24 Facility Location

The facility location problem has occupied a central place in operations research since the early 1960's. It models design situations such as deciding placements of factories, warehouses, schools, and hospitals. Modern day applications include placement of proxy servers on the web.

In this chapter, we will present a primal–dual schema based factor 3 approximation algorithm for the special case when connection costs satisfy the triangle inequality. The algorithm differs in two respects from previous primal–dual algorithms. First, the primal and dual pair of LPs have negative coefficients and do not form a covering-packing pair. Second, we will relax primal complementary slackness conditions rather than the dual ones. Also, the idea of synchronization, introduced in the primal–dual schema in Chapter 22, is developed further, with an explicit timing of events playing a role.

Problem 24.1 (Metric uncapacitated facility location) Let G be a bipartite graph with bipartition (F, C), where F is the set of *facilities* and C is the set of *cities*. Let f_i be the cost of opening facility i, and c_{ij} be the cost of connecting city j to (opened) facility i. The connection costs satisfy the triangle inequality. The problem is to find a subset $I \subseteq F$ of facilities that should be opened, and a function $\phi : C \to I$ assigning cities to open facilities in such a way that the total cost of opening facilities and connecting cities to open facilities is minimized.

Consider the following integer program for this problem. In this program, y_i is an indicator variable denoting whether facility i is open, and x_{ij} is an indicator variable denoting whether city j is connected to the facility i. The first set of constraints ensures that each city is connected to at least one facility, and the second ensures that this facility must be open.

$$
\begin{aligned}
\text{minimize} \quad & \sum_{i \in F, \, j \in C} c_{ij} x_{ij} + \sum_{i \in F} f_i y_i & (24.1)
\end{aligned}
$$

$$
\begin{aligned}
\text{subject to} \quad & \sum_{i \in F} x_{ij} \geq 1, & j \in C \\
& y_i - x_{ij} \geq 0, & i \in F, \; j \in C \\
& x_{ij} \in \{0, 1\}, & i \in F, \; j \in C \\
& y_i \in \{0, 1\}, & i \in F
\end{aligned}
$$

The LP-relaxation of this program is:

$$\text{minimize} \quad \sum_{i \in F, \, j \in C} c_{ij} x_{ij} + \sum_{i \in F} f_i y_i \qquad (24.2)$$

$$\text{subject to} \quad \sum_{i \in F} x_{ij} \geq 1, \qquad\qquad j \in C$$

$$y_i - x_{ij} \geq 0, \qquad\qquad i \in F, \; j \in C$$

$$x_{ij} \geq 0, \qquad\qquad i \in F, \; j \in C$$

$$y_i \geq 0, \qquad\qquad i \in F$$

The dual program is:

$$\text{maximize} \quad \sum_{j \in C} \alpha_j \qquad (24.3)$$

$$\text{subject to} \quad \alpha_j - \beta_{ij} \leq c_{ij}, \quad i \in F, \; j \in C$$

$$\sum_{j \in C} \beta_{ij} \leq f_i, \qquad i \in F$$

$$\alpha_j \geq 0, \qquad\qquad j \in C$$

$$\beta_{ij} \geq 0, \qquad\qquad i \in F, \; j \in C$$

24.1   An intuitive understanding of the dual

Let us first give the reader some feel for how the dual variables "pay" for a primal solution by considering the following simple setting. Suppose LP (24.2) has an optimal solution that is integral, say $I \subseteq F$ and $\phi : C \to I$. Thus, under this solution, $y_i = 1$ iff $i \in I$, and $x_{ij} = 1$ iff $i = \phi(j)$. Let $(\boldsymbol{\alpha}, \boldsymbol{\beta})$ denote an optimal dual solution.

The primal and dual complementary slackness conditions are:

(i) $\qquad \forall i \in F, j \in C : x_{ij} > 0 \Rightarrow \alpha_j - \beta_{ij} = c_{ij}$

(ii) $\qquad \forall i \in F : y_i > 0 \Rightarrow \sum_{j \in C} \beta_{ij} = f_i$

(iii) $\qquad \forall j \in C : \alpha_j > 0 \Rightarrow \sum_{i \in F} x_{ij} = 1$

(iv) $\qquad \forall i \in F, j \in C : \beta_{ij} > 0 \Rightarrow y_i = x_{ij}$

By condition (ii), each open facility must be fully paid for, i.e., if $i \in I$, then

$$\sum_{j: \; \phi(j)=i} \beta_{ij} = f_i.$$

Consider condition (iv). Now, if facility i is open, but $\phi(j) \neq i$, then $y_i \neq x_{ij}$, and so $\beta_{ij} = 0$, i.e., city j does not contribute to opening any facility besides the one it is connected to.

By condition (i), if $\phi(j) = i$, then $\alpha_j - \beta_{ij} = c_{ij}$. Thus, we can think of α_j as the total price paid by city j; of this, c_{ij} goes towards the use of edge (i, j), and β_{ij} is the contribution of j towards opening facility i.

24.2 Relaxing primal complementary slackness conditions

Suppose the primal complementary slackness conditions were relaxed as follows, while maintaining the dual conditions:

$$\forall j \in C : \quad (1/3)c_{\phi(j)j} \;\leq\; \alpha_j - \beta_{\phi(j)j} \leq c_{\phi(j)j},$$

and

$$\forall i \in I : \quad (1/3)f_i \;\leq\; \sum_{j:\ \phi(j)=i} \beta_{ij} \;\leq\; f_i.$$

Then, the cost of the (integral) solution found would be within thrice the dual found, thus leading to a factor 3 approximation algorithm. However, we would like to obtain the stronger inequality stated in Theorem 24.7. Now, the dual pays at least one-third the connection cost, but must pay completely for opening facilities. This stronger inequality will be needed in order to use this algorithm to solve the k-median problem in Chapter 25.

For this reason, we will relax the primal conditions as follows. The cities are partitioned into two sets, *directly connected* and *indirectly connected*. Only directly connected cities will pay for opening facilities, i.e., β_{ij} can be nonzero only if j is a directly connected city and $i = \phi(j)$. For an indirectly connected city j, the primal condition is relaxed as follows:

$$(1/3)c_{\phi(j)j} \leq \alpha_j \leq c_{\phi(j)j}.$$

All other primal conditions are maintained, i.e., for a directly connected city j,

$$\alpha_j - \beta_{\phi(j)j} = c_{\phi(j)j},$$

and for each open facility i,

$$\sum_{j:\ \phi(j)=i} \beta_{ij} = f_i.$$

24.3 Primal–dual schema based algorithm

The algorithm consists of two phases. In Phase 1, the algorithm operates in a primal–dual fashion. It finds a dual feasible solution and also determines a set of tight edges and temporarily open facilities, F_t. Phase 2 consists of choosing a subset I of F_t to open, and finding a mapping, ϕ, from cities to I.

Algorithm 24.2

Phase 1
We would like to find as large a dual solution as possible. This motivates the following underlying process for dealing with the non-covering-packing pair of LPs. Each city j raises its dual variable, α_j, until it gets connected to an open facility. All other primal and dual variables simply respond to this change, trying to maintain feasibility or satisfying complementary slackness conditions.

   A notion of *time* is defined in this phase, so that each event can be associated with the time at which it happened; the phase starts at time 0. Initially, each city is defined to be *unconnected*. Throughout this phase, the algorithm raises the dual variable α_j for each unconnected city j uniformly at unit rate, i.e., α_j will grow by 1 in unit time. When $\alpha_j = c_{ij}$ for some edge (i, j), the algorithm will declare this edge to be *tight*. Henceforth, dual variable β_{ij} will be raised uniformly, thus ensuring that the first constraint in LP (24.3) is not violated. β_{ij} goes towards paying for facility i. Each edge (i, j) such that $\beta_{ij} > 0$ is declared *special*.

   Facility i is said to be *paid for* if $\sum_j \beta_{ij} = f_i$. If so, the algorithm declares this facility *temporarily open*. Furthermore, all unconnected cities having tight edges to this facility are declared *connected* and facility i is declared the *connecting witness* for each of these cities. (Notice that the dual variables α_j of these cities are not raised anymore.) In the future, as soon as an unconnected city j gets a tight edge to i, j will also be declared connected and i will be declared the connecting witness for j (notice that $\beta_{ij} = 0$ and thus edge (i, j) is not special). When all cities are connected, the first phase terminates. If several events happen simultaneously, the algorithm executes them in arbitrary order.

Remark 24.3   At the end of Phase 1, a city may have paid towards temporarily opening several facilities. However, we want to ensure that a city pays only for the facility that it is eventually connected to. This is ensured in Phase 2, which chooses a subset of temporarily open facilities for opening permanently.

Phase 2
Let F_t denote the set of temporarily open facilities and T denote the subgraph of G consisting of all special edges. Let T^2 denote the graph that has edge (u, v) iff there is a path of length at most 2 between u and v in T, and let H

be the subgraph of T^2 induced on F_t. Find any maximal independent set in H, say I. All facilities in the set I are declared *open*.

For city j, define $\mathcal{F}_j = \{i \in F_t \mid (i,j) \text{ is special}\}$. Since I is an independent set, at most one of the facilities in $\mathcal{F}_j$ is opened. If there is a facility $i \in \mathcal{F}_j$ that is opened, then set $\phi(j) = i$ and declare city j *directly connected*. Otherwise, consider tight edge (i', j) such that i' was the connecting witness for j. If $i' \in I$, again set $\phi(j) = i'$ and declare city j *directly connected* (notice that in this case $\beta_{i'j} = 0$). In the remaining case that $i' \notin I$, let i be any neighbor of i' in graph H such that $i \in I$. Set $\phi(j) = i$ and declare city j *indirectly connected*.

I and ϕ define a primal integral solution: $x_{ij} = 1$ iff $\phi(j) = i$ and $y_i = 1$ iff $i \in I$. The values of α_j and β_{ij} obtained at the end of Phase 1 form a dual feasible solution.

24.4 Analysis

We will show how the dual variables α_j's pay for the primal costs of opening facilities and connecting cities to facilities. Denote by α_j^f and α_j^e the contributions of city j to these two costs respectively; $\alpha_j = \alpha_j^f + \alpha_j^e$. If j is indirectly connected, then $\alpha_j^f = 0$ and $\alpha_j^e = \alpha_j$. If j is directly connected, then the following must hold:

$$\alpha_j = c_{ij} + \beta_{ij},$$

where $i = \phi(j)$. Now, let $\alpha_j^f = \beta_{ij}$ and $\alpha_j^e = c_{ij}$.

Lemma 24.4 *Let $i \in I$. Then,*

$$\sum_{j:\, \phi(j)=i} \alpha_j^f = f_i.$$

Proof: Since i is temporarily open at the end of Phase 1, it is completely paid for, i.e.,

$$\sum_{j:\, (i,j) \text{ is special}} \beta_{ij} = f_i.$$

The critical observation is that each city j that has contributed to f_i must be directly connected to i. For each such city, $\alpha_j^f = \beta_{ij}$. Any other city j' that is connected to facility i must satisfy $\alpha_{j'}^f = 0$. The lemma follows. $\square$

Corollary 24.5 $\sum_{i \in I} f_i = \sum_{j \in C} \alpha_j^f$.

Recall that α_j^f was defined to be 0 for indirectly connected cities. Thus, only the directly connected cities pay for the cost of opening facilities.

Lemma 24.6 *For an indirectly connected city j, $c_{ij} \leq 3\alpha_j^e$, where $i = \phi(j)$.*

Proof: Let i' be the connecting witness for city j. Since j is indirectly connected to i, (i, i') must be an edge in H. In turn, there must be a city, say j', such that (i, j') and (i', j') are both special edges. Let t_1 and t_2 be the times at which i and i' were declared temporarily open during Phase 1.
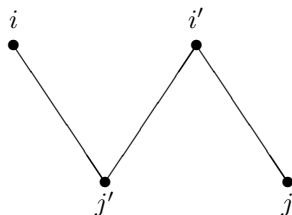
Since edge (i', j) is tight, $\alpha_j \geq c_{i'j}$. We will show that $\alpha_j \geq c_{ij'}$ and $\alpha_j \geq c_{i'j'}$. Then, the lemma will follow by using the triangle inequality.

Since edges (i', j') and (i, j') are tight, $\alpha_{j'} \geq c_{ij'}$ and $\alpha_{j'} \geq c_{i'j'}$. Since both these edges are special, they must both have gone tight before either i or i' is declared temporarily open. Consider the time $\min(t_1, t_2)$. Clearly, $\alpha_{j'}$ cannot be growing beyond this time. Therefore, $\alpha_{j'} \leq \min(t_1, t_2)$. Finally, since i' is the connecting witness for j, $\alpha_j \geq t_2$. Therefore, $\alpha_j \geq \alpha_{j'}$, and the required inequalities follow. $\square$

Theorem 24.7 *The primal and dual solutions constructed by the algorithm satisfy:*

$$\sum_{i \in F, \, j \in C} c_{ij} x_{ij} + 3 \sum_{i \in F} f_i y_i \leq 3 \sum_{j \in C} \alpha_j.$$

Proof: For a directly connected city j, $c_{ij} = \alpha_j^e \leq 3\alpha_j^e$, where $\phi(j) = i$. Combining with Lemma 24.6 we get

$$\sum_{i \in F, j \in C} c_{ij} x_{ij} \leq 3 \sum_{j \in C} \alpha_j^e.$$

Adding to this the equality stated in Corollary 24.5 multiplied by 3 gives the theorem. $\square$

24.4.1   Running time

A special feature of the primal–dual schema is that it yields algorithms with good running times. Since this is especially so for the current algorithm, we will provide some implementation details. We will adopt the following notation: $n_c = |C|$ and $n_f = |F|$. The total number of vertices $n_c + n_f = n$, and the total number of edges $n_c \times n_f = m$.

Sort all the edges by increasing cost – this gives the order and the times at which edges go tight. For each facility, i, we maintain the number of cities that are currently contributing towards it, and the *anticipated time*, t_i, at which it would be completely paid for if no other event happens on the way. Initially all t_i's are infinite, and each facility has 0 cities contributing to it. The t_i's are maintained in a binary heap so we can update each one and find the current minimum in $O(\log n_f)$ time. Two types of events happen, and they lead to the following updates.

- An edge (i, j) goes tight.
  - If facility i is not temporarily open, then it gets one more city contributing towards its cost. The amount contributed towards its cost at the current time can be easily computed. Therefore, the anticipated time for facility i to be paid for can be recomputed in constant time. The heap can be updated in $O(\log n_f)$ time.
  - If facility i is already temporarily open, city j is declared connected, and α_j is not raised anymore. For each facility i' that was counting j as a contributor, we need to decrease the number of contributors by 1 and recompute the anticipated time at which it gets paid for.
- Facility i is completely paid for. In this event, i will be declared temporarily open, and all cities contributing to i will be declared connected. For each of these cities, we will execute the second case of the previous event, i.e., update facilities that they were contributing towards.

The next theorem follows by observing that each edge (i, j) will be considered at most twice. First, when it goes tight. Second, when city j is declared connected. For each consideration of this edge, we will do $O(\log n_f)$ work.

Theorem 24.8 *Algorithm 24.2 achieves an approximation factor of 3 for the facility location problem and has a running time of $O(m \log m)$.*

24.4.2   Tight example

The following infinite family of examples shows that the analysis of our algorithm is tight: The graph has n cities, $c_1, c_2, \ldots, c_n$ and two facilities f_1 and f_2. Each city is at a distance of 1 from f_2. City c_1 is at a distance of 1 from f_1, and $c_2, \ldots, c_n$ are at a distance of 3 from f_1. The opening cost of f_1 and f_2 are ε and $(n + 1)\varepsilon$, respectively, for a small number ε.
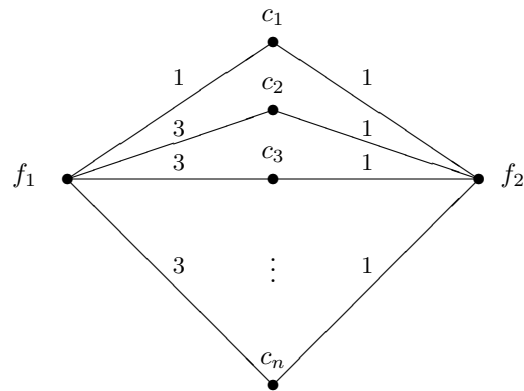
The optimal solution is to open f_2 and connect all cities to it, at a total cost of $(n+1)\varepsilon + n$. Algorithm 24.2 will however open facility f_1 and connect all cities to it, at a total cost of $\varepsilon + 1 + 3(n-1)$.

24.5 Exercises

24.1 Consider the general uncapacitated facility location problem in which the connection costs are not required to satisfy the triangle inequality. Give a reduction from the set cover problem to show that approximating this problem is as hard as approximating set cover and therefore cannot be done better than $O(\log n)$ factor unless $\mathbf{NP} \subseteq \tilde{\mathbf{P}}$. Also, give an $O(\log n)$ factor algorithm for this problem.

24.2 In Phase 2, instead of picking all special edges in T, pick all tight edges. Show that now Lemma 24.6 does not hold. Give a suitable modification to the algorithm that restores Lemma 24.6.
Hint: Order facilities in H in the order in which they were temporarily opened, and pick I to be the lexicographically first maximal independent set.

24.3 Give a factor 3 tight example for Algorithm 24.2 in which the set of cities and facilities is the same, i.e., $C = F$.

24.4 Consider the proof of Lemma 24.6. Give an example in which $\alpha_j > t_2$.

24.5 The vector $\boldsymbol{\alpha}$ found by Algorithm 24.2 is maximal in the sense that if we increase any α_j in this vector, then there is no way of setting the β_{ij}'s to get a feasible dual solution. Is every maximal solution $\boldsymbol{\alpha}$ within 3 times the optimal solution to dual program for facility location?

Hint: It is easy to construct a maximal solution that is $2/n$ times the optimal. Consider n facilities with an opening cost of 1 each and n cities connected to distinct facilities by edges of cost ε each. In addition, there is another city that is connected to each facility with an edge of cost 1.

24.6 Consider the following modification to the metric uncapacitated facility location problem. Define the cost of connecting city j to facility i to be c_{ij}^2. The c_{ij}'s still satisfy the triangle inequality (but the new connection costs, of c_{ij}^2, do not). Show that Algorithm 24.2 achieves an approximation guarantee of factor 9 for this case.

24.7 Consider the following generalization to arbitrary demands. For each city j, a nonnegative demand d_j is specified, and any open facility can serve this demand. The cost of serving this demand via facility i is $c_{ij}d_j$. Give an IP and LP-relaxation for this problem, and extend Algorithm 24.2 to get a factor 3 algorithm.
Hint: Raise α_j at rate d_j.

24.8 In the *capacitated* facility location problem, we are given a number u_i for each facility i, and facility i can serve at most u_i cities. Show that the modification of LP (24.2) to this problem has an unbounded integrality gap.

24.9 Consider the variant of the capacitated metric facility location problem in which each facility can be opened an unbounded number of times. If facility i is opened y_i times, it can serve at most $u_i y_i$ cities. Give an IP and LP-relaxation for this problem, and extend Algorithm 24.2 to obtain a constant factor algorithm.

24.10 (Charikar, Khuller, Mount, and Narshimhan [40]) Consider the *prize-collecting* variant of the facility location problem, in which there is a specified penalty for not connecting a city to an open facility. The objective is to minimize the sum of the connection costs, facility opening costs, and penalties. Give a factor 3 approximation algorithm for this problem.

24.11 (Jain and Vazirani [140]) Consider the *fault tolerant* variant of the facility location problem, in which the additional input is a connection requirement r_j for each city j. In the solution, city j needs to be connected to r_j distinct open facilities. The objective, as before, is to minimize the sum of the connection costs and the facility opening costs.

Decompose the problem into k phases, numbered k down to 1, as in Exercise 23.7. In phase p, all cities having a residual requirement of p are provided one more connection to an open facility. In phase p, the facility location algorithm of this chapter is run on the following modified graph, G_p. The cost of each facility that is opened in an earlier phase is set to 0. If city j is connected to facility i in an earlier phase, then c_{ij} is set to ∞.

1. Show that even though G_p violates the triangle inequality at some places, the algorithm gives a solution within factor 3 of the optimal solution for this graph.

   Hint: Every time short-cutting is needed; the triangle inequality holds.

2. Show that the solution found in phase p is of cost at most $3 \cdot \text{OPT}/p$, where OPT is the cost of the solution to the entire problem.

   Hint: Remove ∞ cost edges of G_p from the optimal solution and divide the rest by p. Show that this is a feasible fractional solution for phase p.

3. Show that this algorithm achieves an approximation factor of $3 \cdot H_k$ for the fault tolerant facility location problem.

24.12 (Mahdian, Markakis, Saberi, and Vazirani [201]) This exercise develops a factor 3 greedy algorithm for the metric uncapacitated facility location problem, together with an analysis using the method of dual fitting.

Consider the following modification to Algorithm 24.2. As before, dual variables, α_j, of all unconnected cities, j, are raised uniformly. If edge (i, j) is tight, β_{ij} is raised. As soon as a facility, say i, is paid for, it is declared open. Let S be the set of unconnected cities having tight edges to i. Each city $j \in S$ is declared connected and stops raising its α_j. So far, the new algorithm is the same as Algorithm 24.2. The main difference appears at this stage: Each city $j \in S$ withdraws its contribution from other facilities, i.e., for each facility $i' \neq i$, set $\beta_{i'j} = 0$. When all cities have been declared connected, the algorithm terminates. Observe that each city contributes towards the opening cost of at most one facility – the facility it gets connected to.

1. This algorithm actually has a simpler description as a greedy algorithm. Provide this description.

   Hint: Use the notion of cost–effectiveness defined for the greedy set cover algorithm.

2. The next 3 parts use the method of dual fitting to analyze this algorithm. First observe that the primal solution found is fully paid for by the dual computed.

3. Let i be an open facility and let $\{1, \ldots, k\}$ be the set of cities that contributed to opening i at some point in the algorithm. Assume w.l.o.g. that $\alpha_1 \leq \alpha_j$ for $j \leq k$. Show that for $j \leq k$, $\alpha_j - c_{ij} \leq 2\alpha_1$. Also, show that

$$\sum_{j=1}^{k} \alpha_j \leq 3 \sum_{j=1}^{k} c_{ij} + f_i.$$

   Hint: Use the triangle inequality and the following inequality which is a consequence of the fact that at any point, the total amount contributed for opening facility i is at most f_i:

$$\sum_{j:\ c_{ij}\le\alpha_1}\alpha_1-c_{ij}\le f_i.$$

4. Hence show that $\boldsymbol{\alpha}/3$ is a dual feasible solution.
5. How can the analysis be improved – a factor 1.86 analysis is known for this algorithm.
6. Give a time efficient implementation of this algorithm, matching the running time of Algorithm 24.2
7. Do you see room for improving the algorithm?
   Hint: Suppose city j is connected to open facility i at some point in the algorithm. Later, facility i' is opened, and suppose that $c_{ij} > c_{i'j}$. Then, connecting j to i' will reduce the cost of the solution.

24.13 (Mahdian, Markakis, Saberi, and Vazirani [201]) Consider the following variant of the metric uncapacitated facility location problem. Instead of f_i, the opening cost for each facility $i \in F$, we are provided a startup cost s_i and an incremental cost t_i. Define the new opening cost for connecting $k > 0$ cities to facility i to be $s_i + kt_i$. Connection costs are specified by a metric, as before. The object again is to connect each city to an open facility so as to minimize the sum of connection costs and opening costs. Give an approximation factor preserving reduction from this problem to the metric uncapacitated facility location problem.
Hint: Modify the metric appropriately.

24.6 Notes

The first approximation algorithm for the metric uncapacitated facility location problem, due to Hochbaum [124], achieved an approximation guarantee of $O(\log n)$. The first constant factor approximation algorithm, achieving a guarantee of 3.16, was due to Shmoys, Tardos, and Aardal [239]. It was based on LP-rounding. The current best algorithm, achieving an approximation guarantee of 1.61, is due to Jain, Mahdian, and Saberi [138]. This algorithm, a small modification of the greedy algorithm presented in Exercise 24.12, is analyzed using the method of dual fitting. The primal–dual schema based Algorithm 24.2 is due to Jain and Vazirani [141].

25 k-Median

The k-median problem differs from the facility location problem in two respects – there is no cost for opening facilities and there is an upper bound, k, on the number of facilities that can be opened. It models the problem of finding a minimum cost clustering, and therefore has numerous applications.

The primal–dual schema works by making judicious local improvements and is not suitable for ensuring a global constraint, such as the constraint in the k-median problem that at most k facilities be opened. We will get around this difficulty by borrowing the powerful technique of Lagrangian relaxation from combinatorial optimization.

Problem 25.1 (Metric k-median) Let G be a bipartite graph with bipartition (F, C), where F is the set of *facilities* and C is the set of *cities*, and let k be a positive integer specifying the number of facilities that are allowed to be opened. Let c_{ij} be the cost of connecting city j to (opened) facility i. The connection costs satisfy the triangle inequality. The problem is to find a subset $I \subseteq F, |I| \leq k$, of facilities that should be opened and a function $\phi : \; C \to I$ assigning cities to open facilities in such a way that the total connecting cost is minimized.

25.1 LP-relaxation and dual

The following is an integer program for the k-median problem. The indicator variables y_i and x_{ij} play the same role as in (24.1).

$$\text{minimize} \quad \sum_{i \in F, \; j \in C} c_{ij} x_{ij} \qquad\qquad (25.1)$$

$$\text{subject to} \quad \sum_{i \in F} x_{ij} \geq 1, \qquad j \in C$$

$$y_i - x_{ij} \geq 0, \qquad i \in F, \; j \in C$$

$$\sum_{i \in F} -y_i \geq -k$$

$$x_{ij} \in \{0, 1\}, \qquad i \in F, \; j \in C$$

$$y_i \in \{0, 1\}, \qquad i \in F$$

The LP-relaxation of this program is:

$$\text{minimize} \qquad \sum_{i \in F,\, j \in C} c_{ij} x_{ij} \qquad\qquad (25.2)$$

$$\text{subject to} \quad \sum_{i \in F} x_{ij} \geq 1, \qquad j \in C$$

$$y_i - x_{ij} \geq 0, \qquad i \in F,\ j \in C$$

$$\sum_{i \in F} -y_i \geq -k$$

$$x_{ij} \geq 0, \qquad\qquad i \in F,\ j \in C$$

$$y_i \geq 0, \qquad\qquad i \in F$$

The dual program is:

$$\text{maximize} \qquad \sum_{j \in C} \alpha_j - zk \qquad\qquad (25.3)$$

$$\text{subject to} \quad \alpha_j - \beta_{ij} \leq c_{ij}, \qquad i \in F,\ j \in C$$

$$\sum_{j \in C} \beta_{ij} \leq z, \qquad\qquad i \in F$$

$$\alpha_j \geq 0, \qquad\qquad j \in C$$

$$\beta_{ij} \geq 0, \qquad\qquad i \in F,\ j \in C$$

$$z \geq 0$$

25.2   The high-level idea

The similarity between the two problems, facility location and k-median, leads to a similarity in their linear programs, which will be exploited as follows. Take an instance of the k-median problem, assign a cost of z for opening each facility, and find optimal solutions to LP (24.2) and LP (24.3), say $(\boldsymbol{x}, \boldsymbol{y})$ and $(\boldsymbol{\alpha}, \boldsymbol{\beta})$, respectively. By the strong duality theorem,

$$\sum_{i \in F,\, j \in C} c_{ij} x_{ij} + \sum_{i \in F} z y_i \;=\; \sum_{j \in C} \alpha_j.$$

Now, suppose that the primal solution $(\boldsymbol{x}, \boldsymbol{y})$ happens to open exactly k facilities (fractionally), i.e., $\sum_i y_i = k$. Then, we claim that $(\boldsymbol{x}, \boldsymbol{y})$ and $(\boldsymbol{\alpha}, \boldsymbol{\beta}, z)$ are optimal solutions to LP (25.2) and LP (25.3), respectively. Feasibility is easy to check. Optimality follows by substituting $\sum_i y_i = k$ in the above equality and rearranging terms to show that the primal and dual solutions achieve the same objective function value:

$$\sum_{i \in F, \, j \in C} c_{ij} x_{ij} \;=\; \sum_{j \in C} \alpha_j - zk.$$

Let's use this idea, together with Algorithm 24.2 and Theorem 24.7, to obtain a "good" integral solution to LP (25.2). Suppose with a cost of z for opening each facility, Algorithm 24.2, happens to find solutions $(\boldsymbol{x}, \boldsymbol{y})$ and $(\boldsymbol{\alpha}, \boldsymbol{\beta})$, where the primal solution opens exactly k facilities. By Theorem 24.7,

$$\sum_{i \in F, \, j \in C} c_{ij} x_{ij} + 3zk \;\le\; 3 \sum_{j \in C} \alpha_j.$$

Now, observe that $(\boldsymbol{x}, \boldsymbol{y})$ and $(\boldsymbol{\alpha}, \boldsymbol{\beta}, z)$ are primal (integral) and dual feasible solutions to the k-median problem satisfying

$$\sum_{i \in F, \, j \in C} c_{ij} x_{ij} \;\le\; 3 \big(\sum_{j \in C} \alpha_j - zk \big).$$

Therefore, $(\boldsymbol{x}, \boldsymbol{y})$ is a solution to the k-median problem within thrice the optimal.

Notice that the factor 3 proof given above would not work if less than k facilities were opened; if more than k facilities are opened, the solution is infeasible for the k-median problem. The remaining problem is to find a value of z so that *exactly k* facilities are opened. Several ideas are required for this. The first is the following principle from economics. Taxation is an effective way of controlling the amount of goods coming across a border – raising tariffs will reduce inflow and vice versa. In a similar manner, raising z should reduce the number of facilities opened and vice versa.

It is natural now to seek a modification to Algorithm 24.2 that can find a value of z so that exactly k facilities are opened. This would lead to a factor 3 approximation algorithm. Such a modification is not known. Instead, we present the following strategy which leads to a factor 6 algorithm. For the rest of the discussion, assume that we never encountered a run of the algorithm which resulted in exactly k facilities being opened.

Clearly, when $z = 0$ the algorithm will open all facilities, and when z is very large it will open only one facility. The latter value of z can be picked to be $n c_{\max}$, where $c_{\max}$ is the length of the longest edge. We will conduct a binary search on the interval $[0, n c_{\max}]$ to find z_2 and z_1 for which the algorithm opens $k_2 > k$ and $k_1 < k$ facilities, respectively, and, furthermore, $z_1 - z_2 \le (c_{\min}/12 n_c^2)$, where $c_{\min}$ is the length of the shortest nonzero edge. As before, we will adopt the following notation: $n_c = |C|$ and $n_f = |F|$. The total number of vertices $n_c + n_f = n$ and the total number of edges $n_c \times n_f = m$. Let $(\boldsymbol{x}^s, \boldsymbol{y}^s)$ and $(\boldsymbol{x}^l, \boldsymbol{y}^l)$ be the two primal solutions found, with $\sum_{i \in F} y_i^s = k_1$ and $\sum_{i \in F} y_i^l = k_2$ (the superscripts s and l denote "small" and "large," respectively). Further, let (α^s, β^s) and (α^l, β^l) be the corresponding dual solutions found.

Let $(\boldsymbol{x}, \boldsymbol{y}) = a(\boldsymbol{x}^s, \boldsymbol{y}^s) + b(\boldsymbol{x}^l, y^l)$ be a convex combination of these two solutions, with $ak_1 + bk_2 = k$. Under these conditions, $a = (k_2 - k)/(k_2 - k_1)$ and $b = (k - k_1)/(k_2 - k_1)$. Since $(\boldsymbol{x}, \boldsymbol{y})$ is a feasible (fractional) solution to the facility location problem that opens exactly k facilities, it is also a feasible (fractional) solution to the k-median problem. In this solution each city is connected to at most two facilities.

Lemma 25.2 *The cost of $(\boldsymbol{x}, \boldsymbol{y})$ is within a factor of $(3 + 1/n_c)$ of the cost of an optimal fractional solution to the k-median problem.*

Proof: By Theorem 24.7 we have

$$\sum_{i \in F,\, j \in C} c_{ij} x_{ij}^s \;\leq\; 3(\sum_{j \in C} \alpha_j^s - z_1 k_1),$$

and

$$\sum_{i \in F,\, j \in C} c_{ij} x_{ij}^l \;\leq\; 3(\sum_{j \in C} \alpha_j^l - z_2 k_2).$$

Since $z_1 > z_2$, $(\boldsymbol{\alpha}^l, \boldsymbol{\beta}^l)$ is a feasible dual solution to the facility location problem even if the cost of facilities is z_1. We would like to replace z_2 with z_1 in the second inequality, at the expense of the increased factor. This is achieved using the upper bound on $z_1 - z_2$ and the fact that $\sum_{i \in F,\, j \in C} c_{ij} x_{ij}^l \;\geq\; c_{\min}$. We get

$$\sum_{i \in F,\, j \in C} c_{ij} x_{ij}^l \;\leq\; \left(3 + \frac{1}{n_c}\right) \left(\sum_{j \in C} \alpha_j^l - z_1 k_2\right).$$

Adding this inequality multiplied by b with the first inequality multiplied by a gives

$$\sum_{i \in F,\, j \in C} c_{ij} x_{ij} \;\leq\; \left(3 + \frac{1}{n_c}\right) \left(\sum_{j \in C} \alpha_j - z_1 k\right),$$

where $\boldsymbol{\alpha} = a\boldsymbol{\alpha}^s + b\boldsymbol{\alpha}^l$. Let $\boldsymbol{\beta} = a\boldsymbol{\beta}^s + b\boldsymbol{\beta}^l$. Observe that $(\boldsymbol{\alpha}, \boldsymbol{\beta}, z_1)$ is a feasible solution to the dual of the k-median problem. The lemma follows.   □

In Section 25.3 we give a randomized rounding procedure that obtains an integral solution to the k-median problem from $(\boldsymbol{x}, \boldsymbol{y})$, with a small increase in cost. In Section 25.3.1 we derandomize this procedure.

25.3   Randomized rounding

We give a randomized rounding procedure that produces an integral solution to the k-median problem from $(\boldsymbol{x}, \boldsymbol{y})$. In the process, it increases the cost by a multiplicative factor of $1 + \max(a, b)$.

Let A and B be the sets of facilities opened in the two solutions, $|A| = k_1$ and $|B| = k_2$. For each facility in A, find the closest facility in B – these facilities are not required to be distinct. Let $B' \subset B$ be these facilities. If $|B'| < k_1$, arbitrarily include additional facilities from $B - B'$ into B' until $|B'| = k_1$.

With probability a, open all facilities in A, and with probability $b = 1 - a$, open all facilities in B'. In addition, a set of cardinality $k - k_1$ is picked randomly from $B - B'$ and facilities in this set are opened. Notice that each facility in $B - B'$ has a probability of b of being opened. Let I be the set of facilities opened, $|I| = k$.

The function $\phi : C \to I$ is defined as follows. Consider city j and suppose that it is connected to $i_1 \in A$ and $i_2 \in B$ in the two solutions. If $i_2 \in B'$, then one of i_1 and i_2 is opened by the procedure given above, i_1 with probability a and i_2 with probability b. City j is connected to the open facility.
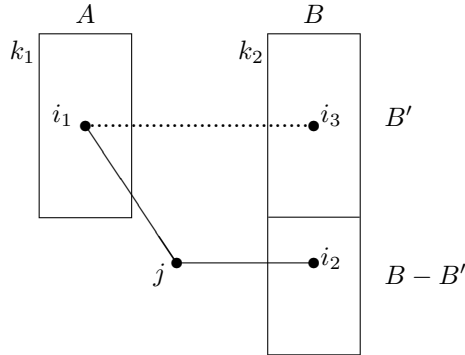
If $i_2 \in B - B'$, let $i_3 \in B'$ be the facility in B that is closest to i_1. City j is connected to i_2 if it is open. Otherwise, it is connected to i_1 if it is open. If neither i_2 or i_1 is open, then j is connected to i_3.

Denote by $\mathrm{cost}(j)$ the connection cost for city j in the fractional solution $(\boldsymbol{x}, \boldsymbol{y})$; $\mathrm{cost}(j) = ac_{i_1 j} + bc_{i_2 j}$.

Lemma 25.3 *The expected connection cost for city j in the integral solution, $\mathbf{E}[c_{\phi(j)j}]$, is $\leq (1 + \max(a, b))\mathrm{cost}(j)$. Moreover, $\mathbf{E}[c_{\phi(j)j}]$ can be efficiently computed.*

Proof: If $i_2 \in B'$, $\mathbf{E}[c_{\phi(j)j}] = ac_{i_1 j} + bc_{i_2 j} = \mathrm{cost}(j)$. Consider the second case, that $i_2 \notin B'$. Now, i_2 is open with probability b. The probability that i_2 is not open and i_1 is open is $(1 - b)a = a^2$, and the probability that both i_2 and i_1 are not open is $(1 - b)(1 - a) = ab$. This gives

$$\mathbf{E}[c_{\phi(j)j}] \le bc_{i_2 j} + a^2 c_{i_1 j} + abc_{i_3 j}.$$

Since i_3 is the facility in B that is closest to i_1, $c_{i_1 i_3} \le c_{i_1 i_2} \le c_{i_1 j} + c_{i_2 j}$, where the second inequality follows from the triangle inequality. Again, by the triangle inequality, $c_{i_3 j} \le c_{i_1 j} + c_{i_1 i_3} \le 2c_{i_1 j} + c_{i_2 j}$. Therefore,

$$\mathbf{E}[c_{\phi(j)j}] \le bc_{i_2 j} + a^2 c_{i_1 j} + ab(2c_{i_1 j} + c_{i_2 j}).$$

Now, $a^2 c_{i_1 j} + abc_{i_1 j} = ac_{i_1 j}$. Therefore,

$$\mathbf{E}[c_{\phi(j)j}] \le (ac_{i_1 j} + bc_{i_2 j}) + ab(c_{i_1 j} + c_{i_2 j})$$
$$\le (ac_{i_1 j} + bc_{i_2 j})(1 + \max(a, b)).$$

Clearly, in both cases, $\mathbf{E}[c_{\phi(j)j}]$ is easy to compute.   □

Let $(\boldsymbol{x}^k, \boldsymbol{y}^k)$ denote the integral solution obtained to the k-median problem by this randomized rounding procedure. Then,

Lemma 25.4 $$\mathbf{E}\left[\sum_{i \in F,\, j \in C} c_{ij} x_{ij}^k\right] \le (1 + \max(a, b)) \left(\sum_{i \in F,\, j \in C} c_{ij} x_{ij}\right)$$

and, moreover, the expected cost of the solution found can be computed efficiently.

25.3.1   Derandomization

Derandomization follows in a straightforward manner using the method of conditional expectation. First, the algorithm opens the set A with probability a and the set B' with probability $b = 1 - a$. Pick A, and compute the expected value if $k - k_1$ facilities are randomly chosen from $B - B'$. Next, do the same by picking B' instead of A. Choose to open the set that gives the smaller expectation.

Second, the algorithm opens a random subset of $k - k_1$ facilities from $B - B'$. For a choice $D \subset B - B'$, $|D| \le k - k_1$, denote by $\mathbf{E}[D, B - (B' \cup D)]$ the expected cost of the solution if all facilities in D and additionally $k - k_1 - |D|$ facilities are randomly opened from $B - (B' \cup D)$. Since each facility of $B - (B' \cup D)$ is equally likely to be opened, we get

$$\mathbf{E}[D, B - (B' \cup D)] =$$
$$\frac{1}{|B - (B' \cup D)|} \sum_{i \in B - (B' \cup D)} \mathbf{E}[D \cup \{i\}, B - (B' \cup D \cup \{i\})].$$

This implies that there is an i such that

$$\mathbf{E}[D \cup \{i\}, B - (B' \cup D \cup \{i\})] \le \mathbf{E}[B', B - (B' \cup D)].$$

Choose such an i and replace D with $D \cup \{i\}$. Notice that the computation of $\mathbf{E}[D \cup \{i\}, B - (B' \cup D \cup \{i\})]$ can be done as in Lemma 25.4.

25.3.2   Running time

It is easy to see that $a \le 1 - 1/n_c$ (this happens for $k_1 = k - 1$ and $k_2 = n_c$) and $b \le 1 - 1/k$ (this happens for $k_1 = 1$ and $k_2 = k + 1$). Therefore, $1 + \max(a, b) \le 2 - 1/n_c$. Altogether, the approximation guarantee is $(2 - 1/n_c)(3 + 1/n_c) < 6$. This procedure can be derandomized using the method of conditional probabilities, as in Section 25.3.1. The binary search will make $O(\log_2(n^3 c_{\max}/c_{\min})) = O(L + \log n)$ probes. The running time for each probe is dominated by the time taken to run Algorithm 24.2; randomized rounding takes $O(n)$ time and derandomization takes $O(m)$ time. Hence we get

Theorem 25.5 *The algorithm given above achieves an approximation factor of 6 for the k-median problem, and has a running time of $O(m \log m(L + \log(n)))$.*

25.3.3   Tight example

A tight example for the factor 6 k-median algorithm is not known. However, below we give an infinite family of instances which show that the analysis of the randomized rounding procedure cannot be improved.

The two solutions $(\boldsymbol{x}^s, \boldsymbol{y}^s)$ and $(\boldsymbol{x}^l, \boldsymbol{y}^l)$ open one facility, f_0, and $k + 1$ facilities, $f_1, \dots, f_{k+1}$, respectively. The distance between f_0 and any other f_i is 1, and that between two facilities in the second set is 2. All n cities are at a distance of 1 from f_0, and at a distance of ε from f_{k+1}. The rest of the distances are given by the triangle inequality. The convex combination is constructed with $a = 1/k$ and $b = 1 - 1/k$.
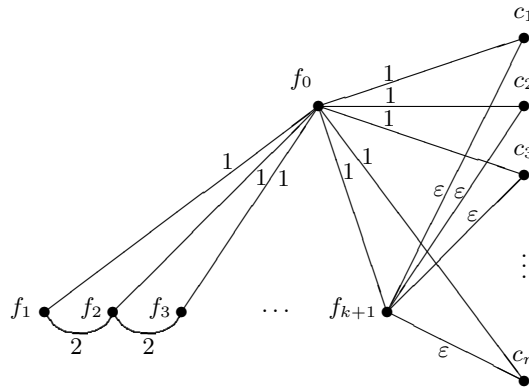
Now, the cost of the convex combination is $an + b\varepsilon n$. Suppose the algorithm picks f_1 as the closest neighbor of f_0. The expected cost of the solutions produced by the randomized rounding procedure is then $n(b\varepsilon + a^2 + ab(2+\varepsilon))$. Letting ε tend to 0, the cost of the convex combination is essentially na and that of the rounded solution is $na(1 + b)$.

25.3.4   Integrality gap

The algorithm given above places an upper bound of 6 on the integrality gap of relaxation (25.2). The following example places a lower bound of essentially 2. The graph is a star with $n + 1$ vertices and unit cost edges. F consists of all $n + 1$ vertices, C consists of all but the center vertex and $k = n - 2$. An optimal integral solution is to open facilities at $n - 2$ vertices of C and has a cost of 2. Consider the following fractional solution. Open a facility to the extent of $1/(n - 1)$ on the center vertex and $(n - 2)/(n - 1)$ on each vertex of C. This has a cost of $n/(n - 1)$, giving a ratio of $2(n - 1)/n$.
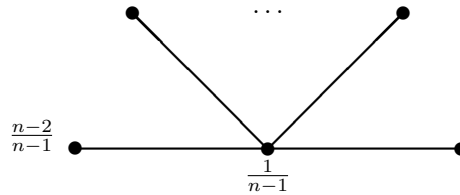
25.4   A Lagrangian relaxation technique for approximation algorithms

In this section we will abstract away the ideas developed above so they may be more widely applicable. First, let us recall the fundamental technique of Lagrangian relaxation from combinatorial optimization. This technique consists of relaxing a constraint by moving it into the objective function, together with an associated Lagrange multiplier.

Let us apply this relaxation to the constraint, in the k-median IP (25.1), that at most k facilities be opened. Let λ be the Lagrangian multiplier.

$$\text{minimize} \quad \sum_{i \in F,\, j \in C} c_{ij} x_{ij} + \lambda \left(\sum_{i \in F} y_i - k \right) \tag{25.4}$$

$$\text{subject to} \quad \sum_{i \in F} x_{ij} \geq 1, \qquad\qquad j \in C$$

$$y_i - x_{ij} \geq 0, \qquad\qquad i \in F, \ j \in C$$
$$x_{ij} \in \{0, 1\}, \qquad\qquad i \in F, \ j \in C$$
$$y_i \in \{0, 1\}, \qquad\qquad i \in F$$

This is precisely the facility location IP, with the restriction that the cost of each facility is the same, i.e., λ. It contains an additional constant term of $-\lambda k$ in the objective function. We may assume w.l.o.g. that an optimal solution, $(\boldsymbol{x}, \boldsymbol{y})$, to IP (25.1) opens exactly k facilities. Now, $(\boldsymbol{x}, \boldsymbol{y})$ is a feasible solution to IP (25.4) as well, with the same objective function value. Hence, for each value of λ, IP (25.4) is a lower bound on IP (25.1).

We have shown that a Lagrangian relaxation of the k-median problem is the facility location problem. In doing so, the global constraint that at most k facilities be opened has been replaced with a penalty for opening facilities, the penalty being the Lagrangian multiplier. (See Exercise 25.4 for another application of this idea.)

The next important observation was to notice that in the facility location approximation algorithm, Theorem 24.7, the duals pay one-for-one for the cost of opening facilities, i.e., with approximation factor 1. (See Exercise 22.9 for another such algorithm.)

The remaining difficulty was finding a value of λ so that the facility location algorithm opened exactly k facilities. The fact that the facility location algorithm works with the linear relaxation of the problem helped. The convex combination of two (integer) solutions was a feasible (fractional) solution. The last step was rounding this (special) fractional solution into an integral one. For the k-median problem we used randomized rounding (see Exercise 25.4 for a different rounding procedure).

25.5 Exercises

25.1 (Lin and Vitter [188]) Consider the general k-median problem in which the connection costs are not required to satisfy the triangle inequality. Give a reduction from the set cover problem to show that approximating this problem is as hard as approximating set cover, and therefore cannot be done with a factor better than $O(\log n)$ unless $\mathbf{NP} \subseteq \tilde{\mathbf{P}}$.

25.2 Obtain the dual of LP-relaxation to (25.4). (The constant term in the objective function will simply carry over.) How does it relate with the dual of the k-median LP?

25.3 Use the Lagrangian relaxation technique to give a constant factor approximation algorithm for the following common generalization of the facility location and k-median problems. Consider the uncapacitated facility location

problem with the additional constraint that at most k facilities can be opened. This is a common generalization of the two problems solved in this paper: if k is made n_f, we get the first problem, and if the facility costs are set to zero, we get the second problem.

25.4 (Garg [94] and Chudak, Roughgarden, and Williamson [47]) Consider the following variant of the metric Steiner tree problem.

Problem 25.6 (Metric k-MST)   We are given a complete undirected graph $G = (V, E)$, a special vertex $r \in V$, a positive integer k, and a function cost : $E \to \mathbf{Q}^{+}$ satisfying the triangle inequality. The problem is to find a minimum cost tree containing exactly k vertices, including r.

   We will develop a factor 5 algorithm for this problem.

1. Observe that a Lagrangian relaxation of this problem is the prize-collecting Steiner tree problem, Problem 22.12, stated in Exercise 22.9.
2. Observe that the approximation algorithm for the latter problem, given in Exercise 22.9, pays for the penalties one-for-one with the dual, i.e., with an approximation factor of 1.
3. Use the prize-collecting algorithm as a subroutine to obtain two trees, T_1 and T_2, for very close values of the penalty, containing k_1 and k_2 vertices, with $k_1 < k < k_2$. Obtain a convex combination of these solutions, with multipliers α_1 and α_2.
4. We may assume that every vertex in G is at a distance of $\leq$ OPT from r. (Use the idea behind parametric pruning, introduced in Chapter 5. The parameter t is the length of the longest edge used by the optimal solution, which is clearly a lower bound on OPT. For each value of t, instance $G(t)$ is obtained by restricting G to vertices that are within a distance of t of r. The algorithm is run on each graph of this family, and the best tree is output.) Consider the following procedure for rounding the convex combination. If $\alpha_2 \geq 1/2$, then $\text{cost}(T_2) \leq 4 \cdot$ OPT; remove $k_2 - k$ vertices from T_2. Otherwise, double every edge of T_2, find an Euler tour, and shortcut the tour to a cycle containing only those vertices that are in T_2 and not in T_1 (i.e., at most $k_2 - k_1$ vertices). Pick the cheapest path of length $k - k_1 - 1$ from this cycle, and connect it by means of an edge to vertex r in T_1. The resulting tree has exactly k vertices. Show that the cost of this tree is $\leq 5 \cdot$ OPT.
   Hint: Use the fact that $\alpha_2 = (k - k_1)/(k_2 - k_1)$.

25.5 Let us apply the Lagrangian relaxation technique to the following linear program.

$$
\begin{aligned}
\text{minimize} \quad & \boldsymbol{c}^T \boldsymbol{x} & \quad (25.5)\\
\text{subject to} \quad & \boldsymbol{Ax} = \boldsymbol{b}
\end{aligned}
$$

Then the lower bound is given by

$$\max_{\boldsymbol{y}} \min_{\boldsymbol{x}} \left(\boldsymbol{c}^T\boldsymbol{x} - \boldsymbol{y}^T(\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b})\right) = \max_{\boldsymbol{y}} \left(\left(\min_{\boldsymbol{x}} (\boldsymbol{c}^T - \boldsymbol{y}^T\boldsymbol{A})\boldsymbol{x}\right) + \boldsymbol{y}^T\boldsymbol{b}\right)$$

If $\boldsymbol{y}$ does not satisfy $\boldsymbol{A}^T\boldsymbol{y} = \boldsymbol{c}$, then by a suitable choice of $\boldsymbol{x}$, the lower bound given by this expression can be made as small as desired and therefore meaningless. Meaningful lower bounds arise only if we insist that $\boldsymbol{A}^T\boldsymbol{y} = \boldsymbol{c}$. But then we get the following LP:

$$\begin{aligned} \text{maximize} \quad & \boldsymbol{y}^T\boldsymbol{b} && (25.6) \\ \text{subject to} \quad & \boldsymbol{A}^T\boldsymbol{y} = \boldsymbol{c} \end{aligned}$$

Notice that this is the dual of LP (25.5)! Hence, the Lagrangian relaxation of a linear program is simply its dual and is therefore tight.

Obtain the Lagrangian relaxation of the following LP:

$$\begin{aligned} \text{minimize} \quad & \boldsymbol{c}^T\boldsymbol{x} && (25.7) \\ \text{subject to} \quad & \boldsymbol{A}\boldsymbol{x} \geq \boldsymbol{b} \\ & \boldsymbol{x} \geq 0 \end{aligned}$$

25.6 (Jain and Vazirani [141]) Consider the $\boldsymbol{l}_2^2$ *clustering* problem. Given a set of n points $S = \{v_1, \ldots, v_n\}$ in $\mathbf{R}^d$ and a positive integer k, the problem is to find a minimum cost k-clustering, i.e., to find k points, called *centers*, $f_1, \ldots, f_k \in \mathbf{R}^d$, so as to minimize the sum of squares of distances from each point v_i to its closest center. This naturally defines a partitioning of the n points into k clusters. Give a constant factor approximation algorithm for this problem.
Hint: First show that restricting the centers to be a subset S increases the cost of the optimal solution by a factor of at most 2. Apply the solution of Exercise 24.6 to this modified problem.

25.7 (Korupolu, Plaxton, and Rajaraman [176] and Arya et al. [15]) For a set S of k facilities, define $\text{cost}(S)$ to be the total cost of connecting each city to its closest facility in S. Define a *swap* to be the process of replacing one facility in S by a facility from $\overline{S}$. A natural algorithm for metric k-median, based on local search, is: Start with an arbitrary set S of k facilities. In each iteration, check if there is a swap that leads to a lower cost solution. If so, execute any such swap and go to the next iteration. If not, halt. The terminating solution is said to be *locally optimal*.

Let $G = \{o_1, \ldots, o_k\}$ be an optimal solution and $L = \{s_1, \ldots, s_k\}$ be a locally optimal solution. This exercise develops a proof showing $\text{cost}(L) \leq 5 \cdot \text{cost}(G)$, as well as a constant factor approximation algorithm.

1. For $o \in G$, let $N_G(o)$ denote the set of cities connected to facility o in the optimal solution. Similarly, for $s \in L$, let $N_L(s)$ denote the set of cities connected to facility s in the locally optimal solution. Say that $s \in L$ *captures* $o \in G$ if $|N_G(o) \cap N_L(s)| > |N_G(o)|/2$. Clearly, each $o \in G$ is captured by at most one facility in L. In this part let us make the simplifying assumption that each facility $s \in L$ captures a unique facility in G. Assume that the facilities are numbered so that s_i captures o_i, for $1 \leq i \leq k$. Use the fact that for $1 \leq i \leq k$, $\text{cost}(L + o_i - s_i) \geq \text{cost}(L)$ to show that $\text{cost}(L) \leq 3 \cdot \text{cost}(G)$.

   Hint: $\text{cost}(L + o_i - s_i)$ is bounded by the cost of the following solution: The cities in $\overline{N_L(s_i) \cup N_G(o_i)}$ are connected as in the locally optimal solution. Those in $N_G(o_i)$ are connected to facility o_i. Cities in $N_L(s_i) - N_G(o_i)$ are connected to facilities in $L - s_i$ using "3 hops" in such a way that each connecting edge of G and each connecting edge of L is used at most once in the union of all these hops.

2. Show that without the simplifying assumption of the previous part, $\text{cost}(L) \leq 5 \cdot \text{cost}(G)$.

   Hint: Consider k appropriately chosen swaps so that each facility $o \in G$ is swapped in exactly once and each facility $s \in L$ is swapped out at most twice.

3. Strengthen the condition for swapping so as to obtain, for any $\varepsilon > 0$ a factor $5 + \varepsilon$ algorithm running in time polynomial in $1/\varepsilon$ and the size of the instance.

25.6 Notes

The first approximation algorithm, achieving a factor of $O(\log n \log \log n)$, was given by Bartal [21]. The first constant factor approximation algorithm for the k-median problem, achieving a guarantee of $6\frac{2}{3}$, was given by Charikar, Guha, Tardos, and Shmoys [39], using ideas from Lin and Vitter [189]. This algorithm used LP-rounding. The results of this chapter are due to Jain and Vazirani [141]. The current best factor is $3 + 2/p$, with a running time of $O(n^p)$, due to Arya et al. [15]. This is a local search algorithm that swaps p facilities at a time (see Exercise 25.7 for the algorithm for $p = 1$).

The example of Section 25.3.4 is due to Jain, Mahdian, and Saberi [138]. The best upper bound on the integrality gap of relaxation (25.2) is 4, due to Charikar and Guha [38]. For a factor 2 approximation algorithm for the l_2^2 clustering problem (Exercise 25.6), see Drineas, Kannan, Frieze, Vempala, and Vinay [62].