Mathl Comput. Modelling, Vol. 10, No. 2, pp. 145–153, 1988
Printed in Great Britain

0895-7177/88 \$3.00 + 0.00
Pergamon Press plc

DEVELOPING A STRATEGY FOR "BATTLESHIP"

E. Y. RODIN

assisted by

J. COWLEY, K. HUCK, S. PAYNE and D. POLITTE

Department of Systems Science and Mathematics, Washington University, St Louis, MO 63130, U.S.A.

TABLE OF CONTENTS

1. INTRODUCTION

The game of "Battleship" combines strategy and luck and thus presents a challenge to the mathematical modeler. The game is played on a 10×10 unit grid, shown in Fig. 1. Each player is assigned five ships:

1. An aircraft carrier (5 units);
2. A battleship (4 units);
3. A submarine (3 units);
4. A cruiser (3 units);
5. A destroyer (2 units).

These ships are secretly deployed by each player on her ocean grid. Players alternate taking "shots" at the opponent's ships by calling out a grid location [e.g. (A, 4) indicates the first row and fourth column]. The first player to locate and "destroy" all of his opponent's ships is the winner. A complete summary of the rules can be found in the Appendix to this module.

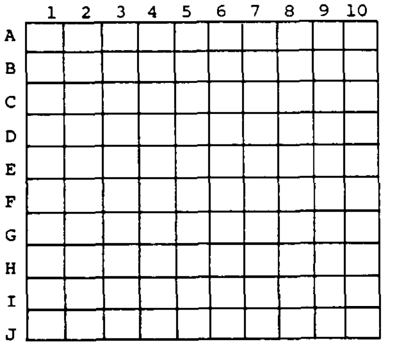

Fig. 1. The "Battleship" grid.

We shall develop a strategy for locating and destroying the opponent's ships using an algorithm which assumes that the opponent has placed her ships randomly on defense but is an intelligent attacker.

2. FINDING THE SHIPS

The first step in our plan will be to develop a strategy for finding the opponent's ships with the fewest possible shots. To simplify this process, we will first search only for the 5-unit long aircraft carrier and later combine these results with similar analyses for the smaller ships.

2.1. The aircraft carrier

Consider the 10×10 grid and note that if ship length $n = 5$, the grid can be divided into $(10/n)^2 = 4$ subgrids, each 5×5. Although a pattern of shots along two adjacent sides of each grid will hit at least one unit of any ship on the grid using nine shots, a diagonal configuration will also hit any ship on the grid with only five shots. If the four 5×5 grids are put back together so that the diagonal search patterns are parallel to each other (Fig. 2) we see that no horizontal or vertical ship placement can avoid detection. Thus 20 shots are needed to complete this pattern.

Now that we've established the best search pattern, a priority ranking of locations must be found to determine the best ordering of shots. If locations where the ship is most likely to be found are tried first, the number of necessary shots may be decreased. If we assume that any possible ship placement on the grid is equally likely, which is the case for random placement, then the number of different placements which cause a ship to occupy a particular square will be smaller near the edge of the grid and higher near the center. The method for determining the relative occupation frequency for each square is demonstrated in Figs 3 and 4. In Fig. 3, the darkened grid location is designated as a "2" for two possible ship placements. The frequency determination proceeds similarly for other locations. The darkened grid location (C, 4) shown in Fig. 4 is assigned a frequency number of "7" for the seven possible ship placements which cover that square. This procedure is continued for one quadrant of the grid and then replicated, resulting in the center weighted relative square occupation frequency for the aircraft carrier shown in Fig. 5. Notice that ship placement on the four middle squares has interference from the sides of the grid resulting in a total of $2n = 10$ possibilities.

The next step in choosing the best search pattern is assigning priorities to the search positions in Fig. 2 according to the relative frequency of occupation just determined. Considering "10" high priority and "2" the lowest priority, the ordered search pattern, called the strafing pattern, for a 5-unit ship is shown in Fig. 6.

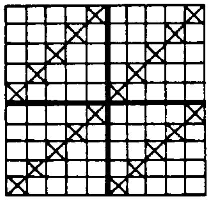

Fig. 2. Search pattern for the aircraft carrier.

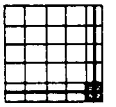

Fig. 3. Relative occupation frequency; two possible ship placements.

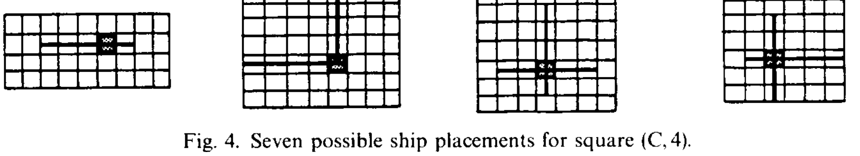

Fig. 4. Seven possible ship placements for square (C, 4).

2	3	4	5	6	6	5	4	3	2
3	4	5	6	7	7	6	5	4	3
4	5	6	7	8	8	7	6	5	4
5	6	7	8	9	9	8	7	6	5
6	7	8	9	10	10	9	8	7	6
6	7	8	9	10	10	9	8	7	6
5	6	7	8	9	9	8	7	6	5
4	5	6	7	8	8	7	6	5	4
3	4	5	6	7	7	6	5	4	3
2	3	4	5	6	6	5	4	3	2

Fig. 5. Relative square occupation frequency for the aircraft carrier.

				15					20
			11					18	
		6					7		
	10					3			
14					1				
				2					16
			4					12	
		5					8		
	17					9			
19					13				

Fig. 6. Strafing pattern for the aircraft carrier.

2.2. *The other ships*

For the complete Battleship game the search pattern must be altered to find five ships of four different lengths. Since the probability of hitting the destroyer (2 units long) is so low, the algorithm will search for all the other ships and go back to find the destroyer with a mop-up procedure if that ship has not already been found. The search pattern for the 3-unit ships is similar to that for the aircraft carrier except that the diagonals are proportionately closer (Fig. 7). Although $10^2/n = 33\frac{1}{3}$, only 33 locations need to be searched because of the boundary effect of the grid.

The relative square occupation frequencies for the shorter ships can be computed in the same manner as for the aircraft carrier. These relative frequencies are then combined to obtain a composite relative square occupation frequency. This composite, shown in Fig. 8, can be used to determine the priority of shots for the complete game.

Using this set of priorities with the search pattern of Fig. 7, we arrive at the strafing pattern for the complete game shown in Fig. 9. This strafing pattern is followed until five hits are registered and then the "destroy" algorithm discussed in the next section is implemented.

If the strafing pattern of Fig. 9 is followed in order, the resulting curve resembles a spiral. This is interesting, for when the Navy searches for a ship, they start from the last known location of the missing ship and proceed radially outward in a spiral. Thus we have obtained the Navy solution with the constraint that our search points will be confined to the points shown in Figs 7 or 9.

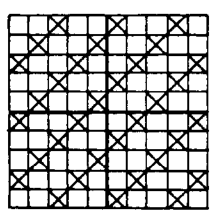

Fig. 7. Search pattern for 3-unit ships.

	1	2	3	4	5	6	7	8	9	10
A	10	15	19	21	22	22	21	19	15	10
B	15	20	24	26	27	27	26	24	20	15
C	19	24	28	30	31	31	30	28	24	19
D	21	26	30	32	33	33	32	30	26	21
E	22	27	31	33	34	34	33	31	27	22
F	22	27	31	33	34	34	33	31	27	22
G	21	26	30	32	33	33	32	30	26	21
H	19	24	28	30	31	31	30	28	24	19
I	15	20	24	26	27	27	26	24	20	15
J	10	15	19	21	22	22	21	19	15	10

Fig. 8. Composite relative frequencies.

	1	2	3	4	5	6	7	8	9	10
A			29			24			30	
B		25			14			20		
C	28			10			11			26
D			9			2			15	
E		13			1			6		
F	23			5			3			21
G			8			4			16	
H		19			7			12		
I	33			18			17			31
J			27			22			32	

Fig. 9. Strafing pattern for the complete game.

2.3. Comparison with random shooting

One method of evaluating the efficiency of our search algorithm is to compare it to the results we would expect to obtain from random guessing. Recall from Section 2.1 that the strafing pattern for the aircraft carrier was certain to locate that ship in 20 shots or less. We will find the number of shots on average which random shooting will require to find the aircraft carrier.

Let the random variable x be the number of the trial on which the ship is first hit. In order to determine the expected number of shots before we are successful, we must develop the probability distribution which describes the chances for each possible value of x. Let p be the probability of a successful shot and let $q = 1 - p$ be the probability of a miss. For the first shot the probability of success is

$$P(x = 1) = p. \tag{1}$$

If the first shot missed the aircraft carrier (probability q), then the chance that the ship is hit on the second try is

$$P(x = 2) = pq \tag{2}$$

provided the sample space remains constant from the first to the second shot, i.e. the same grid location can be called more than once. In general the probability distribution for the random variable x can be written as

$$P(x = K) = pq^{K-1}, \qquad x = 1, 2, \dots, 96. \tag{3}$$

This distribution when x goes from 1 to ∞ is known as the geometric distribution and can be applied to any case in which trials are continued until success is achieved.

The expected or average value of a random variable is found by multiplying each possible value of the variable by the probability that the random variable takes on that value and then summing over all possible values:

$$\begin{aligned} E(x) &= \text{expected value of } x \\ &= (1)p + (2)pq + 3pq^2 + \dots + 96pq^{95} \\ &= \sum_{x=1}^{96} xpq^{x-1} \\ &\simeq \sum_{x=1}^{\infty} xpq^{x-1}. \end{aligned} \tag{4}$$

The last approximation can be made because the remaining values of the series are very small compared to the others. The approximation can be rewritten as

$$\begin{aligned} E(x) &\simeq p \sum_{x=1}^{\infty} xq^{x-1} \\ &= p\frac{\mathrm{d}f(q)}{\mathrm{d}q}, \end{aligned} \tag{5}$$

where we have defined

$$f(q) = \sum_{x=1}^{\infty} q^x = \frac{1}{1-q} \tag{6}$$

since $|q| < 1$. But the derivative of $f(q)$ with respect to q is given by

$$\frac{\mathrm{d}f(q)}{\mathrm{d}q} = \frac{1}{(1-q)^2} = \frac{1}{p^2}. \tag{7}$$

Thus by equations (5) and (7) the expected value of the random variable is approximately

$$E(x) \simeq \frac{p}{p^2} = \frac{1}{p}. \tag{8}$$

For the problem under consideration the probability of hitting the aircraft carrier with one shot is the length of the ship divided by the number of squares on the board:

$$p = \frac{5}{100} = \frac{1}{20}. \tag{9}$$

Therefore, the average number of shots needed to locate the ship is

$$E(x) \simeq 20. \tag{10}$$

This value is probably somewhat larger than would be obtained by experimentation since the use of the geometric distribution required an unaltered sample space. Since a second shot is not likely to be fired at a square which is known to be empty, this is not very realistic and the probability distribution function should be appropriately modified. However, this approximation does at least indicate that the search strategy developed in the previous section is an improvement on random shooting.

3. DESTROYING THE SHIPS

3.1. The "destroy" algorithm

Once the ships have been found, the object of the game is to destroy them. Once the strafing pattern has produced five hits, control is directed to the "destroy" algorithm. The algorithm flow chart shown in Fig. 10 gives a simple outline of decision steps to follow in order to minimize the number of attempted shots which are required to sink all of the ships. If the opponent is not familiar with the strafing pattern or the "destroy" algorithm, this flow chart should be followed with the additional stipulations:

(1) Once 5 hits are registered, if two of them were on the battleship and/or the aircraft carrier, after these ships have been sunk, the strafing pattern is again resumed until five ship positions have again been located.
(2) If, after the entire strafing pattern has been implemented, the 2-unit long destroyer has still escaped detection, the other ships should be destroyed. If the destroyer is not accidentally located during this procedure, the "mop-up" strafing pattern should be used.

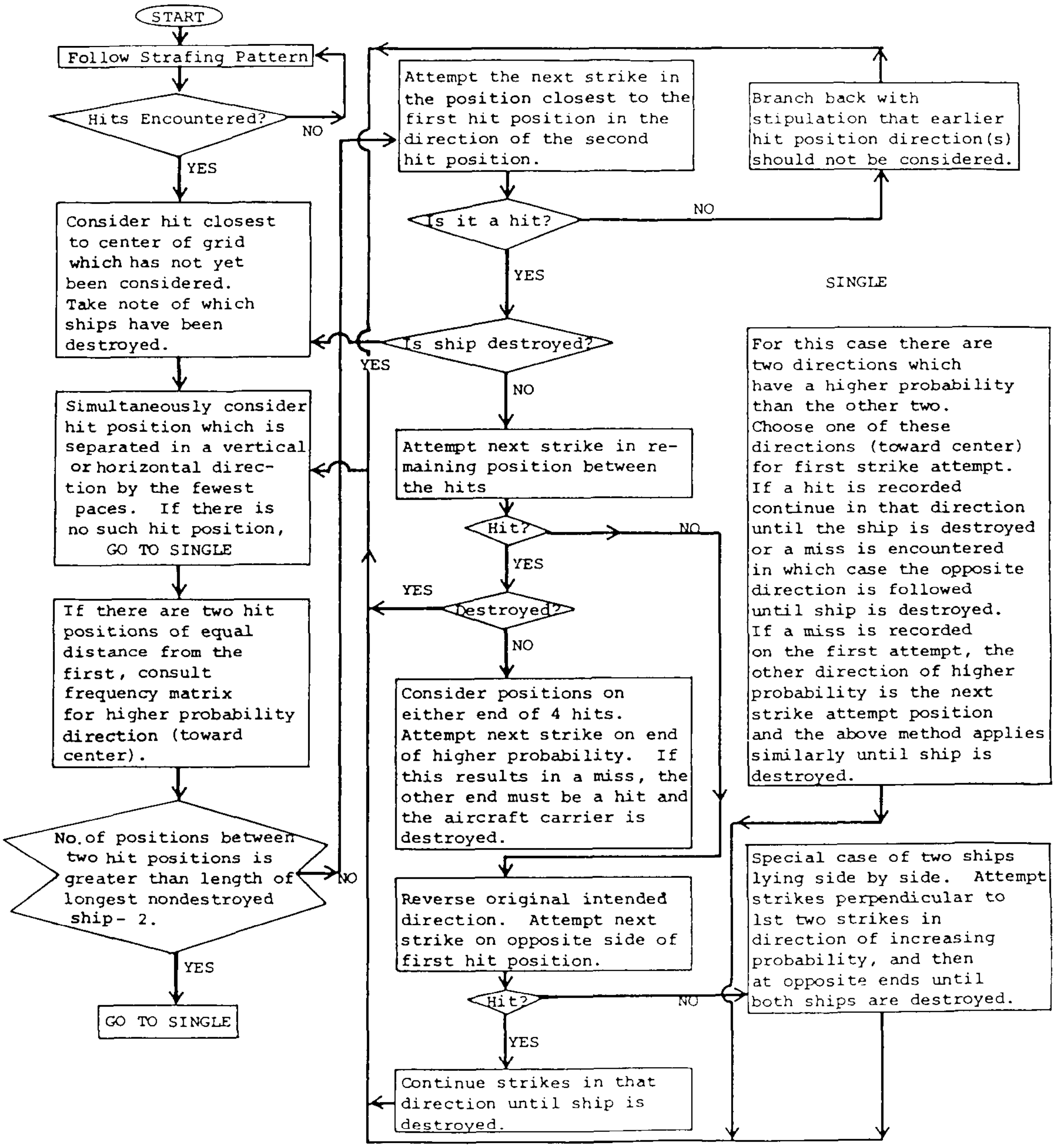

Fig. 10. Flow chart for the "destroy" algorithm.

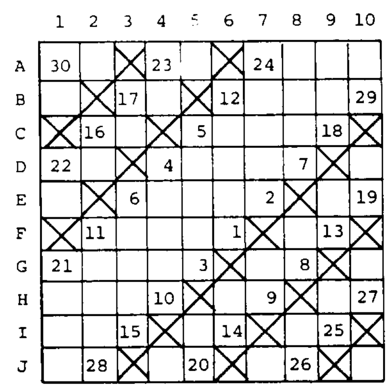

Fig. 11. "Mop-up" strafing pattern.

The "mop-up" strafing pattern shown in Fig. 11 minimizes the number of shots necessary to detect the destroyer after the regular strafing pattern is finished.

4. TESTING THE ALGORITHM

The search and destroy algorithm described in Sections 2 and 3 was tested by a group of students to determine whether or not it was an effective strategy for the game of Battleship. Members of a group familiar with the algorithm played opponents who were not. The games were played on 10 × 10 grids and were completed even after one of the players had won in order to compare the results. The results of 32 test games are shown in Table 1. The first column shows the average number of shots it took the algorithm and the opponent to destroy all ships. These averages are close due to the difficulty the current algorithm has in locating the destroyer. In games in which the destroyer was hit by the strafing pattern (24 games), the algorithm worked well and won 75% of the games. However, when the destroyer was not located by strafing, the algorithm was much worse than random shooting.

Table 1. Average number of shots needed to destroy all ships

	All games	Games when destroyer hit by strafing	Games when destroyer not hit by strafing
Algorithm	48.9	44.5	62.4
Opponent	49.5	50.6	46.25
Percentage of games won by algorithm	56.25	75	0

Those games that were lost all had one thing in common—the opponent's ships were well placed. Looking at the strafing pattern it is clear that the best defense against the algorithm is to place the ships around the perimeter of the grid, starting in the lower left-hand corner and rotating counterclockwise. This insures that a maximum number of shots will have to be taken in order to destroy all ships.

The algorithm is only effective against opponents who are not familiar with the strafing pattern and the order in which the destroy procedure proceeds. If an opponent is familiar with the algorithm, she can devise an effective defense which will probably beat the algorithm. If the same opponent is played several times in succession, care should be taken to modify the algorithm, possibly by rotation or translation, to keep the opponent from effectively defending against it.

5. REFERENCE

1. D. A. S. Fraser, *Probability and Statistics: Theory and Applications*. Duxbury Press, North Scituate, Mass. (1976).

6. EXERCISES

1. Compute the relative square occupation frequencies for the 2-, 3- and 4-unit long ships. How are these combined to give the composite of Fig. 8?

2. Suppose you decided to toss a coin a number of times until a head appeared. Find and graph the probabilities of this occurring on tries 1–4.
3. Compute the first five terms of the series for the expected value of the random variable x, the number of the trial on which the hit occurs, assuming that once a square is hit it is removed from the sample space. Compare these terms to those for the case in which the sample space remains constant. Is the difference between corresponding terms becoming larger or smaller? Write the formula for the expected value with the altered subspace in concise form.
4. Find a strafing pattern which is just as effective as the one shown in Fig. 9.
5. If your opponent placed his destroyer on (C, 8) and (C, 9) and you were following the algorithm, would you have a good chance of winning?
6. How many different ship placements are possible for the battleship?
7. Develop and test an algorithm of your own. Try to devise one which is more effective against the destroyer than the current one. You may want to do this by building some randomness into the strategy.

7. ANSWERS

1. Relative square occupations for 2-, 3- and 4-unit ships are given in Figs 12, 13 and 14, respectively. To obtain the composite, add together the frequency for the 2-, 4- and 5-unit ships and then add double the frequency of the 3-unit ship.
2. p = probability of a head = 1/2; q = probability of tail = 1/2;

$$\begin{aligned} P(x = 1) &= p &&= 1/2 \\ P(x = 2) &= pq &&= 1/4 \\ P(x = 3) &= pq^2 &&= 1/8 \\ P(x = 4) &= pq^3 &&= 1/16 \end{aligned}$$

A graph of these probabilities is given in Fig. 15.

3. First terms of series for altered sample space:

$$\frac{1}{20} + (2)\left(\frac{5}{99}\right)\left(\frac{19}{20}\right) + (3)\left(\frac{5}{98}\right)\left(\frac{19}{20}\right)\left(\frac{94}{99}\right) + 4\left(\frac{5}{97}\right)\left(\frac{19}{20}\right)\left(\frac{94}{99}\right)\left(\frac{93}{98}\right) + 5\left(\frac{5}{96}\right)\left(\frac{19}{20}\right)\left(\frac{94}{99}\right)\left(\frac{93}{98}\right)\left(\frac{92}{97}\right) \simeq 0.6719.$$

First terms of series for unaltered sample space:

$$\frac{1}{20} + 2\left(\frac{1}{20}\right)\left(\frac{19}{20}\right) + 3\left(\frac{1}{20}\right)\left(\frac{19}{20}\right)^2 + 4\left(\frac{1}{20}\right)\left(\frac{19}{20}\right)^3 + 5\left(\frac{1}{20}\right)\left(\frac{19}{20}\right)^4 \simeq 0.6555.$$

2	3	3	3	3	3	3	3	3	2
3	4	4	4	4	4	4	4	4	3
3	4	4	4	4	4	4	4	4	3
3	4	4	4	4	4	4	4	4	3
3	4	4	4	4	4	4	4	4	3
3	4	4	4	4	4	4	4	4	3
3	4	4	4	4	4	4	4	4	3
3	4	4	4	4	4	4	4	4	3
3	4	4	4	4	4	4	4	4	3
2	3	3	3	3	3	3	3	3	2

Fig. 12. Relative square occupations for a 2-unit ship.

2	3	4	4	4	4	4	4	3	2
3	4	5	5	5	5	5	5	4	3
4	5	6	6	6	6	6	6	5	4
4	5	6	6	6	6	6	6	5	4
4	5	6	6	6	6	6	6	5	4
4	5	6	6	6	6	6	6	5	4
4	5	6	6	6	6	6	6	5	4
4	5	6	6	6	6	6	6	5	4
3	4	5	5	5	5	5	5	4	3
2	3	4	4	4	4	4	4	3	2

Fig. 13. Relative square occupations for a 3-unit ship.

2	3	4	5	5	5	5	4	3	2
3	4	5	6	6	6	6	5	4	3
4	5	6	7	7	7	7	6	5	4
5	6	7	8	8	8	8	7	6	5
5	6	7	8	8	8	8	7	6	5
5	6	7	8	8	8	8	7	6	5
5	6	7	8	8	8	8	7	6	5
4	5	6	7	7	7	7	6	5	4
3	4	5	6	6	6	6	5	4	3
2	3	4	5	5	5	5	4	3	2

Fig. 14. Relative square occupations for a 4-unit ship.

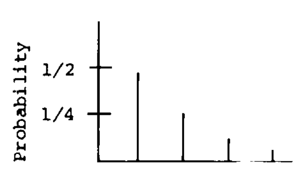

Fig. 15. Graph of the probabilities in Exercise 2.

Clearly the difference between corresponding terms is becoming larger:

$$E(x) = \sum_{x=1}^{96} x\left(\frac{5}{101 - x}\right)\left(\frac{95!(101 - x)!}{(96 - x)!100!}\right).$$

4. Rotate the pattern in Fig. 9 by 90° in either direction or flip it upside down.
5. No. Your opponent has placed his destroyer in a configuration which escapes your strafing pattern. You will probably lose.
6. For the 4-unit long battleship, there are 140 possible horizontal or vertical configurations on a 10×10 grid.

8. APPENDIX

Rules for the Basic Game

Set up the fleet

Two players sit *facing each other*, each with his kit on the table in front of him. They open their *barrier lids* so they cannot see the *ocean grid* of their opponent's kit. The lids are kept open all during the playing of the game.

Each player *secretly* places his fleet of 5 ships on his *ocean grid*. The *bottom* of each ship has two "anchoring pegs" which must be pushed *through* the holes in the *ocean grid* for placing them. Ships may be placed in *any horizontal* (back and forth) or *vertical* (up and down) position—but *not diagonally*. The ship's "anchoring pegs" will *not* fit in the grid holes if placed diagonally. All holes on the *top* of the *ships* must be lined up over holes on the *ocean grid*. *Do not* place a ship so that a part of it is overhanging the grid holes or over letters and numbers.

When both players have placed their 5 ships as desired, they announce "READY". From then on, during the game, they *may not* change the position of any ship. To do so would be cheating!

Call out the shots

In this *basic* game, the players call out *one* shot each turn to try to hit an opponent's ship.

The player with the *red* kit takes the first shot. Players then alternate, taking one shot at a time (red, blue, red, etc.).

A shot is made by calling a *letter* and a *number* to locate which hole in the opponent's *ocean grid* that shot is to be placed. That hole is located by going straight across, horizontally, from the called *letter* (printed on the side) and down, vertically, from the called *number* (printed across the top).

When a shot is called, the opponent immediately tells the player whether it is a "hit" or "miss". It is a "hit" if the called hole on his *ocean grid* is covered by a ship; and a "miss" if no ship occupies that hole. If the shot is a "hit", the opponent tells the player what *kind* of *ship* was "hit" (cruiser, carrier, etc.).

Mark shots with pegs

After a player has called his shot and found out whether it is a "hit" or "miss", he places a marker peg in his *target grid* (the one in the lid)—a *white* peg for a "miss" and a *red* peg for a "hit", to mark the location of that shot. This will guide him in placing future shots and prevent him from calling the same holes more than once.

A player does not have to mark his opponent's "misses" with white pegs, but he *must mark* any "hits" that the opponent makes on his ships with a *red* peg. When a hit has been made on a player's ship, he places a *red* peg in the *ship* at that location on his *ocean grid*.

Examples of marking the shots:

(a) John calls "F-4" to Harry. Harry announces it as a "miss". John places a *white peg* at "F-4" on his *target grid*. Harry does not place a peg in his kit.

(b) Harry calls "H-6" to John. John announces it as a "hit"—"on a destroyer." Harry places a *red peg* at "H-6" in his *target grid*. John places a *red peg* in a hole of his destroyer at "H-6" on his *ocean grid*.

Sink the fleet

Players continue taking turns, calling shots and marking them.

Whenever a ship has received enough "hits" to fill all of its holes with *red pegs*, it is *sunk* and is removed from the *ocean grid*. The player whose ship is sunk must announce it to his opponent.

The number of "hits" each ship must receive to be *sunk* is as follows: carrier—5 hits; battleship—4 hits; cruiser—3 hits; submarine—3 hits; destroyer—2 hits.

It is expected that players be *honest* in announcing "hits" when they are made. Occasionally players may make a mistake in calling a hole they didn't mean or in locating the correct hole called. If a player feels an error has been made, he may call a *truce*—and stop the game temporarily to review shots he has made in past turns. He can easily do this by calling out the location of the pegs he has placed on his *target grid* and asking the opponent to verify the "hits" and "misses" he has marked.

Win the game

The first player to sink all 5 of his opponent's ships is the *winner*.