INVESTIGATING THE ST. PETERSBURG PARADOX WITH MAPLE

Dr. Joel C. Fowler
Mathematics Department, Southern Polytechnic State University
1100 South Marietta Parkway
Marietta, GA 30060-2896
jfowler@spsu.edu

The Saint Petersburg Paradox is a well known illustration of how the psychology of risk and reward can diverge from the underlying mathematics. In the standard game a fair coin is tossed repeatedly until the first head occurs. The payoff is \$1 if the game terminates in one flip, \$2 if in two flips, \$4 if in three, and so on, doubling the payoff for each additional flip required to obtain the first head. The basic analysis of the game is a straightforward computation involving expected value, which turns out to be infinite (despite the fact that few individuals are willing to pay more than a few dollars to play). When one looks more closely at the game from the standpoint of the likelihood of making a profit after many repeated plays, however, more complex and interesting questions and computations arise. We use Maple to investigate these and other questions relating to the Saint Petersburg Paradox. This approach sheds some light on the nature of the paradox, because the human psychology of risk is often connected more closely to the probability of loss, rather than the somewhat more involved notion of expected value. The computations associated with finding exact probabilities are intractable enough that the computational power of Maple is a necessity. We also use Maple in our investigations to simulate repeated plays of the game, and check these against the theoretical probabilities. The mathematics and Maple used is at the level of an undergraduate special project.

For the standard game, with a payoff of $\$2^{n-1}$ when the first head occurs on the n-th flip, we may easily find the expected payoff as:

$$1\left(\frac{1}{2}\right) + 2\left(\frac{1}{4}\right) + 4\left(\frac{1}{8}\right) + 8\left(\frac{1}{16}\right) + \ldots = \sum_{n=0}^{\infty} 2^{n-1}\left(\frac{1}{2^n}\right) = \sum_{n=0}^{\infty} \frac{1}{2} \ ,$$

which is a divergent series. Hence the expected value is considered to be infinite. The paradox is that few people are willing to pay more than a few dollars to play this game, even after the expected value computation is presented. The question is why the human psychology of the game is so far from the mathematics.

For example, compare the game with the following: With a cost to play of \$8, flip a fair coin repeatedly until the first head is obtained. If the first head is obtained on the n-th flip, the payoff is:

For n = 1, 2, 3, 4, 5 :	\$ 0;
For n = 6, 7, 8 :	\$ 16;

For n = 9, 10, 11 :	$ 28;
For n = 12, 13, . . . 18 :	$ 400;
For n = 19, 20, 21, 22 :	$ 40,000;
For n = 23, 24, 25, 26, 27 :	$ 4,000,000;
For n > 27 :	$ 50,000,000;

Comparing this with the doubling payoffs of the St. Petersburg game, starting with $1, shows that this game is vastly less favorable to the player. However, thousands of people choose to play the mathematical equivalent of this modified game every day. It is a representation of 4 plays of the Georgia Lottery Power Ball Game in which 5 numbers are chosen from 1 - 59, plus one "powerball" number from 1 - 35.

Many explanations that have been offered to explain the paradox. I shall focus on estimating the likelihood of a profit after N plays of the game, for a fixed cost to play per game, with assistance from Maple.

First we load the packages we shall use.

```
> restart:
> with(RandomTools); with(plots); with(Statistics):
```

Using the built in Random Variable features of Maple we can simulate play very efficiently.

```
> X := RandomVariable(Geometric(.5));
> Sample(2^X,10);
        [1., 4., 8., 2., 4., 2., 1., 2., 1., 2.]
```

Finding the average return per play for the simulation is a simple computation.

```
> ExpAve := N -> add( Sample(2^X-1,N)[i], i=1..N)/N + 1:
> ExpAve(100);
                5.250000000
> ExpAve(500);
                5.552000000
> ExpAve(1000);
                6.822000000
```

We see that the growth in average payoff appears to be slow. For larger numbers of games the speed of the simulation can be increased by dividing into groups of 100. We define the procedure:

```
> ExpAveBig := proc(N::integer)        description "find an experimental average payoff
for N plays of the game, computed in groups of 100";
      Fl := floor(N/100); Temp := 0;
```

```
    for j from 1 to Fl do Temp := ( 100*(j-1)*Temp + add(Sample(2^X-1,100)[i],
   i=1..100) + 100 )/(100*j)      end do;
  if modp(N,100)<>0 then (Fl*100*Temp + ExpAve(N-100*Fl)*(N - Fl*100) )/N else
  Temp end if;
end proc;
```

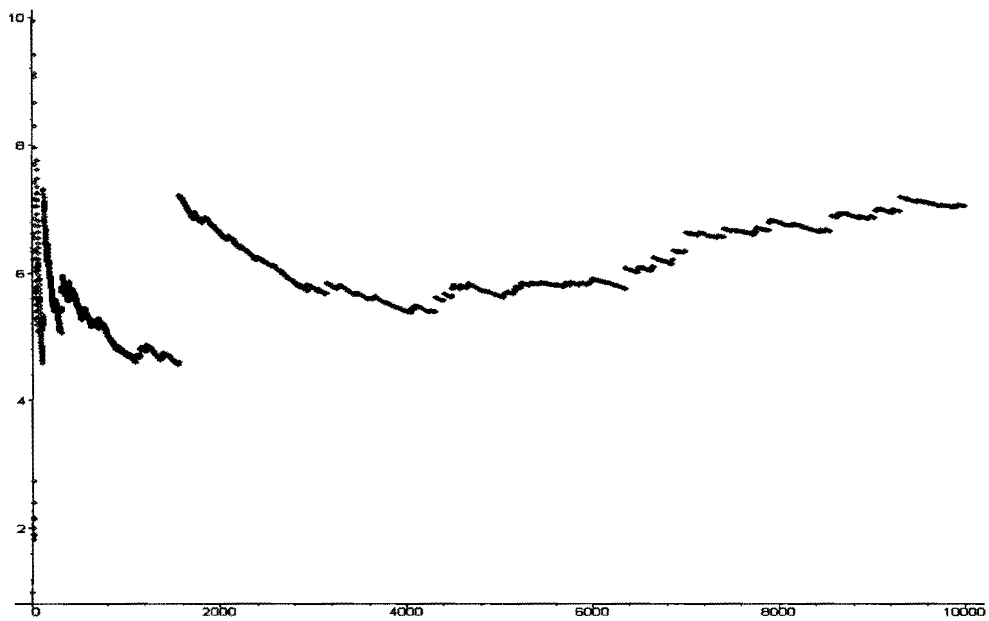
```
> st:= time(): ExpAveBig(100000); time() - st;
                    8.609750000
                      41.984
```

Thus we find that simulating 100,000 plays of the game requires roughly 42 second of Maple computing time, and yields an average payoff of only 8.6 . A reasonable question is how typical these average payoffs are. We can simulate 20 size 1000 sessions.

```
> seq(evalf[5](ExpAveBig(1000)),i=1..20);
 7.6230, 6.4400, 4.9190, 6.0950, 8.6553, 4.4740, 3.9030, 8.5857, 26.872, 4.3150, 10.311,
6.4820, 4.1770, 4.5820, 5.4460, 5.8840, 6.0260, 6.2260, 4.5850, 13.099
```

It appears that it will take a large number of plays to recoup our losses, even when paying as little as $10 per play. We can graph the behavior of the simulated average payoff over time for 10000 plays.

```
> N := 10000; Values := Sample(2^X,N): RunningAverage := i -> add((Values[j]),j=1..i)/i:
pointplot({seq([n,RunningAverage(n)],n=1..N)});
>
```



The simulations serve to show:
1) How unstable the behavior of the typical average payoff per game appears to be.
2) How small the typical average payoff per game appears to be, even for large n.
3) While the expectation is infinite, making a profit appears to be relatively unlikely, even for paying modest amounts to play.

We will now focus on examining the likelihood of profit. That is, for various costs per play, how does the probability of achieving a particular average payoff change as the number of games increases. First consider the exact probability of earning a total of at least M after N plays of the game. From this we can easily compute the probability that the average payoff exceeds various amounts.

Let P(M, N) be the probability of a gross total payoff of at least M after N games are played. Hence for N > 1 we have

$$P(M,\ N) = (\frac{1}{2})P(M-1,N-1) + (\frac{1}{4})P(M-2,N-1) + (\frac{1}{8})P(M-4,N-1) + \ldots$$

$$\ldots. + (\frac{1}{2^k})P(M-2^{k-1},N-1) + \frac{1}{2^k}\ ,$$

$$\text{and}\quad P(M,\ 1) = \frac{1}{2^k},\quad \text{where } k = \mathrm{ceil}(\log_2 M)\,.$$

This recursive relationship is easily implemented in Maple.

```
> MaxM := 512 ; MaxN := 100;
> for m from 1 to MaxM  do CompProb(m,1) := 1/(2^(ceil(log[2](m)))) end do:
   for n from 2 to MaxN do
    for m from 1 to MaxM do
      CompProb( m, n ) := add( CompProb(m-2^i,n-1)/(2^(i+1)) ,i=0..(ceil(log[2](m))-1)
                                         )+1/(2^ceil(log[2](m)))
    end do:
  end do:
```

And so we have some sample values:

```
>[evalf(CompProb(32,4)),evalf(CompProb(256,32)),evalf(CompProb(256,64)),evalf(Co
mpProb(512,32))];
    [0.1262207031, 0.1551353768, 0.5582247605, 0.06451659038]
```

We can define the average payoff likelihood function:

ProbAve(A, N) = Prob(average payoff is at least A after N plays) = P(A N , N) , and implement it in Maple.

```
> ProbAve := (m,n) -> CompProb(m*n,n);
```

We may now, for example, examine how likely it is to earn an average payoff of at least 10 as N increases.

```
> Average := 5;  for i from 1 to 10 do  10*i, evalf(ProbAve(Average,10*i)) end do;
                         Average := 5
                        10, 0.2508370436
                        20, 0.2978779218
                        30, 0.3183649276
                        40, 0.3476678720
                        50, 0.3657424617
                        60, 0.3792860107
                        70, 0.3934439328
                        80, 0.4081270388
                        90, 0.4226036559
                       100, 0.4353082165
```

Unfortunately, the growth of the numbers makes an exact approach unproductive. For asymptotic behavior we will use the Central Limit Theorem. For convenience we will let:

the number of plays of the game $= N = 2^n$, for some integer n,

and

the minimum desired average payoff per game $= M = 2^m$, for some integer m.

Now for $i = 1, 2, \ldots, N = 2^n$ let

$$X_i = \text{payoff of the i-th game} .$$

Hence, if $\bar{X}$ is the sample mean for these N random variables we have:

$$P(\bar{X} \geq M = 2^m) = P\left(\frac{X_1 + X_2 + \ldots + X_N}{2^n} \geq 2^m \right) = P(X_1 + X_2 + \ldots + X_N \geq 2^{n+m}) .$$

Unfortunately, since each X_i has an infinite mean, the Central Limit Theorem cannot be used directly. For each $i = 1, 2, \ldots, N = 2^n$ we let:

$$Y_i = X_i, \text{ if } X_i < 2^{n+m}; \quad X_i = 2^{n+m}, \text{ otherwise} .$$

Letting $\bar{Y}$ be the sample mean of these N random variables yields:

$$P(\bar{X} \geq 2^m) = P(X_1 + \ldots + X_N \geq 2^{n+m}) = P(Y_1 + \ldots + Y_N \geq 2^{n+m}) = P(\bar{Y} \geq 2^m)$$

Each Y_i has a finite mean and variance which are easily computed as:

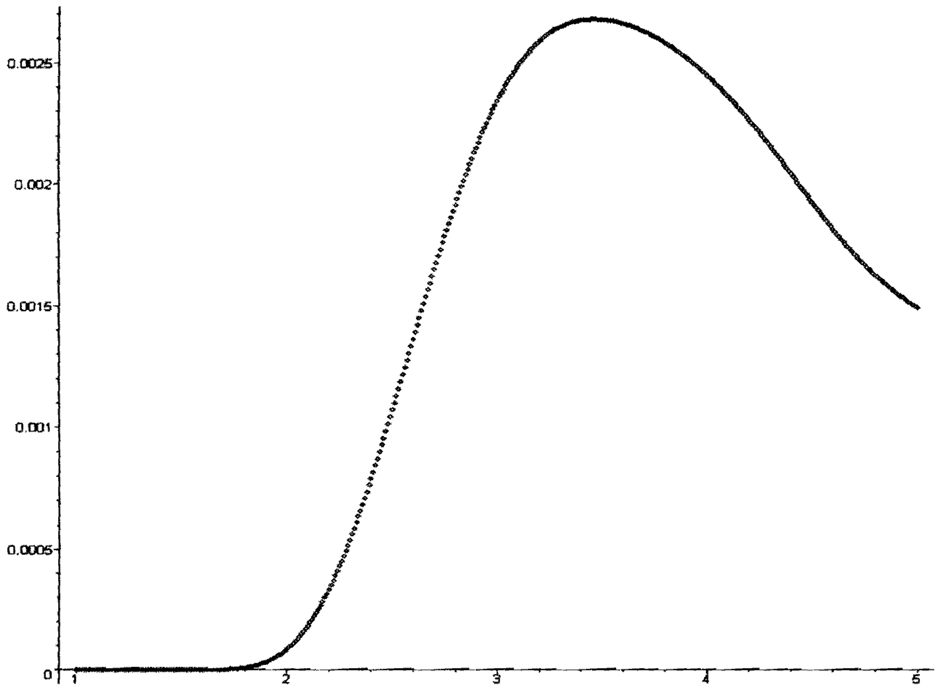
$$\mu = \frac{n+m+2}{2}, \quad \sigma^2 = 3(2^{n+m-1}) - \frac{(n+m+2)^2+2}{4} .$$

Using these values, setting up the normal approximation in Maple, and adjusting for when M and N are not powers of 2, gives:

```
> NormalApprox := (M, N) -> Probability( RandomVariable(Normal(0, 1)) > (M - (ceil(log[2](M*N))+2)/2)/sqrt( ( 3*2^(log[2](ceil(M*N))-1) - ( (log[2](ceil(M*N))+2)^2 + 2 )/4 ) / N ) , 'numeric' );
```

In general, the normal approximation is considered appropriate for N > 30, and to be best near the mean. Before working more with the approximation, we examine the distribution we're approximating for N = 100.

```
> N:= 100 : Max := floor(500/N): DistCheck := i -> CompProb(N*i,N)-CompProb(N*(i+1/N),N) : pointplot({seq([n,DistCheck(n)],n=1..Max, 1/N)});
```



For N = 100 and M = 4, the normal approximation has a mean of 5.5, which is to the right of the maximum value of the actual distribution. Hence, because our probabilities are for the likelihood of exceeding a particular average value, we would expect the normal approximation to be higher than the actual value. We can see this by comparing:

```
> NormalApprox(4,100); evalf(ProbAve(4,100));
                    0.7348778558
                    0.6298449494
```

If we do place a small amount of trust in the normal approximation, work near the mean, and treat the computations as providing a very rough estimate, then we may approximate the number of games needed to raise the probability of a profit to at least .5 . Recall that we have the mean of our approximating normal as $\frac{n+m+2}{2}$, where the number of games = $N = 2^n$, and the desired average payoff per game is $M = 2^m$. Using the normal, we have that the probability of being above the mean is .5 . Hence, for a fixed value of $M = 2^m$, the number of games, $N = 2^n$ required to raise the probability above .5 occurs when:

$$M = 2^m > \frac{n+m+2}{2} \quad .$$

This is easily solved for n to produce:

$$n > 2^{m+1} - m - 2$$

Hence, based on this approximation, the number of games required to raise the probability of a profit to around .5, when paying $M = 2^m$ per play is around:

$$N = 2^n = 2^{2^{m+1}-m-2} \approx 4^M$$

Unfortunately, it is impossible to check this approximation against exact values because N grows so quickly. It does, however, shed some light on the paradox. Many people's reactions to the claim that, in the long run, they would make money paying any amount to play (based on the infinite expected value) is to question how long that would take, since large payoffs are rare. In fact, based on this analysis, it would take a very long time. For example, if paying \$25 per play, and using Maple to streamline the game playing (at a rate of 1000 games per second) we would need to play continuously for approximately:

```
> evalf((4^25)/(1000*60*60*24*365) );
                    35702.05184
```

YEARS, in order to at least break even, with a probability around .5 .

So, in some sense, there is no St. Petersburg paradox. The intuition that it would take an unacceptably long time to realize a profit is correct. Of course, it is worth remembering that this analysis is based on our normal approximation, which certainly deserves a better justification. This will be the subject of future research.

We close with several interesting variations on the St. Petersburg Paradox. The original version of the game can be modeled by making repeated draws, with replacement, from an urn with one red and one green ball. If the n-th draw is the first time a green is drawn, then the payoff is 2^{n-1}. By modifying these rules we may produce several other games.

For each of the following:
1) The urn begins with one green and one red ball.
2) Each drawing is a single ball and done with replacement.
3) Drawing continues until the first green ball is drawn from the urn.
4) The payoff is simply the number of draws (no doubling of payoff values).
Now consider the following games:.

Game A:
Each time a red ball is drawn (and so the game does not end), one green ball is added to the urn, in addition to the replacement of the red ball drawn.

Game B:
Each time a red ball is drawn (and so the game does not end), one red ball is added to the urn, in addition to the replacement of the red ball drawn.

Game C:
Each time a red ball is drawn (and so the game does not end), it is replaced, but no other balls are added to the urn.

Before reading on, purely from an expected value standpoint, what preference order do these games have and how much would you be willing to pay to play each?

Game A: These conditions yield a probability function of: $P(n) = \frac{n}{(n+1)!}$, where n is the number of draws the game lasts . Using the series for e^x yields an expected value of
$e - 1 \sim$ \$1.72. Interestingly, if one doubles the payoff per draw (1, 2, 4, 8, . . .), the average payoff still only rises to $\frac{e^2+1}{4} \sim$ \$2.10 .

Game B: These conditions yield a probability function of: $P(n) = \frac{1}{n(n+1)}$, where n is the number of draws the game lasts. The expected payoff is then the harmonic series, which diverges. Hence the expected value is infinite, just as it is for the St. Petersburg paradox. However, the average payoff growth rate here is even slower.

Game C: This yields a probability function of: $P(n) = \frac{1}{2^n}$, where n is the number of draws the game lasts. Using the geometric series yields an expected value of $2.00.