Chapter 11

Graph Separators

The focus of this note is on the sparsest cut in a capacitated, undirected graph. It turns out that the problem of finding a sparsest cut is NP-hard, yet the problem is interesting and important. The problem has applications to divide-and-conquer algorithms for approximately solving many hard combinatorial problems, such as balanced cuts, feedback arc set, minimum fill-in ordering, VLSI layout, minimum crossing-number layout, etc.

First, we will motivate and prove a key theorem due to Leighton & Rao showing that the objective value of a sparsest cut is within a logarithmic factor of the optimal value of an LP (linear programming) relaxation of the sparsest cut problem, and moreover, there is an efficient algorithm for finding such an approximately sparsest cut, i.e., a cut whose objective value is within a logarithmic factor of the objective value of the sparsest cut.

After that, we introduce the notions of metric spaces and ℓ_1-embeddings, discuss some of the connections to sparsest cuts and multicommodity flows, and prove a generalization of the Leighton-Rao theorem and algorithm, based on results of Bourgain and others. For completeness, we include a proof of Bourgain's theorem: by allowing a logarithmic distortion, any metric space has an ℓ_1-embedding.

Finally, we discuss a paradigm for approximately solving hard combinatorial problems by using a subroutine for finding approximately sparsest cuts. This paradigm easily gives a log-squared approximation guarantee.

11.1 The sparsest cut problem and an LP relaxation

Let $G = (V, E)$ be an undirected graph, and let n denote $|V|$. Let $u : E \to \Re_+$ assign a nonnegative capacity to each edge. Throughout this chapter, for a cut $\delta(S)$ of G, the *ratio* means the quantity $\dfrac{u(\delta(S))}{|S| \cdot |V \backslash S|}$, and a *sparsest cut* is one whose ratio is minimum. (Note that $\delta(S)$ is a set of edges, but we also associate the node set S with it.)

Let us write down an LP formulation of the sparsest cut problem. For motivation, let us quickly recapitulate the LP for the minimum s-t cut problem. The goal is to assign a nonnegative weight d_e to each edge $e \in E$ such that $\sum_{e \in E} u_e d_e$ is minimized, while the weight of a shortest path between s and t (w.r.t. edge weights d) is at least one. It is easily checked that the 0-1 incidence vector of every s-t cut is a feasible solution to this LP, i.e., taking $d_e = 1$ iff e is in the s-t cut satisfies all constraints of the LP. Moreover, by the max-flow min-cut theorem, one optimal solution to the LP is given by the 0-1 incidence vector of a minimum s-t cut.

Here is our first attempt at formulating the sparsest cut problem as a "fractional LP", i.e., as an LP where the objective function is rational rather than linear. For two nodes v and w, we use $\mathcal{P}_{v,w}$ to denote the set of all v-w paths in the graph; the cardinality of $\mathcal{P}_{v,w}$ may be exponential in n.

$$\begin{array}{llllll}
\text{(F1) } z = \text{minimize} & \dfrac{\sum_{e\in E} u_e d_e}{\sum_{v,w\in V} \lambda_{v,w}} & & & \\
\\
\text{subject to} & \sum(d_e \mid e \in P) & \geq & \lambda_{v,w} & \forall \text{ paths } P \in \mathcal{P}_{v,w} \\
& d_e & \geq & 0 & \forall e \in E \\
& \lambda_{v,w} & \geq & 0 & \forall v, w \in V.
\end{array}$$

Above, the edge weights d_e are supposed to represent incidence vectors of cuts, and the variables $\lambda_{v,w}$ are supposed to represent the weight of a shortest path (w.r.t. d) between nodes v and w. To obtain an LP formulation, we remove the denominator from the objective function and add an extra constraint that the denominator be at least one.

$$\begin{array}{llllll}
\text{(F2) } z = \text{minimize} & \sum_{e\in E} u_e d_e & & & \\
\text{subject to} & \sum_{v,w\in V} \lambda_{v,w} & \geq & 1 & \\
& \sum(d_e \mid e \in P) & \geq & \lambda_{v,w} & \forall \text{ paths } P \in \mathcal{P}_{v,w} \\
& d_e & \geq & 0 & \forall e \in E \\
& \lambda_{v,w} & \geq & 0 & \forall v, w \in V.
\end{array}$$

At this point, the reader may like to prove that the formulations (F1) and (F2) are equivalent in the following sense: for every optimal solution d, λ of (F1) there exists a feasible solution of (F2) with the same objective value, and vice versa. Here is another LP formulation (F3) that we claim is equivalent to (F2). The reason for studying (F3) is that it has only $O(n^3)$ constraints and $O(n^2)$ variables, i.e., it is a compact LP, in contrast with (F2) (and (F1)) whose number of constraints may be exponential in n. Again, we will leave it to the reader to prove that (F2) and (F3) are equivalent. In the LP (F3), the goal is to find a metric d on V such that the "total weighted capacity" is minimum and the sum over all entries of the metric is at least one.

$$\begin{array}{llllll}
\text{(F3) } z = \text{minimize} & \sum_{e\in E} u_e d_e & & & \\
\text{subject to} & \sum_{v,w\in V} d_{v,w} & \geq & 1 & \\
& d_{v,w} & \leq & d_{v,r} + d_{r,w} & \forall r, v, w \in V \\
& d_{v,r} & \leq & d_{v,w} + d_{w,r} & \forall r, v, w \in V \\
& d_{w,r} & \leq & d_{w,v} + d_{v,r} & \forall r, v, w \in V \\
& d_{v,w} & \geq & 0 & \forall v, w \in V.
\end{array}$$

First, focus on the formulation (F1). Consider the 0-1 incidence vector χ of an arbitrary cut, say, $\delta(S)$ such that both shores ($G[S]$ and $G[V \backslash S]$) are connected. If we take $d = \chi$ and $\lambda_{v,w} = 1$ or 0 depending on whether or not v and w are separated by $\delta(S)$ (i.e., $\lambda_{v,w} = 0$ if either $v, w \in S$ or $v, w \notin S$, otherwise $\lambda_{v,w} = 1$), then it is easily seen that d, λ forms a feasible solution whose

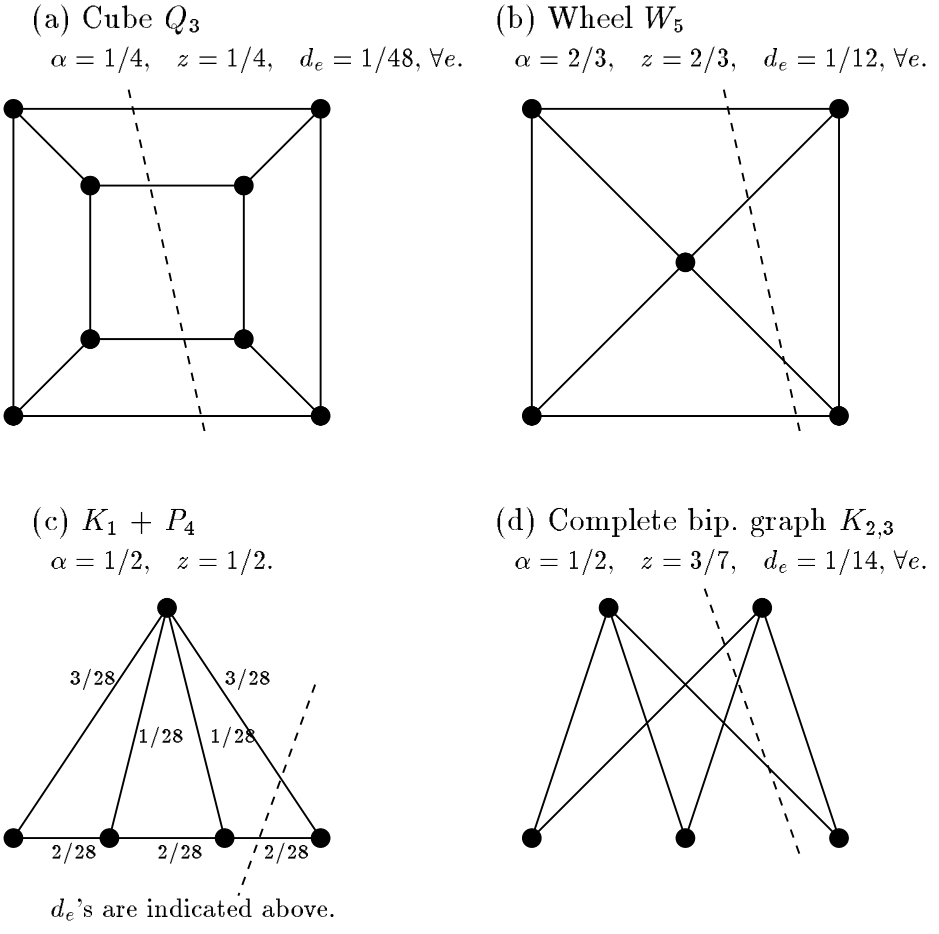

Figure 11.1: Sparsest cuts and optimal solutions to the LP (F2) for four graphs (a) the cube Q_3 (b) the wheel W_5, (c) $K_1 + P_4$, and (d) $K_{2,3}$. For each graph a sparsest cut is indicated by a dashed line. The ratio of the sparsest cut is given as α, the optimal value of LP (F2) is given as z, and the optimal solution of (F2) is given as $d_e, \forall e \in E$.

objective value equals the ratio of the cut $\delta(S)$. Hence, the optimal value of (F1) is a lower bound on the ratio of a sparsest cut.

The big question is: does the optimal value of (F1) equal the ratio of a sparsest cut, and if not, then how much can these values differ? Let us look at a few examples. It is convenient to work with the formulation (F2) or (F3) rather than (F1). Figure 11.1 shows four graphs, the cube Q_3 on 8 nodes, the wheel W_5 on 5 nodes, the graph $K_1 + P_4$ (5-cycle with two adjacent diagonals), and the complete bipartite graph $K_{2,3}$. We take each edge to have unit capacity. The cube Q_3 has sparsest-cut ratio $1/4$ and (F2) optimal value $z = 1/4$. The wheel W_5 has sparsest-cut ratio $2/3$ and (F2) optimal value $z = 2/3$. The graph $K_1 + P_4$ has sparsest-cut ratio $1/2$ and (F2) optimal value $z = 1/2$. The graph $K_{2,3}$ has sparsest-cut ratio $1/2$ and (F2) optimal value $z = 3/7$. (For the three graphs Q_3, W_5 and $K_{2,3}$, the optimal solution has $d_e = z/|E|$ for each edge e, and has $\lambda_{v,w}$ equal to the weight of a shortest v-w path w.r.t. edge weights d.) As a side remark, we mention that $K_{2,3}$ is the smallest graph such that the sparsest-cut ratio differs from the (F2) optimal value. (Q: IS THIS ASSUMING UNIT CAPACITIES ONLY?)

Let us look at another example (due to Leighton & Rao) where the sparsest-cut ratio is $\Omega(\log n)$ times the optimal value of (F2). A bounded-degree expander G is a k-regular graph with $k = O(1)$ (i.e., every node has degree $k = O(1)$) such that for every set of nodes S with $|S| \leq n/2$,

$$u(\delta(S)) \geq c \cdot |S|,$$

where c is an absolute constant (note that every edge has unit capacity). Clearly, the sparsest-cut ratio is $\geq c/n$. Now, consider z, the optimal value of (F2). We claim that there are $n^2/4$ pairs of nodes v, w such that $\lambda'_{v,w} \geq \log_k(n/2) = (\log n/2)/(\log k) \geq c' \log n$, where $\lambda'_{v,w}$ denotes the number of edges in a shortest v-w path of G, and $c' < 1$ is an absolute constant. To see this, just note that for a fixed node w the number of nodes within a radius $1, 2, \ldots, i$ of w is at most $(1+k), (1+k^2), \ldots, k^i$, i.e., $|\{v \mid \lambda'_{v,w} \leq i\}| \leq k^i$, for $i \geq 3$. Now consider the feasible solution to (F2) given by $d_e = (c'(\log n)n^2/4)^{-1} = 4/(c'n^2 \log n)$ for each edge e, and $\lambda_{v,w}$ equal to the weight of a shortest v-w path of G w.r.t. edge weights d. The constraint $\sum_{v,w \in V} \lambda_{v,w} \geq 1$ holds by our claim, and so d, λ is indeed a feasible solution. Since each edge has unit capacity and $|E| = kn/2$, the objective value of this feasible solution is $\sum_{e \in E} u_e d_e = 4|E|/(c'n^2 \log n) = 2k/(c'n \log n) = \Omega(1)/(n \log n)$. Since z, the optimal value of (F2), is at most this value, we are done: the sparsest-cut ratio is at least $\Omega(\log n)$ times z.

The main result in the next section is that the expander example is the worst possible (up to constant factors), because there always exists a cut whose ratio is at most $O(\log n)$ times the optimal value of (F2).

11.2 The sparsest cut problem: A region-growing algorithm

The strategy is to start with an optimal solution $d : E \to \Re_+$ to the LP, and then to run a "region-growing procedure" on the graph with edge weights d to find a cut whose ratio is at most $O(\log n)z$, where z is the optimal value of the LP. The intuition behind the region-growing procedure is to imitate Dijkstra's shortest paths algorithm to construct a moat packing $\{y_S \mid S \subset V\}$. Throughout this section a region means a set of nodes. The procedure keeps growing the current region as long as there is sufficient expansion, and as a result, when the procedure stops, the region is shallow, i.e., it has small radius.

The inputs to the region-growing procedure are the edge weights d, the optimal LP value $z = \sum_{e \in E} u_e d_e$, a set of candidate root nodes $V' \subset V$, and a parameter ϵ (whose role will be clear later). For our application to sparsest cuts, $V' = V$, but other choices of V' are needed in other applications. We start by choosing a root node $r \in V'$. The region-growing procedure maintains a set of nodes A, called the *active set*, and for each such set there is a variable y_A. Initially, the active set is the singleton given by the root, $\{r\}$. One important constraint holds throughout. The y-variables must form a *packing* w.r.t. the edge weights d, i.e., $\sum\limits_{S | e \in \delta(S)} y_S \leq d_e, \forall e \in E$. The weight of r is defined to be $wt(r) = \dfrac{z}{|V'|} = \dfrac{z}{n}$, and the *weight* of an active set A is defined to be $wt(A) = wt(r) + \sum\limits_{S \subset A} y_S u(\delta(S))$. The algorithm increases the variable y_A till the packing constraint becomes tight, i.e., till some edge $e = pq$ with $p \in A$, $q \notin A$, has $d_e = \sum_{S \subset A,\ e \in \delta(S)} y_S$. Then we make $A' = A \cup \{q\}$ the active set and check whether

$$u(\delta(A')) > \epsilon \cdot wt(A'). \tag{11.1}$$

If yes, then we start increasing $y_{A'}$, and continue as above. For a set A, define the radius to be $rad(A) = \sum_{S \subset A} y_S$.

Lemma 11.1 *Let A^* denote the active set at the termination of the region-growing procedure, and w.l.o.g. assume that y_{A^*} is positive. Then*

$$rad(A) < \frac{\ln(1 + |V'|)}{\epsilon} \leq \frac{\ln(1+n)}{\epsilon}.$$

Proof: Let the active sets with positive y-variables be $S_1, S_2, \ldots, S_\ell$ where the sets became active in this sequence. For notational convenience, let $y_1, y_2, \ldots, y_\ell$ denote the y-variables of $S_1, S_2, \ldots, S_\ell$, and let $U_1, U_2, \ldots, U_\ell$ denote the capacities of the cuts $\delta(S_1), \delta(S_2), \ldots, \delta(S_\ell)$. Focus on $wt(S_\ell) = wt(r) + \sum_{i=1}^{\ell} y_i U_i$. Since the y-variables form a packing, we have

$$\sum_{i=1}^{\ell} y_i U_i = \sum_{i=1}^{\ell} y_{S_i} \left(\sum_{e \in \delta(S_i)} u_e \right) = \sum_{e \in E} u_e \left(\sum_{S_i | e \in \delta(S_i)} y_{S_i} \right) \leq \sum_{e \in E} u_e d_e = z.$$

Hence, $wt(S_\ell) \leq (z/n) + z$.

We can derive a lower bound on $wt(S_\ell)$ too. The weight of S_i can be written as

$$wt(S_i) = wt(S_{i-1}) + y_i U_i \geq wt(S_{i-1}) + \epsilon \cdot y_i wt(S_i), \tag{11.2}$$

where we used the fact that $U_i > \epsilon \cdot wt(S_i)$, for $i = 1, 2, \ldots, \ell - 1$, because the algorithm did not terminate with S_i, whereas the last active set $S_\ell = A^*$ has $U_\ell = \epsilon \cdot wt(S_\ell)$. This gives a useful inequality,

$$wt(S_i) \geq \frac{wt(S_{i-1})}{(1 - \epsilon y_i)},$$

where $\epsilon y_i < 1$, as can be seen from inequality (11.2), since $wt(S_{i-1})$ is positive. Hence,

$$wt(S_\ell) \geq \frac{wt(S_0)}{(1-\epsilon y_1)\dots(1-\epsilon y_\ell)}.$$

Combining this inequality and the upper-bound on $wt(S_\ell)$ obtained in the previous paragraph, we have

$$\frac{z}{n}(n+1) \geq \frac{z}{n}\left(\frac{1}{(1-\epsilon y_1)\dots(1-\epsilon y_\ell)}\right).$$

Taking natural logarithms on both sides,

$$\ln(n+1) \geq \sum_{i=1}^{\ell} \ln(1-\epsilon y_i)^{-1} \geq \epsilon \sum_{i=1}^{\ell} y_i,$$

since $\ln(1/(1-y)) \geq y$, for all y with $0 < y < 1$.

The proof is done since $rad(A^*) = \sum_{i=1}^{\ell} y_i \leq \epsilon^{-1}\ln(n+1)$. ∎

Corollary 11.2 *For all pairs of nodes $p, q \in A^*$, $d(r,p) \leq rad(A^*)$ and $d(p,q) \leq 2rad(A^*)$. Here, $d(p,q)$ is the weight of a shortest path between nodes p and q in the graph G w.r.t. the edge weights d.*

Lemma 11.3 *Let $A_1^*, A_2^*, \dots, A_h^*$ ($h \leq |V'| \leq n$) be the regions obtained by iterating the region-growing procedure on the graphs $G_1 = G, G_2 = G_1 - A_1^*, \dots, G_j = G_{j-1} - A_{j-1}^*, \dots, G_h = G_{h-1} - A_{h-1}^*$. That is, A_i^* ($1 \leq i \leq h$) is the active set at the termination of the ith iteration, where the graph is taken to be $G_i = G - A_1^* - \dots - A_{i-1}^*$. Then (i) the total capacity of the union of the coboundaries in G, $\mu = u(\delta(A_1^*) \cup \dots \cup \delta(A_h^*))$ is $\leq 2\epsilon z$. Also, (ii) $\sum_{i=1}^{h} u(\delta(A_i^*)) \leq 4\epsilon z$.*

Proof: Let $\delta_i(S)$ denote the set of edges of G_i that have exactly one end in $S \subseteq V$. Note that $\delta(A_1^*), \delta(A_2^*), \dots, \delta(A_h^*)$ may have edges in common since these are coboundaries in G, but $\delta_1(A_1^*), \delta_2(A_2^*), \dots, \delta_h(A_h^*)$ have no edges in common by definition of $G_1, G_2, \dots, G_h$. Also, every edge of $\delta(A_1^*) \cup \dots \cup \delta(A_h^*)$ is present in $\delta_1(A_1^*) \cup \dots \cup \delta_h(A_h^*)$. By termination of the region-growing procedure, $u(\delta_i(A_i^*)) \leq \epsilon \cdot wt(A_i^*)$, for each $i = 1, 2, \dots, h$, where $wt(A_i^*) = wt(r_i) + \sum_{S \subseteq A_i^*} y_S u(\delta(S))$, and r_i is the root at the ith iteration. Hence,

$$\begin{aligned}
\mu = \sum_{i=1}^{h} u(\delta_i(A_i^*)) \quad &\leq \quad \epsilon \sum_{i=1}^{h} wt(A_i^*) = \\
&\epsilon \left(\sum_{i=1}^{h} \sum_{S \subseteq A_i^*} y_S u(\delta_i(S)) + \sum_{i=1}^{h} wt(r_i) \right) \leq \\
&\epsilon \left(\sum_{e \in E(G)} u_e d_e + \sum_{i=1}^{h} \frac{z}{n} \right) \leq 2\epsilon z.
\end{aligned}$$

In the second last inequality, $\sum_{S\subseteq A_i^*} y_S u(\delta_i(S)) \le \sum_{e\in E(G)} u_e d_e$ holds by the packing constraints.

The last part of the lemma follows from the first part because an edge in two coboundaries $u(\delta(A_i^*))$ and $u(\delta(A_j^*))$, $1 \le i < j \le h$, contributes once to the first part but two times to the second part. ∎

The next theorem is essentially due to Leighton & Rao, but we give a tighter version due to Garg, Vazirani & Yannakakis. The proof due to Garg, Vazirani & Yannakakis is based on the original proof of Leighton & Rao, but is cleaner and simpler than the original proof.

For a cut $\delta(S)$ of G, recall that the ratio means the quantity $\dfrac{u(\delta(S))}{|S|\cdot|V\backslash S|}$.

Theorem 11.4 (Leighton & Rao (1988)) *Let $d : E \to \Re_+$ be the optimal solution to the LP, and let $z = \sum_{e\in E} u_e d_e$ be the optimal value of the LP. Also, let α be the ratio of the sparsest cut of G, $\alpha = \min_{S\subset V} \dfrac{u(\delta(S))}{|S|\cdot|V\backslash S|}$. Then*

$$\frac{\alpha}{8\ln(n+1)+2} \le z \le \alpha.$$

Moreover, there is an efficient algorithm that given the optimal edge weights d finds a cut whose ratio is $\le (8\ln(n+1)+2)z = O(z\log n)$.

Proof: The goal of the proof is to construct a cut with small ratio, by using d and z (which are assumed to be known). We use the region-growing procedure (as in the above lemmas) and choose the parameter ϵ such that one of regions $A_1^*, \ldots, A_h^*$ either has cardinality $\ge n/2$, or gives a cut with small ratio. If one of the regions A_i^* has cardinality $\ge n/2$, then we "erase" the other regions, and keep growing A_i^* node by node till we find a cut with small ratio.

Let $\hat\alpha = (2+8\ln(n+1))z$. We apply the region-growing procedure iteratively, with the parameter ϵ fixed at $\hat\alpha n^2/(8z)$.

Let the regions be $A_1^*, \ldots, A_h^*$. If one of the regions $A_i^* = A$ gives a cut with ratio $u(\delta(A))/|A|\cdot|V\backslash A| \le \hat\alpha$, then we stop and output this cut. Clearly, such a cut has ratio at most $(2+8\ln(n+1)) = O(\log n)$ times the ratio of a sparsest cut. If we do not find such a cut, then we claim that one of the regions $A_1^*, \ldots, A_h^*$ has cardinality $\ge n/2$. The claim is proved by contradiction. By assumption, each of the regions A_i^* has $u(\delta(A_i^*)) > \hat\alpha|A_i^*|\cdot|V\backslash A_i^*| \ge (\hat\alpha n/2)|A_i^*|$. Hence,

$$\sum_{i=1}^h u(\delta(A_i^*)) > (\hat\alpha n/2)\sum_{i=1}^h |A_i^*| = \hat\alpha n^2/2.$$

On the other hand, by Lemma 11.3 part (ii), and by our choice of ϵ,

$$\sum_{i=1}^h u(\delta(A_i^*)) \le 4\epsilon z = \hat\alpha n^2/2.$$

This contradiction proves our claim: one of the regions has cardinality $\ge n/2$.

Let us denote this region by A^*, so $|A^*| \ge n/2$. We continue to grow A^* further, attempting to include all nodes in this set. All y-variables corresponding to sets S, $S \not\subseteq A^*$, are reset to zero.

In this last stage, we use a modified version of the region-growing procedure that does not check for the expansion condition (i.e., the condition in inequality (11.1)), but continues until either the currently active set A gives a cut with small ratio (i.e., $u(\delta(A))/|A| \cdot |V \backslash A| \leq \hat{\alpha}$) or all nodes are included in A. We claim that the algorithm will find a cut with small ratio. Again, the proof is by contradiction.

Let r be the root of the region A^*. Let the sets with positive y-variables be $\{r\} = S_1 \subset \ldots \subset S_\ell = A^* \subset \ldots \subset V$. For every node v, recall that $d(r,v) = \sum_{S \subseteq S_i} y_S$, where $v \in S_{i+1} \backslash S_i$. Focus on

$$\sum_{v \in V} d(r,v) = \sum_{S} y_S(n - |S|) = \sum_{S \subseteq A^*} y_S(n-|S|) + \sum_{S \supset A^*} y_S(n-|S|).$$

The first term satisfies

$$\sum_{S \subseteq A^*} y_S(n-|S|) \leq n \sum_{S \subseteq A^*} y_S = n \cdot rad(A^*) \leq n \ln(n+1)/\epsilon \leq 8\ln(n+1)z/(\hat{\alpha}n).$$

Consider the second term. By our assumption, each superset S of A^* satisfies $u(\delta(S))/|S| \cdot |V \backslash S| > \hat{\alpha}$, and since $|S| > n/2$, we have $|V \backslash S| < 2u(\delta(S))/(\hat{\alpha}n)$. Hence,

$$\sum_{S \supset A^*} y_S(n-|S|) < (2/\hat{\alpha}n) \sum_{S \supset A^*} y_S u(\delta(S)) \leq (2/\hat{\alpha}n)z,$$

where the last inequality follows from the packing constraints on the y-variables. Consequently, $\sum_{v \in V} d(r,v) < (2z/\hat{\alpha}n)(4\ln(n+1)+1)$.

Finally, we apply the inequality

$$\sum_{p,q \in V} d(p,q) \leq (n-1) \sum_{v \in V} d(r,v),$$

where the summation on the left is over *unordered pairs* p, q of nodes; note that there are $\binom{n}{2}$ such pairs. This inequality easily follows from the triangle inequality for triples p, q, r, namely, $d(p,q) \leq d(p,r) + d(r,q)$.

Going back to our proof, we see that

$$\sum_{p,q \in V} d(p,q) \leq n \sum_{v \in V} d(r,v) < (2z/\hat{\alpha})(4\ln(n+1)+1) \leq 1,$$

where the last inequality follows from our choice of $\hat{\alpha}$. This is a contradiction, since the edge weights d form a feasible solution to the LP and so satisfy the LP constraint, $\sum_{p,q \in V} d(p,q) \geq 1$. Our claim follows: one of the sets found by the last stage of the region-growing procedure gives a cut of ratio $\leq \hat{\alpha}$. ∎

11.3 Metrics, ℓ_1 embeddings and a logarithmic guarantee for the generalized sparsest cut problem

Our goal in this section is to give another proof of the Leighton-Rao theorem. In fact, the result here is more general since it applies to the version of the sparsest cut problem where an arbitrary nonnegative demand $dem_{v,w}$ is given for each (unordered) node pair v, w; note that the sparsest cut problem is the special case where $dem_{v,w} = 1, \forall v, w \in V$. Let k be the number of node pairs that have positive demands. Following Aumann & Rabani, and Linial, London & Rabinovich, we will prove that there exists a cut whose ratio is at most $O(\log k)$ times the optimal value of an LP relaxation (F4). The proof is an easy consequence of a theorem of Bourgain on "near isometric embeddings" of metric spaces into ℓ_1.

We start with some definitions and preliminary results. Most readers may prefer to skip the next subsection, and to refer back to the definitions when needed.

11.3.1 Definitions and preliminaries

Let V be a finite set of points, and let n denote $|V|$. A *metric* d on V associates a nonnegative real number $d_{v,w}$ with each pair $v, w \in V$ and satisfies

$$d_{v,w} \leq d_{v,r} + d_{r,w}, \qquad \text{for all ordered triples } v, w, r \in V \times V \times V.$$

We also use $d(v, w)$ to denote $d_{v,w}$. The metric d may be viewed as a vector in $\Re_+^{\binom{n}{2}}$, i.e., d is an $\binom{n}{2}$-dimensional vector with one entry per pair $v, w \in V$. We allow $d_{v,w} = 0$ for distinct points $v, w \in V$, hence, d is a semimetric rather than a metric. A familiar example is the shortest paths metric of a graph, i.e., $d_{v,w}$ is taken to be the minimum number of edges in a v-w path of G.

A *norm* associates a real number $||v||$ with every point $v \in \Re^h$ (the h-dimensional real space) and satisfies

- $||v|| \geq 0$, with equality only if $v = 0$,
- $||rv|| = |r| \cdot ||v||$, for every real number r, and
- $||v + w|| \leq ||v|| + ||w||$, for every $v, w \in \Re^h$.

A norm $|| \cdot ||$ has an associated metric, namely,

$$d_{v,w} = ||v - w||, \forall v, w \in \Re^h.$$

For a point (or vector) $v \in \Re^h$ we denote the coordinates (or entries) of v by $v_1, \ldots, v_h$. For our purposes, three familiar norms (and their associated metrics) are of interest:

- the ℓ_1 norm, $||v||_1 = \sum_{i=1}^{h} |v_i|$, and the associated *Manhattan* metric,
- the ℓ_2 norm, $||v||_2 = \sqrt{\sum_{i=1}^{h} (v_i)^2}$, and the associated *Euclidean* metric, and
- the ℓ_∞ norm, $||v||_\infty = \max_{i=1}^{h} |v_i|$, and the associated ℓ-infinity metric.

We use ℓ_1^h, ℓ_2^h, or ℓ_∞^h to denote $\Re^h$ equipped with the ℓ_1 norm, the ℓ_2 norm or the ℓ_∞ norm, respectively.

An *isometric embedding* (or isometry) is a mapping ϕ from a metric space (V, d) to a metric space (X, d') that preserves all "distances", i.e., $\phi : V \to X$ is an isometry if

$$d'(\phi(v), \phi(w)) = d(v, w), \quad \forall v, w \in V.$$

Also, V is said to isometrically embed into X.

Proposition 11.5 *Every metric space (V, d) has an isometric embedding into ℓ_∞^n, i.e., into $\Re^n$ equipped with the ℓ_∞ norm, where n denotes $|V|$.*

Proof: Let $V = \{v^{(1)}, \ldots, v^{(n)}\}$. We map each $v^{(i)}$ to the vector $\phi(v^{(i)}) \in \Re^n$, where

$$\phi(v^{(i)}) = [d(v^{(1)}, v^{(i)}), d(v^{(2)}, v^{(i)}), \ldots, d(v^{(n)}, v^{(i)})].$$

In other words, $v^{(i)}$ is mapped to the ith column of the $n \times n$ "distance" matrix $[d_{v,w} \mid v, w \in V \times V]$. Then observe that

$$\|\phi(v^{(i)}) - \phi(v^{(j)})\|_\infty = \max_{k=1}^{n} |d(v^{(k)}, v^{(i)}) - d(v^{(k)}, v^{(j)})| \geq d(v^{(i)}, v^{(j)}).$$

On the other hand, the triangle inequality gives

$$|d(v^{(k)}, v^{(i)}) - d(v^{(k)}, v^{(j)})| \leq d(v^{(i)}, v^{(j)}).$$

The proof is done since $\|\phi(v^{(i)}) - \phi(v^{(j)})\|_\infty = d(v^{(i)}, v^{(j)})$. ∎

Proposition 11.6 *Let (V, d_1) be the metric space ℓ_1^h for some integer $h > 0$, i.e., d_1 is the metric induced by ℓ_1 for the set of points V in $\Re^h$. Then d_1 is the sum of h metrics $d_{1,1}, \ldots, d_{1,h}$, where $d_{1,i}$ is a metric induced by ℓ_1 in one dimension.*

Proof: For each i, $i = 1, \ldots, h$, define $d_{1,i} = |v_i - w_i|$, i.e., $d_{1,i}$ gives the "distances" in the ith coordinate. The result is obvious, since for each pair $v, w \in V$,

$$d_1(v, w) = \|v - w\|_1 = \sum_{i=1}^{h} |v_i - w_i| = d_{1,1}(v, w) + d_{1,2}(v, w) + \ldots + d_{1,h}(v, w).$$

∎

11.3.2 ℓ_1 embeddings and generalized sparsest cuts

Recall that the sparsest cut problem on a graph $G = (V, E)$ with edge capacities $u : E \to \Re_+$ is to find a cut $\delta(S)$ whose ratio $u(\delta(S))/|S| \cdot |V \backslash S|$ is minimum. We will also use u to denote the $\binom{n}{2}$-dimensional vector indexed by node pairs $v, w \in V$ such that $u_{v,w}$ equals either the capacity of the edge vw (if it exists) or zero (if there is no edge vw). Let $\mathbf{1}$ denote the $\binom{n}{2}$-dimensional vector of all ones.

Based on Proposition 11.5, we may view the LP formulation (F3) of the sparsest cut problem as follows: the optimal solution finds an embedding $\phi : V \to \ell_\infty^n$ (i.e., the nodes are embedded into $\Re^n$ equipped with the ℓ_∞ norm) such that (i) the sum of the $\binom{n}{2}$ "distances" in ℓ_∞^n is at least one (i.e., $\sum\limits_{v,w \in V} \|\phi(v) - \phi(w)\|_\infty \geq 1$), and (ii) the sum of the capacities of the edges weighted by the edge "lengths" is minimum. What (if any) is the relation of such an embedding to the sparsest cut in the graph? There seems to be no direct relation, but there is a relation via an embedding into ℓ_1^n. To see this, suppose that we can somehow find an embedding $\phi' : V \to \ell_1^n$ such that (i) and (ii) hold w.r.t. "distances" and edge "lengths" in ℓ_1^n, i.e., such that $\sum_{v,w \in V} \|\phi'(v) - \phi'(w)\|_1 \geq 1$ and $z' = \sum_{e=vw \in E} u_{vw} \cdot \|\phi'(v) - \phi'(w)\|_1$ is minimum. There is something remarkable about such an ℓ_1 embedding because it turns out that z' equals the ratio of a sparsest cut. The proof is given below, in a few easy steps. (Since computing the ratio of a sparsest cut is NP-hard, there is no hope of computing ϕ' efficiently, modulo P $\neq$ NP. Nevertheless, it will be useful to study ϕ'.)

For a subset S of a point set V, the *cut metric* d_S is defined to be the metric on V given by

$$d_S(v, w) = \begin{cases} 1 & \text{if either } v \in S, w \notin S \text{ or } v \notin S, w \in S \\ 0 & \text{otherwise.} \end{cases}$$

In other words, the cut metric is the 0-1 incidence vector of the point pairs separated by the cut. Observe two things. For a cut $\delta(S)$ of the graph G, the capacity $u(\delta(S))$ may be written as the inner product of two $\binom{n}{2}$-dimensional vectors, $u \cdot d_S$, and secondly, the ratio of the cut may be written as $u \cdot d_S / (\mathbf{1} \cdot d_S)$.

Proposition 11.7 *Let (V, d_1) be a metric space such that d_1 is induced by ℓ_1, i.e., $d_1(v, w) = \|v - w\|_1, \forall v, w \in V$. Then d_1 is in the cone generated by the cut metrics $\{d_S \mid S \subset V\}$, i.e., d_1 can be written as a linear combination of cut metrics*

$$d_1 = \sum_j \{a_j d_{S_j} \mid S_j \subset V\}$$

such that each coefficient a_j is nonnegative. Moreover, the number of cut metrics required is at most $h \cdot |V|$, where h is the dimension of (V, d_1).

Proof: Since d_1 is the sum of h "one-dimensional ℓ_1 metrics" (by Proposition 11.6), it will suffice to prove that a "one-dimensional ℓ_1 metric" d_1 can be written as a conical combination (i.e., a linear combination with nonnegative coefficients) of at most $|V|$ cut metrics. Observe that (V, d_1) is an embedding of V into the real line $\Re^1$, because d_1 is one-dimensional. Let the points in V occur in the order $v^{(1)}, \ldots, v^{(i)}, \ldots, v^{(n)}$ in the real line. Define n cut metrics d_{S_j} by taking $S_1 = \{v^{(1)}\}, \ldots, S_i = \{v^{(1)}, \ldots, v^{(i)}\}, S_n = V$. Clearly, $d_1 = \sum_{j=1}^{n-1} a_j d_{S_j}$, where $a_j = d_1(v^{(j)}, v^{(j+1)})$. ∎

Our claim (z' equals the ratio of a sparsest cut) follows from Proposition 11.7 and a simple fact about ratios.

Fact 11.8 *Let $x_1, \ldots, x_k$ be nonnegative numbers and let $y_1, \ldots, y_k$ be positive numbers. Then*

$$\min_{i=1}^{k} \frac{x_i}{y_i} \leq \frac{x_1 + x_2 + \ldots + x_k}{y_1 + y_2 + \ldots + y_k} \leq \max_{i=1}^{k} \frac{x_i}{y_i}.$$

In the next proposition, rather than using the linear objective function of the LP (F3) we use the rational objective function of (F1).

Proposition 11.9 *Let $\delta(S^*)$ be a sparsest cut of the (capacitated) graph $G = (V, E), u$, and let α^* be the ratio of this cut. Let h be a positive integer. Then*

$$\alpha^* = z' = \min_{\phi: V \to \Re^h} \frac{\sum_{e=vw \in E} u_{vw} \cdot \|\phi(v) - \phi(w)\|_1}{\sum_{v,w \in V} \|\phi(v) - \phi(w)\|_1}.$$

Proof: Let $(\phi(V), d_1)$ be the metric space that minimizes z'. First, note that $z' \leq \alpha^*$ since we can take d_1 to be the cut metric d_{S^*} of the sparsest cut (i.e., we take $\phi(v) = 1$ if $v \in S^*$ and $\phi(v) = 0$ otherwise). To see that $z' \geq \alpha^*$, use Proposition 11.7 to write d_1 as $\sum_{j=1}^{p} a_j d_{S_j}$, where each S_j corresponds to a cut $\delta(S_j)$, and $p \leq hn$. Then

$$z' = \frac{\sum_{j=1}^{p} u \cdot (a_j d_{S_j})}{\sum_{j=1}^{p} \mathbf{1} \cdot (a_j d_{S_j})} \geq \min_{j=1}^{p} \frac{u \cdot (a_j d_{S_j})}{\mathbf{1} \cdot (a_j d_{S_j})} = \min_{j=1}^{p} \frac{u(\delta(S_j))}{|S_j| \cdot |V \backslash S_j|} \geq \alpha*,$$

where the first inequality follows from the previous fact. The proof is done. ∎

The optimal solution of (F1) (or of the LP (F3)) is an ℓ_∞-induced metric minimizing our objective function, whereas a sparsest cut corresponds to an ℓ_1-induced metric minimizing our objective function. To efficiently find an approximately sparsest cut via the optimal solution to the LP, we can try to embed the optimal ℓ_∞-induced metric into ℓ_1 such that all pairwise "distances" are preserved as much as possible. This motivates the next definition. Note that both z (the optimal value of (F1)) and z' (in Proposition 11.9) remain unchanged if all the pairwise "distances" are scaled by the same quantity.

Let (V, d) be a metric space and let ϕ be an embedding of (V, d) into (V', d') such that $d'(\phi(v), \phi(w)) \leq d(v, w), \forall v, w \in V$. The *distortion* of ϕ is defined to be

$$\max_{v,w \in V} \frac{d(v, w)}{d'(\phi(v), \phi(w))}.$$

Another proof of the Leighton-Rao theorem follows from an important theorem of Bourgain and Proposition 11.9. (A proof of Bourgain's theorem is given in the next section.)

Theorem 11.10 (Bourgain (1986)) *Let (V, d) be a metric space, and let n denote $|V|$. There exists an embedding ϕ from (V, d) into ℓ_1^h, where $h = O(\log^2 n)$, such that the distortion is $O(\log n)$. Moreover, ϕ can be computed in polynomial time by a randomized algorithm.*

Theorem 11.11 (Leighton & Rao (1988)) *Let d be the optimal solution to the LP (F3), and let z be the optimal value of the LP (F3). Let α be the ratio of a sparsest cut of G, u. Then*

$$\frac{\alpha}{O(\log n)} \leq z \leq \alpha.$$

Moreover, there is an efficient (deterministic) algorithm that finds a cut whose ratio is $O(z \log n)$.

Proof: After computing an optimal solution d for the LP (F3), we use Bourgain's theorem to compute an embedding ϕ of (V, d) into ℓ_1^h, where $h = O(\log^2 n)$, such that the distortion Δ is $O(\log n)$. Let d_1 denote the ℓ_1-induced metric. Then,

$$z' = \frac{u \cdot d_1}{\mathbf{1} \cdot d_1} \leq \Delta \cdot \frac{u \cdot d}{\mathbf{1} \cdot d} = \Delta z,$$

where the inequality follows because $d(v, w)/\Delta \leq d_1(v, w) \leq d(v, w), \forall v, w \in V$. By Proposition 11.9, we can find a cut $\delta(S)$ whose ratio is $\leq z'$ by examining the ratios of at most $h \cdot |V|$ cuts. This completes the proof. A randomized algorithm is easily obtained because the construction in the proof of Bourgain's theorem is randomized. Linial, London & Rabinovich (1995) (and independently Garg (1995)) have given a method for derandomizing the Bourgain construction. ∎

In fact, a generalization of the Leighton-Rao theorem can be proved by the method of ℓ_1 embeddings. For each pair of nodes $v, w \in V$ of the (capacitated) graph G, u, let there be a nonnegative real-valued demand $dem_{v,w}$. Also, let dem denote the corresponding $\binom{n}{2}$-dimensional vector indexed by node pairs $v, w \in V$. Define the demand of a cut $\delta(S)$ to be

$$dem(S) = \sum_{v,w \in V} \{dem_{v,w} \mid v \in S, w \notin S \text{ or } v \notin S, w \in S\} = dem \cdot d_S.$$

Now, redefine the *ratio* of a cut $\delta(S)$ to mean

$$u(\delta(S))/dem(S) = u \cdot d_S/(dem \cdot d_S),$$

and define a *generalized sparsest cut* to be a cut whose ratio is minimum. By changing one constraint in the LP (F3), we get the following LP relaxation of the generalized sparsest cut problem.

$$\begin{array}{llll} \text{(F4)} \quad z = \text{minimize} & \sum_{e \in E} u_e d_e & & \\ \text{subject to} & \sum_{v,w \in V} dem_{v,w} d_{v,w} & \geq & 1 \\ & d \text{ is a metric.} & & \end{array}$$

Theorem 11.12 (Aumann & Rabani (1994), Linial, London & Rabinovich (1995))
Let $z = \sum_{e \in E} u_e d_e$ be the optimal value of the LP (F4). Let α be the ratio of a generalized sparsest cut of G, u, dem, and let k denote the number of pairs $v, w \in V$ with positive demands. Then

$$\frac{\alpha}{O(\log k)} \leq z \leq \alpha.$$

Moreover, there is an efficient algorithm that finds a cut whose ratio is $O(z \log k)$.

We leave the proof to the reader, since it is similar to the proof of Theorem 11.11 given above. To obtain an approximation guarantee of $O(\log k)$ rather than $O(\log n)$ we need a small generalization of Bourgain's theorem.

Proposition 11.13 *Let (V, d) be a metric space, and let T be a subset of V. There exists an embedding ϕ from (V, d) into ℓ_1^h, where $h = O(\log^2 |T|)$, such that for each $v, w \in V$, $\|\phi(v) - \phi(w)\|_1 \leq d(v, w)$, and for each $v, w \in T$, $d(v, w)/\Delta \leq \|\phi(v) - \phi(w)\|_1$, where $\Delta = O(\log |T|)$. Moreover, ϕ can be computed in polynomial time by a randomized algorithm.*

11.4 Bourgain's theorem

We give a proof of Bourgain's theorem, omitting some computational details. The reader may either fill in the details (by solving Exercise 8), or refer to Linial, London & Rabinovich (Combinatorica 1995) or to the survey paper by Shmoys (1996).

Theorem 11.10 *Let (V, d) be a metric space, and let n denote $|V|$. There exists an embedding ϕ from (V, d) into ℓ_1^h, where $h = O(\log^2 n)$, such that the distortion is $O(\log n)$. Moreover, ϕ can be computed in polynomial time by a randomized algorithm.*

Proof: The mapping $\phi : V \to \Re^h$ is constructed using randomization. For a point $v \in V$ and a subset $S \subseteq V$, let $d(v, S) = \min_{w \in S} d(v, w)$. Let $p = \lfloor \log_2 n \rfloor$, so that $2^p \leq n < 2^{p+1}$, and let $q = 10p$; in fact, we will have $h = pq$. For each $j = 0, 1, 2, \ldots, p-1$, we randomly and independently choose q subsets of V, each of cardinality $n/2^j$, $A_{j,1}, A_{j,2}, \ldots, A_{j,q}$, i.e., each of the $\binom{n}{n/2^j}$ subsets of V of cardinality $n/2^j$ is equally likely to be chosen as a set $A_{j,k}$, and the choices are mutually independent. Next, each point $v \in V$ is mapped to a vector

$$\begin{aligned}\phi'(v) \quad = \quad & [d(v, A_{0,1}), d(v, A_{0,2}), \ldots, d(v, A_{0,q}), d(v, A_{1,1}), d(v, A_{1,2}), \ldots, d(v, A_{1,q}), \\ & \ldots, d(v, A_{p-1,1}), d(v, A_{p-1,2}), \ldots, d(v, A_{p-1,q})],\end{aligned}$$

i.e., v is mapped to a vector of dimension pq whose ith coordinate is the "d-distance" between v and the ith random set in the list $A_{0,1}, \ldots, A_{0,q}, \ldots, A_{p-1,1}, \ldots, A_{p-1,q}$. Let the ith coordinate of the vector $\phi'(v)$ be denoted by $\phi'_i(v)$. We obtain the final mapping ϕ by scaling each vector $\phi'(v)$ by $1/(pq)$, i.e., $\phi(v) = \phi'(v)/(pq), \forall v \in V$.

To see that $\|\phi(v) - \phi(w)\|_1 \leq d(v, w)$, focus on an arbitrary coordinate $i = jq+k$ $(0 \leq j \leq p-1$, $1 \leq k \leq q)$ and let A be the corresponding random set $A_{j,k}$. We have $|\phi'_i(v) - \phi'_i(w)| = |d(v, A) - d(w, A)| \leq d(v, w)$, because w.l.o.g. there exist $a \in A$ and $b \in A$ such that $|d(v, A) - d(w, A)| = d(v, a) - d(w, b) \leq d(v, b) - d(w, b) \leq d(v, w)$, where the last step follows by the triangle inequality on d. Hence, $\|\phi(v) - \phi(w)\|_1 = \dfrac{1}{(pq)} \sum_{i=1}^{pq} |\phi'_i(v) - \phi'_i(w)| \leq d(v, w)$.

The ingenious part of the proof is to show that for all pairs $v, w \in V$, the other inequality $(\|\phi(v) - \phi(w)\|_1 \geq d(v, w)/\Delta)$ holds with high probability. Consider an arbitrary pair $v, w \in V$. Here is an informal argument that skips a few details; these details are handled either in the next paragraph or in the exercises. We will partition the "line segment" joining v and w into $O(\log n)$ segments such that the sum of the "d-lengths" of the first p segments is at least $d(v, w)/3$. See Figure 11.2. Let ρ_j be the "d-length" of the jth segment. Consider an arbitrary coordinate $i = jq + k$ $(0 \leq j \leq p-1, 1 \leq k \leq q)$ of $\phi(v)$ and $\phi(w)$. We need a key claim:

there exists a constant $c < 1$ such that with probability $\geq c$, $|\phi'_i(v) - \phi'_i(w)| \geq \rho_j$.

If we fix j and sum over all coordinates $i = jq + k$, for $k = 1, 2, \ldots, q$, then by applying Chernoff bounds it can be seen that with high probability, $\sum_{k=1}^{q} |\phi'_{jq+k}(v) - \phi'_{jq+k}(w)| \geq qc\rho_j/2$ (see Exercise 8 for details). Finally, summing over all coordinates $i = 1, 2, \ldots, (pq)$, we see that with high

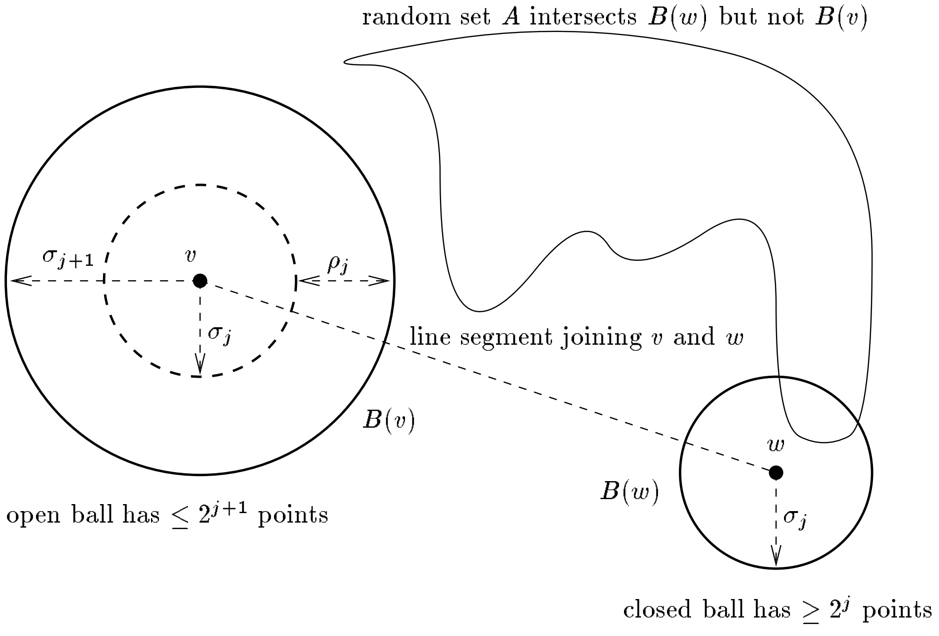

Figure 11.2: An illustration of the proof of Bourgain's theorem.

probability,

$$\sum_{i=1}^{pq} |\phi'_i(v) - \phi'_i(w)| = \sum_{j=0}^{p-1} \left(\sum_{k=1}^{q} |\phi'_{jq+k}(v) - \phi'_{jq+k}(w)| \right) \geq \frac{qc}{2} \sum_{j=0}^{p-1} \rho_j \geq \frac{qc}{2} \frac{d(v,w)}{3}.$$

This proves the theorem, since $\sum_{i=1}^{pq} |\phi_i(v) - \phi_i(w)| \geq \frac{c}{6p} d(v,w)$, and so the distortion is $\Delta = 6p/c = O(\log n)$.

The claim above needs to be proved in detail. For $j = 0, 1, \ldots, p-1$, let σ_j be the minimum of $d(v,w)/3$ and the minimum real number σ such that both $|\{a \in V \mid d(v,a) \leq \sigma\}| \geq 2^j$ and $|\{b \in V \mid d(w,b) \leq \sigma\}| \geq 2^j$, i.e., we compute the minimum radius σ such that each of the balls of d-radius σ centered at v and w, respectively, contains at least 2^j points of V, and further, if $\sigma \leq d(v,w)/3$, then we take $\sigma_j = \sigma$, otherwise we take $\sigma_j = d(v,w)/3$. Note that $\sigma_0 = 0$. See Figure 11.2. Consider j $(0 \leq j \leq p-2)$ such that $\sigma_j < d(v,w)/3$. (The other case $\sigma_j = d(v,w)/3$ is trivial.) W.l.o.g. the set of points of V in the open ball of d-radius σ_{j+1} centered at v, $B^o(v, \sigma_{j+1})$, has cardinality $\leq 2^{j+1}$. Moreover, the set of points of V in the closed ball of d-radius σ_j centered at w, $B(w, \sigma_j)$, has cardinality $\geq 2^j$. Now, the random set $A = A_{j,k}$ (for some k, $1 \leq k \leq q$) has $n/2^j$ points of V. It is an exercise in elementary probability to show that there exists a constant $c < 1$ such that with probability $\geq c$, $A \cap B^o(v, \sigma_{j+1})$ is empty and $A \cap B(w, \sigma_j)$ is nonempty. Hence, with probability $\geq c$, $d(v, A) \geq \sigma_{j+1}$ and $d(w, A) \leq \sigma_j$. Consequently, with probability $\geq c$,

$$|\phi'_{jq+k}(v) - \phi'_{jq+k}(w)| = |d(v, A_{j,k}) - d(w, A_{j,k})| \geq \sigma_{j+1} - \sigma_j.$$

Note that this analysis applies also to the largest j such that $\sigma_j < d(v,w)/3$. Also, note that in terms of the previous paragraph, the jth segment (in the "line segment" joining v and w) has "d-length" $\rho_j = \sigma_{j+1} - \sigma_j$. ∎

11.5 From sparsest cuts to balanced separators

In this section, we sketch how the result on approximating the sparsest cut can be applied to find balanced separators in graphs. An α-balanced separator (for $\alpha \geq \frac{1}{2}$) is defined as a set of edges whose removal breaks up the graph into two components each of size at least $\alpha \cdot n$. A $\frac{1}{2}$-balanced separator is also sometimes called a bisector. By a balanced separator, we mean any α-balanced separator for some fixed constant α. This is because in many applications, what is important is that separator is balanced rather than the exact quality of the balance.

Define the *flux* of a cut $\delta(S)$ of G to mean the quantity $\dfrac{u(\delta(S))}{\min(|S|, |V \backslash S|)}$[1]. A *minimum flux cut* is one whose flux is minimum.

Flux and sparsity are very closely related quantities. Suppose that the sparsest cut has ratio s and the minimum flux cut has flux f, then

$$\frac{n}{2}s \leq f \leq ns$$

since the bigger half of any cut has size between $\frac{n}{2}$ and n. Thus the approximation algorithm from the previous section also gives an $O(\log n)$ approximation for finding a minimum flux cut in a given graph. We can use this result to derive an approximation result of the following type.

Theorem 11.14 *Suppose that we are given an ρ-approximation algorithm for finding a minimum flux cut in a graph, and suppose that B denotes the minimum capacity of any bisector (i.e., $\frac{1}{2}$-balanced separator) in the graph. Then we can find a $\frac{2}{3}$-balanced separator of the graph of capacity at most $8\rho \cdot B$.*

Proof: We prove the theorem by using a simple greedy algorithm to construct the $\frac{2}{3}$-balanced cut. The algorithm is simple. We iterate finding the minimum flux cut in a graph and deleting the vertices in the smaller side of the cut, until the number of vertices that remain is at most $\frac{2n}{3}$. The set of all the deleted vertices form one shore of the cut while the remaining vertices form the other.

Let us denote by u_i and x_i, the capacity and the size of the smaller side of the flux cut found in iteration number i. Say the greedy procedure ran for t iterations; we wish to bound $\sum_{i=1}^{t} u_i$ in terms of B.

The key step is to observe that while we are still running the greedy procedure, the minimum bisector induced in the remaining graph gives a cut of small flux. In particular, since we have deleted at most $\frac{n}{3}$ nodes, in the worst case, all these nodes may be deleted from the same side of the minimum bisector, and this cut in the remaining graph has flux at most $\frac{B}{\frac{n}{6}}$. This is because

[1] The flux has also been called the edge expansion or the quotient cost of a cut.

the smaller side of this cut has at least $\frac{n}{2} - \frac{n}{3}$ nodes in it. Using a ρ-approximation algorithm at every iteration guarantees that

$$\frac{u_i}{x_i} \leq \rho \cdot \frac{6B}{n} \tag{11.3}$$

Next we write down a recurrence for the number of nodes we "need" to delete to achieve our goal. Namely, let n_i represent the number of nodes before iteration i that we need to delete so that the remaining graph has at most $\frac{2n}{3}$ nodes. Thus, e.g., $n_1 = \frac{n}{3}$, and $n_t \geq 1$. We get the following simple recurrence.

$$n_{i+1} = n_i - x_i$$

By equation 11.3, we have $x_i \geq \frac{nu_i}{6\rho B} \geq \frac{n_i u_i}{2\rho B}$. Substituting and simplifying, we get

$$n_{i+1} \leq n_i(1 - \frac{u_i}{2\rho B}) \leq n_i e^{-\frac{u_i}{2\rho B}}$$

Expanding out the recurrence, we get

$$n_t \leq n_1 e^{-\sum_{i=1}^{t-1} \frac{u_i}{2\rho B}}$$

Taking natural logarithm and simplifying, we finally get

$$\sum_{i=1}^{t-1} u_i \leq 2\rho B \log \frac{n_i}{n_t} \leq 2\rho B \log n \tag{11.4}$$

To bound the capacity of the last cut u_t, we use equation 11.3 again to get

$$u_t \leq \rho \cdot x_t \frac{6\rho B}{n} \leq 3\rho B$$

since $x_t \leq \frac{n}{2}$ (we always choose the smaller side of a cut). The two equations above imply that the cost of the $\frac{2}{3}$-balanced separator found is $O(B \log n)$ where B is the capacity of a minimum bisector. ∎

11.6 Applications of separators

Finding balanced separators has several applications in the design of good approximation algorithms for other problems whose objective can be realted to the value of a bisector – these approaches typically employ divide-and-conquer using the separator approximation in the divide part of the procedure. This approach was pioneered in the work of Bhatt and Leighton [4] and applied to a variety of problems by Leighton and Rao [9]. Examples include finding minimum area layout of a graph, finding layouts that minimize the total number of edge crossings, and finding a minimum feedback arc set (arcs whose removal leaves the remaining digraph acyclic). Other more involved applications are in graph completion problems to find the minimum-size supergraph of a given graph that has certain nice properties, e.g., is an interval graph[12] or a chordal graph[2]; The latter problem models finding good orderings for sparse Gaussian elimination.

Two overall features are useful in identifying problems that may be amenable to the divide-and-conquer approximation approach using separators. Firstly, the objective of the problem must be related to the size of the separator, and secondly, an optimal solution for the overall problem should be "decomposable" into solutions to the subproblems that are obtained in the dividing stage. We illustrate these two features by looking at two simple linear arrangement problems.

Linear arrangements

A *linear arrangement* of an undirected graph $G = (V, E)$ (where $|V| = n$) is a linear ordering of its nodes, or more formally, a bijection A of the vertices to the set $\{1, 2, \ldots, n\}$. Pictorially, we can think of a linear arrangement as a drawing of the nodes of a graph in a line in the order specified by the bijection A.

A linear arrangement defines $n-1$ sequential cuts formed by the sets $\{A^{-1}(1), A^{-1}(2), \ldots, A^{-1}(i)\} = S_i$ for each i. Furthermore, every edge uv with $A(v) > A(u)$ say, is "stretched" in the arrangement to an extent $stretch(uv) = A(v) - A(u)$. Four problems arise by minimizing the maximum or the sum of each of the two quantities: the stretch of edges and the size of each of the sequential cuts.

Finding a linear arrangement to minimize the maximum size of any of the sequential cuts corresponds to solving the *minimum cut linear arrangement (MCLA) problem.* Minimizing the maximum stretch of any edge in the arrangement corresponds to finding a *minimum bandwidth ordering.* A moment of though reveals that the two objectives involving minimizing the sum of the stretches and the cuts are really the same (by switching the summation between the edges and the sequential cuts). This ordering problem is to find an *optimal linear arrangement (OLA).* We will use our separator-based method for the minimum cut and optimal linear arrangement problems. No nontrivial approximation algorithm is known for the bandwidth minimization problem on general graphs.

11.6.1 Minimum cut linear arrangement

To discover the relation of the size of a balanced separator to the value of a cut in a linear arrangement, consider an optimal MCLA. Let OPT denote the maximum size of a cut in this arrangement. Since the cut corresponding to the set $S_{\frac{n}{2}}$ is also a candidate for computing this maximum value, and this cut is a bisector, we have that $OPT \geq B$ where B is the size of a minimum bisector. On the other hand, we have a separator approximation that finds a balanced separator of size $O(\log n \cdot B)$.

This motivates the following simple divide and conquer algorithm: use the approximation algorithm for finding an $\frac{2}{3}$-balanced separator to break the graph into two parts, say L and R. Recursively compute an arrangement of the nodes of L and the nodes of R and concatenate them to form the final arrangement. The basis is when the graph has only one node in which case the ordering is trivial. Alternatively, the divide steps of this algorithm can be represented by a "separator tree" whose internal nodes represent a divide step that finds a balanced separator, and the leaves are the original nodes of G. The arrangement output is the preorder traversal of these nodes, or simply the left to right ordering of the leaves.

It is not hard to bound the performance ratio of this algorithm. First we notice that the depth of recursion, or alternatively, the depth of the separator tree, is $O(\log n)$ since we use a banaced separator in each divide step. The maximum value of any of the cuts in the output arrangement is the maximum over the sum of the values of $O(\log n)$ different separators found by following some recursive step in the algorithm. We already argued that the top-level separator has size $O(\log n \cdot B) = O(\log n \cdot OPT)$. If we can argue that each of the separators found in any recursive step is also of size $O(\log n \cdot OPT)$, a performance ratio of $O(\log^2 n)$ would follow.

The decomposition property is useful in showing that for any recursive subproblem, the size of the separator found is $O(\log n \cdot OPT)$. Consider a fixed depth of recursion corresponding to a particular level of the separator tree. Let the subgraphs corresponding to the different portions of the original graph on which we are finding separators be $G_1, \ldots G_k$. Note that the nodes of these subgraphs partition those of G. Consider the ordering induced on the nodes of G_i by the MCLA. The crucial observation is that the maximum cut in this induced arrangement has at most OPT edges. Moreover by looking at the central cut in this arrangement that splits G_i into two equal pieces, we see that the minimum size of a bisector of G_i is at most the size of the central cut. This in turn is at most OPT. The separator we find for G_i has size at most a logarithmic factor larger than the size of a minimum bisector for G_i, and hence at most $O(\log n \cdot OPT)$. This completes the proof that the divide-and-conquer approach gives an $O(\log^2 n)$ approximation algorithm.

11.6.2 Optimal linear arrangement

Before we present an algorithm for this problem we first state a simple extension of the previous result on finding balanced separators. Earlier, we argued that we can find a $\frac{2}{3}$-balanced separator by applying a greedy algorithm and using a minimum bisector to bound the quality of the solution. We can instead attempt to find say a $\frac{3}{4}$-balanced separator by using a minimum $\frac{2}{3}$-balanced separator in the analysis. This gives the following result.

Theorem 11.15 *Suppose that we are given an ρ-approximation algorithm for finding a minimum flux cut in a graph, and suppose that $B_{\frac{2}{3}}$ denotes the minimum capacity of any $\frac{2}{3}$-balanced separator in the graph. Then we can find a $\frac{3}{4}$-balanced separator of the graph of capacity at most $O(\rho \cdot B_{\frac{2}{3}})$.*

We can proceed as before by first trying to relate the value OPT of the optimal linear arrangement to the size of a separator. To do this, note that OPT is a sum of $n-1$ terms corresponding to the sizes of the sequential cuts in the linear order. If we consider the middle third of these cuts ($\delta(S_i)$ for i from $\frac{n}{3}$ to $\frac{2n}{3}$), each of these cuts define a $\frac{2}{3}$-balanced cut of the graph and must have size at that of a minimum such separator, say $B_{\frac{2}{3}}$. Thus we have

$$OPT \geq \frac{n \cdot B_{\frac{2}{3}}}{3}.$$

As before, we use a simple divide and conquer algorithm: apply the approximation algorithm for finding an $\frac{3}{4}$-balanced separator in the above theorem to break the graph into two parts, say L and R. Recursively compute an arrangement of the nodes of L and the nodes of R and concatenate them to form the final arrangement. Again, the ordering can be represented by the left to right ordering of the leaves of the corresponding separator tree.

We use a slightly different argument to show a performance ratio of $O(\log^2 n)$ in this case. At the top level of recursion, we split the graph by using edges of a $\frac{3}{4}$-balanced separator of size at most $O(\log n \cdot B_{\frac{2}{3}})$. The contribution of these edges to the value of this arrangement is the sum of their stretches which is trivially at most $n-1$. Thus the overall contribution of this separator to the objective is $O(n \log n \cdot B_{\frac{2}{3}}) = O(\log n \cdot OPT)$.

To employ a similar strategy at every level of recursion, we must show that for all the different disjoint graphs formed at any depth in the recursion, the contributions of all the separator edges found in this level is $O(\log n \cdot OPT)$. Note that these subgraphs represent a partition of the nodes of G and there are at most $(\frac{4}{3})^i$ of them at depth i of recursion.

We use a decomposition property to prove the claim for a general level of the recursion: consider the linear arrangement induced on each of the subgraphs in this level by the optimal global arrangement. Since the stretch of any edge in an induced arrangement is at most that in the global arrangement, the sum of the values of the induced arrangements, say $OPT_1+OPT_2+\ldots+OPT_k$ is at most OPT. By our previous argument applied to each of the subgraphs, we see that $OPT_i \geq \frac{n_i \cdot B^i_{\frac{2}{3}}}{3}$ where n_i is the number of nodes in the subgraph and $B^i_{\frac{2}{3}}$ is the value of a $\frac{2}{3}$-balanced separator in that subgraph G_i. The number of edges in the separator found for this graph is $O(\log n_i \cdot B^i_{\frac{2}{3}})$ and each has a stretch of at most $n_i - 1$ trivially. Thus the contribution by these edges is $O(n_i \cdot \log n_i \cdot B^i_{\frac{2}{3}}) = O(\log n_i \cdot OPT_i) = O(\log n \cdot OPT_i)$. Thus the total contribution of all the edges in a level is $O(\log n \cdot (OPT_1 + OPT_2 + \ldots + OPT_k)) = O(\log n \cdot OPT)$ as required. This shows that the overall method has performance ratio $O(\log^2 n)$ since there are $O(\log n)$ levels of recursion.

For several of the problems described in [12], the log-squared approximation guarantees were slightly improved to $O(\log n \log\log n)$ by Even et al. [6]. More recently, Rao and Richa [11] further improve some of these results to $O(\log n)$-approximations, including the optimal linear arrangement problem described above.

11.7 Exercises

1. Given a graph $G=(V,E)$ with edge capacities $u : E \rightarrow \Re_+$, the *densest cut* problem is to find a cut $\delta(S)$ such that the ratio $u(\delta(S))/(|S| \cdot |V \backslash S|)$ is **maximum**.

   (a) Prove that a densest cut can be computed efficiently if and only if a sparsest cut can be computed efficiently.

   (b) Prove that the problem of finding the densest cut is NP-hard. Does this mean that the sparsest cut problem is NP-hard?

   (HINT: Give a reduction from the maxcut problem (find a cut of maximum capacity) on a graph G'; the maxcut problem is known to be NP-hard. Construct the prism G'' of G' by taking two copies of G' and adding an edge between the two copies of each G'-node.)

   (c) Give a 2-approximation algorithm for the densest cut problem, i.e., give an efficient algorithm for finding a cut whose ratio is at least half the ratio of a densest cut. Does this mean that there exists an $O(1)$-approximation algorithm for the sparsest cut problem?

   (HINT: Focus on the star cuts, $\delta(v), \forall v \in V$.)

2. Prove that the relaxation (F1) of the sparsest cut problem is equivalent to the LP relaxation (F2), in the sense that for every optimal solution d, λ of (F1) there exists a feasible solution of (F2) with the same objective value, and vice versa. Prove that the LP relaxations (F2) and (F3) are equivalent.

3. In Theorem 11.11, improve the integrality gap (and approximation guarantee) of $8\ln(n+1)+2$ by a factor of 2, by improving the analysis in the proof.

   (HINT: The *flux* of a cut $\delta(S)$ is defined to be $u(\delta(S))/\min(|S|, |V\backslash S|)$. Let β be the minimum flux of a cut. Note that $\alpha n/2 \le \beta \le \alpha n$, where α is the ratio of a sparsest cut. Modify the analysis at appropriate places by using the flux of a cut instead of the ratio of a cut.)

4. The *minimum multicut* problem is defined as follows. The input consists of a graph $G = (V, E)$ with edge capacities $u : E \to \Re_+$, together with a set of demand pairs (or demand edges) $s_1t_1, s_2t_2, \ldots, s_kt_k$ (each demand pair is an unordered pair of nodes; demand pairs have no relation to the edges in E, i.e., an edge s_it_i may or may not be present in E). The problem is to find an edge set $C \subseteq E$ such that the capacity $u(C) = \sum_{e\in C} u_e$ is minimum and such that $G\backslash C$ has no s_i-t_i path for each $i = 1, \ldots, k$; in other words, an optimal solution for the problem is a minimum-capacity edge set whose removal disconnects (or separates) all k demand pairs.

   (a) Write down an LP relaxation (FM) for the minimum multicut problem, based on the LP relaxation (F2) for the sparsest cut problem.

   (HINT: Replace the constraint $\sum_{v,w\in V} \lambda_{v,w} \ge 1$ by the k constraints $\lambda_{s_i,t_i} \ge 1$, $i = 1, \ldots, k$.)

   (b) Prove that the minimum capacity of a multicut is at most $O(\log k)$ times the optimal value of (FC). Give an efficient algorithm for finding a multicut whose capacity is at most $O(\log k)$ times the minimum capacity of a multicut.

   (HINT: Apply the region-growing procedure with the parameter ϵ chosen to guarantee that each region contains at most one node of a demand pair.)

   (c) (N. Kahale 1993) Use the algorithm in part (b) to design an $O((\log D)(\log k))$-approximation algorithm for the generalized sparsest cut problem (see Section 11.3), where D denotes the total demand $\sum_{v,w\in V} dem_{v,w}$. W.l.o.g. assume that each demand $dem_{v,w}$ is a (non-negative) integer.

   (HINT: First, find the optimal solution d of the LP relaxation (F4) of the generalized sparsest cut problem. Next, find a set $Q = \{s_i, t_i\}$ of demand pairs such that

$$d(Q) \ge \frac{1}{dem(Q) \cdot H(dem(Q))},$$

   where $d(Q) = \min\limits_{s_i,t_i\in Q} d_{s_i,t_i}$, $dem(Q) = \sum\limits_{s_i,t_i\in Q} dem_{s_i,t_i}$, and $H(i)$ is the ith harmonic number $1 + 1/2 + \ldots + 1/i$. To find Q, order the pairs $\{s_i, t_i\}$ with positive demands according to decreasing values of d_{s_i,t_i}, and renumber these pairs such that $d_{s_1,t_1} \geq d_{s_2,t_2} \geq \ldots \geq d_{s_i,t_i} \geq \ldots$, and let $D_i = \sum_{j=1}^{i} dem_{s_i,t_i}$. If for each i, $d_{s_i,t_i} < \dfrac{1}{D_i \cdot H(D_i)}$, then obtain a contradiction by showing that $\sum_i d_{s_i,t_i} dem_{s_i,t_i} < 1$. Having found Q, deduce that $d/d(Q)$ is a feasible solution to the LP relaxation (FM) of the minimum multicut problem specified by G, u and Q (i.e., the demand pairs for (FM) are precisely the pairs in Q). Finally, prove that the multicut found by the approximation algorithm in part (b) has ratio at most $O((\log D)(\log k))$ times the ratio of a generalized sparsest cut.)

5. Suppose that the graph $G = (V, E)$ is a tree, and let $u : E \rightarrow \Re_+$ assign capacities to the edges.
   (a) Can a sparsest cut be computed efficiently?

   (HINT: One way is to use the method of ℓ_1 embeddings: for any weight function on the edges $d : E \rightarrow \Re_+$, prove that the resulting shortest-paths metric space of G isometrically embeds into ℓ_1^n.)
   (b) Can a minimum multicut be computed efficiently? Note that the demand pairs may be arbitrary.

   (HINT: Attempt a reduction from the minimum node cover problem.)
6. The r-dimensional cube Q_r is defined recursively in terms of the cartesian product of Q_{r-1} and the complete graph on two nodes, K_2, as follows:

$$\begin{aligned} Q_1 &= K_2 \\ Q_r &= K_2 \times Q_{r-1}. \end{aligned}$$

   Alternatively, Q_r may be defined as a graph whose node set consists of 2^r r-dimensional boolean vectors, where two nodes are adjacent whenever they differ in exactly one coordinate. Note that Q_r has 2^r nodes and $r2^r/2$ edges.
   (a) Find a sparsest cut for Q_r and determine its ratio. True or false: the integrality gap for the LP relaxation (F2) on Q_r is one, i.e., the optimal value of (F2) equals the ratio of a sparsest cut.
   (b) Consider the LP relaxation (FM) of the minimum multicut problem on Q_r. Give a tight estimate (up to constant factors) for the integrality gap, i.e., find a function $g(r)$ such that the minimum capacity of a multicut on Q_r is at most $O(g(r))$ times the optimal value of (FM), and such that the integrality gap is $\Omega(g(r))$ on some example. Note that the demand pairs may be arbitrary.
7. (T. C. Hu's theorem) Consider the generalized sparsest cut problem such that exactly two pairs s_1, t_1 and s_2, t_2 have positive demands. Prove that the ratio of a generalized sparsest cut equals the optimal value of the LP relaxation (F4).

   (HINT: One way is to use the method of ℓ_1 embeddings. Construct an isometric embedding from ℓ_∞^2 to ℓ_1^2.)

8. The goal here is to fill in the details for the proof of Bourgain's theorem (Theorem 11.10).

   (a) Let V be a set of n points, and let B^o and B^c be disjoint subsets of V such that $|B^o| \leq 2^{j+1}$ and $|B^c| \geq 2^j$, where $0 \leq j \leq \lfloor \log_2 n \rfloor$ is an integer. Let A be a random subset of V of cardinality $n/2^j$. Prove that there exists a constant $c < 1$ such that with probability $\geq c$, $A \cap B^o$ is empty and $A \cap B^c$ is nonempty.

   (HINT: Argue that the events "$A \cap B^o$ is empty" and "$A \cap B^c$ is nonempty" are not negatively correlated, i.e., $\Pr(A \cap B^o = \emptyset \text{ and } A \cap B^c \neq \emptyset) \geq \Pr(A \cap B^o = \emptyset) \cdot \Pr(A \cap B^c \neq \emptyset)$. Then show that there exist constants c_1 and c_2 such that $\Pr(A \cap B^o = \emptyset) \geq c_1$ and $\Pr(A \cap B^c = \emptyset) \leq c_2$.)

   (b) Use Chernoff bounds to show that with high probability, $\sum_{k=1}^{q} |\phi'_{jq+k}(v) - \phi'_{jq+k}(w)| \geq qc\rho_j/2$, assuming that with probability $\geq c$ for each $k = 1, \ldots, q$, $|\phi'_{jq+k}(v) - \phi'_{jq+k}(w)| \geq \rho_j$.

9. Generalize the proof of Bourgain's theorem to embed into ℓ_p^h for an arbitary p, $p \geq 1$, instead of embedding into ℓ_1^h. The dimension h stays $O(\log^2 n)$ and the distortion stays $O(\log n)$.

   (HINT: Use the following monotonicity property of pth moment averages: For positive numbers $x_1, \ldots, x_k$, $\dfrac{x_1 + \ldots + x_k}{k} \leq \left(\dfrac{x_1^p + \ldots + x_k^p}{k} \right)^{1/p}$.)

10. Construct an example to show that the estimate of the distortion in Bourgain's theorem is tight up to constant factors, i.e., construct a metric space (V, d) such that every embedding into ℓ_1^h has distortion $\Omega(\log n)$, where the dimension h is arbitrary and $n = |V|$.

    (HINT: Consider the example of the sparsest cut problem on the expander graph G in Section 11.1, and focus on the metric space $(V(G), d)$, where d is the shortest-paths metric of G.)

Bibliography

[1] D. Adolphson and T. C. Hu, *Optimal linear ordering*, SIAM J. Appl. Math. **25** (1973), pp. 403-423.

[2] A. Agrawal, P. Klein, and R. Ravi, *Cutting down on fill using nested dissection: provably good elimination orderings*, in Graph Theory and Sparse Matrix Computation, edited by A. George, J. Gilbert, and J. W. H. Liu, Vol. 56 in the *IMA Volumes in Mathematics and its Applications*, Springer-Verlag (1993).

[3] Y. Aumann and Y. Rabani, *An $O(\log k)$ approximate max-flow min-cut theorem and approximation algorithm*, manuscript (1994) to appear in SIAM J. on Comput.

[4] S. N. Bhatt and F. T. Leighton, *A framework for solving VLSI graph layout problems*, J. Comput. Sys. Sciences, **28** (1994), pp. 300-343.

[5] J. Bourgain, *On Lipschitz embedding of finite metric spaces in Hilbert spaces*, Israeli J. Math. **52** (1985), 46-52.

[6] G. Even, J. Naor, S. Rao and B. Schieber, *Divide-and-conquer approximation algorithms via spreading metrics*, Proc. 36th FOCS '95 (1995), 62-71.

[7] N. Garg, *Approximating sparsest cuts*, manuscript, (1994).

[8] N. Garg, V. Vazirani, and M. Yannakakis, *Approximate max-flow min-(multi)cut theorems and their applications*, SIAM J. on Computing, **25:2** (1996), pp. 235-251.

[9] F. T. Leighton and S. Rao, *An approximate max-flow min-cut theorem for uniform multicommodity flow problems with application to approximation algorithms*, manuscript, 1994. An abstract appears in Proc. of the 29th Annual IEEE Conference on Foundations of Computer Science (1988), pp. 422-431.

[10] N. Linial, E. London and Y. Rabinovich, *The geometry of graphs and some of its algorithmic applications*, Combinatorica **15 (2)** (1995), pp. 215-245.

[11] S. Rao and A. W. Richa, *New approximation techniques for some ordering problems*, Proc. 9th SODA '98 (1998), 211-218.

[12] R. Ravi, A. Agrawal, and P. Klein, *Ordering problems approximated: single-processor scheduling and interval graph completion*, Proc. International Colloquium on Automata, Languages and Processing (ICALP '91) LNCS 510 (1991), pp. 751-762.

[13] D. Shmoys, *Cut problems and their application to divide-and-conquer*, in Approximation Algorithms, ed. Dorit Hochbaum, PWS (1997).