

CHAPTERS 5 & 6: EXTENSIVE GAMES WITH PERFECT INFORMATION

Directed Graphs and Trees

Osborne defines an extensive game in terms of sequences of actions and terminal histories. These actions are represented in the book in terms of trees. Therefore, we start by defining directed graphs and trees. Before giving the formal definition, we consider two examples of making a sequence of choices (but not a game).

Example 1. This example involves the choice of making and marketing a wine. The first choice is the type of wine, generic or estate. The second choice is the number of years of aging, 1 or 2 for a generic wine and 5 or 10 years for an estate wine. The third choice is the extent of advertising, aggressive or minimal. For each set of choices, there is a payoff in terms of the profit. These possible choices and payoffs can be represented in terms of a tree. See Figure ?? . The tree starts at a common vertex and branches to the right as the decisions are made. A set of choices results in a path in the tree from the starting point or root, to one of the terminal vertices with its payoff. ■

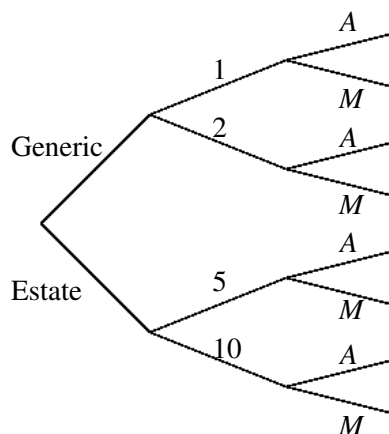


FIGURE 1. Tree for Example ??

Example 2. Most of our extensive games are represented by trees. However, sometimes we consider a directed graph that has more than one path from the starting vertex or root to another vertex. Consider the directed graph in Figure ?? . The payoffs at a terminal vertex can depend on the path from root to that vertex and not just the vertex. Thus, the path $e_1e_4e_7$ can have different payoff than the path $e_2e_5e_7$, even though they both end at vertex v_2 . ■

Example 3. The last example is a tree that has an infinite number of vertices. See Figure ?? . A terminal path can be finite like $e_1, w_1e_2, \dots, w_1 \cdots w_{n-1}e_n$, or the infinite path $w_1 \cdots w_n \cdots$. ■

In our figures, the directed graphs will usually go forward to the right or downward. We do not usually mark the graphs with arrows to indicate a direction.

Next, we give the formal definitions.

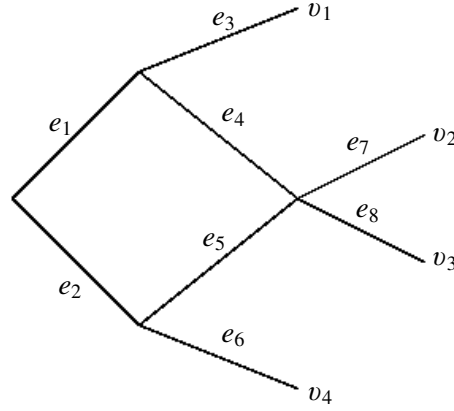


FIGURE 2. Decision graph for Example ??

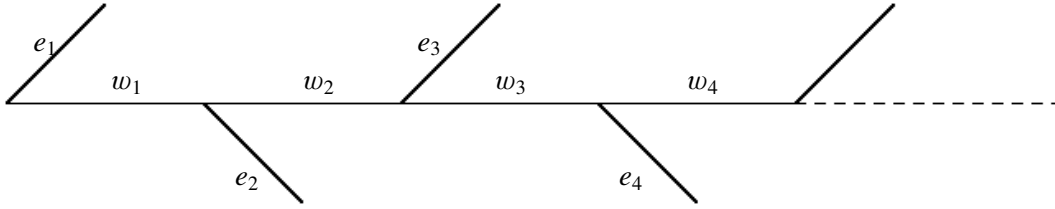


FIGURE 3. Infinite tree for Example ??

Definition. A *directed graph* $\mathcal{G}$ is specified by giving a set of *vertices* (or *nodes*) $\mathbf{V}$ and a set of edges $\mathbf{E}$ where each edge e is an ordered pair of vertices, $e = (v_1, v_2)$ which is the edge from v_1 to v_2 . We designate the beginning vertex of the edge by $b(e) = v_1$ and the ending or terminal vertex of the edge by $t(e) = v_2$.

Definition. A *path* in a direction graph $\mathcal{G}$ is a sequence of edges $\{e_i\}_{i=1}^K$, which can be finite or infinite, such that for each i , the edge e_{i+1} begins at the vertex at which e_i ends, $b(e_{i+1}) = t(e_i)$. For a finite path, we say that $\{e_i\}_{i=1}^K$ is a path from $b(e_1)$ to $t(e_K)$.

A *cycle* is a nonempty finite path $\{e_i\}_{i=1}^K$ from a vertex to itself, $b(e_1) = t(e_K)$.

Definition. For a directed graph, a vertex r is called a *root* provided that

1. there is no edge going into r , $t(e) \neq r$ for all edges e , and
2. for any vertex v , there is at least one finite path from r to v .

We will call a directed graph that has a root and no cycles a *game graph*. (Although this term is not standard, it makes statements easier so we use it.)

A *game tree* or *rooted tree* is a game graph with root r such that for every other vertex v of the graph, there is exactly one finite path from r to v .

Definition. A game graph or tree is said to have a *finite horizon* provided that there is finite bound K such that every path has length less than or equal to K . A game graph or tree has *infinite horizon* provided that it does not have finite horizon. The game tree for Example ?? given in Figure ?? has an infinite horizon.

The tree or game graph is said to be *finite* provided that there are finitely many vertices (and edges). A finite tree always has finite horizon, but a finite horizon tree can have infinitely many vertices (and edges)

because there can be infinitely many edges coming out of a single vertex. We will give examples of this later.

In the examples above, Example ?? is a finite tree, Example ?? is an infinite tree, and Example ?? is a game graph but not a tree. For extensive games, we sometimes consider game graphs that are not trees, because they can be smaller than the corresponding tree.

Definition. We note the correspondences of the terms given in Osborne and terms as given for game trees or graphs. An *action* corresponds to edge of the game tree. A *history* corresponds to a finite or infinite path, $\{e_i\}_{i=1}^k$ or $\{e_i\}_{i=1}^\infty$ in the game tree. A vertex v in a tree determines a unique finite history starting at the root, $b(e_1) = r$, and ending at v , $t(e_k) = v$. The root r corresponds to the empty history $\emptyset$ (no edge). For a game graph, more than one path or history starting at the root can end at the same vertex.

For a finite history $\{e_i\}_{i=1}^k$, a *subhistory* is either (i) the empty history $\emptyset$ or (ii) $\{e_i\}_{i=1}^m$ with $1 \leq m \leq k$. For an infinite history $\{e_i\}_{i=1}^\infty$, a *subhistory* is either (i) the empty history $\emptyset$, (ii) $\{e_i\}_{i=1}^m$ with $1 \leq m < \infty$, or (iii) the entire history $\{e_i\}_{i=1}^\infty$. A *proper subhistory* is a subhistory that does not equal the entire original history. The empty history has no proper subhistory.

Definition. For a game tree or graph, a *terminal vertex* is a vertex v for which there is no edge coming out of v , i.e., $b(e) \neq v$ for all edges e . We designate the set of terminal vertices by $\mathbf{V}_t$ and the set of nonterminal vertices by $\mathbf{V}_n$.

For a game graph or an infinite horizon tree, we need to consider histories rather than vertices to determine where the path ends. A *terminal history* is a (finite or infinite) history $\{e_i\}_{i=1}^k$ that is not a proper subhistory of any other history. For a finite horizon game graph, a terminal history is a path that starts at the root and ends in a terminal vertex. For an infinite horizon tree or game graph, we allow paths that start at the root and stretch off to ‘infinity’. We designate the set of terminal histories starting at the root by $\mathcal{H}_t$ and the set of nonterminal histories by $\mathcal{H}_n$.

§5.1 Extensive games

For an extensive game, there is a game tree (or game graph) with payoffs given for each player at all the terminal vertices (or terminal histories).

Definition. A *extensive game with perfect information* is prescribed as follows: There are a set of players $\mathcal{P}$ and either a game tree or a game graph $\mathcal{T}$ that satisfy the following.

If $\mathcal{T}$ is a game tree with finite horizon, then the following hold:

1. There is a *player function* P that assigns a player $P(v)$ to each nonterminal vertex v , $P : \mathbf{V}_n \rightarrow \mathcal{P}$. We say that the nonterminal vertex v of the tree is *owned* by the player $P(v)$. The player who owns a vertex v needs to decide which edge to take out of v .
2. For each terminal vertex v , there is has a payoff vector $\mathbf{u}(v) = (u_1(v), \dots, u_n(v))$, where $u_i(v)$ is the payoff for the i^{th} -player, $\mathbf{u} : \mathbf{V}_t \rightarrow \mathbb{R}^n$.

If $\mathcal{T}$ is a game tree with infinite horizon or a game graph, then the following hold:

1. There is a *player function* P that assigns a player $P(h)$ to each nonterminal history h , $P : \mathbf{H}_n \rightarrow \mathcal{P}$. We say that the nonterminal history h of the tree is *owned* by the player $P(h)$. The player who owns a history h needs to decide which edge to take out of $t(h)$.
2. For each terminal history $\mathbf{h}$, there is a payoff vector $(u_1(\mathbf{h}), \dots, u_n(\mathbf{h}))$, where $u_i(\mathbf{h})$ is the payoff for the i^{th} -player, $\mathbf{u} : \mathcal{H}_t \rightarrow \mathbb{R}^n$.

Example 4. The extensive game in Figure ?? has two players. Player P_1 owns the root vertex r and player P_2 owns vertices v_1 and v_2 . The vertices v_3, v_4, v_5 , and v_6 are terminal with the payoffs indicated in the figure by the vector $(u_1(v_i), u_2(v_i))$. ■

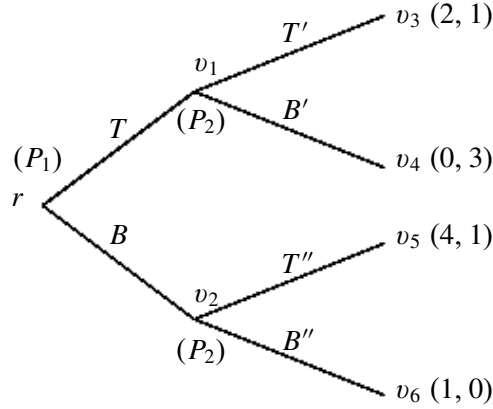


FIGURE 4. Tree for Example ??

§§5.2–5.3 Strategies and Nash equilibria

Definition. A *strategy* for a player for an extensive game is a choice of an outgoing edge at every vertex (or history) owned by that player. A *strategic profile* is a choice of a strategy for each player. Thus, a strategic profile is not just a choice of a path through the game graph or tree, but a choice at every nonterminal vertex (or history) of the edge going out from that vertex.

Example (??, continued). Consider the tree given in Figure ??. (Compare with Figure 159.1.) Player 1 owns the root, r and player 2 owns the two vertices v_1 and v_2 . A strategy by player 1, is just a choice of the edge T or B at r , $s_1 = T$ or B . However, player 2 must make a choice of an edge at both v_1 and v_2 no matter which choice is made by player 1 at r . Thus, player 2 has four possible strategies, $s_2 = T'T'', T'B'', B'T'',$ or $B'B''$. Combining the strategies by the two players, there are eight strategy profiles, giving all the combinations. We indicate the strategy profile by first giving the strategy for player 1 and then for player 2, e.g., $s = (B, T'B'')$.

Specifying a strategy profile determines a path through the tree and payoffs. The book indicates the terminal vertex determined by a strategy profile s (or last edge of the path) by $\mathcal{O}(s)$. The payoffs are $u_1(\mathcal{O}(s))$ and $u_2(\mathcal{O}(s))$. Thus, in the example under consideration, the strategy profile $(B, T'B'')$ determines the history (path) BB'' resulting in the terminal vertex $\mathcal{O}(B, T'B'') = v_6$ with payoffs $u_1(\mathcal{O}(B, T'B'')) = 1$ and $u_2(\mathcal{O}(B, T'B'')) = 0$, and the strategy profile $(B, B'T'')$ determines the history BT'' resulting in the terminal vertex $\mathcal{O}(B, B'T'') = v_5$ with payoffs $u_1(\mathcal{O}(B, B'T'')) = 4$ and $u_2(\mathcal{O}(B, B'T'')) = 1$. ■

Notice, that a terminal history (path) starting at the root is not enough to determine a strategy profile. A strategy profile specifies the choices at all nonterminal vertices and not just those on the path determined by the strategy.

Definition. Consider an extensive game with n players and payoffs u_i on terminal vertices or terminal histories. A strategy profile $s = (s_1, \dots, s_n)$ determines a terminal history $\mathcal{O}(s)$ (that leads from the root to a terminal vertex for a finite horizon game.) The payoffs for the players are $u_i(\mathcal{O}(s))$ for $i = 1, \dots, n$.

Once we have determined the payoffs from a strategy profile, the definition of a Nash equilibrium for an extensive game is similar to the definition for a strategic game.

Definition. Consider an extensive game with n players and payoffs u_i on terminal vertices or terminal histories. A strategy profile s^* is a *Nash equilibrium* provided that, for each $1 \leq i \leq n$ and each strategy s_i for the i^{th} -player,

$$u_i(\mathcal{O}(s^*)) \geq u_i(\mathcal{O}(s_i, s_{-i}^*)).$$

Example 5. Consider the game in Figure ?? for Example ??. We describe a process for determining a Nash equilibrium.

At vertex v_1 , a choice of T' has a payoff of 1 for player P_2 and a choice of B' has a payoff of 3. Thus, B' is the better choice for player P_2 .

At vertex v_2 , a choice of T'' has a payoff of 1 for player P_2 and a choice of B'' has a payoff of 0. Thus, T'' is the better choice for player P_2 .

Assuming that player P_2 has chosen strategy $B'T''$, at r a choice of T by player P_1 yields a payoff of 0 for player P_1 and a choice of B yields a payoff of 1. Thus, B is the better choice for player P_1 .

We claim that the strategy profile $(B, B'T'')$ is a Nash equilibrium.

For $B'T''$ fixed, P_1 maximizes $u_1(e, B'T'')$ by letting $e = B$. For B fixed, P_2 maximizes $u_2(B, s_2)$ by picking T'' and either T' or B' . Thus, $(B, B'T'')$ is a Nash equilibrium.

The above construction not only produced a Nash equilibrium, but also the best choice starting at any vertex, even those vertices that are not reached by the optimal strategy profile. Such a Nash equilibrium is called a subgame perfect equilibrium. (See the formal definition below.)

There is another Nash equilibrium for this game that is not a subgame perfect equilibrium. Consider the strategy profile $(B, T'T'')$. This has the same payoffs as the Nash equilibrium above. The change of choice of edge coming out of v_1 goes not change the path determined. Player P_1 does not have any incentive to change because $u_1(B, T'T'') = 4 > 2 = u_1(T, T'T'')$. Player P_2 does not have any incentive to change because $u_2(B, T'T'') = 1 > u_2(B, *B'')$. This shows that $(B, T'T'')$ is a Nash equilibrium. It is not subgame perfect, because player P_2 is not making the choice of the edge starting at v_1 that produces the largest payoff: this strategy is not a Nash equilibrium for the subgame starting at v_1 . ■

The solution process in the last example that constructed the subgame perfect equilibrium is called *backward induction*. For a finite game, we start at a vertex such that all paths going out from the vertex has length at most one and lead to terminal vertices. The player who owns such a vertex makes the choice of an edge that leads to the largest payoff for him or her starting at this vertex. This choice defines the payoffs for all players at this vertex. We work back through the game tree, successively taking a vertex such that all edges going out from the vertex lead only the vertices where the payoffs have been determined for all players. The player who owns the vertex makes the choice of the edges that leads to the largest payoff for him or her starting at this vertex. This choice of an edges defines the payoffs for all the players at this vertex. Because the graph is finite, we eventually arrive at the root and have specified a complete strategy profile that is always a Nash equilibrium. This process often works for a finite horizon extensive game even if it is not finite. Backward induction constructs not only a Nash equilibrium but also a subgame perfect equilibrium. (See the formal definition below.)

§5.4 Subgame Perfect Equilibrium

Definition. A *subtree* of a game tree $\mathcal{T}$ (resp. a *sub-game graph* of a game graph $\mathcal{T}$) is the graph $\mathcal{T}_{v_0}$ of all the vertices and edges that can be reached by paths starting at a vertex v_0 . The vertex v_0 becomes the root of the subtree or sub-game graph $\mathcal{T}_{v_0}$.

Definition. Let $\mathcal{G}$ be an extensive game with perfect information. Let v_0 be a nonterminal vertex in $\mathcal{G}$. The *subgame* $\mathcal{G}(v_0)$ is the game determined as follows: (i) There is the same set of players as $\mathcal{G}$. (ii) The subtree (or sub-game graph) $\mathcal{T}_{v_0}$ is the game tree. (iii) The same players own the vertices as before. (iv) The payoffs on terminal vertices and/or terminal histories is the same as for the original game.

Definition. A strategy profile s^* is a *subgame perfect equilibrium* for an extensive game $\mathcal{G}$ with perfect information provide that it is a Nash equilibrium for every subgame $\mathcal{G}(v_0)$ of $\mathcal{G}$.

A Nash equilibrium may entail non-credible threats to take action on the part of the game tree that is not reached. Such a threat is not credible because if the actions were changed to reach that part of the game tree, then the player making the threat would change her action to optimize her payoff. A subgame perfect equilibrium rules out such threats.

We illustrated in Example ?? that there are Nash equilibria that are not subgame perfect. The method of backward induction naturally constructs a subgame perfect equilibrium as the following states.

Theorem 6 (172.1 & 173.1). **a.** For a finite horizon extensive game with perfect information, the set of subgame perfect equilibria is equal to the set of strategy profiles determined by the procedure of backward induction.

b. Every finite extensive game with perfect information has a subgame perfect equilibrium.

The extensive game given in Figure 172.1 and discussed on page 171 demonstrates that it is possible to have multiple subgame perfect equilibria for the same game. The existence of terminal vertices with the same payoff for a given player allows the situation to be possible.

Example 7 (Duelling). In this duel, there are two players. The first player takes one step forward and then can choose to fire or wait, F or W . If the first player waits or misses, then the second player takes one step forward and can choose to fire or wait, F or W . If a player misses, then the other player can wait until they are close enough to be assured of hitting the other player.

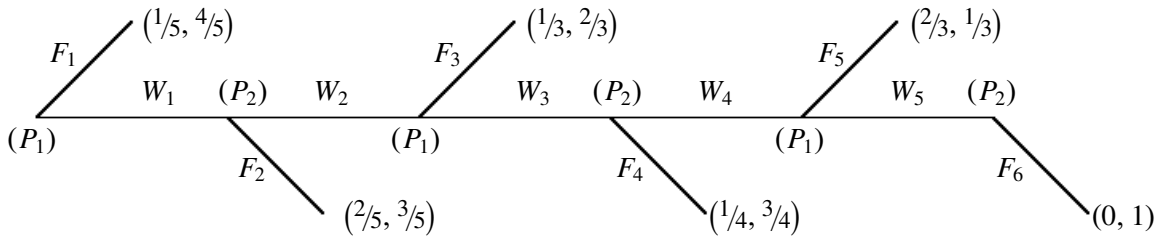


FIGURE 5. Game tree for dueling, Example ??

For example, assume there are six vertices $\{v_i\}_{i=1}^6$, with $\{v_1, v_3, v_5\}$ owned by P_1 and $\{v_2, v_4, v_6\}$ owned by P_2 . Assume the payoff are $\mathbf{u}(F_1) = (1/5, 4/5)$, $\mathbf{u}(F_2) = (2/5, 3/5)$, $\mathbf{u}(F_3) = (1/3, 2/3)$, $\mathbf{u}(F_4) = (1/4, 3/4)$, $\mathbf{u}(F_5) = (2/3, 1/3)$, and $\mathbf{u}(F_6) = (0, 1)$. At v_6 , there is only one choice, so P_2 will fire and the payoffs are $(0, 1)$. At v_5 , the payoffs are $\mathbf{u}(W_5) = (0, 1)$ and $\mathbf{u}(F_5) = (2/3, 1/3)$, so P_1 will fire. At v_4 , the payoffs are $\mathbf{u}(W_4) = (2/3, 1/3)$ and $\mathbf{u}(F_4) = (1/4, 3/4)$, so P_2 will fire. At v_3 , the payoffs are $\mathbf{u}(W_3) = (1/4, 3/4)$ and $\mathbf{u}(F_3) = (1/3, 2/3)$, so P_1 will fire. At v_2 , the payoffs are $\mathbf{u}(W_2) = (1/3, 2/3)$ and $\mathbf{u}(F_2) = (2/5, 3/5)$, so P_2 will wait. At v_1 , the payoffs are $\mathbf{u}(W_1) = (1/3, 2/3)$ and $\mathbf{u}(F_1) = (1/5, 4/5)$, so P_1 will wait. Thus, starting at the beginning of the duel, both players will wait until P_1 fires at v_3 . Note that player one fires, when his/her probability of hitting is less than a half.

To consider the situation more generally, assume that at a distance d apart the probability that the i^{th} player hits the other player if s/he fires is $p_i(d)$. The probability of missing is $1 - p_i(d)$. We assume that $p_i(d)$ increases as d decreases. The game tree is given in Figure ??.

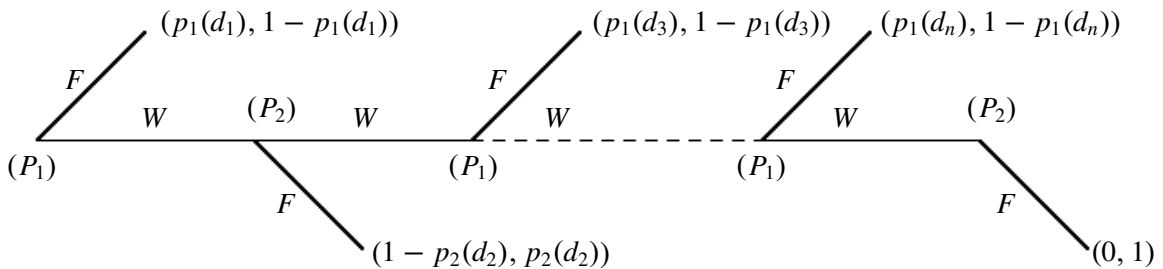


FIGURE 6. Game tree for dueling, Example ??

We use backward induction to solve this problem. For d near 0, $p_i(d) = 1$ and the player should fire. At a distance $d_{n+1} \approx 0$, P_2 fires so the payoff vector is $(0, 1)$. At a distance of d_n , the payoff for P_1 is $p_1(d_n)$ if F and 0 if W . Thus, P_1 should fire (if $p_1(d_n) > 0$.) The payoff vector is $(p_1(d_n), 1 - p_1(d_n))$. At a distance of

d_{n-1} , the payoff for P_2 is $p_2(d_{n-1})$ if F and $1 - p_1(d_n)$ if W . Thus, P_2 should fire if $p_2(d_{n-1}) \geq 1 - p_1(d_n)$ or $p_1(d_n) + p_2(d_{n-1}) \geq 1$. If P_2 fires, the payoff vector is $(1 - p_2(d_{n-1}), p_2(d_{n-1}))$. If P_2 fires at distance d_{n-1} , then P_1 should fire at d_{n-2} if $p_1(d_{n-2}) + p_2(d_{n-1}) \geq 1$. The process continues backward until P_1 waits if $p_1(d_{n-2j}) + p_2(d_{n-2j+1}) < 1$ or P_2 waits if $p_1(d_{n-2j-1}) + p_2(d_{n-2j}) < 1$. If we ignore the difference of distances for the probabilities for P_1 and P_2 , then we wait until a distance d^* with

$$p_1(d^*) + p_2(d^*) = 1.$$

Thus, the optimal strategy profile is

$$s_1(d) = \begin{cases} W & \text{if } p_1(d) + p_2(d) < 1 \\ F & \text{if } p_1(d) + p_2(d) \geq 1 \end{cases} \quad \text{and} \quad s_2(d) = \begin{cases} W & \text{if } p_1(d) + p_2(d) < 1 \\ F & \text{if } p_1(d) + p_2(d) \geq 1. \end{cases}$$

For example, assume that

$$p_1(d) = 1 - \left(\frac{d}{D}\right) \quad \text{and} \quad p_2(d) = 1 - \left(\frac{d}{D}\right)^2.$$

(Player P_2 is a better shot.) Thus, we need

$$\begin{aligned} 1 &= 1 - \left(\frac{d^*}{D}\right) + 1 - \left(\frac{d^*}{D}\right)^2 \\ 0 &= \left(\frac{d^*}{D}\right)^2 + \left(\frac{d^*}{D}\right) - 1 \\ \frac{d^*}{D} &= \frac{-1 + \sqrt{1+4}}{2} = \frac{\sqrt{5}-1}{2}. \end{aligned}$$

The payoffs are

$$\begin{aligned} p_1(d^*) &= 1 - \left(\frac{\sqrt{5}-1}{2}\right) = \frac{3-\sqrt{5}}{2} \approx 0.382 \quad \text{and} \\ p_2(d^*) &= 1 - \left(\frac{\sqrt{5}-1}{2}\right)^2 \approx 0.612. \end{aligned}$$

Thus, the better shot, P_2 , has a larger expected payoff. ■

§6.2 Stackelberg Duopoly

This example is discussed in Sections 6.2 and 3.1. It is an example of a finite horizon extensive game that is not finite, the states are continuous variables. It also illustrates the advantage of choosing first.

As before, we assume that there are two firms with inverse demand function $P_d(q_1 + q_2) = \alpha - \beta(q_1 + q_2)$ giving the price in terms of the quantity available. The profit by the i^{th} firm is

$$\pi_i(q_1, q_2) = q_i[\alpha - \beta q_1 - \beta q_2 - c],$$

where the costs for the firms are the same and $\alpha > c > 0$ and $\beta > 0$. (The result is similar if the costs of the firms are different, but the details are messier.) We assume that q_1 and q_2 are each in the interval $[0, (\alpha - c)/\beta]$.

I. If the firms choose their production levels at the same time, then we saw before that the Nash equilibrium has production levels

$$q_1^* = q_2^* = \frac{1}{3\beta} (\alpha - c),$$

and profits

$$\pi_1(q_1^*, q_2^*) = \pi_2(q_1^*, q_2^*) = \frac{1}{9\beta} (\alpha - c)^2.$$

II. Now we consider the situation where the firms choose sequentially, first P_1 and then P_2 . Using backward induction, P_2 must respond to a choice q_1 of P_1 . For q_1 in $[0, (\alpha - c)/\beta]$ fixed, P_2 chooses the value $q_2 = \hat{q}_2(q_1)$ that maximizes π_2 . Taking the derivatives,

$$\begin{aligned}\frac{\partial \pi_2}{\partial q_2} &= \alpha - c - \beta q_1 - 2\beta q_2, \\ \frac{\partial^2 \pi_2}{\partial q_2^2} &= -2\beta < 0.\end{aligned}$$

Setting the first derivative equal to zero and solving for q_2 ,

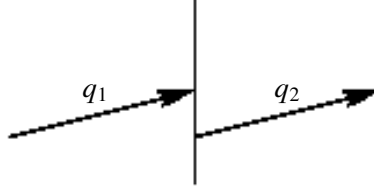


FIGURE 7. Stackelberg

$$\hat{q}_2(q_1) = \frac{1}{2\beta} (\alpha - c - \beta q_1).$$

Substituting this value in the profit for P_1 , we get the function to be maximized is

$$\begin{aligned}f(q_1) &= \pi_1(q_1, \hat{q}_2(q_1)) = q_1 \left[\alpha - c - \beta q_1 - \frac{1}{2}(\alpha - c - \beta q_1) \right] \\ &= \frac{1}{2}q_1(\alpha - c) - \frac{1}{2}\beta q_1^2.\end{aligned}$$

Differentiating,

$$\begin{aligned}f'(q_1) &= \frac{1}{2}(\alpha - c) - \beta q_1, \\ f''(q_1) &= -\beta < 0.\end{aligned}$$

Setting the first derivative equal to zero,

$$\begin{aligned}\hat{q}_1 &= \frac{1}{2\beta}(\alpha - c) > q_1^* \quad \text{and} \\ \hat{q}_2 &= \hat{q}_2(\hat{q}_1) = \frac{1}{4\beta}(2\alpha - 2c - \alpha + c) = \frac{1}{4\beta}(\alpha - c) < q_2^*.\end{aligned}$$

The profits are

$$\begin{aligned}\pi_1(\hat{q}_1, \hat{q}_2) &= \frac{1}{8\beta}(\alpha - c)^2 > \frac{1}{9\beta}(\alpha - c)^2 = \pi_1(q_1^*, q_2^*) \quad \text{and} \\ \pi_2(\hat{q}_1, \hat{q}_2) &= \frac{1}{16\beta}(\alpha - c)^2 < \frac{1}{9\beta}(\alpha - c)^2 = \pi_2(q_1^*, q_2^*).\end{aligned}$$

Thus, by choosing sequentially, the firm that chooses first produces more and gets a larger profit, and the firm that chooses second produces less and gets a smaller profit.