Mathematical
Inequalities
& Applications
Volume 6, Number 4 (2003), 727–743

HYPERMULTITREES AND SHARP BONFERRONI INEQUALITIES

JÓZSEF BUKSZÁR

(*communicated by A. Prekopa*)

Abstract. The concept of (h,m)-hypermultitree (which is a special hypergraph) is introduced to present Bonferroni-type inequalities which are equalities for some families of sets. The main theorem is a common generalization of the earlier results of Hunter, Worsley, Tomescu and recent results of Prékopa and the author. The new bounds are based on significantly fewer probabilities than the same order Bonferroni bounds. This and some other properties explain the very high efficiency of the new bounds in many applications, e.g., estimate the values of multivariate normal distribution functions, which is demonstrated in the end of the paper.

1. Introduction

Let $A_1,\ldots,A_n$ be arbitrary probability events. Our purpose is to present lower and upper bounds for the probability $P(A_1\cup\ldots\cup A_n)$ based on some of the terms $P(A_{k_1}\cap\ldots\cap A_{k_i})$ $(1\leqslant k_1<\ldots<k_i\leqslant n$, $i=1,\ldots,d)$, where $d\geqslant 2$ is a prescribed integer. For a pair of integers h,m with $h+m+1=d$ we introduce a class of hypergraphs on n vertices, called (h,m)-hipermultitrees, and assign a lower (upper) bound to each hypergraph of the class for odd (even) h. We obtain the Tomescu bounds [13] for $m=1$ and the multitree bound [3] for $h=0$. The Hunter-Worsley bound [8] and [14] is a special Tomescu and multitree bound at the same time. We recall the concept of m-multitree and some related definitions and results.

DEFINITION 1. Let m be a positive integer. An m*-multicherry* is a hypergraph of the form $(V,\mathcal{E}_2,\ldots,\mathcal{E}_{m+1})$, 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 $\mathcal{E}_i$ is the set of all subsets of $\{v_1,\ldots,v_{m+1}\}$ containing i vertices with v_{m+1} included, i.e., $\mathcal{E}_i=\{H\mid\ v_{m+1}\in H\subset\{v_1,\ldots,v_{m+1}\},|H|=i\}$. 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 $(\{v_1,\ldots,v_m\},v_{m+1})$.

Note that a 1-multicherry is a single edge together with its incident vertices.

DEFINITION 2. Let m be a positive integer. An m*-multitree* is a hypergraph of the form $(V,\mathcal{E}_2,\ldots,\mathcal{E}_{m+1})$, where V is the set of vertices and $\mathcal{E}_i$'s are sets of hyperedges containing i vertices. An m-multitree is recursively defined by the following two rules.

Mathematics subject classification (2000): 60C05.
Key words and phrases: Probability bounds, hypergraph, multivariate normal distribution.
Partially supported by OTKA T032369.

Paper MIA-06-66

(i) The smallest m-multitree $\Delta = (V, \mathcal{E}_2, \dots, \mathcal{E}_{m+1})$ has m vertices and $\mathcal{E}_i$ is the family of all subsets of V containing i vertices (here $\mathcal{E}_{m+1} = \emptyset$).

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

A 3-multitree $\Delta = (V, \mathcal{E}_2, \mathcal{E}_3, \mathcal{E}_4)$ is illustrated in Figure 1. Given 1, 2, 3 and the hyperedges $\{1,2\}$, $\{1,3\}$, $\{2,3\}$, $\{1,2,3\}$, we subsequently adjoin the 3-multicherries $(\{1,2,3\},4)$, $(\{1,3,4\},5)$ and $(\{2,3,5\},6)$ as shown in the figure. The edges of a 3-multicherry are drawn with the same line-style. The vertices and hyperedges of Δ are $V = \{1,2,3,4,5,6\}$, $\mathcal{E}_2 = \{\{1,2\}, \{1,3\}, \{1,4\}, \{1,5\}, \{2,3\}, \{2,4\}, \{2,6\}, \{3,4\}, \{3,5\}, \{3,6\}, \{4,5\}, \{5,6\}\}$, $\mathcal{E}_3 = \{\{1,2,3\}, \{1,2,4\}, \{1,3,4\}, \{2,3,4\}, \{1,3,5\}, \{1,4,5\}, \{3,4,5\}, \{2,3,6\}, \{2,5,6\}, \{3,5,6\}\}$, $\mathcal{E}_4 = \{\{1,2,3,4\}, \{1,3,4,5\}, \{2,3,5,6\}\}$.

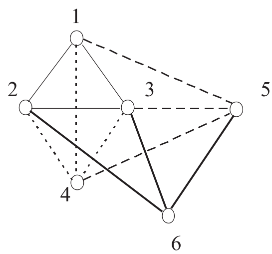

Figure 1

Note that a 1-multitree is a usual tree.

REMARK 3. If $\Delta = (V, \mathcal{E}_2, \dots, \mathcal{E}_{m+1})$ is an m-multitree with $|V| = n$, then $|\mathcal{E}_i| = \binom{m}{i} + \binom{m}{i-1}(n-m)$ for all $i = 2, \dots, m+1$.

DEFINITION 4. Let $A_1, \dots, A_n$ be arbitrary events and let these events be assigned to the vertices of an m-multitree $\Delta = (V, \mathcal{E}_2, \dots, \mathcal{E}_{m+1})$. Then the *weight of the m-multitree* is defined as

$$w(\Delta) = \sum_{\{l_1,l_2\} \in \mathcal{E}_2} P(A_{l_1} \cap A_{l_2}) - \sum_{\{l_1,l_2,l_3\} \in \mathcal{E}_3} P(A_{l_1} \cap A_{l_2} \cap A_{l_3}) + \dots$$

$$\dots + (-1)^{m+1} \sum_{\{l_1,\dots,l_{m+1}\} \in \mathcal{E}_{m+1}} P(A_{l_1} \cap \dots \cap A_{l_{m+1}}).$$

THEOREM 5. *Let $A_1, \dots, A_n$ be arbitrary events and let $\Delta = (V, \mathcal{E}_2, \dots, \mathcal{E}_{m+1})$ be an arbitrary m-multitree with $V = \{1, \dots, n\}$. Then*

$$P(A_1 \cup \dots \cup A_n) \leqslant S_1 - w(\Delta), \tag{1}$$

where $S_1 = \sum_{i=1}^{n} P(A_i)$.

Proof. Without loss of generality we can assume that Δ is obtained by the following recursion. We start from the m-multitree $\Delta^{(m)} = (V^{(m)}, \mathcal{E}_2^{(m)}, \ldots, \mathcal{E}_{m+1}^{(m)})$, where $V^{(m)} = \{1, \ldots, m\}$, $\mathcal{E}_i^{(m)}$ ($i = 2, \ldots, m$) is the set of all subsets of $V^{(m)}$ containing i elements and $\mathcal{E}_{m+1}^{(m)} = \emptyset$. Then, we construct the sequence of m-multitrees $\Delta^{(m+1)}$, $\Delta^{(m+2)}, \ldots$, $\Delta^{(n)} = \Delta$ in the way that we obtain $\Delta^{(j)} = (V^{(j)}, \mathcal{E}_2^{(j)}, \ldots, \mathcal{E}_{m+1}^{(j)})$ from $\Delta^{(j-1)} = (V^{(j-1)}, \mathcal{E}_2^{(j-1)}, \ldots, \mathcal{E}_{m+1}^{(j-1)})$ by adjoining the m-multicherry $(\{i_1^{(j)}, \ldots, i_m^{(j)}\}, j)$, where $1 \leqslant i_1^{(j)} < \ldots < i_m^{(j)} \leqslant j-1$. The recursion gives that

$$\mathcal{E}_{m+1} = \left(\{i_1^{(j)}, \ldots, i_m^{(j)}, j\} \mid j = m+1, \ldots, n \right)$$

and

$$\mathcal{E}_i = \bigcup_{j=m+1}^{n} \left\{ K \mid j \in K \subseteq \{i_1^{(j)}, \ldots, i_m^{(j)}, j\},\ |K| = i \right\} \cup \{K \mid K \subseteq \{1, \ldots, m\},\ |K| = i\},$$

for all $i = 2, \ldots, m$.

$$\begin{aligned}
P(A_1 \cup \ldots \cup A_n) &= P(A_1 \cup \ldots \cup A_{n-1}) + P(A_n) - P((A_1 \cup \ldots \cup A_{n-1}) \cap A_n) \\
&= P(A_1 \cup \ldots \cup A_{n-2}) + P(A_{n-1}) - P((A_1 \cup \ldots \cup A_{n-2}) \cap A_{n-1}) + P(A_n) \\
&\quad - P((A_1 \cup \ldots \cup A_{n-1}) \cap A_n) = \ldots = S_1 - \sum_{j=2}^{n} P((A_1 \cup \ldots \cup A_{j-1}) \cap A_j) \\
&= S_1 - \left(\sum_{j=2}^{m} P((A_1 \cup \ldots \cup A_{j-1}) \cap A_j) + \sum_{j=m+1}^{n} P((A_1 \cup \ldots \cup A_{j-1}) \cap A_j) \right) \\
&\leqslant S_1 - \left(\sum_{j=2}^{m} P((A_1 \cup \ldots \cup A_{j-1}) \cap A_j) + \sum_{j=m+1}^{n} P((A_{i_1^{(j)}} \cup \ldots \cup A_{i_m^{(j)}}) \cap A_j) \right) \\
&= S_1 - \left(\sum_{j=2}^{m} P\left((A_1 \cap A_j) \cup \ldots \cup (A_{j-1} \cap A_j)\right) + \sum_{j=m+1}^{n} P((A_{i_1^{(j)}} \cap A_j) \cup \ldots \cup (A_{i_m^{(j)}} \cap A_j)) \right).
\end{aligned}$$

Applying the inclusion-exclusion formula for $P\left((A_1 \cap A_j) \cup \ldots \cup (A_{j-1} \cap A_j)\right)$ and $P((A_{i_1^{(j)}} \cap A_j) \cup \ldots \cup (A_{i_m^{(j)}} \cap A_j))$ we have

$$\begin{aligned}
&S_1 - \left(\sum_{j=2}^{m} P\left((A_1 \cap A_j) \cup \ldots \cup (A_{j-1} \cap A_j)\right) + \sum_{j=m+1}^{n} P((A_{i_1^{(j)}} \cap A_j) \cup \ldots \cup (A_{i_m^{(j)}} \cap A_j)) \right) \\
&= S_1 - \sum_{j=2}^{m} \Bigg[\sum_{1 \leqslant h_1 \leqslant j-1} P(A_{h_1} \cap A_j) - \sum_{1 \leqslant h_1 < h_2 \leqslant j-1} P(A_{h_1} \cap A_{h_2} \cap A_j) + \ldots \\
&\ldots + (-1)^{j-1} \sum_{1 \leqslant h_1 < \ldots < h_{j-2} \leqslant j-1} (A_{h_1} \cap \ldots \cap A_{h_{j-2}} \cap A_j) + (-1)^j P(A_1 \cap \ldots \cap A_{j-1} \cap A_j) \Bigg]
\end{aligned}$$

$$- \sum_{j=m+1}^{n} \left(\sum_{\{k_1\} \subseteq \{i_1^{(j)},\ldots,i_m^{(j)}\}} P(A_{k_1} \cap A_j) - \sum_{\{k_1,k_2\} \subseteq \{i_1^{(j)},\ldots,i_m^{(j)}\}} P(A_{k_1} \cap A_{k_2} \cap A_j) + \ldots \right.$$

$$\left. \ldots + (-1)^m \sum_{\{k_1,\ldots,k_{m-1}\} \subseteq \{i_1^{(j)},\ldots,i_m^{(j)}\}} P(A_{k_1} \cap \ldots \cap A_{k_{m-1}} \cap A_j) + (-1)^{m+1} P(A_{i_1^{(j)}} \cap \ldots \cap A_{i_m^{(j)}} \cap A_j) \right)$$

$$\stackrel{(*)}{=} S_1 - \sum_{\{l_1,l_2\} \in \mathcal{E}_2} P(A_{l_1} \cap A_{l_2}) + \sum_{\{l_1,l_2,l_3\} \in \mathcal{E}_3} P(A_{l_1} \cap A_{l_2} \cap A_{l_3}) - \ldots$$

$$\ldots + (-1)^m \sum_{\{l_1,\ldots,l_{m+1}\} \in \mathcal{E}_{m+1}} P(A_{l_1} \cap \ldots \cap A_{l_{m+1}}) = S_1 - w(\Delta).$$

$(*)$ The summation of the terms in the square brackets correspond to the starting smallest m-multitree and the rest of the summands arise from the consecutive steps of the recursion. □

2. (h,m)-hypermultitrees

DEFINITION 6. Let $h \geqslant 0$ and $m \geqslant 1$ be arbitrary integers. An (h,m)-*hypermultitree* is a hypergraph of the form $(V, {}_h\mathcal{E}_2, \ldots, {}_h\mathcal{E}_{m+1})$, where V is the set of vertices and ${}_h\mathcal{E}_i$'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 $\Delta = (V, {}_h\mathcal{E}_2, \ldots, {}_h\mathcal{E}_{m+1})$ has $h+m$ vertices and ${}_h\mathcal{E}_i$ consists of all subsets of V containing $h+i$ vertices (here ${}_h\mathcal{E}_{m+1} = \emptyset$).

(iii) From an (h,m)-hypermultitree $\Delta = (V, {}_h\mathcal{E}_2, \ldots, {}_h\mathcal{E}_{m+1})$ we can obtain a new (h,m)-hypermultitree in the following manner. Let $\Gamma = (V, {}_{h-1}\mathcal{E}_2^*, \ldots, {}_{h-1}\mathcal{E}_{m+1}^*)$ be an arbitrary $(h-1,m)$-hypermultitree with the same set of vertices as in Δ. By adjoining a new vertex v to Δ and the hyperedges of Γ extended by v, we obtain the new (h,m)-hypermultitree $\Delta' = (V', {}_h\mathcal{E}_2', \ldots, {}_h\mathcal{E}_{m+1}')$, i.e.,

$$V' = V \cup \{v\} \qquad {}_h\mathcal{E}_i' = {}_h\mathcal{E}_i \cup \{E \cup \{v\} \mid E \in {}_{h-1}\mathcal{E}_i^*\}.$$

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

EXAMPLES. $\Delta = (V, {}_1\mathcal{E}_2, {}_1\mathcal{E}_3)$ with ${}_1\mathcal{E}_2 = \{\{1,2,3\}\}$ and ${}_1\mathcal{E}_3 = \emptyset$ is an $(1,2)$-hypermultitree by (ii). From Δ we can obtain the $(1,2)$-hypermultitree $\Delta' = (V', {}_1\mathcal{E}_2', {}_1\mathcal{E}_3')$ on the basis of $(0,2)$-hypermultitree $\Gamma_1 = (\{1,2,3\}, \{\{1,2\},\{1,3\}, \{2,3\}\}, \{\{1,2,3\}\})$ shown in Figure 2 by adjoining vertex 4 by (iii), where $V' = \{1,2,3,4\}$, ${}_1\mathcal{E}_2' = \{\{1,2,3\}, \{1,2,4\}, \{1,3,4\}, \{2,3,4\}\}$, ${}_1\mathcal{E}_3' = \{\{1,2,3,4\}\}$. From Δ' we obtain $\Delta'' = (V'', {}_1\mathcal{E}_2'', {}_1\mathcal{E}_3'')$ on the basis of Γ_2, where $V'' = \{1,2,3,4,5\}$, ${}_1\mathcal{E}_2'' = {}_1\mathcal{E}_2' \cup \{\{1,3,5\}, \{1,4,5\}, \{2,3,5\}, \{2,4,5\}, \{3,4,5\}\}$, ${}_1\mathcal{E}_3'' = {}_1\mathcal{E}_3' \cup \{\{1,3,4,5\}, \{2,3,4,5\}\}$. Finally, from Δ'' we obtain $\Delta''' = (V''', {}_1\mathcal{E}_2''', {}_1\mathcal{E}_3''')$ on the basis of Γ_3, where $V''' = \{1,2,3,4,5,6\}$, ${}_1\mathcal{E}_2''' = {}_1\mathcal{E}_2'' \cup \{\{1,3,6\}, \{1,4,6\}, \{1,5,6\}, \{2,3,6\}, \{2,4,6\}, \{3,5,6\}, \{4,5,6\}\}$, ${}_1\mathcal{E}_3''' = {}_1\mathcal{E}_3'' \cup \{\{1,3,5,6\}, \{1,4,5,6\}, \{2,3,4,6\}\}$.

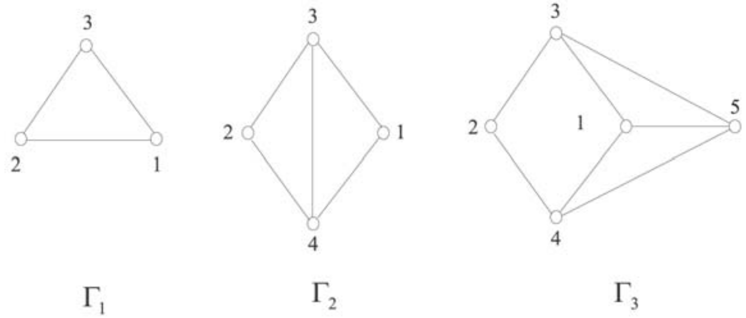

Figure 2

LEMMA 7. *Let $\Delta = (V, {}_h\mathcal{E}_2, \ldots, {}_h\mathcal{E}_{m+1})$ be an arbitrary (h,m)-hypermultitree with n vertices. Then*

$$|{}_h\mathcal{E}_i| = \sum_{k=0}^{h+1} \binom{m}{i-1+k}\binom{n-m}{h+1-k}.$$

Proof. We prove by induction on h. If $h=0$, i.e., Δ is an m-multitree, then the number of hyperedges containing i vertices is $\binom{m}{i-1}(n-m)+\binom{m}{i}$ indeed, according to Remark 3. Assume that the assertion holds if we replace h by $h-1$. In the recursion producing Δ we start from the (h,m)-hypermultitree containing $h+m$ vertices. In a step of the recursion producing Δ we adjoin a new vertex to l ($l = m+h, \ldots, n-1$) vertices and

$$\sum_{k=0}^{h} \binom{m}{i-1+k}\binom{l-m}{h-k}$$

new hyperedges containing $h+i$ vertices because of the induction on h. Consequently, the number of hyperedges containing $h+i$ vertices in Δ is

$$\binom{m+h}{h+i} + \sum_{l=m+h}^{n-1}\sum_{k=0}^{h} \binom{m}{i-1+k}\binom{l-m}{h-k}. \tag{2}$$

Applying the formula

$$\binom{m+h}{h+i} = \binom{m}{h+i} + \sum_{k=1}^{h} \binom{m}{i-1+k}\binom{h}{h+1-k}$$

and interchanging the sums give that

$$\binom{m}{h+i} + \sum_{k=1}^{h} \binom{m}{i-1+k}\binom{h}{h+1-k} + \sum_{k=0}^{h}\sum_{l=m+h}^{n-1} \binom{m}{i-1+k}\binom{l-m}{h-k}$$

$$= \binom{m}{h+i} + \sum_{k=0}^{h} \binom{m}{i-1+k} \left(\binom{h}{h+1-k} + \sum_{l=m+h}^{n-1} \binom{l-m}{h-k} \right).$$

Using the well-known identity

$$\binom{h}{h+1-k} = \sum_{l=m+h-k}^{m+h-1} \binom{l-m}{h-k},$$

we obtain that

$$\binom{m}{h+i} + \sum_{k=0}^{h} \left(\binom{m}{i-1+k} \sum_{l=m+h-k}^{n-1} \binom{l-m}{h-k} \right) = \sum_{k=0}^{h+1} \binom{m}{i-1+k} \binom{n-m}{h+1-k},$$

where the above identity was used again in the form

$$\sum_{l=m+h-k}^{n-1} \binom{l-m}{h-k} = \binom{n-m}{h+1-k}.$$

□

LEMMA 8. *Let g, h, m and n be integers with $0 \leqslant g \leqslant h$, $1 \leqslant m$ and $g+m \leqslant n$. Let $\theta = (V, {}_g\mathcal{E}_2, \ldots, {}_g\mathcal{E}_{m+1})$ be a (g,m)-hypermultitree with $V = \{1, \ldots, n\}$. Assume that the vertices in θ are labeled in increasing order according to a recursion producing θ. If*

$${}_h\mathcal{E}_i = \bigcup_{\{k_1,\ldots,k_{g+i}\} \in {}_g\mathcal{E}_i} \mathcal{F}_{\{k_1,\ldots,k_{g+i}\}},$$

where $\mathcal{F}_{\{k_1,\ldots,k_{g+i}\}} = \{H \cup \{k_1, \ldots, k_{g+i}\} \mid H \subset \{1+\max\{k_1, \ldots, k_{g+i}\}, \ldots, n+h-g\}, |H| = h-g\}$, then $\Delta = (V', {}_h\mathcal{E}_2, \ldots, {}_h\mathcal{E}_{m+1})$ is an (h,m)-hypermultitree with $V' = \{1, \ldots, n+h-g\}$.

Proof. Let $0 \leqslant g$ and $1 \leqslant m$ be fixed numbers. We use induction on both h and n. If $h = g$, then the assertion of the lemma is trivial, because Δ and θ are the same. Let h be an arbitrary integer with $g+1 \leqslant h$, and assume that the assertion of the lemma holds replacing h by $h-1$. Set $n = g+m$. Then $\theta = (V, {}_g\mathcal{E}_2, \ldots, {}_g\mathcal{E}_{m+1})$ is the smallest (g,m)-hypermultitree, hence the hyperedges in ${}_g\mathcal{E}_i$ are all subsets of $\{1, \ldots, m+g\}$ containing $g+i$ vertices, where $i = 2, \ldots, m+1$. From θ we obtain the smallest (h,m)-hypermultitree Δ in the way described in the theorem.

Now fix n, and assume that the assertion of the lemma holds with the fixed h replacing n by $n-1$. Let $\theta = (V, {}_g\mathcal{E}_2, \ldots, {}_g\mathcal{E}_{m+1})$ be a (g,m)-hypermultitree with $V = \{1, \ldots, n\}$ and assume that the vertices in θ are labeled in increasing order according to a recursion producing θ. Let Δ be the hypergraph constructed from θ in the way described in the lemma. We have to prove that Δ is an (h,m)-hypermultitree. Let $\theta^* = (V^*, {}_g\mathcal{E}_2^*, \ldots, {}_g\mathcal{E}_{m+1}^*)$ be the (g,m)-hypermultitree obtained from θ by deleting vertex n and the hyperedges incident to n. (Because of the fact that the vertices in θ are labeled in increasing order according to a recursion producing θ, θ^*

is a (g,m)-hypermultitree indeed.) By induction on n, we can construct an (h,m)-hypermultitree $\Delta^* = (V^*, {}_h\mathcal{E}_2^*, \dots, {}_h\mathcal{E}_{m+1}^*)$ with $V^* = \{1,\dots,n-1+h-g\}$ from θ^* in the way described in the lemma. By induction on h, we can construct an $(h-1,m)$-hypermultitree $\Gamma = (\widehat{V}, {}_{h-1}\widehat{\mathcal{E}}_2, \dots, {}_{h-1}\widehat{\mathcal{E}}_{m+1})$ with $\widehat{V} = \{1,\dots,n+h-1-g\}$ from θ in the way described in the lemma. Note that $V^* = \widehat{V}$. We state that we obtain Δ from Δ^* by adjoining vertex $n+h-g$ and the hyperedges in Γ extended by $n+h-g$. It follows directly from the following assertions.

(i) The hyperedges in Δ not containing vertex $n+h-g$ are the hyperedges in Δ^*.

(ii) The hyperedges in Δ containing vertex $n+h-g$ can be obtained by extending the hyperedges in Γ by vertex $n+h-g$.

Assertion (i) follows from the fact that for a hyperedge $\{k_1,\dots,k_{g+i-1},n\}$ in $\theta\backslash\theta^*$, there is a unique set H with $H \subset \{n+1,\dots,n+h-g\}$ and $|H| = h-g$, namely $H = \{n+1,\dots,n+h-g\}$, which contains vertex $n+h-g$.

Assertion (ii) follows from the fact that the hyperedges in Γ
${}_{h-1}\widehat{\mathcal{E}}_i = \bigcup_{\{f_1,\dots,f_{g+i}\}\in {}_g\mathcal{E}_i} \{H\cup\{f_1,\dots,f_{g+i}\} \mid H \subset \{1+\max\{f_1,\dots,f_{g+i}\},\dots,n+h-1-g\},\ |H| = h-1-g\}$ extended by vertex $n+h-g$ are the elements of $\bigcup_{\{f_1,\dots,f_{g+i}\}\in {}_g\mathcal{E}_i} \{H\cup\{f_1,\dots,f_{g+i}\} \mid n+h-g \in H \subset \{1+\max\{f_1,\dots,f_{g+i}\},\dots,n+h-g\},\ |H| = h-g\}$, which are the hyperedges in ${}_h\mathcal{E}_i$ (of Δ) containing vertex $n+h-g$. Assertions (i) and (ii) imply that Δ is really an (h,m)-hypermultitree. $\square$

DEFINITION 9. Let Δ be a hypergraph of the form $\Delta = (V, {}_h\mathcal{E}_2, \dots, {}_h\mathcal{E}_{m+1})$, where V is the set of vertices and ${}_h\mathcal{E}_i$'s are sets of hyperedges containing $h+i$ vertices for all $i = 2,\dots,m+1$. Δ is called an (h,m)-*hypermultistar* if V has a subset C containing m vertices with

$$ {}_h\mathcal{E}_i = \{I \cup H \mid I \subset C,\ H \subset V\backslash C,\ |H| \leqslant h+1,\ |H| + |I| = h+i\} $$

for all $i = 2,\dots,m+1$. The vertices in C are called *central vertices.*

PROPOSITION 10. *(h,m)-hypermultistars are (h,m)-hypermultitrees.*

Proof. A $(0,m)$-hypermultistar is a $(0,m)$-hypermultitree, i.e., an m-multitree. In fact, a $(0,m)$-hypermultistar can be obtained by the following recursion. We start from the m-multitree, where the vertex set is C and the i element hyperedges are all the i element subsets of C (here $i \geqslant 2$). In all other steps a new vertex and an m-multicherry is adjoined, where the non-dominating vertices of the m-multicherry are the central vertices and its dominating vertex (the new vertex) is a non-central vertex not taken previously. Note that the sets of vertices containing at most one non-central vertex are the hyperedges of a $(0,m)$-hypermultistar.

Let ${}_h\Delta_m^n$ be an (h,m)-hypermultistar with vertex set $\{1,\dots,n\}$ the central vertices of which are labeled by $1,\dots,m$. Then ${}_h\Delta_m^n$ can be obtained by extending the $(0,m)$-hypermultistar ${}_0\Delta_m^{n-h}$, as described in the previous lemma. In fact, let $2 \leqslant i \leqslant m+1$ and let $\{v_1,\dots,v_{h+i}\}$ ($v_1 < \dots < v_{h+i}$) be a hyperedge in ${}_h\Delta_m^n$. Since there are at most $h+1$ non-central vertices in a hyperedge in an (h,m)-hypermultistar, we obtain that $v_1 < \ldots < v_{i-1} \leqslant m$ are central vertices. On the other hand $v_i < \ldots < v_{h+i} \leqslant n$, whence we obtain that $v_i \leqslant n-h$, i.e., that v_i is a vertex in ${}_0\Delta_m^{n-h}$. Since $\{v_1,\ldots,v_i\}$ has at most one non-central vertex, it is a hyperedge in ${}_0\Delta_m^{n-h}$. Therefore, the hyperedge $\{v_1,\ldots,v_{h+i}\}$ in ${}_h\Delta_m^n$ can be obtained as the extension of $\{v_1,\ldots,v_i\}$, as described in the previous lemma.

Because of the fact that an (h,m)-hypermultistar can be obtained as an extension of a $(0,m)$-hypermultistar, which is a $(0,m)$-hypermultitree, Lemma 8 yields that (h,m)-hypermultistars are (h,m)-hypermultitrees. □

3. Hypermultitrees and Bonferroni type inequalities

In this section we present bounds on the probability of the union of finite number of events by means of hypermultitrees.

DEFINITION 11. Let $A_1,\ldots,A_n$ be arbitrary events and let these events be assigned to the vertices of an (h,m)-hypermultitree $\Delta = (V, {}_h\mathcal{E}_2,\ldots, {}_h\mathcal{E}_{m+1})$. Then the *weight of the (h,m)-hypermultitree* is defined as

$$w(\Delta) = \sum_{\{i_1,\ldots,i_{h+2}\}\in {}_h\mathcal{E}_2} P(A_{i_1}\cap\ldots\cap A_{i_{h+2}}) - \sum_{\{i_1,\ldots,i_{h+3}\}\in {}_h\mathcal{E}_3} P(A_{i_1}\cap\ldots\cap A_{i_{h+3}}) + \ldots$$

$$\ldots + (-1)^{m+1} \sum_{\{i_1,\ldots,i_{h+m+1}\}\in {}_h\mathcal{E}_{m+1}} P(A_{i_1}\cap\ldots\cap A_{i_{h+m+1}})$$

THEOREM 12. *Let $A_1,\ldots,A_n$ be arbitrary events and let $\Delta = (V, {}_h\mathcal{E}_2,\ldots, {}_h\mathcal{E}_{m+1})$ be an arbitrary (h,m)-hypermultitree. The following inequalities hold.*

(i) If h is even, then

$$P\left(\bigcup_{i=1}^n A_i\right) \leqslant \sum_{k=1}^{h+1} (-1)^{k-1} S_k - w(\Delta), \tag{3}$$

(ii) if h is odd, then

$$P\left(\bigcup_{i=1}^n A_i\right) \geqslant \sum_{k=1}^{h+1} (-1)^{k-1} S_k + w(\Delta), \tag{4}$$

where $S_k = \sum_{1\leqslant i_1<\ldots<i_k\leqslant n} P(A_{i_1}\cap\ldots\cap A_{i_k})$.

Proof. We prove (i) and (ii) simultaneously by an induction on h. According to Theorem 5, we obtain the m-multitree bound in the inequality in (3) with $h=0$. Let $h\geqslant 1$ and assume that the inequalities in (3) and (4) hold whenever h is replaced with a smaller number. Assume first that h is odd. Now we prove by induction on n that (4) holds with this h. With $n=h+m$ the inequality in (4) is the inclusion-exclusion formula. Let $n\geqslant h+m+1$. Without loss of generality we can assume that vertex

n was adjoined lastly to obtain Δ'. Let $\Delta = (V, {}_h\mathcal{E}_2, \ldots, {}_h\mathcal{E}_{m+1})$ be the (h,m)-hypermultitree and let $\Gamma = (V, {}_{h-1}\mathcal{E}_2^*, \ldots, {}_{h-1}\mathcal{E}_{m+1}^*)$ be the $(h-1,m)$-hypermultitree by which $\Delta' = (V', {}_h\mathcal{E}_2', \ldots, {}_h\mathcal{E}_{m+1}')$ was obtained in the last step of the recursion producing Δ'. By induction on n, the inequality in (4) holds with the events $A_1, \ldots, A_{n-1}$ and Δ:

$$P\left(\bigcup_{i=1}^{n-1} A_i\right) \geqslant \sum_{k=1}^{h+1} (-1)^{k-1} \sum_{\{i_1,\ldots,i_k\}\subset\{1,\ldots,n-1\}} P(A_{i_1}\cap\ldots\cap A_{i_k}) + \sum_{\{i_1,\ldots,i_{h+2}\}\in {}_h\mathcal{E}_2} P(A_{i_1}\cap\ldots\cap A_{i_{h+2}})$$

$$- \sum_{\{i_1,\ldots,i_{h+3}\}\in {}_h\mathcal{E}_3} P(A_{i_1}\cap\ldots\cap A_{i_{h+3}}) + \ldots + (-1)^{m+1} \sum_{\{i_1,\ldots,i_{h+m+1}\}\in {}_h\mathcal{E}_{m+1}} P(A_{i_1}\cap\ldots\cap A_{i_{h+m+1}}) \tag{5}$$

By induction on h, the inequality in (3) holds with the events $A_1 \cap A_n$, $A_2 \cap A_n$, $\ldots$, $A_{n-1} \cap A_n$ and Γ:

$$P\left(\bigcup_{i=1}^{n-1} (A_i \cap A_n)\right) \leqslant \sum_{k=1}^{h} (-1)^{k-1} \sum_{\{i_1,\ldots,i_k\}\subset\{1,\ldots,n-1\}} P(A_{i_1} \cap \ldots \cap A_{i_k} \cap A_n)$$

$$- \sum_{\{i_1,\ldots,i_{h+1}\}\in {}_{h-1}\mathcal{E}_2^*} P(A_{i_1}\cap\ldots\cap A_{i_{h+1}} \cap A_n) + \sum_{\{i_1,\ldots,i_{h+2}\}\in {}_{h-1}\mathcal{E}_3^*} P(A_{i_1}\cap\ldots\cap A_{i_{h+2}} \cap A_n) + \ldots$$

$$\ldots + (-1)^m \sum_{\{i_1,\ldots,i_{h+m}\}\in {}_{h-1}\mathcal{E}_{m+1}^*} P(A_{i_1}\cap\ldots\cap A_{i_{h+m}} \cap A_n). \tag{6}$$

We subtract the inequality in (6) from that in (5), add $P(A_n)$ to both sides and obtain

$$P\left(\bigcup_{i=1}^{n-1} A_i\right) + P(A_n) - P\left(\bigcup_{i=1}^{n-1} (A_i \cap A_n)\right) \geqslant \sum_{k=1}^{h+1} (-1)^{k-1} S_k + \sum_{\{i_1,\ldots,i_{h+2}\}\in {}_h\mathcal{E}_2'} P(A_{i_1}\cap\ldots\cap A_{i_{h+2}})$$

$$- \sum_{\{i_1,\ldots,i_{h+3}\}\in {}_h\mathcal{E}_3'} P(A_{i_1}\cap\ldots\cap A_{i_{h+3}}) + \ldots + (-1)^{m+1} \sum_{\{i_1,\ldots,i_{h+m+1}\}\in {}_h\mathcal{E}_{m+1}'} P(A_{i_1}\cap\ldots\cap A_{i_{h+m+1}}). \tag{7}$$

The term on the left-hand side is $P\left(\bigcup_{i=1}^{n} A_i\right)$. This completes the proof of the inequality in (4).

The proof can be done analogously whenever h is even. □

The theorem has been proved for the special case of $m = 1$ by I. Tomescu [13] and in the more special case of $m = 1$, $h = 0$ by D. Hunter [8] and K. J. Worsley [14]. In the case of $h = 0$, $m = 2$ some other results are detailed in [4].

If all the terms $P(A_{k_1} \cap \ldots \cap A_{k_i})$ $(1 \leqslant k_1 < \ldots < k_i \leqslant n,\ i = 1, \ldots, n)$ not used in the sums S_k and $w(\Delta)$ on the right-hand side of the formula in (3) or (4) are zeros, then the inequality in (3) or (4) becomes equality due to the inclusion-exclusion formula. Such events can be constructed in the following way. Let Δ be an (h,m)-hypermultistar with vertex set $\{1,\ldots,n\}$. Let the number of the hyperedges in Δ be denoted by N and let $B_1,\ldots,B_N$ be elementary probability events with $P(B_j)=\frac{1}{N}$ for all $j=1,\ldots,N$. Label the hyperedges in Δ from 1 to N and let $A_i=\{\cup B_j \mid$ the hyperedge with label j is incident to vertex $i\}$ for all $i=1,\ldots,n$. If $A_{k_1}\cap\ldots\cap A_{k_i}\neq\emptyset$, then $\{k_1,\ldots,k_i\}$ is a subset of a hyperedge in Δ with the following property. If $i\leqslant h+1$, then $P(A_{k_1}\cap\ldots\cap A_{k_i})$ is in S_i on the right-hand side of the formula in (3) or (4). On the other hand if $i\geqslant h+2$, then $\{k_1,\ldots,k_i\}$ is a hyperedge in Δ since a subset of a hyperedge in an (h,m)-hypermultistar is also a hyperedge assuming it has at least $h+2$ vertices. Thus $P(A_{k_1}\cap\ldots\cap A_{k_i})$ is again on the right-hand side of the formula in (3) or (4), consequently, the inequality in (3) or (4) becomes equality.

REMARK 13. An (h,m)-hypermultitree with $h+m$ or $h+m+1$ vertices has a unique structure: its hyperedges are all subsets of its vertex set containing at least $h+2$ vertices.

The right-hand side of the inequalities in (3) and in (4) are also referred as lower and upper bounds for $P\left(A_1\cup\ldots\cup A_n\right)$.

The following theorem has some practical importance.

THEOREM 14. *Let $\Delta=(V,\,{}_h\mathcal{E}_2,\ldots,\,{}_h\mathcal{E}_{m+1})$ be an (h,m)-hypermultitree with at least $h+m+1$ vertices. Then, an $(h,m+1)$-hypermultitree $\Delta'=(V,\,{}_h\mathcal{E}'_2,\ldots,\,{}_h\mathcal{E}'_{m+2})$ can be constructed with ${}_h\mathcal{E}_i\subset{}_h\mathcal{E}'_i$ for all $i=2,\ldots,m+1$; moreover the bound based on Δ' is at least as good as the bound based on Δ.*

Proof. The claim that the bound based on Δ' is at least as good as the bound based on Δ is equivalent to $w(\Delta)\leqslant w(\Delta')$. We use induction on h. For $h=0$ see the proof in [3, pp. 446–447]. We may assume without loss of generality that a recursion producing Δ is as follows: we start from (h,m)-hypermultitree with vertex set $\{1,\ldots,h+m\}$ then adjoin subsequently vertex k with the hyperedges of Γ_k extended by vertex k in icreasing order of $k=h+m+1,\ldots,n$. Here $\Gamma_k=(V^k,\,{}_{h-1}\mathcal{E}^k_2,\ldots,\,{}_{h-1}\mathcal{E}^k_{m+1})$ is an $(h-1,m)$-hypermultitree with $V^k=\{1,\ldots,k-1\}$ for all $k=h+m+1,\ldots,n$. Let $w^k(\Gamma_k)$ be the weight of Γ_k with measure $P^k(\cdot)=P(\cdot\cap A_k)$. It is easy to see that $w(\Delta)=w(\widehat{\Delta})+\sum_{k=h+m+2}^{n}w^k(\Gamma_k)$, where $\widehat{\Delta}$ was obtained by adjoining vertex $h+m+1$ with the hyperedges of Γ_{h+m+1} extended by vertex $h+m+1$. Since $\widehat{\Delta}$ has $h+m+1$ vertices, its hyperedges are all subsets of its vertex set containing at least $h+2$ vertices, by Remark (13). By induction on h, an $(h-1,m+1)$-hypermultitree $\Gamma'_k=(V^k,\,{}_{h-1}\mathcal{E}'^k_2,\ldots,\,{}_{h-1}\mathcal{E}'^k_{m+2})$ can be constructed with

$$ {}_{h-1}\mathcal{E}^k_i\subset{}_{h-1}\mathcal{E}'^k_i \quad \text{for all} \quad i=2,\ldots,m+1 \tag{8}$$

and

$$ w^k(\Gamma_k)\leqslant w^k(\Gamma'_k). \tag{9}$$

Now Δ' can be constructed as follows. We start from $(h,m+1)$-hypermultitree with vertex set $\{1,\ldots,h+m+1\}$, that is $\widehat{\Delta}$ by Remark (13), then adjoin subsequently vertex k with the hyperedges of Γ'_k extended by vertex k in icreasing order of $k=$

$h+m+2,\dots,n$. Since $w(\Delta')=w(\widehat{\Delta})+\sum\limits_{k=h+m+2}^{n} w^k(\Gamma'_k)$, $w(\Delta)\leqslant w(\Delta')$ follows from (9) and ${}_h\mathcal{E}_i \subset {}_h\mathcal{E}'_i$ $(i=2,\dots,m+1)$ follows from (8). $\square$

Using also complement events, inequlities in (1) and in (4) in case of $h=1$ can be reformulated as described in the two theorems below. These new formulae have significant practical benefit, some of them are detailed in the rest of the paper.

THEOREM 15. *Let $A_1,\dots,A_n$ be arbitrary events and let $\Delta=(V,\mathcal{E}_2,\dots,\mathcal{E}_{m+1})$ be an arbitrary m-multitree with $V=\{1,\dots,n\}$. Let $v_1,\dots,v_n$ be a permutation of the vertices in V and assume that a recursion producing Δ is as follows: we start from m-multitree with vertex set $\{v_1,\dots,v_m\}$ then adjoin subsequently the m-multicherries $(\{u_1^i,\dots,u_m^i\},v_{m+i})$ in icreasing order of $i=1,\dots,n-m$. Then*

$$P\left(A_1\cup\ldots\cup A_n\right)\leqslant 1-P(\overline{A}_{v_1}\cap\ldots\cap\overline{A}_{v_{m+1}})+\sum_{i=2}^{n-m}P\left(A_{v_{m+i}}\cap\overline{A}_{u_1^i}\cap\ldots\cap\overline{A}_{u_m^i}\right). \quad (10)$$

Proof. Since the vertices of m-multitree obtained by adjoining $(\{u_1^1,\dots,u_m^1\},v_{m+1})$ are $v_1,\dots,v_{m+1}$, its hyperedges are all subsets of $\{v_1,\dots,v_{m+1}\}$ with at least two elements, by Remark 13. These hyperedges contribute to the weight of Δ with

$$\sum_{1\leqslant j_1<j_2\leqslant m+1}P(A_{v_{j_1}}\cap A_{v_{j_2}})-\sum_{1\leqslant j_1<j_2<j_3\leqslant m+1}P(A_{v_{j_1}}\cap A_{v_{j_2}}\cap A_{v_{j_3}})+\ldots+(-1)^{m+1}P(A_{v_1}\cap\ldots\cap A_{v_{m+1}})$$

$$=\sum_{j=1}^{m+1}P(A_{v_j})-P(A_{v_1}\cup\ldots\cup A_{v_{m+1}})=\sum_{j=1}^{m+1}P(A_{v_j})-1+P(\overline{A}_{v_1}\cap\ldots\cap\overline{A}_{v_{m+1}}), \quad (11)$$

which follows from the inclusion-exclusion formula.

An m-multicherry $(\{u_1^i,\dots,u_m^i\},v_{m+i})$ contribute to the weight of Δ with

$$\begin{aligned}\sum_{1\leqslant j_1\leqslant m}P(A_{v_{m+i}}\cap A_{u_{j_1}^i})-\sum_{1\leqslant j_1<j_2\leqslant m}P(A_{v_{m+i}}\cap A_{u_{j_1}^i}\cap A_{u_{j_2}^i})+\ldots\\ \ldots+(-1)^{m+1}P(A_{v_{m+i}}\cap A_{u_1^i}\cap\ldots\cap A_{u_m^i})\\ =P\left((A_{v_{m+i}}\cap A_{u_1^i})\cup\ldots\cup(A_{v_{m+i}}\cap A_{u_m^i})\right)\\ =P\left(A_{v_{m+i}}\cap(A_{u_1^i}\cup\ldots\cup A_{u_m^i})\right)\\ =P\left(A_{v_{m+i}}\right)-P\left(A_{v_{m+i}}\cap(\overline{A_{u_1^i}\cup\ldots\cup A_{u_m^i}})\right)\\ =P\left(A_{v_{m+i}}\right)-P\left(A_{v_{m+i}}\cap\overline{A}_{u_1^i}\cap\ldots\cap\overline{A}_{u_m^i}\right),\end{aligned} \quad (12)$$

where the first equality is the inclusion-exclusion formula. By summing up the right-hand side of formula in (11) and (12) for all $i=2,\dots,n-m$, we obtain that the weight of Δ can be written in the form

$$w(\Delta)=\sum_{j=1}^{m+1}P(A_{v_j})-1+P(\overline{A}_{v_1}\cap\ldots\cap\overline{A}_{v_{m+1}})+\sum_{i=2}^{n-m}\left[P\left(A_{v_{m+i}}\right)-P\left(A_{v_{m+i}}\cap\overline{A}_{u_1^i}\cap\ldots\cap\overline{A}_{u_m^i}\right)\right]$$

$$= \sum_{j=1}^{n} P(A_{v_j}) - 1 + P(\overline{A}_{v_1} \cap \ldots \cap \overline{A}_{v_{m+1}}) - \sum_{i=2}^{n-m} P\left(A_{v_{m+i}} \cap \overline{A}_{u_1^i} \cap \ldots \cap \overline{A}_{u_m^i}\right). \quad (13)$$

Since $S_1 = \sum_{j=1}^{n} P(A_{v_j})$, the theorem follows from the inequality in (1). $\square$

THEOREM 16. *Let $A_1, \ldots, A_n$ be arbitrary events and let $\Delta = (V, {}_1\mathcal{E}_2, \ldots, {}_1\mathcal{E}_{m+1})$ be an arbitrary $(1,m)$-hypermultitree with $V = \{1, \ldots, n\}$. Let $w_1, \ldots, w_n$ be a permutation of the vertices in V and assume that a recursion producing Δ is as follows: we start from $(1,m)$-hypermultitree with vertex set $\{w_1, \ldots, w_{m+1}\}$ then adjoin subsequently the vertices w_{m+1+k} with the hyperedges of Γ_k extended by vertex w_{m+1+k} in icreasing order of $k = 1, \ldots, n-m-1$. Here Γ_k is an m-multitree on the vertices $w_1, \ldots, w_{m+k}$ for all $k = 1, \ldots, n-m-1$. Furthermore, assume that a recursion producing Γ_k is as follows: we start from m-multitree with vertex set $\{{}_kv_1, \ldots, {}_kv_m\}$ then adjoin subsequently the m-multicherries $(\{{}_ku_1^i, \ldots, {}_ku_m^i\}, {}_iv_{m+i})$ in icreasing order of $i = 1, \ldots, k$. Then*

$$P(A_1 \cup \ldots \cup A_n) \geqslant 1 - P(\overline{A}_{w_1} \cap \ldots \cap \overline{A}_{w_{m+2}}) +$$
$$\sum_{k=2}^{n-m-1} \left[P(A_{w_{m+1+k}} \cap \overline{A}_{{}_kv_1} \cap \ldots \cap \overline{A}_{{}_kv_{m+1}}) - \sum_{i=2}^{k} P\left(A_{w_{m+1+k}} \cap A_{{}_kv_{m+i}} \cap \overline{A}_{{}_ku_1^i} \cap \ldots \cap \overline{A}_{{}_ku_m^i}\right)\right]. \quad (14)$$

Proof. Since the vertices of $(1,m)$-hypermultitree obtained by adjoining w_{m+2} with the appropriate hyperedges are $w_1, \ldots, w_{m+2}$, its hyperedges are all subsets of $\{w_1, \ldots, w_{m+2}\}$ with at least three elements, by Remark 13. These hyperedges contribute to the weight of Δ with

$$\sum_{1 \leqslant j_1 < j_2 < j_3 \leqslant m+2} P(A_{w_{j_1}} \cap A_{w_{j_2}} \cap A_{w_{j_3}}) - \sum_{1 \leqslant j_1 < j_2 < j_3 < j_4 \leqslant m+2} P(A_{w_{j_1}} \cap A_{w_{j_2}} \cap A_{w_{j_3}} \cap A_{w_{j_4}}) +$$
$$\ldots + (-1)^{m+1} P(A_{w_1} \cap \ldots \cap A_{w_{m+2}})$$
$$= -\sum_{1 \leqslant j \leqslant m+2} P(A_{w_j}) + \sum_{1 \leqslant j < k \leqslant m+2} P(A_{w_j} \cap A_{w_k}) + P(A_{w_1} \cup \ldots \cup A_{w_{m+2}})$$
$$= -\sum_{1 \leqslant j \leqslant m+2} P(A_{w_j}) + \sum_{1 \leqslant j < k \leqslant m+2} P(A_{w_j} \cap A_{w_k}) + 1 - P(\overline{A}_{w_1} \cap \ldots \cap \overline{A}_{w_{m+2}}), \quad (15)$$

which follows from the inclusion-exclusion formula.

Similarly to the equality in (13), it can be seen that the hyperedges of Γ_k ($k = 2, \ldots, n-m-1$) extended by w_{m+1+k} contribute to the weight of Δ with

$$\sum_{j=1}^{m+k} P(A_{w_{m+1+k}} \cap A_{w_j}) - P(A_{w_{m+1+k}}) + P(A_{w_{m+1+k}} \cap \overline{A}_{{}_kv_1} \cap \ldots \cap \overline{A}_{{}_kv_{m+1}})$$
$$- \sum_{i=2}^{k} P\left(A_{w_{m+1+k}} \cap A_{{}_kv_{m+i}} \cap \overline{A}_{{}_ku_1^i} \cap \ldots \cap \overline{A}_{{}_ku_m^i}\right). \quad (16)$$

By summing up the right-hand side of formula in (15) and (16) for all $k = 2, \ldots, n-m-1$, we obtain that the weight of $(1,m)$-hipermultitree can be written in the form

$$w(\Delta) = -\sum_{1 \leqslant j \leqslant n} P(A_{w_j}) + \sum_{1 \leqslant j < k \leqslant n} P(A_{w_j} \cap A_{w_k}) + 1 - P(\overline{A}_{w_1} \cap \ldots \cap \overline{A}_{w_{m+2}})$$

$$+ \sum_{k=2}^{n-m-1} \left[P(A_{w_{m+1+k}} \cap \overline{A}_{kv_1} \cap \ldots \cap \overline{A}_{kv_{m+1}}) - \sum_{i=2}^{k} P\left(A_{w_{m+1+k}} \cap A_{kv_{m+i}} \cap \overline{A}_{ku_1^i} \cap \ldots \cap \overline{A}_{ku_m^i} \right) \right].$$

Since $S_1 = \sum_{1 \leqslant j \leqslant n} P(A_{w_j})$ and $S_2 = \sum_{1 \leqslant j < k \leqslant n} P(A_{w_j} \cap A_{w_k})$, the theorem follows from the inequality in (4) with $h = 1$. $\square$

REMARK 17. Similarly, inequlities in (3) and in (4) can be reformulated for any h. Their practical benefit is based on the fact that they supply lower and upper bounds for $P(A_1 \cup \ldots \cup A_n)$ based on as few as $\binom{n-m}{h+1}$ probabilities of the intersection of events.

4. Algorithms

In some applications, e.g., in computing bounds for the values of multivariate distribution functions or network-reliability, evaluation of intersection probabilities $P(A_{k_1} \cap \ldots \cap A_{k_i})$ is usually quite expensive. Therefore, it is crucial that already a relatively few of them should be enough to provide us with a good bound. In the view of Remark 17, an upper (lower) bound for $P(A_1 \cup \ldots \cup A_n)$ can be obtained based on as few as $n-m$ ($\binom{n-m}{2}$) intersection probabilities by means of m-multitrees ($(1,m)$-hypermultitrees).

There are a huge number of m-multitrees and $(1,m)$-hypermultitrees on n vertices and each of them provide us with a bound for $P(A_1 \cup \ldots \cup A_n)$. In the view of Theorem 12, our aim is to find heavy hypermultitrees. We have seen in Theorem 14 that an (h,m)-hypermultitree bound can be improved by increasing m. We gave an algorithm in [3] to obtain a heavy m-multitree. The algorithm starts from the Hunter-Worsley bound, i.e. the bound based on the heaviest 1-multitree, then r-multitree is extended to $(r+1)$-multitree recursively ($r = 1, \ldots, m$) to obtain better bounds.

Our 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 $(1,r)$-hypermultitree is extended to $(1,r+1)$-hypermultitree recursively ($r = 1, \ldots, m$) to obtain better bounds.

The first phase (constructing heavy $(1,1)$-hypermultitree):

Step 1: Construct $(1,1)$-hypermultitree $(\{k_1,k_2,k_3\}, \{\{k_1,k_2,k_3\}\})$, where $P(A_{k_1} \cap A_{k_2} \cap A_{k_3})$ is a maximum of the probabilities of the intersection of three events.

Let $L = \{1, \ldots, n\} \setminus \{k_1,k_2,k_3\}$, i.e., the list of still not taken vertices. Set the first three elements of array AR to k_1, k_2 and k_3 in any order.

Step 2: Let k be an element of list L. Introduce a weight function on the edges of the complete graph whose vertices are the vertices of the $(1,1)$-hypermultitree already constructed: let the weight of edge $\{i,j\}$ be $P(A_i \cap A_j \cap A_k)$. Assign the maximum weight spanning tree to vertex k. Do so with all elements of list L.

Step 3: Let k be an element of the list.the weight of the spanning tree assigned to is maximal. Adjoin vertex k and the edges of the assigned tree extended by k to the $(1,1)$-hypermultitree already constructed. Remove k from list L and put k into array AR.

Step 4: If the list is not empty, then go to Step 2, else STOP.

Notice that a tree is assigned to each $AR(i)$ with $i \geqslant 4$. In general, when an $(1,r)$-hypermultitree is already constructed, an r-multitree is assigned to each $AR(i)$ with $i \geqslant r+3$.

The second phase (extending $(1,r)$-hypermultitree to $(1,r+1)$-hypermultitree):

Step 1: Construct $(1,r+1)$-hypermultitree whose vertices are the first $r+3$ elements of array AR. By Remark 13, this hypermultitree has a unique structure. Set $counter = r+4$.

Step 2: Let $k = AR\ (counter)$. Introduce a weight function on the edges of the complete graph whose vertices are the vertices of the $(1,r+1)$-hypermultitree already constructed: let the weight of edge $\{i,j\}$ be $P(A_i \cap A_j \cap A_k)$.

Step 3: Extend the r-multitree assigned to k to $(r+1)$-multitree as described in [3, section Algorithms]. Adjoin the hyperedges of $(r+1)$-multitree not in r-multitree extended by k. Set $counter$ to $counter+1$.

Step 4: If $counter \leqslant n$, then go to Step 2, else STOP.

Note that the algorithm requires the knowledge of all probabilities in the sum S_3.

5. Comparison with other inequalities. Lower and upper bounds for the values of the multivariate normal probability distribution function.

In this section we compare our bounds based on m-multitrees and $(1,m)$-hypermultitrees with Grable's bounds [6]. We generalized Tomescu's bounds [13], which are $(h,1)$-hypermultitree bounds in our terminology. $(h,1)$-hypermultitrees are k-uniform hypergraphs, where $k = h+2$. They are generalized by Grable to k-matroid trees, which are also k-uniform hypergraphs. Grable's bound is obtained by replacing the weight of $(h,1)$-hypermultitree in Tomescu's bound with the weight of k-matroid tree. The fact that all k-matroid trees on a given vertex set constitute a matroid enables us to obtain the best bound based on k-matroid trees by a greedy algorithm. Thus, Grable's bound is at least as good as the same order Tomescu's bound (kth order bounds are those which are based on probabilities of intersections of at most k events).

To compute Grable's k-matroid tree bounds we used the algorithm that Grable recommends in his paper [7]. To compute a bound based on a k-matroid tree we need to compute all probabilities in the sums $S_1,\dots,S_{k-1}$ and $\binom{n-1}{k-1}$ probabilities in the sum S_k. To execute Grable's algorithm we need to compute all probabilities in the sum S_k. However, to compute $(k-1)$-multitree bounds or $(1,k-2)$-hypermultitree bounds, which are of the same order as the k-matroid tree bound, we need only $n-k+1$, $\binom{n-k+2}{2}$ probabilities in S_k resp. (see Remark 17), and to execute our algorithm that finds a heavy $(k-1)$-multitree ($(1,k-2)$-hypermultitree) we need to compute only the probabilities in the sum S_2 (S_3) (see the previous section). This explains why the computation of m-multitree and $(1,m)$-hypermultitree bounds require significantly less time.

Another advantage of (h,m)-hypermultitree bounds is that they can be improved by increasing m (see previous section and Theorem 14). In other words, higher order bounds are better than lower order ones when using our algorithm. As our numerical examples show, the same is not true for k-matroid tree bounds.

Below we present four numerical examples and list the bounds obtained by different methods. We take the event A_i as $\{\xi_i < x_i\}$, $i = 1,\dots,n$, for given $x_1,\dots,x_n$, where the random vector $(\xi_1,\dots,\xi_n)$ has an n-variate standard normal probability distribution. We give lower and upper bounds for the probability $P(\overline{A}_1 \cup \ldots \cup \overline{A}_n)$ to obtain upper and lower bounds, respectively for $F(x_1,\dots,x_n) = P(\xi_1 < x_1,\dots,\xi_n < x_n)$, by the use of the equation $P(\xi_1 < x_1,\dots,\xi_n < x_n) = 1 - P(\overline{A}_1 \cup \ldots \cup \overline{A}_n)$. The algorithms were implemented in FORTRAN, the required probabilities $P(\overline{A}_i)$, $P(\overline{A}_i \cap \overline{A}_j)$ $(1 \leqslant i < j \leqslant n)$ were computed by the IMSL subroutines MDNOR and MDBNOR, $P(\overline{A}_{k_1} \cap \ldots \cap \overline{A}_{k_i})$ $(1 \leqslant k_1 < \ldots < k_i \leqslant n,\ i \geqslant 3)$ were computed by Genz's FORTRAN code, SADMVN [5] with accuracy 10^{-6}. All computations have been carried out by a CELERON II 850MHz computer.

EXAMPLE 1. et $x_1 = x_2 = \ldots = x_{20} = 2.5$, and the correlations: $r_{i,j} = \sqrt{\frac{i}{j}}$ for all $1 \leqslant i < j \leqslant 20$.

Dimension 20

	Lower bound	Upper bound	Time (seconds)
4-matroid tree	0.922785		19.44
Hunter-Worsley	0.948492		0.01
2-multitree	0.954413		0.01
3-multitree	0.956893		0.01
4-multitree	0.958254		0.05
5-multitree	0.959109		0.11
6-multitree	0.959727		0.28
7-multitree	0.960121		0.48
8-multitree	0.960435		2.14
3-matroid tree		0.983400	2.47
5-matroid tree		1.021199	196.86
$(1,1)$-hypermultitree		0.983406	0.40
$(1,2)$-hypermultitree		0.967585	0.40
$(1,3)$-hypermultitree		0.963973	0.40
$(1,4)$-hypermultitree		0.962691	0.56
$(1,5)$-hypermultitree		0.962107	0.62
$(1,6)$-hypermultitree		0.961798	1.00
$(1,7)$-hypermultitree		0.961601	1.94

EXAMPLE 2. Let $x_1 = 0.1$, $x_2 = 0.2$, $x_3 = 0.3$, $\ldots$, $x_{19} = 1.9$, $x_{20} = 2.0$, and $r_{i,j} = 0.8$ for all $1 \leqslant i < j \leqslant 20$.

Dimension 20

	Lower bound	Upper bound	Time (seconds)
4 -matroid tree	0.111152		33.29
Hunter-Worsley	0.199069		0.01
2 -multitree	0.341837		0.01
3 -multitree	0.370904		0.01
4 -multitree	0.379209		0.05
5 -multitree	0.381928		0.06
6 -multitree	0.382864		0.16
7 -multitree	0.383196		0.27
8 -multitree	0.383310		0.71
3 -matroid tree		0.614862	2.52
5 -matroid tree		0.668914	530.80
$(1,1)$ -hypermultitree		0.614862	0.34
$(1,2)$ -hypermultitree		0.396591	0.45
$(1,3)$ -hypermultitree		0.387411	0.45
$(1,4)$ -hypermultitree		0.384342	0.56
$(1,5)$ -hypermultitree		0.383590	0.62
$(1,6)$ -hypermultitree		0.383422	0.89
$(1,7)$ -hypermultitree		0.383380	1.50

EXAMPLE 3. Let $x_1 = 3.0$, $x_2 = 2.9$, $x_3 = 2.8$, $\ldots$, $x_{29} = 0.2$, $x_{30} = 0.1$, and $r_{1,2} = 0.990$, $r_{1,3} = 0.988$, $r_{2,3} = 0.986$, $r_{1,4} = 0.984$, $r_{2,4} = 0.982$, $r_{3,4} = 0.980$, $r_{1,5} = 0.978$, $\ldots$, $r_{27,30} = 0.126$, $r_{28,30} = 0.124$, $r_{29,30} = 0.122$.

Dimension 30

	Lower bound	Upper bound	Time (seconds)
4 -matroid tree	-5.191175		408.15
Hunter-Worsley	-0.973654		0.01
2 -multitree	-0.341882		0.05
3 -multitree	-0.094775		0.07
4 -multitree	0.007970		0.11
5 -multitree	0.054646		0.50
6 -multitree	0.076328		1.10
7 -multitree	0.087042		2.63
8 -multitree	0.092530		4.62
3 -matroid tree		2.517407	11.54
$(1,1)$ -hypermultitree		2.517407	2.31
$(1,2)$ -hypermultitree		0.161507	2.35
$(1,3)$ -hypermultitree		0.134832	2.51
$(1,4)$ -hypermultitree		0.120210	2.96
$(1,5)$ -hypermultitree		0.111012	4.23
$(1,6)$ -hypermultitree		0.105752	6.69
$(1,7)$ -hypermultitree		0.102767	9.82

EXAMPLE 4. Let $x_1 = 1.84$, $x_2 = 1.88$, $x_3 = 1.92$, $\dots$, $x_{29} = 2.96$, $x_{30} = 3.0$, and the correlations: $r_{i,j} = \sqrt{\frac{i}{j}}$ for all $1 \leqslant i < j \leqslant 30$.

Dimension 30	Lower bound	Upper bound	Time (seconds)
4-matroid tree	0.719174		428.86
Hunter-Worsley	0.861747		0.01
2-multitree	0.877985		0.05
3-multitree	0.884730		0.07
4-multitree	0.888459		0.08
5-multitree	0.890727		0.20
6-multitree	0.892263		0.77
7-multitree	0.893355		2.04
8-multitree	0.894148		8.63
3-matroid tree		0.972666	13.40
$(1,1)$-hypermultitree		0.972666	1.14
$(1,2)$-hypermultitree		0.909455	1.27
$(1,3)$-hypermultitree		0.903721	1.43
$(1,4)$-hypermultitree		0.900948	1.65
$(1,5)$-hypermultitree		0.899480	2.46
$(1,6)$-hypermultitree		0.898524	3.08
$(1,7)$-hypermultitree		0.897911	5.43

REFERENCES

[1] G. BOOLE, *"Laws of Thought"*, American reprint of 1854 editon, Dover, New York, 1854.

[2] C. E. BONFERRONI, *Teoria Statistica Delle Classi e Calcolo Delle Probabilitá*, *Pubbl. Ist. Sup. Sci. Econ. Commerciali Firenze* **8** (1936), 1–62.

[3] J. BUKSZÁR, *Probability Bounds with Multitrees*, *Adv. Appl. Prob.* **33** (2001), 437–452.

[4] J. BUKSZÁR AND A. PRÉKOPA, *Probability Bounds with Cherry Trees*, *Math. Operat. Res.* **26** (2001), 174–192.

[5] A. GENZ, *Numerical Computation of the Multivariate Normal Probabilities*, *J. Comput. Graph. Stat.* **1** (1992), 141–150.

[6] D. A. GRABLE, *Two Packing Problems on k-Matroid Trees*, *Europ. J. Combinatorics* **12** (1991), 309–316.

[7] D. A. GRABLE, *Sharpened Bonferroni Inequalities*, *J. Combin. Theory Ser. B* **57** (1993), 131–137.

[8] D. HUNTER, *An Upper Bound for the Probability of a Union*, *J. Appl. Prob.* **13** (1976), 597–603.

[9] A. PRÉKOPA, *"Stochastic Programming"*, Kluwer Academic Publishers, Dordrecht, 1995.

[10] A. PRÉKOPA, *Boole-Bonferroni Inequalities and Linear Programming*, *Oper. Res.* **36** (1988), 145–162.

[11] A. PRÉKOPA, *Sharp Bounds on Probabilities Using Linear Programming*, *Oper. Res.* **38** (1990), 227–239.

[12] T. SZÁNTAI AND J. BUKSZÁR, *Probability Bounds Given by Hypercherry Trees*, *Optimization Methods and Software* (accepted).

[13] I. TOMESCU, *Hypertrees and Bonferroni Inequalities*, *J. Combin. Theory Ser. B* **41** (1986), 209–217.

[14] K. J. WORSLEY, *An Improved Bonferroni Inequality and Applications*, *Biometrika* **69** (1982), 297–302.

(Received April 15, 2002)

Institute of Mathematics
University of Miskolc
Miskolc-Egyetemvaros, H-3515
Hungary e-mail: matbuk@uni-miskolc.hu