JCALM 2007
Submodular functions

Lecture 1: Introduction to submodular functions

Binh-Minh Bui-Xuan

Foreword

This short introduction to submodular functions is meant for the second JCALM (Journée Combinatoire et Algorithmes du Littoral Méditéranéen) on April 26-27, 2007, at the Lirmm. We borrow heavily on Maurice Queyranne's seminar talk [1].

1 Definition

Let V be a finite set. The set of all subsets of V will be denoted by 2^V. A set function $f:\ 2^V \to \mathbb{R}$ is *submodular* if it meets the following equivalent conditions

i. $\forall A, B \in 2^V,\ f(A) + f(B) \geq f(A \cup B) + f(A \cap B)$
ii. $\forall A \subseteq B \subseteq V,\ \forall v \in V \setminus B,\ f(A+v) - f(A) \geq f(B+v) - f(B)$
iii. $\forall A \subseteq V,\ \forall u, v \in V \setminus A,\ f(A+u) - f(A) \geq f(A+u+v) - f(A+u)$,

where $A + v$ stands for $A \cup \{v\}$, provided that $v \notin A$.

We say that f is *supermodular* if $(-f)$ is submodular, and *modular* if it is both submodular and supermodular.

2 Examples

2.1 Example 1: Numerical Submodular Functions

From the fact that $|A|+|B| = |A \cup B|+|A \cap B|$ the following function is by definition modular:

$$\begin{aligned} f:\quad & 2^V \to \mathbb{N} \\ & A \mapsto |A|. \end{aligned}$$

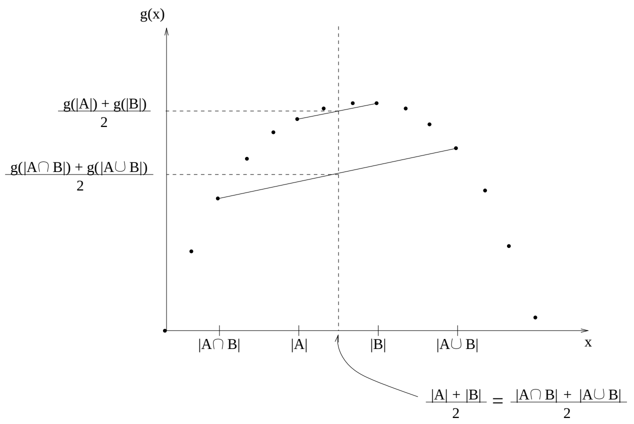

Fig. 1. Concavity and chords.

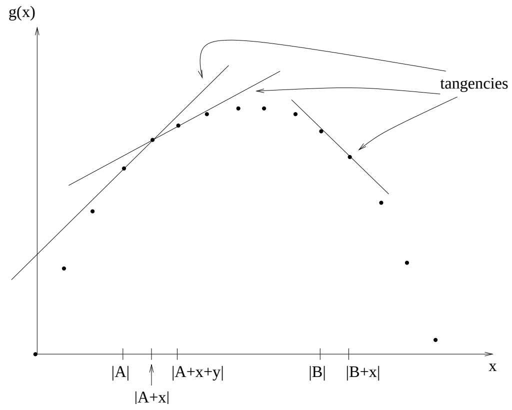

Fig. 2. Concavity and derivative.

More generally, if a submodular function only depends on the cardinality of the given subsets, namely if there exists $g :\ \mathbb{N} \to \mathbb{R}$ such that $f(A) = g(|A|)$, then

$$f \text{ is submodular } \Leftrightarrow g \text{ is concave},$$

with the following slight abusiveness on the definition of concavity. A real valued function g on a discrete domain is concave if $g(\frac{x+y}{2}) \geq \frac{g(x)+g(y)}{2}$ for all x, y in the domain of g such that $\frac{x+y}{2}$ also is in the domain of g. This highly relates to the condition i. in the definition of submodularity of Section 1 (see Figure 1).

Notice moreover that, if g is differentiable, its concavity is equivalent to the decreasing monotonicity of its derivative. This could be seen as an informal illustration of conditions ii. and iii. in the definition of submodularity of Section 1 (see Figure 2).

Examples of differentiable concave functions on $\mathbb{R}$ include the negation of a power $x \mapsto -x^2$ and the logarithm function $x \mapsto \log x$. An example on $[0, 1]$ can be the entropy function $p \mapsto -p \log p - (1 - p) \log(1 - p)$.

2.2 Example 2: Cut functions (a.k.a. border functions)

Let $G = (V, A)$ be a digraph. One might want to put a weight function on the arc set with $\omega : A \to \mathbb{R}^+$. The *cut*, or *border*, of a vertex subset $U \subseteq V$ is defined as

$$\delta^+(U) = \{(u, v) \in A \mid u \in U \text{ and } v \notin U\}.$$

The *cut function* on G is the function $f : 2^V \to \mathbb{R}$ defined as

$$f(U) = \omega(\delta^+(U)) = \sum_{a \in \delta^+(U)} \omega(a).$$

When ω is a *positive valued* function, the cut function f is submodular. Indeed, if U and W are two vertex subsets, there are three types of arcs, defined in Figure 3: type 1 arcs are counted once in both $f(U) + f(W)$ and $f(U \cup W) + f(U \cap W)$; type 2 arcs are counted twice in both $f(U) + f(W)$ and $f(U \cup W) + f(U \cap W)$; type 3 arcs are counted once in $f(U) + f(W)$ and not in $f(U \cup W) + f(U \cap W)$.

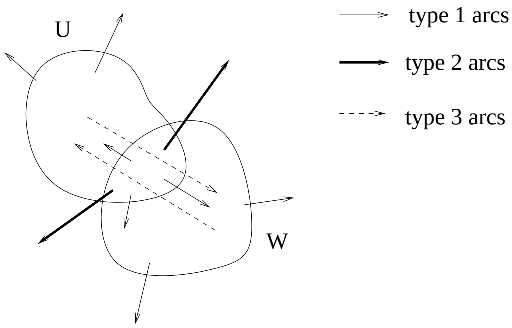

Fig. 3. Three types of arcs involved with two vertex subsets U and W.

2.3 Example 3: Spanning forests in a graph

Let $G = (V, E)$ be a graph. A *spanning forest* of G is a collection $F \subseteq E$ of edges of G such that the partial subgraph (V, F) is cycle-free. (A *spanning tree* is a connected spanning forest.)

The *graphic rank* $r(H)$ of a subset of edges H is the maximum size of a spanning forest of the partial subgraph (V, H):

$$r(H) = \max\{\ |F|\ ,\ F \subseteq H \text{ and } (V, F) \text{ is a spanning forest}\}.$$

The *graphic rank function* denotes

$$\begin{aligned} r:\ 2^E &\to \mathbb{N} \\ H &\mapsto r(H) \end{aligned}$$

We have the two following properties. First, the graphic rank function is submodular. Second, the *number of connected components* in the partial subgraph (V, H) is

$$c(H) = |V| - r(H)$$

(which by definition leads to a supermodular function).

2.4 Example 4: Matroid rank

Let V be a finite set, and $\mathcal{F} \subseteq 2^V$ a family of subsets of V. The *rank function* of (an arbitrary subset family) $\mathcal{F}$ is defined as

$$\forall A \in 2^V\ ,\ r(A) = \max\{\ |B|\ ,\ B \subseteq A \text{ and } B \in \mathcal{F}\}.$$

We say that $\mathcal{F}$ is a *matroid* if it satisfies the three following conditions.

- $\emptyset \in \mathcal{F}$;
- if $(A \subseteq B)$ and $(B \in \mathcal{F})$ then $A \in \mathcal{F}$;
- if $(A \in \mathcal{F})$ and $(B \in \mathcal{F})$ and $(|A| < |B|)$ then there exists $v \in B \setminus A$ such that $A + v \in \mathcal{F}$.

Conversely, let $r:\ 2^V \to \mathbb{N}$ be an arbitrary positive valued set function. Let $\mathcal{F} = \{\ A \in 2^V\ ,\ r(A) = |A|\ \}$. Then, $\mathcal{F}$ is a matroid if and only if for all $A, B \in 2^V$

- $0 \leq r(A) \leq |A|$,
- $A \subseteq B \Rightarrow r(A) \leq r(B)$, and
- r is submodular.

3 Links to Greedy Algorithms

A popular relation between matroids and greedy algorithms could be stated as follows. Let V be a finite set, $\mathcal{F} \subseteq 2^V$ a family of subsets of V, and $\omega:\ V \to \mathbb{R}^+$ a *positive* weight function. For convenience, we denote $\omega(A) = \sum_{v \in A} \omega(v)$ for all subset A of V.

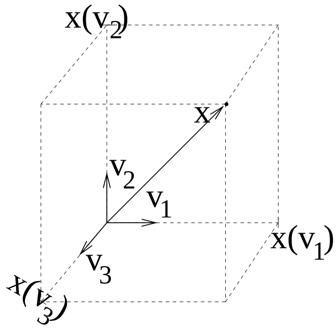

Fig. 4. Here $V = \{v_1, v_2, v_3\}$ and $x :\ V \to \mathbb{R}$.

The $\textsc{Max}(V, \mathcal{F}, \omega)$ problem consists of finding the elements of $\mathcal{F}$ with maximum weight, that is $X \in \mathcal{F}$ such that $\omega(X) = \max_{A \in \mathcal{F}} \omega(A)$. The $\textsc{Greedy}(V, \mathcal{F}, \omega)$ algorithm proceeds as follows.

- Sort the elements of V with respect to ω: $\omega(v_1) \geq \omega(v_2) \geq \cdots \geq \omega(v_n)$.
- Let $X = \emptyset$. For i varying from 1 to n, if $X + v_i$ belongs to $\mathcal{F}$, replace X with $X + v_i$.

Then, the Greedy algorithm solves the Max problem if and only if $\mathcal{F}$ is a matroid.

This result has been improved to submodular polyhedra by Edmonds in 1970 [1] [2–4]. However, before showing the result, let us discuss on some terminologies. A subset A of V can also be seen as a choice function on the elements of the ground set $A :\ V \to \{0, 1\}$. This is why the set of all subsets of V usually is denoted by 2^V. We now relax this notion and consider the set $\mathbb{R}^V$ of all real valued functions $x :\ V \to \mathbb{R}$.

Obviously, a subset $A :\ V \to \{0, 1\}$ of V is an element of $\mathbb{R}^V$. In particular, a singleton $A = \{v\}$ with $v \in V$ is an element of $\mathbb{R}^V$. Then, each element $x :\ V \to \mathbb{R}$ of $\mathbb{R}^V$ can be seen as a vector in the $\mathbb{R}-$vector space spanned by the basis $\mathcal{B} = \{\ \{v\} \mid v \in V\ \}$ formed by all singletons of V (see Figure 4). For all $v \in V$, $x(v)$ is the projection of the vector x w.r.t. the base vector $\{v\} \in \mathcal{B}$, which, not without a certain abusiveness, will also be denoted by v (see Figure 4).

If $B \in 2^V$ is seen as a sub-basis of $\mathcal{B}$, we define $x(B) = \sum_{v \in B} x(v)$. Let $f :\ 2^V \to \mathbb{R}$ be a function over the subsets of V. The *f-polyhedron* is defined as

$$P(f) = \{\ x \in \mathbb{R}^V \mid \forall B \in 2^V\ ,\ x(B) \leq f(B)\ \}.$$

The polyhedron $P(f)$ is called *submodular polyhedron* if f is submodular.

We now define a linear programming problem $\textsc{MaxLP}$ as follows. If V is a finite

[1] The original version [2] probably is unaccessible now. However, you still can buy the revised article at the hyperlink given with the reference [3]. Also, it seems that this result is better explained by Lovász in [4]. Since I could not access Lovász paper, I favoured Queyranne's terminologies

set, $\omega : V \rightarrow \mathbb{R}^+$ a positive weight function over V, and $f : 2^V \rightarrow \mathbb{R}$ a set function over the subsets of V, then MaxLP(V, f, ω) consists of finding $x \in P(f)$ such that $\omega^t x$ is maximum.

The Polyhedron-Greedy(V, f, ω) algorithm proceeds as follows.

- Sort the elements of V with respect to ω: $\omega(v_1) \geq \omega(v_2) \geq \cdots \geq \omega(v_n)$.
- Let $B_0 = \emptyset$.
- For i varying from 1 to n, do
  - $B_i \leftarrow B_{i-1} + v_i$,
  - $x(v_i) \leftarrow f(B_i) - f(B_{i-1})$.

Theorem (Edmonds): The Polyhedron-Greedy algorithm solves the MaxLP problem if and only if f is submodular (see e.g. [2–4]).

References

[1] M. Queyranne. `http://www.ima.umn.edu/optimization/seminar/queyranne.pdf`

[2] J. Edmonds. Submodular functions, matroids, and certain polyhedra. *Combinatorial Structures and Their Application*, pages 68–87, 1970.

[3] J. Edmonds. Submodular functions, matroids, and certain polyhedra. *Combinatorial Optimization - Eureka, you shrink!*, pages 11–26, 2003. `http://www.springerlink.com/content/bnv45cwcne26yxvl/`

[4] L. Lovász. Submodular functions and convexity. *Mathematical Programming: The State of the Art – Bonn 1982*, pages 235–257, 1983.

Lecture 2: Minimizing submodular functions

Thierry Monteil and Stéphan Thomassé

Foreword

We present Schrijver's algorithm for minimizing a submodular function [1].

1 Introduction.

Let V be a finite set. A function f defined from the set 2^V of all subsets of V into the reals is *submodular* if $f(A)+f(B) \geq f(A \cup B)+f(A \cap B)$, for all $A, B \subseteq V$. The topic of this lecture is to present a way of minimizing f over 2^V, i.e. finding a subset $A \subseteq V$ for which $f(A)$ is minimum. Since we are concerned with an efficient algorithm (polynomial in $|V|$), the way in which the function f is coded is of importance - we will assume that an oracle can calculate in constant time any value $f(A)$ which is asked for. Let us now turn to some examples in which appears this minimization problem:

- *Cut functions.* We have already seen that the *border* function δ defined on subsets of vertices of a graph $G = (V, E)$, where $\delta(X)$ is the number of edges across X and $V - X$, is a submodular function. Minimizing δ results in a not-so-exciting problem since the emptyset (or equivalently the whole set of vertices) certainly reaches the minimum value of 0. However, if we fix two distinct vertices u, v of our graph, minimizing δ over all subsets of vertices containing u and not v is the classical MINCUT problem of finding the minimum number of edges which disconnect u from v.
- *Weighted matroids.* The rank function of a matroid $\mathcal{M}$ on ground set E is defined by letting $\mathrm{rk}(F)$ to be the maximum size of an independent set of $\mathcal{M}$ contained in F. This function is submodular, and again minimizing it is a triviality since the emptyset has rank 0. Now assign a real valued weight function w on E, which extends naturally to 2^E by letting the weight of a subset be the sum of the weights of its elements. This function w is submodular (indeed it is modular), hence the function $\mathrm{rk} + w$ is submodular. Minimizing it is now much less easy.
- *Matroid intersection.* König's theorem asserts that in a bipartite graph (X, Y, E), the size of a maximum matching is equal to the minimum size of a set of vertices incident to all edges. A particularly complicated way of stating this result is the following: the subsets F of edges of E such that every vertex of X has degree at most one in F form the independent sets of a matroid $\mathcal{M}_X$ on the ground set E. Indeed, this is a partition matroid. We similarly define $\mathcal{M}_Y$. The intersection of $\mathcal{M}_X$ and $\mathcal{M}_Y$ consists of these subsets of E which are both independent in $\mathcal{M}_X$ and $\mathcal{M}_Y$. In other words, these are subsets of edges which have degree at most

one in X and Y - a weird way of defining a brave old matching. In this particular case of matroid intersection, König's theorem says that finding the maximum size of a common independent set is an easy problem. More generally Edmonds completely characterized the matroid intersection problem as follows: let $\mathcal{M}$ and $\mathcal{M}'$ be two matroids on a ground set E with respective rank function rk and rk$'$. The maximum size of a common independent set of $\mathcal{M}$ and $\mathcal{M}'$ is equal to the minimum of $\Phi(F) := \text{rk}(F) + \text{rk}'(E - F)$ over all subsets F of E. This function Φ, as a sum of two submodular functions, is in turn submodular, and the matroid intersection maximum independent set problem therefore consists of minimizing a submodular function.

2 The base polytope.

This section consists of results and observations which are not absolutely necessary for presenting the minimization algorithm but since the whole method is based upon these notions, a short survey is needed. The idea, due to Cunningham [3], is to approach the submodular function we want to minimize by some modular functions.

Recall that a function f defined from 2^V into the reals is *modular* if $f(A) + f(B) = f(A \cup B) + f(A \cap B)$. An example of a modular function is obtained by taking any weight function on V and extending it to the subsets of V. Every modular function is not of this form since weight functions value the emptyset by 0. However, every modular function f with $f(\emptyset) = 0$ is a weight function. Indeed, for every subset F and any partition (A, B) of F, we get $f(F) = f(A) + f(B)$. Iterating this, we get that $f(F)$ is the sum of the weight of its elements.

Thus if one would like to use modular functions to approach submodular ones, the first step is to consider only submodular functions f for which $f(\emptyset) = 0$. This is certainly not a strong restriction since one can always shift f by the constant value $-f(\emptyset)$. In particular, the minimum value we are looking for is nonpositive. We now introduce a result of Edmonds [4], which gives a dual statement for the minimization problem. The *base polytope* $B(f)$ of f consists of all weight functions x with $x(V) = f(V)$ and such that for all $A \subseteq V$, we have $x(A) \leq f(A)$:

$$B(f) = \{x \in P(f) \mid x(V) = f(V)\}.$$

Although it is not clear that this polytope is even non empty (we will prove it later in Lemma 1), let us state the key-result of this section. Here the *negative part* of a weight function x is the sum over V of its negative values and is denoted by $x^-(V)$.

Theorem 1 *The minimum value of f over 2^V is equal to the maximum of the negative part of a weight function x over $B(f)$:*

$$\min\{f(X) \mid X \subseteq V\} = \max\{x^-(V) \mid x \in B(f)\}.$$

Proof: The inequality $\geq$ is clear. On the other hand, let us define for any $X \subseteq V$, $f^0(X) = \min\{f(Y) \mid Y \subseteq X\}$. f^0 is submodular and nonpositive. Now let us choose x^0 in $B(f^0)$. Since $f^0 \leq f$, we have $B(f^0) \subseteq P(f^0) \subseteq P(f)$, so we can choose x in $B(f)$ such that $x^0 \leq x$. We have:

$$\min\{f(X) \mid X \subseteq V\} = f^0(V) = x^0(V) \leq x^-(V).$$

□

This min/max results gives a hint for our original problem. Instead of minimizing f, we can equivalently try to maximize the negative part of some weight functions. This poses now the problem of describing the elements of the base polytope. It turns out that the situation is not so bad, since it is possible to generate all the vertices of this polytope. Indeed, they can be constructed in the nice following combinatorial way:

Let $<$ be a total order $v_1 < v_2 < \cdots < v_n$ on the elements of V. We inductively define a weight function $x^<$ as follows: we set $x^<(v_1) = f(v_1)$, then $x^<(v_2) = f(\{v_1, v_2\}) - x^<(v_1)$, and, inductively, if $x^<$ have been defined for v_i, we let $x^<(v_{i+1}) = f(S_{i+1}) - x^<(S_i)$, where S_j is the set $\{v_1, \ldots, v_j\}$. By construction, we obtain a weight function, which we call a *permutation weight function*.

Lemma 1 *Every permutation weight function belongs to the base polytope.*

Proof: First observe that for every set S_i we have

$$x^<(S_i) = x^<(S_{i-1}) + x^<(v_i) = x^<(S_{i-1}) + f(S_i) - x^<(S_{i-1}) = f(S_i),$$

in particular $x^<(V) = f(V)$. Now, given a subset Y of V, we want to prove that $f(Y) \geq x^<(Y)$. The proof goes by induction on the size of Y. Assume that v_i is the maximum vertex of Y with respect to $<$. We use the submodularity of f to get $f(Y \cap S_{i-1}) + f(Y \cup S_{i-1}) \leq f(Y) + f(S_{i-1})$. By induction and the fact that $x^<(S_{i-1}) = f(S_{i-1})$, we get $x^<(Y - v_i) + f(S_i) \leq f(Y) + x^<(S_{i-1})$. Hence $x^<(Y - v_i) + x^<(v_i) \leq f(Y)$, and since $x^<(Y) = x^<(Y - v_i) + x^<(v_i)$, we have our conclusion. □

This proves at least that the base polytope is not empty, and, indeed, the following result completely characterizes it:

Theorem 2 *The extremal points of the base polytope are exactly the permutation weight functions.*

Proof: (sketch) By construction, the Polyhedron-Greedy algorithm described in the first lecture always ends to a permutation weight function. Since any extremal point of a polytope can be detected by a convenient linear form $x \mapsto \omega^t x$, the extremal points of $B(f)$ are permutation weight functions. Conversely, any permutation weight function corresponds to the case of equality of n independent equalities $x^<(S_i) \leq f(S_i)$ defining $P(f)$, so there are all extremal points. □

Thus every element of the base polytope is a barycentric combination of permutation weight functions. This is how the minimization algorithm works for the submodular function f: the idea is to find linear combinations of permutation weight functions in order to approach f.

3 The algorithm.

We start with an element $x = x^{<}$ of $B(f)$ ($<$ is any total order on V).

Checking directly that a given weight function x is in $B(f)$ is hard. Hence, we will maintain it as a convex combination $x = \sum_{i \in I} \lambda_i x^{<_i}$ during the whole algorithm. The only *a priori* bound on the size of I is $n!$. But thanks to Caratheodory Theorem, we can continually reorganize the convex combination to ensure that the size of I never exceeds n so this data does not explode. Note that this is not clear that such a transformation can be done in such a way that the size (in memory) of the λ_i do not explode, we admit that we can describe those numbers and this transformation in an efficient way.

To such a combination we associate an oriented graph whose vertices is the set V and there is an edge from v to w if there is some i in I such that $v <_i w$. In this graph we define two sets $N = \{v \in V | x(v) < 0\}$ and $P = \{v \in V | x(v) > 0\}$. There are two cases.

First, suppose that at some step, there is no path from P to N: the set U of vertices that can reach N by a directed path contains N and and is disjoint from P. For each i in I, U is an initial segment for the order $<_i$ so $x^{<_i}(U) = f(U)$ and by convex combination $x(U) = f(U)$. Now, if $W \subseteq V$ we have:

$$f(W) \geq x(W) \geq x(U) = f(U).$$

So U minimizes f: this is our termination condition.

Second, we are in the case where there is a path from P to N. We have to make a "small" move along $B(f)$ that will improve the quality of x. A move from x to x' in $B(f)$ corresponds in particular to add to x a weight function of total mass 0, The most elementary are those where we increase the weight of an element t of V by $\delta \geq 0$ and decrease the weight of another element s of V by the same value. The quality of x is improved when $(t, s) \in N \times P$. It turns out that if $s <_i t$, we are able to describe effectively $x^{<_i} + \delta(\chi_t - \chi_s)$ as a convex combination of the $x^{<'}$ where the $<'$ are obtained by inserting an element of $(s, t]_{<_i}$ just before s (for a particular $\delta \geq 0$).

We now make such choices in an extremal way to be able to bound the complexity: we choose t in N such that the distance in the graph from P to t is maximal, and we choose s just before t on a minimizing path from P to t. Hence, for some i in I, $s <_i t$ and we can chose i such that $[s,t]_{<_i}$ has maximal length: this is the triple (t,s,i) on which we execute the elementary move. If necessary, we reduce δ to ensure that t won't become of positive weight. All this ensures that no element from N will pass in P and that no distance from P to a given vertex decreases. If t and s are chosen in a consistent way from a step to the next one, Schrijver proves that one distance from P really increases after at most $|V|^4$ iterations. Hence the algorithm finishes after at most $|V|^6$ iterations since then all distances from P to N will be greater than $|V|$ hence infinite, leading us to the first case.

Another polynomial time algorithm for submodular function minimization appears in the same time in [5], it has then been improved to minimize submodular functions defined in a general ordered group $(G,+,<)$ (note that in Schrijver's algorithm, we use the fact that f has real values where extra-operations such as multiplications and divisions are allowed). A comparison of those different approaches leading to polynomial minimization of submodular functions can be found in [2].

References

[1] A. Schrijver. A combinatorial algorithm minimizing submodular functions in strongly polynomial time. *Journal of Combinatorial Theory. Series B*, 80(2):346–355, 2000. `http://homepages.cwi.nl/~lex/files/minsubm6.pdf`

[2] L. Fleischer. Recent Progress in Submodular Function Minimization. *OPTIMA: Mathematical Programming Society Newsletter*, 64:1–11, 2000. `http://www.ise.ufl.edu/~optima/optima64.pdf`

[3] W. Cunningham. On submodular function minimization. *Combinatorica*, 5(3):185–192, 1985.

[4] J. Edmonds. Submodular functions, matroids, and certain polyhedra. *Combinatorial Structures and their Applications (Proc. Calgary Internat. Conf., Calgary, Alta., 1969)*, pages 69–87, 1970

[5] S. Iwata, L. Fleischer, and S. Fujishige. A combinatorial, strongly polynomial-time algorithm for minimizing submodular functions. In *Proceedings of the Thirty-Second Annual ACM Symposium on Theory of Computing*, pages 97–106, 2000. `http://www.misojiro.t.u-tokyo.ac.jp/~iwata/sfm.pdf`

A list of open problems

Florian Huc

These open problems were proposed during the JCALM which took place in Montpellier on April 26th and 27th, 2007.

1 Questions asked by Frédéric Havet:

Coloring square of graphs:

A coloring of a graph $G = (V, E)$ is a function $c : V \to S$ s.t. $\forall uv \in E(G), c(u) \neq c(v)$. The square of a graph $G = (V, E)$ is a graph on vertex V and with $uv \in E(G^2)$ iff $d_G(u, v) \leq 2$ and $u \neq v$.

The problem is to find upper bounds on $\chi(G^2)$ and $\omega(G^2)$ in terms of $\Delta(G)$, where $\chi(G)$ and $\omega(G)$ are respectively the minimun number of colours needed to colour G and the size of the biggest clique which is a subgraph of G.

First remark that $\Delta(G^2) \leq \Delta(G) + \Delta(G)(\Delta(G) - 1) = \Delta(G)^2 + 1$.

This is tight but achieved only by few graphs, namely : C_5, Peterson graph, Hoffman-Singleton graph, and maybe one with $\Delta = 51$

What can we say about $M_\Delta = \max_{G\ with\ max\ degree\ \Delta}(\omega(G^2))$?

For Δ large we have $\Delta^2 + 1 \geq M_\Delta \geq \Delta^2 - \Delta + 1$.

The lower bound is given by the projective plane but only for some values of Δ. (such a projective plane contains $\Delta^2 - \Delta + 1$ points and as many lines such that each line contains Δ vertices and each vertex is in Δ lines). Given a projective plane we construct a bipartite graph G such that the left handside vertices represent the points while the right handside vertices represent the lines. There is an edge in between a left and a right vertex if the corresponding point is in the corresponding line. In G^2 we have cliques of size $\Delta^2 - \Delta + 1$ since both parts become a clique.

The conjecture is that if Δ is large enough, for G with max degree Δ then $\chi(G^2) \leq M_\Delta$.

Also, considering only subcubic planar graph, we have the following theorem:

Theorem 1 (Thomassen) *Let G be a planar and subcubic graph $\chi(G^2) \leq 7$.*

This theorem can be proved by the following conjecture:

Conjecture 1 *Let G planar and subcubic. We can partition $V(G)$ into V_1 and V_2 so that $G(V_1)$ has max degree 1 and $G(V_2)$ has min degree 1 and no P_4.*

Considering list coloring, we have the following:

Conjecture 2 *Let G planar and subcubic. $\chi_l(G^2) \leq 7$.*

Conjecture 3 (Kostochka and Woodall) *Let G planar and subcubic. $\chi_l(G^2) = \chi(G^2)$.*

How big can $\chi(G^2)$ be in terms of $\omega(G^2)$? There exist grahs with $\chi(G^2) \geq 5/4\omega(G^2)$

2 Question asked by Stéphanne Bessy:

Consider a symmetric grid network (i.e. each edge of the grid represents two arcs, one in each direction) with on each vertex instances to achieve. To achieve these instances only paths with one turn are allowed. Furthermore no arcs of the grid can be used more than 2 times, i.e. the maximum authorized load is 2.

Definition 1 *The Conflict graph is G_C with vertex set $V_C = \{paths\ used\ for\ the\ routing\}$ and edge set $E_C = \{P_1P_2,\ withP_1\ and\ P_2\ sharing\ an\ edge\ in\ the\ grid\}$.*

If given a column of the grid you consider only the paths using this column, G_C reduced to these paths is a caterpillar. Same for the lines. Consequently, we can see that $|E| \leq 2|V| - 2$. The fact that G_C is caterpillar implies that G is 3-degenerate and so $\chi \leq 4$.

Question: is $\chi = 3$ or 4?

What happens if we increase the maximum authorized load?

3 Question asked by Omid Amini:

The following problem is due to Chen and Chvátal [2]:

Let $G = (V, E)$ be a connected graph. The function $d : V \times V \to \mathbb{N} \cup \{0\}$ is the distance function in G, i.e. for two vertices x and y in V, it gives the length of the shortest path between x and y. For two vertices x and y in V, the line defined by

x and y, denoted $\ell(x, y)$, is the collection of all points $v \in V$, such that one of the following equalities is verified:

- $d(x, y) = d(x, v) + d(v, y)$
- $d(x, v) = d(x, y) + d(y, v)$
- $d(v, y) = d(v, x) + d(x, y)$

Example 1 *(1) Let $e = \{x, y\}$ be a bridge in G. Then the line defined by x and y contains all the vertices of G.*

(2) Let $e = \{x, y\}$ be an edge. Then $\ell(x, y) = V \setminus \{v \in V \mid d(v, x) = d(v, y)\}$. In this case all $v \notin \ell(x, y)$ appears on an odd cycle containing x and y. This implies for example that if G is bipartite then every line defined by an edge contains all the vertices.

Conjecture 4 *Let $G = (V, E)$ be a connected graph on n vertices. Suppose that for every two vertices $x, y \in V$ we have $\ell(x, y) \neq V$. Then G contains at least n different lines.*

Remark: The origin of this conjecture is a theorem of De Bruijn-Erdös [3], which says that every set of n noncollinear points in the plane defines at least n different lines. A nice proof of this result can be obtained by induction, using the fact that for such a set there exists always a line which contains exactly two points. This last statement is also true for every finite metric space, and in particular for graphs, see [1].

References

[1] X. Chen, *The Sylvester-Chvátal Theorem*, `http://users.encs.concordia.ca/ chvatal/chen.pdf`

[2] X. Chen and V. Chvátal, *Problems related to a de Bruijn - Erdös theorem*, `http://users.encs.concordia.ca/ chvatal/dbe.pdf`

[3] N.G. de Bruijn and P. Erdös, *On a combinatorial problem*, Indigationes Mathematicae 10 (1948), 421-423.

4 Question asked by Florian Huc:

Conjecture 5 (The removal path conjecture of Lovász) *There exists a function $f = f(k)$ such that the following holds: for every $f(k)$-vertex-connected graph G and two distinct vertices s and t in G, there exists a path P with endpoints s and t such that $G - V(P)$ is k-connected.*

Remark that a weaker version of this conjecture asking only for the removal of the edges has been proved in [1]. This paper also gives some direction that have been tried to approach this conjecture.

References

[1] Ken-ichi Kawarabayashi and Orlando Lee and Bruce Reed and Paul Wollan, *Progrees on Lovász Path Removal Conjecture*, http://www.math.uni-hamburg.de/home/wollan/lovconj_web.pdf

5 Questions asked by Yan van den Heuvel:

5.1 First Question

Problem 1 (Kotzig) *Given an integer k, $k \geq 3$, does there exist a graph G with the property that for any two vertices there is a unique path of length k between them?*

It is true for $k = 1$ and 2.

5.2 Second Question

This other question is originally from Rota:

Conjecture 6 (Rota) *Let V be an n-dimensional linear space and $B_1, \dots, B_n$ be n bases. There exists an $n \times n$ array made of elements of $B_1, \dots, B_n$ such that the line i is made of elements of B_i and that each column forms a base.*

It is known for $n = 1, 2$ and 3.

A special case is when the problem is restricted to graphs: Let $T_1, \dots, T_n$ be spaning trees, can we put theier edges in a nxn array such as before so that each columns form a tree?

6 Questions asked by Stephan Thomassé:

6.1 First Question

Conjecture 7 (Kuhn, Osthus) *Let G and H be two graphs on a same vertex set V. There exist a cut (V_1, V_2) with at least half of the edges of both H and G: $e_G(V_1, V_2) \geq \lfloor e(G)/2 \rfloor$ and $e_H(V_1, V_2) \geq \lfloor e(H)/2 \rfloor$.*

Remark: $\geq$ cannot be replaced by $>$, in particular due to C_5. Furthermore we can achieve easily $e(G) - \sqrt{e(G)}$ and $e(G) - \sqrt{e(G)}$.

If we consider 3 graphs instead of 2, we can also achieve $e(G) - c\sqrt{e(G)}$ for some c.

6.2 Second Question

An other conjecture is related to the following fact:

Claim 2 *Let D be a digraph with minimun outdegrre 3, then the vertices can be partitionned into two sets V_1 and V_2 such that for all i, $\forall u \in U_i$, u has a neighbour in V_i.*

Conjecture 8 *There exists a function $f(k)$ such that every digraph with minimun outdegree $f(k)$ has a partition of its vertices into two sets V_1 and V_2 such that each vertex in V_1 has k outneighbours in V_1 and every vertex in V_2 has at least one outneighbour in V_2.*

What happens if we ask for a greater minimun degree?