

# A MICROSOFT EXCEL VERSION OF PARRONDO'S PARADOX

HUMBERTO BARRETO  
hbarreto@depauw.edu  
DePauw University  
July 24, 2009

Parrondo's paradox is analyzed via Monte Carlo simulation and Markov chains within Microsoft Excel. The properties of individual and mixed games are clearly demonstrated. The accompanying Excel workbook, *Parrondo.xls*, available at <fs6.depauw.edu/~hbarreto/working>, enables the reader to replicate results, verify claims, and extend the analysis in unforeseen ways.

2000 Mathematics Subject Classification: 91A60, 65C05, 60J10.

SSRN Working Paper Version—submitted July 9, 2009. Version 2: July 24, 2009.

Do not quote without permission. Comments welcome.

## 1. Introduction

"Parrondo's paradox is the counterintuitive result where mixing two or more losing games can surprisingly produce a winning outcome. These games are named after the Spanish physicist, J.M.R. Parrondo, who inspired them." [1]

Game A is a simple coin flip. The player gets 1 unit of money ( $M$ ) for every head flipped and loses 1 unit of money if the coin comes up tails. If the coin is unbiased, so the probability of getting a head or tail is 50%, then the game is fair because the expected value is zero. If the probability of tails is greater than that of heads, the player will lose money, on average, and these losses mount the longer the game is played.

Game B is more complicated. The player is faced with two coins, one favorable and the other not. The unfavorable coin is used whenever the player's accumulated money is evenly divisible by three. It will be easy to see, when this game is implemented in Excel, that Game B, like Game A, can be a loser in the long run. Implementation of Game B will also enable discussion of what constitutes fairness in games.

Parrondo's insight, originally presented at a conference in Italy in 1996, is that by switching back and forth randomly from the two individually losing games, a player can end up winning, on average. This is called a paradox because it is so counter-intuitive—"it is, in fact, downright mystifying!" Nahin [11], p. 74. Parrondo's paradox will be demonstrated by Monte Carlo simulation and by casting the games as Markov chains in a Microsoft Excel workbook. Finally, the optimal combination of games A and B (even more favorable than random switching) will be determined.

## 2. Analyzing Game A via Simulation

Download the Excel workbook, *Parrondo.xls*, from [fs6.depauw.edu/~hbarreto/working](http://fs6.depauw.edu/~hbarreto/working) and open it. Be sure to enable macros when opening the workbook to obtain full functionality. If needed, see the instructions in the Word file *EnableMacros.doc* (available on the web site). If the buttons do not work, macros have not been enabled.

Proceed to the *DGP* (data generating process) sheet, a small portion of which is displayed in Figure 1. Click on cell B8 to see its formula, `=IF(RAND()<=$B$5,1,0)`. This formula enables flipping a virtual coin in Excel. The `RAND()` function is Excel's pseudo-random number generator. It mimics a uniform distribution on the interval from zero to one. To see `RAND()` in action, click on an empty cell and type in the formula, `=RAND()`, and hit the Enter key. Now, hit the F9 key repeatedly to recalculate the workbook and draw new random numbers in every cell that contains the `RAND()` function. Notice that the numbers in columns B and C also bounce as one presses the F9 key.

	A	B	C	D	E	F
1	Understabnding Parrondo's Paradox					
2					MC Sim	Game B
3	Game A					
4	epsilon	0.005	amount of bias in the coin			
5	Prob Heads	49.50%				
6						
7	Flip #	Result	End M			
8	1	0	-1			
9	2	0	-2			
10	3	1	-1			
11	4	0	-2			
12	5	0	-3			

Figure 1: Game A in the *DGP* sheet.

Cell B8 shows a one, representing heads, whenever the random draw is less than the probability in cell B5; otherwise, it shows a zero (or tails). Since the coin is a little biased, based on the parameter epsilon in cell B4, the player will lose money, on average.

Each cell in column C reports the player's money earned or lost after each flip so the amount of money at the end of the 100<sup>th</sup> flip gives us the final result. Scroll down to the 107<sup>th</sup> row (which shows the 100<sup>th</sup> coin flip) and click on cell C107 to see the formula used in cell range C9:C107. The formula shows that each time a head is tossed, the player's previous End M increments by 1 unit; while a tail results in a decrease of 1 unit. The chart shows how the game went, from the first to the last toss. Each press of the

F9 key plays a new game and sometimes the player wins (the ending value of M is positive) and other times loses (End M < 0).

The expected value of this game can be approximated by Monte Carlo simulation—play the game many times and record the results, and then compute the average amount of money obtained. To do this in

Excel, click the **MC Sim** button. In the dialog box, if needed, click in the Required, Select a cell input box, and then click cell C107. Click Proceed to play Game A 1,000 times. Your results should look like Figure 2.

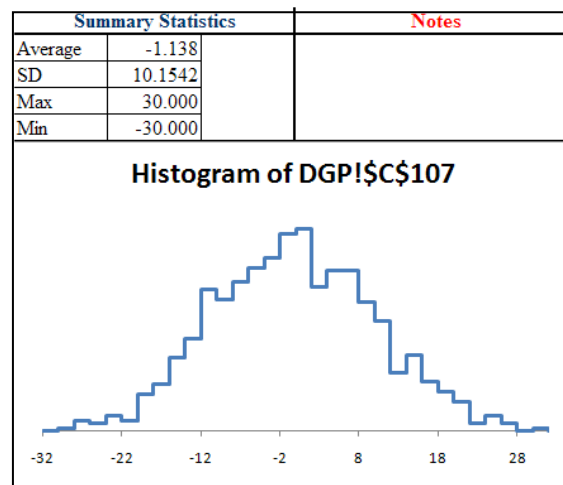
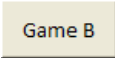


Figure 2: Monte Carlo simulation results of Game A.

In the particular 1,000 games played in Figure 2, the player lost a little over 1 unit of money, with the biggest win 30 units and the biggest loss 30 units. The SD, standard deviation, of 10.1542 is a measure of the variability in results. Your results will be slightly different because your random numbers and, hence, game results, are different.

There are analytical methods available to determine the exact, long run results. The expected value of Game A, after 100 flips, is simply  $(0.495 - 0.505) \cdot 100$ , or  $-1$ . The standard error is 9.995. Thus, the simulation results agree with the true, exact results. To improve the approximation, return to the *DGP* sheet and run another simulation, increasing the number of repetitions (say, to 10,000). As the number of repetitions increase, the average of the simulation converges to  $-1$  and the SD to 9.995. With an infinite number of repetitions (obviously impossible), the simulation would agree exactly with the results obtained by the expected value and standard error formulas.

### 3. Analyzing Game B via Simulation

To evaluate the behavior of Game B, click the  button in the *DGP* sheet. New columns appear. Game B is comprised of two coins, 1 and 2, which are played depending on the player's current capital. Figure 3 shows how the game is played.

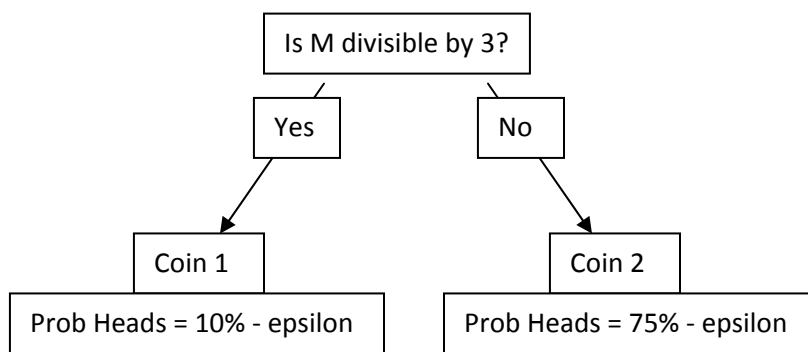
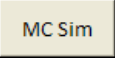


Figure 3: The rules of Game B.

Click on cell I8 to see its formula,  $=\text{MOD}(H8,3)$ . Excel computes the value of cell H8 (starting capital) modulo 3. Since zero is exactly divisible by three, the cell reports zero. The values in column I are zero, one, or two. If the player has 5 units of money, then  $\text{MOD}(5,3)$  is 2; while  $\text{MOD}(-2,3)$  is 1.

If the player's current capital is exactly divisible by three, then the unfavorable coin 1 is played. This coin wins only 9.5% of the time (with epsilon set at 0.005 in cell B4). The player gets to toss the favorable coin 2 (with a probability of heads of 74.5%) if current capital is not exactly divisible by three. Click on cell J8 to see how an IF statement is used to determine which coin is tossed.

Cell K8 determines the outcome of each round of Game B. The formula tosses the unfavorable coin if the player's capital is exactly divisible by three and uses the favorable coin otherwise:  $=\text{IF}(I8=0,\text{IF}(\text{RAND}()\leq\$H\$5,1,0),\text{IF}(\text{RAND}()\leq\$K\$5,1,0))$ . Column L keeps track of the total money earned or lost. The chart titled "Game B" (after row 107) shows the results of each individual game, from 1 to 100 flips. A second chart (below the Game B chart) shows the results of five games of B from a simulation—it does not update after the F9 key is pressed. A simulation with 10,000 repetitions (coming next) would produce 10,000 (instead of 5) series on a chart. Each series is called a sample function. Taking the average at each coin flip approximates the expected value at each of the 100 flips.

As with Game A, simulation can be used to evaluate this game, but instead of focusing on just the last cell, every toss will be tracked. Select the column of cells from L8 to L107, and then click the  button. In the Select a cell box, change the cell from L8 to L107, be sure to check the Record All Selected Cells option, and request 10,000 repetitions. Your results will be displayed in an *MCSim* sheet with a chart similar to Figure 4. In addition, an *MCRaw* sheet is produced. Because the simulation is writing

10,000 rows and 100 columns of data to the *MCRaw* sheet, it takes much more time (depending on the speed of your computer) to get the final results. The progress bar goes fairly rapidly, but it may take several minutes for the program to finish.

Simulation results make clear that Game B is a loser. On average, the player loses about – 1.4 units of money after 100 coin flips. This is worse than Game A, where the player could expect to lose 0.01 units of money on each toss, resulting in a loss of one unit of money, on average, in 100 tosses.

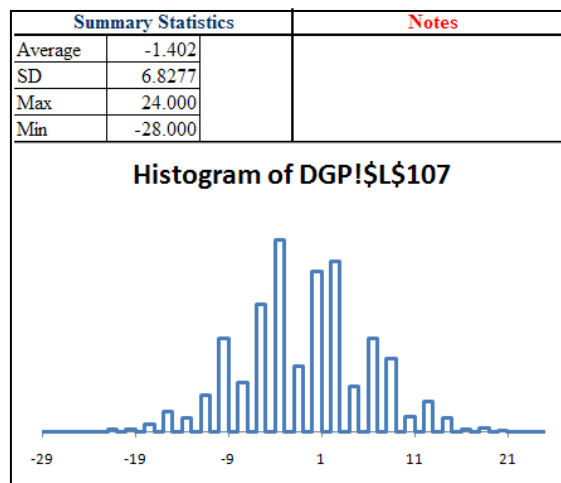


Figure 4: Simulating Game B.

The *MCRaw* sheet enables further analysis. The first row has a label for each cell. Each of the other 10,000 rows is a game (composed of 100 tosses). Each of the 100 columns reports the amount of money for a given flip in the sequence of 100 tosses for each of the 10,000 games.

Scroll to the bottom of your *MCRaw* sheet and in cell A10003, enter a 1. In cell A10004, compute the average of the first flip for all 10,000 games (with the formula =AVERAGE(A2:A10001)). The result should be about – 0.8. Starting with no capital, the unfavorable coin is played (because MOD(0,3) = 0). The expected value is (0.095 – 0.905) = - 0.81. Once again, the simulation agrees with the analytical result.

In cell B10003, enter a 2 and in cell B10004 compute the average of the 10,000 second flips. Select the cell range A10003:B10004, and fill right, all the way to column CV. (With cell range A10003:B10004 selected, move the cursor to the bottom-right corner of cell B10004 so that the cursor becomes a thin crosshair, then click and drag right.) The cells in row 10003 should run from 1 to 100 and the row below it should compute the average of the cells in rows 2 to 10001 above. Create an XYScatter graph (Scatter with Straight Lines) of the data in these two rows. The chart should look like Figure 5, which displays an ensemble average of the 10,000 sample functions. Each point plotted is the average of the 10,000 values at a given flip number.

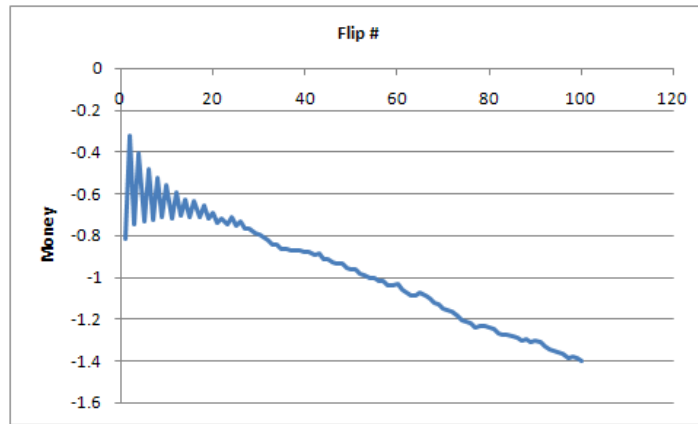


Figure 5: Simulation of evolution of Game B.

Figure 5 shows that Game B is a loser, but its behavior in early flips is erratic, oscillating until settling down to steady losses by around the 25<sup>th</sup> toss. This behavior can be compared with Game A by tracking all 100 tosses of Game A (by selecting cells C8:C107 before clicking the MC Sim button and checking the Record All Selected Cells option), then computing the average of each toss in the *MCRaw* sheet. Figure 6 is produced by adding Game A simulation results to Figure 5.

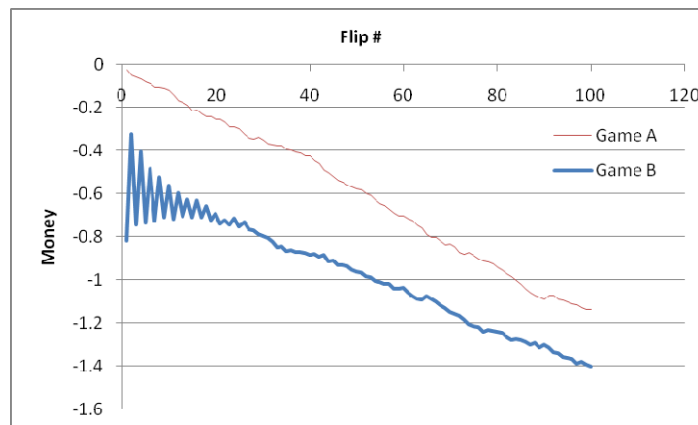


Figure 6: Comparing Games A and B.

Unlike Game B, Figure 6 shows that Game A steadily declines from the very first toss. Figure 6 makes clear that both are losing games. By the last toss, the player can expect to have lost – 1 with Game A and – 1.4 when playing Game B.

#### 4. Game B via Markov Chain

Click the Markov button at the top of column M in the *DGP* sheet to reveal the *MC* (Markov Chain) sheet, which shows the exact probabilities of Game B for each coin flip. Figure 7 shows the set up. The transition matrix shows the probabilities of moving from one state to another. The rows represent the three possible starting points, 0, 1, or 2, and the columns reflect the three possible ending positions, 0, 1, and 2. The diagonal values are zero because one cannot stay in the same position. With a capital position of 1 (the middle row), for example, a flip of tails (with probability 25.5%) will subtract one from M and move the player to  $M \bmod 3 = 0$  (the first column), while a flip of heads (with probability 74.5%) would add one and move the player to  $M \bmod 3 = 2$  (the third column).

Columns E, F, and G compute the evolution of the probabilities of being in the three possible positions by iteration, using Excel's MMULT array function. (If you click in one of the cells in the range E6:G107, you must hit the ESC key to exit the array formula.) Starting from an initial position of  $M \bmod 3 = 0$  (i.e., any M that is exactly divisible by 3), there is a 9.5% chance of being in position 1 for the next toss (a heads was flipped) and a 90.5% chance of being in position 2 (the result was tails).

	A	B	C	D	E	F	G	H	I	J
1	Analytical Solution via Markov Chain									
2									Initial M	0
3	epsilon	0.005			Mod(M,3)					
4	Transition Matrix: Game B				0	1	2		Flip #	Expected M
5	0	0.095	0.905		1	0	0		1	-0.810000
6	0.255	0	0.745		0	0.095	0.905		2	-0.320000
7	0.745	0.255	0		0.69845	0.230775	0.070775		3	-0.737985
8					0.111575	0.0844	0.804025		4	-0.393033
9					0.62052	0.215626	0.163854		5	-0.709709
10					0.177056	0.100732	0.722212		6	-0.449881

Figure 7: Set up of the *MC* sheet.

Column J computes the expected winnings at each toss by multiplying the probability of being in each state by the difference in the probability of heads and tails and adding the previous Expected M. This is illustrated by cell J8's formula,  $=F5*(\$B\$5-\$C\$5)+G5*(\$C\$6-\$A\$6)+H5*(\$A\$7-\$B\$7)+J2$ , which shows that starting from a capital position that is exactly divisible by three yields an expected value of  $-0.81$  units of money (a calculation computed previously). The expected amount won at each flip is computed the same way.

Figure 8 (which is the chart on the *MC* sheet) displays expected M as a function of the coin flip number. The results mirror the simulation (see Figure 5) and confirm that Game B is a loser and more money is

lost the more flips are played. By the 100<sup>th</sup> flip, the player has an exact expected value of -1.39232. To be clear, Figure 8 offers an exact, long run picture of Game B, while Figure 5, based on simulation, provides an approximation (which would converge to Figure 8 as the number of repetitions increased).

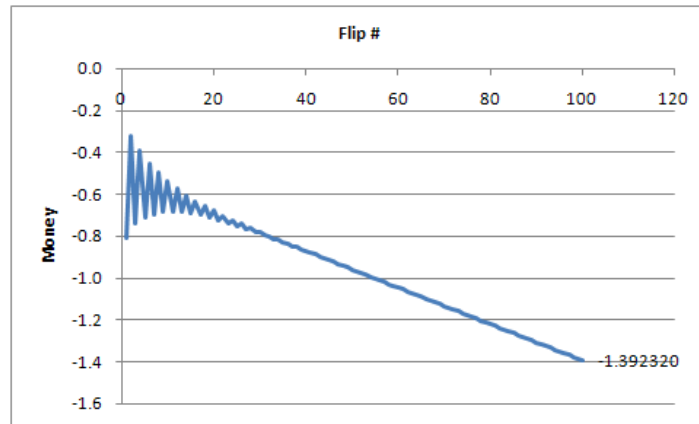


Figure 8: Exact evolution of Game B.

Scroll down to the last flip to see the probabilities after 100 flips. They are clearly converging to their steady-state values. To find the probabilities as the number of flips approaches infinity, the transition matrix along with the constraint that the probabilities must sum to one is used. Scroll down past the 100<sup>th</sup> flip to see the computations. After 100 flips, the probabilities are almost (but not exactly) equal to the steady-state probabilities.

The steady-state transition probabilities for positions 0, 1, and 2 are roughly 0.383611759, 0.154280573, and 0.462107668. In other words, when playing Game B, the probabilities of being in each position (0, 1, and 2) converge to stable values as the game is played. At these stable values, called the steady-state solution, there is a little higher than 38% chance of having an amount of money that is exactly divisible by 3 on a flip. The chances that current capital has a remainder of 1 or 2 when divided by 3 are roughly 15% and 46%, respectively. When current capital is divisible by 3, the extremely unfavorable coin must be tossed; otherwise, the favorable coin is used. Clearly, negative outcomes associated with the unfavorable coin swamp the positive results obtained from the favorable coin and the player ends up losing, on average.

Game A

Click the [Game A](#) button (at the top of column L) to see how Game A can be analyzed with a Markov chain. The transition matrix now reflects Game A's probabilities. Notice that the probabilities of moving up or down by one dollar are the same for each starting position. For Game A, the probability of heads is always 49.5%, regardless of the current capital position. The chart (and column J) shows that the player steadily loses 0.01 units of money, so that by the 100<sup>th</sup> flip, the expected value is -1 (which agrees with the simulation results in Figures 2 and 6). The steady-state solution of Game A is 1/3 for each position.



## 5. Fairness

Game A with  $\epsilon = 0.005$  is unfair. The expected value of the game (based on 100 coin flips) is  $-1$ , so the player loses, on average, 1 unit of money by playing this game repeatedly. This game can be made fair by playing with an unbiased coin. With Game A's transition matrix being used on the *MC* sheet (click the **Game A** button if needed), change the value of cell B3 ( $\epsilon$ ) in the *MC* sheet to 0 (zero) and Game A is now fair because the expected value after 100 flips is zero.

Does Game B behave like Game A when played with unbiased coin? With  $\epsilon$  set to zero in the *MC* sheet, click the **Game B** button to produce Figure 9.

A quick look at Figures 5 and 8 shows that with  $\epsilon = 0.005$ , expected  $M$  trended down as more flips were played. By comparison, the horizontal line in Figure 9 means that Game B with  $\epsilon = 0$  does not continue losing money as more flips are played. With  $\epsilon = 0$ , Game B is behaving like Game A in the sense that the player's expected  $M$  stabilizes as more flips are played. Unlike Game A, however, the expected value of Game B is negative—roughly half a unit of money is lost, on average, when 100 flips of this game are played.

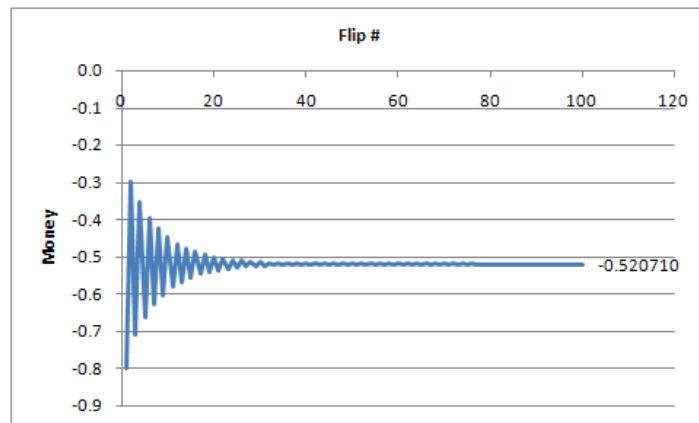


Figure 9: Game B with an unbiased coin.

With  $\epsilon$  set equal to zero and playing Game B, change the initial money amount from zero to one (in cell J2). The sheet and chart update, and the expected value at the end of the game is now positive. When starting the game with 1 unit of money, Game B with  $\epsilon = 0$  is favorable to the player, who wins, on average, roughly 0.55 units of money. The sheet makes clear why this happens. Instead of playing the unfavorable coin (with a probability of heads of only 25%), the game starts out (see cells E2:G2) with the player tossing the favorable coin. This seemingly small change carries through the rest of the game and yields a positive expected value.

Change cell J2 to 2, to see the effect of starting the game with two units of money. Once again, this game favors the player, but not by as much as starting with one unit of money, because the expected net gain is roughly 0.25 units of money, on average. Comparing rows 2 and 3 (in columns E, F, and G) while changing cell J2 from 1 to 2 shows the advantage of starting from 1 rather than 2 units of money. The first toss uses the favorable coin in either case, but the next toss shows that starting from one unit of money yields better odds for the player. If a head is obtained (with probability 75%) in the first toss, adding one unit of money to the starting value means that starting with two units of money will trigger use of the unfavorable coin in the next toss. Starting from one unit of money delays the use of the unfavorable coin.

Of course, these three examples exhaust the possibilities because any other initial value of money will begin the game from 0, 1, or 2, once Initial M modulo 3 is computed. We conclude then that Game B with  $\epsilon = 0$  is unfavorable whenever the player starts from any value evenly divisible by 3 (including starting from zero) and favors the player starting from any value of money not evenly divisible by three. In none of the three initial cases is the expected value of Game B zero. It is clear, however, that with  $\epsilon = 0$  Game B is like Game A no matter the value of initial M because it does not tend to positive or negative infinity as the game goes on. The computations at the bottom of the MC sheet show that if the player began the game with the steady-state probabilities of being in positions 0, 1, and 2, then Game B with  $\epsilon = 0$  has an expected value of zero.

The effect of the initial condition on Game B carries through to playing with a biased coin. To see this, set the value of  $\epsilon$  to 0.005 (in cell B3) and compare the expected value of the 100<sup>th</sup> flip when starting from zero, one, and two. Once again, starting from zero yields the worst outcome.

It is clear, however, that the value of initial M does not affect the qualitative character of the process. In the steady-state, the unbiased coin yields neither further losses nor wins, no matter the starting capital; while  $\epsilon > 0$  yields ever mounting losses, regardless of whether the game starts at 0, 1, or 2.

Simulation (from the DGP sheet) could be used to confirm these results. The initial capital can be set in cell H8 and cell L107 can be tracked. As expected, simulations support the exact results obtained via the Markov chain analysis.

## 6. Parrondo's Paradox

Game A with a biased coin is a loser. Its expected value after 100 tosses is  $-1$  units of money. Game B with a biased coin is also a loser. After 100 flips, starting from zero units of money, a player will lose about 1.4 units of money, on average. Keep playing beyond 100 flips and the losses will increase, on average. But what happens when the two games are mixed? Mixing can take place deterministically or randomly. We begin with the latter.

Return to the *DGP* sheet and click the **Parrondo** button (near the top of column M). Additional columns are revealed. Column O determines (with yet another unbiased coin flip) whether Game A or B is played. Columns P through S play Game B, based on the current value of M. Column T uses an IF statement to display the outcome of Game A or B. The last column keeps track of the money. Hit F9 repeatedly to see this randomly mixed game in action. The chart titled “Games A and B Mixed” at the bottom of the sheet shows the result of each game played (for each time F9 is hit and the workbook recalculates). Clearly, there is a great deal of variability in this randomly mixed game. What will happen, on average? Two approaches can be used to provide an answer: simulation and analytical methods.

Confirm that the value of epsilon is set to 0.005. Select the cell range from U8:U107, and then click the **MC Sim** button. In the dialog box, track cell U107 with 10,000 repetitions, and be sure to check the Record All Selected Cells option. Click Proceed. As before, writing 10,000 rows and 100 columns of results to a worksheet takes a few minutes.

The result of the simulation demonstrates Parrondo’s paradox. Whereas Game A has an expected value of  $-1$  and Game B has an expected value of about  $-1.4$ , the random mixture of these two losing games yields an expected value of about  $+1.2$ . “Are you surprised, perhaps even astonished, by the result from the second [mixture of Games A and B] simulation? Nearly everyone is, and that’s why this problem is called Parrondo’s paradox.” Nahin [11], p. 76.

Perhaps an even more striking visual of the paradox can be obtained by plotting the three ensemble averages on the same chart. In row 10004 of the *MCRaw* sheet, compute the average value of the 10,000 values of M, given the coin flip number. Then add these results to the chart with the simulation results of Games A and B (Figure 6), to produce a chart similar to Figure 10.

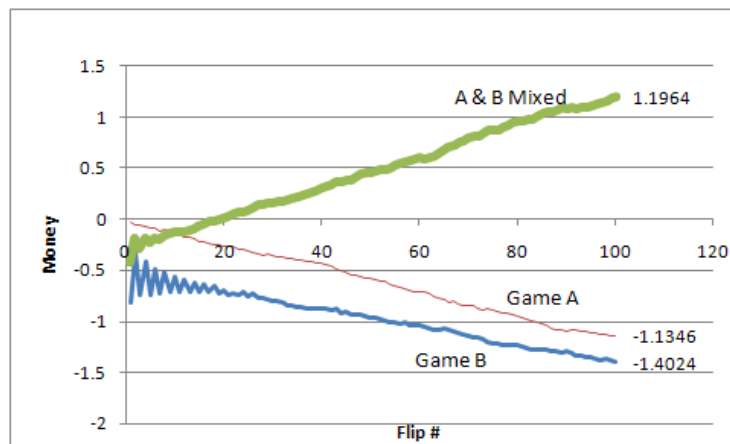


Figure 10: Signature graph of Parrondo’s paradox.

If the two games are played individually, Figure 10 shows that they trend downward; but randomly mixing the two games leads to an upward drift and, by the 100<sup>th</sup> flip, the expected value is about 1.2. “That is, randomly switching back and forth between two loser games is a winning strategy—and if you're not amazed by that, well, I find that even more amazing!” Nahin [11], p. 172.

To see the exact evolution of the stochastic process produced by randomly mixing Games A and B, return to the *DGP* sheet and click the Markov  
Parrondo button, near the top of column V. A new sheet, *MCParr* (Markov chain Parrondo), is displayed. The transition matrix mixes the transition matrices of Games A and B. Click on cell B5, for example, to reveal its formula,  $=0.5*(0.5-B3)+0.5*(0.1-B3)$ . Since the two games are played with equal probability, the transition matrix of this mixed game multiplies each of the probabilities of the two games by  $\frac{1}{2}$  and adds them together.

With the transition matrix specified, the rest of the sheet operates exactly as before. Figure 11, which charts the expected winnings at each coin flip of the randomly mixed games, shows the upward drift seen in the simulation results in Figure 10. Mixing the two losing games randomly results in a steady increase (after a few oscillations in the first few flips) in the expected value as the number of flips increases. That is quite surprising.

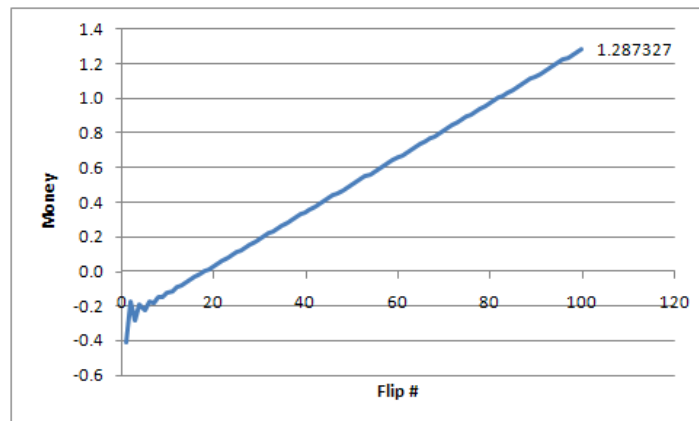


Figure 11: Exact evolution of randomly mixing A and B.

The probabilities of being in the three positions (0, 1, and 2), are displayed in columns E, F, and G, and provide a hint for explaining the paradox. The unfavorable coin is tossed whenever winnings are divisible by 3 and the probability of being in this position is given by column E. The first toss is most unfavorable (starting from zero or any initial M that is exactly divisible by three), but things immediately improve because the unfavorable coin is never tossed on the second flip. As the probabilities converge to their steady-state values, roughly 0.3451, 0.2541, and 0.4008 (scroll down to the bottom of the sheet), the expected value of each flip increases steadily and by the 100<sup>th</sup> flip, the player can expect to have about 1.287 units of money. At the steady state, the player makes about 0.016 units of money on every play (as shown in cell E141).

Notice that the steady-state probabilities are different in this mixed game than those obtained in Games A and B played individually. Table 1 shows the transition matrices, steady-state probabilities, and expected value of the next flip. Games A and B are losers, but the mixed game wins. Table 1 shows that the mixed game has a lower probability of being in a position where the player's capital is evenly divisible by zero (34.5% versus 38.5%) and the loss when in that state is much lower (0.41 versus 0.81). Game B is a loser because 38.5% of -0.81 plus 61.5% of 0.49 yields -0.01. The mixed game is a winner because 34.5% of -0.41 plus 65.5% of 0.24 gives +0.016.

Epsilon = 0.005			Steady-state probabilities			Next Flip
Game A			0	1	2	Expected M
0	0.495	0.505	0.333333	0.333333	0.333333	-0.01
0.505	0	0.495	33.3%	33.3%	33.3%	
0.495	0.505	0	lose 0.01	lose 0.01	lose 0.01	
Game B						
0	0.095	0.905	0.384615	0.153846	0.461538	-0.0100
0.255	0	0.745	38.5%	15.4%	46.2%	
0.745	0.255	0	lose 0.81	win 0.49	win 0.49	
Randomly Mixed						
0	0.295	0.705	0.34507	0.254108	0.4008216	0.0157
0.38	0	0.62	34.5%	25.4%	40.1%	
0.62	0.38	0	lose 0.41	win 0.24	win 0.24	

Table 1: Comparing steady-state outcomes.

Parrondo's paradox is surprising because we fail to take into account the way the two losing games combine to form the third game. Instead of a simple summation of a loss and a loss, Table 1 shows that the combined game drastically reduces the expected value of the capital evenly divisible by three position from -0.31 (38.5% of -0.81) to -0.14 (34.5% of -0.41). This is the key to understanding the paradox.

For a captivating visual explanation of the paradox, see Bogolmony's Java applet [5], available at <http://www.cut-the-knot.org/ctk/Parrondo.shtml>. It makes clear the idea of a ratcheting effect, which is often invoked as a physical analogue to explain Parrondo's paradox.

## 7. Optimal Mixing

Other ways of mixing the two games are possible. Return to the *DGP* sheet, click the ABAB option in the list near the top of column O. This changes the games in column O from randomly playing A and B to

playing games A and B in sequential order. If the flips are played this way, what is the expected value after the 100<sup>th</sup> flip? Run a simulation to find out.

What about mixing the two games by playing A twice, then B twice, sequentially? Click the AABB option, and then run a simulation. Figure 12 adds simulation results of ABAB and AABB sequences to Figure 10. The winner so far is AABB, but can we do even better?

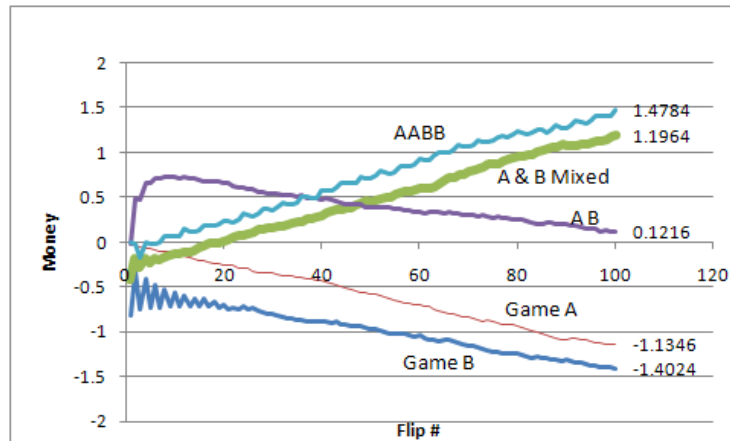


Figure 12: Simulation results of various games.

To see the exact evolution of the expected value of any sequence of mixes, click the  button near the top of column V in the *DGP* sheet. A new sheet appears called *MCParrOpt* (Markov chain Parrondo Optimal) that is the same as the *MCParr* sheet, except that column D contains which game is to be played.

The sheet opens with the ABAB mix. The non-linear shape obtained in the simulation is confirmed (it is more exaggerated in the chart on the *MCParrOpt* sheet because the y axis scale is different). AABB can also be found by entering these values in cells D5:D8 and then filling down. As expected, the simulation results (series AABB in Figure 12) are confirmed. Return the sheet to the initial AB mix.


Column D can be used to determine the evolution of any mix of A and B games. Change cell D5 from A to B to see the result of playing two Bs first, then alternating A and B. This mixed game is now a loser because the expected value of the 100<sup>th</sup> flip (reported in the chart and cell N21) is negative. Return cell D5 to A and change cell D6 to A. This is not a good idea because the expected value of Final M falls.

The outcome of playing one game versus another can also be seen in column J. With cell D5 = A, switch cell D6 back and forth from A to B and watch cell J6. Clearly, it is better to play Game B on the second toss because the expected value after that toss is higher.


In fact, the expected value of the second toss is positive! Play A twice and the expected value on the second flip is  $-0.02$ . Play B twice and the expected loss is  $-0.32$ . But play AB and the expected value is  $+0.48$ . That is Parrondo's paradox in a nutshell. Why is AB positive while AA and BB are negative? The transition matrices and probabilities of being in positions 0, 1, or 2 hold the key. It is easy to see that A beats B at the beginning because, starting from  $M = 0$ , B forces the player to use the really unfavorable coin. In the second toss, playing B has a positive expected value because it is guaranteed that the favorable coin will be used. By playing Game B when it is likely that  $M$  is not exactly divisible by three, the player uses the favorable coin and the expected value rises.


Considering just two tosses shows that there is an inverse Parrondo's Paradox—an even bigger loser game can be made by combining two loser games. Play BA and you will do even worse than playing just A or B twice.

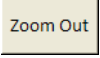
Armed with the knowledge that we can pick the better game to play based on the expected value of that flip, the following strategy can be implemented: Simply walk down column D and test whether A or B

has higher expected value. The  button runs a macro that does exactly this. Here is the Visual Basic code (with explanatory text) associated with this button (right-click the button, select Assign Macros, and click Edit to see the actual code):

For I = 1 To 100	Begin a loop with I = 1 for the first flip
myGame = Cells(I + 4, 4).Value	Get the initial game played in cell D5 (row 5, col 4)
myEV = Cells(I + 4, 10).Value	Get the initial EV of the game played in cell J5 (row 5, col 10)
If myGame = "A" Then	
Cells(I + 4, 4).Value = "B"	
Else	Use an IF statement to change the game being played
Cells(I + 4, 4).Value = "A"	
End If	
If myEV > Cells(I + 4, 10).Value Then	Use an IF statement to test if the initial EV is bigger
Cells(I + 4, 4).Value = myGame	than the current EV and if it is, replace the current
Else	game with the initial one; if it is not, do nothing.
End If	
Next I	Increment I by one to find the better game for each flip

Click the  button to see that starting with AB, and then playing ABB repeatedly gives an expected value after 100 flips of about 6.14 units of money. That is much better than random mixing or the AABB sequence.

The chart of the expected value at each flip produced by the sequential choice algorithm reveals a staircase-like pattern that is a hallmark of the Parrondo paradox. Click the  button (below the chart) to see this more clearly. The sequence ABB is drifting upwards, but not by equal increments on each flip. In the steady-state, playing A brings M down by 0.01 units of money, but then B increases M and the next B increases M again (but not by as much). The cycle is then repeated and the player's capital climbs ever higher by taking one step down and two bigger steps up. The random mixing of games A and B that yields a winning game out of the two individually losing games does not take optimal advantage of this ratcheting effect. However, random mixing plays Game B in positions where  $M \bmod 3$  is less likely to not be zero often enough to produce an overall positive outcome—which is called the Parrondo paradox.

While picking the higher expected value from A or B at each flip, yielding the sequence AB for the first two flips, then ABB repeatedly, is better than ABAB or AABB, there is an even better strategy. Click the  button to return the chart to its original position. Select the cell range D5:D9 (the sequence ABABB) and fill it down (by moving the cursor to the bottom-right corner of cell D9 so that the cursor becomes a thin crosshair and double-clicking). The expected value of the 100<sup>th</sup> flip in this sequence is 6.845. This is the maximum expected value. For a formal derivation of this optimal sequence, see Dinis [6]. Switching cell D12 from A to B shows that expected M at the 8<sup>th</sup> flip rises (which is why B is chosen by the sequential choice algorithm), but the expected M at the 100<sup>th</sup> flip (in cell N21 and on the chart) falls. The short term improvement is not worth it—the sequence that maximizes the expected value at the 100<sup>th</sup> flip cannot be found by simply choosing the higher expected value of A or B at each flip.

## 8. Conclusion

Parrondo's paradox has been intensively studied ([1], [2], [4], [6], [8], [9], and [12]). The results presented here are not new, but the exposition, using Microsoft Excel, is novel. A stochastic process can be analyzed with a variety of software and Parrondo's paradox can be seen via applets on a web page, e.g., Bogomolny [5] and Spector [13], as a Maple program, Ekhad and Zeilberger [7], and a Mathematica notebook, Vellman and Wagon [15], but there are advantages to using a spreadsheet. The reader can see each formula to better understand how individual games are played and, for example, how Markov chains can be used to determine the probability vector at a given flip number. The reader can replicate each graph and verify claims, for example, about the consequences of using a biased versus an unbiased coin.



Perhaps most importantly, the reader can explore the effects of new parameter values and extend the analysis in unforeseen ways. There are obvious questions to ask, such as, “What happens if we vary epsilon?” or “What is the effect of the parameters 10% and 75% in Game B?” And what about modulo 3—does the paradox survive in modulo 2 or 4 or any other number? But perhaps you have other questions in mind? Modify the workbook as needed and use simulation or the Markov chain (if possible) to find an answer.

The MCSim button in the *Parrondo.xls* workbook is freely available as an Excel add-in from the web site for Barreto and Howland’s *Introductory Econometrics* [3] at <[www.wabash.edu/econometrics](http://www.wabash.edu/econometrics)>. This add-in enables simulation of any stochastic process implemented in Excel.

The literature on Parrondo’s paradox is quite extensive and growing. In fact, it has even made it into a novel:

“Her luck’ll change. You’ll see. Nobody wins all the time,” the obnoxious brute proclaimed.

“If you’re really interested in winning, maybe you should try table hopping,” she suggested, hoping he might take the hint.

“Why the hell for?” the rude brute stammered. As if he were inhaling it, Dick drunk deeply of a beer. It was a chaser to the shot of Wild Turkey that had fortified this gamble and steadied his hand. Ten out of ten casinos would recommend such an action.

“A Michigan State study revealed that by playing two losing games there is a ratcheting effect that enables players to win.”

“So what are you trying to say?”

“They did this despite playing in two games where the odds were stacked against them. They dubbed it Parrondo’s Paradox.”

“Sounds like BS,” Dick the tasteless brute burped.

Hill [10], p. 2.

Of course, Parrondo’s paradox does not say that randomly mixing any two losing games automatically yields a winner, so the paradox is not a secret for success in a casino. It is clear that under certain conditions, however, the paradox does hold and a winning game can be created from two losing games.

The author gratefully acknowledges financial support from the Elizabeth P. Allen Distinguished Professor fund at DePauw University and helpful comments and corrections from Frank M. Howland and Yavor Kovachev.

## References

- [1] Adelaide Parrondo's Paradox Group's home page, welcome message at [www.eleceng.adelaide.edu.au/Groups/parrondo/index.html](http://www.eleceng.adelaide.edu.au/Groups/parrondo/index.html).
- [2] P. Amengual, P. Meurs, B. Cleuren, and R. Toral, "Reversals of chance in paradoxical games," *Physica A* 371 (2006), 641–648.
- [3] H. Barreto and F. M. Howland, *Introductory Econometrics: Using Monte Carlo Simulation with Microsoft Excel*, Cambridge University Press, 2006.
- [4] G. C. Berresford and A. M. Rockett, "Parrondo's paradox," *International Journal of Mathematics and Mathematical Sciences*, Vol. 2003 (2003), Issue 62, 3957-3962. doi:10.1155/S0161171203301425, [www.hindawi.com/journals/ijmms/2003/321921.abs.html](http://www.hindawi.com/journals/ijmms/2003/321921.abs.html).
- [5] A. Bogomolny, "Parrondo Paradox," June 2001, [www.cut-the-knot.org/ctk/Parrondo.shtml](http://www.cut-the-knot.org/ctk/Parrondo.shtml).
- [6] L. Dinis, "Optimal sequence for Parrondo games," *Physical Review*, E 77, 021124, 2008.
- [7] S. B. Ekhad and D. Zeilberger, "Remarks on the Parrondo Paradox," unpublished, undated, <http://www.math.rutgers.edu/~zeilberg/programs.html>.
- [8] S. N. Ethier and J. Lee, "Limit theorems for Parrondo's paradox," 2009, [arxiv.org/abs/0902.2368v1](http://arxiv.org/abs/0902.2368v1).
- [9] G. P. Harmer and D. Abbott, "Parrondo's Paradox," *Statistical Science*, Vol. 14, No. 2 (May, 1999), 206-213.
- [10] W. Hill, *Vegas Vampires*, Otter Creek Press (2005).
- [11] P. J. Nahin, *Digital Dice: Computational Solutions to Practical Probability Problems*, Princeton University Press, 2008.
- [12] J. M. R. Parrondo, Parrondo's paradoxical games, [seneca.fis.ucm.es/parr/](http://seneca.fis.ucm.es/parr/).
- [13] L. Spector , On-line Simulator for Parrondo's Paradox, [hampshire.edu/l spectator/parrondo/parrondo.html](http://hampshire.edu/l spectator/parrondo/parrondo.html).
- [14] D. Velleman and S. Wagon, "Parrondo's paradox," *Mathematica in Education and Research* 9.3-4 (2001): 85-90.