1
Introduction to Conflicts and Thesis Structure

This thesis is about the *conflict* problems in computational geometry. *Conflicts* are special type of constraints imposed on solution such that if an object is part of solution, it may restrict other objects to be part of the solution if they are in *conflict* with this object. These problems have been extensively studied in computer science as we will see in upcoming sections. Our aim here is to give a systematic framework to study such type of problems in computational geometry.

Computational Geometry Computational Geometry is the systematic study of algorithms and data structures for geometric problems. The subject has wide range of applications in areas such as robotics, computer graphics, CAD/CAM, and geographic information systems etc [32],[48][66],[74],[76].

For example, consider the problem where we are given two curves and want to measure similarity between them. Fréchet distance is one of tools that measures similarity between two curves by considering an ordering of the points along the two curves. An intuitive example of Fréchet distance is a dog and its handler walking on their respective curves. Both can control their speed but can only go forward. The Fréchet distance of these two curves is the minimum length of any leash necessary for the handler and the dog to move from the starting points of the two curves to their respective endpoints [5].

Eiter and Mannila [29] introduced discrete Fréchet distance. Intuitively, the discrete Fréchet distance replaces the dog and its owner by a pair of frogs that can only reside on any of the n and m specific pebbles on the curves A and B respectively. These frogs hop from one pebble to the next without backtracking. Formally let $A = \{a_1, a_2, \dots a_n\}$ and $B = \{b_1, b_2, \dots b_m\}$ be a sequence of points. For any $r \in \mathbb{R}$ we define the graph G_r with vertices $A \times B$ and there exists an edge between (a_i, b_j) and (a_{i+1}, b_j) if $d(a_{i+1}, b_j) \leq r$ and there exists an edge between (a_i, b_j) and (a_i, b_{j+1}) if $d(a_i, b_{j+1}) \leq r$, where $d(,)$ represents distance between two points. Discrete Fréchet distance between A and B is the infimum value of r such that in G_r there is a path between (a_1, b_1) and (a_n, b_m). This problem is polynomial time solvable by using dynamic programming.

Another example in a geometric setting is Geometric Set Cover. It is defined over a set system $\Sigma = (X, \mathscr{R})$ where X is a set of points representing universe. The set $\mathscr{R}$ is family of subsets X called ranges, defined by the intersection of X with geometric objects. Our aim is to pick a minimum number of sets from $\mathscr{R}$ so as to cover the point set X. This problem is special case of the Set Cover problem which is one of the Karp's 21 NP-complete problems [53]. In this problem we are given a set of elements called universe U of size n, a family $\mathscr{F}$ of size m of subsets of U and a positive integer d. Our aim is to find whether there exists a set $\mathscr{F}' \subseteq \mathscr{F}$ of size at most d such that union of sets in $\mathscr{F}'$ is U. The dual of set cover is Hitting Set where given U, $\mathscr{F}$, and d, we need to find if there exits a set $U' \subset U$ of size d such that all sets $F' \in \mathscr{F}$ contain at least one element from U'. This is one of the most well-studied problems in the area of algorithms and complexity as many natural problems can be modelled using Set Cover or its dual. Thus, researchers have considered various variants of Set Cover with more structural properties or other constraints that arise from real life problems so that these can be exploited to give better results. One of these variants is Geometric Set Cover.

Although Set Cover generalises many practically important problems, but major difficulty is that this problem in its general version is hard in both approximation and parameterized settings. The SET COVER admits a lower bound of $(1-o(1))\ln n$ [30] (under some standard complexity theoretic assumptions) in approximation and is also *W[2]-Hard* [25],[26].

Given the hardness in both approximation and parameterized paradigms, researchers started looking for various restricted versions of SET COVER. Such restriction can be inclusion of structural properties as in the case of GEOMETRIC SET COVER. For example, if the universe is represented by a set of points and each set in $\mathscr{F}$ is represented by unit squares, the resulting problem is called DISCRETE UNIT SQUARE COVER (DUSC). In contrast to various hardness results in case of SET COVER, in GEOMETRIC SET COVER, if the set system $(X, \mathscr{R})$ has bounded VC-dimension, say c, then it admits approximation factor of $\log(c)$ [12]. In addition to this, for DUSC, we have PTAS [68]. The UNIT SQUARE COVER is *W[1]-Hard* parameterized by size of solution [64].

In this thesis, we will look at Fréchet distance and special case of GEOMETRIC SET COVER in *conflict* framework in realm of parameterized complexity and approximation algorithm. Next we show the need of considering *conflicts* in geometry and will see few of the previous works done on *conflicts*.

1.1 MOTIVATION FOR CONFLICT

First we consider special case of GEOMETRIC SET COVER called WIRELESS ANTENNA COVER problem. In this problem, we need to place a minimum number of wireless antennas at some predefined locations in order to service a group of clients. For simplicity, we model the locations of wireless clients as points in a plane. Let this point set be P, having n points in $\mathbb{R}^2$. The possible locations of the antennas are also represented by points in plane. Let this point set, say S, has m points in $\mathbb{R}^2$. Assume, that the coverage area of each antenna is a unit radius disk centered at position of the antenna. An antenna s is said to cover a client c if the client falls in coverage region of the antenna. We need to select a subset of minimum number of antennas from S such that the entire set of wireless clients is covered. Here, our universe is the set of clients and subsets are the clients present in each region. We model this problem as DUDC (DISCRETE UNIT DISK COVER). The DUDC is special case of GEOMETRIC SET COVER where the covering objects are unit disks. This problem is known to be NP-hard [51] and admits PTAS [68]. But here we drop many real life constraints. Now we look into how to alternatively model this problem such that it also includes some real life restrictions into the definition.

Consider the scenario of mobile towers allocation which can be considered as a special case of WIRELESS ANTENNA COVER. In this problem, we cover a set of cell phone users by the minimum number of mobile towers. In the view of the adverse effect of radio waves, it might not be wise to place two towers in close proximity. That is, the placement of a tower at one point restrains us from placing towers in all nearby points. For example, consider Figure 1.1 for the intersection area of two mobile towers. As the mobile towers have a large intersection area, we may want to put a restriction that either of the mobile tower can be chosen but not both. To model this, we need to incorporate these constraints into our previous model. We call these restrictions as *conflicts*. Thus two mobile towers are in conflict when at most one of them can be chosen.

To further investigate the nature of such limitations in various problems in computational geometry, let us consider our next example. In GEOMETRIC PACKING, we attempt to pack a set of objects into a container without any overlapping. Let us formally define the GEOMETRIC PACKING problem.

GEOMETRIC PACKING
Input: A set of n objects $\mathscr{O} = \{O_1, \ldots, O_n\}$ in $\mathbb{R}^d$ a closed container with fixed volume.
Question: Does there exist a configuration of objects in $\mathscr{O}$ such that objects can fit into the container without any overlaps?

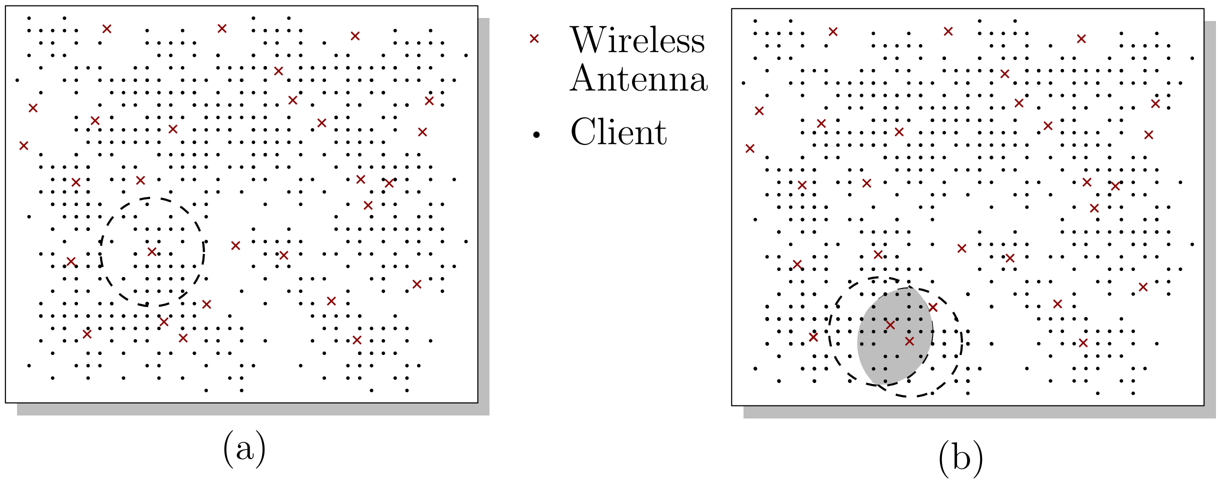


Figure 1.1 : WIRELESS ANTENNA COVER with conflicts

This problem has a wide range of applications in various disciplines, such as VLSI chip design, loading vehicle with items, creating file backups in media etc. Consider its application in VLSI chip packing. In VLSI, floor planning deals with designing the geometric layout of integrated circuits. In the process of designing the arrangement of modules on the layout surface, there are various constraints. These include minimizing the area of placement of given set of geometric modules/ blocks/ shapes of arbitrary sizes without any overlap. For simplicity, we assume that each module is of rectangular shape. Assume the shape of chip in which we want to arrange the modules to be rectangular as well.

We now define the basic underlying mathematical problem. Given a closed rectangular region A and a set of n closed rectangles $R = \{r_1, r_2, \ldots, r_n\}$, not necessarily of same size, find whether all the rectangles can be packed inside A without any overlap. We call this problem RECTANGLE PACKING. The problem of packing rectangles in a larger rectangle is NP-hard and there exists a $(2+\varepsilon)$ factor algorithm to solve this problem [50].

In real life, there are many other practical constraints apart from minimizing the area in VLSI chip design. In order to improve the performance by reducing signal delays, certain modules should be kept together. There can also be restrictions on placement of certain blocks in certain places of the circuit. Another major issue of practical concern is chip overheating. Efforts have been made to address the overheating problem by incorporating thermal constraints where the objective is to distribute heat more evenly over the chips and reduce hot spots. Again we need to add this constraint to the RECTANGLE PACKING problem. We model this constraint using *conflicts*. For example, in Figure 1.2.(a), let the rectangles r_1, r_2 and r_3 be the modules that can not be placed alongside each other. Thus, these need to be placed separately. We can term this restriction as a *conflict*. Hence we say that rectangles r_1, r_2 and r_3 are in conflict with each other due to heat constraints. In Figure 1.2.(b), we can see that there exists another placement of rectangles such that the *conflict* between r_1, r_2 and r_3 is resolved. Notice that in general scenario, it might be a case that the conflict can not be resolved.

1.2 MODELLING THE RESTRICTIONS/ CONFLICTS

Our main contribution is to give a framework for modelling the real life restrictions in underlying problems which we termed as conflicts. Let us give intuition as to how we are trying to model these restrictions. The formal treatment of our framework is given in later in the chapter.

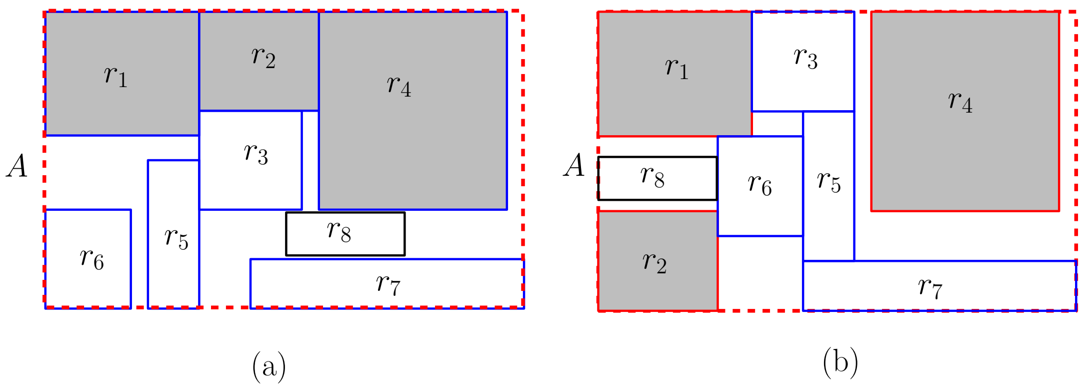


Figure 1.2 : Rectangle Cover Problem with conflicts

Let us consider WIRELESS ANTENNA COVER problem. We represent the point set S corresponding to wireless antenna by vertex set V of graph $CG(V,E)$, called *conflict graph*. The two wireless antennas $s_1, s_2 \in S$ are said to be in conflict if there is a restriction that only one of them can be chosen. For any two wireless antennas $s_1, s_2 \in S$, we add an edge $(s_1, s_2) \in E$ in G if and only if s_1, s_2 are in conflict. Now our goal is to find a minimum number of wireless antennas from S such that the corresponding vertices in CG form an independent set.

Similarly, in VLSI chip design problem, let us represent each rectangular module as a vertex in graph $CG(V,E)$ (conflict graph). If for any two modules $r_1, r_2 \in R$ there is a restriction that both modules can not be placed with alongside each other, then we say that the the two modules are in *conflict*. We represent every such conflict by an edge in CG. Now our aim is to maximize the number of modules that we can fit into A such that each module and its immediate neighbours in the configuration form an independent set.

We consider conflicts as real life restrictions on some underlying problem. The solution to the new problem with conflicts is a feasible solution to underlying problem that also satisfies the conflicts(constraints).

We want to point out that if we model the restrictions/ conflicts by a graph which we call conflict graph, then our aim might not necessarily be to find an independent set in conflict graph. It can be anything like finding a clique, an independent set or a vertex cover etc in the conflict graph. In this thesis, we generally aim at finding an independent set in conflict graph. We further discuss the model for conflicts used in this thesis more formally later in this chapter. Let us first see some of the work in literature related to conflicts.

1.3 MORE PROBLEMS IN LITERATURE ON CONFLICTS

Here we mention few problems related to conflicts studied in literature. Along with that, we also give some intuition and motivation as to how and why we are trying to abstract these restrictions from problem specific constraints to a generalized conflict model.

1.3.1 Conflicts in Software Engineering

As per our knowledge, the first problem with conflicts was studied by Gabow et al. in [37]. The problem was automatic path generation for programs which satisfies certain restrictions.

Let us divide a program into its basic blocks. A block is a straight-line code sequence with no branches in it except entry branches and exit branches. Consider the problem of constructing a path satisfying some restrictions and passing through basic blocks of program. Such a problem can be used in testing.

We may fragment a program into small blocks (even single statements) and can construct test cases for each block. A test case is a set of actions executed to verify a particular feature or functionality of given code. The goal is to find a path through the program such that each block of program is covered by at least one test case. But there might be a case where program semantics do not allow generation of paths that are unexecutable. For example, in Figure 1.3, consider $if-else$ construct in C language. Here if $Block\ 1$ in if clause is executed then $Clause\ 2$ in $else$ won't be executed and vice versa. This notion of pairs having conflicts was studied by Kruass et al. in [56]. Hence, we can analyse branching conditions in programs and constrain the test path such that every test path contains at most one branch of each pair with conflict. Work done in [15],[71] and [55] provides insights on automatic generation of non trivial impossible pairs.

```
if (x == y && x ≤ 100)
{
    printf("Value of y is %d. \n", y );      ↑
    x++;                                     |   Block 1
    y--;                                     ↓
}
else
{
    printf("Value of x is %d. \n", x );      ↑
    x = x++ + (x − y)%y;                     |   Block 2
    x = x%y;                                 ↓
}
```

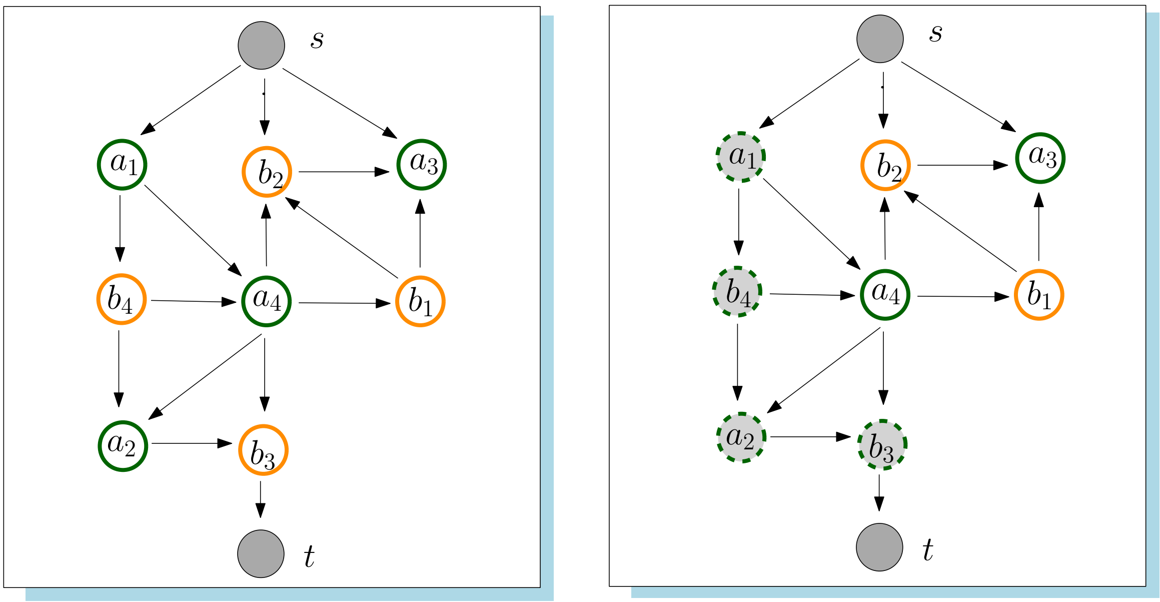
Figure 1.3 : CONFLICTING PAIRS IN IMPOSSIBLE PAIR CONTAINED PATH (IPP)

Gabow et al. also provided graph theoretic insight for this problem in software engineering. Here each program is represented by a digraph where each node represents a basic block in the program and each edge represents a possible transfer of control. Such a graph is termed as a program flow graph. The problem can be stated formally as follows,

IMPOSSIBLE PAIR CONTAINED PATH (IPP)
Given: A program flow graph $G(V,E)$, single source node s and single sink node t. The set V also contains n pairs of nodes $(a_i,b_i), 1 \leq i \leq n$. Thus $V = \{s,t,a_1,b_1,\ldots a_n,b_n\}$.
Aim: The objective is to find a path (with no vertex repetition) in G from s to t such that the path contains exactly one node from each pair.

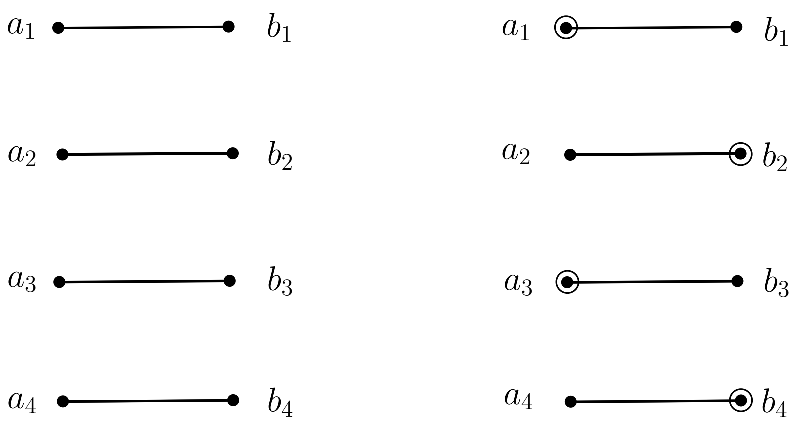
Consider the example of this problem in Figure 1.4. Here we represent each node as a vertex in the conflict graph $CG(V',E')$. We say that the two nodes are in conflict if they belong to same pair in V and hence restricts us from picking both the nodes in conflict. That is, at most one of them can be picked in our solution. A conflict between two nodes is represented as an edge in CG. Our objective is to pick a n size independent set from CG such that the corresponding vertices form a path from s to t in G. Notice that CG in this case is simply a matching. See Figure 1.5 for a conflict graph corresponding to the problem in

Figure 1.4.



Program flow graph $Path = (s, a_1, b_4, a_2, b_3, t)$

Figure 1.4 : Impossible Pair Contained Path (IPP)



Conflict Graph

$Solution = \{a_1, b_2, a_3, b_4\}$
Solution to IPP is independent set in Conflict Graph

Figure 1.5 : Conflict graph in Impossible Pair Contained Path (IPP)

This problem is NP-hard [53] even when the underlying flow graph is acyclic and in-degree and out-degree of each vertex is at most two.

1.3.2 Conflicts in Some Classical Problems

Itai, Rodeh and Tanimoto [78] studied the conflict version Restricted Matching Problem (RCM) of matching problem. Maximum Matching in bipartite graph is the problem of finding a set of maximum number of vertex disjoint edges from the graph. The problem Restricted Matching Problem (RCM) can be defined as follows,

RESTRICTED MATCHING PROBLEM (RCM)

Given: A bipartite graph $G(V,E)$ where V is partitioned into two disjoint sets, L and R, such that all the edges have one end-point in L and another in R. Also $E_1, E_2, \ldots E_m$ are subsets of E, and $r_1, r_2, \ldots r_m$ are positive integers.
Aim: The objective is to find whether there exists a perfect matching M in G such that it satisfies the following restriction:

$$|M \cap E_j| \leq r_j \text{ for all } j = 1,2,\ldots k$$

The authors have proved RCM to be NP-complete when all subsets are disjoint in [78] .

Consider the vertex set in $CG(V',E')$ to be the edge set in $G(V,E)$ that is, $V' = E$. In the case when all subsets are disjoint, we say two vertices $u,v \in V'$ to be in conflict and add an edge $(u,v) \in E'$, if and only if $u,v \in E_i$ for some $1 \leq i \leq m$. Thus, any feasible solution to RESTRICTED MATCHING PROBLEM (RCM) consists of disjoint subgraphs $S_1(V_1',E_1'), S_2(V_2',E_2') \ldots S_m(V_m',E_m')$ in $CG(V',E')$ such that for each $S_i(V_i',E_i')$, $1 \leq i \leq m$, $V_i' \in V'$ corresponds to E_i and $|V_1'| \leq r_i$. Also the corresponding edges in E must form a perfect matching in G.

Consider the case when subsets of E are not disjoint. Here, consider $E' = \emptyset$. Assign colour k to each vertex $v \in V'$, if and only if corresponding edge $v \in E$ is assigned to the subset E_k. The vertices of V' can be assigned many colors. Our goal is to find a vertex set C in V' such that the number of vertices of each color p in C are not more than r_p. Also, the edges in E corresponding to vertices in C should form a perfect matching in G.

KNAPSACK is another important problem which has been studied with conflicts as MULTIPLE-CHOICE KNAPSACK problem (MCKP). In KNAPSACK we are given n number of objects in object set $O = \{O_1, O_2, \ldots, O_n\}$. Each element $O_i \in O$ is associated with weight W_i and value V_i. A bag B with capacity C is also given. The objective is find a set of objects from O such that their total weight is at most W while maximizing their total value. In MULTIPLE-CHOICE KNAPSACK problem (MCKP) the set of items is partitioned into classes. The binary choice of picking an item is replaced by the selection of exactly one item out of each class of items.

MULTIPLE-CHOICE KNAPSACK(MCKP)

Given: A set X of n items. The set of items is divided into k non-empty disjoint classes $X_1, X_2, \ldots X_k$. The j^{th} item in class $X_i, 1 \leq i \leq k$ is denoted as $x_{ij}, 1 \leq i+j \leq n$. Also, $N_i = \{j | x_{ij} \in X_i\}, 1 \leq i \leq k$. Each item x_{ij} is associated with profit p_{ij} and weight w_{ij}. We are also given the capacity W of knapsack.
Aim: The objective is to select exactly one item from each class such that the profit is maximized and the total weight of the items is not greater than W. The integer programming formulation for the problem is as follows.

$$\begin{aligned} & maximize \sum_{i=1}^{k} \sum_{j \in N_i} x_{ij} p_{ij} \\ & subject\ to \\ & \qquad \sum_{i=1}^{k} \sum_{j \in N_i} w_{ij} x_{ij} \leq W \\ & \qquad \sum_{j \in N_i} x_{ij} = 1 \quad For\ all\ 1 \leq i \leq k \\ & \qquad x_{ij} \in \{0,1\} \qquad For\ all\ 1 \leq i \leq k\ and\ all\ j \in N_i \end{aligned} \tag{1.1}$$

We create a conflict graph $CG(V,E)$ for this problem as follows. We say any two items $x_{iu}, x_{iv} \in X$ to be in conflict if both the items belong to same class X_i. Let us represent each item by a vertex in $CG(V,E)$.

We add an edge between two vertices in E if both of them are in conflict. Notice that the conflict graph here is collection of cliques and the set of items in any feasible solution to MULTIPLE-CHOICE KNAPSACK forms an independent set in CG.

The MULTIPLE-CHOICE KNAPSACK problem is a well known optimization problem and was first formulated by Eilon and Christofides [28] as a cargo loading problem, and can be solved using algorithms presented by Martello and Toth [62] and Pisinger [73]. It is NP hard as KNAPSACK can be considered special case of (MCKP) and can be solved exactly with dynamic programming. This problem has been extensively studied in literature [22],[63],[54],[18],[77] and is shown to admit PTAS by Chekuri and Khanna [14]. It has further been studied in multidimensional settings. [67],[35],[41],[75],[38]

1.3.3 Conflicts in Geometry

Arkin and Hassin [2] studied same problems with conflicts in geometric settings. Consider that there are m firms and a set of n locations V. Consider that each firm f_i can choose a possible location from the set of locations in $S_i \subseteq V$. Each firm wants to communicate with every other firm and hence wants to be at close distance with each firm. The objective is to minimize the maximum distance between firms. Mathematically, let the set $V = S_1 \cup S_2 \cup \ldots \cup S_m$ for some m. A cover is subset of V such that it contains at least one representative from each $S_i, 1 \leq i \leq m$. Let us define a distance measure on elements of V such that there is finite distance between every element pair of V. Then we define diameter as maximum distance between any pair of elements in the cover. Using these definitions, we formally state above problem as follows.

MULTIPLE CHOICE COVER
Given: A set V of n elements and a collection of subsets $S_1, S_2, \ldots S_m$ of V. A real symmetric $n \times n$ matrix M describing distance between every two elements of V.
Aim: The objective is to compute cover of minimum diameter.

Consider the case when all subsets in given collection of subsets are disjoint. Here, we represent each element in V as a vertex in $CG(V', E')$ (conflict graph) and add an edge in E' between vertices corresponding to two elements if and only if both elements belong to the same subset $S_i,\ 1 \leq i \leq m$. We call such elements to be in conflict. We can see that the graph CG consists of disjoint cliques. Also, any feasible solution to MULTIPLE CHOICE COVER should span all the cliques in CG. That is, the set of elements in optimal solution contains at least one element from each disjoint clique in CG such that the diameter of set is minimum.

Let us construct the conflict graph $CG(V', E')$ for general case of this problem. Here subsets in given collection of subsets are not necessarily disjoint. In this case, we represent each element in V to be a vertex in $CG(V', E')$ and $E' = \emptyset$. Assume $C = \{1, 2, \ldots m\}$ be a color set. We assign a color $i \in C$ to a vertex $v \in V'$ if and only if $v \in S_i$. Thus a vertex may be assigned many colors. Let the set $\mathscr{S}$ contain all those sets consisting of at least one vertex $v \in V'$ of each color from C. Our aim is find a set $S \in \mathscr{S}$ with minimum possible diameter.

Arkin and Hassin [2] showed that the pair-choice cover problem of finding a vertex cover of minimum diameter is polynomial time solvable. Further, the triple-choice cover problem is NP-complete even when the sets are disjoint and are represented by points in plane, and can not be approximated with constant error guarantee, assuming $P \neq NP$. With triangle equality on distances, no approximation of error-guarantee less than 2 is possible for this problem. With triangle inequality, they give a 2-approximation algorithm for multiple choice cover problem. Further if V is a set of points in plane, then they give a $\frac{2}{\sqrt{3}}$-approximation algorithm for the MULTIPLE CHOICE COVER problem.

Now let us look at a slightly different problem with conflicts. In MULTIPLE CHOICE k PARTITION problem, studied by same authors in [2], it is assumed that each cover can be partitioned into k clusters and objective is to minimize the maximum diameter of a cluster. For example, consider MULTIPLE CHOICE 2 PARTITION. Here we are given a set V of n elements, a collection of subsets $S_1, S_2, \ldots S_m$ of V (can be non disjoint subsets) and a real symmetric $n \times n$ matrix M describing distance between each pair of elements in V. Let C be the cover. Also, assume C_1 and C_2 to be a 2 *partition* of C. Then, the diameter is defined as $\max_{i \in \{1,2\}} \max\{d(x,y) | x, y \in C_i\}$ where $d(,)$ is the distance between two elements given in matrix M. Now the objective is to find a cover OPT and a 2 *partition* of C of minimum diameter. They proved that the PAIR CHOICE 2 PARTITION problem is NP-hard even when we allow the sets $S_i, 1 \leq i \leq m$ to be non-disjoint. It also holds when we assume triangle inequality. The PAIR CHOICE 2-PARTITION problem is polynomial time solvable in plane. Also, the PAIR CHOICE 2 PARTITION problem can not be approximated with constant error guarantee and even with triangle equality, no approximation of error guarantee 2 is possible. Also, 3 PARTITION problem is NP-hard even in a $2-d$ plane [51], [39]. The k PARTITION problem in one dimension where points are on a line, can be solved in polynomial time [13]. The $2-$approximation for k PARTITION with triangle inequality is given in [39], [47].

Arkin and Hassin [2] further considered ℓ CHOICE k PARTITION problem restricting the cardinality of each $S_i, 1 \leq i \leq m$ to be bounded by ℓ. They showed that with triangle equality this problem can be approximated with a bound of 2, for any fixed ℓ and k. Also, in a plane, it can be approximated with error bound of $\frac{2}{\sqrt{3}}$. It was showed in [39], [47] that a $2-\varepsilon$ approximation for SINGLE CHOICE k PARTITION problem where k is input to the problem, is NP-hard for any $\varepsilon > 0$. It also holds when V is a set of points in 3 dimensional Euclidean plane [39].

Another problem considered in same paper is related to dispersion. Dispersion of a set is the distance between two closest elements in the set. The problem aims to maximize the dispersion of the cover. Arkin and Hassin [2] proved that the PAIR CHOICE MAXIMUM DISPERSION problem of finding a vertex cover of maximum dispersion is polynomial time solvable. The TRIPLE CHOICE MAXIMUM DISPERSION problem is NP-complete even when sets are disjoint and represented by points in plane and can not be approximated with constant error guarantee assuming $P \neq NP$. Further, the ℓ CHOICE k PARTITION MAXIMUM DISPERSION problem cannot be solved or approximated in polynomial time, even if $\ell = k = 2$ assuming $P \neq NP$.

Now we look at some p-CENTER problems in conflict setting. We define the problem as follows:

p-CENTER
Given: A complete graph $G(V,E)$. Also, the non-negative integer $w(u,v)$ is assigned to every edge (u,v) which is termed as distance between vertices u and v.
Aim: Choose a set $C \subset V$ of p vertices such that the maximum distance of any vertex to the nearest vetex in C is minimized.

We call each vertex $v \in C$ as center. This problem has been proved to be NP-hard [52]. Further, Hudec [49], introduced p-CENTER problem with more restrictions. In this version, p nodes in the center must be chosen from p prespecified disjoint pairs of nodes, with exactly one node from each pair selected. Let us take nodes in disjoint pairs of nodes as vertices of graph $CG(V,E)$ and add an edge in E between two nodes if and only if these nodes belong to same pair. We call the nodes in same pair to be in conflict and $CG(V,E)$ as conflict graph. Observe that CG is a matching. Also, any solution to the problem must form an independent set in CG.

In [46], Hochbaum and Pathria generalized these problems and studied in approximation settings. Formally, generalized version is defined as follows :

GENERALIZED p-CENTER
Given: Given a complete weighted graph, $G = (V \cup W, E)$, where V and W are disjoints sets of nodes such that $|W| = n$ and $V = V_1 \cup \ldots \cup V_k$ where $V_i \cap V_j = \emptyset,\ i \neq j,\ 1 \leq i, j \leq k$. where different $V_i,\ 1 \leq i \leq k$ forms a partition of V. For each edge $(u,v) \in E,\ u, v \in V \cup W,\ w(u,v)$ denotes the non negative integer weight

assigned to the edge.
Aim: Choose a set C of p nodes from V with at most one node from each $V_i, 1 \leq i \leq k$ such that the maximum distance of any node in $V \cup W$ to its nearest neighbour in C is minimized.

Again these additional restrictions to p-CENTER in GENERALIZED p-CENTER can be modelled as conflict in the same way as done in MULTIPLE CHOICE COVER.

Consuegra, Narasimhan and Tanigawa [17] also considered various geometric problems with such additional restrictions. Two of these problems are MAXIMUM MINGAP problem and MINIMUM MAXGAP problem. For a set of n points on a line, the MINGAP is the smallest gap between consecutive points in sorted order. Similarly, MINIMUM MAXGAP is the largest gap between consecutive points in sorted order. The problems are defined as follows,

MAXIMUM MINGAP
Given: A set $\mathscr{P}$ of n k-point sets.
Aim: The objective is to choose exactly one point from each k-element set so as to maximize the MINGAP.

MINIMUM MAXGAP
Given: A set $\mathscr{P}$ of n k-point sets.
Aim: The objective is to choose exactly one point from each k-element set so as to minimize the MAXGAP.

Here, consider the points to be vertices in $CG(V,E)$ and add an edge between two points if and only if these points belong to same point set in $\mathscr{P}$. The points in same point set in $\mathscr{P}$ are said to be in conflict with each other. Notice that CG is a collection of n k-sized cliques. Also, the solution to the problems must form a maximum sized independent set in CG.

The authors give polynomial time algorithm for MAXIMUM MINGAP problem when $k = 2$ for points given in $\mathbb{R}^d$. Also, for $k > 2$, this problem is NP-hard. The authors also give a 2-approximation algorithm for MINIMUM MAXGAP problem when k is a constant and points are given on a line. They further considered convex hull, geometric minimum spanning Euclidean tree and many other such problems with such additional restrictions (conflicts) from the perspective of approximation algorithms.

These are few among several problems studied in literature where conflict comes into picture. The conflict is introduced on some underlying problem. For example, in the problem RESTRICTED MATCHING, the conflict is introduced on matching problem. But most of these problems that have been studied in literature consist of multiple copies of an element/item/node which we call a class. These aliases in a class are in conflict with each other. Hence the objective is to choose at most one or exactly one of these aliases from each class. If we represent these elements by vertices of a graph and the conflicts by the edges, then such graph for conflict belongs to some special graph classes such as bipartite graphs or cliques graphs (graph consists of disjoint cliques).

In this thesis we focus on problems where we want to remove such restrictions and consider more *c*omplex conflict relationships. To do so, as our main contribution we need to first formally propose the framework to model the conflicts. Furthermore, this helps in abstracting conflicts from problem specific restrictions, to more general class of restrictions. Next, we propose the models for capturing conflicts regardless of the underlying problem on which conflict has been introduced.

1.4 OUR FRAMEWORK FOR CONFLICTS

To handle the conflicts, we propose following two models.

1. Graphical Model

2. Matroidal Model

1.4.1 Graphical Model

A natural way to model conflict is by using graphs. The intuition and examples to this have been provided previously. Here we use graphs to represent conflicts. Consider the graph $G(V,E)$ where V is a set of elements and E is the edge set. We call G to be a *conflict graph* if and only if each edge $(u,v) \in E$; $u,v \in V$ represents a conflict between elements u and v. Hence any feasible solution to the underlying problem should be a independent set in the conflict graph.

We illustrate this using SET COVER. Here, we are given A universe U of size n, a family $\mathscr{F}$ of size m of subsets of U and a positive integer d. We are also given a graph $CG_{\mathscr{F}}$, on the vertex set $\mathscr{F}$ and there is an edge between two sets $F_i, F_j \in \mathscr{F}$ if F_i and F_j are in conflict. We call $CG_{\mathscr{F}}$ a *conflict graph*. Let $CG_{\mathscr{F}}[\mathscr{F}']$ be the graph induced on $\mathscr{F}'$. The set cover $\mathscr{F}'$ such that $CG_{\mathscr{F}}[\mathscr{F}']$ is an independent set is called a *conflict free set cover*. Our aim in CONFLICT-FREE SET COVER is to find whether there exist a set cover $\mathscr{F}' \subseteq \mathscr{F}$ of size at most d such that $CG_{\mathscr{F}}[\mathscr{F}']$ is an independent set. An example of CONFLICT-FREE SET COVER is shown in the figure 1.6.

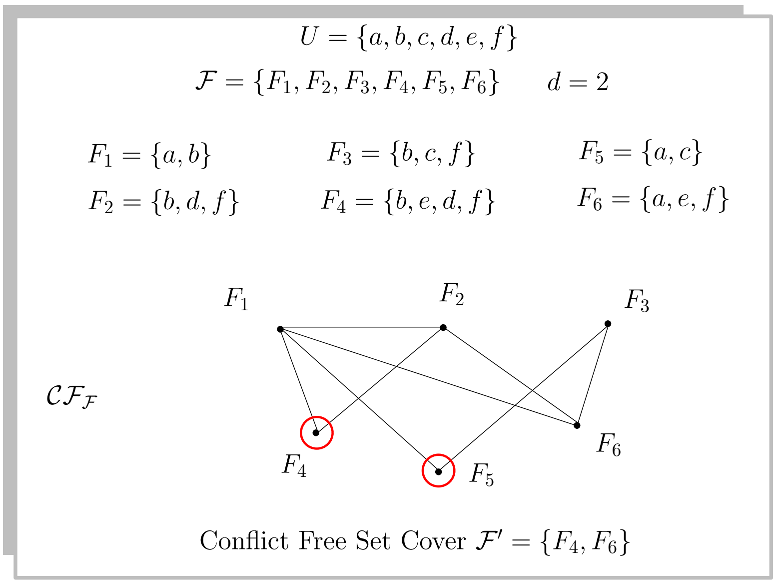


Figure 1.6 : An example of CONFLICT-FREE SET COVER

We define the parametrized version of SET COVER as follows,

> GRAPHICAL CONFLICT FREE SET COVER (GRAPHICAL CF-SC) **Parameter:** k
> **Input:** A universe U of size n, a family $\mathscr{F}$ of size m of subsets of U, a conflict graph $CG_{\mathscr{F}}$ and a positive integer k,
> **Question:** Does there exist a set cover $\mathscr{F}' \subseteq \mathscr{F}$ of size at most k such that $CG_{\mathscr{F}}[\mathscr{F}']$ is an independent set?

Let us observe some properties of CONFLICT-FREE SET COVER.

Observation 1.4.1. *The* CONFLICT-FREE SET COVER *satisfies following properties:*

1. *It is NP-hard.*
2. *It can be trivially solved if $CG_{\mathscr{F}}$ is a complete graph.*

3. *There does not exist better than polynomial factor approximation for* CONFLICT-FREE SET COVER.

The first property can be seen as SET COVER is a special case of CONFLICT-FREE SET COVER when $CG_{\mathscr{F}}$ is an edgeless graph. The second property follows from the fact that any feasible solution to CONFLICT-FREE SET COVER consists of a single element $F \in \mathscr{F}$. In the optimization version of CONFLICT-FREE SET COVER, we want to maximize the number of elements of universe covered using conflict free subset of $\mathscr{F}$. To observe the third property, consider the following scenario. Let every subset $F \in \mathscr{F}$ contains a single element from universe U. Let $CG_{\mathscr{F}}$ be any conflict graph on $\mathscr{F}$. Then CONFLICT-FREE SET COVER is equivalent to finding a MAXIMUM INDEPENDENT SET in $CG_{\mathscr{F}}$. As there is no better than polynomial factor approximation for MAXIMUM INDEPENDENT SET, same holds for CONFLICT-FREE SET COVER.

Conflict-free set other than independent set Above we consider the conflicits as edges in conflict graph $\mathscr{CG}(V,E)$. We also assumed the conflict-free set to be an independent set. Thus it is a graph induced on $\mathscr{CG}(V,E)$ such that the graph consists of disjoint vertices. But we may not model every conflict problem using this approach. For example, consider RESTRICTED MATCHING PROBLEM (RCM) mentioned above which was introduced by Itai, Rodeh and Tanimoto [78]. Let us recall the problem,

RESTRICTED MATCHING PROBLEM (RCM)
Given: A bipartite graph $G(V,E)$ where V is partitioned into two disjoint sets, L and R such that all the edges have one end-point in L and another in R. Also $E_l, E_2, \ldots E_m$ are subsets of E and $r_1, r_2, \ldots r_m$ be positive integers.
Aim: The objective is to find whether there exists a perfect matching M in G such that it satisfies following restriction:

$$|M \cap E_j| \leq r_j \text{ for all } j = 1,2,\ldots k$$

.

If we consider the vertex set in $\mathscr{CG}(V',E')$ to be the edge set for $G(V,E)$ that is $V' = E$. Also, there is an edge $(u,v) \in E'$, $u,v \in E$ if and only if $u,v \in E_i$ for some $1 \leq i \leq m$. Thus, the conflict-free set for RESTRICTED MATCHING PROBLEM (RCM) consists of disjoint subgraphs $S_1(V_1',E_1'), S_2(V_2',E_2') \ldots S_m(V_m',E_m')$ in $\mathscr{CG}(V',E')$ such that for each $S_i(V_i',E_i')$, $1 \leq i \leq m$, $V_i' \in V'$ corresponding to E and $|V'| \leq r_i$. In RCM, we want a conflict-free set that forms a perfect matching in G.

1.4.2 Matroidal Model

Let $(U,\mathscr{F},k)$ be an instance of SET COVER. In the matroidal model of representing conflicts, we are given a matroid $M = (E,\mathscr{I})$, where the ground set $E = \mathscr{F}$, and $\mathscr{I}$ is a family of subsets of $\mathscr{F}$ satisfying all the three properties of a matroid which are as follows :

1. The emptyset $\emptyset$ is in $\mathscr{I}$.

2. If $A \in \mathscr{I}$ where $A \subset E$, then every subset $A^{'} \subset A$ also belongs to $\mathscr{I}$.

3. If there are two subsets $A,B \in \mathscr{I}$ and A has more elements than B, then there exist an element $a \in A - B$ such that $B \cup \{a\}$ also belongs to $\mathscr{I}$.

Let A be a matrix over an arbitrary field $\mathbb{F}$ and let E be the set of columns of A. For A, we define matroid $M = (E,\mathscr{I})$ as follows. A set $X \subseteq E$ is independent (that is $X \in \mathscr{I}$) if the corresponding columns are linearly independent over $\mathbb{F}$. The matroids that can be defined by such a construction are called *linear matroids*, and if a matroid can be defined by a matrix A over a field $\mathbb{F}$, then we say that the matroid is representable over $\mathbb{F}$. That is, a matroid $M = (E,\mathscr{I})$ of rank d is representable over a field $\mathbb{F}$ if there exist vectors in $\mathbb{F}^d$ corresponding to the elements such that linearly independent sets of vectors correspond to independent sets of the matroid. In this thesis we assume that $M = (E, \mathscr{J})$ is a *linear or representable matroid*, and the corresponding linear representation is given as part of the input.

Similarly, a partition matroid $M = (E, \mathscr{J})$ is defined by a ground set E being partitioned into (disjoint) sets $E_1, \ldots, E_\ell$ and by ℓ non-negative integers $k_1, \ldots, k_\ell$. A set $X \subseteq E$ is independent if and only if $|X \cap E_i| \leq k_i$ for all $i \in \{1, \ldots, \ell\}$. In CONFLICT-FREE SET COVER let $\mathscr{Q}$ denotes the family of conflict free subsets of $\mathscr{F}$. One can define a *partition matroid* on $\mathscr{F}$ such that $\mathscr{J} = \mathscr{Q}$. Thus, the question of finding a conflict free subset of $\mathscr{F}$ covering the universe U becomes a problem of finding an independent set in $\mathscr{J}$ that covers all the points in P. We see in Chapter 4 how matroids can be used to give algorithms for problems in conflict settings.

1.5 PROBLEMS CONSIDERED IN THESIS

We define the problems with conflicts using the above framework. Most of the problems we considered are NP-hard. We study these problems from either the parameterized perspective or approximation perspective. Now we define the problems that we consider in this thesis.

We first consider RAINBOW COVERING, introduced in [3, 4, 58]. To define the problem formally we first give some definitions. Let P be a set of points on the X-axis, and let $\mathscr{I} = \{I_1, \ldots, I_m\}$ be a set of intervals on the X-axis. Furthermore, let $\mathscr{C} = \{C_1, C_2, \ldots, C_\ell\}$ denote a set of pair of intervals from $\mathscr{I}$. Moreover, for any pair of integers i, j ($1 \leq i \leq j \leq \ell$), $C_i \cap C_j = \emptyset$. Thus, there is a conflict between interval I_1 and I_2 if $\{I_1, I_2\} = C_i$ for some $C_i \in C$. We term $\mathscr{C}$ a *matching family*. For a set of intervals $Q \subseteq \mathscr{I}$, Q is *conflict free* if Q contains at most one interval from each each pair, i.e. $\forall_{1 \leq i \leq \ell} |Q \cap C_i| \leq 1$. Finally, for an interval $I = [a, b]$ and a point p on X-axis, we say *I covers p* if and only if $a \leq p \leq b$. Now we are ready to define the problem formally.

> RAINBOW COVERING
> **Input:** A set of points P on the X-axis, a set of intervals $\mathscr{I} = \{I_1, \ldots, I_m\}$ on the X-axis and a matching family $\mathscr{C} = \{C_1, C_2, \ldots, C_\ell\}$.
> **Question:** Does there exist a conflict free subset Q of intervals that covers all the points in P?

That is, in any family of intervals that covers all the points, we are allowed to select at most one interval from each pair in $\mathscr{C}$. Thus, the set $\mathscr{C}$ represents *conflicts* between pairs of intervals. If $\mathscr{C}$ is an empty set, that is, if there are *no conflicts* among the intervals, then the problem (known as COVERING POINTS BY INTERVALS) is polynomial time solvable [27, pg 240]. On the other hand if $\mathscr{C}$ is *non-empty* then Arkin et al. [3] showed that RAINBOW COVERING is NP-complete. Observe that in the RAINBOW COVERING problem, the family $\mathscr{C}$ would correspond to $CG_{\mathscr{C}}$ with degree at most one. That is, edges of $CG_{\mathscr{C}}$ form a matching. And the question of finding a conflict free subset Q of intervals covering all the points in P becomes a problem of finding a set Q of intervals that covers all the points in P and $CG_{\mathscr{C}}[Q]$ is an independent set. For an example, see Figure 1.7.

Another problem we define is generalized conflict-free interval covering problem. Here $CG_{\mathscr{I}}$ is any graph and not restricted to some graph classes.

> GENERALIZED CONFLICT-FREE INTERVAL COVER
> **Input:** A set of points P on the X-axis, a set of intervals $\mathscr{I} = \{I_1, \ldots, I_m\}$ on the X-axis and conflict graph $CG_{\mathscr{I}}$.
> **Question:** Does there exist a conflict free subset Q of intervals that covers all the points in P?

Recall that a conflict free subset is an independent set in $CG_{\mathscr{I}}$. We can show that this problem is NP-complete using RAINBOW COVERING. Now we define the parametrized version of this problem.

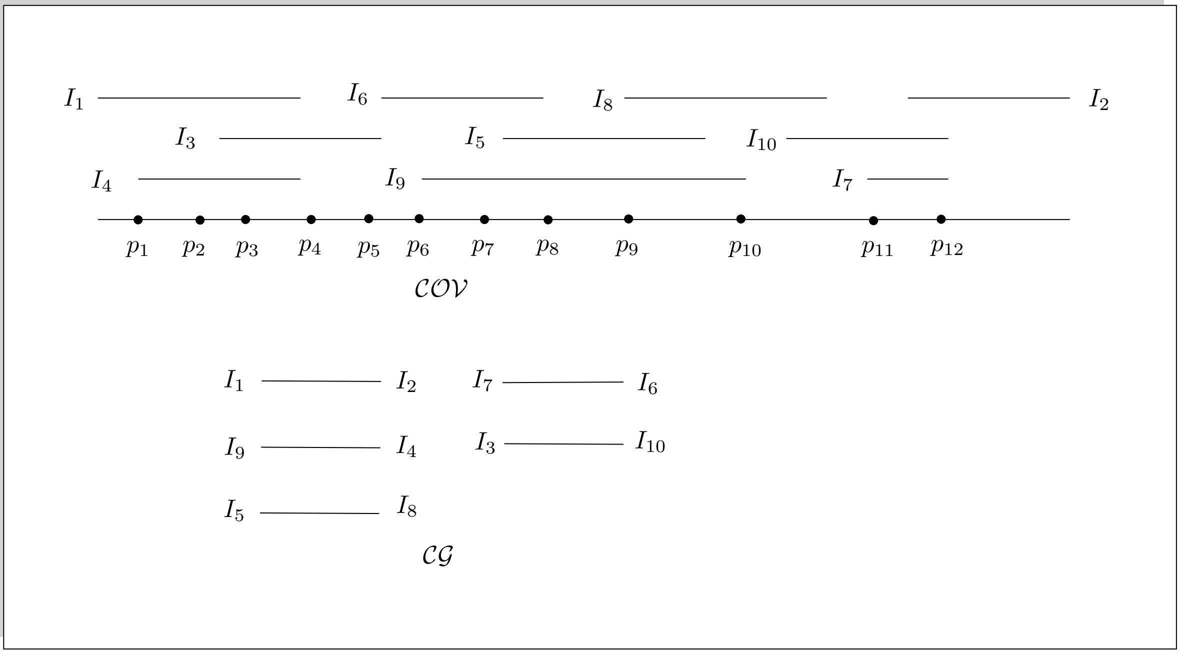


Figure 1.7 : Rainbow Covering

Parameterized Generalized Conflict-Free Interval Cover	**Parameter:** k
Input: A set of points P on the X-axis, a set of intervals $\mathscr{I} = \{I_1, \ldots, I_m\}$ on the X-axis and conflict graph $CG_{\mathscr{I}}$.	
Question: Does there exist a conflict free subset Q of intervals with size at most k such that it covers all the points in P?	

For general class of graphs to which *conflict graph* belongs, we prove later that Parameterized Generalized Conflict-Free Interval Cover is W[1]-hard. So we try to find the family of graph classes for which Parameterized Generalized Conflict-Free Interval Cover is tractable. In order to restrict the family of graphs to which a *conflict graph* belongs, we need to define the notion of *arboricity*. The arboricity of an undirected graph is the minimum number of forests into which its edges can be partitioned. A graph G is said to have *arboricity* d if the edges of G can be partitioned into at most d forests. Let $\mathscr{G}_d$ denote the family of graphs of arboricity d. This family includes the family of intersection graphs of low density objects in low dimensional Euclidean space as explained in [42, 43].

Specifically, this includes planar graphs, graphs excluding a fixed graph as a minor, graphs of bounded expansion, and graphs of bounded degeneracy. In most applications, conflict graphs themselves belong to a family of geometric graphs. Har-Peled and Quanrud [42, 43] showed that low-density geometric objects form a subclass of the class of graphs that have polynomial expansion, which in turn, is contained in the class of graphs of bounded arboricity. Thus, our restriction of the family of conflict graphs to a family of graphs of bounded arboricity covers a large class of low-density geometric objects.

Now, we consider problems with *conflicts* which we analyse from the perspective of approximation algorithms in the thesis. For this, we first talk about Fréchet distance. As we mentioned earlier Fréchet distance measures similarity between two curves by considering an ordering of the points along the two curves. We gave an intuitive example of the Fréchet distance which is to imagine that a dog and its handler are walking on their respective curves. Both can control their speed but can only go forward. The Fréchet distance of these two curves is the minimum length of any leash necessary for the handler and the dog to move from the starting points of the two curves to their respective endpoints [5]. While for the discrete Fréchet distance, we replace the dog and its owner by a pair of frogs that can only reside on any of the n and m specific pebbles on the curves A and B respectively. These frogs hop from a pebble to the next without backtracking. Formally let $A = \{a_1, a_2, \dots a_n\}$ and $B = \{b_1, b_2, \dots b_m\}$ be a sequence of points. For any $r \in \mathbb{R}$ we define the graph G_r with vertices $A \times B$ and there exists an edge between (a_i, b_j) and (a_{i+1}, b_j) if $d(a_{i+1}, b_j) \leq r$ and there exists an edge between (a_i, b_j) and (a_i, b_{j+1}) if $d(a_i, b_{j+1}) \leq r$, where $d(,)$ represents distance between two points. Discrete Fréchet distance between A and B is the infimum value of r such that in G_r there is a path between (a_1, b_1) and (a_n, b_m).

Here, we consider semi-discrete Fréchet distance which is, given a continuous curve S and a set of points P, the minimum length of a leash that simultaneously allows the owner to walk on S continuously and the frog to have discrete jumps from one point to another in P without backtracking. Hence the leash is allowed to switch discretely when frog jumps from one point to another. In this thesis, we consider the case when S is a line segment. We denote it by ℓ. We take the cardinality of P to be n. Let α is a continuous, non-decreasing, surjection from $[0,1]$ to ℓ. Suppose β be any function from $[0,1]$ to P such that there exists disjoint subdivisions of $[0,1]$ into a set of intervals $\lambda_1, \lambda_2, ..\lambda_k$ for some $k \in \mathbb{N}, k \leq n$. We have $\bigcup_{i=1}^{k} \lambda_i = [0,1]$. From any two points $t_1, t_2 \in [0,1]$, $\beta(t_1) = \beta(t_2)$ iff t_1 and t_2 belong to the same interval. Let $d(a,b)$ for two points a and b is the Euclidean distance between a and b. Semi- discrete Fréchet distance between ℓ and P is defined as:

$$d^F(\ell, P) = \underset{\alpha,\beta}{inf} \max_{t \in [0,1]} \{d(\alpha(t), \beta(t))\}$$

For an illustration, see Figure 1.8.

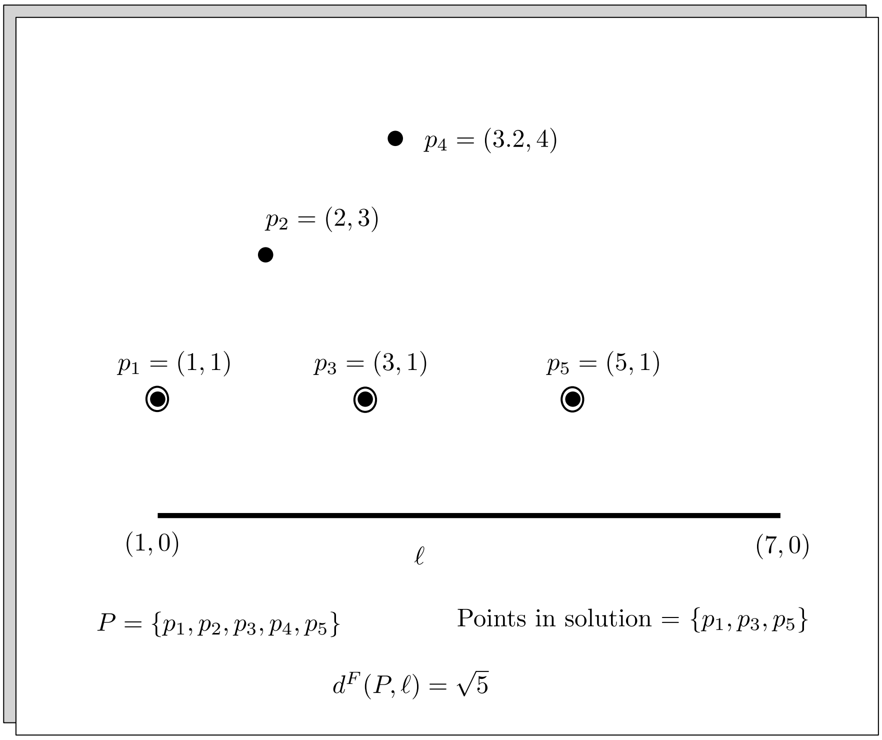


Figure 1.8 : Semi-discrete Fréchet distance

Our main point of consideration is introducing *conflicts* in this setting. Here instead of a set of points P, we are given a set of pairs of points $\mathcal{Q} = \{Q_1, Q_2, \dots Q_n\}$ in $\mathbb{R}^2$. We call $\mathcal{Q}$ a choice set. Our objective is to choose at most one point from each pair so that the length of leash needed is minimized. We call such a choice of points "conflict-free" with respect to the choice set $\mathscr{Q}$. Formally we define the problem as follows,

SEMI-DISCRETE FRÉCHET DISTANCE
Input: A set of points $P = \{p_1, p_2, \ldots, p_n\}$ and a line segment ℓ in $\mathbb{R}^2$.
Question: Find $d^F(P, \ell)$.

Next we consider the following problems involving choices.

CONFLICT-FREE FRÉCHET DISTANCE
Input: A set $\mathscr{Q} = \{Q_1, Q_2, \ldots Q_n\}$ of pairs of points, and a line segment ℓ in $\mathbb{R}^2$.
Question: Find a conflict free subset of points $P^* \subset \bigcup_{i=1}^{n} Q_i$ which minimizes $d^F(P^*, \ell)$

The natural decision versions of these problems are as follows.

CONFLICT-FREE FRÉCHET DISTANCE (DECISION VERSION)
Input: A set $\mathscr{Q} = \{Q_1, Q_2, \ldots Q_n\}$ of pairs of points, a line segment ℓ in $\mathbb{R}^2$, and $d \in \mathbb{R}$.
Question: Is there a conflict free set of points $P^* \subset \bigcup_{i=1}^{n} Q_i$ such that $d^F(P^*, \ell) \leq d$.

These problems are motivated by 2D curve fitting and object construction from noisy data which can be used in computer vision for data comparison and biomolecules structure comparison. Here the "resemblance" corresponds to minimizing the semi-discrete Fréchet distance. For example, given a noisy data with/without multiple choice constraints, we may construct a curve/object resembling the standard curve/object and may find the resemblance parameter (specified by semi-discrete Fréchet distance). To illustrate this, consider the scenario where we receive a discrete set of points from a curve. But due to errors in the channel, we get a bag of possible data points for each actual data point. At the end of the transmission, we have a set of bags where each bag contains a few points. We also have possible shapes of transmitted data in our database. Our objective is to choose one point from each bag such that the curve from the set of chosen points resembles most with one of the known shapes. If we know that the transmitted data is an alphabet from English language, then possible choices of curve for transmitted data can be the set of 26 alphabets.

We study approximation algorithm for CONFLICT-FREE FRÉCHET DISTANCE (minimization version) in this thesis. We also analyse the approximation results for GENERALIZED CONFLICT-FREE INTERVAL COVER where conflict graph has constant arboricity. We consider another case where both conflict graph and covering problem is represented by a unit interval graph. We define the problem as follows.

UNIT INTERVAL CF-SC
Input: A set of points P, a set of unit intervals $\mathscr{I} = \{I_1, \ldots, I_m\}$, a unit interval graph as conflict graph $CG_{\mathscr{I}}$ and a positive integer k.
Question: Does there exist a conflict free set cover covering atleast k points?

MAX UNIT INTERVAL CF-SC
Input: A set of points P, a set of unit intervals $\mathscr{I} = \{I_1, \ldots, I_m\}$, a unit interval graph as conflict graph $CG_{\mathscr{I}}$.
Question: Maximize the number of points covered using conflict free unit intervals?

For an example, see Figure 1.9.

Another problem we consider from approximation perspective is UNIT DISK - UNIT INTERVAL CF-SC.

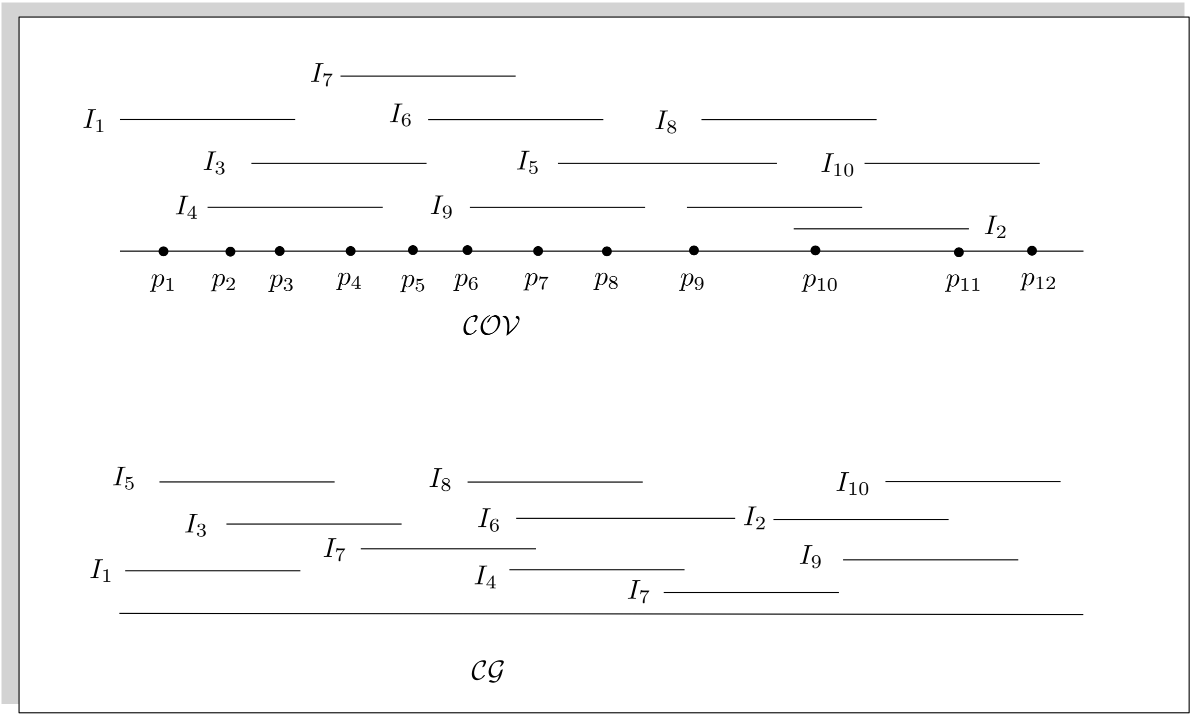


Figure 1.9 : Unit Interval CF-SC

We define the problem and its optimisation version as follows:

Unit Disk - Unit Interval CF-SC
Input: A set of points P in $\mathbb{R}^2$, a set of unit intervals $\mathscr{I} = \{I_1, \ldots, I_m\}$ covering P, a unit disk graph as conflict graph $CG_{\mathscr{D}}$ on a set of unit disks $\mathscr{D} = \{D_1, D_2, \ldots, D_m\}$ and a positive integer k.
Question: Does there exist a conflict free set cover covering atleast k points?

Max Unit Disk - Unit Interval CF-SC
Input: A set of points P in $\mathbb{R}^2$, a set of unit intervals $\mathscr{I} = \{I_1, \ldots, I_m\}$ covering P, a unit disk graph as conflict graph $CG_{\mathscr{D}}$ on a set of unit disks $\mathscr{D} = \{D_1, D_2, \ldots, D_m\}$.
Question: Maximize the number of points covered using conflict free unit intervals?

Here, two intervals $I_i, I_j \in \mathscr{I}$ are in conflict if the corresponding disks $D_i, D_j \in \mathscr{D}$ intersects.

1.6 OVERVIEW OF RESULTS AND STRUCTURE OF THESIS

We give some preliminaries related to parameterized complexity and approximation algorithms in Chapter 2.

We first study the conflicts in parameterized paradigm. In Chapter 3 this chapter, we study the problems where the conflict graph is either a matching or a collection of vertex disjoint cliques. We start by proving that Conflict-free Fréchet Distance (Decision Version) is NP-Complete. Next we show that Semi-discrete Fréchet Distance is solvable in $O(n \log n)$ time. Next we consider the parameterized complexity of the problem, i.e, Parameterized Conflict-free Fréchet Distance and show that it admits FPT. We give FPT algorithms based on randomization and branching for the problem. We further show that under standard complexity theoretic assumptions, the problem does not admit polynomial kernel.

In Chapter 4, we consider the conflicts in graphs with more complex settings. We restrict the conflict graphs to be belonging to the class of bounded arboricity graphs. As our first contribution, we propose two natural models in which the conflict relations are given:
(a) by a graph on the covering objects, and
(b) by a representable matroid on the covering objects. Our main result in this chapter is that as long as the conflict graph has bounded arboricity (that includes all the families of intersection graphs of low density objects in low dimensional Euclidean space), there is a parameterized reduction to the conflict-free version. This is achieved through a randomization-derandomization trick. As a consequence, we have the following results when the conflict graph has bounded arboricity.

- If the Geometric Coverage problem is fixed parameter tractable (FPT), then so does the conflict free version.
- If the Geometric Coverage problem admits a factor α-approximation, then the conflict free version admits a factor α-approximation algorithm running in FPT time.

As a corollary to our main result we get a plethora of approximation algorithms running in FPT time. Our other results include an FPT algorithm and a W[1]-hardness proof for the conflict-free version of Covering Points by Intervals. The FPT algorithm is for the case when the conflicts are given by a representable matroid, and the W[1]-hardness result is for all the family of conflict graphs for which the Independent Set problem is W[1]-hard.

Next we study conflicts in approximation paradigm. We study the case where conflict graph is a matching in Chapter 5. We present a 3-factor approximation algorithm to optimization version of Conflict-free Fréchet Distance.

Next, in Chapter 6, we propose a general framework where if the geometric graphs $\mathscr{COV}$ and $\mathscr{CG}$ satisfy some properties, then Graphical Conflict Free Set Cover admits constant factor approximation where the constant is based on those properties. As an application to this, we give an 8-factor approximation algorithm for Max Unit Interval CF-SC. We also present a 36-factor approximation algorithm for Max Unit Disk - Unit Interval CF-SC. We also prove the APX-hardness of various geometric problems. In this regard, we show that Max Unit Interval CF-SC and Max Unit Disk - Unit Interval CF-SC does not admit PTAS under standard computer theoretic assumptions. We prove APX-hardness for more general case where conflict graph is tree or 1-arboricity graph (thus holds for higher arboricity graphs too) in point interval covering. We also show problem is APX-hard when both $\mathscr{CG}$ and $\mathscr{COV}$ are unit coin graphs.