Available online at www.sciencedirect.com
ScienceDirect

European Journal of Operational Research 183 (2007) 805–811

EUROPEAN JOURNAL OF OPERATIONAL RESEARCH

www.elsevier.com/locate/ejor

O.R. Applications

Dynamic programming analysis of the TV game “Who wants to be a millionaire?”

Federico Perea *, Justo Puerto

University of Seville, Department of Statistics and Operations Research, Faculty of Mathematics, c/Tarfia sn, 41012 Seville, Spain

Received 10 February 2006; accepted 5 October 2006
Available online 15 December 2006

Abstract

This paper uses dynamic programming to investigate when contestants should use lifelines or when they should just stop answering in the TV quiz show ‘Who wants to be a millionaire?’. It obtains the optimal strategies to maximize the expected reward and to maximize the probability of winning a given amount of money.

Keywords: Dynamic programming; Markov processes; Recreation

1. Introduction

‘Who wants to be a millionaire?’ is a successful television game show in many countries. One contestant addresses 15 multiple-choice questions in a row. In each step, a question and four possible answers are shown. After being shown the question, the contestant can decide to stop playing and keep the money accumulated up till then, and the game is over, or to answer the question. If they decide to stay in the game, they can use up to three lifelines to answer the question. Each lifeline may only be used once during a contestant’s entire game. These lifelines are:

- Lifeline 1 or the *50:50 option*: two of the three incorrect answers are removed.
- Lifeline 2 or *phone a friend*: the contestants may speak to a friend or relative on the phone for 30 s to discuss the question.
- Lifeline 3 or *ask the audience*: the audience votes with their keypads on their choice of answer. The result of this poll is listed in percentages and shown to the contestant.

There are two stages (“guarantee points”) where the money is banked and cannot be lost even if the candidate gives an incorrect response. Those questions are the 5th one and the 10th one.

The decision of when to stop playing or when to use the lifelines should be treated rationally, although contestants rarely seem to make such rational decisions. In this paper, we address the problem of when to stop playing and when to use the lifelines as a dynamic programming (DP for short) problem and the optimal strategies are identified. The probabilities of correctly answering are based on observation of the Spanish game and the

* Corresponding author. Tel.: +34 654111231.
E-mail addresses: perea@us.es (F. Perea), puerto@us.es (J. Puerto).

0377-2217/$ - see front matter © 2006 Elsevier B.V. All rights reserved.
doi:10.1016/j.ejor.2006.10.041

empirical model. There have been some approaches to the mathematical analysis of the game using simplified versions and as an educational resource in classrooms, for instance Cochran (2001) and Rump (2001). For other examples on the use of DP to analyze other contests see Thomas (2003), who analyzes 'The weakest link' or Sniedovich (2005) and Smith (2007) for all sorts of board games. The interested reader is also referred to the analysis of the HI-LO game in Freeman (2001) and of Cricket in Clarke and Norman (2003) and the references there. Our formulation of 'Who wants to be a millionaire?' works for all existing tables of prices of the game. We give the results for the Spanish version in 2003, where the monetary values of the questions were 150, 300, 450, 900, 1800, 2100, 2700, 3600, 4500, 9000, 18 000, 36 000, 72 000, 144 000 and 300 000 Euros, respectively.

The rest of the paper is organized as follows: Section 2 shows the general mathematical model (states, feasible actions, rewards, transition function, probabilities of answering correctly and their estimation). We present in Section 3 the description of the two particular models to be studied in this paper, in which we want to maximize the expected reward and the probability of reaching and correctly answering a given question respectively. To finish, Section 4 presents some concluding remarks based on simulations of how to play in a dynamic way.

2. Basic ideas

In the game, the contestant makes a decision each time a question and four possible answers are shown. The planning horizon is finite, there are $N = 16$ stages, where the 16th stage stands for the situation after answering question number 15 correctly. To make a decision, contestants have to know the index of the question they are facing and the lifelines they have already used. The history of the game is summarized in this information. Let $\mathscr{S}$ be the set of state vectors, whose elements are of the form $s = (k, l_1, l_2, l_3)$. Variable k is the index of the current question and

$$l_i = \begin{cases} 1 & \text{if lifeline } i \text{ may be used,} \\ 0 & \text{if lifeline } i \text{ was already used.} \end{cases} \tag{1}$$

Let $\mathscr{A}(s)$ denote the set of feasible actions in state s. $\mathscr{A}(s)$ depends on the question index and the lifelines left. If $k = 16$, the game is over and there are no feasible actions. If $k \leqslant 15$, the contestant has several possibilities:

- To answer the question without using lifelines.
- To answer the question employing one or more lifelines. In this case, contestants must also specify the lifelines they are going to use, remembering that each lifeline may only be used once during the whole game.
- To stop and quit the game.

If the player decides to stop, the immediate reward is the monetary value of the last question answered. If the candidate decides to answer, the immediate reward is a random variable and depends on the probability of answering correctly. If the candidate fails, the immediate reward is the last guarantee point reached before failing. If the candidate chooses the correct answer, there is no immediate reward and he or she goes on to the next question, and the reward is the expected (final) reward from the resulting state.

Denote r_k the immediate reward if the candidate decides to quit the game after answering question k correctly, and denote r_k^* the immediate reward if the candidate fails in question $k + 1$. The values of r_k and r_k^* are shown in Table 1.

After a decision is made, the process proceeds to a new state.

- If the contestant decides to stop at a question or answers it incorrectly, the game is over.
- If the contestant decides to play and chooses the correct answer, there is a transition to another

Table 1
Immediate versus assured rewards

r_0	0	r_0^*	0
r_1	150	r_1^*	0
r_2	300	r_2^*	0
r_3	450	r_3^*	0
r_4	900	r_4^*	0
r_5	1800	r_5^*	1800
r_6	2100	r_6^*	1800
r_7	2700	r_7^*	1800
r_8	3600	r_8^*	1800
r_9	4500	r_9^*	1800
r_{10}	9000	r_{10}^*	9000
r_{11}	18 000	r_{11}^*	9000
r_{12}	36 000	r_{12}^*	9000
r_{13}	72 000	r_{13}^*	9000
r_{14}	144 000	r_{14}^*	9000
r_{15}	300 000	r_{15}^*	300 000

state $t(s,a) = (k+1, l'_1, l'_2, l'_3) \in \mathscr{S}$, where the lifeline indicators l'_i are

$$l'_i = \begin{cases} l_i - 1 & \text{if lifeline } i \text{ is used in question } k, \\ l_i & \text{otherwise.} \end{cases}$$

It is an assumption of the model that the probability of success is dependent on the stage. We further assume that the probabilities can be influenced by using lifelines, which are supposed to be helpful (i.e. to increase the probability of answering correctly).

One of the cornerstones in the resolution of the actual problem is to get a good estimate of the probabilities in the decision process. For a realistic estimation, one would need detailed data: for each question and for each possible combination of lifelines, there should be a certain number of candidates who answered correctly and of those who failed, and this number should be high enough to estimate the probabilities. The actual data are only available for approximately 40 games broadcast on the Spanish TV, and, of course, for most combinations of lifelines there are no observations, making it impossible to give an estimate of all the probabilities. Nevertheless, this lack of data is solved as follows.

Let p_k^* denote the probability of answering correctly without using any lifeline and p_k^i be the probability of correctly answering question k using the ith lifeline. From the available data, we performed a constrained linear regression analysis by using the "least squares" so that all probabilities lie between 0 and 1. This way we obtain a model $p_k = c + m(k-1)$ where $c \leqslant 1$ and $c - 14m \geqslant 0$, p_k representing the probabilities p_k^*, p_k^1, p_k^2 or p_k^3. The resultant regression lines and their respective r^2 parameters are

$$\begin{aligned} p_k^* &= 0.996 - 0.051(k-1), \quad r^2 = 0.941, \\ p_k^1 &= 1.000 - 0.037(k-1), \quad r^2 = 0.883, \\ p_k^2 &= 1.000 - 0.029(k-1), \quad r^2 = 0.858, \\ p_k^3 &= 1.000 - 0.041(k-1), \quad r^2 = 0.865. \end{aligned} \tag{2}$$

The goodness of fit was quite satisfactory for each regression line, as in all of them the corresponding values of r^2 were close to 1. Because of that, in the rest of the paper we will consider the estimated values of p_k^* and p_k^i from their regression lines, $i = 1,2,3$.

To estimate the probabilities of correctly answering a question using several lifelines, we assume that there exists a multiplicative relationship between the probability of failing in a given state using lifeline i and the probability of failure without lifelines. This relation is such that the probability of failing decreases by a fixed factor c^i, $0 < c^i < 1$ $i = 1,2,3$. Mathematically:

$$q_k^i = q_k^* c_k^i \Longleftrightarrow p_k^i = 1 - (1 - p_k^*)c_k^i, \tag{3}$$

where $q_k^* = 1 - p_k^*$ and $q_k^i = 1 - p_k^i$, $i = 1,2,3$, $k = 1,\ldots,15$. We assume further that the combination of several lifelines modifies the original probabilities p_k^*, q_k^* in a multiplicative way, by multiplying the different 'c' constants. This simplification allows us to give an empirical expression of the probabilities. Under this assumption, we can use the information we have about the candidates to estimate the probabilities of answering correctly with any feasible combination of lifelines.

From Eq. (3) and the values obtained from the regression lines in (2), the values of the 'c' constants are derived, see Table 2.

Table 2
Correction factors

k	c_k^1	c_k^2	c_k^3
1	0	0	0
2	0.672	0.527	0.745
3	0.698	0.547	0.773
4	0.707	0.554	0.783
5	0.711	0.557	0.788
6	0.714	0.559	0.791
7	0.716	0.561	0.793
8	0.717	0.562	0.795
9	0.718	0.563	0.796
10	0.719	0.563	0.796
11	0.719	0.564	0.797
12	0.720	0.564	0.798
13	0.720	0.564	0.798
14	0.721	0.565	0.799
15	0.721	0.565	0.799

3. The mathematical models

In this section we present the two models analyzed in this paper. The first one, Section 3.1, is intended to maximize the expected reward. The second one, Section 3.2, finds the optimal strategy so that the probability of reaching and correctly answering a given question is maximized.

3.1. Maximizing the expected reward

Let p_s^a denote the probability of answering correctly if in state $s \in \mathscr{S}$ action $a \in \mathscr{A}(s)$ is chosen.

We assume that p_s^a only depends on the question index and on the lifelines that are used $\forall s \in S$, $\forall a \in \mathscr{A}(s)$.

Let $f(s)$ be the maximum expected reward that can be obtained starting at the state $s = (k, l_1, l_2, l_3)$. We can evaluate $f(s)$ in the following way. The maximum expected reward from s on is the maximum among the expected rewards that can be obtained from all the possible states that can be achieved from s. At that point, we can either quit the game, thus ensuring r_k, or go to the next question (assume indexed by $k+1$). In the latter case, if we choose an action $a \in \mathscr{A}(s)$ then we answer correctly with probability p_s^a and fail with probability $(1 - p_s^a)$. The reward when failing is given by the assured prior reward in question $k+1$, i.e. r_k^*. On the other hand, answering question $k+1$ correctly produces a transition to the next question with the remaining lifelines. Denote by $t(s,a)$ the transition function that gives the new state when action a is chosen in state s. Then, from that point on the expected reward is $f(t(s,a))$. To summarize, the expected reward under action 'a' is

$$p_s^a f(t(s,a)) + (1 - p_s^a) r_k^*. \tag{4}$$

Hence

$$f(s) = \max_{a \in \mathscr{A}(s)} \{ r_k, p_s^a f(t(s,a)) + (1 - p_s^a) r_k^* \}. \tag{5}$$

In order to get the maximum expected reward we have to evaluate the functional f in the departing state. The values of f can be recursively computed by backward induction once we know the value of f at any feasible state of the terminal stage, that is, being in question 15 with any possible combination of lifelines. These values are easily computed and their values are shown in Table 3.

Example 3.1. The maximum expected reward when starting in question 1 with all lifelines available, $f(1,1,1,1)$, is equal to 2386.7 and an optimal strategy to achieve this expected reward is shown in the first column of Table 4.

In order to see the robustness of the given solution, we analyze the optimal strategies when the model is modified by changing each coefficient by its value plus or minus one standard deviation. The optimal strategies for each of those four new models are shown in Table 4.

Table 3
Expected reward in the 8 possible terminal states of model 1

State	f (State)
15,1,1,1	231 858.5
15,0,0,1	144 000
15,0,1,0	181 854
15,0,1,1	205 549
15,1,1,0	214 763.7
15,1,0,1	179 493.6
15,1,0,0	149 262
15,0,0,0	144 000

Table 4
Solution to model 1 showing the optimal strategy at each question (QI) and the expected reward for the basic model (column 1)

QI	$c - m(k-1)$	$c + s_c - m(k-1)$	$c - s_c - m(k-1)$
1	No lifelines	No lifelines	No lifelines
2	No lifelines	No lifelines	No lifelines
3	No lifelines	No lifelines	No lifelines
4	No lifelines	No lifelines	No lifelines
5	No lifelines	No lifelines	No lifelines
6	No lifelines	No lifelines	No lifelines
7	No lifelines	No lifelines	No lifelines
8	No lifelines	No lifelines	Audience
9	50:50	Audience	50:50
10	Phone	Phone	Phone
11	No lifelines	No lifelines	No lifelines
12	Audience	No lifelines	No lifelines
13	Stop	50:50	Stop
14	–	Stop	–
ER	2386.7	3350.7	1683.4

QI	$c - (m + s_m)(k-1)$	$c - (m - s_m)(k-1)$
1	No lifelines	No lifelines
2	No lifelines	No lifelines
3	No lifelines	No lifelines
4	No lifelines	No lifelines
5	No lifelines	No lifelines
6	No lifelines	No lifelines
7	No lifelines	No lifelines
8	Audience	No lifelines
9	50:50	Audience
10	Phone	Phone
11	No lifelines	No lifelines
12	No lifelines	No lifelines
13	Stop	50:50
14	–	Stop
ER	2017.7	2885.9

The other columns show the sensitivity of the solution, by changing the two regression coefficients (c and m) by one standard deviation.

One can observe from Table 4 that adding (subtracting) one standard deviation to $c(m)$ results in increasing the probability of correctly answering. Hence, the resulting strategies become more risky delaying the use of lifelines and reaching the final question. On the other hand, subtracting (adding) one standard deviation to $c(m)$ diminishes probabil-

Table 5
Optimal strategies for $w = 1$–5 and the corresponding probabilities of success

QI	$w = 1$	$w = 2$	$w = 3$	$w = 4$	$w = 5$
1	All lifelines	No lifelines	No lifelines	No lifelines	No lifelines
2		All lifelines	Audience	No lifelines	No lifelines
3			50:50, phone	50:50	No lifelines
4				Phone, audience	50:50
5					Phone, audience
Pr.	1	0.982	0.916	0.812	0.680

ities of correctly answering and the strategies tend to use lifelines soon after question 8.

3.2. Reaching a question

In this section we address a different objective, with a correspondingly different recurrence relation, to the contest. Now we want to find the optimal strategy in order to maximize the probability of reaching and correctly answering a given question. Moreover, we also give the probability of doing that if we follow an optimal strategy.

Let us define the new problem. Recall that a state s is defined as a four-dimensional vector, as before

$$s = (k, l_1, l_2, l_3).$$

Let w, $w = 1, 2, \ldots, 15$, be a fixed number. Our goal is to correctly answer question number w. We denote by $f(s)$ the maximum probability of reaching and correctly answering question number w, starting in state s.

We evaluate $f(s)$ in the following way. The maximum probability of reaching and correctly answering the question number w starting in state s is the maximum among the possible actions $a \in \mathscr{A}(s)$ of the probability of answering the current question correctly times the maximum probability of getting our goal from the state $t(a,s), a \in \mathscr{A}(s)$, where $t(a,s)$ is the transition state after choosing the action a in the state s if answering correctly.

Then, we have

$$f(k, l_1, l_2, l_3) = \max_{\substack{0 \leqslant g_i \leqslant l_i \\ g_i \in \mathbb{Z}\ \forall i}} \{p_{k,g_1,g_2,g_3} \cdot f(k+1, l_1 - g_1, l_2 - g_2, l_3 - g_3)\},$$

where $p_{k,g_1,g2,g_3}$ is the probability of correctly answering the kth question, using lifeline i if $g_i = 1$, $i = 1, 2, 3$.

The function f is a recursive functional, therefore to obtain its evaluation by backward induction we need its value at each state in the terminal stage.

Table 6
Probabilities of success for $w = 6$–15. In each case, the strategy is to use no lifelines until question $w - 2$, and then use the lifelines in the order 'Audience, 50:50, phone'

QI	$w = 6$	$w = 7$	$w = 8$	$w = 9$	$w = 10$
Pr.	0.538	0.400	0.278	0.179	0.107
QI	$w = 11$	$w = 12$	$w = 13$	$w = 14$	$w = 15$
Pr.	0.058	0.029	0.013	0.005	0.002

Notice that the goal in this formulation is to correctly answer question w. Thus, the probability of having done so if we are already at question $w + 1$ is clearly 1. Hence, we have

$$f(w + 1, l_1, l_2, l_3) = 1 \quad \forall l_i \in \{0, 1\}, \quad i = 1, 2, 3.$$

Once we have the evaluation of the functional at the terminal stage, the solution of this model is f (*departing state*).

The optimal strategies and the probabilities of reaching and answering correctly any possible question $w = 1, \ldots, 15$ are shown in Tables 5 and 6. Note that the strategies have the same pattern, that is, they all use lifelines at the end and, from goal $w = 6$ on, in the same order: audience, 50:50 and phone.

4. Further analysis of the game

In previous sections, the problem has been analyzed in a static way, since it was assumed that all the probabilities are determined "a priori", that is, without the actual knowledge of each question. But the game is actually played changing the probabilities of correctly answering each time that the player faces the current question. For this reason, a new approach to the problem is proposed. In this approach we assume that the player is able to estimate his/her probability of correctly answering the current question, the probabilities of correctly answering the following questions remaining unchanged as estimated in Eq. (2).

In this analysis contestants modify at each stage k the probability p_k^* of correctly answering according to their own knowledge of the subject, having this way a more realistic way of playing the game. This feature has been incorporated in our computer code so that at each stage the player can change the probability of answering the current question correctly. Notice that this argument does not modify our recursive analysis of the problem. It only means that we allow variation of the probability p_k^* at each step of the analysis.

4.1. Simulation

In this section we are going to show simulations of our model of the game played in its dynamic version. We will assume that on each actual question the probability of correctly answering is modified once the question is known. Suppose that the contestant is now facing the question kth, deciding whether answering the question or not depending on the degree of difficulty of the actual question. The dynamic model assumes that the probabilities of correctly answering the following questions, that is from $k+1$ on, are the ones estimated before. For any $k=1,\ldots,15$ let X_k be the random variable defined via

$$X_k := \text{Probability of correctly answering question } k. \tag{6}$$

In order to simplify the simulation we assume that the probabilities of answering correctly without using lifelines can be

- 1 if the contestant knows the right answer.
- $\frac{1}{2}$ if the contestant doubts between two possible answers.
- $\frac{1}{3}$ if the contestant is sure that one of the answers is incorrect, the other three answers being possibly correct.
- $\frac{1}{4}$ if the contestant does not know anything about the answer and the four of them are equally possible to him or her.

In other words, $X_k \in \left\{\frac{1}{4},\frac{1}{3},\frac{1}{2},1\right\}$. We will implement and run two different simulations, in which the probability functions of X_k are

1. $P[X_k=1]=P\left[X_k=\frac{1}{2}\right]=P\left[X_k=\frac{1}{3}\right]$ $=P\left[X_k=\frac{1}{4}\right]=\frac{1}{4}\ \forall k=1,\ldots,15.$
2. But for a more realistic approach, we consider in the second simulation that the probabilities vary depending on the question index. That is, the higher the index is, the more difficult the corresponding question becomes. To do so we consider that $P[X_k=1]=M_{1,k}, P\left[X_k=\frac{1}{2}\right]= M_{2,k}$, $P\left[X_k=\frac{1}{3}\right]= M_{3,k}, P\left[X_k=\frac{1}{4}\right]= M_{4,k}\ \forall k= 1,\ldots,$ 15, where M is the following matrix:

$$M := \frac{1}{24}\begin{pmatrix} 14 & 13 & 12 & 11 & 10 & 9 & 9 & 8 & 7 & 6 & 5 & 4 & 3 & 2 & 1 \\ 6 & 5 & 5 & 5 & 5 & 5 & 5 & 4 & 4 & 4 & 4 & 4 & 4 & 4 & 4 \\ 3 & 4 & 4 & 5 & 5 & 5 & 6 & 6 & 6 & 6 & 6 & 7 & 7 & 8 & 9 \\ 1 & 2 & 3 & 3 & 4 & 5 & 5 & 6 & 7 & 8 & 9 & 9 & 10 & 10 & 10 \end{pmatrix}$$

Both simulations were implemented and run 10000 times. In Table 7 the frequencies in which the scheme stopped at a each question for both simulations are presented. Note that no instance stopped at questions 1, 2, 3, 4, 6, 7, 8, 10 or 11 in any simulation. This comes from the fact that in those questions the risk one takes when answering is not high enough to stop, because the contestant is either too close to the beginning of the game or not far enough after answering a question at a guarantee point. Note also that stopping at question 16 means to correctly answer question 15, that is, to finish the game successfully. When taking into account that the final questions are more difficult than the first ones, one can see that the optimal strategies stop with higher probability at questions 9, 12, 13 than at the beginning or at the end of the game.

Notice that any kind of "a priori" probabilistic information, based on the knowledge of the actual player, can be incorporated into the model. This incorporation is done by computing "posterior" probabilities using Bayes' rule. It is clear that the strategies change depending on the probabilities of correctly answering the question that the contestant is facing. As can be seen, depending on the simulated probability, the strategies can vary from stopping at the fifth question until being on the game until the very end.

Table 7
Frequency in percentages of stopping at a given question

QI	5	9	12	13	14	15	16
Sim. 1	25.08	19.18	27.88	13.88	6.89	3.32	3.77
Sim. 2	17.15	24.58	38.81	13.64	4.49	1.05	0.28

5. Conclusion

This paper presents an analysis of the TV game “Who wants to be a millionaire?” based on dynamic programming. Such analysis was used to analyze two situations: one in which the objective is to maximize the expected reward and another in which the goal is to maximize the probability of reaching a given question, meaning to win a given amount of money.

Dynamic programming analysis shows that stopping at question 13 and using the lifelines not before question 9 is optimal to maximize the expected reward. The robustness of such optimal strategy is also analyzed. In another model of the game, we prove also by dynamic programming techniques that when one wants to maximize the probability of winning a given amount of money, the optimal strategies consist of using the lifelines at the end of the game in a given order, see Table 5.

To finish the paper we incorporate to our analysis a new feature: the possibility of changing at each stage the probability of correctly answering the current question. Such model was tested in two different cases: (1) when the probabilities of correctly answering are uniform and do not depend on the question index and (2) when the questions become more difficult as the game approaches the final question. Two interesting results can be stated: contestants should not stop when they are before question 4, nor after the two guarantee points, as the risk they take when answering those questions after the guarantee points is not high enough to make them stop.

Acknowledgments

The authors want to acknowledge the partial financial support obtained through the European project MaMaEuSch, project reference number 94,342-CP-1-2001-1-DE-COMENIUS-C21, “Grupo de Investigación de la Junta de Andalucía” FQM 331 and Spanish Ministry of Education Grant No. MTM2004-0909.

References

Clarke, S.R., Norman, J.M., 2003. Dynamic programming in cricket: Choosing a night watchman. Journal of the Operational Research Society 54, 838–845.

Cochran, J.J., 2001. Who Wants To Be A Millionaire®: The Classroom Edition INFORMS Transactions on Education, 1(3), http://ite.informs.org/Vol1No3/Cochran/.

Freeman, J.M., 2001. Variable wager HI-LO: A stochastic dynamic programming analysis. Journal of the Operational Research Society 52, 352–357.

Rump, C.M., 2001. Who Wants to See a $Million Error?, INFORMS Transactions on Education, 1(3), http://ite.informs.org/Vol1No3/Rump/.

Smith, D.K., 2007. Dynamic programming and board games: A survey. European Journal of Operational Research 176 (3), 1299–1318.

Sniedovich, M. 2005. Educational OR/MS Games: Why and how” Working paper, presented at IFORS Conference, Honolulu 2005. Department of Mathematics and Statistics, University of Melbourne, Parkville, Australia.

Thomas, L.C., 2003. The best banking strategy when playing The Weakest Link. Journal of the Operational Research Society 54, 747–750.