the top graphs of Figure 9, while they vary in the range of $[10, 300]$ with interval 10 in the left bottom and $[10, 630]$ with interval 20 in the right bottom. The first-row plots of Figure 9 illustrate that the bigger A and B, the smaller δ_T. The exact property holds for Pareto distributions with tail index two, as shown in the first row of Figure 10. When $A = B = 10$ in the first rows of Figures 9 and 10, δ_T are smallest. The second rows of these figures illustrates that the smaller δ_T are distributed in the plots as a rotated 90 degrees "L" shape in both heatmaps. Moreover, when $A = B = 10$ in the second-rows of Figures 9 and 10, δ_T are not in the rotated "L" shape. δ_T with A and B in the rotated "L" shape are smaller than δ_T with $A = 10$ and $B = 10$. Therefore, the optimal parameters of A and B are not in $[1, 10]$. The rotated "L" shape moves down left in terms of the number of risks d.

To further study the effect of parameters A and B on accuracy, we plot δ_T with $k = 100$, $T = 2000$ and $n = 1000$ on a 3D surface. Similar to the setting of Figures 9 and 10, Figures 11 and 12 illustrate δ_T respectively for uniform and Pareto risks when implementing BRA Beta with various parameters A and B. Specifically, we focus on $A, B \in [10, 300]$ and $A, B \in [10, 630]$ with intervals 10 and 20, respectively. It is evident that the optimal A and B are not located in the range of $[1, 10]$; see Figures 9 and 10. To investigate the best choices of A and B in terms of accuracy, the contour plots in the bottom face of the 3D

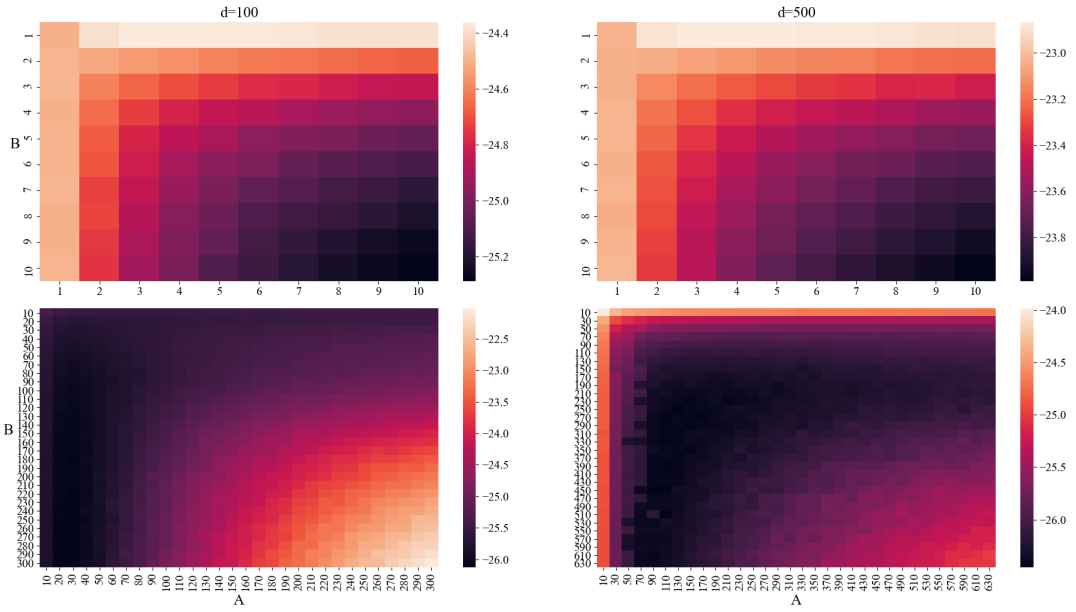

Figure 9: **Uniform risks:** The heatmaps of δ_T with $k = 100$, $T = 2000$ and $n = 1000$ when implementing the BRA Beta. The left (resp., right) graphs show heatmaps when $d = 100$ (resp., $d = 500$) variables. The top (resp., left and right bottom) figures present the case of A and B in the range of $[1, 10]$ (resp., $[10, 300]$ and $[10, 630]$).

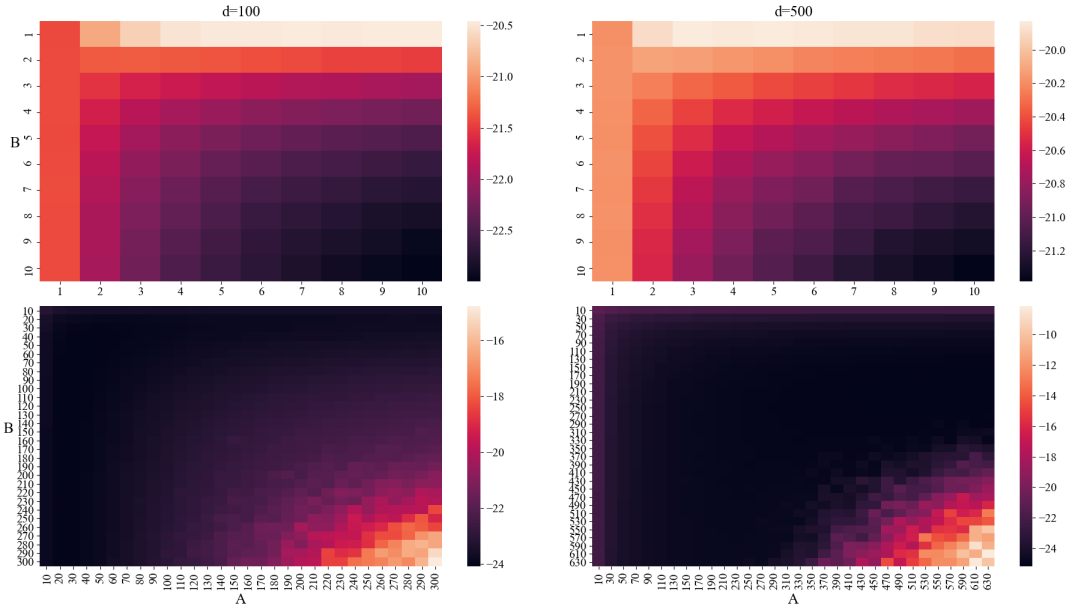

Figure 10: **Pareto risks:** The heatmaps of δ_T with $k = 100$, $T = 2000$ and $n = 1000$ when implementing the BRA Beta. The left (resp., right) graphs show heatmaps when $d = 100$ (resp., $d = 500$). The top (resp., left and right bottom) figures present the case of A and B in the range of $[1, 10]$ (resp., $[10, 300]$ and $[10, 630]$). The tail indexes of the Pareto distributions are 2.

plots are critical. The colder colour line of the contour represents the better δ_T. The areas circled by deep blue lines, which move in terms of d and marginal distributions, are the

potential locations for optimal A and B. These two figures narrow down our search area compared to the heatmaps.

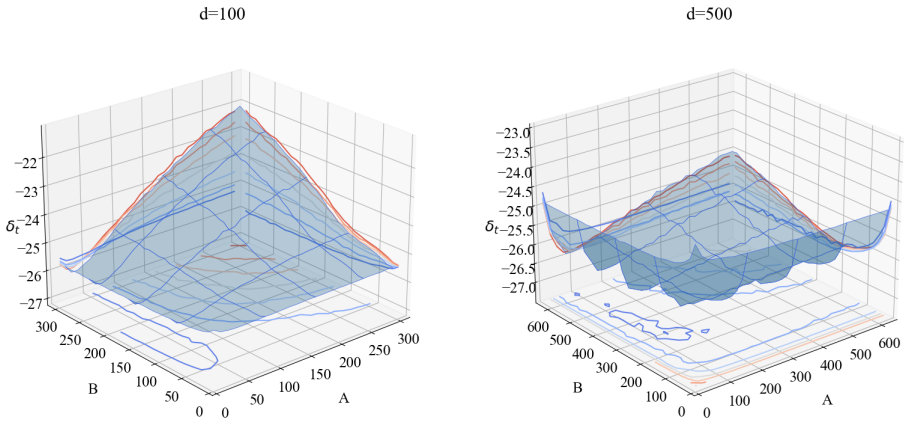

Figure 11: **Uniform risks:** The 3D surface plots of δ_T with $k = 100$, $T = 2000$ and $n = 1000$ when implementing the BRA Beta. The left (resp., right) graph, $d = 100$ (resp., $d = 500$), shows a plot in the case of A and B in the range of $[10, 300]$ (resp., $[10, 630]$).

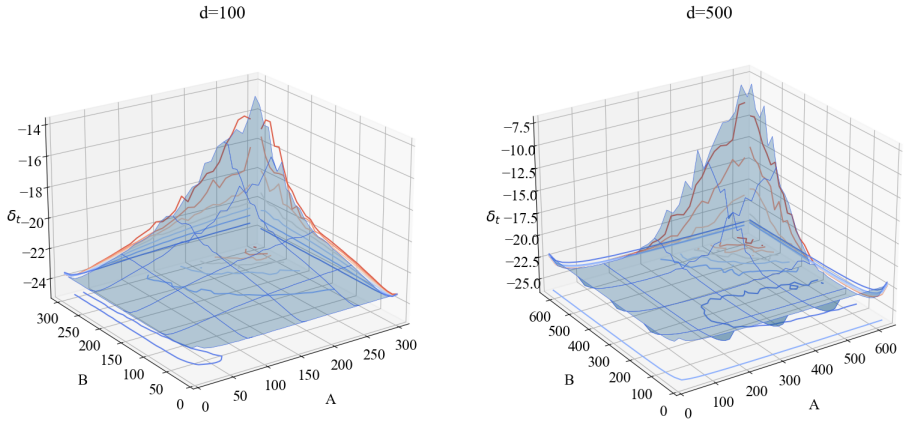

Figure 12: **Pareto risks:** The 3D surface plots of δ_T with $k = 100$, $T = 2000$ and $n = 1000$ when implementing the BRA Beta. The left (resp., right) graph, $d = 100$ (resp., $d = 500$), shows a plot in the case of A and B in the range of $[10, 300]$ (resp., $[10, 630]$). The tail indexes of the Pareto distributions are 2.

We set 0.3 and 0.5 as the ratios of A and d and B and d, i.e., $A = 0.3d$ and $B = 0.5d$. For example, we set $A = 30$ and $B = 50$ if $d = 100$ while $A = 300$ and $B = 500$ if $d = 1000$.

To save space, we only report four cases of combination of two marginal distributions, i.e., the uniform and Pareto with tail index two, and two dimensions $d \in \{100, 500\}$ in Figures 9–12. However, one needs a mass of data to find the best choices of A and B ($A = 0.3d$ and $B = 0.5d$). For instance, combining different $T \in \{500, 1000, 2000\}$ and $d \in \{100, 500, 1000\}$ gives us nine cases for one marginal distribution. Note that the choices of A and B presented here are best suited for various cases, including those with different sample sizes n, dimensions d, marginal distributions, and relatively small numbers of iterations T. However, if we specify these parameters, alternative values of A and B may potentially achieve lower values for the δ_T. Finding such values can be computationally expensive and time-consuming. Furthermore, if we set a sufficiently large number of iterations T, the performance of BRA Beta is comparable to other variants of the BRA. Therefore, the choices of A and B suggested here provide a good trade-off between performance and computational efficiency.

Figure 13 presents the probability density function of r for the BRA Beta algorithm with $A = 30$ and $B = 50$, considering $T = 2000$, $n = 1000$, and $d = 100$. The plot illustrates the low probability of obtaining large r_t values, i.e. $r_t \approx 50$, and the high likelihood of obtaining small r_t values, i.e. $r_t \in [1, 10]$. This time-dependent behaviour of r_t allows BRA Beta to perform similarly to the BRA Binomial at the beginning and the RA towards the end, as anticipated in the previous section.

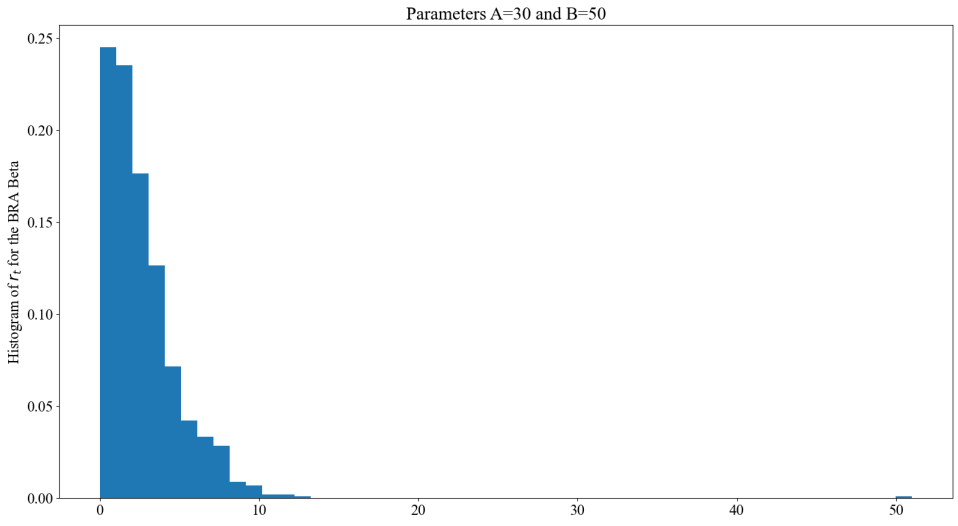

Figure 13: Histogram of the cardinality r_t of the subset I when implementing BRA Beta, $A = 30$ and $B = 50$, with $T = 2000$, $n = 1000$ and $d = 100$.

In summary, there is one recommendation for better BRA performance in variance reduction. We suggest using the BRA Beta with $A = 0.3d$ and $B = 0.5d$ to approximate sharp bounds in (1.1a) and (1.1b).

5 Performance evaluation of variance reduction

In the previous sections, we studied four versions of BRA: RA, BRA Binomial, BRAVE and BRA Beta. In this section, we compare their performances in accuracy, convergence speed and efficiency when minimizing the variance of row sums.

Boudt et al. (2018) propose the BRAVE in which, for each BRA step, selecting the number r_t in terms of the variance equalization, i.e., $\mathrm{Var}(\sum_{j\in I} X_j) = \mathrm{Var}(\sum_{j\in I^c} X_j)$. They conclude that for number partitioning problems, the BRAVE outperforms RA, BRA Binomial and well-known existing number partitioning algorithms such as the greedy algorithm. Specifically, we use the greedy algorithm to find the subset I for BRAVE, i.e., BRAVE(greedy)[1]. See Korf (1998) for an introduction to the greedy algorithm. We propose the BRA Beta approach in the previous section to solve the same problem and recommend parameters $A = 0.3d$ and $B = 0.5d$. We vary the choice of n and d to check the performance of these algorithms, specifically for $n \in \{10, 100, 1000\}$ and $d \in \{50, 100, 250\}$. We consider, $A \in \{15, 30, 75\}$ and $B \in \{25, 50, 125\}$ concerning the size r_t in the BRA Beta. We repeat the experiments in Section 5 of Boudt et al. (2018) but replace the BRAVE+RA with the BRA Beta. The basic reason for this choice is that if we assess the distribution of BRA Beta in Figure 13 and its block size changing pattern, BRA Beta with $A = 0.3d$ and $B = 0.5d$ performs as BRA at the beginning then as RA until the end. Hence, it bears similarity with BRAVE+RA in this regard. The advantage of the BRA Beta is that it locates the turning point in switching from BRA to RA automatically instead of manually (ad-hoc choice).

5.1 Accuracy

In this subsection, we use numerical experiments to test the accuracy of our proposed BRA Beta. The accuracy is assessed by measuring the average log variance of row sums after t steps, i.e., by monitoring δ_t.

In Figures 14 and 15, we plot the trajectories of δ_t in (3.2) in terms of various n and d as a function of t. The δ_t is computed by implementing RA, BRA Binomial, BRAVE(greedy) and BRA Beta. In particular, we set the number of repeating experiments $k = 100$ and the maximum number of iterations $T = 2000$. The solid blue lines show the changes in the δ_t of RA, while the dotted orange lines plot that of BRA Binomial. Furthermore, the dashed green and the dash-dotted red lines correspond to the trajectories of BRAVE(greedy) and BRA Beta, respectively.

Figure 14 compares the effectiveness of RA, BRA Binomial, BRAVE and BRA Beta for minimizing the variance of the row sums. We implement those algorithms by initializing them with a $\tilde{\boldsymbol{X}}$ sampled from standard uniform marginal distributions. The variance of

[1]We use this type of BRAVE because it is less time-consuming than another BRAVE called BRAVE(KK). See Boudt et al. (2018) for details.

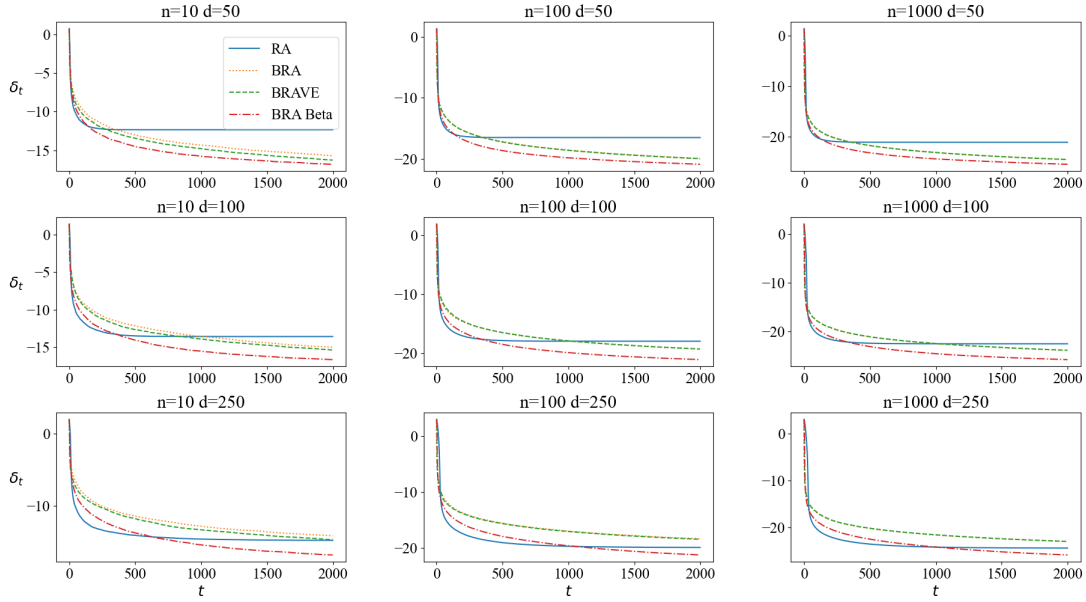

Figure 14: **Uniform risks:** The performance of four BRA-based algorithms on accuracy δ_T with $k = 100$ and $T = 2000$ for different n and d.

the row sums decreases sharply in the first few steps for all algorithms, regardless of the numbers of n and d. BRA Beta shows the highest effect of the variance reduction under these configurations. With $T = 2000$, the effectiveness of improving the objective function for all types of BRA decreases when d increases.

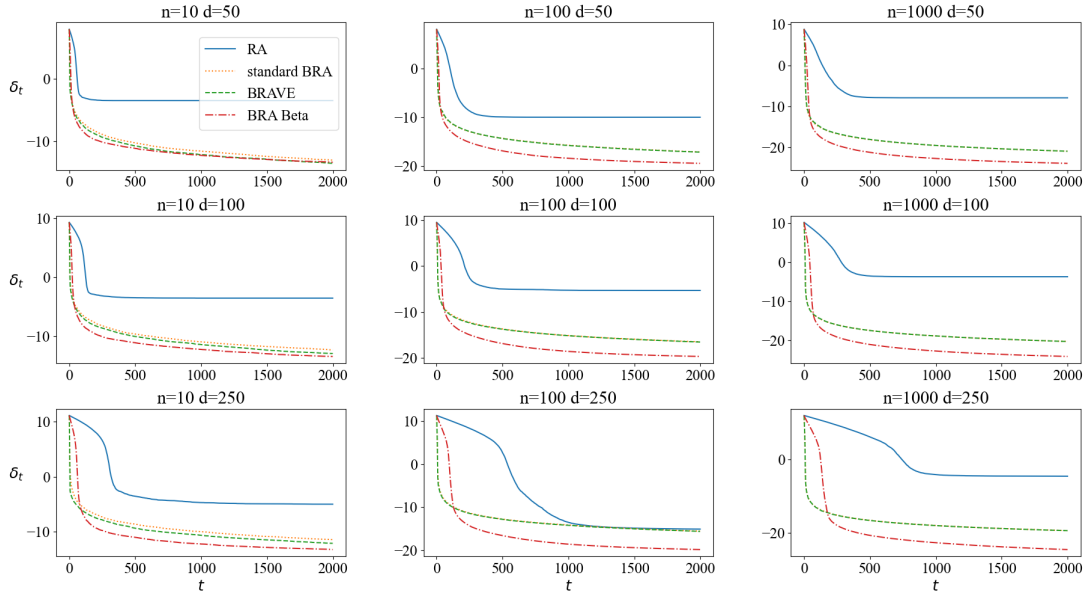

Figure 15: **Homogeneous Pareto risks:** The performance of four BRA-based algorithms on accuracy δ_T with $k = 100$ and $T = 2000$ for different n and d. The tail indexes of the Pareto distributions are 2.

Figure 15 displays line plots with the same parameter settings as those in Figure 14, but

the initial matrix $\tilde{\boldsymbol{X}}$ is sampled from Pareto distributions with a tail index value of 2. In contrast to the scenarios with uniform risks, the figure indicates a consistent performance of the rearrangement algorithms on improving the objective function. The effectiveness of BRA Beta is still the highest after 2000 BRA steps. Figures in the last column of Figure 15 illustrate that the accuracy improvement of BRA Beta is significant compared with other BRA-based algorithms when the discretization level is relatively high. Note that the parameter n has more effect on the magnitude of δ_t but less in improving the variance of the row sum.

To further show that the BRA Beta has a dominant performance in accuracy, we use the standard lognormal and Pareto distributions with different tail index to simulate the input data; see Figures 16 and 17. Except for sampling data from two extremal distributions (Figures 14 and 15) indicating that the BRA Beta works well, Figure 16 stands for the intermediate case. That is, Uniform and Pareto are respective symmetric and heavy-tailed distributions, while the standard lognormal has a considerable skewness of about 6.18. Moreover, Figure 17 presents the trajectories of δ_t with $k = 100$ and $T = 2000$ when the data are sampled from the heterogeneous Pareto risks X_j with tail indexes $1.5 + \frac{j-1}{d-1}$ for $j = 1, 2, \ldots, d$.

The numerical experiments in Figures 14–17 reveal that BRA Beta, using parameters $A = 0.3d$ and $B = 0.5d$, outperforms RA, BRA Binomial and BRAVE(greedy) when we use different distributions to sample the initial matrix. Although the δ_t values produced by BRA Beta for $t = 1, \ldots, T$ are not consistently lower than those from other algorithms, δ_T are consistently the smallest across various n and d values. This aligns with our initial purpose, which is to design the BRA Beta to obtain the lowest δ_T. The performance of BRA Beta is consistent even if we change the distribution and dependence structure for the initial matrix. The effects of BRA Binomial and BRAVE(greedy) meet the expectations outlined in Section 3.2. Such a result further confirms that the distribution of the cardinality r_t of subset I influences the effectiveness of BRA. It is evident that the improvement of the BRA Beta in accuracy is double that of the other algorithms when the risks are heterogeneous and the portfolio is large; see, e.g., $n = 1000$ and $d = 250$ in Figure 17.

5.2 Convergence speed

In this subsection, we test the convergence speed of the BRA Beta with $A = 0.3d$ and $B = 0.5d$. The number of BRA steps T measures the convergence speed. Note that we need to compare the BRA-based algorithms for different T because the series of the block size r_t generated by the BRA Beta is related to T.

Figure 18 illustrates the performance of four BRA-based algorithms in the convergence speed when $k = 100$, $n = 1000$ and $d = 100$. The initial matrix $\tilde{\boldsymbol{X}}$ is sampled from a standard lognormal distribution. Nine different T in the range of $[300, 1900]$ with the

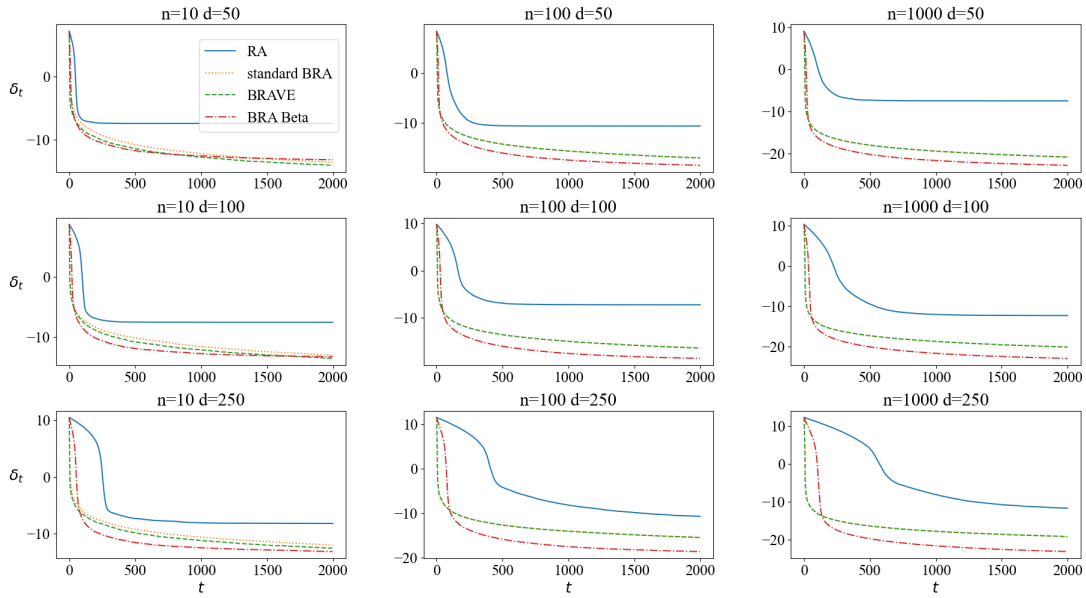

Figure 16: **Standard lognormal risks:** The performance of four BRA-based algorithms on accuracy δ_T with $k = 100$ and $T = 2000$ for different n and d.

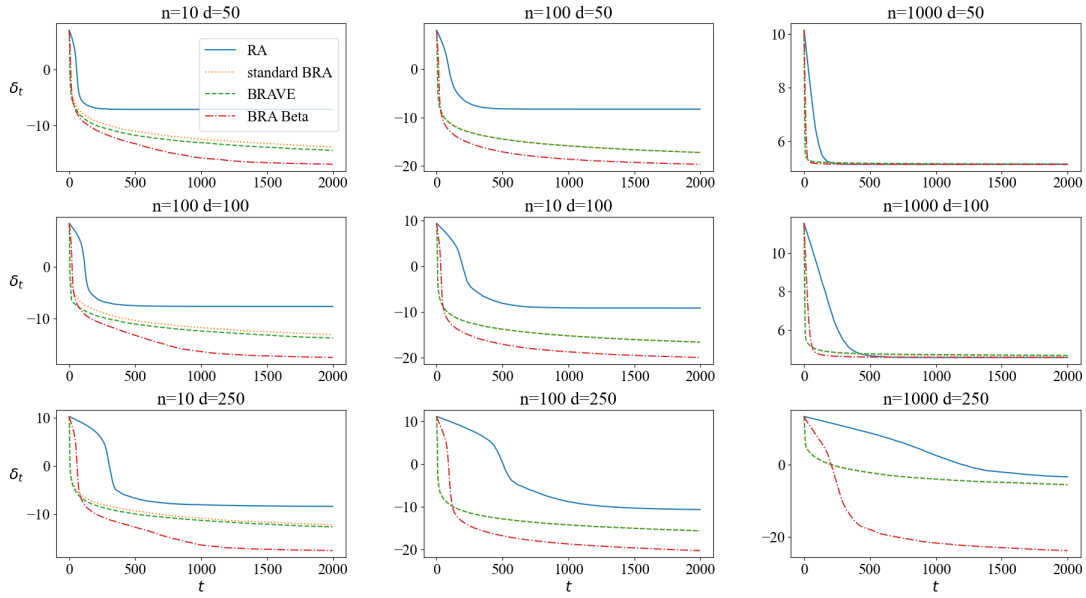

Figure 17: **Heterogeneous Pareto risks:** The performance of four BRA-based algorithms on accuracy δ_T with $k = 100$ and $T = 2000$ for different n and d. The tail indexes of the Pareto distributions of random variable X_j are $1.5 + \frac{j-1}{d-1}$ for $j = 1, 2, \ldots, d$.

interval 200 are selected. Under such a setting, RA has the worst performance, while BRA Binomial and BRAVE are comparable. It is clear that the BRA Beta has the best performance in the speed of variance reduction for all nine cases. It is true that even we use different distributions, such as uniform and Pareto, to sample the input matrix $\tilde{\boldsymbol{X}}$. We have also done experiments for various types of distributions. However, we omit the plots for these different distributions to save space. Overall, the variance reduction of the BRA

Beta with parameters $A = 0.3d$ and $B = 0.5d$ is faster than that of the other BRA-based algorithms.

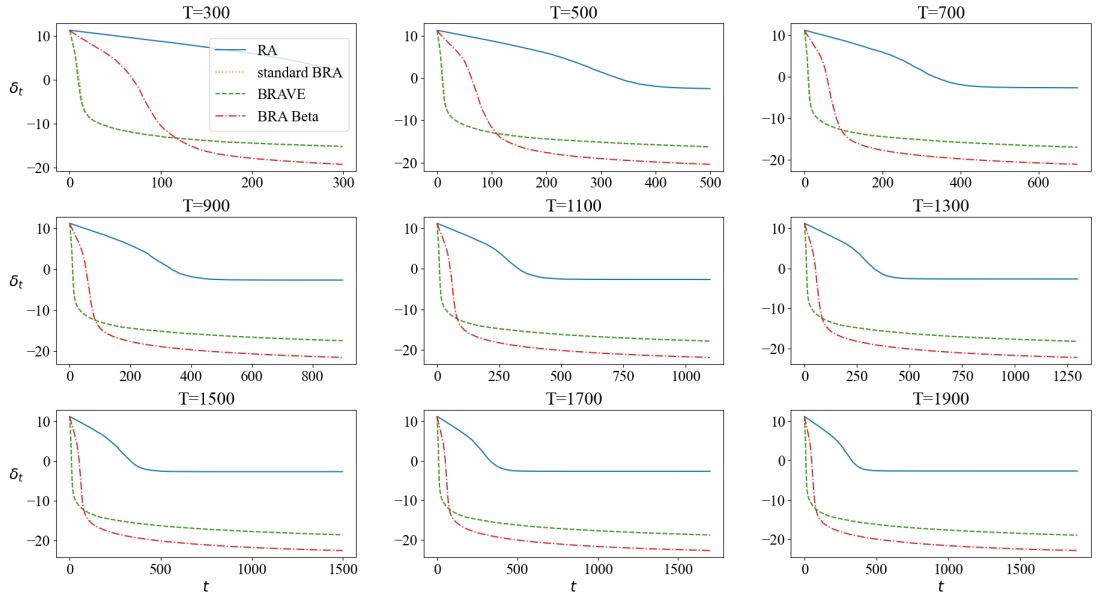

Figure 18: **Standard lognormal risks:** The performance of four BRA-based algorithms on convergence speed with $k = 100$, $n = 1000$ and $d = 100$ for T in the range of $[300, 1900]$.

5.3 Efficiency

This subsection illustrates the comparable efficient performance of the designed BRA Beta by comparing the average running times of four types of BRA algorithms.

Table 1 shows the average running times in seconds across 100 experiments for RA, BRA Binomial, BRAVE(greedy) and BRA Beta starting from the $\tilde{\boldsymbol{X}}$ sampled by a standard uniform distribution. We only report the case with uniform marginals because the initial matrix does not affect the execution speed of these algorithms. Except for varying parameters $n \in \{10, 100, 1000\}$ and $d \in \{10^2, 10^3, 10^4\}$ like in previous experiments, we also vary the maximum BRA steps $T \in \{50, 100, 300, 500\}$. It is easy to see and understand that the running time increases if n and d increase. Unsurprisingly, BRAVE(greedy) is the most time-consuming algorithm because it searches the equal variance for each BRA step. By contrast, the proposed BRA Beta is a much less time-consuming algorithm. Its running time is comparable with RA and BRA Binomial.

6 Application to Value-at-Risk

A problem of considerable interest in risk management and the financial industry is to determine sharp upper and lower bounds for the VaR at confidence level q of the sum of

Table 1: The average running times of 100 experiments in seconds for RA, BRA Binomial, BRAVE(greedy) and BRA Beta in minimizing the variance of the row sums of $\tilde{\boldsymbol{X}}$. The entries of $\tilde{\boldsymbol{X}}$ are sampled from a standard uniform distribution.

T	$n = 10$			$n = 100$			$n = 1000$		
	$d = 10^2$	$d = 10^3$	$d = 10^4$	$d = 10^2$	$d = 10^3$	$d = 10^4$	$d = 10^2$	$d = 10^3$	$d = 10^4$
RA									
50	0.005	0.010	0.066	0.005	0.027	0.722	0.021	1.109	11.695
100	0.008	0.019	0.138	0.010	0.055	1.401	0.041	2.523	23.385
300	0.023	0.058	0.458	0.032	0.167	4.157	0.132	7.408	93.927
500	0.040	0.099	0.744	0.068	0.269	6.802	0.261	9.156	119.725
BRA Binomial									
50	0.005	0.011	0.094	0.007	0.032	0.852	0.042	1.305	17.887
100	0.009	0.023	0.193	0.014	0.073	1.614	0.080	2.530	36.535
300	0.025	0.070	0.590	0.041	0.215	4.918	0.207	6.182	109.435
500	0.042	0.124	0.989	0.083	0.409	11.430	0.426	9.543	212.276
BRAVE(greedy)									
50	0.126	0.110	0.224	0.685	0.857	2.210	7.558	11.189	33.163
100	0.211	0.223	0.466	1.422	1.754	4.230	15.151	22.140	67.583
300	0.637	0.699	1.430	4.314	5.237	30.961	43.301	90.625	204.918
500	0.983	1.129	2.350	7.810	9.720	36.013	74.472	108.246	669.277
BRA Beta									
50	0.008	0.014	0.071	0.010	0.035	0.693	0.027	1.227	11.692
100	0.016	0.028	0.147	0.019	0.074	1.451	0.064	2.421	23.741
300	0.046	0.085	0.468	0.087	0.318	4.583	0.237	5.775	139.836
500	0.075	0.139	0.774	0.108	0.565	10.169	0.316	9.587	199.671

The average times are obtained using Python 3.8 on a Intel(R) Core(TM) i7-10710U CPU and 16 GB of RAM, running on Windows 10 Enterprise.

d dependent risks, with known marginal distributions of the risk vector $(X_1, \ldots, X_d)$ but unknown dependence structure. Specifically, banks are concerned with an upper bound on the VaR; see Embrechts and Puccetti (2010).

Let us first define the VaR of the sum of d dependent risks at confidence level q.

Definition 6.1 (VaR)**.** The VaR of the sum of d dependent risks at confidence level q denoted as $\mathrm{VaR}_q(S)$ is given as

$$\mathrm{VaR}_q(S) = \inf\{x \in \mathbb{R} | F_S(x) \geq q\},$$

where $q \in (0, 1)$, $S = X_1 + \cdots + X_d$ and $F_S(x)$ is the distribution function of S.

Next, the upper bound of $\mathrm{VaR}_q(S)$ can be formulated as

$$M_+ = \sup\{\mathrm{VaR}_q(S);\ S = X_1 + \cdots + X_d \text{ and } X_j \sim F_j,\ 1 \leq j \leq d\}.$$

The upper bound M_+ has been studied by Denuit et al. (1999), Puccetti and Rüschendorf (2012), Embrechts et al. (2013) and Bernard et al. (2017). Note that M_+ is a particular case when $\psi(X_1, \ldots, X_d) := \mathbb{1}_{X_1 + \cdots + X_d > x},\ x \in \mathbb{R}$ in Equation (1.1b). Wang et al. (2013) find the solution of M_+ when $F_j = F$ such that the distribution F has a monotone density

on its support; see Theorem 3.4 and 3.6 therein.

To demonstrate the excellent quality of the BRA Beta, we compare the results of the BRA Beta with the exact analytic values available in some cases for the upper bound problem.

When F is standard uniform or Pareto distribution, analytic results are available from Wang et al. (2013). For the case of standard uniform distributed risks,

$$M_+ = \frac{d(1+q)}{2}, \tag{6.1}$$

see Example 3.9 in Wang et al. (2013). For the case of Pareto distributed risks with tail index θ,

$$M_+ = (d-1)(1-(q+(d-1)c_d(q)))^{-1/\theta} + (1-(1-c_d(q)))^{-1/\theta}, \tag{6.2}$$

where $c_d(q)$ is the smallest $c \in [0, \frac{1}{d}(1-q)]$ such that

$$\frac{\theta}{\theta-1}((1-q-(d-1)c)^{1-1/\theta} - c^{1-1/\theta}) \geq \left(\frac{1}{d}(1-q)-c\right)((d-1)(1-q-(d-1)c)^{-1/\theta} + c^{-1/\theta}),$$

see Example 3.10 in Wang et al. (2013).

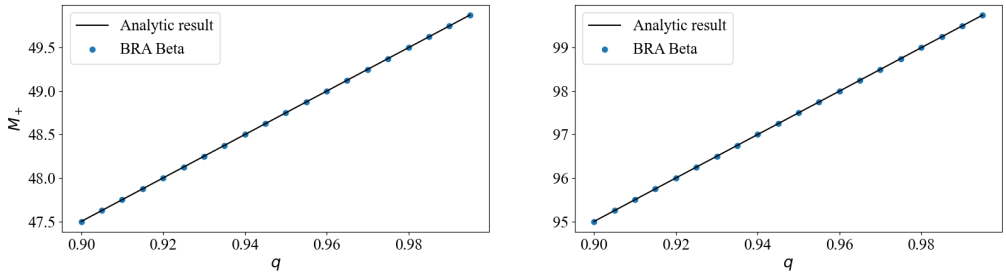

Figure 19: Upper bound on $\mathrm{VaR}_q(S)$, calculated using Equation (6.1), when $X_j \sim U[0,1]$. We set $d = 50$ (left) and $d = 100$ (right). Moreover, $T = 2000$ and $n = 10^5$ for BRA Beta.

To apply the BRA Beta with $A = 0.3d$ and $B = 0.5d$ to solve M_+, we simulate a $n \times d$ matrix $\boldsymbol{Y} = (y_1, \dots, y_d)$ where $y_j = (F_j^{-1}(\frac{1}{n+1}), \dots, F_j^{-1}(\frac{n}{n+1}))^T$ denotes the jth column, $j = 1, \dots, d$. Then the input matrix for BRA is a $(1-q)n \times d$ matrix $\tilde{\boldsymbol{X}} = (x_1, \dots, x_d)$ where $x_j = (F_j^{-1}(\frac{nq}{n+1}), \dots, F_j^{-1}(\frac{n}{n+1}))^T$. The upper bound on $\mathrm{VaR}_q(S)$ equals the minimum value of row sums of $\boldsymbol{X}^*$.

In Figure 19, we plot the upper bounds on $\mathrm{VaR}_q(S)$ when $d = 50$ (left) and $d = 100$ (right) uniformly distributed risks. M_+ at each confidence level q is computed either analytically or numerically. The solid line illustrates the analytic upper bounds calculated by Equation (6.1), while the dotted scatter plot represents the numerical upper bounds

from BRA Beta. We set a discretization level of $n = 10^5$ and a maximum number of BRA steps $T = 2000$ for BRA Beta. It is evident that the BRA Beta with $A = 0.3d$ and $B = 0.5d$ obtains reliable approximated upper bounds for $\mathrm{VaR}_q(S)$.

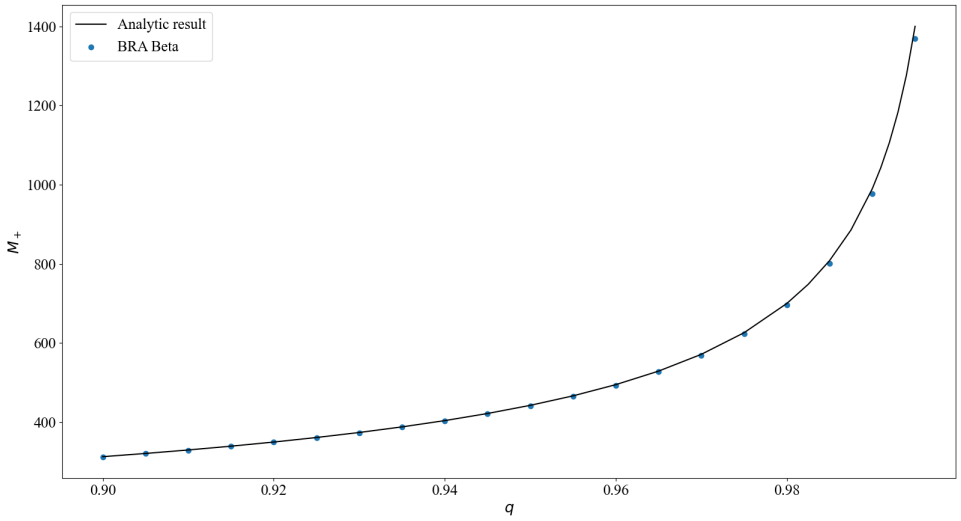

Figure 20: Upper bound on $\mathrm{VaR}_q(S)$, calculated using Equation (6.1), when $d = 50$ and $X_j \sim \mathrm{Pareto}(2)$. We set $T = 2000$ and $n = 10^5$ for BRA Beta.

In Figure 20, we plot the upper bounds of $\mathrm{VaR}_q(S)$ with the same parameter settings as the last figure, but $X_j \sim \mathrm{Pareto}(2)$. The analytic bounds are computed by Equation (6.2). The figure verifies the previous statement that BRA Beta works efficiently for finding the upper bound of $\mathrm{VaR}_q(S)$. Similarly, the quality of the BRA Beta for the lower bound problem is expected to be excellent.

While Wang et al. (2013) solve the upper bound problem under specific assumptions, a general solution for explicit values is challenging. However, our designed BRA Beta is an improved algorithm that can solve M_+ without constraints on the marginal distributions. Table 2 reports the numerical upper bounds on $\mathrm{VaR}_q(S)$, obtained by implementing BRA Beta, when $d = 50$ heterogeneous risks and $X_j \sim \mathrm{Pareto}\left(1.5 + \frac{j-1}{d-1}\right)$, $j = 1, \dots, d$.

Table 2: Upper bound on $\mathrm{VaR}_q(S)$, approximated using BRA Beta when $d = 50$ and $X_j \sim \mathrm{Pareto}\left(1.5 + \frac{j-1}{d-1}\right)$. We set $T = 2000$ and $n = 10^5$ for BRA Beta.

	M_+		M_+
q=0.90	346.140	q=0.95	501.839
q=0.91	366.194	q=0.96	565.822
q=0.92	390.015	q=0.97	660.659
q=0.93	418.933	q=0.98	822.231
q=0.94	455.037	q=0.99	1195.758

7 Conclusion

In this paper, we recall several versions of the block rearrangement algorithm and investigate the factors that impact its convergence speed and accuracy performance. These algorithms enable us to approximate sharp bounds for the expectation of some functions of dependent random variables X_j, $j = 1, \ldots, d$, where $X_j \sim F_j$.

To enhance the efficiency of the algorithm, we propose an improved version of BRA, named BRA Beta, which uses a time-dependent cardinality of the chosen subsets $I \subset \{1, \ldots, d\}$. For the choice of block sizes, BRA Beta utilizes Beta distributions with two dynamic parameters, α_t and β_t, controlled by $A = 0.3d$ and $B = 0.5d$. The BRA Beta performs similarly to the BRA Binomial at the beginning and more like the RA until the end. Specifically, when the number of risks is relatively high, the BRA Beta outperforms the RA from Puccetti and Rüschendorf (2012), the BRA Binomial from Bernard and McLeish (2016) and Bernard et al. (2018), and the BRAVE(greedy) from Boudt et al. (2018). Furthermore, the proposed BRA Beta algorithm is less time-consuming than BRAVE(greedy). The BRA Beta is applied and tested for the problem of approximating sharp bounds on VaR, showing improved convergence speed and accuracy performance w.r.t. the other variants of (B)RA in this kind of application.

Acknowledgments

The authors thank two reviewers for their valuable hints and remarks which helped to improve the paper. The authors gratefully acknowledge funding from Fonds Wetenschappelijk Onderzoek (grant numbers FWO SBOS006721N and FWO G015320N). Jinghui Chen would like to thank Xin Liu for his comments on the experiments. We thank the participants and the discussant at the 4th Financial Economic Meeting 29–30 June (Paris, France).

Compliance with Ethical Standards

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

Conflict of Interest All authors declare that they have no conflict of interest.

References

Bernard, C., O. Bondarenko, and S. Vanduffel (2018). Rearrangement algorithm and maximum entropy. *Annals of Operations Research 261*(1), 107–134.

Bernard, C., J. Chen, L. Rüschendorf, and S. Vanduffel (2023). Coskewness under dependence uncertainty. *Statistics & Probability Letters 199*, 109853.

Bernard, C. and D. McLeish (2016). Algorithms for finding copulas minimizing convex functions of sums. *Asia-Pacific Journal of Operational Research 33*(05), 1650040.

Bernard, C., L. Rüschendorf, and S. Vanduffel (2017). Value-at-risk bounds with variance constraints. *Journal of Risk and Insurance 84*(3), 923–959.

Boudt, K., E. Jakobsons, and S. Vanduffel (2018). Block rearranging elements within matrix columns to minimize the variability of the row sums. *4OR 16*(1), 31–50.

Chong, K. M. and N. Rice (1971). *Equimeasurable rearrangements of functions*. Number 28. Queen's University.

Day, P. W. (1972). Rearrangement inequalities. *Canadian Journal of Mathematics 24*(5), 930–943.

Denuit, M., C. Genest, and É. Marceau (1999). Stochastic bounds on sums of dependent risks. *Insurance: Mathematics and Economics 25*(1), 85–104.

Embrechts, P. and G. Puccetti (2006). Bounds for functions of dependent risks. *Finance and Stochastics 10*(3), 341–352.

Embrechts, P. and G. Puccetti (2010). Risk aggregation. In *Copula Theory and Its Applications: Proceedings of the Workshop Held in Warsaw, 25-26 September 2009*, pp. 111–126. Springer.

Embrechts, P., G. Puccetti, and L. Rüschendorf (2013). Model uncertainty and VaR aggregation. *Journal of Banking & Finance 37*(8), 2750–2764.

Hardy, G. H., J. E. Littlewood, G. Pólya, G. Pólya, et al. (1952). *Inequalities*. Cambridge University Press.

Hsu, W.-L. (1984). Approximation algorithms for the assembly line crew scheduling problem. *Mathematics of Operations Research 9*(3), 376–383.

Jakobsons, E. and R. Wang (2016). Negative dependence in matrix arrangement problems. *Annals of Operations Research*, 1–22.

Kirkpatrick, S., C. D. Gelatt Jr, and M. P. Vecchi (1983). Optimization by simulated annealing. *science 220*(4598), 671–680.

Korf, R. E. (1998). A complete anytime algorithm for number partitioning. *Artificial Intelligence 106*(2), 181–203.

Luxemburg, W. (1967). Rearrangement invariant banach function spaces. *Queens Papers in Pure and Applied Mathematics 10*, 83–144.

Puccetti, G. and L. Rüschendorf (2012). Computation of sharp bounds on the distribution of a function of dependent risks. *Journal of Computational and Applied Mathematics 236*(7), 1833–1840.

Puccetti, G. and L. Rüschendorf (2013). Sharp bounds for sums of dependent risks. *Journal of Applied Probability 50*(1), 42–53.

Puccetti, G. and L. Rüschendorf (2015). Computation of sharp bounds on the expected value of a supermodular function of risks with given marginals. *Communications in Statistics-Simulation and Computation 44*(3), 705–718.

Puccetti, G. and R. Wang (2015). Extremal dependence concepts. *Statistical Science 30*(4), 485–517.

Rüschendorf, L. (1983). Solution of a statistical optimization problem by rearrangement methods. *Metrika 30*(1), 55–61.

Rüschendorf, L., S. Vanduffel, and C. Bernard (2024). *Model Risk Management: Risk Bounds Under Uncertainty*. Cambridge University Press.

Van Laarhoven, P. J., E. H. Aarts, P. J. van Laarhoven, and E. H. Aarts (1987). *Simulated annealing*. Springer.

Wang, B. and R. Wang (2011). The complete mixability and convex minimization problems with monotone marginal densities. *Journal of Multivariate Analysis 102*(10), 1344–1360.

Wang, R., L. Peng, and J. Yang (2013). Bounds for the sum of dependent risks and worst value-at-risk with monotone marginal densities. *Finance and Stochastics 17*, 395–417.

Wilkinson, L. and M. Friendly (2009). The history of the cluster heat map. *American Statistician 63*(2), 179–184.