Quorum Constraints and Filters in Boolean Lattices

Zbigniew Lonc
Department of Computer Science
University of Kentucky
Lexington, KY 40506, USA
and
Faculty of Mathematics and Information Science
Warsaw University of Technology
00-661 Warsaw, Poland

Victor W. Marek
Department of Computer Science
University of Kentucky
Lexington, KY 40506, USA

Abstract

We investigate constraints in finite Boolean lattices $\langle \mathcal{P}(X), \subseteq \rangle$ where X is a finite set. The constraints studied here are of the form $\langle Z, k \rangle$ where $Z \subseteq X$, $1 \leq k \leq |Z|$. A set $I \subseteq X$ *satisfies* $\langle Z, k \rangle$ if $|I \cap Z| \geq k$. We characterize the sets satisfying collections of such constraints as filters (final segments) in $\langle \mathcal{P}(X) \rangle$. We find yet other characterizations of filters including one by means of families of sets indexed by elements of X so that the elements of the filter correspond to subfamilies with an empty intersection. Our characterizations are supported for algorithms. We also study the families of negated constraints and mixed families and find their characterizations. In the positive case, formulas built of constraints can be used to measure the complexity of filters (and thus also of antichains of their minimal elements). We find pathological filters with very simple descriptions when the disjunctions are allowed, but extremely complex descriptions when only conjunctions are allowed.

1 Introduction

In this paper we investigate various constraints on subsets of a given finite set X. First, we focus on the class of constraints that require the presence of "sufficiently many elements". Those are constraints of the form $q = \langle Z, k \rangle$ where Z is a subset of X, k is a natural number, and $|Z| \geq k$. Such a constraint, called below a *positive quorum constraint*, or simply a *constraint*, requires that a putative set I satisfying it has the property that $|I \cap Z| \geq k$. That is, the set I contains at least k of the elements of Z. We will be

JCMCC **48** (2004), pp. 115-137

concerned mostly with sets Q consisting of positive constraints (although other constraints will also be considered). We will study collections of sets satisfying constraints in a given set Q. Sets satisfying all constraints in a set Q are called *access structures* with respect to Q [10]. When Q consists of just one constraint $\langle X, k\rangle$ (where X is the entire set under consideration, $k \in N, |X| \geq k$), our problem reduces to the following well-known problem of distribution of keys [9], see also [7, 4].

> *A group of scientists is working on a secret project. They wish to lock up the documents in a cabinet so that the cabinet can be opened if and only if k or more scientists are present. How many locks are needed? How the keys to the locks have to be distributed among the scientists?*

More formally: "Given the set X of size $\geq k$ and a key K, one wants to distribute K so that any k or more parts reconstruct K, while any less than k parts cannot reconstruct K." Thus, if we are interested in the family of those subsets of X that can reconstruct the key K, then it is precisely the family $\mathcal{F}$ of subsets of X that satisfy the constraint $\langle X, k\rangle$. Shamir [9] credits this problem to Liu [5]. Here we do not limit to Shamir's constraint, but consider arbitrary collections Q of constraints. Our Theorem 2.1 characterizes the set of access structures with respect to Q. In a related paper [6] we study an application of the theory developed here. We show how the access structures can be used in areas such as computer security, key partitioning, distributed certificates, confirmation of identity, and other related topics.

We prove that given a set of positive constraints Q, the family of those sets $I \subseteq X$ that satisfy all constraints in Q forms a filter (final segment) in the Boolean lattice $\langle \mathcal{P}(X), \subseteq\rangle$. Conversely, given a filter $\mathcal{F}$ we can find a set of constraints Q so that $\mathcal{F}$ is precisely the collection of sets that satisfy Q. We also prove a number of other characterizations of filters. One of these characterizations can be used to solve the *quorum problem* (see below) for sets of positive constraints.

While positive constraints are investigated and characterized in Section 2, in the next section we study *negative constraints* that is constraints of the form $q = \neg\langle Z, k\rangle$. Such a constraint is satisfied by a set $I \subseteq Y$ if $|I \cap Z| < k$. While positive constraints are closely associated with filters (i.e. final segments) in $\langle \mathcal{P}(X), \subseteq\rangle$, the negative constraints are associated with ideals, i.e. initial segments in $\langle \mathcal{P}(X), \subseteq\rangle$. Consequently sets of *quorum literals*, i.e. positive and negative constraints, are related to segments in $\langle \mathcal{P}(X), \subseteq\rangle$. Although sets of positive constraints are always satisfiable,

and similarly sets of negative constraints are satisfiable, the mixed sets (of positive and negative literals) are not necessarily satisfiable.

Since we are interested in an equivalent characterization of sets of constraints, we will construct various families of sets that can be used to test the satisfaction of constraints. Specifically, one such characterization involves constructing a family of sets $\{G_m : m \in X\}$ so that a set $I \subseteq X$ satisfies Q if and only if $\bigcap_{m \in I} G_m = \emptyset$. This characterization can be used to solve the *quorum problem*. To explicate this problem notice that for the single constraint $q = \langle X, k \rangle$ both the Shamir and Mignotte solutions of the key partitioning problem amount to giving techniques to show if a given set $I \subseteq X$ satisfies q without, actually, storing I. Extending this problem we formulate the *quorum problem* as follows

> *Given a set of constraints Q devise a technique for checking if any set $I \subseteq C$ satisfies Q without describing explicitly the access structure.*

Theorem 2.1 (the equivalence (a) $\Leftrightarrow$ (e)) gives us a solution of the quorum problem for positive constraints. In [6] we show how to apply this solution to design a secret sharing scheme. Our solution concerns a general case of sets of constraints instead of just one constraint but it is computationally expensive (see [6] for the complete analysis of the complexity of our solution). In Section 3 we deal with the sets of negative constraints and mixed collections of constraints. Theorems 3.1 (the equivalence (a) $\Leftrightarrow$ (e)) and 3.2 (the equivalence (a) $\Leftrightarrow$ (c)) provide a solution of the quorum problem for both negative and mixed collections of constraints. However, it is not clear to us if these results can be applied somehow in secret sharing scheme design. In Section 4 we investigate the issue of description of filters. The language of constraints allows us to measure the complexity of filters, namely by assigning to a filter the simplest formula built of constraints that describes that filter. It turns out that the form of formulas makes a a fundamental difference in such descriptions. We construct an example of a family of filters that is simply describable when both conjunctions and disjunctions are allowed, but is immensely complex when the alphabet allows only for conjunctions. We also give estimates for the measure of descriptions of filters. Our algorithms are presented in Section 5. Finally, we discuss the issues associated with other types of constraints in Section 6.

2 Positive Quorum constraints

Example 2.1 The college intellectual property committee consists of nine individuals, $\{x_1, \ldots, x_9\}$. Of these five are faculty, $\{x_1, \ldots, x_5\}$, and four are students $\{x_6, \ldots, x_9\}$. Dr. x_1 and Ms. x_6 are co-chairs of the committee.

The meeting of the committee is legal if the following three conditions are met:

1. At least one of co-chairs is present,
2. At least three faculty members are present,
3. At least two student members are present.

We want to assign to each individual a set of indices so that the quorum can be tested automatically – by the computation of the intersection of the sets assigned to each member of the committee. More precisely, we would like to assign to each member x_i of the committee a set G_i so that:

> *A set $\{x_i : i \in I\}$ forms a legal meeting of the committee if and only if the intersection $\bigcap_{i \in I} G_i$ is empty.*

Here is such family of sets $\{G_1, \ldots, G_9\}$. All sets G_i will be included in the set $[15] = \{1, \ldots, 15\}$. We list them below.

$$
\begin{aligned}
G_1 &= \{2, 3, 4, 5, 12, 13, 14, 15\} \\
G_2 &= \{1, 2, 6, 7, 8, 12, 13, 14, 15\} \\
G_3 &= \{1, 3, 6, 9, 10, 12, 13, 14, 15\} \\
G_4 &= \{1, 4, 7, 9, 11, 12, 13, 14, 15\} \\
G_5 &= \{1, 5, 8, 10, 11, 12, 13, 14, 15\} \\
G_6 &= \{2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12\} \\
G_7 &= \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 13\} \\
G_8 &= \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 14\} \\
G_9 &= \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 15\}
\end{aligned}
$$

For instance, the set $\{x_2, x_3, x_5, x_6, x_9\}$ forms a legal meeting of the committee. Indeed the co-chair x_6 is present, three faculty x_2, x_3, x_5 are present, as well as two students: x_6, and x_9.

Now, it is easy to see that the intersection $G_2 \cap G_3 \cap G_5 \cap G_6 \cap G_9$ is, indeed empty. To see this notice that $1 \notin G_6$, $2 \notin G_3$, $3 \notin G_2$, $4 \notin G_3$, $5 \notin G_2$,

$6 \notin G_5$, $7 \notin G_3$, $8 \notin G_3$, $9 \notin G_2$, $10 \notin G_2$, $11 \notin G_2$, $12 \notin G_9$, $13 \notin G_6$, $14 \notin G_6$ and $15 \notin G_6$. It may be of interest that our set $\{x_2, x_3, x_5, x_6, x_9\}$ is a minimal set satisfying all our constraints, and that every proper subfamily of the family $\{G_2, G_3, G_5, G_6, G_9\}$ has a nonempty intersection. □

Generalizing from this example we define, a *positive quorum constraint* (or simply a *constraint*) as a pair $\langle Y, k\rangle$ where $k \leq |Y|$ (other quorum constraints will be considered in Section 3). Intuitively, a positive quorum constraint is a constraint on a putative set I requiring that out of the elements of Y, at least k elements belong to I. In our example 2.1 we had three constraints:

1. $\langle\{x_1, x_6\}, 1\rangle$ expressing the constraint that at least one of co-chairs is present,
2. $\langle\{x_1, x_2, x_3, x_4, x_5\}, 3\rangle$ expressing the constraint that at least three faculty members are present, and
3. $\langle\{x_6, x_7, x_8, x_9\}, 2\rangle$ expressing the constraint that at least two student members are present.

Formally, we will say that a set I is an *access structure with respect to a (positive) quorum constraint* $q = \langle Y, k\rangle$ if $|I \cap Y| \geq k$. We denote this by $I \models q$.

We call a family $\mathcal{I} \subseteq \mathcal{P}(X)$ an *ideal* if for all $Y \in \mathcal{I}$, and for all Z, if $Z \subseteq Y$ then $Z \in \mathcal{I}$. Similarly, we call a family $\mathcal{F} \subseteq \mathcal{P}(X)$ a *filter* if for all $Y \in \mathcal{F}$, and for all Z, if $Y \subseteq Z$ then $Z \in \mathcal{F}$. Let $\mathcal{A}$ be a family of sets. A set T is a *transversal* of $\mathcal{A}$, if $T \cap A \neq \emptyset$, for every $A \in \mathcal{A}$. A family $\mathcal{A} \subseteq \mathcal{P}(X)$ is an *antichain* in if $A \not\subseteq B$, for every $A, B \in \mathcal{A}$, $A \neq B$.

If I is an access structure with respect to some positive quorum constraint q and $I \subseteq J$ then J is also an access structure with respect to q. Thus the same holds for sets Q of quorum constraints. Hence, whenever Q is a set of quorum constraints, then the collection of access structures with respect to Q forms a filter. We will denote this filter by $\mathcal{F}_Q = \{I \subseteq X : I \models Q\}$. It turns out (see Theorem 2.1 below) that the converse of this statement is also true. More precisely, every family of subsets of $[n]$ which is a filter is the set of access structures with respect to some set of quorum constraints. In other words quorum constraints suffice to define every filter in $\mathcal{P}(X)$. Actually, a stronger statement is true. Specifically, every filter in $\mathcal{P}(X)$ is a set of access structures with respect to a set of some very simple quorum constraints Q called unitary quorum constraints.

A quorum constraint $\langle Y, k\rangle$ is *unitary* if $k = 1$. Clearly, a set I is an access structure with respect to a unitary quorum constraint $\langle Y, 1\rangle$ if and only if $Y \cap I \neq \emptyset$.

The following theorem states a number of characterizations of families of access structures with respect to a given set of quorum constraints.

Theorem 2.1 *The following conditions are equivalent.*

(a) *A family $\mathcal{F} \subseteq \mathcal{P}([m])$ is a filter in $\mathcal{P}([m])$.*

(b) *There is a set of quorum constraints Q such that $\mathcal{F} = \mathcal{F}_Q$.*

(c) *There is a set of unitary quorum constraints Q' such that $\mathcal{F} = \mathcal{F}_{Q'}$.*

(d) *There is a family of sets $\mathcal{A}$ such that $\mathcal{F}$ is the family of transversals of $\mathcal{A}$.*

(e) *There is a family of sets $\mathcal{G} = \{G_1, ..., G_m\}$ such that*

$$\bigcap_{i \in I} G_i = \emptyset \text{ if and only if } I \in \mathcal{F}.$$

(f) *There is a family of sets $\mathcal{G}' = \{G'_1, ..., G'_m\}$ such that*

$$\bigcap_{i \in \bar{I}} G'_i \neq \emptyset \text{ if and only if } I \in \mathcal{F}.$$

(Notice that the intersection in **(f)** is taken over the complement $\bar{I}$ of I rather than I itself)

Proof: **(a)** $\Rightarrow$ **(d)**

Consider the ideal $\mathcal{I} = \mathcal{P}([m]) \setminus \mathcal{F}$. Since m is finite, the ideal $\mathcal{I}$ has maximal elements. Define $\mathcal{M}_\mathcal{I}$ to be the set of all maximal elements of $\mathcal{I}$. Enumerate $\mathcal{M}_\mathcal{I}$ into $\{M_1, \ldots, M_p\}$.

We claim that $\mathcal{F}$ is the family of transversals of the family

$$\mathcal{A} = \{[m] \setminus M_j : 1 \leq j \leq p\}.$$

Indeed, denote $X_j = [m] \setminus M_j$. What we need to show is that $\mathcal{F}$ consists of those sets I that have nonempty intersection with all sets X_j.

First, assume that $I \in \mathcal{F}$. Then $I \not\subseteq M_j$, for every $1 \leq j \leq p$, so

$$I \cap X_j = I \cap ([m] \setminus M_j) \neq \emptyset,$$

for all $1 \leq j \leq p$.

Conversely, if $I \cap X_j \neq \emptyset$, $1 \leq j \leq p$, then $I \not\subseteq [m] \setminus X_j = M_j$, so $I \notin \mathcal{I}$. Thus $I \in \mathcal{F}$.

(d) $\Rightarrow$ (c)

Let $\mathcal{F}$ be the family of transversals of a family $\mathcal{A}$. Since $I \models \langle Y, 1 \rangle$ if and only if $I \cap Y \neq \emptyset$, we get $\mathcal{F} = \mathcal{F}_{Q'}$, where $Q' = \{\langle Y, 1 \rangle : Y \in \mathcal{A}\}$ is a set of unitary quorum constraints.

(a) $\Rightarrow$ (e)

Assign to each $M_j \in \mathcal{M}_{\mathcal{I}}$ a new object a_j and let $M = \{a_1, \ldots, a_p\}$. Define

$$G_i = \{a_j \in M : i \in M_j\}, \quad 1 \leq i \leq m.$$

Let $I \in \mathcal{F}$ and suppose that for some $I \subseteq [m]$, $\bigcap\{G_i : i \in I\} \neq \emptyset$. Let $a_j \in \bigcap\{G_i : i \in I\}$. By the definition of G_i, it must be the case that $i \in M_j$, for every $i \in I$. Thus $I \subseteq M_j$. But $\mathcal{I}$ is an ideal and $M_j \in \mathcal{I}$. Hence $I \in \mathcal{I} = \mathcal{P}([m]) \setminus \mathcal{F}$, a contradiction. Thus $\bigcap\{G_i : i \in I\} = \emptyset$.

Conversely, let $\bigcap\{G_i : i \in I\} = \emptyset$ and assume that $I \notin \mathcal{F}$, that is $I \in \mathcal{I}$. Since m is finite, every $I \in \mathcal{I}$ is included in some maximal element of $\mathcal{I}$. Let M_j be maximal in $\mathcal{I}$ so that $I \subseteq M_j$. Given $i \in I$, $i \in M_j$, thus, by the definition of G_i, $a_j \in G_i$. But then $a_j \in G_i$ for every $i \in I$, that is $a_j \in \bigcap\{G_i : i \in I\}$, thus $\bigcap\{G_i : i \in I\} \neq \emptyset$, a contradiction. Hence $I \in \mathcal{F}$.

Since the implications **(e)** $\Rightarrow$ **(a)**, **(c)** $\Rightarrow$ **(b)** and **(b)** $\Rightarrow$ **(a)** are trivial, we get equivalence of the conditions **(a)** $-$ **(e)**.

(a) $\Leftrightarrow$ (f)

Define $\mathcal{F}' = \{\bar{I} \in \mathcal{P}([m]) : I \notin \mathcal{F}\}$. Clearly, $\mathcal{F}'$ is a filter if and only if $\mathcal{F}$ is a filter and $I \in \mathcal{F}$ if and only if $\bar{I} \notin \mathcal{F}'$. The equivalence **(a)** $\Leftrightarrow$ **(f)** now follows easily from the equivalence **(a)** $\Leftrightarrow$ **(e)** applied to the family $\mathcal{F}'$. □

Example 2.1 is an illustration of an application of the characterization of a family of access structures given in Theorem 2.1 **(e)**. This condition allows us to reduce the problem of verification if a set I is an access structure with respect to a given set of quorum constraints to testing if the intersection of some family of sets is empty.

Let Q be a set of quorum constraints. Denote by $\mathcal{G}_Q = \{G_1, \ldots, G_m\}$ a family of sets such that

$$\bigcap_{i \in I} G_i = \emptyset \text{ if and only if } I \models Q \qquad (\star)$$

whose existence is guaranteed by Theorem 2.1.

Theorem 2.1 does not tell us how to construct the family $\mathcal{G}_Q$ for a given set of quorum constraints Q. Yet, the proof of this theorem suggests an algorithm for such a construction. We will provide one such algorithm and analyze its complexity in Section 5.

Example 2.2 We are now in the position to explain the construction of the family $\{G_1, \ldots G_9\}$ in our intellectual property committee example.

To see how this family is constructed, recall the construction given in the proof of the implication (a) $\Rightarrow$ (e) of our Theorem 2.1.

It is easy to see that the ideal $\mathcal{I}_Q$ has the following 15 maximal elements:

1. $\{x_2, x_3, x_4, x_5, x_7, x_8, x_9\}$ (both co-chairs missing)
2. $Y \cup \{x_6, x_7, x_8, x_9\}$ where Y is a two-element subset of $\{x_1, \ldots, x_5\}$ (two professors only, but all the students). There are 10 such sets.
3. $\{x_1, \ldots, x_5\} \cup \{x_i\}$ where $6 \leq i \leq 9$ (one student only, but all the professors). There are four such elements.

In this manner we construct auxiliary sets $M_1, \ldots, M_{15}$. Consequently the sets $G_i, 1 \leq i \leq 9$ will be subsets of $[15]$. We recall that

$$G_i = \{j : x_j \in M_i\}.$$

This way we get the sets we listed in our motivational example.

Let us conclude this section with a corollary of Theorem 2.1.

A family of sets $\mathcal{G}$ is a *k-threshold* family if the intersection of every $k-1$ sets from $\mathcal{G}$ is nonempty, but the intersection of any k distinct sets from $\mathcal{G}$ is empty. Applying Theorem 2.1 for the filter $\mathcal{F}_k = \{A \in \mathcal{P}([m]) : |A| \geq k\}$ we get the following corollary.

Corollary 2.2 *Let $m \geq 2$. Then for every integer k, $2 \leq k \leq m$, there is a k-threshold family $\mathcal{G}$ such that $|\mathcal{G}| = m$.* □

Let us conclude this section with a solution of a generalization of the 'problem of scientists working on a secret project' that was the original motivation of Shamir's problem.

Suppose that the set of groups of scientists permitted to open a cabinet with secret documents is described by a set of constraints rather than just a single constraint considered in the original problem. By our Theorem 2.1, we can assign to every scientist i a set G_i such that a group I of scientists is allowed to open the cabinet if and only if $\bigcap_{i \in I} G_i = \emptyset$. This equality is equivalent to $\bigcup_{i \in I}(M \setminus G_i) = M$, where $M = \bigcup_i G_i$. Now, we install $|M|$ locks labelled with members of M in the cabinet and the ith scientist receives keys to the locks belonging to the set $M \setminus G_i$. Clearly, all locks can be opened by a group of scientists if and only if $\bigcup_{i \in I}(M \setminus G_i) = M$ which (by Theorem 2.1) is satisfied if and only I is a group of scientists permitted to open the cabinet.

3 Other types of constraints

Motivated by applications in cryptography, in the previous section we have been considering access structures defined by sets of positive quorum constraints. It turns out that negative quorum constraints have some motivations in secret sharing schemes as well (see [1], [8]).

Suppose the college intellectual property committee considered in Example 2.1 contains one additional person x_{10} (the dean?) who has a veto right, i.e. a meeting of the committee is legal if the presence conditions formulated in Example 2.1 are satisfied and the dean does not use his veto right to cancel legality of the meeting. This veto right can be formulated as a negative quorum constraint $\neg\langle\{x_{10}\}, 1\rangle$. Clearly, we can generalize the veto right to a group of persons instead of just one person by assuming that legality of a meeting can be cancelled if any k members of a certain special group of persons Y are in favor of the cancellation but it can not be done by any $k-1$ persons in Y. This is equivalent to the negative constraint $\neg\langle Y, k\rangle$.

As we have seen in our discussion in Section 2 a set of quorum constraints determines a filter. Thus a negative quorum constraint, i.e. a literal of the form $n = \neg\langle Z, k\rangle$ determines an ideal. In other words, the family of sets that do not satisfy $\langle Z, k\rangle$ forms an ideal. Consequently, any family of negative quorum constraints also determines an ideal.

We will call positive quorum constraints and their negations, *quorum literals.*

Notice that the lattice dual to $\langle \mathcal{P}([m]), \subseteq\rangle$ is $\langle \mathcal{P}([m]), \supseteq\rangle$. Thus the lattice $\langle \mathcal{P}([m]), \subseteq\rangle$ is isomorphic to its own dual. The isomorphism is es-

tablished by the mapping $c : I \mapsto \bar{I}$, where $\bar{I} = [m] \setminus I$. Under the mapping c, the filters are mapped onto ideals and conversely, ideals are mapped to filters. Thus we get a theorem dual to Theorem 2.1. Before we formulate it, we need some additional definitions.

A negative quorum constraint $q = \neg\langle Y, k\rangle$ is *antiunitary* if $k = |Y|$, i.e. a set I satisfies such a constraint if and only if $Y \not\subseteq I$. As before we will write $I \models q$ if I satisfies such constraint. Let $\mathcal{A}$ be a family of sets. A set T is an *antitransversal* of $\mathcal{A}$, if $A \not\subseteq T$, for every $A \in \mathcal{A}$.

Let Q be a set of negative quorum constraints. Denote by $\mathcal{I}_Q = \{I \subseteq [m] : I \models Q\}$ the ideal of sets satisfying all negative quorum constraints in Q.

Theorem 3.1 *The following conditions are equivalent.*

(a) *A family $\mathcal{I} \subseteq \mathcal{P}([m])$ is an ideal in $\mathcal{P}([m])$.*

(b) *There is a set of negative quorum constraints Q such that $\mathcal{I} = \mathcal{I}_Q$.*

(c) *There is a set of antiunitary quorum constraints Q' such that $\mathcal{I} = \mathcal{I}_{Q'}$.*

(d) *There is a family of sets $\mathcal{A}$ such that $\mathcal{I}$ is the family of antitransversals of $\mathcal{A}$.*

(e) *There is a family of sets $\mathcal{H} = \{H_1, ..., H_m\}$ such that*

$$\bigcap_{i \in I} H_i \neq \emptyset \text{ if and only if } I \in \mathcal{I}.$$

(f) *There is a family of sets $\mathcal{H}' = \{H'_1, ..., H'_m\}$ such that*

$$\bigcap_{i \in \bar{I}} H'_i = \emptyset \text{ if and only if } I \in \mathcal{I}.$$

□

Consider any set Q consisting of quorum literals. Q splits in a natural fashion into the union $Q' \cup Q''$ where Q' consists of positive quorum constraints in Q', and Q'' consists of negative quorum constraints in Q. Define $\mathcal{S}_Q = \{I \subseteq [m] : I \models Q\}$. An intersection of a filter and an ideal will be called a *segment*. Segments are characterized by the following property. A subfamily $\mathcal{S} \subseteq \mathcal{P}([m])$ is a segment if $X_1, X_2 \in \mathcal{S}$ and $X_1 \subseteq Y \subseteq X_2$ imply that $Y \in \mathcal{S}$. Bringing together Theorems 2.1 and 3.1 we get the following result.

Theorem 3.2 *The following conditions are equivalent.*

(a) *A family $\mathcal{S} \subseteq \mathcal{P}([m])$ is a segment in $\mathcal{P}([m])$.*

(b) *There is a set of quorum literals Q such that $\mathcal{S} = \mathcal{S}_Q$.*

(c) *There are families of sets $\mathcal{G} = \{G_1, ..., G_m\}$ and $\mathcal{H} = \{H_1, ..., H_m\}$ such that*

$$I \in \mathcal{S} \text{ if and only if } (\bigcap_{i \in I} G_i = \emptyset \text{ and } \bigcap_{i \in I} H_i \neq \emptyset).$$

(d) *There are families of sets $\mathcal{G}' = \{G'_1, ..., G'_m\}$ and $\mathcal{H}' = \{H'_1, ..., H'_m\}$ such that*

$$I \in \mathcal{S} \text{ if and only if } (\bigcap_{i \in I} G'_i = \emptyset \text{ and } \bigcap_{i \in I} H'_i = \emptyset).$$

□

4 Some issues related to expressive power

In the proof of Theorem 2.1 we have shown how a description of a filter by means of a set of positive quorum constraints can be used to construct a set of unitary quorum constraints determining the same filter. This transformation, in effect, can be treated as a kind of logical operation. Namely, when a filter is specified by a conjunction of a set Q of positive constraints, we constructed a set of unitary constraints Q' so that the conjunction of Q' specifies the same filter. Thus we may think about this transformation as a form of normal form computation.

Once this position is taken, namely that positive constraints are atoms of some logical language, it is natural to consider other formulas of that logical language, not only conjunctions of atoms. The characterization theorem for ideals can be interpreted in these terms as stating that ideals are determined by conjunctions of negated atoms, and our result on segments as a characterization of segments by means of sets of literals.

Notice that filters can be described by means of disjunctions of positive constraints. Indeed, take $\mathcal{C}$ consisting of all minimal elements of the filter $\mathcal{F}$. Then the formula $\varphi = \bigvee_{C \in \mathcal{C}} \langle C, |C| \rangle$ characterizes the filter $\mathcal{F}$. That is the family of those sets T that satisfy the formula φ coincides with $\mathcal{F}$.

Thus we noticed that a filter has a description both as a conjunction and as a disjunction of atoms. In fact, the fact that we deal with a filter is really related to the positive formulas. A formula is called *positive* if it is built out of atoms by means of conjunctions and disjunctions. The following is easily provable by induction on the length of a formula.

Proposition 4.1 *If φ is a positive formula then the family of those sets X that satisfy φ forms a filter.* □

We will denote that filter $\mathcal{F}_\varphi$ and we will say that φ *represents* a filter $\mathcal{F}$, if $\mathcal{F} = \mathcal{F}_\varphi$. Clearly, filters may have multiple representations. To see this, write any positive formula φ, take the filter $\mathcal{F}_\varphi$ and then write the representation of that filter by means of a conjunction of atoms and yet another as a disjunction of atoms. This raises the question whether the complexity of description of a filter can be somehow measured. To this end, let us assign to a formula φ the number $m(\varphi)$ of atoms occurring in φ. Call $m(\varphi)$ the *size* or *measure* of φ. Assign to a filter $\mathcal{F}$ its *measure* $\mu(\mathcal{F})$ by

$$\mu(\mathcal{F}) = \min\{m(\varphi) : \mathcal{F}_\varphi = \mathcal{F}\}.$$

In principle, the result of our Theorem 2.1 is that as concerns the descriptions of filters, all we need are conjunctions of atoms. But we will see that when it comes to the measure of complexity, the formulas we are allowed to use make a lot of difference.

To set up the stage, notice that Shamir's filter [9] consisting of all subsets of X of the size greater or equal than k has a very simple description, namely we can describe it by means of a single constraint. Thus this filter has the measure equal to 1.

Notice that whenever we define filters using formulas of our language, there is a hidden parameter - the set X itself. We will now construct a family of filters that have very complex descriptions if only the conjunctions are allowed, but each of them has the measure equal to 2. In fact, that filter has the following property: if only conjunctions of atoms are allowed then the shortest description is of the order $O(|\mathcal{A}|)$ where $\mathcal{A}$ is the largest antichain in $\mathcal{P}(X)$. Thus our construction will provide us with a filter that is truly pathological - description via the conjunction of atoms is huge, the measure drops to the least possible value, namely, 2, if disjunctions are also allowed (Shamir filters are the only filters with descriptions of size 1).

To this end, let X be of even size, $|X| = m = 2n$, $n \geq 1$, and let x_0 be a fixed element of X. Define $\mathcal{C}_{x_0} = \{C \subseteq X : |C| = n \text{ and } x_0 \in C\}$. Now,

let $Q = \{\langle C, 1\rangle : C \in \mathcal{C}_{x_0}\}$. The set of positive constraints Q determines a filter that we will denote $\mathcal{F}_{x_0}$.

It is easy to see that the size of Q is $\binom{m-1}{n-1}$. Thus our description of $\mathcal{F}_{x_0}$ as a conjunction of atoms is of the size $\binom{m-1}{n-1}$. Notice that the collection $\mathcal{C}_{x_0}$ forms an antichain. Moreover, the complement of any set from $\mathcal{C}_{x_0}$ does not belong to $\mathcal{F}_{x_0}$.

We will now show that any description of $\mathcal{F}_{x_0}$ by means of a conjunction of atoms must have the size that is greater than or equal to $\binom{m-1}{n-1}$.

To this end we will prove the following two results.

Proposition 4.2 *Let $\langle T, k\rangle$ be any constraint satisfied by all elements of $\mathcal{F}_{x_0}$. Then $k = 1$, and for some $C \in \mathcal{C}_{x_0}$, $C \subseteq T$.*

Proof: Case 1. $|T| < n$.

Then $Z = X \setminus T$ has size bigger than n. Hence Z must meet all constraints in Q because whenever $\langle C, 1\rangle \in Q$ then $|Z| + |C| > m$ and so $Z \cap C \neq \emptyset$ i.e. $Z \models \langle C, 1\rangle$. Hence $Z \in \mathcal{F}_{x_0}$. But then it must be the case that $Z \models \langle T, k\rangle$ which is a contradiction since $T \cap Z = \emptyset$. Thus this case is impossible.

Case 2. $|T| \geq n$.

Notice that $\{x_0\}$ belongs to the filter $\mathcal{F}_{x_0}$. Thus it must be the case that $\{x_0\} \models \langle T, k\rangle$. Then, obviously, $k = 1$, and $x_0 \in T$. But since $|T| \geq n$, T has a subset belonging to $\mathcal{C}_{x_0}$. □

Proposition 4.3 *If $\mathcal{F}$ is a filter defined by a set $R = \{\langle C, 1\rangle : C \in \mathcal{C}\}$ of unitary constraints such that $\mathcal{C}$ is an antichain then for every set of unitary constraints R' defining $\mathcal{F}$, $R \subseteq R'$.*

Proof: Suppose R' defines $\mathcal{F}$ but $\langle C_0, 1\rangle \notin R'$, for some $C_0 \in \mathcal{C}$. Since $X \setminus C_0 \notin \mathcal{F}$ because it does not satisfy the constraint $\langle C_0, 1\rangle \in R$, there is a constraint $\langle C_1, 1\rangle \in R'$ such that $C_1 \cap (X \setminus C_0) = \emptyset$, i.e. $C_1 \subseteq C_0$. Since $\langle C_0, 1\rangle \notin R'$, $C_1 \neq C_0$. Now, $(X \setminus C_1) \cap C_0 \neq \emptyset$ and, for every $C \in \mathcal{C}$, $C \neq C_0$, $(X \setminus C_1) \cap C \neq \emptyset$, for otherwise $C \subseteq C_1 \subseteq C_0$ contradicting the assumption that $\mathcal{C}$ is an antichain. Hence, $X \setminus C_1 \in \mathcal{F}$, a contradiction because $X \setminus C_1$ does not satisfy the constraint $\langle C_1, 1\rangle \in R'$. □

Now we are able to show that Q is the simplest description of $\mathcal{F}_{x_0}$ by means of conjunction. Indeed, by Proposition 4.2, any description of that

filter by means of conjunction of constraints Q' must consist of only unitary constraints. Since $\mathcal{C}_{x_0}$ is an antichain, Q is the simplest description of $\mathcal{F}_{x_0}$ by means of conjunction by Proposition 4.3.

In order to see that $\mathcal{F}_{x_0}$ has a description of size 2 when the disjunctions are allowed, we will now look at the minimal elements of the filter $\mathcal{F}_{x_0}$.

Proposition 4.4 *The following are all the minimal elements of* $\mathcal{F}_{x_0}$.

(a) $\{x_0\}$

(b) *all* $(n+1)$*-element subsets of* $X \setminus \{x_0\}$.

Proof: Clearly, $\{x_0\}$ is a minimal set belonging to our filter. Next, it is easy to see that each $(n+1)$-element subset A of $X \setminus \{x_0\}$ belongs to the filter because, for each $C \in \mathcal{C}_{x_0}$, $|A \cap C| = |A| + |C| - |A \cup C| \geq n+1+n-m = 1$. Next, proper subsets of such A have at most n elements and do not contain x_0. Therefore their complements contain subsets in $\mathcal{C}_{x_0}$. Hence proper subsets of such sets do not belong to the filter, and so all the sets listed in **(b)** are minimal in our filter.

Conversely, let C be a subset of X that is a minimal element of our filter. If C contains x_0 then $C = \{x_0\}$. Otherwise, C is included in the set $X \setminus \{x_0\}$. The requirement that C satisfies all constraints in Q reduces to the fact that C has nonempty intersection with all $(n-1)$-element subsets of $X \setminus \{x_0\}$. The minimal sets satisfying these conditions are precisely $(n+1)$-element subsets of $X \setminus \{x_0\}$. This shows that C is one of the sets in **(b)**. □

Now form the following formula φ

$$\varphi = \langle \{x_0\}, 1 \rangle \vee \langle X \setminus \{x_0\}, n+1 \rangle.$$

Clearly $m(\varphi) = 2$.

Proposition 4.5 *The formula* φ *determines the filter* $\mathcal{F}_{x_0}$.

Proof: By Proposition 4.4 the minimal elements of the filter $\mathcal{F}_{x_0}$ are: $\{x_0\}$ and all $(n+1)$-element subsets of $X \setminus \{x_0\}$. The first of these satisfies the first disjunct, the remaining ones the second one. Since every element of the filter $\mathcal{F}_{x_0}$ contains a minimal one, every element of our filter satisfies one of the disjuncts in φ, thus φ itself.

Conversely, if C is a filter determined by the formula φ, then two cases are possible. Either C satisfies the first disjunct, that is contains x_0. Then clearly C belongs to $\mathcal{F}_{x_0}$. Or C satisfies the second disjunct, and it contains some minimal element of $\mathcal{F}_{x_0}$. Thus also in this case $C \in \mathcal{F}_{x_0}$. □

Thus we see that, by definition $\mu(\mathcal{F}_{x_0}) \leq 2$. But it is easy to verify that $\mu(\mathcal{F}_{x_0}) \neq 1$, for $n \geq 2$. We leave a simple proof of this statement to the reader. Since a single negated constraint $\neg\langle T, k\rangle$ defines an ideal not a filter, we proved that our filter $\mathcal{F}_{x_0}$ has no description of length 1, and so $\mu(\mathcal{F}_{x_0}) = 2$.

Similarly we can find an example of a filter whose description via disjunctions is of order $O(|\mathcal{A}|)$ but there is a description via a conjunction of size 2.

Let $\mathcal{F}'_{x_0}$ be the filter whose minimal elements are precisely the members of $\mathcal{C}_{x_0} = \{C \subseteq X : |C| = n \text{ and } x_0 \in C\}$. One can check similarly as in the case of $\mathcal{F}_{x_0}$ that the shortest description of $\mathcal{F}'_{x_0}$ using disjunctions is the disjunction of the constraints $\langle C, n\rangle$, $C \in \mathcal{C}_{x_0}$, whose size is $O(|\mathcal{A}|)$. On the other hand $\mathcal{F}'_{x_0}$ is the access structure with respect to the set of quorum constraints $Q' = \{\langle\{x_0\}, 1\rangle, \langle X \setminus \{x_0\}, n-1\rangle\}$ of size 2.

In view of the examples the following extremal problem seems to be of interest. Given an integer m, find the maximum value of $\mu(\mathcal{F})$ over the set of all filters $\mathcal{F} \subseteq \mathcal{P}([m])$. With the notation $\mu(m) = \max\{\mu(\mathcal{F}) : \mathcal{F} \text{ is a filter in } \mathcal{P}([m])\}$, the problem is to investigate and estimate the values of $\mu(m)$.

In the arguments below we will use several times the following relationship between quorum constraints.

Proposition 4.6 *If $X_1 \subseteq X_2$ and $n_2 \leq n_1$, then*

$$I \models \langle X_1, n_1\rangle \text{ implies } I \models \langle X_2, n_2\rangle. \tag{1}$$

□

We now have the following property.

Proposition 4.7

$$\frac{1}{3m}\binom{m}{\lfloor \frac{m}{2} \rfloor} < \mu(m) \leq \binom{m}{\lfloor \frac{m}{2} \rfloor}$$

Proof: The inequalities are trivially true for $m = 1$ so assume that $m \geq 2$. It follows from Theorem 2.1(c) that every filter $\mathcal{F} \subseteq \mathcal{P}([m])$ can be

defined as a conjunction of unitary constraints $Q' = \{\langle C, 1\rangle : C \in \mathcal{C}\}$, for some $\mathcal{C} \subseteq \mathcal{P}([m])$. By Proposition 4.6 we may assume that $\mathcal{C}$ is an antichain so Sperner's lemma implies

$$\mu(\mathcal{F}) \leq |Q'| = |\mathcal{C}| \leq \binom{m}{\lfloor \frac{m}{2} \rfloor}.$$

Hence $\mu(m) \leq \binom{m}{\lfloor \frac{m}{2} \rfloor}$.

On the other hand consider nonempty antichains being subsets of the $\lfloor \frac{m}{2} \rfloor$th level of $\mathcal{P}([m])$, i.e. of $\mathcal{L} = \{C \in \mathcal{P}([m]) : |C| = \lfloor \frac{m}{2} \rfloor\}$. Clearly there are $2^{\binom{m}{\lfloor \frac{m}{2} \rfloor}} - 1$ of them and each such antichain defines a filter whose minimal elements are the elements of the antichain. Therefore we have at least $2^{\binom{m}{\lfloor \frac{m}{2} \rfloor}}$ distinct filters in $\mathcal{P}([m])$. Notice that the number of possible constraints $\langle C, t\rangle$, $C \subseteq [m]$, $1 \leq t \leq |C|$ is equal to $\sum_{i=1}^{m} i\binom{m}{i} = m2^{m-1}$. Let us count all possible formulas of size at most k (with k atoms) composed of constraints and the symbols $\wedge$ and $\vee$. Assume that they are written in the reverse Polish notation so they are strings of length at most $2k-1$ over an alphabet of $m2^{m-1}+2$ symbols. The number of such strings is $\sum_{i=1}^{k}(m2^{m-1}+2)^{2i-1}$ so we have at most

$$\sum_{i=1}^{k}(m2^{m-1}+2)^{2i-1} \leq \sum_{i=1}^{k}(m2^m)^{2i-1} = 2 \cdot \frac{(m2^m)^{2k}-1}{m2^m-1} < (m2^m)^{2k}$$

$$= 2^{2k(m+\log m)} \leq 2^{3km},$$

formulas, for $m \geq 2$.

If $k \leq \frac{1}{3m}\binom{m}{\lfloor \frac{m}{2} \rfloor}$ then $2^{3km} \leq 2^{\binom{m}{\lfloor \frac{m}{2} \rfloor}}$ so there is a filter $\mathcal{F}$, for which $\mu(\mathcal{F}) > \frac{1}{3m}\binom{m}{\lfloor \frac{m}{2} \rfloor}$. This counting argument demonstrates that $\mu(m) > \frac{1}{3m}\binom{m}{\lfloor \frac{m}{2} \rfloor}$. □

5 Algorithms

From the point of view of applications it is important to design an algorithm which, for an input set of constraints Q constructs a family of sets $\mathcal{G}_Q$ satisfying the condition $(\star)$ (Condition $(\star)$ was introduced after the proof of Theorem 2.1). A general idea of such an algorithm is described in the proof of the implication (a) $\Rightarrow$ (e) in Theorem 2.1. To construct the family $\mathcal{G}_Q$ we need to find all maximal elements of the ideal $\mathcal{I}_Q = \mathcal{P}([m]) \setminus \mathcal{F}_Q$. We shall follow this approach in our Algorithm 2.

Let us start, however, with a simpler (but in most cases slower) algorithm which makes use of an observation that in the proof of the implication (a) $\Rightarrow$ (e) in Theorem 2.1 we can replace the set $\mathcal{M}_{\mathcal{I}}$ of maximal elements of the ideal $\mathcal{I}$ by the ideal $\mathcal{I}$ itself. Let $M = \{a_1, ..., a_q\}$, where $q = |\mathcal{I}|$. Define $G_i = \{a_j \in M : i \in A_j\}$, $1 \leq i \leq m$. One can easily check that the family of sets $\mathcal{G}_Q = \{G_1, ..., G_m\}$ defined this way satisfies $(\star)$. This observation leads us to the first (generic) algorithm.

Let $Q = \{\langle A_1, k_1\rangle, ..., \langle A_t, k_t\rangle\}$ be our input set of constraints.

Algorithm 1

```
1   I := ∅;
2   for all subsets B ⊆ [m] do
3     { for i = 1, ..., t do
4       { if |B ∩ A_i| < k_i then I := I ∪ {B}}
5   end for (2) };
6   G_1, ..., G_m := ∅;
7   for every B = {i_1, ..., i_{s_B}} ∈ I do
8     { for j = 1, ..., s_B do
9       { G_{i_j} := G_{i_j} ∪ {a_B}}
10  end for (7) };
11  stop;
```

In lines 1-5 of this algorithm we find all sets in $[m]$ which are not access structures with respect to the input set of constraints Q, i.e. we find all the elements of the ideal $\mathcal{I}_Q$. This ideal is then used to construct the desired family $\mathcal{G}_Q$ in lines 6-10.

Let us discuss time complexity of Algorithm 1 by counting the number of executions of the most embedded loops. The number of passes of the loop in lines 3-4 is equal to $t2^m$. The loop in lines 8-9 is executed

$$\sum_{B \in \mathcal{I}} s_B \leq m|\mathcal{I}| \leq m2^m$$

times. Hence the worst case running time of Algorithm 1 is $O((m+t)2^m)$. In practice, if the number t of quorum constraints is small and the constraints are not very restrictive (i.e., the ideal $\mathcal{I}$ is small) then the actual running time of this algorithm may be quite satisfactory.

There is another disadvantage of Algorithm 1, however. The sets of the family $\mathcal{G}_Q$ constructed by this algorithm may be very large. For example, if

our set of quorum constraints is $Q = \{\langle [m], m\rangle\}$ then $\mathcal{I}_Q = \mathcal{P}([m]) \setminus \{[m]\}$ and each set G_i constructed by Algorithm 1 has $2^m - 1$ elements. On the other hand, for an "optimal" family $\mathcal{G}_Q = \{G_1, ..., G_m\}$ with the required property $(\star)$, $G_i = [m] \setminus \{i\}$, $i = 1, ..., m$, so $|G_i| = m - 1$.

Therefore we shall introduce another, subtler, algorithm which requires to use only the maximal elements of the ideal $\mathcal{I}_Q$ in the construction of a family $\mathcal{G}_Q$ instead of the entire ideal $\mathcal{I}$.

We shall need several auxiliary facts. The first of them will be used to reconstruct the set of maximal elements of the ideal $\mathcal{I}_Q$ out of the set of unitary quorum constraints Q.

Proposition 5.1 *Let $Q = \{\langle X_1, 1\rangle, \ldots, \langle X_p, 1\rangle\}$ be a set of unitary quorum constraints such that $X_1, \ldots, X_p$ form an antichain in $\mathcal{P}([m])$. Then the sets $Y_j = [m] \setminus X_j$, $1 \leq j \leq p$, are precisely the maximal elements of $\mathcal{I}_Q$.*

Proof: If M is a maximal element of $\mathcal{I}_Q$ then $X_j \cap M = \emptyset$ for some j, $1 \leq j \leq p$. So $M \subseteq [m] \setminus X_j$. If there is an x so that $x \in ([m] \setminus X_j) \setminus M$ then $X_j \cap (M \cup \{x\}) = \emptyset$ so $M \cup \{x\} \in \mathcal{I}_Q$. This contradicts the maximality of M in $\mathcal{I}_Q$. Hence $M = [m] \setminus X_j$.

Conversely, for every j, $1 \leq j \leq p$, $[m] \setminus X_j \in \mathcal{I}_Q$. Let $x \in X_j$. Then

$$X_j \cap (([m] \setminus X_j) \cup \{x\}) = \{x\} \neq \emptyset.$$

Moreover, for $j_1 \neq j_2$,

$$X_{j_1} \cap (([m] \setminus X_{j_2}) \cup \{x\}) \supseteq X_{j_1} \cap ([m] \setminus X_{j_2}) \neq \emptyset$$

because otherwise $X_{j_1} \subseteq X_{j_2}$ contradicting to our assumption that the sets X_j, $1 \leq j \leq p$ form an antichain. Thus $([m] \setminus X_j) \cup \{x\}$ does not belong to $\mathcal{I}_Q$ for any x, that is $[m] \setminus X_j$ is a maximal element in $\mathcal{I}_Q$, as required. □

The next proposition allows us to reduce an arbitrary quorum constraint to a set of unitary quorum constraints.

Proposition 5.2 *Let $q = \langle X, k\rangle$, $k \geq 2$, be a quorum constraint. Then $I \models q$ if and only if for every $x \in X$, $I \models \langle X \setminus \{x\}, k - 1\rangle$.*

Proof: Assume $I \models q$. Choose $x \in X$. If $x \in I$ then I contains at least $k - 1$ elements of X different from x. Thus $I \models \langle X \setminus \{x\}, k - 1\rangle$. If $x \notin I$, then I contains at least k elements of $X \setminus \{x\}$, and, a fortiori, $k - 1$ of them.

Conversely, assume that $I \not\models q$. Then $|I \cap X| \leq k-1$. If $I \cap X = \emptyset$ then since $k \geq 2$, $I \cap (X \setminus \{x\}) = \emptyset$ for any $x \in X$ and in particular $I \not\models \langle X \setminus \{x\}, k-1 \rangle$. Otherwise, select $x \in I \cap X$. Then $|I \cap (X \setminus \{x\})| \leq k-2$, thus $I \not\models \langle X \setminus \{x\}, k-1 \rangle$. □

Iterating Proposition 5.2 we get the following result.

Corollary 5.3 *Let $q = \langle X, k \rangle$, where $k \geq 2$, be a quorum constraint. Then $I \models q$ if and only if for every $(k-1)$-element subset Y of set X, $I \models \langle X \setminus Y, 1 \rangle$.* □

We now have all the facts necessary for writing our Algorithm 2.

Let $Q = \{\langle A_i, k_i \rangle : i = 1, ..., p\}$ be a set of quorum constraints. Since, by (1), $I \models \langle A, n_1 \rangle$ implies $I \models \langle A, n_2 \rangle$, for $n_1 \geq n_2$, we can assume without loss of generality, that $A_i \neq A_j$, for $i \neq j$. Define, for every $A \in \mathcal{P}([m])$,

$$q_A = \begin{cases} k & \text{if } \langle A, k \rangle = \langle A_i, k_i \rangle \in Q \\ 0 & \text{otherwise.} \end{cases}$$

In the algorithm we represent the input set of quorum constraints Q by the set $\{q_A : A \in \mathcal{P}([m])\}$. Let $\{a_B : B \in \mathcal{P}([m])\}$ be a set of elements which will contain all constructed sets $G_1, ..., G_m$ in $\mathcal{G}_Q$.

Algorithm 2

```
1   for i = m, m − 1, ..., 1 do
2     { for all subsets A ⊆ [m] such that |A| = i do
3       { if q_A > 1 then
4         { for every a ∈ A do q_{A−{a}} := max(q_{A−{a}}, q_A − 1) ;
5           q_A := 0 }
6     end for (2) }
7   end for (1) };
8   V := ∅;
9   M := ∅;
10  for i = 1, 2, ..., m do
11    { for all subsets A ⊆ [m] such that |A| = i do
12      { if A ∈ V then for every a ∈ [m] − A do V := V ∪ {A ∪ {a}};
13      if (A ∉ V and q_A = 1) then do
14        { V := V ∪ {A};
15        M := M ∪ {[m] − A};
16        for every a ∈ [m] − A do
17          { V := V ∪ {A ∪ {a}} }
```

```
18        end if (13) }
19      end for (11) }
20    end for (10) };
21    G_1, ..., G_m := ∅;
22    for every B = {i_1, ..., i_{s_B}} ∈ M do
23      { for j = 1, ..., s_B do
24          { G_{i_j} := G_{i_j} ∪ {a_B} }
25    end for (22) };
26    stop;
```

For a detailed discussion of the correctness of this algorithm we refer the reader to [6].

It was shown in [6] that the running time of Algorithm 2 is $O(m2^m)$. On the other hand (see [6]), for $Q = \{\langle [m], \lfloor \frac{m}{2} \rfloor \rangle\}$ to print the family $\mathcal{G}_Q$ alone we need $m\binom{m-1}{\lfloor \frac{m}{2} \rfloor - 2} = \Omega(m^{1/2}2^m)$ time.

For negative quorum constraints we can prove results dual to Propositions 5.1 and 5.2 and Corollary 5.3. Consequently, we can design an algorithm for finding, for a given set of negative quorum constraints Q, a family $\mathcal{H} = \{H_1, \ldots, H_m\}$ so that for every $I \subseteq [m]$, $I \models Q$ if and only if

$$\bigcap \{H_i : i \in I\} \neq \emptyset.$$

Below we sketch how it can be achieved.

The following three facts can be shown in an analogous way as their counterparts for positive quorum constraints.

Proposition 5.1' *Let $Q = \{\neg\langle X_1, |X_1|\rangle, \ldots, \neg\langle X_p, |X_p|\rangle\}$ be a set of antiunitary quorum constraints such that $X_1, \ldots, X_p$ form an antichain in $\mathcal{P}([m])$. Then the sets X_j, $1 \leq j \leq p$, are precisely the minimal elements of the filter $\mathcal{F}_Q = \mathcal{P}([m]) - \mathcal{I}_Q$.* □

Proposition 5.2' *Let $q = \neg\langle X, k\rangle$, $k \leq |X| - 1$, be a negative quorum constraint. Then $I \models q$ if and only if for every $x \in X$, $I \models \neg\langle X \setminus \{x\}, k\rangle$.* □

Corollary 5.3' *Let $q = \neg\langle X, k\rangle$, where $k \leq |X| - 1$, be a negative quorum constraint. Then $I \models q$ if and only if for every k-element subset Y of X, $I \models \neg\langle Y, k\rangle$.* □

Below we introduce Algorithm 3. It finds, for a given input set of negative quorum constraints Q on $[m]$, a family $\mathcal{H}_Q = \{H_1, ..., H_m\}$ so that for every $I \subseteq [m]$, $\bigcap_{i \in I} H_i \neq \emptyset$ if and only if $I \models Q$.

Let $Q = \{\neg\langle A_i, k_i\rangle : i = 1, ..., p\}$. As in Algorithm 2 we can assume that $A_i \neq A_j$, for $i \neq j$. Similarly, we define, for every $A \in \mathcal{P}([m])$,

$$q_A = \begin{cases} k & \text{if } \neg\langle A, k\rangle = \neg\langle A_i, k_i\rangle \in Q \\ 0 & \text{otherwise.} \end{cases}$$

Algorithm 3 works in a very similar way to Algorithm 2. Only the lines 3, 4, 13 and 15 in Algorithm 3 are different from the corresponding lines in Algorithm 2.

Algorithm 3

```
1   for i = m, m − 1, ..., 1 do
2     { for all subsets A ⊆ [m] such that |A| = i do
3       { if q_A < |A| + 1 then
4         { for every a ∈ A do q_{A−{a}} := min(q_{A−{a}}, q_A) ;
5           q_A := 0 }
6     end for (2) }
7   end for (1) };
8   V := ∅;
9   M := ∅;
10  for i = 1, 2, ..., m do
11    { for all subsets A ⊆ [m] such that |A| = i do
12      { if A ∈ V then for every a ∈ [m] − A do V := V ∪ {A ∪ {a}};
13      if (A ∉ V and q_A = |A| + 1) then do
14        { V := V ∪ {A};
15        M := M ∪ {A};
16        for every a ∈ [m] − A do
17          { V := V ∪ {A ∪ {a}} }
18      end if (13) }
19    end for (11) }
20  end for (10) };
21  G_1, ..., G_m := ∅;
22  for every B = {i_1, ..., i_{s_B}} ∈ M do
23    { for j = 1, ..., s_B do
24      { G_{i_j} := G_{i_j} ∪ {a_B} }
25  end for (22) };
26  stop;
```

Clearly, Algorithm 3 has the same running time as Algorithm 2, i.e. $O(m2^m)$.

6 Conclusions

In this paper we give several characterizations of filters, ideals and segments in Boolean lattices of subsets of a finite set ordered by inclusion. One of them uses so-called quorum constraints (or their negations) and characterizes filters (respectively ideals and segments) as a families of sets satisfying these constraints. This is a natural way of defining these families motivated by applications. Another characterization allows us to test membership of sets in filters (respectively ideals and segments) by verifying whether or not some specific sets have an empty intersection. This characterization has important consequences in a problem of distribution of keys in cryptography.

Several problems remain open. First, an algorithm of verification if a given set belongs to a filter that we give is computationally expensive. It would be important, especially from the point of view of applications, to find a faster algorithm solving this problem.

Second, in our paper we consider only a certain type of constraints. It seems that there are more kinds of constraints that are worth to examine. For example, a constraint requiring that the putative set I contains more elements of some set Y than of some other set Z can not be expressed as a conjunction of quorum literals that we consider in this paper. In fact, it may happen that the family of sets I satisfying such a constraint is not a segment.

The third problem is a question how to find, for a given filter, a minimal set of quorum constraints which determine the filter. Similar problems can be formulated for ideals and segments. A general solution of this problem is hard because in a special case it reduces to the problem of covering edges of a graph by a minimum number of cliques. The decision version of this problem is known to be NP-complete (see [2]).

References

[1] C. Blundo, A. De Santis, L. Gargano and U. Vaccaro, Secret sharing with veto capabilities, in *Algebraic Coding, First French-Israeli Work-*

shop (ed. G. Cohen, S. Litsyn, A. Lobstein and G. Zemor), *Lecture Notes in Computer Science* 781 (1993), 82-89.

[2] M. R. Garey, D. S. Johnson, *Computers and Intractability*, Freeman, New York, 1997.

[3] *Handbook of Combinatorics* (ed. R. L. Graham, M. Grötschel, L. Lovász), Elsevier, Amsterdam, 1995.

[4] E. Kranakis, *Primality and Cryptography*, Wiley, 1986.

[5] C. L. Liu, *Introduction to Combinatorial Mathematics*, McGraw-Hill, New York, 1968.

[6] Z. Lonc and M. Srebrny, A Combinatorial Algorithm for Sharing a Key, in *Advanced Computer Systems* (ed. J. Soldek and J. Pejaś), 391-404, 8th International Conference ACS'2001, Kluwer, 2001.

[7] M. Mignotte, How to Share a Secret, *Lecture Notes in Computer Science*, vol. 149, 371–375, Springer–Verlag, 1983.

[8] S. Obana and K. Kurosawa, Veto is impossible in secret sharing schemes, *Information Processing Letters* 58 (1996), 293-295.

[9] A. Shamir, How to share a secret, *Communications of the ACM*, 22 (1979), 612–613.

[10] D. R. Stinson, *Cryptography, Theory and Practice*, CRC Press 1995.