Nazarbayev University

School of Science and Technology

Mathematics Department

Optimization of convex geometries: component quadratic and general

Author:
Zhanbota Myrzakul

Supervisor:
Prof. Kira Adaricheva

Second reader:
Prof. Rustem Takhanov

2016

Abstract

In this Capstone Project, we worked with a class of closure systems called *convex geometries*, which are closure systems with a closure operator that satisfies the anti-exchange property. We first looked at the result of optimization algorithm of *component quadratic* systems, which are discussed in [4], and reproved it for the case of convex geometries. We then investigated the following question: if a convex geometry is given by a set of implications, is it possible to find its optimum basis in polynomial time when the convex geometry does not have particular properties (for instance, not component quadratic)?

1 Introduction

Let S be a set, and ϕ be a closure operator. If $A, B \in 2^S$, the relation $B \subseteq \phi(A)$ between A and B in a closure system $\langle S, \phi \rangle$ can be written in the form of an implication $A \to B$. Thus the closure system $\langle S, \phi \rangle$ can be given by the set of implications $\Sigma_\phi = \{A \to B : A \subseteq S \text{ and } B \subseteq \phi(A)\}$. Many subsets of Σ_ϕ may define ϕ. The set Σ_ϕ is called an *implicational basis*. Vice versa, given a set of implications Σ, one can generate a closure operator $\phi_\Sigma : 2^S \to 2^S$ with respect to it, which is defined as follows:

$$\phi_\Sigma(Y) = \cap\{Z \in 2^S : Y \subseteq Z \text{ and for every } (A \to B) \in \Sigma \text{ if } A \subseteq Z \text{ then } B \subseteq Z\}.$$

A set A is called a *premise* and a set B is called a *conclusion* of the implication $A \to B$. In fact, several facets of closure operators were introduced. Some examples are: logical formulas, directed hypergraph, and Boolean functions. Implications are one of these facets. They have many applications in the fields such as education, artificial intelligence, and databases. There exist several types of bases for a closure system such as canonical basis, D-basis, canonical direct basis, etc. If $\Sigma = \{X_i \to Y_i : 1 \leq i \leq k\}$ is an implicational basis for a closure system $\langle S, \phi \rangle$, then the *size* of Σ is defined as $s(\Sigma) = |X_1| + |X_2| + ... + |X_k| + |Y_1| + |Y_2| + ... + |Y_k|$. An *optimum basis* of a closure system $\langle S, \phi \rangle$ is a basis with the minimal size among all possible implicational bases of $\langle S, \phi \rangle$. By optimizing an implicational basis Σ one will have several benefits such as saving time for working with a closure system, carrying all information about a closure system in a compact way. Unfortunately, it was shown in [3, 11] that, in general, an optimum basis cannot be obtained from any basis in polynomial time.

However, there are several approaches taken to deal with this problem. First approach is that a basis can be reduced to give another basis with less size, but that basis will not be an optimum one. Second approach is to study and describe subclasses of closure systems for which effective optimization of a given basis is possible. By effective optimization we mean the optimization in polynomial time. One of the interesting classes for studies of optimization is a class of convex geometries. Convex geometries are closure systems whose closure operator satisfies the *anti-exchange property.* Algorithms for effective optimization of several subclasses of convex geometries were already obtained such as optimization procedure for affine convex geometries [1, 7]. The question of optimization of convex geometries in general without satisfying additional properties was raised in [1, 7]. There is a class of closure systems called *component quadratic* (CQ) which are first introduced in [4]. In [1] it was shown that there are examples of convex geometries that are not CQ, and that there are examples of CQ closure systems which are not convex geometries. In this paper, we worked on optimization of convex geometries of CQ type, and we have done some programming work towards optimization of convex geometries in general.

The first part of this thesis is devoted to the convex geometries which are CQ. More precisely, the proof of the effective optimization algorithm for CQ closure systems is reconstructed for CQ convex geometries (the original proof for general CQ systems is given in [4]). In the second part, the goal is to find an algorithm to optimize convex geometries in general, particularly, non-CQ. For that purpose, some programming work was done and shown with several hypotheses. However, before introducing the main work, all necessary definitions and concepts are revised.

2 Definitions and Preliminary Statements

2.1 General concepts

Closure operator $\phi : 2^S \to 2^S$ is called a *closure operator* on a set S, if:
1) $Y \subseteq \phi(Y)$ (ϕ is increasing)
2) $Y \subseteq Z \Rightarrow \phi(Y) \subseteq \phi(Z)$ (ϕ is monotone)
3) $\phi(\phi(Y)) = \phi(Y)$ (ϕ is idempotent)
The set S, called a *base set*, together with ϕ forms a *closure system*.

Alignment Let S be a set. Then a family $\mathcal{F}$ of subsets of S is called an *alignment* of S, if $\mathcal{F}$ has the following properties:
1) $S \in \mathcal{F}$.
2) $Z, Y \in \mathcal{F} \Rightarrow Z \cap Y \in \mathcal{F}$.
The elements of an alignment are called *closed sets*.

Proposition 1 Given a closure operator ϕ and an alignment $\mathcal{F}$, one can build a relation between them.
(1) For any closure system $\langle S, \phi \rangle$, the family $\mathcal{F} = \{Y \subseteq S : Y = \phi(Y)\}$ is an alignment.
(2) If $\mathcal{F} \subseteq 2^S$ is an alignment, then for a set Y, $\phi(Y) = \cap\{Z \in \mathcal{F} : Y \subseteq Z\}$.
(3) The alignment and closure operator obtained in (1) and (2) are equivalent. More precisely, given a closure operator ϕ, one can build an alignment $\mathcal{F}_\phi$ with respect to it. Then from this alignment one can generate a closure operator $\phi_{\mathcal{F}_\phi}$, which is equivalent to the original closure operator ϕ. The same is true, if one starts with a given alignment $\mathcal{F}$, and finally obtains $\mathcal{F}_{\phi_\mathcal{F}}$.

Properties of a closure operator and alignment Let $\langle S, \phi \rangle$ be a closure system. ϕ is said to satisfy the *anti−exchange property*, if for a closed set A of S (i. e. $\phi(A) = A$) and $x, y \notin A$ such that $x \neq y$, $x \in \phi(A \cup \{y\}) \Rightarrow y \notin \phi(A \cup \{x\})$.
There is an equivalent property of an alignment. If $\langle S, \mathcal{F} \rangle$ is a closure system, then the equivalent property of $\mathcal{F}$ to the anti-exchange property of ϕ is the following: for $A \in \mathcal{F}, A \neq S$, there exists a point $x \in S \setminus A$ such that $A \cup \{x\} \in \mathcal{F}$.

Convex geometry A system $\langle S, \phi \rangle$ is called a *convex geometry*, if ϕ is a closure operator, $\phi(\emptyset) = \emptyset$ and ϕ satisfies the anti-exchange property. Equivalently, a system $\langle S, \mathcal{F} \rangle$ is a *convex geometry*, if $\mathcal{F}$ is an alignment of S with the above equivalent property.

Let us give an example of a convex geometry defined by an alignment.
Example 1 Let $S = \{a, b, c, d\}$. Define a convex geometry on S through an alignment $\mathcal{F} = \{\{a\}, \{b\}, \{c\}, \{d\}, \{a, b\}, \{b, c\}, \{c, d\}, \{a, b, c\}, \{b, c, d\}, \{a, b, c, d\}\}$. It is easy to check that $\mathcal{F}$ indeed satisfies the above mentioned property.

Implicational basis Let $\langle S, \phi \rangle$ be a closure system. An *implicational basis* of $\langle S, \phi \rangle$, Σ, is a set of ordered pairs $(X_i, Y_i) \in 2^S \times 2^S$ with $Y_i \neq \emptyset$. If $A, B \in 2^S$ and $B \subseteq \phi(A)$, this relation between subsets $A, B \subseteq S$ can be written in the form of an implication $A \to B$. Thus the closure system $\langle S, \phi \rangle$ can be given by an implicational basis $\Sigma_\phi = \{A \to B : A \subseteq S \textit{ and } B \subseteq \phi(A)\}$.

Vice versa, a closure operator can be defined through a given set of implications Σ, which was given in the Introduction part. In practice, the following proposition is used for determining if an element is in the closure of a set.
Proposition 2 $z \in \phi_\Sigma(Y)$ if and only if there exists a sequence $(A_i \to B_i), i < k$, implications in Σ such that $A_1 \subseteq Y, A_2 \subseteq Y \cup B_1, ..., A_k \subseteq Y \cup B_1 \cup ... \cup B_{k-1}$ and $z \in B_k$. This sequence forms a *path* from any element of A_1 to any element of B_k.

Example 2 Let us revisit the convex geometry in Example 1. The same convex geometry can be given by the set of implications $\Sigma = \{ad \to bc, ac \to b, bd \to c\}$, and the size of Σ is $s(\Sigma) = 10$.

Minimum basis A basis Σ is called *minimum*, if the cardinality of Σ (i.e. the number of implications in Σ) is smallest among all possible bases defining a closure system. A well-known minimum basis for closure systems is the *canonical basis* [11, 12].

Extreme points of a set A set of *extreme points* of a closed set A, i. e. $\phi(A) = A$, is the set $ex(A) = \{a \in A : \phi(A \setminus a) \subset \phi(A)\}$.

Theorem A closure system $\langle S, \phi \rangle$ is a convex geometry if and only if for a closed subset $Y \subseteq S$, $\phi(Y) = \phi(ex(Y))$. [6]

Quasi-order Let $\sqsubseteq$ be a binary relation on a set S. Now take its transitive closure and reflexive closure. Let that new relation be denoted as $\sqsubseteq^*$. Then $\sqsubseteq^*$ is a quasi-order. Any quasi-order is represented as a partial order of equivalence classes. We will now construct such a partial order.

Partition Let S be a given set. Define an equivalence relation $\sim$ on S as follows:
$a \sim b \Leftrightarrow a \sqsubseteq^* b \sqsubseteq^* a$.
This equivalence relation produces equivalence classes on S, and the set of these equivalence classes is denoted as $S/\sim$. Define a partial order $\leq$ on $(S/\sim, \leq)$ as the following: $(a/\sim) \leq (b/\sim) \Leftrightarrow a \sqsubseteq^* b$. Thus, $(S/\sim, \leq)$ is a poset.

2.2 Bases of convex geometries

For any closure system, the following is known:
1. Any optimum basis is optimization of the canonical basis, i.e., the shortening of the premises and conclusions of the implications of the canonical basis. See, for example [2].
2. For any $Y = \phi(Y) \subset S$, the subset $\Sigma(Y) = \{(A \to B) \in \Sigma : A, B \subseteq Y\}$ is an *exclusive* set of implications. If Σ is optimum for $\langle S, \phi \rangle$, then $\Sigma(Y)$ is optimum for $\langle Y, \phi_Y \rangle$, where ϕ_Y is the restriction of ϕ on Y. See the definition of an exclusive set of implications and

Theorem 5.7 in [5].

Moreover, for any optimum Σ of a convex geometry $\langle S, \phi \rangle$, the following is known:
3. For any $(X \to Y)$ in the optimum basis, $X = ex(\phi(X))$. Extreme points of $\phi(X)$ can be found in polynomial time [9].
4. For any two implications $X_1 \to Y_1, X_2 \to Y_2 \in \Sigma$, $\phi(X_1) \neq \phi(X_2)$. This is because convex geometries are closure systems with unique critical sets [2]. Thus, applying the observation in 2, the optimization can be done by one implication at a time, increasing height of corresponding closure in the alignment.
5. For a binary implication $(x \to Y) \in \Sigma_b$, one needs to include in optimum $Y = ex(\phi(x) \setminus \{x\})$ [1, 2].

3 Optimization of CQ convex geometries

For our purposes, we define a quasi-order in the following way.
Let Σ be a set of implications defining a convex geometry on S. We define $b \sqsubseteq a$, if $a \in A$, $b \in B$ for $A \to B \in \Sigma$. Then, as in the section 2.1, we build a quasi-order $\sqsubseteq$ and make a partition of the set S into equivalence classes by a binary relation $\sim$ as defined above. These equivalence classes on S we call *components*. Then we define a partial order $\leq$ on the set of all components as given above.

CQ systems were introduced in [4] in the form of Horn Boolean functions. A closure system $\langle S, \phi \rangle$ is said to be *component quadratic* (CQ), if there exists a basis Σ with such implications $B \to a$ that B contains no more than one element from the component $a/\sim$, where $a/\sim$ is defined as above.

Let us give an example of a CQ convex geometry.
Example 3 Assume a system $\langle S, \phi \rangle$ is a CQ convex geometry, and Σ is the canonical basis for this system with optimized left sides. Let $S = \{x, y, a, b, c, d\}$, where x,y are extreme points of S. The canonical basis Σ of this system consists of the following implications:
$xy \to abcd$
$xb \to cd$
$ya \to b$
$yd \to b$
$ac \to d$
The components derived from Σ are $\{a\}$ and $\{b, c, d\}$. Since all implications in Σ that have one element from the component $\{b, c, d\}$ in the conclusion, have only one element from this component in the premise, $\langle S, \phi \rangle$ is CQ.

Let $\langle S, \phi \rangle$ be a quadratic component convex geometry with a basis Σ. Suppose that all implications in Σ are already optimized except the last one $X \to A$, which is non-binary, i.e. $|X| > 1$, and such that $\phi(X) = S$, i.e. $A = S/X$. We need to optimize $X \to A$ by $X \to A^*$, where $A^* \subset A$.

Let $\sqsubseteq^*$ be a quasi-order defined on A with respect to Σ, and let $\langle A/\sim, \leq \rangle$ be a naturally defined poset of components on A. Also define a property *depth* of a component $C \in A/\sim$ as 0, if the component is maximal in the poset $\langle A/\sim, \leq \rangle$, and as k if the maximal chain in $\langle A/\sim, \leq \rangle$, with C being minimal element, has $k+1$ elements. Having only extreme points of set S on the left side of the implication $X \to A$ gives us optimized left side, due to the result in [10]. The need to optimize each $X_i \to A_i$ at greater depths before optimizing the implication $X \to A$ is shown in the theorem 5.7 in [5]. So, here we assume that all such $X_i \to A_i$ at depths greater than the depth of $X \to A$ are already optimized.

Before discussing the procedure of optimization we need to introduce the following definition.

Strong implication for a component Let C be a component of $A\backslash \sim$. We call an implication $(U \to V) \in \Sigma$ *strong* for a component C, if:
1. V has at least one element from C.
2. U does not have any element from C.

Procedure for optimization To optimize $X \to A$ by $X \to A^*$, include into A^* elements from A using the following rule:
1. One element from each maximal component.
2. One element from a non-maximal component when there is no strong implication for it.

Proposition 3 Let $\langle S, \phi \rangle$ be a component quadratic convex geometry with a basis Σ. Suppose that all implications in Σ are already optimized and we need to optimize a non-binary implication $X \to A$ such that $\phi(X) = S$. If $X \to A$ is optimized by $X \to A^*$ using the above procedure, then one obtains the basis that defines the same convex geometry. Moreover, this basis is optimum.

Proof Before applying the rules of the proposed procedure, we need to clarify the set on which components that need to be considered are taken. Some elements of the set A on the right of the implication $X \to A$ could appear as a conclusion of the rest of implications (i. e. implications except $X \to A$). Hence we do not need the last implication $X \to A$ to obtain those elements. Moreover, if at least element of a component appears in this way, then the whole component can be obtained without the implication $X \to A$. For this reason, we work only on the set $A \setminus A^1$, where $A^1 = \cup\{$*components obtained by* $\Sigma \setminus (X \to A)\}$.

Let C_k^j be a notation for the jth component at the *depth* $= k$, where $1 \leq k \leq n$ and $j \in J \subset N$. At depth=0 since C_0^i's are maximal in $A/\sim$, according to our procedure we include one element from each C_0^i, and denote them by a_0^i. The basis which is obtained by replacing the component with one of its element is equivalent to the original basis, which is shown in the following claim.

Claim 1 At *depth* $= 0$, if we replace $X \to A$ by $X \to A \setminus \{\cup C_0^i\} \cup \{a_0^i\}$, then $\Sigma_0 = \Sigma \setminus \{X \to A\} \cup \{X \to A_0\}$, where $A_0 = A \setminus \{\cup C_0^i\} \cup \{a_0^i\}$, is equivalent to Σ. Equivalently, $X \to A$ follows from Σ_0.

Proof of Claim 1 Let C_0^s be a component at depth=0, and assume that C_0^s has at least two elements. Then since $\langle S, \phi \rangle$ is CQ and C_0^s is a component, there is a path connecting any two elements in C_0^s. Suppose a_0 is picked from C_0^s, and a_0^* is any element in C_0^s not equal to a_0. Because a_0 and a_0^* are in the same component, there is a path between them through implications in Σ, which is in the following form:

$X_1 a_0 \to A_1 a_0^1$, $X_2 a_0^1 \to A_2 a_0^2$, ... , $X_k a_0^k \to A_k a_0^*$. Since $a_0^* \in A^*$, implications $X \to A^*$ and $X_1 a_0 \to A_1 a_0^1$ will give $a_0^1 \in \phi_{\Sigma_0}(X)$. Applying the same procedure further, we obtain that $a_0^2, ..., a_0^* \in \phi_{\Sigma_0}(X)$. Since a_0^* was arbitrary, we get $C_0^s \subseteq \phi_{\Sigma_0}(X)$.

In fact, we need to add at least one element from each maximal component to obtain the whole component through an implicational basis. This fact is stated in the following claim.

Claim 2 At $depth = 0$ addition of at least one element from a maximal component into A^* is necessary when optimizing $X \to A$ by $X \to A^*$. In particular, since addition of one element only is enough to get the component, this results in the optimum implicational basis.
Proof of Claim 2 Since at $depth = 0$ all components are maximal, there is no component such that there is an implication where this component implies maximal components. Hence we cannot obtain elements from maximal components through any implication except the one in which the left side is the set of extreme points of the closure system. To have an optimum basis we add only one element from each maximal component.

By doing the same procedure for each C_0^i at $depth = 0$, the closure of X, $\phi_{\Sigma_0}(X)$, will include every C_0^i.
Now suppose we have optimized $(X \to A)$ with $(X \to A_{n-1} \cup C) \in \Sigma_{n-1}$, where $A_{n-1} = \cup\{a_i^k : i \in I_k, k \leq n-1\}$ and C is a component at the lowest $depth = n$, such that $\phi_{\Sigma_{n-1}}(X) \supseteq C^*$ for all $C^* > C$ in the poset $(A/\sim, \leq)$ with $depth < n$. We need to prove the proposition for the case when $depth = n$.
We can have two cases when $depth = n$:
1. There is a strong implication for a component C at $depth = n$. In this case, according to the hypothesis, we do not include any element from C into A^*. Thus, $\Sigma_n = \Sigma_{n-1}$. There is an implication $\breve{X}\breve{A} \to a_n$ such that $\breve{X} \subset X$,$\breve{A} \subset A$, $a_n \in C$, and for $\tilde{a} \in \breve{A}$, we have $\tilde{a} \in \breve{C}$ for $\breve{C} > C$. Then because $\breve{C} \subseteq \phi_{\Sigma_n}(X)$, we have $\breve{A} \subseteq \phi_{\Sigma_n}(X)$. Therefore, $a_n \in \phi_{\Sigma_n}(X)$. Since C is a component, there is a path from a_n to any other element a_n^* in C through implications in Σ_n in the following form:
$X_{1n}A_{1n}a_n \to a_n^1$, $X_{2n}A_{2n}a_n^1 \to a_n^2$, ... , $X_{kn}A_{kn}a_n^k \to a_n^*$. Since we have CQ system, elements in A_{sn} are from components $C' > C$, thus, they are in $\phi_{\Sigma_n}(X)$. Because X_{n1} , X_{n2}, ... , $X_{kn} \subset X$, $\phi_{\Sigma_n}(X) \supset a_n \cup a_n^1 \cup ... \cup a_n^*$. Hence $\phi_{\Sigma_n}(X) \supseteq C$.

2. There is *no* strong implication for a component C at $depth = n$: to prove this case we first need to prove the following lemma.

Lemma If there is no strong implication for a component $C \in A/\sim$, then in every optimum basis for the implication $X \to A^{**}$ one has $A^{**} \cap C \neq \emptyset$.
Proof of Lemma Since there is no strong implication for C, if $B \to D$ is in Σ, and $D \cap C \neq \emptyset$, then $B \cap C \neq$ as well.
Since $c \in \phi_\Sigma(X)$, there exists some sequence of implications $X \to A^{**}$, $X_1A_1 \to A_2 \ldots, X_kA_k \to D$, where $c \in D$. Here $X_1, \ldots, X_k \subseteq X$ and $A_1 \subseteq A^{**}, A_2 \subseteq A^{**} \cup A_1, \ldots, A_k \subseteq A^{**} \cup \cdots \cup A_{k-1}$. (We assume that component C cannot be obtained without using implication $X \to A^{**}$.) Due to assumption, $A_{k-1} \cap C \neq \emptyset$. Then $A_t \cap C \neq \emptyset$, for some $t < k-1$, hence, $A_{t-1} \cap C \neq \emptyset$, again by assumption of no strong implication. Proceeding by this argument, we obtain that $A^{**} \cap C \neq \emptyset$, which is needed. *End of Lemma's Proof*
Thus, we will need to include one element a_n^0 from component C: replace $X \to A_{n-1} \cup C$ by $X \to A_{n-1} \cup \{a_n^0\}$ choosing any element from $a_n^0 \in C$. This will define new Σ_n. Then, by argument similar to case 1, we will be able to obtain every element from C by using inference from Σ_n.
It is estimated that the time spent on this procedure to add elements to A^* is $\mathcal{O}(|S|^2)$, and the time needed to find extreme points of the set S is $\mathcal{O} = (s(\Sigma))$ [13].

4 Optimization of convex geometries in general

The question of optimization of convex geometries in polynomial time remains open. In the second part of this Thesis, we worked toward this question. Several algorithms were generated to obtain different properties that define convex geometries. To see how the results will look like, one example is considered for all algorithms, where the base set is $\{a, b, c, d, e, x, y\}$ with 4 different orderings.

4.1 Alignment

First we generated a code for obtaining an alignment of a convex geometry through *compatible orderings*, represented in [6].

Compatible ordering Let $\langle S, \mathcal{F} \rangle$ be a convex geometry. A *compatible ordering* of $\mathcal{F}$ is a total ordering of the elements of S, $s_1 < s_2 < ... < s_n$, such that the set $C_i = \{s_1, s_2, ..., s_i\} \in \mathcal{F}$ for all $i, 1 \leq i \leq n$. The set of all compatible orderings of $\mathcal{F}$ is denoted by $Comp(\mathcal{F})$.

Monotone alignment Monotone alignment $\mathcal{D}_E$ on a (finite) set S is the order alignment corresponding to some total order E on S: if E is $s_1 < s_2 < ... < s_n$, then $\mathcal{D}_E = \{\{s_1, ...s_k\} : k \leq n\}$.

Theorem For every convex geometry $\langle S, \mathcal{F} \rangle$ we have $\mathcal{F} = \vee_{E \in Comp(\mathcal{F})} \mathcal{D}_E$. [6]

Hence, constructing a system by this way guarantees that the system is a convex geometry. Generating an alignment uses this constructional method and involves the following steps:
1. We need to order elements of a given system on each line. Let m be the number of elements in a given base set S, and n the number of orderings.
2. Define n sets that represent orderings: $Q_1, Q_2, ..., Q_n$ such that $Q_1 = \{\{x_1\}, \{x_1, x_2\}, ..., \{x_1, x_2, ..., x_m\}\}$, ... , $Q_n = \{\{y_1\}, \{y_1, y_2\}, ..., \{y_1, y_2, ..., y_m\}\}$, where $\{x_1, x_2, ..., x_m\}$, ..., $\{y_1, y_2, ..., y_m\}$ are all equivalent sets and equivalent to the base set.
3. Let an initial alignment $\mathcal{F}_0 = Q_1$. Then intersect each element of $\mathcal{F}_0$ with each element of Q_2, and add the result to the initial alignment. Hence new alignment is $\mathcal{F}_1 = \mathcal{F}_0 \cup \{Q_2 \cap \mathcal{F}_0\}$. Repeat this procedure till the last ordering, Q_n. Finally, denote the resulting set as $\mathcal{F}$. This is the alignment.
In the example below, there are four orderings of S and they are given in the code. Using the algorithm, the following code was produced using the program Python:

```
n = 4 #number of orders
m = 6 #number of elements in a base set
org_set = ["a", "b", "c", "d", "x", "y"]; #given base set
order = [
        ["x", "a", "d", "c", "b", "y"],
        ["y", "b", "a", "d", "c", "x"],
        ["x", "c", "d", "b", "a", "y"],
        ["y", "c", "b", "d", "a", "x"],
      ]

Q = []
```

```
for i in range(n):
    Q.append([])
    s = ""
    for j in range(m):
        s += order[i][j]
        Q[i].append(s)

F = Q[0]

for i in range(1, n):
    P = []
    for j in range(len(F)):
        for k in range(len(Q[i])):
            s = ""
            for l in range(len(Q[i][k])):
                if Q[i][k][l] in F[j]:
                    s += Q[i][k][l]
            if len(s) > 0:
                P.append(s)
    for j in range(len(F)):
        F[j] = ''.join(sorted(F[j]))

    for j in range(len(P)):
        if "".join(sorted(P[j])) not in F:
            F.append("".join(sorted(P[j])))

print ("Alignment of the given set: " + str(F), len(F))
```

This code produces the following alignment for the given set and its length without counting an empty set:

```
Alignment of the given set: ['x', 'ax', 'adx', 'acdx', 'abcdx',
'abcdxy', 'a', 'ad', 'acd', 'b', 'ab', 'abd', 'abcd', 'y', 'by',
'aby', 'abdy', 'abcdy', 'dx', 'cx', 'cdx', 'bcdx', 'd', 'c',
'cd', 'bd', 'bcd', 'bc', 'cy', 'bcy', 'bcdy', 'bdy'] 32
```

4.2 Bases

We worked mostly with convex geometries that are given by their canonical basis or D-basis. The codes for obtaining these bases are obtained from [8] using the alignment code in the section 4.1 as the input.

The *canonical basis* is obtained in the following way:

1. Use the following definition to identify $quasi-closed\ sets$.

Let $\mathcal{F}$ be an alignment for a base set S. A set $Q \in S$ is called $quasi-closed$ if $Q \notin \mathcal{F}$ and $\mathcal{F} \cup Q$ is again an alignment for S. In other words, for every $F \in \mathcal{F}$ either $Q \subset F$ or $Q \cap F \in \mathcal{F}$. If Q is quasi-closed, then $\phi(Q)$ is called *essential*.

2. If $F \in \mathcal{F}$ is essential, then let $\mathcal{Q} = \left\{Q_i : quasi-closed\ and\ \phi(Q_i) = F, i \in [1, n]\right\}$.

3. We have $\phi(Q_1) = \phi(Q_2) = ... = \phi(Q_n)$. Pick minimal one with respect to $\subseteq$ for every F that is essential. These sets are called *critical*, C_i.

4. Canonical basis is $\Sigma = \{C_i \to \phi(C_i) \setminus C_i\}$.

```
def critical(X, closed, cls):
    if isIn(X, closed):
        return False
    Y = cls[X] | X
    subsets = powerset(X).difference(rset([X]))
    for Z in subsets:
        if cls[Z] | Z == Y and quasiClosed(Z, closed):
            return False
    return quasiClosed(X, closed)

def quasiClosed(X, closed):
    cnt = 1
    for Y in closed:
        if X < Y:
            continue
        if not isIn(Y.intersection(X), closed):
            cnt = 0
            break

    if cnt == 1:
        return True
    return False

def findCBasis(S, PS, closed):

    implies = createmap(S, PS, closed)
    cbasis = []
    #place is where potential quasi-closed sets are taken to be checked
    for i in implies:
        if critical(i[0], closed, dict(implies)):
            cbasis.append(i)

    return cbasis
```

The other type of basis that we work with is a D-basis. The definition of D-basis is the following:
$A \to x$ is in $D-basis$, if for any $a \in A$, any $B \subseteq \phi(a)$, implication $(A \setminus a) \cup B \to x$ does not hold.

```
def binaryImplications(A, binaries):
    implied = set(A)
    for a in A:
        for B in binaries:
            if a in B[0]:
                implied = implied | B[1]
    return implied

def findDBasis(cdubasis):
        dbasis = set(cdubasis)
```

```
    binary = set()
    nb = set()
    for i in cdubasis:
        if len(i[0])==1:
            binary.add(i)
        else:
            nb.add(i)
    for X1 in nb:
        for X2 in nb:
            if X1[1]==X2[1] and X1 != X2 and X1 in dbasis and X2 in dbasis:
                if X1[0] < binaryImplications(X2[0], binary):
                    dbasis.remove(X2)
    return list(dbasis)
```

The above code gives the following result for our example set:

```
Canonical Basis: [bx->cd, dy->b, xy->dbca, ca->d, ay->b]
D Basis: [xy->b, bx->d, ay->b, xy->a, xy->c, xy->d,
dy->b, ca->d, bx->c]
```

4.3 Alignment from a given basis

After optimizing a given basis, we always need to check that we did not miss any closed set in the system. The general procedure to obtain an alignment from a basis is the following:
1. Given a base set X, for all $Y \subseteq 2^X$, find all $A \to B \in \Sigma$ such that $A \subseteq Y$.
2. A closure of Y is obtained by constructing a cycle, where the first element is $Y_1 = Y \cup \{B_1 : A_1 \subseteq Y, A_1 \to B_1 \in \Sigma\}$. Then $Y_2 = Y_1 \cup \{B_2 : A_2 \subseteq Y_1, A_2 \to B_2 \in \Sigma\}$. Continuing in the same way until we reach the n-th step when there is only one implication $A_n \to B_n$ left such that either $A_n \not\subseteq Y_n$ or $B_n \subseteq Y_n$.
The following code was generated using the above notion to obtain an alignment from any given basis:

```
def getAlignment(basis, base_set):
    closed_set = []
    power_set = powerset(base_set)
    for Y in power_set:
        y = set(Y)
        prev_y = y
        while 1 > 0:
            for i in basis:
                if i[0] <= y:
                    print (i[0],y)
                    for x in i[1]:
                        y.add(x)
            if len(y) == len(prev_y):
                break
            prev_y = y
        if len(y) > 0:
            closed_set.append(rset(y))
    res = []
```

```
    for x in closed_set:
        found = False
        for y in res:
            if x == y:
                found = True
                break
        if found == False:
            res.append(x)
    closed_set = res

    return tuple(closed_set)
```

After obtaining the alignment, we need to check that the optimized basis gives the same closure system as the original one. To do so, we generated the following code that returns the value $TRUE$ if the original alignment is equivalent to the new one after optimization, otherwise it returns $FALSE$.

```
def Equal(cl1, cl2):
    cl_t_1 = []
    cl_t_2 = []

    for x in cl1:
        str_x = ""
        for y in x:
            str_x += y
        cl_t_1.append(sorted(str_x))

    for x in cl2:
        str_x = ""
        for y in x:
            str_x += y
        cl_t_2.append(sorted(str_x))

    cl_t_1 = sorted(cl_t_1)
    cl_t_2 = sorted(cl_t_2)

    if len(cl_t_1) != len(cl_t_2):
        return False

    for i in range(len(cl_t_1)):
        if cl_t_1[i] != cl_t_2[i]:
            return False

    return True
```

Taking our example set as an illustration with the original canonical basis gives the desired result:

```
Closed sets: (('a', 'b', 'c', 'd', 'x', 'y'), ('b', 'c'),
('d', 'x'), ('b', 'c', 'd', 'x'), ('y',), ('a', 'b', 'y'),
('c', 'x'), ('b', 'y'), ('x',), ('a', 'b', 'd', 'y'),
```

```
('a', 'c', 'd'), ('a', 'd', 'x'), ('d',), ('b', 'c', 'd'),
('a', 'b', 'c', 'd'), ('b',), ('a', 'b', 'c', 'd', 'x'),
('c', 'd'), ('b', 'c', 'y'), ('b', 'd', 'y'), ('c',),
('c', 'y'), ('a', 'b', 'c', 'd', 'y'), ('a',), ('a', 'd'),
('a', 'c', 'd', 'x'), ('a', 'b', 'd'),('c', 'd', 'x'),
('a', 'x'), ('a', 'b'), ('b', 'c', 'd', 'y'), ('b', 'd'))
32
True
```

4.4 Hypotheses

During the work on optimization of non-CQ convex geometries, we had several hypotheses. Let $\langle S, \phi \rangle$ be a non-CQ convex geometry with an implicational basis Σ. Suppose all implications in Σ are optimized except a non-binary implication $X \to A$ such that $\phi(X) = S$. Our goal is to optimize $X \to A$ by $X \to A^*$, where $A^* \subset A$, as in the previous chapter. Suppose as in the previous chapter, we constructed a poset of equivalence classes of S that we called components with a binary relation $\leq$. However, to check if each hypothesis was true, we tried to construct counter-examples and obtained them. Here are the hypotheses and counter-examples:

1. To optimize $X \to A$ by $X \to A^*$, one only needs to include into A^* one element from each maximal component from the defined poset; hence no element from other components are added.
   $Counter-example:$
   Suppose we are given a convex geometry with the following basis: $X = \{x_1, x_2, x_3\}$, $A = \{a, b, c, d, e, f\}$
   $X \to abcdef$
   $x_1ac \to b$
   $x_2x_3bd \to ea$
   $x_3af \to bc$
   $x_1x_2cd \to f$
   In this example, $\{d\}$ is the maximal component. But if we optimize the first implication by $X \to d$, then since there is no implication having only d with elements from X in a premise, we will not obtain elements from other components that are not maximal. Hence the hypothesis is not true.

2. To optimize $X \to A$ by $X \to A^*$, one needs to search for subsets of A in premises of implications in Σ that have at least one element from X. Then one must find a subset B of A such that if $A^* = B$, then one will obtain all elements from A using $X \to A^*$ and other implications in Σ.

   $Counter-example$:
   Suppose we are given a convex geometry with the following basis: $X = \{x, y\}$, $A = \{a, b, c, d, e\}$
   $X \to abcde$
   $xde \to a$
   $xab \to cd$
   $ye \to c$
   $ac \to d$

$abe \to cd$
$xce \to ad$
$b \to a$
In this example, the subsets of A that to be checked are $\{d, e\}$, $\{a, b\}$, $\{e\}$, $\{c, e\}$. However, if one replaces A in $X \to A$ by any of these subsets, one will not get the whole set A. For example, suppose $X \to A$ is optimized by $X \to \{d, e\}$. Then one obtains first element d, e. Then by applying the implication $xde \to a$, one obtains a. Then one cannot use any other implication. Hence one lacks elements b, c. Hence the hypothesis is not true.

5 Conclusion

In this project a type of closure systems called convex geometries were considered. In particular, a CQ subclass of convex geometries was considered in depth. A proof of the optimization of CQ convex geometries was given by referring to the general optimization algorithm of a CQ subclass of closure systems. Moreover, optimization of convex geometries without additional property, in particular, non-CQ, was studied. More precisely, to have more intuition on an optimization process we first needed to generate a number of examples that would allow us to see some pattern in an algorithm. For that purposes, several programs were created and one was used from a work which was done before [8]. After having all tools ready, we started to make hypotheses towards an optimization procedure, and obtained several. However, by searching and obtaining counter examples we could see that those hypotheses were not correct in general; they worked for particular cases only.

While working on this project, more precisely, on the problem of optimizing non-CQ convex geometries, several interesting questions appeared that are worth investigating. One of them is:

- How far the optimization algorithm for CQ case can be extended to non-CQ case systems, i.e. algorithm of choosing one element from each component?
- Can we build an optimum basis given $cdim(\mathcal{F})$, where $cdim(\mathcal{F})$ is the minimum number of compatible orderings needed to realize the alignment $\mathcal{F}$?

Acknowledgements

I would like to thank all people who supported me in doing my Capstone Thesis. I express sincere gratitude to Professor Kira Adaricheva for being my supervisor, extending my knowledge in the field of algebra and geometry, and supporting me in every step of this project during the year. I also thank Professor Rustem Takhanov for being my second reader, and helping me to improve the thesis. I would like to thank my collaborator Madina Bolat, with whom I discussed my thesis during the whole year, and my friend Abzal Serekov who helped me to work in Python.

References

[1] K. Adaricheva, *Optimum basis of finite convex geometry*, to appear in Discrete Applied Mathematics, available at http://arxiv.org/abs/1205.3236

[2] K. Adaricheva and J.B. Nation, *On implicational bases of closure systems with unique critical sets*, Disc. Appl. Math. **162** (2014), 51–69.

[3] G. Ausiello, A. D'Atri and D. Saccá, *Minimal representation of directed hypergraphs*, Journal on computing **15** (1986), 418–431.

[4] E. Boros, O. Čepek, A. Kogan and P. Kučera, *A subclass of Horn CNFs optimally compressible in poynomial time*, Ann. Math. Artif. Intell. **57**(3-4) (2009), 249-291.

[5] E. Boros, O. Čepek, A. Kogan and P. Kucěra, *Exclusive and essential sets of implicates of Boolean functions*, Disc. Appl. Math. **158** (2010), 81–96.

[6] P. Edelman, R. Jamison, *The theory of convex geometries*, Geom. Dedicata **19** (1985), 247-274.

[7] K. Kashiwabara, M. Nakamura, *The prime stems of rooted circuits of closure spaces and minimum implicational bases*, The Electronic J. of Comb. **20** (2013), paper 22, 13 pages.

[8] R. Rand, *Ordered directness and efficiency computing closures*, Honors project, Yeshiva University, 2011.

[9] M. Wild, *A theory of finite closure spaces based on implications*, Adv. Math. **108** (1994), 118–139.

[10] M. Wild, *The joy of implications, aka pure Horn formulas: mainly a survey*, to appear in Theoretical Computer Science, available in arxiv.

[11] D. Maier, *Minimum covers in the relational database model*, J. of Assoc. Comp. Machinery **27** (1980), 664–674.

[12] J.L. Guigues, V. Duquenne, *Familles minimales d'implications informatives résultant d'une tables de données binares*, Math. Sci. Hum. **95** (1986), 5–18.

[13] G. Dowling, J. Gallier, *Linear-time algorithms for testing the satisfiability of propositional Horn formulae*, L.Logic Programming **3** (1984), 267–284.