An Optimal Dice Rolling Policy for Risk

Ger Koole

CWI

P.O. Box 94079, 1090 GB Amsterdam, The Netherlands

Published in *Nieuw Archief voor Wiskunde* **12**:49-52, 1994.

Abstract

An optimal policy for rolling dice in the game of Risk is determined, using dynamic programming.

1. Introduction

Risk is a board game in which the world is split up into 42 territories. The territories are divided amongst the players, who keep each of them occupied with one or more of their own armies. Neighbouring territories can be won in battles, which are fought with dice. In such a battle the attacker rolls three dice (representing three armies on his territory), after which the defender can choose between rolling one or two dice. Depending on the outcome, one or two armies are removed from the board, and the attacker rolls three dice again. This can be repeated until the number of armies in either territory drops below a certain level, or until the attacker decides to stop the attack. (The rules described here are the Dutch rules; the British instructions differ. See the conclusion for some remarks on this.)

Let us formulate the exact rules. Suppose the attacker has a armies on his territory, the defender d. Then the attacker is allowed to attack with $\min\{a-1, 3\}$ dice or less. (The subtracted army represents the army occupying the territory.) From what follows it will be clear that the attacker will always use as many dice as possible, usually three. The defender is allowed to roll with $\min\{d, 2\}$ dice. At any moment during the attack the attacker is allowed to stop.

Now, after the attacker's roll, his dice are sorted in descending order, for example 5-3-2. In case the defender subsequently decides to roll one die, this one is compared with the highest die of the attacker. The attacker loses an army if the defender has rolled higher or equally high, in the example 5 or 6. In all other cases the defender loses an army.

If the defender rolls two dice, then the highest of the two is compared with the highest of the attacker, and the lowest with the second highest of the attacker. Again, in case of a tie, the attacker loses an army. Thus, if the defender rolls 4-3 against 5-3-2, then the 5 and the 4, and both 3's are compared, and both players loose one army. The question we will answer is the following: For each roll of the attacker, how many dice should the defender use?

2. The Reward Function

To determine the best play for the defender, we have to decide upon a function to be maximised. Such a reward function should take the losses of both the attacker and the defender into account. As for each die rolled by the defender exactly one army is removed from the board, it does seem reasonable to maximise the probability that this is one of the attacker's armies. Now assume, for fixed n, that $a \geq n+3$ and $b \geq n+1$. Then the battle can be fought until n armies are removed from the board, while the attacker can always roll three dice, and the defender one or two. If the probabilities that the attacker looses an army over the n armies at stake are summed, then maximizing the attacker's loss per army results in maximising the expected number of armies lost, or, equivalently, in *maximising the expected attacker's loss per army at stake*.

Other objectives can be thought of, like maximising the probability of not losing your territory, or maximising the number of remaining armies after a battle. A disadvantage of these types of criteria (besides having to choose between them) is the fact that the optimal policy will depend on the distribution of armies over different territories, resulting in a very complicated policy. Instead, we restrict ourselves to the reward function formulated above.

3. Analysis

To compute the optimal policy, we could consider all possible rolls by the attacker one by one, and for each compute the defender's best action. In the sequel we will find that this is not the right solution, but let us just follow this line of reasoning for a moment.

For example, assume that the defender has rolled 5-3 (omitting the third and lowest die). If we defend with one die, then the expected reward is $\frac{1}{3}$, as in 2 out of 6 possibilities the attacker loses an army.

If we roll two dice, there are 36 possibilities, each with probability $\frac{1}{36}$. (Note that most possible rolls occur twice.) It is easy to verify that the probabilities of losing 0, 1 or 2 armies are all exactly $\frac{1}{3}$, for both players. Thus the expected reward is $\frac{1}{3}(0+1+2) = 1$, per army at stake $\frac{1}{2}$. Therefore, it seems reasonable to roll two dice against 5-3. However, let us repeat this reasoning for an attack roll of 6-5.

If we defend with one die, the expected attacker's loss is $\frac{1}{6}$. With 2 dice against 6-5, his loss is $\frac{1}{12}2 + \frac{1}{4}1 + \frac{2}{3}0 = \frac{5}{12}$. Per army at stake this is $\frac{5}{24} \geq \frac{4}{24} = \frac{1}{6}$. Should we conclude from this that it is better to defend with two dice against 6-5?

The flaw in our reasoning is that we considered the rolls in isolation. The computed actions should only be taken if the attacker always rolls 5-3 or 6-5. For example, we will show that it is better to roll one die against 6-5, the intuition being that it is better to spare the second army for a later, more favourable roll of the attacker.

We need some notation. Let p_{xy} denote the probability of the attacker rolling x-y, with $x \geq y$. Thus, to compute p_{53}, we sum the probabilities of rolls like 5-3-3, 2-5-3 and 3-1-5. Similarly, write q^2_{rs} for the probability of the defender rolling r-s. For convenience we also write q^1_r for the probabilities of a single die roll. Furthermore, denote with $R^2_{xy,rs}$ ($R^1_{xy,r}$) the attacker's loss in case of an attack roll of x-y and a defense roll of r-s (r). For example, $R^2_{53,43} = 1$, $R^2_{31,62} = 2$ and $R^1_{64,5} = 0$.

Now we define V_n as the maximum expected loss incurred by the attacker, when there are n armies at stake. We will be able to express V_n in V_{n-1} and V_{n-2}, giving us the possibility to compute V_n recursively. The following events occur successively, assuming there are n armies at stake. First the attacker rolls, upon which the defender optimally decides between rolling with one or two dice. Depending on the outcome one or two armies are removed from the board, and then, again depending on the defender's choice, there are still $n-1$ or $n-2$ more armies to go. As a formula:

$$V_n = \sum_{x,y} p_{xy} \max\Big\{ \sum_r q_r^1 R^1_{xy,r} + V_{n-1}, \sum_{r,s} q^2_{rs} R^2_{xy,rs} + V_{n-2}\Big\}.$$

This is an example of *dynamic programming*. The actions maximising V_n are the optimal actions, if there are n armies at stake, assuming that both players have enough armies. To compute V_n and the corresponding actions we also have to specify V_0 and V_1. Naturally, we take $V_0 = 0$ and $V_1 = \sum_{x,y} p_{xy} \sum_r q_r^1 R^1_{xy,r}$. For a general reference to dynamic programming, see Bertsekas [1].

The optimal actions depend on n. A small computer program learns us that for $n \geq 5$ the chosen actions remain the same. According to Markov decision theory, these actions maximise the average reward, i.e., they maximise the attacker's loss per army. The optimal policy is quite simple: one should use two dice only if the second highest roll is 1, 2 or 3. Thus against 5-4 one die should be rolled, while against 6-3 two dice is optimal. Furthermore, the difference $V_n - V_{n-1}$ tends to the attacker's expected loss per army. We find that $\lim_{n\to\infty}(V_n - V_{n-1}) \approx 0.500257$, remarkably close to 0.5. (We can be sure that it is bigger than 0.5; the algorithm used has a high enough precision. Besides, taking as initial values $V_0 = 0$ and $V_1 = 0.5$ does not give $V_2 = 1$.)

Note that, when calculating V_n, we assumed that the attacker has at least $n+3$ armies, and the defender at least $n+1$. Thus we have shown that if both players have a sufficient number of armies in the territories involved, then the defender should only roll two dice if the second highest die rolled by the attacker is 1, 2 or 3. Moreover, this dice game is almost fair, i.e., the losses of both players are almost equal.

4. Conclusion

In the previous section we derived an optimal defence policy for the game of Risk, assuming that there are enough armies in the territories involved in the attack. The optimal policy (according to which we have to roll two dice only if the second highest die of the attacker's roll is 1, 2 or 3) leads to an average loss per army very close to $\frac{1}{2}$ for both players. Thus, each player can make the decision to attack or not just based on tactical arguments, as the number of lost armies hardly depends on who is attacking or defending.

It is remarkable that the British rules differ from the Dutch rules in that the defender has to roll its dice together with the attacker, basically giving two possible policies, always rolling one die or always rolling two dice. Always rolling two is more advantageous than always rolling one die, as $\sum_{xy,rs} p_{xy} R^2_{xy,rs} q^2_{rs}/2 \approx 0.46$, while $\sum_{xy,r} p_{xy} R^1_{xy,r} q^1_r \approx 0.34$. The attacker's advantage in this case is an incentive to attack sooner.

Bibliography

[1] D.P. Bertsekas. *Dynamic Programming.* Prentice-Hall, 1987.