ROOSTER CHASE

This game is played on the grid shown in Figure 2.18. The F counter at

the lower left corner represents a farmer; the R counter at the upper right

corner represents a rooster. The farmer and the rooster move alternately

until the rooster is captured. On each move, each of them can move to a

neighboring point on the grid: up, down, left, or right. A capture occurs

when the farmer can move on a point occupied by the rooster.

(a) Can the farmer catch the rooster if he moves first? If he can, provide an

algorithm to do this in the minimum number of moves. If he cannot,

explain why.

Algorithmic Puzzles 50

(b) Can the farmer catch the rooster if he moves second? If he can,

provide an algorithm to do this in the minimum number of moves.

If he cannot, explain why.

Of course, you should assume that the rooster is not going to cooperate in

his capture.

F

R

FIGURE 2.18 Initial position of the Rooster Chase game.

Rooster Chase How can the farmer force a capture of the rooster if the

rooster tries to avoid it? What algorithm makes reaching such a position as

quick as possible?

Rooster Chase

Solution It is helpful to consider the game’s grid as obtained by connecting

centers of the squares of a standard 8 × 8 chessboard (Figure 4.47a). Under this

interpretation, the capture can only occur on the farmer’s move when the counters

occupy two adjacent squares (horizontally or vertically), which always have the

opposite colors. Initially, the counters are on the squares of the same color. Since

both counters move to a square of the opposite color on each move, the capture

cannot happen if the farmer moves first.

If the farmer moves second, he can always push the rooster into a corner to

guarantee capture by always moving to a square on the same diagonal with the

rooster and closer to him. We will denote positions of the farmer and the rooster

as (iF, jF) and (iR, jR), respectively, where i and j are the row and column of

the squares they occupy. Geometrically, the farmer’s position (iF, jF) and squares

Algorithmic Puzzles 142

F

8

7

6

5

4

3

2

1

1 2 3 4 5 6 7 8

R

(a) (b)

F

8

7

6

5

4

3

2

1

1 2 3 4 5 6 7 8

R

(c)

8

7

6

5

4

3

2

1

1 2 3 4 5 6 7 8

R

F

FIGURE 4.47 The Rooster Chase game. (a) Starting position. (b) Winning strategy.

(c) Position before the last rooster move.

(8, jF), (8, 8), (iF, 8) form a rectangle that contains the rooster and from which the

rooster cannot escape (Figure 4.47b). The rectangle shrinks along one of its side

on each move until the rooster is pushed into the upper right corner (Figure 4.47c).

A more formal way to determine the farmer’s move is to compute iR − iF, the

row distance between his current square (iF, jF) and the rooster’s current square

(iR, jR); compute jR − jF, the column distance between these squares; and then

find the maximum of these two values:

d = max{iR − iF, jR − jF}.

The farmer makes the move to decrease d, that is, he moves to the right if the

column distance jR − jF is larger than the row distance iR − iF and he moves up

otherwise. (Either the column distance or the row distance is always larger than the

other after a move by the rooster.)

143 Solutions

Since the Manhattan distance from the current position of the farmer to the

upper right corner—computed as (8 − iF) + (8 − jF)—decreases by 1 after each

move of the farmer, after 12 moves the farmer and the rooster will be in the position

shown in Figure 4.47c, from which the farmer will catch the rooster on his next

move. Thus, at the longest, it will take 14 moves of each piece for the game to end,

provided the rooster goes first from the starting position in Figure 4.47a. Of course,

the rooster may hasten the inevitable by getting to a square adjacent to the one

occupied by the rooster in as little as seven moves.

Comments The solution exploits the invariant idea to determine who needs to

go first and the greedy strategy for the capture algorithm. The transformation from

the grid to the chessboard may also be noted, although it does not play the crucial

role here.

The problem is a simplification of the Chickens in the Corn puzzle discussed

in the book’s second tutorial. Similar puzzles has been included in many puzzle

collections, for example, [Gar61, p. 57] and [Tan01, Problem 29.3].