RUTCOR RESEARCH REPORT

Computing bounds for the probability of the union of events by different methods

József Bukszár [a] Gergely Mádi-Nagy [b]
Tamás Szántai [c]

RRR 10-2011, June 2011

RUTCOR
Rutgers Center for
Operations Research
Rutgers University
640 Bartholomew Road
Piscataway, New Jersey
08854-8003
Telephone: 732-445-3804
Telefax: 732-445-5472
Email: rrr@rutcor.rutgers.edu
http://rutcor.rutgers.edu/∼rrr

[a]Center for Biomarker Research and Personalized Medicine, School of Pharmacy, Virginia Commonwealth University, P.O. Box 980126, Richmond, VA 23298-0126, USA, jbukszar@vcu.edu

[b]Department of Operations Research, Eötvös University, Pázmány Péter sétány 1/c, Budapest, Hungary, H-1117, gergely@cs.elte.hu

[c]Institute of Mathematics, Budapest University of Technology and Economics, Műegyetem rakpart 1-3., Budapest, Hungary, 1111, szantai@math.bme.hu

RUTCOR RESEARCH REPORT
RRR 10-2011, JUNE 2011

COMPUTING BOUNDS FOR THE PROBABILITY OF THE UNION OF EVENTS BY DIFFERENT METHODS

József Bukszár   Gergely Mádi-Nagy   Tamás Szántai

Abstract. Let $A_1, \dots, A_n$ be arbitrary events. The underlying problem is to give lower and upper bounds on the probability $P(A_1 \cup \dots \cup A_n)$ based on $P(A_{i_1} \cap \dots \cap A_{i_k})$, $1 \leq i_1 < \dots < i_k \leq n$, where $k = 1, \dots, d$, and $d \leq n$ (usually $d << n$) is a certain integer, called the order of the problem or the bound. Most bounding techniques fall in one of the following two main categories: those that use (hyper)graph structures and the ones based on binomial moment problems. In this paper we compare bounds from the two categories with each other, in particular the bounds yielded by univariate and multivariate moment problems are compared with Bukszár's hypermultitree bounds. In the comparison we considered several numerical examples, most of which have important practical applications, e.g., the approximation of the values of multivariate cumulative distribution functions or the calculation of network reliability. We compare the bounds based on how close they are to the real value and the time required to compute them, however, the problems arising in the implementations of the methods as well as the limitations of the usability of the bounds are also illustrated.
Keywords: Discrete moment problem, Linear programming, Hypergraphs, Expectation bounds, Probability bounds, Multivariate distribution function, Network reliability
AMS: 62H99, 90C05

Acknowledgements: The European Union and the European Social Fund have provided financial support to the project under the grant agreement no. TÁMOP 4.2.1./B-09/1/KMR-2010-0003.

1 Introduction

Let $A_1, \ldots, A_n$ be arbitrary events. The underlying problem is to give lower and upper bounds on the probability $P(A_1 \cup \cdots \cup A_n)$ based on

$$P(A_{i_1} \cap \cdots \cap A_{i_k}), \quad 1 \le i_1 < \cdots < i_k \le n, \tag{1.1}$$

where $k = 1, \ldots, d$, and $d \le n$ (usually $d << n$) is a certain integer and it is called *the order of the problem.*

Probably the best known bounds of this type are the Boole and Bonferroni inequalities (see Boole, 1854 and Bonferroni, 1937), which, however, usually fall fairly far from the probability of the union, often even out of [0,1]. Recent research results are based on two main bounding techniques, one of them relies on (hyper)graph structures and the other one relies on binomial moment problems.

In the field of (hyper)graph-based bounds, the first remarkable result is the Hunter-Worsley second order bound (see Hunter, 1976, Worsley, 1982). Later the Hunter's bound was generalized by Tomescu (1986). Bukszár and Prékopa (see Bukszár and Prékopa, 2001) improved on Hunter's bound using special hypergraph structures, which they named cherry trees. They also proved that bounds computed by special type of cherry trees, the so called t-cherry trees, can be identified with dual feasible bases of the Boolean problem, which is an LP whose optimal solution is the sharp bound. Although the Boolean problem is typically too large to solve in practice, the above result ensures that the cherry trees provide us with the sharp bound in numerous cases and also give insight into the nature of the bound. Generalizations of Hunter's bound and the cherry tree bounds are multitree bounds of Bukszár (2001) and hypercherry tree bounds (see Bukszár and Szántai, 2002). Further improvements are the hypermultitree bounds (Bukszár, 2003), that also generalize Tomescu's bounds. Another bounding technique is to construct so-called Bonferroni-type inequalities, which give better bounds than the classical sieve formulae. Several results in this field has been achieved by Seneta (see Recsei and Seneta, 1987, Hoppe and Seneta, 1990) and Dohmen (2003).

The univariate binomial moment problem (BMP) was formulated as an LP problem by Prékopa, who also developed a numerically stable dual method to solve it, see Prékopa (1990,1992,1995). He used the BMP to construct bounds for the probability of the union of events as well. The multivariate binomial moment problem (MBMP) was also introduced by Prékopa (1998), however, the solution of the problem is very difficult (practically impossible, even for middle-sized problems), because of the adverse numerical condition of the coefficient matrix. Fortunately, for the bivariate case Mádi-Nagy and Prékopa (2004) found a wide variety of dual feasible bases of the LP problem and used them to give numerically stable bounds. Mádi-Nagy (2009) generalized the bivariate method for the multivariate case and presented usually better bounds on the probability of the union of events. Mádi-Nagy (2005) improved further the algorithm that finds bounds. Further methods to construct Bonferroni-type inequalities based on binomial moments can be found in Galambos and Simonelli (1996).

Although numerical examples are used to demonstrate the utility of the newly introduced bounds, only a few comparisons between the bounds and the methods that calculate them can be found in the literature. In addition, the comparison of a novel bound is often confined to the existing bounds obtained with the same type of bounding technique. The aim of this paper is twofold. First to analyze the efficiency and usability of the recent methods of the two main bounding techniques through numerical examples. Second to present numerical examples for several applications of the Bonferroni-type bounds, e.g., approximation the values of the c.d.f. of multivariate normal distribution and of multivariate Dirichlet distribution, bounding network reliability.

The paper is organized as follows. In Section 2 the methods of both fields, which will be applied in the numerical examples, are shortly introduced. Section 3 describes the approximation method of the values of multivariate c.d.f.s. Here, numerical examples based on multivariate normal and Dirichlet distribution are presented. Section 4 shows the application on network reliability illustrated by numerical examples. Section 5 presents the results on randomly generated event systems. Section 6 concludes the paper.

2 The bounding methods

2.1 Graph-based bounds

Bukszár and Prékopa (2001), Bukszár and Szántai (2002) and Bukszár (2003) introduced the concepts of cherry-tree, t-cherrytree, hypercherry-tree, multitree, and hypermultitree. These are special type of graph and hypergraph structures designed to give lower and upper bounds for the probability of the union of events.

Let $A_1, \ldots, A_n$ be arbitrary events in a probability space (Ω, A, P). Our goal is to find an upper bound for $P(A_1 \cup \ldots \cup A_n)$ based on some of the probabilities $P(A_{k_1} \cap \ldots \cap A_{k_i})$, where $1 \leq k_1 \leq \ldots \leq k_i \leq n$, $i = 1, \ldots, d$. The bounds of this type are called d-th order upper bounds. For $m = d - 1$, in Bukszár (2001) a special hypergraph structure with n vertices called m-multitree was introduced. To each m-multitree an upper bound can be assigned. (For $m = 1$ one obtains the Hunter–Worsley bound, for $m = 2$ one obtains the cherry tree bound.)

Definition 2.1 *(Bukszár (2003)) Let m be a positive integer. An m-multicherry is a hypergraph of the form $(V, \varepsilon_2, \ldots, \varepsilon_{m+1})$, where $V = \{v_1, \ldots, v_{m+1}\}$ is the set of vertices and for each $i = 2, \ldots, m+1$ the family of hyperedges ε_i is the set of all subsets of ε_i^{m+1} is the set of all subsets of $\{v_1, \ldots, v_{m+1}\}$ containing i vertices with v_{m+1} included, i.e., $\varepsilon_i = \{H | v_{m+1} \in H \subset \{v_1, \ldots, v_{m+1}\}, |H| = i\}$. The vertex v_{m+1} is called the dominating vertex of the m-multicherry. The m-multicherry with dominating vertex v_{m+1} and with non-dominating vertices $v_1, \ldots, v_m$ is denoted by $(\{v_1, \ldots, v_m\}, v_{m+1})$.*

Definition 2.2 *(Bukszár (2003)) Let m be a positive integer. An m-multitree is a hypergraph of the form $(V, \varepsilon_2, \ldots, \varepsilon_{m+1})$, where V is the set of vertices and ε_i's are sets of hyperedges containing i vertices. An m-multitree is recursively defined by the following two rules.*

(i) The smallest m*-multitree* $\Delta = (V, \varepsilon_2, \ldots, \varepsilon_{m+1})$ *has* m *vertices and* ε_i *is the family of all subsets of* V *containing* i *vertices (here* $\varepsilon_{m+1} = \emptyset$*).*

(ii) From an m*-multitree* $\Delta = (V, \varepsilon_2, \ldots, \varepsilon_{m+1})$ *we can obtain a new* m*-multitree* $\Delta' = \left(V', \varepsilon'_2, \ldots, \ \varepsilon'_{m+1}\right)$ *by adjoining an* m*-multicherry* $(\{v_1, \ldots, v_m\}, v_{m+1})$*, where* $v_1, \ldots, v_m \in V$ *and* v_{m+1} *is a new vertex (i.e.,* $v_{m+1} \notin V$*). More precisely* $V' = V \cup \{v_{m+1}\}$*,* $\varepsilon'_i = \varepsilon_i \cup \{H | v_{m+1} \in H \subset \{v_1, \ldots, \ v_{m+1}\}, |H| = i\}$*.*

Let $\{A_1, \ldots, A_n\}$ be a set of n events and let $V = \{1, \ldots, n\}$ the set of indices.

Definition 2.3 *(Bukszár (2003)) The weight of an* m*-multitree according to the set* $\{A_1, \ldots, A_n\}$ *of events is given by the following formula:*

$$\begin{aligned} w(\Delta^m) &= \sum_{\{l_1,l_2\}\in\varepsilon_2} P(A_{l_1} \cap A_{l_2}) \\ &- \sum_{\{l_1,l_2,l_3\}\in\varepsilon_3} P(A_{l_1} \cap A_{l_2} \cap A_{l_3}) \\ &+ \cdots \\ &+ (-1)^{m+1} \sum_{\{l_1,\ldots,l_{m+1}\}\in\varepsilon_{m+1}} P\left(A_{l_1} \cap \ldots \cap A_{l_{m+1}}\right). \end{aligned}$$

Theorem 2.1 *If* $w(\Delta)$ *is the weight of an arbitrary* m*-multitree according to the set of events* $\{A_1, \ldots, A_n\}$ *then*

$$P(A_1 \cup \ldots \cup A_n) \leq \sum_{i=1}^{n} P(A_i) - w(\Delta^m).$$

It is obvious from the inequality of the theorem that it is important to construct the heaviest m-multitree possible.

In Bukszár (2001) an algorithm is given for finding heavy m-multitrees. The algorithm starts from the heaviest 1-multitree, which can be found by a greedy algorithm. In a general step an r-multitree is extended to an $(r+1)$-multitree in such a way that the bound the $(r+1)$-multitree provides us is at least as good as, and typically better than, the one provided by the r-multitree. Starting from $r = 1$, the algorithm improves on the bounds step by step by involving a relatively few number of intersection probabilities.

Later Bukszár (2003) introduced the concept of hypermultitree in order to give lower and upper bounds for the probability of the union of n events. These bounds are a generalization of Tomescu's lower and upper bounds (I. Tomescu (1986)).

The definition of the (h,m)-hipermultitree uses the definition of m-multitree, and is given recursively.

Definition 2.4 *(Bukszár (2003)) Let* $h \geq 0$ *and* $m \geq 1$ *be arbitrary integers. An* (h, m)*-hypermultitree is a hypergraph of the form* $(V,_h \varepsilon_2, \ldots,_h \varepsilon_{m+1})$*, where* V *is the set of vertices and* ${}_h\varepsilon_i$*'s are sets of hyperedges containing* $h+i$ *vertices. An* (h, m)*-hypermultitree is defined recursively by the following rules.*

(i) The $(0,m)$-hypermultitrees are the same as the m-multitrees.

(ii) The smallest (h,m)-hypermultitree $\Delta = (V,_h\varepsilon_2,\ldots,_h\varepsilon_{m+1})$ has $h+m$ vertices and ${}_h\varepsilon_i$ consists of all subsets of V containing $h+i$ vertices (here ${}_h\varepsilon_{m+1} = \emptyset$.

(iii) From an (h,m)-hypermultitree $\Delta = (V,_h\varepsilon_2,\ldots,_h\varepsilon_{m+1})$ we can obtain a new (h,m)-hypermultitree in the following manner. Let $\Gamma = \left(V,_{h-1}\varepsilon_2^*,\ldots,_{h-1}\varepsilon_{m+1}^*\right)$ be an arbitrary $(h-1,m$-hypermultitree with the same set of vertices as in Δ. By joining a new vertex v to Δ and the hyperedges of Γ extended by v, we obtain the new (h,m)-hypermultitree $\Delta' = \left(V,_h\varepsilon_2',\ldots,_h\varepsilon_{m+1}'\right)$, i.e.,

$$V' = V \cup \{v\} \qquad {}_h\varepsilon_i' =_h \varepsilon_i \cup \{E \cup \{v\} | E \in_{h-1} \varepsilon_i^*\}$$

The (h,m)-hypermultitrees are generalizations of Tomescu's hypertrees, which are the $(h,1)$-hypermultitrees in our definition.

Definition 2.5 *(Bukszár (2003)) The weight of the $\Delta^{h,m} = (V,_h\varepsilon_2,\ldots,_h\varepsilon_{m+1})$,$(h,m)$-hypermultitree, according to the set of events $\{A_1,\ldots,A_n\}$ is*

$$\begin{aligned} w\left(\Delta^{h,m}\right) &= \sum_{\{i_1,\ldots,i_{h+2}\}\in_h\varepsilon_2} P\left(A_{i_1}\cap\ldots\cap A_{i_{h+2}}\right) \\ &- \sum_{\{i_1,\ldots,i_{h+3}\}\in_h\varepsilon_3} P\left(A_{i_1}\cap\ldots\cap A_{i_{h+3}}\right) \\ &+ \ldots \\ &+ (-1)^{m+1} \sum_{\{i_1,\ldots,i_{m+1}\}\in_h\varepsilon_{m+1}} P\left(A_{i_1}\cap\ldots\cap A_{i_{h+m+1}}\right) \end{aligned}$$

Theorem 2.2 *(Bukszár (2003)) If $\Delta^{h,m} = (V,_h\varepsilon_2,\ldots,_h\varepsilon_{m+1})$ is an arbitrary (h,m)-hypermultitree, according to the set of events $\{A_1,\ldots,A_n\}$ then the following inequalities hold:*

- *if h is even, then:*

$$P\left(\bigcup_{i=1}^{n} A_i\right) \leq \sum_{k=1}^{h+1} (-1)^{k-1} S_k - w\left(\Delta^{h,m}\right);$$

- *if h is odd, then:*

$$P\left(\bigcup_{i=1}^{n} A_i\right) \geq \sum_{k=1}^{h+1} (-1)^{k-1} S_k + w\left(\Delta^{h,m}\right),$$

where $S_k = \sum_{1\leq i_1\leq\ldots\leq i_k\leq n} P\left(A_{i_1}\cap\ldots\cap A_{i_k}\right)$.

Remark 2.1 *For the special case $m=1$ we obtain the Tomescu bounds, for $h=0$ and $m=1$ we obtain the Hunter-Worsley bound.*

Bukszár (2003) gives an algorithm that uses Theorem 2.2 in order to achieve lower and upper bounds for the union of events. The algorithm that finds a heavy $(1,m)$-hypermultitree consists of two phases. In the first phase a heavy $(1,1)$-hypermultitree is constructed and in the second phase the $(1,r)$-hypermultitree is extended to an $(1,r+1)$-hypermultitree recursively $(r=1,\ldots,m-1)$ in order to obtain better bounds.

2.2 Bounds based on binomial moments

First we introduce the univariate BMP. We need the following

Definition 2.6 *Let X be a random variable with the support $I \subset \mathbb{N}$. The kth binomial moment of X is*

$$E\left[\binom{X}{k}\right],$$

$k = 1, 2, ...$ $E\left[\binom{X}{0}\right] = 1$ by definition.

Theorem 2.3 *Consider the events $A_1, A_2, \ldots A_n$. Let the random variable X with the support $\{0, 1, \ldots, n\}$ be the number of those events which occur. Then the kth binomial moment of X equals the following,*

$$S_k = \sum_{0 \le i_1 < i_2 < \cdots < i_k \le n} P(A_{i_1} \cap A_{i_2} \cap \cdots \cap A_{i_k}), \tag{2.1}$$

$k = 1, 2, ..., n$.

Proof. It is easy to prove by the use of indicator variables. See, e.g., Prékopa (1995, p.182). □

It follows from the above theorem that if the probabilities of the intersections are given up to the mth order, then the values of the binomial moments can be calculated up to the same order.

The univariate binomial moment problem of order m can be formulated as

$$\begin{array}{llllllll}
\min(\max) & f_0p_0 & +f_1p_1 & +\cdots & +f_np_n & & \\
\text{subject to} & & & & & & \\
& p_0 & +p_1 & +\cdots & +p_n & = & S_0(=1) \\
& \binom{0}{1}p_0 & +\binom{1}{1}p_1 & +\cdots & +\binom{s}{1}p_n & = & S_1 \\
& \binom{0}{2}p_0 & +\binom{1}{2}p_1 & +\cdots & +\binom{s}{2}p_n & = & S_2 \\
& & & & & \vdots & \\
& \binom{0}{m}p_0 & +\binom{1}{m}p_1 & +\cdots & +\binom{s}{m}p_n & = & S_m \\
& p_0, & p_1, & \ldots & p_n & \ge & 0,
\end{array} \tag{2.2}$$

where the probabilities

$$p_i = P(X = i),\ i = 0, 1, \ldots, n,$$

are the unknown variables while the values of the right hand side are given. If we would like to give bounds for the probability of the union of the n events, then we have to consider

$$f_i = \begin{cases} 0 \text{ if } i = 0 \\ 1 \text{ otherwise} \end{cases}. \tag{2.3}$$

The advantage of the univariate BMP is that it has a moderate size contrary to the linear programming model of the original problem of the paper.

The coefficient matrix of the problem (2.2) is very bad conditioned, hence the general solvers usually do not give reliable results. Fortunately, Prékopa (1990b) developed a numerically stable dual method for the solution. This method is based on theorems which give the subscript structures of columns of all dual feasible bases. In the following this method is used to give the best bounds based on the information of univariate binomial moments. The numerical implementation of the method in C++ can be found at Mádi-Nagy (2009b).

The univariate BMP yields bounds based on the aggregated information of the (univariate) binomial moments. The aggregation is needed to decrease the size of the problem, however, due to the aggregation the amount of information is reduced as well, which results in weaker bounds. Better bounds can be obtained with less aggregation. In case of the multivariate BMP more information is used than in the univariate case, hence better bounds can be achieved, while the size of the problem remains tractable.

In order to introduce the multivariate BMP we need the following

Definition 2.7 *The* $(\alpha_1,\dots,\alpha_s)$-order cross-binomial moment *of the random vector* $(X_1,\dots,X_s)$*, with the support* $Z\subset \mathbb{N}^s$*, is defined as*

$$S_{\alpha_1\dots\alpha_s}=E\left[\binom{X_1}{\alpha_1}\cdots\binom{X_s}{\alpha_s}\right],$$

where $\alpha_1,\dots,\alpha_s$ *are nonnegative integers. The sum* $\alpha_1+\cdots+\alpha_s$ *will be called* the total order of the moment.

Theorem 2.4 *Assume that we have* n *arbitrary events. We subdivided them into* s *subsequences. Let the* j*th subsequence be designated as* $A_{j1},\dots,A_{jn_j}$*,* $j=1,\dots,s$*. Certainly,* $n_1+\cdots+n_s=n$*. Let the random variable* X_j *with the support* $Z_j=\{0,1,\dots,n_j\}$ *be the number of events that occur in the* j*th sequence,* $j=1,\dots,s$*. In case of event sequences*

$$E\left[\binom{X_1}{\alpha_1}\cdots\binom{X_s}{\alpha_s}\right]=\sum_{1\le i_{j1}<\cdots<i_{j\alpha_j}\le n_j,\ j=1,\dots,s}P[A_{1i_{11}}\cap\cdots\cap A_{1i_{1\alpha_1}}\cap\cdots\cap A_{si_{s1}}\cap\cdots\cap A_{si_{s\alpha_s}}].$$

It follows from the above theorem that if the probabilities of the intersections are given up to the mth order, then the values of the cross-binomial moments can be calculated up to the same total order.

The multivariate BMP can be formulated as

$$\begin{aligned}
&\min(\max)\ \sum_{i_1=0}^{n_1}\cdots\sum_{i_s=0}^{n_s} f_{i_1\dots i_s}p_{i_1\dots i_s}\\
\text{subject to}\quad&\\
&\sum_{i_1=0}^{n_1}\cdots\sum_{i_s=0}^{n_s}\binom{i_1}{\alpha_1}\cdots\binom{i_s}{\alpha_s}p_{i_1\dots i_s}=S_{\alpha_1\dots\alpha_s}\\
&\text{for } (\alpha_1\dots\alpha_s)\in H\\
&p_{i_1\dots i_s}\ge 0,\ \text{all } i_1,\dots,i_s.
\end{aligned}\tag{2.4}$$

If we would like to give bounds for the probability of the union of all the n events, then we have to consider

$$f_{i_1 \dots i_s} = \begin{cases} 0 \text{ if } (i_1, \dots, i_s) = (0, \dots, 0) \\ 1 \text{ otherwise} \end{cases} . \tag{2.5}$$

If the probabilities of the intersections of the events are given up to the order m, then it would be natural to consider the following set of H:

$$H = \{(\alpha_1, \dots, \alpha_s) |\ 0 \le \alpha_j,\ \alpha_j \text{ integer},\ \alpha_1 + \dots + \alpha_s \le m,\ j = 1, \dots, s\}. \tag{2.6}$$

However, the LP problem (2.4) with subscript set (2.6) is numerically unstable and in most cases cannot be solved or bounded even by special methods. Fortunately, a useful bounding method has been found for the following subscript set:

$$\begin{aligned} H \ = \ & \{(\alpha_1, 0, \dots, 0, \alpha_j, 0, \dots, 0) |\ 0 \le \alpha_1, \alpha_j,\ \alpha_1, \alpha_j \text{ integer},\ \alpha_1 + \alpha_j \le m, \\ & j = 2, \dots, s.\} \end{aligned} \tag{2.7}$$

In the bivariate case ($s = 2$) the method of Mádi-Nagy and Prékopa (2004) gives usually good lower and upper bounds. We remark, that in the bivariate case the sets (2.6) and (2.7) coincide. In case of higher dimensions the method of Mádi-Nagy (2009) yields upper bounds. Unfortunately, the bounds are not sharp in the sense that they are not the extreme values of the LP problem. However, the bounds are based on known dual feasible basis structures, hence they can be improved by the execution of some steps of the dual simplex method starting from those bases.

In this paper the above methods are used to give bounds based on the information of multivariate binomial moments. The numerical implementation of the methods in C++ can be found at Mádi-Nagy (2009b).

We emphasize again the reason that all the above methods are taken into account is that only the univariate method gives sharp bound. Usually the methods of higher dimensions give better bounds (because they use more information), however, sometimes the bounds of the univariate BMP are better. This later case means, that even the sharp bound of a multivariate problem is better than the univariate one, the (non-sharp) bound yielded by the method of Mádi-Nagy and Prékopa (2004) can be weaker.

3 Approximation of c.d.f. of multivariate distributions

For any multivariate cumulative distribution function we have

$$F(x_1, \dots, x_n) = P(\xi_1 \le x_1, \dots, \xi_n \le x_n) = 1 - P(A_1 \cup \dots \cup A_n), \tag{3.1}$$

where

$$A_i = \{\xi_i > x_i\},\ i = 1, \dots, n.$$

Assume that the value of the CDF can be calculated easily up to m dimensions (usually $n >> m$). In this case the probabilities

$$P(A_{i_1} \cap \dots \cap A_{i_k}), \quad 1 \le i_1 < \dots < i_k \le n,\ k \le m, \tag{3.2}$$

can be found and we can calculate lower and upper bounds on the values of (3.1) by use of our bounding techniques. The above method is very useful in case of higher dimensions, where the c.d.f. cannot be calculated by integration.

The running time of the method consists of two parts. The first part is about the calculation of the probabilities (3.2) that are used in the bounding method. The second part is the time of the bounding procedure. *Particular advantage of the graph-based methods that they only use a few probabilities of (3.2).* This dramatically reduces the first part of the running time comparing to the binomial moment methods, where all probabilities of (3.2) have to be calculated, up to the given order, in order to get the moments. Additionally, the use of less probabilities reduces the accumulation of numerical errors arising in the calculation of the values of (3.2).

3.1 Examples on multivariate Dirichlet distribution

Not too high dimensional c.d.f. values of the Dirichlet distribution can be calculated by an implicit recursive algorithm developed recently by A. Gouda and T. Szántai (see Gouda and Szántai (2010)). In Tables 1–9 we give the results of three different 20 dimensional c.d.f. calculations. In case of binomial moment methods the first part of the running time, i.e. calculation of the probabilities of all intersections up to the given order, is dominant part of the whole running time. Hence the two parts of the running time are shown separately: "first part+second part".

Table 1: Parameter and c.d.f. argument values of the Dirichlet distribution in Example 1.

index	ϑ values	x values	index	ϑ values	x values
1	1.5	0.19	11	1.5	0.1
2	1.4	0.19	12	1.4	0.1
3	1.3	0.19	13	1.3	0.1
4	1.2	0.19	14	1.2	0.1
5	1.1	0.19	15	1.1	0.1
6	1.2	0.19	16	1.2	0.1
7	1.3	0.19	17	1.3	0.1
8	1.4	0.19	18	1.4	0.1
9	1.2	0.19	19	1.2	0.1
10	1.3	0.19	20	1.3	0.1
			21	2.1	

3.2 Examples on multivariate normal distribution

We will test the bounds on the example investigated by T. M. Costigan (see Costigan (1996) Example 6.3). Let $\xi_1, \ldots, \xi_n$ be normally distributed random vector with

Table 2: 3-rd order bounds for the Dirichlet c.d.f. of Example 1.

	bound value	CPU in seconds
Bukszár's lower	0.103087	0.04
Bukszár's upper	0.293917	1.40
Aggregated lower	0.262142	1.18+0.00
Aggregated upper	0.305193	1.18+0.00
Bivariate lower	0.262142	1.18+0.01
Bivariate upper	0.297989	1.18+0.01
Multivariate lower	0.262142	1.18+0.01

Table 3: 5-th order bounds for the Dirichlet c.d.f. of Example 1.

	bound value	CPU in seconds
Bukszár's lower	0.188656	14.19
Bukszár's upper	0.274744	125.05
Aggregated lower	0.264887	31285.25+0.01
Aggregated upper	0.264928	31285.25+0.00
Bivariate lower	0.264887	31285.25+0.11
Bivariate upper	0.264914	31285.25+0.50
Multivariate lower	0.264887	31285.25+0.49

$$\begin{aligned} E(\xi_i) &= 0, i = 1, \ldots, n \\ D(\xi_i) &= 1, i = 1, \ldots, n \end{aligned}$$

and

$$\mathrm{corr}(\xi_i, \xi_j) = \begin{cases} 1 - 0.1|i-j|, & \text{if } |i-j| \le 9, \\ 0, & \text{if } |i-j| > 9 \end{cases} \quad i, j = 1, \ldots, n.$$

T. M. Costigan calculated lower bounds on the probability

$$P(\xi_1 \le c, \ldots, \xi_n \le c)$$

with $c = 2/\sqrt{10}$. As in joint constrained probabilistic programming problems we are interested in c.d.f. values close to one, we changed the argument values c to $c = 2 + 2/\sqrt{10}$. The lower dimensional marginal c.d.f. values were calculated by Genz's numerical integration method (see A. Genz (1992)). In case of the 5th order bounds the BMP bounds cannot be applied because the cumulated numerical errors in the calculation of the higher order moments lead to infeasible problems. The numerical results can be seen in Table 11-14.

Table 4: Parameter and c.d.f. argument values of the Dirichlet distribution in Example 2.

index	ϑ values	x values	index	ϑ values	x values
1	1.5	0.1	11	1.5	0.1
2	1.4	0.2	12	1.4	0.2
3	1.3	0.2	13	1.3	0.2
4	1.2	0.2	14	1.2	0.2
5	1.1	0.2	15	1.1	0.2
6	1.2	0.3	16	1.2	0.1
7	1.3	0.2	17	1.3	0.2
8	1.4	0.1	18	1.4	0.1
9	1.2	0.2	19	1.2	0.2
10	1.3	0.1	20	1.3	0.1
			21	2.1	

Table 5: 3-rd order bounds for the Dirichlet c.d.f. of Example 2.

	bound value	CPU in seconds
Bukszár's lower	0.290188	0.02
Bukszár's upper	0.377891	1.41
Aggregated lower	0.366196	1.22+0.00
Aggregated upper	0.385005	1.22+0.00
Bivariate lower	0.366196	1.22+0.01
Bivariate upper	0.382324	1.22+0.02
Multivariate lower	0.366196	1.22+0.02

4 Application on network reliability

Consider a compact, directed and acyclic network $(\mathcal{N}, \mathcal{A})$. Assume that $\mathcal{N} = \{c_1, \ldots, c_n\}$ is the set of nodes, and $\mathcal{A} \subset \mathcal{N} \times \mathcal{N}$ is the set of arcs. Without restricting generality, we may assume that there is exactly one node with the property that no arc leads into it and there is exactly one node with the property that no arc goes out of it. These two nodes will be called start and terminal nodes, respectively. Suppose that c_1 is the single start node and c_n is the single terminal node. Suppose that each arc is alive with probability p independently of each other. The probability of the event that one can get from the start node c_1 to the terminal node c_n along living arcs is called the reliability of the network. Our goal is to find bounds (both lower and upper) on the reliability of large sized networks.

One possible way to do this is to determine all paths leading from the start node c_1 to the terminal node c_n. Denote these paths by $P_1, \ldots, P_N$ and denote the event that all arcs along the path P_i are alive by $A_i, i = 1, \ldots, N$. With these notations the reliability of the network equals

$$P(A_1 \cup \cdots \cup A_N).$$

Table 6: 5-th order bounds for the Dirichlet c.d.f. of Example 2.

	bound value	CPU in seconds
Bukszár's lower	0.345615	14.15
Bukszár's upper	0.368858	119.46
Aggregated lower	0.367014	27116.96+0.00
Aggregated upper	0.367021	27116.96+0.00
Bivariate lower	0.367014	27116.96+0.13
Bivariate upper	0.367019	27116.96+0.60
Multivariate lower	0.367014	27116.96+0.06

Table 7: Parameter and c.d.f. argument values of the Dirichlet distribution in Example 3.

index	ϑ values	x values	index	ϑ values	x values
1	1.5	0.1	11	1.5	0.5
2	1.4	0.3	12	1.4	0.8
3	1.3	0.2	13	1.3	0.1
4	1.2	0.3	14	1.2	0.9
5	1.1	0.2	15	1.1	0.5
6	1.2	0.1	16	1.2	0.3
7	1.3	0.3	17	1.3	0.2
8	1.4	0.2	18	1.4	0.1
9	1.2	0.1	19	1.2	0.2
10	1.3	0.3	20	1.3	0.7
			21	2.1	

The probability of the event that all arcs are alive along a few, say not more than five, paths can be calculated quickly. Even if there exist a number of paths leading from the start node to the terminal node we can calculate bounds on the probability of the union of events using these probabilities only.

As a numerical example let us regard the following network with 8 number of nodes and 16 number of arcs. $\mathcal{N} = \{c_1, \ldots, c_8\}$ and
$\mathcal{A} = \{(c_1, c_2), (c_1, c_3), (c_1, c_4), (c_1, c_5), (c_2, c_3), (c_2, c_5), (c_2, c_6), (c_3, c_4), (c_3, c_5), (c_4, c_6), (c_4, c_7), (c_5, c_6), (c_5, c_8), (c_6, c_7), (c_6, c_8), (c_7, c_8)\}$. In this network there exists altogether 23 paths, the path-arc incidence matrix can be seen on Table 10, the rows represent the paths and the columns represent the arcs.

For each arc, let p be the probability that it is alive. Suppose that the events that the arcs are alive are independent of each other. As the number of the paths is not too large we can calculate the exact reliability that one can get from node c_1 to node c_8 along living arcs:

$$p^2 + 6p^3 + 5p^4 - 18p^5 - 33p^6 + 26p^7 + 129p^8 - 108p^9$$
$$-273p^{10} + 605p^{11} - 547p^{12} + 279p^{13} - 84p^{14} + 14p^{15} - p^{16}$$

Table 8: 3-rd order bounds for the Dirichlet c.d.f. of Example 3.

	bound value	CPU in seconds
Bukszár's lower	0.524795	0.01
Bukszár's upper	0.544452	0.75
Aggregated lower	0.542883	0.65+0.00
Aggregated upper	0.546904	0.65+0.00
Bivariate lower	0.542883	0.65+0.01
Bivariate upper	0.546528	0.65+0.01
Multivariate lower	0.542883	0.65+0.02

Table 9: 5-th order bounds for the Dirichlet c.d.f. of Example 3.

	bound value	CPU in seconds
Bukszár's lower	0.542030	10.82
Bukszár's upper	0.542992	22.77
Aggregated lower	0.542963	2662.03+0.00
Aggregated upper	0.542963	2662.03+0.01
Bivariate lower	0.542963	2662.03+0.11
Bivariate upper	0.542963	2662.03+0.50
Multivariate lower	0.542963	2662.03+0.49

The probabilities of the intersections up to the 3rd and 5th order can also be calculated easily. The exact probabilities and bounds, based on the probabilities of intersections, are depicted on Figure 1 and 2.

Table 10: The path-arc incidence matrix of the example network.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0
2	1	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0
3	0	0	0	1	0	0	0	0	0	0	0	1	0	0	1	0
4	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0
5	0	0	0	1	0	0	0	0	0	0	0	1	0	1	0	1
6	1	0	0	0	1	0	0	1	0	0	1	0	0	0	0	1
7	1	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0
8	1	0	0	0	1	0	0	1	0	1	0	0	0	0	1	0
9	1	0	0	0	0	0	1	0	0	0	0	0	0	1	0	1
10	1	0	0	0	1	0	0	1	0	1	0	0	0	1	0	1
11	1	0	0	0	1	0	0	0	1	0	0	1	0	0	1	0
12	1	0	0	0	0	1	0	0	0	0	0	1	0	0	1	0
13	1	0	0	0	1	0	0	0	1	0	0	1	0	1	0	1
14	1	0	0	0	0	1	0	0	0	0	0	1	0	1	0	1
15	0	1	0	0	0	0	0	0	1	0	0	0	1	0	0	0
16	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	1
17	0	1	0	0	0	0	0	1	0	0	1	0	0	0	0	1
18	0	0	1	0	0	0	0	0	0	1	0	0	0	0	1	0
19	0	1	0	0	0	0	0	1	0	1	0	0	0	0	1	0
20	0	0	1	0	0	0	0	0	0	1	0	0	0	1	0	1
21	0	1	0	0	0	0	0	1	0	1	0	0	0	1	0	1
22	0	1	0	0	0	0	0	0	1	0	0	1	0	0	1	0
23	0	1	0	0	0	0	0	0	1	0	0	1	0	1	0	1

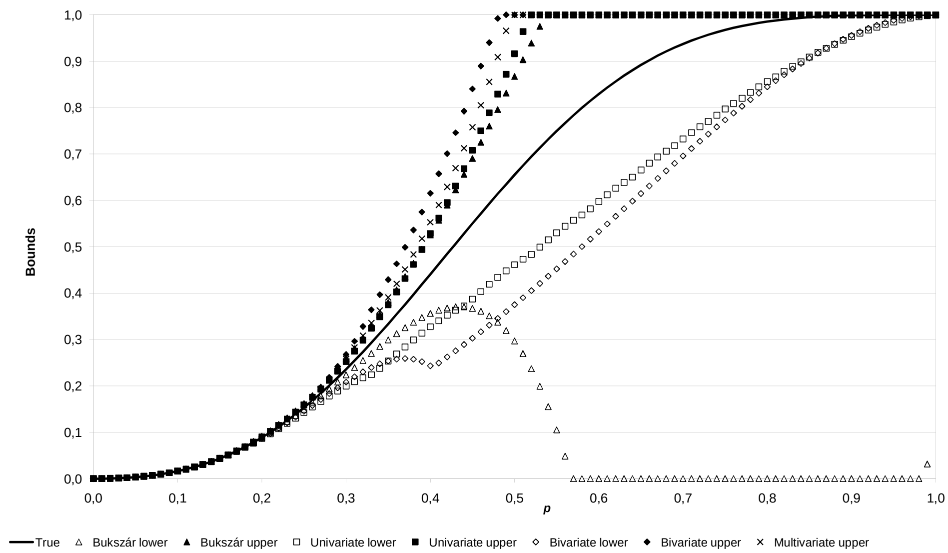

Figure 1: 3rd order bounds on the network reliability

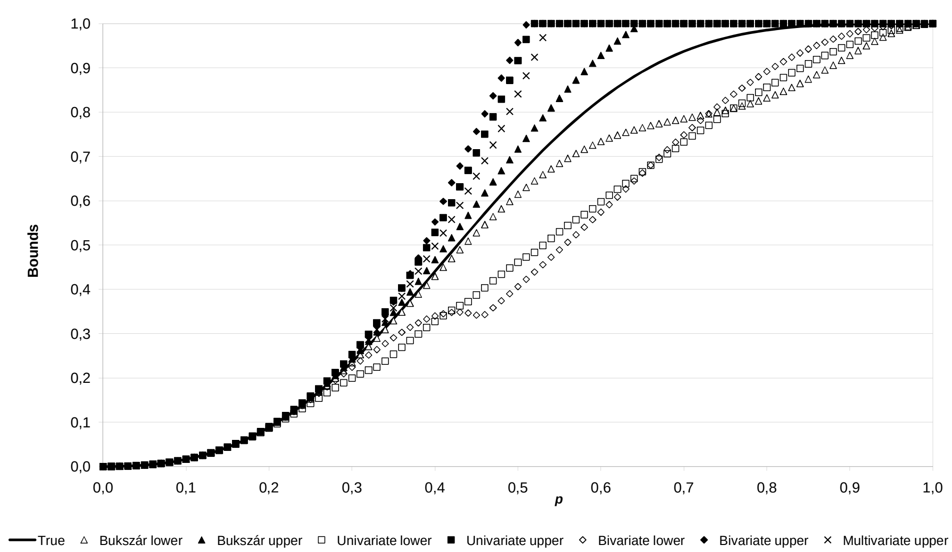

Figure 2: 5th order bounds on the network reliability

Table 11: 3-rd order univariate BMP bounds for the multivariate normal c.d.f.

Dimension	lower bound	CPU time	upper bound	CPU time
6	0.984959	0.02+0.00	0.988391	0.02+0.00
11	0.973471	0.02+0.00	0.982684	0.02+0.00
16	0.962383	0.03+0.00	0.977844	0.03+0.00
21	0.951323	0.06+0.00	0.972245	0.06+0.00
31	0.929253	0.19+0.00	0.961307	0.19+0.00
41	0.907232	0.40+0.00	0.950721	0.40+0.00

Table 12: 3-rd order bivariate and multivariate bounds for the multivariate normal c.d.f.

Dim.	lower bound	CPU time	upper bound	CPU time	low. bound (mult.)	CPU time
6	0.985333	0.02+0.01	0.987825	0.02+0.00	0.985333	0.02+0.00
11	0.972199	0.02+0.01	0.985463	0.02+0.01	0.972711	0.02+0.02
16	0.956843	0.03+0.01	0.986924	0.03+0.00	0.959705	0.03+0.04
21	0.939767	0.06+0.01	0.988308	0.06+0.01	0.946720	0.06+0.08
31	0.902680	0.19+0.04	0.987952	0.19+0.00	0.920451	0.19+0.22
41	0.863515	0.40+0.11	0.987779	0.40+0.01	0.894478	0.40+0.67

5 Tests on randomly generated event systems

In this section we generate test problems in the following very useful and flexible way. The original idea is from Example 2 in Kuai, Alajaji and Takahara (2000).

Let $x_0, x_1, \dots x_{2n}$ be all of the possible outcomes of a sample space with probabilities of occurrence $P(x_0), P(x_1), \dots, P(x_{2n})$, respectively. Any set of n events $\{A_1, ..., A_n\}$ in this probability space can be identified with a matrix $R = (r_{ij})$, where $r_{ij} = 1$ if $x_i \in A_j$, and $r_{ij} = 0$ otherwise. Suppose that outcome x_0 is the event that none of $A_1, ..., A_n$ occurs. Then the probability of the union of the n events can be obtained as

$$P(A_1 \cup \dots \cup A_n) = 1 - P(x_0).$$

As regards the number of events we consider the cases $n = 10, 20, 40$. The probabilities are set as $P(x_1) = \dots = P(x_{2n})$ and the cases $P(x_0) = 0.1, 0.5, 0.9$ are considered. The entries of the matrix R will be generated randomly and independently. For every entry in matrix R the probability that it will be 1 is identical, and this probability will be called the density of the matrix. Three types of matrices will be considered regarding their density, which will be $0.2, 0.5, 0.8$, in particular. The probabilities of the intersection of the events are calculated up to the order five. The numerical implementation of the generation method in C++ can be found at Mádi-Nagy (2009b).

The bounds are summarized below. The notation 'Mx_n_p_d' indicates the problem of n events, with $P(x_0) = p/10$ and with the density of the matrix $d/10$.

Table 13: 3-rd order multitree and hypermultitree bounds for the multivariate normal c.d.f.

Dimension	lower bound	CPU time	upper bound	CPU time
6	0.986001	0.09	0.986808	0.03
11	0.976450	0.10	0.979549	0.13
16	0.966897	0.11	0.972548	0.24
21	0.957343	0.11	0.965775	0.44
31	0.938236	0.11	0.952888	0.84
41	0.919130	0.14	0.940908	1.36

Table 14: 5-th order multitree and hypermultitree bounds for the multivariate normal c.d.f.

Dimension	lower bound	CPU time	upper bound	CPU time
6	0.986298	0.94	0.986323	0.97
11	0.977306	0.99	0.977554	1.06
16	0.968310	1.04	0.968883	1.29
21	0.959312	1.11	0.960324	1.33
31	0.941327	1.21	0.943534	1.75
41	0.923342	1.33	0.927165	2.45

Table 15: 3-rd order bounds.

Problem	Aggr. min	CPU	Aggr. max	CPU	Biv. lower	CPU	Biv. upper	CPU	Mult upper	CPU	Buksz lower	CPU	Buksz upper	CPU
Mx_10_9_2	0.0895	0.00	0.1167	0.00	0.0750	0.03	0.1600	0.01	0.1400	0.04	0.1000	0.00	0.1150	0.00
Mx_10_9_5	0.0964	0.00	0.1210	0.00	0.0937	0.02	0.1600	0.02	0.1690	0.03	0.0250	0.00	0.1250	0.00
Mx_10_9_8	0.0996	0.00	0.1092	0.00	0.1000	0.01	0.1000	0.01	0.1000	0.05	0.1000	0.00	0.1000	0.00
Mx_10_5_2	0.4100	0.00	0.5333	0.00	0.3500	0.02	0.6500	0.02	0.6000	0.40	0.5000	0.00	0.5250	0.00
Mx_10_5_5	0.4735	0.00	0.5650	0.00	0.4625	0.02	0.7964	0.02	0.8417	0.02	0.1000	0.00	0.6500	0.00
Mx_10_5_8	0.4978	0.00	0.5542	0.00	0.5000	0.02	0.5550	0.02	0.5550	0.02	0.2500	0.00	0.5000	0.00
Mx_10_1_2	0.7335	0.00	0.9945	0.00	0.5918	0.02	1.0000	0.02	1.0000	0.04	0.9000	0.00	1.0000	0.00
Mx_10_1_5	0.8577	0.00	1.0000	0.00	0.8550	0.02	1.0000	0.01	1.0000	0.04	0.5400	0.00	1.0000	0.00
Mx_10_1_8	0.8968	0.00	1.0000	0.00	0.9000	0.02	0.9750	0.01	0.9750	0.04	0.0900	0.00	1.0000	0.00
Mx_20_9_2	0.0894	0.00	0.1195	0.00	0.0581	0.06	0.2651	0.01	0.1775	0.18	0.0625	0.01	0.1875	0.00
Mx_20_9_5	0.0976	0.01	0.1375	0.00	0.0889	0.06	0.2758	0.01	0.2915	0.06	0.0000	0.00	0.1975	0.00
Mx_20_9_8	0.0998	0.00	0.1173	0.00	0.1000	0.06	0.1294	0.02	0.1294	0.06	0.0000	0.00	0.1225	0.00
Mx_20_5_2	0.4535	0.00	0.6375	0.01	0.2875	0.05	1.0000	0.02	0.9250	0.33	0.2875	0.00	0.8625	0.00
Mx_20_5_5	0.4894	0.00	0.6648	0.00	0.4875	0.05	1.0000	0.01	1.0000	0.33	0.0000	0.00	0.8250	0.00
Mx_20_5_8	0.4991	0.00	0.5585	0.00	0.5000	0.06	0.6625	0.01	0.6625	0.33	0.0000	0.00	0.5750	0.00
Mx_20_1_2	0.7616	0.00	1.0000	0.00	0.5400	0.06	1.0000	0.01	1.0000	0.33	0.4950	0.00	0.0000	0.00
Mx_20_1_5	0.8672	0.00	1.0000	0.00	0.8393	0.06	1.0000	0.01	1.0000	0.33	0.0000	0.00	1.0000	0.00
Mx_20_1_8	0.8965	0.00	1.0000	0.00	0.9000	0.06	1.0000	0.01	1.0000	0.32	0.0000	0.00	1.0000	0.00
Mx_40_9_2	0.0932	0.00	0.1488	0.00	0.0528	0.09	0.4949	0.00	0.3400	0.42	0.0000	0.01	0.3338	0.00
Mx_40_9_5	0.0987	0.00	0.1439	0.00	0.0925	0.09	0.4108	0.00	0.4108	0.77	0.0000	0.01	0.3413	0.00
Mx_40_9_8	0.0999	0.00	0.1156	0.00	0.1000	0.08	0.1869	0.01	0.2174	0.09	0.0000	0.01	0.1475	0.00
Mx_40_5_2	0.4614	0.00	0.7083	0.00	0.2375	0.08	1.0000	0.01	1.0000	0.76	0.0000	0.01	1.0000	0.00
Mx_40_5_5	0.4927	0.00	0.7168	0.00	0.4471	0.09	1.0000	0.00	1.0000	0.77	0.0000	0.01	1.0000	0.00
Mx_40_5_8	0.4993	0.00	0.6214	0.00	0.4966	0.08	1.0000	0.01	1.0000	0.76	0.0000	0.02	0.8063	0.00
Mx_40_1_2	0.8275	0.00	1.0000	0.00	0.4871	0.08	1.0000	0.00	1.0000	0.77	0.0000	0.01	1.0000	0.00
Mx_40_1_5	0.8868	0.00	1.0000	0.00	0.8016	0.10	1.0000	0.00	1.0000	0.76	0.0000	0.03	1.0000	0.00
Mx_40_1_8	0.8984	0.00	1.0000	0.00	0.8939	0.10	1.0000	0.01	1.0000	0.76	0.0000	0.01	1.0000	0.00

Table 16: 5-th order bounds.

Problem	Aggr. min	CPU	Aggr. max	CPU	Biv. lower	CPU	Biv. upper	CPU	Mult upper	CPU	Buksz lower	CPU	Buksz upper	CPU
Mx_10_9_2	0.0969	0.00	0.1007	0.00	0.0950	0.40	0.1030	0.35	0.1030	0.39	0.1000	0.00	0.1050	0.00
Mx_10_9_5	0.0995	0.00	0.1014	0.00	0.1000	0.39	0.1070	0.35	0.1070	0.39	0.0950	0.00	0.1000	0.00
Mx_10_9_8	0.1000	0.00	0.1005	0.00	0.1000	0.40	0.1000	0.36	0.1000	0.40	0.1000	0.00	0.1000	0.00
Mx_10_5_2	0.4930	0.01	0.5000	0.00	0.4900	0.39	0.5000	0.36	0.5000	0.40	0.5000	0.00	0.5000	0.00
Mx_10_5_5	0.4980	0.01	0.5087	0.01	0.5000	0.36	0.5250	0.36	0.5250	0.39	0.4750	0.00	0.5250	0.00
Mx_10_5_8	0.5000	0.01	0.5024	0.01	0.5000	0.36	0.5000	0.40	0.5000	0.40	0.5000	0.00	0.5000	0.00
Mx_10_1_2	0.8906	0.01	0.9000	0.00	0.8280	0.39	0.9000	0.36	0.9000	0.40	0.9000	0.00	0.9900	0.00
Mx_10_1_5	0.8836	0.00	0.9016	0.01	0.9000	0.40	0.9540	0.36	0.9540	0.40	0.8550	0.00	0.9450	0.00
Mx_10_1_8	0.9000	0.01	0.9013	0.01	0.9000	0.40	0.9000	0.35	0.9000	0.40	0.7650	0.00	0.9000	0.00
Mx_20_9_2	0.0966	0.01	0.1022	0.01	0.0792	2.14	0.1726	0.36	0.1125	4.54	0.0875	0.00	0.1250	0.00
Mx_20_9_5	0.0998	0.01	0.1016	0.03	0.0980	2.13	0.1369	0.36	0.1369	2.14	0.0000	0.00	0.1125	0.00
Mx_20_9_8	0.1000	0.02	0.1004	0.02	0.1000	2.19	0.1000	0.36	0.1000	7.48	0.0875	0.02	0.1000	0.00
Mx_20_5_2	0.4906	0.01	0.5143	0.01	0.3818	2.13	0.9050	0.35	0.5624	4.54	0.4750	0.01	0.6125	0.00
Mx_20_5_5	0.4990	0.01	0.5131	0.03	0.5000	2.15	0.5675	0.36	0.5775	7.48	0.0750	0.00	0.5500	0.00
Mx_20_5_8	0.5000	0.01	0.5010	0.03	0.5000	2.14	0.5000	0.36	0.5022	2.13	0.4500	0.00	0.5000	0.00
Mx_20_1_2	0.8523	0.00	0.9398	0.01	0.7200	2.15	1.0000	0.35	1.0000	7.43	0.7875	0.00	1.0000	0.00
Mx_20_1_5	0.8954	0.01	0.9387	0.02	0.9000	2.16	1.0000	0.36	1.0000	7.45	0.2250	0.00	1.0000	0.00
Mx_20_1_8	0.8999	0.01	0.9066	0.04	0.9000	2.13	0.9000	0.36	0.9000	7.47	0.5625	0.00	0.9000	0.00
Mx_40_9_2	0.0991	0.00	0.1056	0.01	0.0721	12.00	0.3239	0.14	0.1925	37.52	0.0000	0.05	0.2050	0.00
Mx_40_9_5	0.0999	0.01	0.1020	0.01	0.0975	11.88	0.1680	0.14	0.1680	64.40	0.0000	0.05	0.1513	0.00
Mx_40_9_8	0.1000	0.01	0.1001	0.02	0.1000	11.89	0.1000	0.13	0.1000	51.00	0.0000	0.04	0.1000	0.00
Mx_40_5_2	0.4931	0.00	0.5264	0.01	0.3345	12.04	1.0000	0.14	0.8500	46.53	0.0000	0.05	0.9900	0.00
Mx_40_5_5	0.4996	0.01	0.5139	0.01	0.4875	11.96	0.9510	0.14	0.9510	20.45	0.0000	0.05	0.7188	0.00
Mx_40_5_8	0.5000	0.01	0.5015	0.02	0.5000	11.96	0.5108	0.14	0.5193	11.97	0.0000	0.04	0.5063	0.00
Mx_40_1_2	0.8814	0.00	0.9343	0.01	0.6862	11.99	1.0000	0.13	1.0000	60.17	0.0000	0.05	1.0000	0.00
Mx_40_1_5	0.8990	0.01	0.9265	0.01	0.8752	11.89	1.0000	0.13	1.0000	64.16	0.0000	0.05	1.0000	0.00
Mx_40_1_8	0.9000	0.01	0.9029	0.02	0.9000	11.87	0.9221	0.14	0.9221	20.52	0.0000	0.05	0.9113	0.00

6 Conclusions

As regards the quality of the bounds:

- For the normal c.d.f. the Bukszár's bounds are the best, while for the Dirichlet c.d.f the bivariate BMP gave the tightest bounds; the running times for the 3rd order bounds were comparable, however, for the 5th order bounds the running time of both BMP was fairly high (>7 hours), whereas it was less than 3 minutes for Bukszár's bounds,
- For the network reliability the Bukszar's upper bounds and the lower bounds of the univariate BMP are the tightest third order bounds, while among the fifth order bounds the Bukszar's lower and upper bounds and the bivariate lower bounds yield the best approximation for most values of p.
- In case of randomly generated event systems any type of bounds can be the best, depending on the problem. However, in case of greater number of events the Bukszár's bounds with the current heavy (hyper)multitree finding algorithm often provide the trivial bounds only.

The running times are usually within a reasonable range, the only exception is the c.d.f. approximation, where the calculation of binomial moments can take huge amount of time. Therefore, for this application the graph-based methods are recommended.

References

[1] Bonferroni, C. E. 1937. Teoria statistica delle classi e calcolo delle probabilitá. *Volume in onore di Riccardo Dalla Volta* Universitá di Firenze, Florence, Italy, 1–62.

[2] Boole, G. 1854. *Laws of Thought* American reprint of 1854 editon, Dover, New York.

[3] Bukszár, J. 2001. Upper bounds for the probability of a union by multitrees. *Adv. in Appl. Probab.* **33**(2) 437-452.

[4] Bukszár, J. and A. Prékopa. 2001. Probability Bounds with Cherry Trees. *Mathematics of Operations Research* **26**(1) 174–192.

[5] Bukszár, J. and T. Szántai. 2002. Probability bounds given by hypercherry trees. *Optimization Methods and Software* **17** 409-422.

[6] Bukszár, J. 2003. Hypermultitrees and Sharp Bonferroni Inequalities. *Mathematical Inequalities & Applications* **6**(4) 727-743.

[7] Costigan, T.M. 1996. Combination Setwise-Bonferroni-Type Bounds. *Naval Research Logistics* **43** 59-77.

[8] Dohmen, K. 2003. *Improved Bonferroni inequalities via abstract tubes.* Inequalities and identities of inclusion-exclusion type. Lecture Notes in Mathematics, 1826. Springer-Verlag, Berlin

[9] Galambos, J. and I. Simonelli. 1996. *Bonferroni-Type Inequalities with Applications.* Springer-Verlag, Berlin/New York

[10] Genz, A. 1992. Numerical Computation of the Multivariate Normal Probabilities. *Journal of Computational and Graphical Statistics* **1** 141-150.

[11] Gouda, A. A. and T. Szántai. 2010. On numerical calculation of probabilities according to Dirichlet distribution. *Annals of Operations Research* **177**(1) 185-200.

[12] Hoppe, F. M. and E. Seneta. 1990. Bonferroni-type inequalities and the methods of indicators and polynomials. *Adv. in Appl. Probab.* **22**(1) 241-246.

[13] Hunter, D. 1976. An upper bound for the probability of a union. *J. Appl. Prob.* **13** 597–603.

[14] Kuai, H., F. Alajaji and G. Takahara (2000). A Lower Bound on the Probability of a Finite Union of Events, *Discrete Mathematics* **215**(2) 147-158.

[15] Mádi-Nagy, G. 2005. A method to find the best bounds in a multivariate discrete moment problem if the basis structure is given. *Studia Scientiarum Mathematicarum Hungarica* **42**(2) 207-226.

[16] Mádi-Nagy, G. 2009. On Multivariate Discrete Moment Problems: Generalization of the Bivariate Min Algorithm for Higher Dimensions. *SIAM Journal on Optimization* **19**(4) 1781-1806.

[17] Mádi-Nagy, G. 2009b. Numerical applications.
http://www.math.bme.hu/ gnagy/appl/applications.htm

[18] Mádi-Nagy, G. and A. Prékopa. 2004. On Multivariate Discrete Moment Problems and their Applications to Bounding Expectations and Probabilities. *Mathematics of Operations Research* **29** (2) 229-258.

[19] Prékopa, A. 1990. Sharp bounds on probabilities using linear programming. *Operations Research* **38** 227-239.

[20] Prékopa, A. 1990b. The discrete moment problem and linear programming. *Discrete Applied Mathematics* **27** 235-254.

[21] Prékopa, A. 1992. Inequalities on Expectations Based on the Knowledge of Multivariate Moments. M. Shaked and Y.L. Tong, eds., *Stochastic Inequalities.* Institute of Mathematical Statistics, Lecture Notes — Monograph Series, Vol 22, 309–331.

[22] Prékopa, A. 1995. *Stochastic Programming.* Kluwer Academic Publishers, Dodrecht, Boston.

[23] Prékopa, A. 1998. Bounds on probabilities and expectations using multivariate moments of discrete distributions. *Studia Sci. Math. Hungar.* **34**(1-3) 349-378.

[24] Recsei, E. and E. Seneta. 1987. Bonferroni-type inequalities. *Adv. in Appl. Probab.* **19**(2) 508-511.

[25] Tomescu, I. 1986. Hypertrees and Bonferroni inequalities. *J. Combinatorial Theory Ser. B* **41**(2) 209-217.

[26] Worsley, K. J. 1982. An improved Bonferroni inequality and applications. *Biometrika* **69** 197-302.