12.7 Best-Response Dynamics and Nash Equilibria

Thus far we have been considering local search as a technique for solving optimization problems with a single objective—in other words, applying local operations to a candidate solution so as to minimize its total cost. There are many settings, however, where a potentially large number of agents, each with its own goals and objectives, collectively interact so as to produce a solution to some problem. A solution that is produced under these circumstances often reflects the "tug-of-war" that led to it, with each agent trying to pull the solution in a direction that is favorable to it. We will see that these interactions can be viewed as a kind of local search procedure; analogues of local minima have a natural meaning as well, but having multiple agents and multiple objectives introduces new challenges.

The field of game theory provides a natural framework in which to talk about what happens in such situations, when a collection of agents interacts strategically—in other words, with each trying to optimize an individual objective function. To illustrate these issues, we consider a concrete application, motivated by the problem of routing in networks; along the way, we will introduce some notions that occupy central positions in the area of game theory more generally.

The Problem

In a network like the Internet, one frequently encounters situations in which a number of nodes all want to establish a connection to a single *source node* s. For example, the source s may be generating some kind of data stream that all the given nodes want to receive, as in a style of one-to-many network communication known as *multicast*. We will model this situation by representing the underlying network as a directed graph $G = (V, E)$, with a cost $c_e \geq 0$ on each edge. There is a designated source node $s \in V$ and a collection of k agents located at distinct *terminal nodes* $t_1, t_2, \ldots, t_k \in V$. For simplicity, we will not make a distinction between the agents and the nodes at which they reside; in other words, we will think of the agents as being $t_1, t_2, \ldots, t_k$. Each agent t_j wants to construct a path P_j from s to t_j using as little total cost as possible.

Now, if there were no interaction among the agents, this would consist of k separate shortest-path problems: Each agent t_j would find an s-t_j path for which the total cost of all edges is minimized, and use this as its path P_j. What makes this problem interesting is the prospect of agents being able to *share* the costs of edges. Suppose that after all the agents have chosen their paths, agent t_j only needs to pay its "fair share" of the cost of each edge e on its path; that is, rather than paying c_e for each e on P_i, it pays c_e divided by the number of agents whose paths contain e. In this way, there is an incentive for the agents to choose paths that overlap, since they can then benefit by splitting the costs of edges. (This sharing model is appropriate for settings in which the presence of multiple agents on an edge does not significantly degrade the quality of transmission due to congestion or increased latency. If latency effects do come into play, then there is a countervailing penalty for sharing; this too leads to interesting algorithmic questions, but we will stick to our current focus for now, in which sharing comes with benefits only.)

Best-Response Dynamics and Nash Equilibria: Definitions and Examples

To see how the option of sharing affects the behavior of the agents, let's begin by considering the pair of very simple examples in Figure 12.8. In example (a), each of the two agents has two options for constructing a path: the middle route through v, and the outer route using a single edge. Suppose that each agent starts out with an initial path but is continually evaluating the current situation to decide whether it's possible to switch to a better path.

In example (a), suppose the two agents start out using their outer paths. Then t_1 sees no advantage in switching paths (since $4 < 5 + 1$), but t_2 does (since $8 > 5 + 1$), and so t_2 updates its path by moving to the middle. Once this happens, things have changed from the perspective of t_1: There is suddenly an advantage for t_1 in switching as well, since it now gets to share the cost of the middle path, and hence its cost to use the middle path becomes $2.5 + 1 < 4$. Thus it will switch to the middle path. Once we are in a situation where both

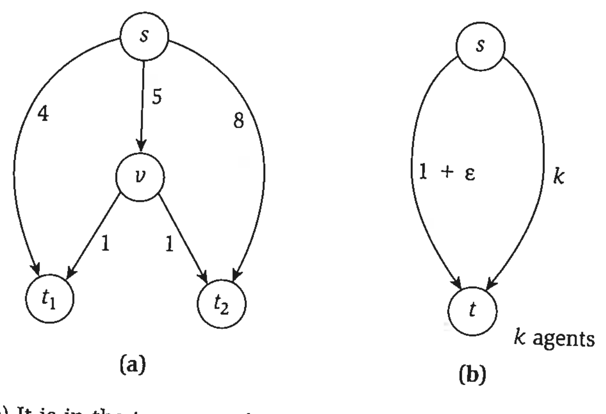

Figure 12.8 (a) It is in the two agents' interest to share the middle path. (b) It would be better for all the agents to share the edge on the left. But if all k agents start on the right-hand edge, then no one of them will want to unilaterally move from right to left; in other words, the solution in which all agents share the edge on the right is a bad Nash equilibrium.

sides are using the middle path, neither has an incentive to switch, and so this is a stable solution.

Let's discuss two definitions from the area of game theory that capture what's going on in this simple example. While we will continue to focus on our particular multicast routing problem, these definitions are relevant to any setting in which multiple agents, each with an individual objective, interact to produce a collective solution. As such, we will phrase the definitions in these general terms.

- First of all, in the example, each agent was continually prepared to improve its solution in response to changes made by the other agent(s). We will refer to this process as *best-response dynamics*. In other words, we are interested in the dynamic behavior of a process in which each agent updates based on its best response to the current situation.
- Second, we are particularly interested in stable solutions, where the best response of each agent is to stay put. We will refer to such a solution, from which no agent has an incentive to deviate, as a *Nash equilibrium*. (This is named after the mathematician John Nash, who won the Nobel Prize in economics for his pioneering work on this concept.) Hence, in example (a), the solution in which both agents use the middle path is a Nash equilibrium. Note that the Nash equilibria are precisely the solutions at which best-response dynamics terminate.

The example in Figure 12.8(b) illustrates the possibility of multiple Nash equilibria. In this example, there are k agents that all reside at a common node t (that is, $t_1 = t_2 = \cdots = t_k = t$), and there are two parallel edges from s to t with different costs. The solution in which all agents use the left-hand edge is a Nash equilibrium in which all agents pay $(1+\varepsilon)/k$. The solution in which all agents use the right-hand edge is also a Nash equilibrium, though here the agents each pay $k/k = 1$. The fact that this latter solution is a Nash equilibrium exposes an important point about best-response dynamics. If the agents could somehow synchronously agree to move from the right-hand edge to the left-hand one, they'd all be better off. But under best-response dynamics, each agent is only evaluating the consequences of a unilateral move by itself. In effect, an agent isn't able to make any assumptions about future actions of other agents—in an Internet setting, it may not even know anything about these other agents or their current solutions—and so it is only willing to perform updates that lead to an immediate improvement for itself.

To quantify the sense in which one of the Nash equilibria in Figure 12.8(b) is better than the other, it is useful to introduce one further definition. We say that a solution is a *social optimum* if it minimizes the total cost to all agents. We can think of such a solution as the one that would be imposed by a benevolent central authority that viewed all agents as equally important and hence evaluated the quality of a solution by summing the costs they incurred. Note that in both (a) and (b), there is a social optimum that is also a Nash equilibrium, although in (b) there is also a second Nash equilibrium whose cost is much greater.

The Relationship to Local Search

Around here, the connections to local search start to come into focus. A set of agents following best-response dynamics are engaged in some kind of gradient descent process, exploring the "landscape" of possible solutions as they try to minimize their individual costs. The Nash equilibria are the natural analogues of local minima in this process: solutions from which no improving move is possible. And the "local" nature of the search is clear as well, since agents are only updating their solutions when it leads to an immediate improvement.

Having said all this, it's important to think a bit further and notice the crucial ways in which this differs from standard local search. In the beginning of this chapter, it was easy to argue that the gradient descent algorithm for a combinatorial problem must terminate at a local minimum: each update decreased the cost of the solution, and since there were only finitely many possible solutions, the sequence of updates could not go on forever. In other words, the cost function itself provided the progress measure we needed to establish termination.

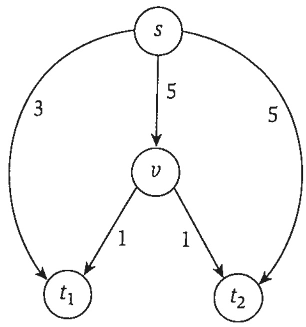

Figure 12.9 A network in which the unique Nash equilibrium differs from the social optimum.

In best-response dynamics, on the other hand, each agent has its own personal objective function to minimize, and so it's not clear what overall "progress" is being made when, for example, agent t_i decides to update its path from s. There's progress for t_i, of course, since its cost goes down, but this may be offset by an even larger increase in the cost to some other agent. Consider, for example, the network in Figure 12.9. If both agents start on the middle path, then t_1 will in fact have an incentive to move to the outer path; its cost drops from 3.5 to 3, but in the process the cost of t_2 increases from 3.5 to 6. (Once this happens, t_2 will also move to its outer path, and this solution—with both nodes on the outer paths—is the unique Nash equilibrium.)

There are examples, in fact, where the cost-increasing effects of best-response dynamics can be much worse than this. Consider the situation in Figure 12.10, where we have k agents that each have the option to take a common outer path of cost $1+\varepsilon$ (for some small number $\varepsilon > 0$), or to take their own alternate path. The alternate path for t_j has cost $1/j$. Now suppose we start with a solution in which all agents are sharing the outer path. Each agent pays $(1+\varepsilon)/k$, and this is the solution that minimizes the total cost to all agents. But running best-response dynamics starting from this solution causes things to unwind rapidly. First t_k switches to its alternate path, since $1/k < (1+\varepsilon)/k$.

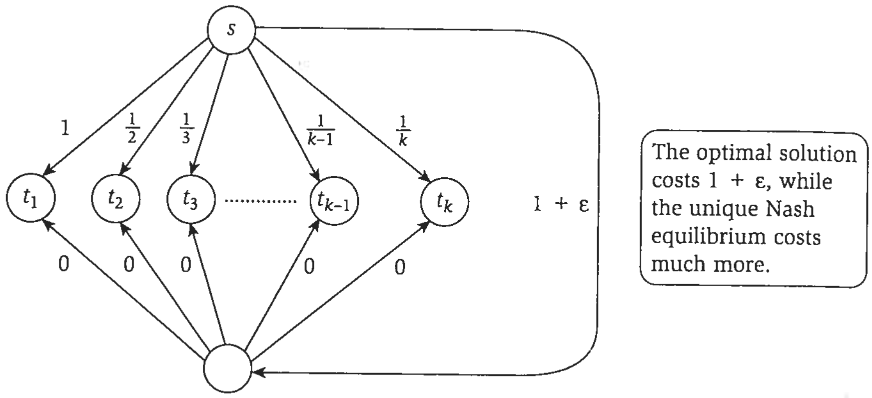

Figure 12.10 A network in which the unique Nash equilibrium costs $H(k) = \Theta(\log k)$ times more than the social optimum.

As a result of this, there are now only $k-1$ agents sharing the outer path, and so t_{k-1} switches to its alternate path, since $1/(k-1) < (1+\varepsilon)/(k-1)$. After this, t_{k-2} switches, then t_{k-3}, and so forth, until all k agents are using the alternate paths directly from s. Things come to a halt here, due to the following fact.

(12.13) *The solution in Figure 12.10, in which each agent uses its direct path from s, is a Nash equilibrium, and moreover it is the unique Nash equilibrium for this instance.*

Proof. To verify that the given solution is a Nash equilibrium, we simply need to check that no agent has an incentive to switch from its current path. But this is clear, since all agents are paying at most 1, and the only other option—the (currently vacant) outer path—has cost $1+\varepsilon$.

Now suppose there were some other Nash equilibrium. In order to be different from the solution we have just been considering, it would have to involve at least one of the agents using the outer path. Let $t_{j_1}, t_{j_2}, \ldots, t_{j_\ell}$ be the agents using the outer path, where $j_1 < j_2 < \cdots < j_\ell$. Then all these agents are paying $(1+\varepsilon)/\ell$. But notice that $j_\ell \geq \ell$, and so agent t_{j_ℓ} has the option to pay only $1/j_\ell \leq 1/\ell$ by using its alternate path directly from s. Hence t_{j_ℓ} has an incentive to deviate from the current solution, and hence this solution cannot be a Nash equilibrium. ∎

Figure 12.8(b) already illustrated that there can exist a Nash equilibrium whose total cost is much worse than that of the social optimum, but the examples in Figures 12.9 and 12.10 drive home a further point: The total cost to all agents under even the *most favorable* Nash equilibrium solution can be worse than the total cost under the social optimum. How much worse? The total cost of the social optimum in this example is $1+\varepsilon$, while the cost of the unique Nash equilibrium is $1+\frac{1}{2}+\frac{1}{3}+\cdots+\frac{1}{k}=\sum_{i=1}^{k}\frac{1}{i}$. We encountered this expression in Chapter 11, where we defined it to be the *harmonic number* $H(k)$ and showed that its asymptotic value is $H(k)=\Theta(\log k)$.

These examples suggest that one can't really view the social optimum as the analogue of the global minimum in a traditional local search procedure. In standard local search, the global minimum is always a stable solution, since no improvement is possible. Here the social optimum can be an unstable solution, since it just requires one agent to have an interest in deviating.

Two Basic Questions

Best-response dynamics can exhibit a variety of different behaviors, and we've just seen a range of examples that illustrate different phenomena. It's useful at this point to step back, assess our current understanding, and ask some basic questions. We group these questions around the following two issues.

- **The existence of a Nash equilibrium.** At this point, we actually don't have a proof that there even *exists* a Nash equilibrium solution in every instance of our multicast routing problem. The most natural candidate for a progress measure, the total cost to all agents, does not necessarily decrease when a single agent updates its path.

  Given this, it's not immediately clear how to argue that the best-response dynamics must terminate. Why couldn't we get into a cycle where agent t_1 improves its solution at the expense of t_2, then t_2 improves its solution at the expense of t_1, and we continue this way forever? Indeed, it's not hard to define other problems in which exactly this can happen and in which Nash equilibria don't exist. So if we want to argue that best-response dynamics leads to a Nash equilibrium in the present case, we need to figure out what's special about our routing problem that causes this to happen.

- **The price of stability.** So far we've mainly considered Nash equilibria in the role of "observers": essentially, we turn the agents loose on the graph from an arbitrary starting point and watch what they do. But if we were viewing this as protocol designers, trying to define a procedure by which agents could construct paths from s, we might want to pursue the following approach. Given a set of agents, located at nodes $t_1, t_2, \ldots, t_k$, we could propose a collection of paths, one for each agent, with two properties.

  (i) The set of paths forms a Nash equilibrium solution; and

  (ii) Subject to (i), the total cost to all agents is as small as possible.

Of course, ideally we'd like just to have the smallest total cost, as this is the social optimum. But if we propose the social optimum and it's not a Nash equilibrium, then it won't be stable: Agents will begin deviating and constructing new paths. Thus properties (i) and (ii) together represent our protocol's attempt to optimize in the face of stability, finding the best solution from which no agent will want to deviate.

We therefore define the *price of stability*, for a given instance of the problem, to be the ratio of the cost of the best Nash equilibrium solution to the cost of the social optimum. This quantity reflects the blow-up in cost that we incur due to the requirement that our solution must be stable in the face of the agents' self-interest.

Note that this pair of questions can be asked for essentially any problem in which self-interested agents produce a collective solution. For our multicast routing problem, we now resolve both these questions. Essentially, we will find that the example in Figure 12.10 captures some of the crucial aspects of the problem in general. We will show that for any instance, best-response dynamics starting from the social optimum leads to a Nash equilibrium whose cost is greater by at most a factor of $H(k) = \Theta(\log k)$.

Finding a Good Nash Equilibrium

We focus first on showing that best-response dynamics in our problem always terminates with a Nash equilibrium. It will turn out that our approach to this question also provides the necessary technique for bounding the price of stability.

The key idea is that we don't need to use the total cost to all agents as the progress measure against which to bound the number of steps of best-response dynamics. Rather, any quantity that strictly decreases on a path update by any agent, and which can only decrease a finite number of times, will work perfectly well. With this in mind, we try to formulate a measure that has this property. The measure will not necessarily have as strong an intuitive meaning as the total cost, but this is fine as long as it does what we need.

We first consider in more detail why just using the total agent cost doesn't work. Suppose, to take a simple example, that agent t_j is currently sharing, with x other agents, a path consisting of the single edge e. (In general, of course, the agents' paths will be longer than this, but single-edge paths are useful to think about for this example.) Now suppose that t_j decides it is in fact cheaper to switch to a path consisting of the single edge f, which no agent is currently using. In order for this to be the case, it must be that $c_f < c_e/(x + 1)$. Now, as a result of this switch, the total cost to all agents goes up by c_f: Previously,

$x+1$ agents contributed to the cost c_e, and no one was incurring the cost c_f; but, after the switch, x agents still collectively have to pay the full cost c_e, and t_j is now paying an additional c_f.

In order to view this as progress, we need to redefine what "progress" means. In particular, it would be useful to have a measure that could offset the added cost c_f via some notion that the overall "potential energy" in the system has dropped by $c_e/(x+1)$. This would allow us to view the move by t_j as causing a net decrease, since we have $c_f < c_e/(x+1)$. In order to do this, we could maintain a "potential" on each edge e, with the property that this potential drops by $c_e/(x+1)$ when the number of agents using e decreases from $x+1$ to x. (Correspondingly, it would need to increase by this much when the number of agents using e increased from x to $x+1$.)

Thus, our intuition suggests that we should define the potential so that, if there are x agents on an edge e, then the potential should decrease by c_e/x when the first one stops using e, by $c_e/(x-1)$ when the next one stops using e, by $c_e/(x-2)$ for the next one, and so forth. Setting the potential to be $c_e(1/x + 1/(x-1) + \cdots + 1/2 + 1) = c_e \cdot H(x)$ is a simple way to accomplish this. More concretely, we define the *potential* of a set of paths $P_1, P_2, \ldots, P_k$, denoted $\Phi(P_1, P_2, \ldots, P_k)$, as follows. For each edge e, let x_e denote the number of agents whose paths use the edge e. Then

$$\Phi(P_1, P_2, \ldots, P_k) = \sum_{e\in E} c_e \cdot H(x_e).$$

(We'll define the harmonic number $H(0)$ to be 0, so that the contribution of edges containing no paths is 0.)

The following claim establishes that Φ really works as a progress measure.

(12.14) *Suppose that the current set of paths is $P_1, P_2, \ldots, P_k$, and agent t_j updates its path from P_j to P_j'. Then the new potential $\Phi(P_1, \ldots, P_{j-1}, P_j', P_{j+1}, \ldots, P_k)$ is strictly less than the old potential $\Phi(P_1, \ldots, P_{j-1}, P_j, P_{j+1}, \ldots, P_k)$.*

Proof. Before t_j switched its path from P_j to P_j', it was paying $\sum_{e\in P_j} c_e/x_e$, since it was sharing the cost of each edge e with $x_e - 1$ other agents. After the switch, it continues to pay this cost on the edges in the intersection $P_j \cap P_j'$, and it also pays $c_f/(x_f+1)$ on each edge $f \in P_j' - P_j$. Thus the fact that t_j viewed this switch as an improvement means that

$$\sum_{f\in P_j'-P_j} \frac{c_f}{x_f+1} < \sum_{e\in P_j-P_j'} \frac{c_e}{x_e}.$$

Now let's ask what happens to the potential function Φ. The only edges on which it changes are those in $P'_j - P_j$ and those in $P_j - P'_j$. On the former set, it increases by

$$\sum_{f \in P'_j - P_j} c_f [H(x_f + 1) - H(x_f)] = \sum_{f \in P'_j - P_j} \frac{c_f}{x_f + 1},$$

and on the latter set, it decreases by

$$\sum_{e \in P_j - P'_j} c_e [H(x_e) - H(x_e - 1)] = \sum_{e \in P_j - P'_j} \frac{c_e}{x_e}.$$

So the criterion that t_j used for switching paths is precisely the statement that the total increase is strictly less than the total decrease, and hence the potential Φ decreases as a result of t_j's switch. ■

Now there are only finitely many ways to choose a path for each agent t_j, and (12.14) says that best-response dynamics can never revisit a set of paths $P_1, \ldots, P_k$ once it leaves it due to an improving move by some agent. Thus we have shown the following.

(12.15) *Best-response dynamics always leads to a set of paths that forms a Nash equilibrium solution.*

Bounding the Price of Stability Our potential function Φ also turns out to be very useful in providing a bound on the price of stability. The point is that, although Φ is not equal to the total cost incurred by all agents, it tracks it reasonably closely.

To see this, let $C(P_1, \ldots, P_k)$ denote the total cost to all agents when the selected paths are $P_1, \ldots, P_k$. This quantity is simply the sum of c_e over all edges that appear in the union of these paths, since the cost of each such edge is completely covered by the agents whose paths contain it.

Now the relationship between the cost function C and the potential function Φ is as follows.

(12.16) *For any set of paths $P_1, \ldots, P_k$, we have*

$$C(P_1, \ldots, P_k) \leq \Phi(P_1, \ldots, P_k) \leq H(k) \cdot C(P_1, \ldots, P_k).$$

Proof. Recall our notation in which x_e denotes the number of paths containing edge e. For the purposes of comparing C and Φ, we also define E^+ to be the set of all edges that belong to at least one of the paths $P_1, \ldots, P_k$. Then, by the definition of C, we have $C(P_1, \ldots, P_k) = \sum_{e \in E^+} c_e$.

A simple fact to notice is that $x_e \leq k$ for all e. Now we simply write

$$C(P_1, \ldots, P_k) = \sum_{e \in E^+} c_e \leq \sum_{e \in E^+} c_e H(x_e) = \Phi(P_1, \ldots, P_k)$$

and

$$\Phi(P_1, \ldots, P_k) = \sum_{e \in E^+} c_e H(x_e) \leq \sum_{e \in E^+} c_e H(k) = H(k) \cdot C(P_1, \ldots, P_k). \quad \blacksquare$$

Using this, we can give a bound on the price of stability.

(12.17) *In every instance, there is a Nash equilibrium solution for which the total cost to all agents exceeds that of the social optimum by at most a factor of $H(k)$.*

Proof. To produce the desired Nash equilibrium, we start from a social optimum consisting of paths $P_1^*, \ldots, P_k^*$ and run best-response dynamics. By (12.15), this must terminate at a Nash equilibrium $P_1, \ldots, P_k$.

During this run of best-response dynamics, the total cost to all agents may have been going up, but by (12.14) the potential function was decreasing. Thus we have $\Phi(P_1, \ldots, P_k) \leq \Phi(P_1^*, \ldots, P_k^*)$.

This is basically all we need since, for any set of paths, the quantities C and Φ differ by at most a factor of $H(k)$. Specifically,

$$C(P_1, \ldots, P_k) \leq \Phi(P_1, \ldots, P_k) \leq \Phi(P_1^*, \ldots, P_k^*) \leq H(k) \cdot C(P_1^*, \ldots, P_k^*). \quad \blacksquare$$

Thus we have shown that a Nash equilibrium always exists, and there is always a Nash equilibrium whose total cost is within an $H(k)$ factor of the social optimum. The example in Figure 12.10 shows that it isn't possible to improve on the bound of $H(k)$ in the worst case.

Although this wraps up certain aspects of the problem very neatly, there are a number of questions here for which the answer isn't known. One particularly intriguing question is whether it's possible to construct a Nash equilibrium for this problem in polynomial time. Note that our proof of the existence of a Nash equilibrium argued simply that as best-response dynamics iterated through sets of paths, it could never revisit the same set twice, and hence it could not run forever. But there are exponentially many possible sets of paths, and so this does not give a polynomial-time algorithm. Beyond the question of finding any Nash equilibrium efficiently, there is also the open question of efficiently finding a Nash equilibrium that achieves a bound of $H(k)$ relative to the social optimum, as guaranteed by (12.17).

It's also important to reiterate something that we mentioned earlier: It's not hard to find problems for which best-response dynamics may cycle forever and for which Nash equilibria do not necessarily exist. We were fortunate here that best-response dynamics could be viewed as iteratively improving a *potential function* that guaranteed our progress toward a Nash equilibrium, but the point is that potential functions like this do not exist for all problems in which agents interact.

Finally, it's interesting to compare what we've been doing here to a problem that we considered earlier in this chapter: finding a stable configuration in a Hopfield network. If you recall the discussion of that earlier problem, we analyzed a process in which each node "flips" between two possible states, seeking to increase the total weight of "good" edges incident to it. This can in fact be viewed as an instance of best-response dynamics for a problem in which each node has an objective function that seeks to maximize this measure of good edge weight. However, showing the convergence of best-response dynamics for the Hopfield network problem was much easier than the challenge we faced here: There it turned out that the state-flipping process was in fact a "disguised" form of local search with an objective function obtained simply by adding together the objective functions of all nodes—in effect, the analogue of the total cost to all agents served as a progress measure. In the present case, it was precisely because this total cost function did not work as a progress measure that we were forced to embark on the more complex analysis described here.

Solved Exercises

Solved Exercise 1

The Center Selection Problem from Chapter 11 is another case in which one can study the performance of local search algorithms.

Here is a simple local search approach to Center Selection (indeed, it's a common strategy for a variety of problems that involve locating facilities). In this problem, we are given a set of sites $S = \{s_1, s_2, \ldots, s_n\}$ in the plane, and we want to choose a set of k centers $C = \{c_1, c_2, \ldots, c_k\}$ whose *covering radius*—the farthest that people in any one site must travel to their nearest center—is as small as possible.

We start by arbitrarily choosing k points in the plane to be the centers $c_1, c_2, \ldots, c_k$. We now alternate the following two steps.

(i) Given the set of k centers $c_1, c_2, \ldots, c_k$, we divide S into k sets: For $i = 1, 2, \ldots, k$, we define S_i to be the set of all the sites for which c_i is the closest center.

(ii) Given this division of S into k sets, construct new centers that will be as "central" as possible relative to them. For each set S_i, we find the smallest