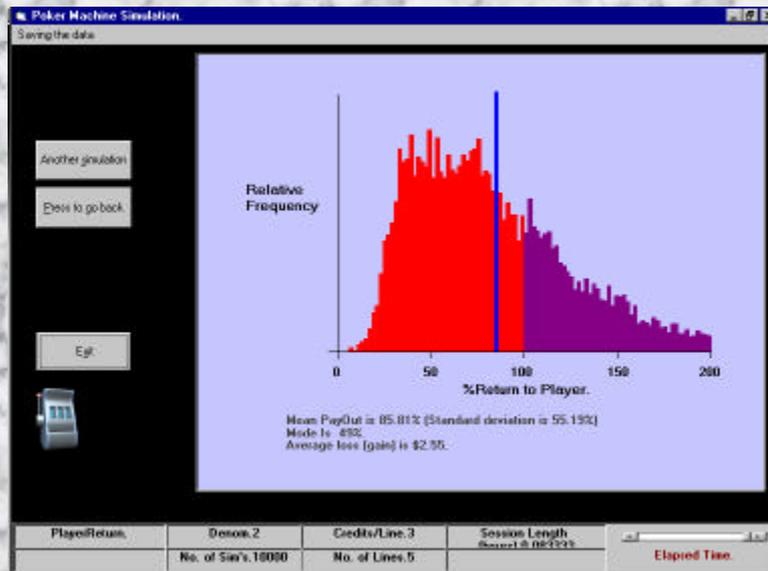




*Simulation software for an
Australian Electronic Gaming
Machine*

PC EGMSim



PC EGMSim User's Manual

Written and designed by S&R Lattimore for the Productivity Commission's inquiry into *Australia's Gambling Industries*.

Version 2.0, November 1999

Freeware

Contact: rlattimore@pc.gov.au

Contents

Contents	2
Acknowledgments	4
1 Introduction	5
2 Loading the software	6
3 Getting started	6
4 Duration of a gambling session (Ctrl D)	7
5 The player rate of return and player losses	10
6 The cost of gambling	13
7 Simulation specifications	16
Machine denomination (2 cents, 5 cents or 10 cents)	16
Number of lines (1, 3, 5, 7 or 9)	16
Credits per line (1, 2, 3, 5, or 10)	16
Number of simulations	16
Amount outlayed - Initial amount (\$)	17
End amount (\$) — prize amount prompting a stop in play	17
Time limit	17
Session duration (hours)	18
Sessions per month	19
8 Saving the results from the program	19
9 To get help	21
10 Chebyshev's inequality	21
11 Some examples	22
References	28

Acknowledgments

The Productivity Commission wishes to thank Aristocrat Leisure Industries for providing information about the game on which the simulation is based as well as trialing the software. They are not responsible for any errors in the software or its documentation.

1 Introduction

This program simulates a typical Australian electronic gaming machine. There are two kinds of game simulators:

- games that mimic the playing of a gaming machine or poker machine in real time, with realistic looking slots and even sound; or
- software that reveals the outcomes of many thousands of games in a few seconds.

The first — like the physical gaming machines — are mainly for fun. The second is a tool for examining how outcomes vary with playing style — and to explain better the nature of gaming machines. This simulator is in the second category.¹

Australian gaming machines typically involve no skill and provide returns based on chance alone. Nevertheless, they are rather complicated devices, with complex payoffs that many consumers are not well informed about. It can take a long time to discover the payoff structure of a gaming machine — so long that a person will typically have lost a lot of money to do so. Simulation allows a person to investigate the payoff structure associated with a fairly typical machine (for free). This software compresses thousands of hours of playing time into a few seconds. For example, to simulate the outcomes of 1000 people playing 3 lines on a gaming machines, each for a one hour session of play, takes less than ten seconds on a 266 Mhz Pentium II.

Some consumers and gambling providers say that players can work out the odds by playing the machines. Certainly, this is possible, but only if the consumer plays a long time. This package indicates how the average odds can vary substantially over even prolonged periods of playing. This demonstrates that unless a gambler kept a diary and played for a long time, it is going to be very hard to calculate the odds of winning on real machines.

The program is also intended to be a useful tool for investigating how the different choices made by consumers affect the outcomes they will face. For example, if a person played once a week for one hour on a poker machine, playing nine lines and five credits, it is almost impossible that he or she will be ahead financially at the end of the year. That's fine, of course, if the gambler

¹ The interested reader may also like to look at the mathematical study by Brawley and Mateer (1999) on video poker machines in South Carolina, the examination of the mathematics of Australian-type machines produced by Toneguzzo (1996) and Legato's (1999) very accessible material on random number generators used in gaming machines.

gets good entertainment value from playing the machines. After all, no one says they get bad value from a movie because they 'lose' money when they see one.

But some consumers, and especially problem gamblers, have misconceptions about poker machines. They think that they can win. They think that bad stretches are followed by good ones, they chase losses, and they re-invest wins to try to try expand their winnings. They think that winning is possible when it is very unlikely. A program like this, which exactly mimics a gaming machine, demonstrates that stratagems like re-investment, playing longer, or playing regularly do not provide financial returns.

To see more material about gambling, including the report, submissions and transcripts of hearings of the inquiry into Australia's gambling industries, visit the web site of the Australian Productivity Commission at www.pc.gov.au.

2 Loading the software

The software is 32 bit and so will only run on Windows 95+ and NT systems. The monitor should be set for 600 times 800 resolution. To load the software double click on the setup.exe file in the 'pokegame' folder and follow the set up instructions. In most cases the files will take up about 360k of hard disk space. However, if you do not have the grid32.ocx or comdlg32.ocx files in your windows system, these will also be added, and these take up several megabytes.

If there are any difficulties in loading the software please contact rlattimore@pc.gov.au.

3 Getting started

When the program opens it will appear as in figure 1.

Select 'Simulation options' in the menu and choose one of:

- Duration of a gambling session (or CTRL D as the short cut)
- Rate of return from a gambling session (or CTRL R) or
- The cost of gambling (or CTRL Y).

4 Duration of a gambling session (Ctrl D)

This examines how long a gambling session will go on for a set amount of outlay by the gambler. It simulates the amount of time purchased by a gambler when playing the machines. The program reports the average duration of the game and its standard deviation (the latter allows you to calculate rough confidence intervals using Chebyshev's Inequality). It also allows scope for the game to be ended before all the money runs out if a sufficiently big prize is paid out. If the gambler always recycles or 're-invests' winnings regardless of the prize won, then with big prizes the game duration can be very protracted indeed. Consequently, by specifying a prize amount that will prompt a stop in play, this implausible situation is avoided. This option also shows the average player losses associated with the playing style adopted.

Figure 1 The opening page of the program



When 'Duration' is selected from the menu, you are asked to input the key data on playing style, the machine and the simulation (as shown in figure 2):

- Machine denomination

- Number of lines
- Credits per line
- Number of simulations
- Amount outlayed
- End amount
- Time limit

Figure 2 The input form for play duration

GameLength Input.

Machine Denomination. [2, 5, 10 Cents]

Credits/Line. [1, 2, 3, 5, 10]

No. of Lines. [1, 3, 5, 7, 9]

No. of Simulations.

Initial Amount(\$).

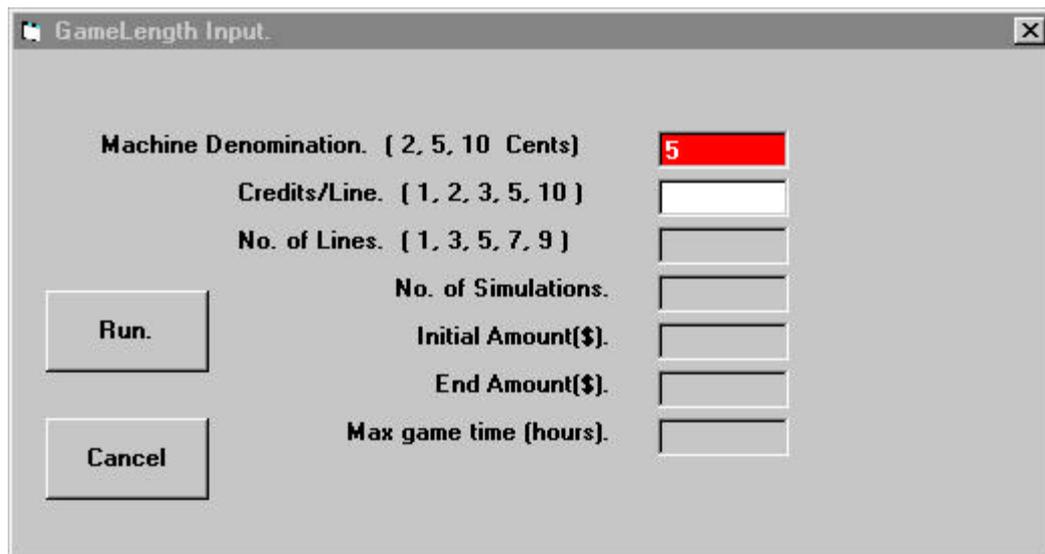
End Amount(\$).

Max game time (hours).

Run. **Cancel**

To select an item in the first three categories either position the mouse cursor over the desired item shown in the brackets, or type in the desired entry into the text box. For example, in figure 3, the cursor has been positioned on the value 5 in the 'Machine Denomination (2, 5 , 10 Cents)' choice and the mouse button clicked.

Figure 3 **Selecting an item for the text box using the mouse cursor**



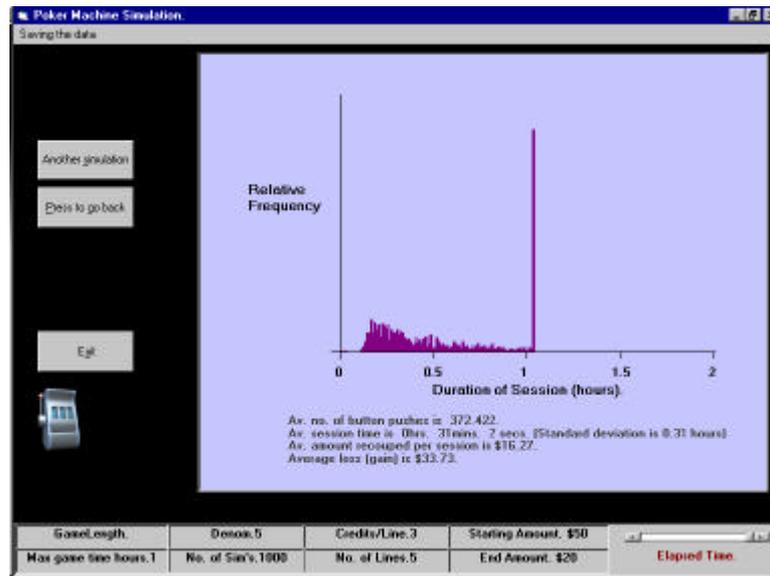
When all the simulation parameters are entered then mouse click on 'run' on the input form to run the simulation.

The results of a typical run is shown in figure 4. The bottom panel of the results output indicates the settings that were used to run the program. For example, in this case, 1000 simulations were run on a 5 cent machine, with 3 credits per line (ie 15 cents) and five lines per button push. In this case, the player is betting 75 cents per button push. They start playing with \$50. The simulation scenario has been set up so that they will stop playing if either they win more than \$20 with a button push (End Amount) or after one hour of play (Max game time). The simulation results show that the average number of button pushes is about 372 or an average game duration of 31 minutes and 2 seconds. The standard deviation is 0.31 hours, which suggests that the game length is quite variable. When the player stops gambling they leave with an average of \$16.27 so that the average loss is \$33.73.

The graph shows the likelihood of different outcomes. The height of any segment in the graph indicates the frequency of a particular game duration (as shown on the vertical axis). The horizontal axis represents the different duration possibilities. The spike at 1 hour indicates that in a large number of cases people could have played on for longer given the initial outlay made and the playing style they adopted.

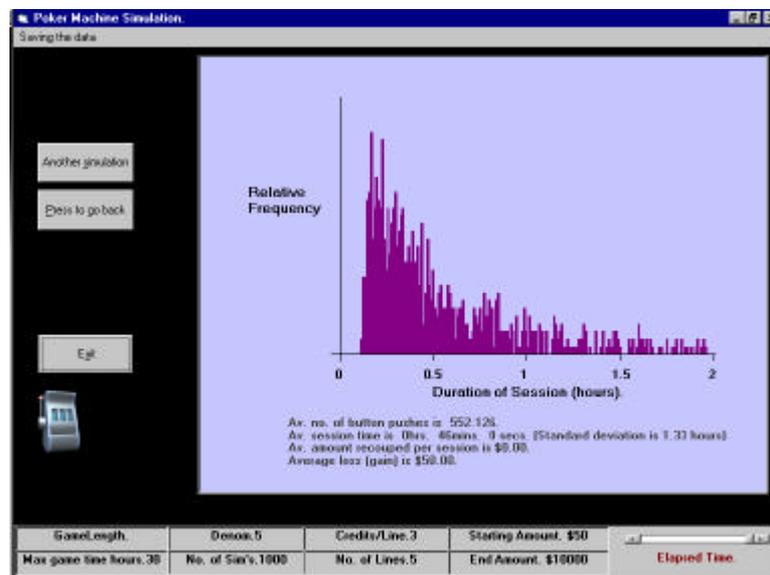
It is worth contrasting this outcome with that when the time and end amount constraints are relaxed (figure 5). In this case, the distribution of game duration is no longer truncated at one hour, but has a long right hand tail. The average playing time increases and players leave with no money.

Figure 4 The results of running the duration simulation with time limits



Data source: Results from EGMSim.

Figure 5 The results of running the duration simulation with effectively no time limits



Data source: Results from EGMSim.

5 The player rate of return and player losses

This calculates the return rate for a gambling session. The expected return rate for this game is known (at 87.84 per cent), but will not be realised except over

many hundreds of thousands of games. Simulation demonstrates how variable the rate of return is. The rate of return is equal to total wins divided by game turnover. One minus the rate of return is the player loss rate. So if a player gets a return of 80 per cent in a game, then they have faced a loss of 20 per cent on turnover. The program reports the average return and its standard deviation. The latter can be used to construct approximate confidence intervals (see Chebyshev's Inequality in section 10). The program also shows the average loss made by the gambler.

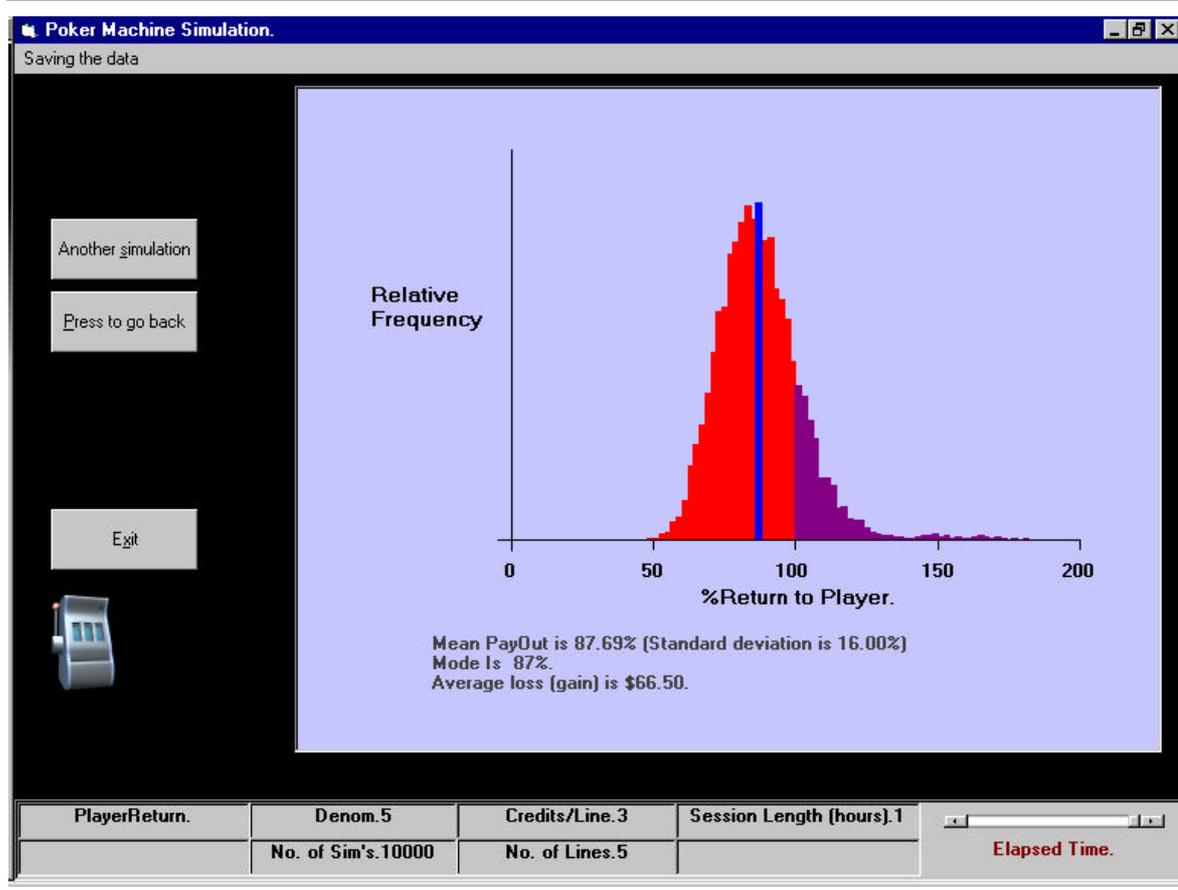
To run this simulation, the following data must be input:

- machine denomination;
- credits per line;
- number of lines;
- number of simulations; and
- session duration.

Then mouse click on 'run' on the input form to run the simulation.

What is being simulated is the player return rate associated with a style of play over a fixed period of time. Figure 6 shows a typical outcome associated with a run with the settings as described in the lower panel of the figure. The dark line in the middle of the distribution marks the mean.

Figure 6 The rate of return on a gaming machine and player losses



Data source: Results from EGMSim.

For a run of 10 000 simulations, the program gets a mean return rate of 87.69 per cent (very close to the 87.84 per cent that is the theoretically expected average over an infinite number of trials). The modal (most likely) outcome is an 87 per cent return. Playing on a five cent machine with 3 credits per line and five lines will cost on average around \$66.50 for one hour of play. The distribution shown in the figure shows that the rate of return (and therefore the amount of player losses) varies significantly. Some people get a return as low as 50 per cent. This would imply that their losses would be:

$$\begin{aligned}
 \text{Losses} &= \text{Denomination} \times \text{Credits per line} \times \text{Lines played} \times \frac{\text{Duration} \times 60 * 60}{5} \times (1 - \text{Return}) \\
 &= 0.05 \times 3 \times 5 \times \frac{1 \times 60 \times 60}{5} \times (1 - 0.5) = \$270
 \end{aligned}$$

where *Denomination* is the dollar value of the machine (in this case a 5 cent machine so 0.05), *Credits per line* are the number of credits per line bet, and *Lines played* are the number of lines simultaneously played with each button press. The *Duration* is one hour, which equates with 3600 seconds and allows

3600/5 or 720 button presses if each button press takes 5 seconds. The *Return* is in this case 50 per cent.

Others actually make a net profit on the machines. For example, some people win 150 per cent or more. This implies a net gain of \$270 or more.

But most people lose at least something (shown as the shaded area to the left of 100).

6 The cost of gambling

This calculates the overall financial consequences of gambling over an average session, month and year for different gambling styles. A month is regarded as 4 weeks, so that playing 3 times a month is equal to 39 times a year (ie multiplied by 13, not 12).

To run this simulation, the following needs to be input:

- machine denomination;
- credits per line;
- number of lines;
- number of simulations;
- session duration; and
- sessions per month.

Then mouse click on 'run' on the input form to run the simulation.

The output shows the average cost (gain) per session, per week and for the year overall. It also indicates the best outcome associated with each of these, and the standard deviations of the cost estimates for each of the three periods (which can be used to construct rough confidence intervals using Chebyshev's Inequality).

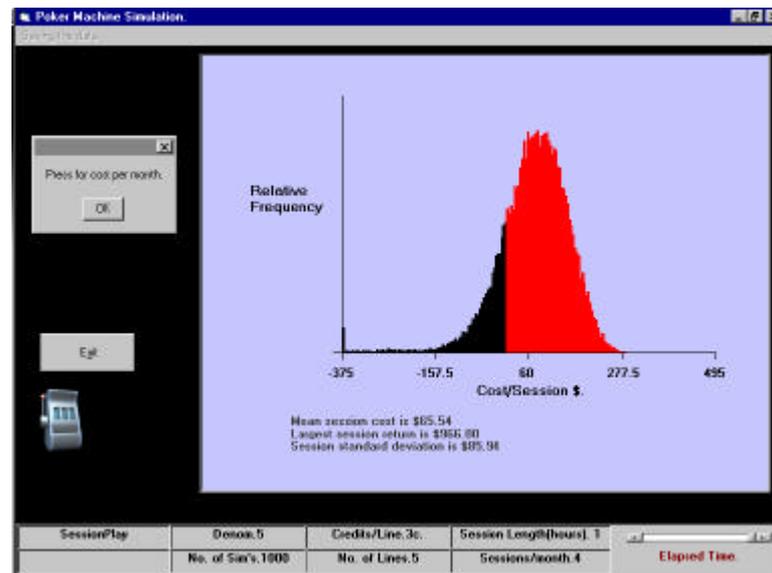
Note that problem gambling behaviour can be contrasted with recreational gambling because problem gamblers will tend to:

- play for longer (reflecting a greater preoccupation with gambling and the time taken in recycling wins);
- choose more lines and credits; and
- play more sessions per month.

A typical set of outputs associated with this simulation option is shown in figures 7 to 9 (note that to go onto the next graph simply click on OK in the dialogue box).

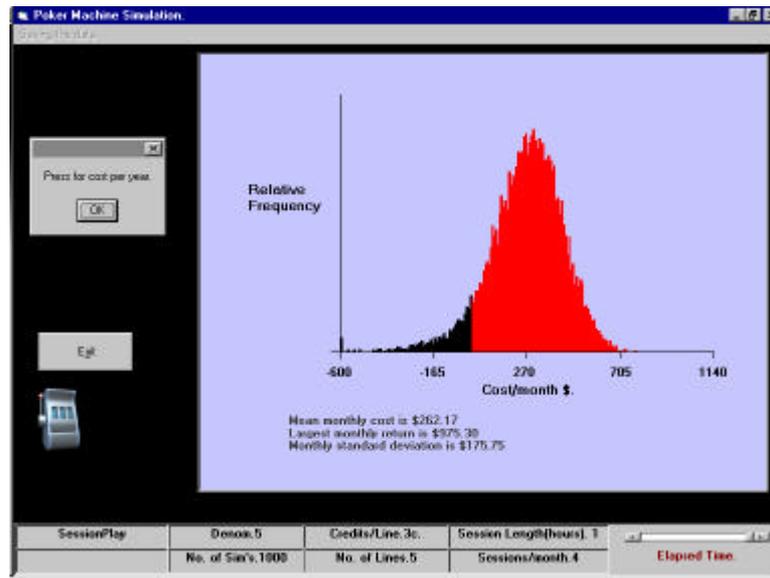
This simulation option can take some time. For example, just to run 50 simulations of once weekly playing takes around 15 seconds on a Pentium-II 266. 1000 simulations would take around 5 minutes. If by mistake an excessively long simulation is chosen then it is possible to abort the simulation using the abort command on the simulation sheet. An idea of how long it will take to complete any simulation is also shown with the 'elapsed time' indicator at the right hand bottom end of the screen. The sliding control will move to the right as simulations are completed — indicating roughly how long it will take to finish the run.

Figure 7 **The cost of gambling**
The per session cost



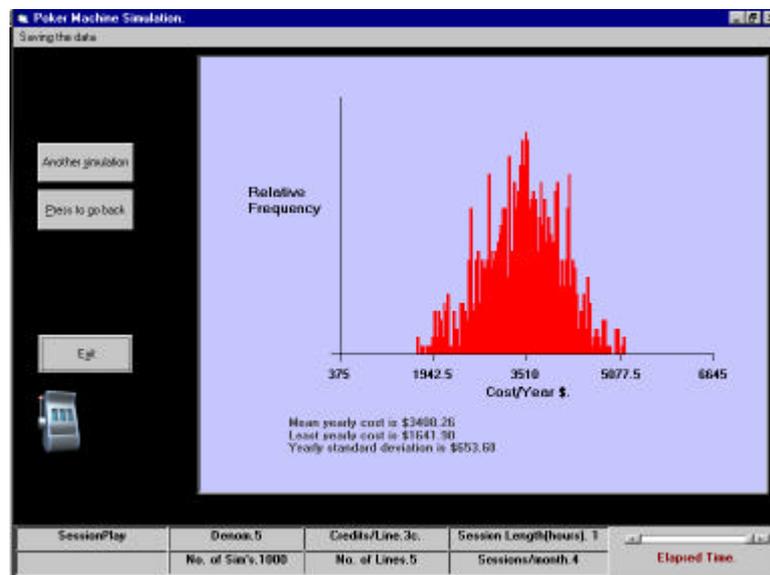
Data source: Results from *EGMSim*.

Figure 8 **The cost of gambling**
The monthly results



Data source: Results from EGMSim.

Figure 9 **The cost of gambling**
The yearly results



Data source: Results from EGMSim.

7 Simulation specifications

Machine denomination (2 cents, 5 cents or 10 cents)

There are three options for machine denomination. Denomination refers to the value of any given credit. The minimum starting amount bet is always \$1, but on a 2 cent machine this provides 50 credits, while on a ten cent machine it provides only ten.

Note that to select denomination you can click on the pre-set options in the brackets and your choice will appear in the text box. Or you can simply type in the appropriate option.

Number of lines (1, 3, 5, 7 or 9)

This indicates the number of lines you wish to play. Australian machines typically have three rows and five reels. But more than three lines can be obtained by having different pathways across the rows and reels. For example, while each of the straight rows form a reel, another line is formed by starting in reel1, row1, then reel2, row 2, then reel3, row3 then reel4, row2 and finally reel5, row1. With up to 9 lines and 10 credits per line, it is possible to bet from between 1 credit per button push to 90 credits per button push. On a 10 cent machine, it is therefore possible to outlay between 10 cents and \$9.00 per button push.

To select the number of lines, you can either type in the desired number in the appropriate box or select one of the value options within the brackets.

Credits per line (1, 2, 3, 5, or 10)

This indicates how many credits you wish to stake on any one line. By allowing variation, the machine allows people to play at higher levels of intensity if they wish to.

You can either type in the desired number in the appropriate box or select one of the value options within the brackets.

Number of simulations

You need to type in the number of simulations you want. This can be as small as one, but can also be in the thousands. Large simulations take time, so

experiment, before starting a very big simulation. A large simulation - of about 1000 - will typically provide a very good idea of the likely distribution of session duration. A 1000 simulations is like writing the outcomes of 1000 real world games in a diary and then summarising them - but the computer program can do this at rates thousands times faster than running experiments on physical machines.

Amount outlayed - Initial amount (\$)

This is the amount that the gambler outlays. It is equal to cash brought into the venue, plus any withdrawn from an ATM, and put into the machine. It excludes recycled winnings. For example, if a person starts gambling by inserting a \$50 dollar note into the machine and then supplements that by another \$50 dollars from his wallet sometime during play, then the amount of outlay is \$100.

End amount (\$) — prize amount prompting a stop in play

This is the prize size that will prompt a player to stop. For example, even when a gambler is buying entertainment on the machine, they will often elect to stop playing if they make a big win, such as \$200. If they recycle all winnings regardless of the prize size, then the distribution of game duration will have a very long 'tail' on the right side. This reflects those circumstances when a gambler has a big win and continues playing.

If you wish to examine the (theoretical) unconstrained distribution of duration then set the value of the End Amount at a high level (say \$10 000 - which is greater than the maximum possible prize).

To model a more realistic distribution of duration it is appropriate to set either a limit on the time played (time limit) or specify a prize amount that will prompt the gambler to stop playing.

A recreational gambler will typically stop for a smaller prize amount (or time) than problem gamblers. The program enables the differences in the outcomes from the these different styles of play to be modelled.

Time limit

The duration simulation can measure the maximum playing time that can be obtained associated with a given stake. But it can also be useful to look at the:

- distribution of duration if the player stops playing when they make a certain size of win (setting the End amount).

-
- distribution of duration if the player stops playing after a certain time limit.

As well, this simulation indicates the net loss (or gain) associated with different playing styles. If a player sets a limit on the time spent or stops playing if they make a certain size of win, then their average losses will be less.

One of the behavioural traits of problem gamblers is that they tend to re-invest their winnings and do not typically set time limits on play. So to model a problem gambler set:

- a large value for the End amount (a winning amount that stops play) (eg \$250); and
- a large value for the maximum duration (eg 3 hours).

This can be contrasted with that of a recreational gambler by setting:

- a smaller value for the End amount (eg \$50); and
- a small value for the maximum duration (eg 1 hour).

Session duration (hours)

This is the length of time that a person gambles. They play at a rate of one button push every five seconds - or 720 per hour. The program simulates the play duration, including the mean from the simulation.

However, it is also possible to calculate the loss of play associated with a given duration. The expected loss from playing is calculable from the duration and the style of play.

It is equal to:

Expected loss (\$) = 720 x Hours of play x Credits per line x Lines played x Machine denomination (\$) x (1 - Player return rate)

For example the expected loss from playing 1 hour, 10 credits per line, 9 lines on a 10 cent machine is:

$$720 \times 1 \times 10 \times 9 \times 0.10 \times (1 - 0.8784) = \$787.97$$

A given simulation may not return this result exactly because the simulations represent the mean of a series of random outcomes, whereas the above formula provides the mathematical expectation (that would arise as the number of simulations tends to infinity).

Sessions per month

This is the number of times a gambler goes back to a venue each week for a specified session of play.

The simulation allows up to 240 times. This is not intended to be a realistic form of behaviour by any gambler, but it allows the simulation of the likely lifetime costs of playing on gaming machines. That is, if you wished to see what the effect of playing twice a week would be over 30 years, then that is equivalent to asking about playing 240 times a month over a year.²

8 Saving the results from the program

If you wish to keep a record of the outcome, then select 'Saving the data' from the menu and click on file. Choose a file name (pokegame.dat is the default name) and the data (and the game style settings will be saved to a file. This file can be read later by a text editor (such as MS Word or Programmer's File Editor), or from MS Excel and equivalents.

An example of what the file looks like for 50 simulations of the rate of return is shown below (box 1). The first line of the file indicates the type of simulation run — in this case, a simulation of player returns, based on a five cent machine, five credits per line and five lines. The file indicates the summary outcome from the simulation (such as average loss and the mean payout). It also provides the data for the distribution of outcomes. The left hand column relates to the interval or bin width and the right hand column is the number of simulations which fell into this interval. To interpret the left hand column, it is necessary to use the formula for the interval. In the case of the rate of return, the formula is $2n \leq x < 2(n+1)$. This means that the first interval lies between a rate of return of 60 per cent to less than 62 per cent. The last item in the interval column, 80, relates to an interval of 160 percent or over.

If MS Excel is used to generate a graph of the whole or a portion of the data, it is sensible to use the concatenate function to set up the interval for the graph (figure 10).

² Or the effect of playing for 60 years just once a week — which is not inconceivable for some patrons.

Box 1 Typical file contents

```
PlayerReturn.  
Denom.5  
Credits/Line.5  
No. of Lines.5  
No. of Sim's.50  
Session Length (hours).1  
Mean PayOut is 88.62% (Standard deviation is 17.22%)  
Mode Is 91%.  
Average loss (gain) is $102.40.  
Results are given as % player return.  
Results are given in 101 intervals(n=0 to 100).  
2*n =< % < 2*(n+1).  
Any overflow is put in n=100.  
Only intervals with entries are shown.  
30            1  
32            2  
33            2  
34            1  
35            1  
36            2  
37            2  
38            2  
39            2  
40            3  
41            1  
42            3  
43            4  
44            2  
45            5  
46            2  
47            4  
49            1  
50            1  
51            2  
52            3  
56            1  
60            1  
62            1  
80            1
```

Source: File contents from pokegame.dat

Figure 10 **Entering data into MS Excel**

	A	B	C	D	E	F
1	Interval	Frequency	Interval			
2	30	1	60 to 62			
3	32	2				
4	33	2				
5	34	1				
6	35	1				
7	36	2				
8	37	2				

9 To get help

Go to the Help menu on the start up screen in the program or push F1 at any time.

10 Chebyshev's inequality

This can be used to construct confidence intervals around an average, even when the properties of the distribution are unknown. The probability that any random variable, X , falls within k standard deviations of the mean is at least $1 - 1/(k^2)$. That is:

$$\Pr(M - k \cdot S < X < M + k \cdot S) \geq 1 - 1/(k^2)$$

where S is the standard deviation and M is the mean.

For example, suppose that the mean rate of return of a particular simulation is 88% with a standard deviation of 18%. Consider $k=2$. Then $1 - 1/(k^2) = 1 - 1/4 = 0.75$. Then at least 75 per cent of the observations of the simulation will fall within 2 standard deviations of the mean - or in this case between 52% and 124%. Consider $k=4$. Then at least 93.75% of observations fall between 16% and 160%.

Note that Chebyshev's inequality holds for any distribution and consequently the results are usually weak. It indicates that conservatively a certain share of observations lie between two bounds. For example, if in fact the distribution for a random variable was normal, then a little more than 95% of observations fall between 2 standard deviations (ie 20 percentage points more than given by Chebyshev's inequality).

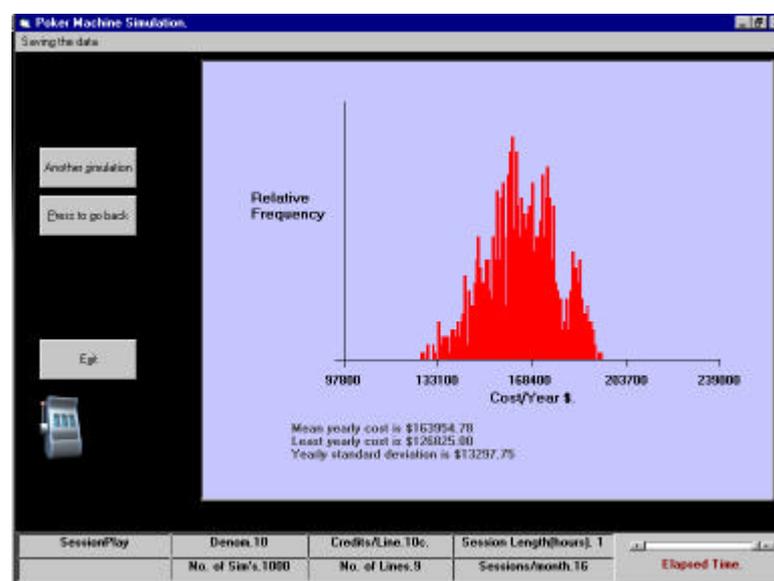
Results for a year, with a reasonably large number of games played tend towards normal, and using this as the underlying distribution in this case will give a better picture of the confidence intervals.

11 Some examples

Can a person lose much on a machine?

Yes. For example, consider the outcome of playing for two years, playing a 10 cent machine at maximum intensity, twice a week for just one hour per session. To model this, choose 'Cost of playing'. Enter 10 cents for the denomination, 10 credits per line and 9 lines. Run 100 simulations (this will take about 2 minutes on a Pentium II – 260). Choose 1 hour of session time. Select 16 times a month for the frequency. This is not formally correct, but it means that the 'yearly' result will equate with 2 years of playing twice a week (ie twice a week times 4 times a month = 8 times a month. To get two years worth of results, multiply by 2 to get 16). The program output suggests that this style of highly intensive gambling will cost around \$160 000 over the two year period. Among 1000 people who were to pursue this style of play, the minimum loss was over \$125 000 (figure 11).

Figure 11 The cost of playing intensively (twice a week) for two years

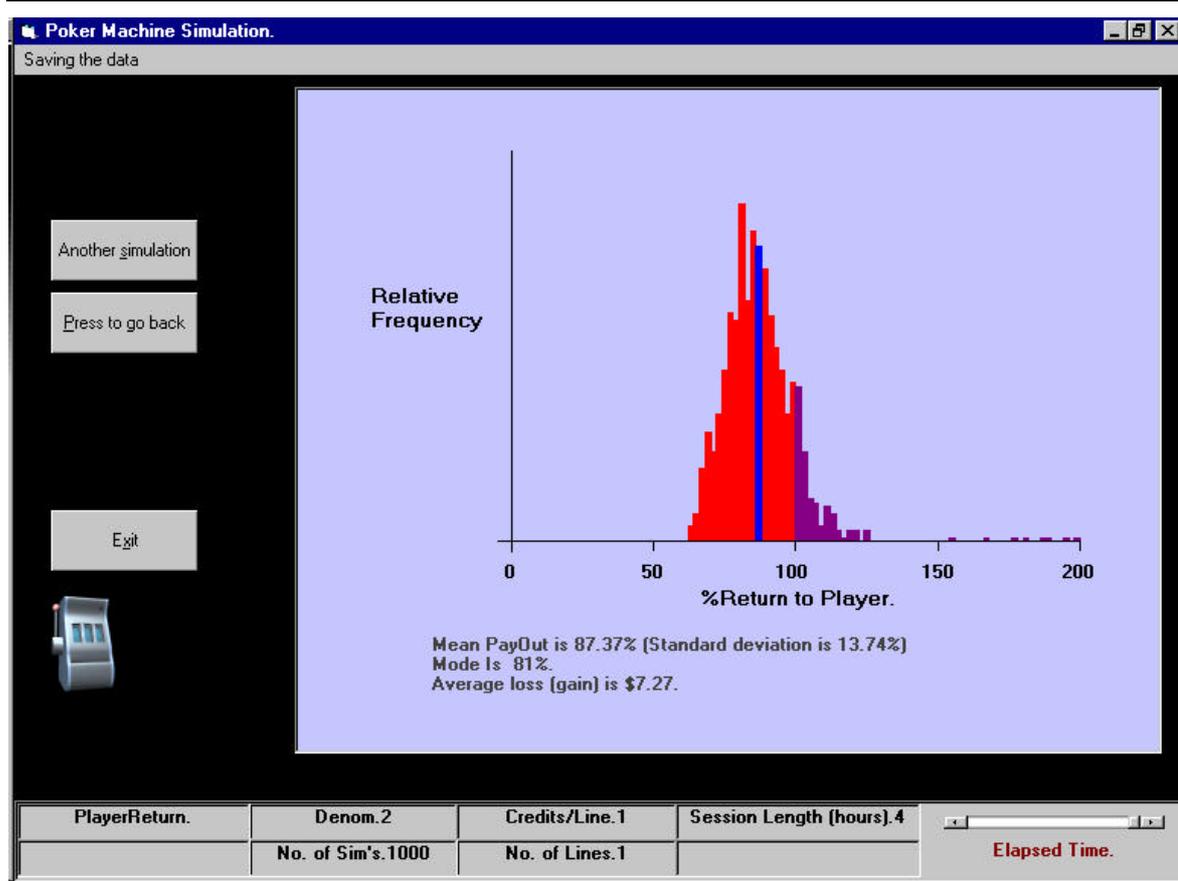


Data source: Results from EGMSim.

Is it always 'expensive' to play for a long time?

No. For example, run 'Return rate for a gambling session' (CTRL R) and apply settings of a 2 cent machine, 1 line, 1 credit, 4 hours, and 1000 simulations. The mean cost is \$7.27, but even the worst case scenarios are not expensive (figure 12). At low intensity, a gambler can play a gaming machine for hours and lose very little.

Figure 12 Playing low intensity for a long time



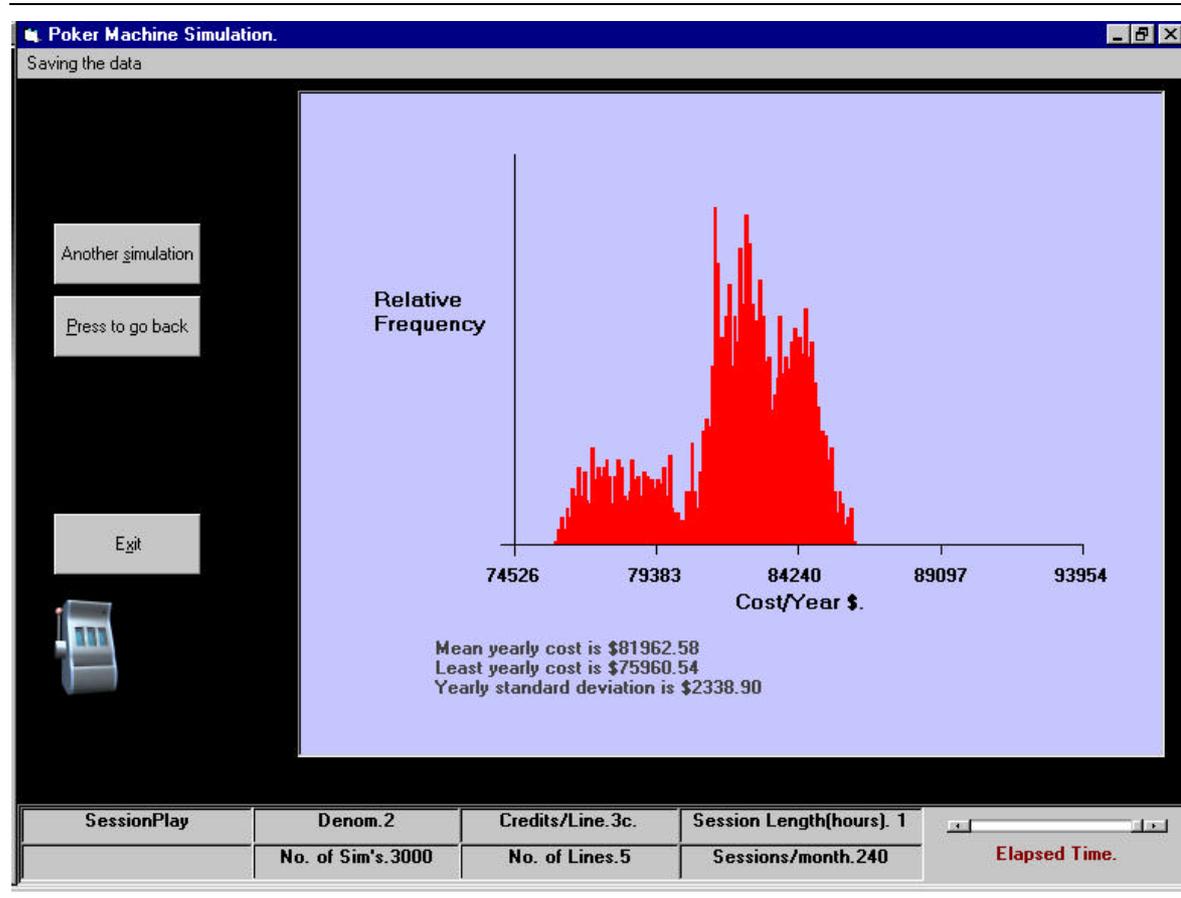
Data source: Results from EGMSim.

What is a lifetime of costs of playing?

Say that someone plays the machines on average once a week at a relatively low level of intensity (say 5 lines, 3 credits per line on a 2 cent machine for one hour per week on average — or 30 cents per button push). They played from aged 18 to age 78 or for 60 years. What is the expected cost and how much will that vary from the average? Can some people win over such an extended lifetime?

Again the simulation set up will take account of the fact that a year will have to represent 60 times the usual amount of gambling activity that takes place in that time. Thus go to the 'Yearly/loss gain from gambling' menu item. Enter the denomination, lines, credits details, have a session duration of 1 hour and 240 sessions per month (ie 60 times 4). The result from 3000 simulations suggests that the mean loss over this sixty year period is \$81 963. The least loss was \$75 960. With a standard deviation of only \$2339 it is clear that even in a population of millions no one could be expected to win overall after a lifetime of playing (figure 13). They may, however, have derived a lot of entertainment value from the machines.

Figure 13 The costs of a lifetime of low intensity playing



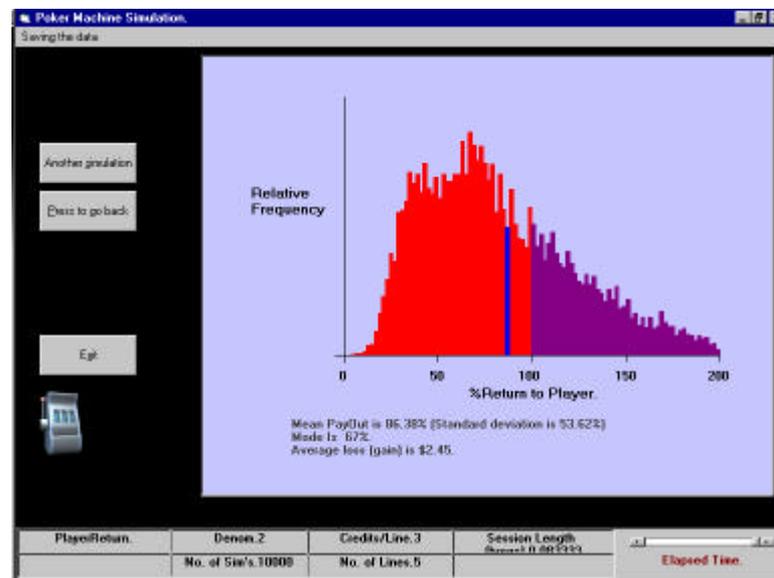
^a The distribution appears to be trimodal and certainly bimodal. However, the area under the distribution symmetrically about the mean is still close to the normal distribution. For example, the area under the distribution about the mean between $\pm 0.3\sigma$ is 29.5 per cent (cf 23.6% for a normal), between $\pm 0.5\sigma$ is 42.5 per cent (cf 38.3% for a normal), between $\pm \sigma$ is 65.4 per cent (cf 68.3% for a normal), between $\pm 1.5\sigma$ is 86.2 per cent (cf 86.6% for a normal), and between $\pm 1.96\sigma$ is 94.2 per cent (cf 95% for a normal), where σ is the standard deviation. Given the size of the standard deviation it is effectively impossible that a regular player could make a lifetime net monetary gain from gambling on such gaming machines.

Data source: Results from EGMSim.

How does the distribution of returns vary from playing a long time of a short time?

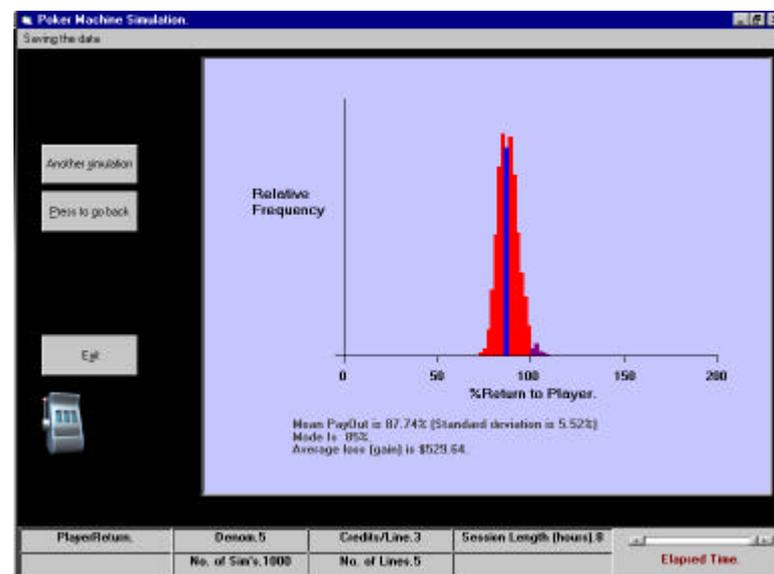
Figures 14 and 15 contrast the results of two simulations.

Figure 14 **Distribution of returns for a short playing time**



Data source: Results from EGMSim.

Figure 15 **Distribution of returns for a long playing time**



Data source: Results from EGMSim.

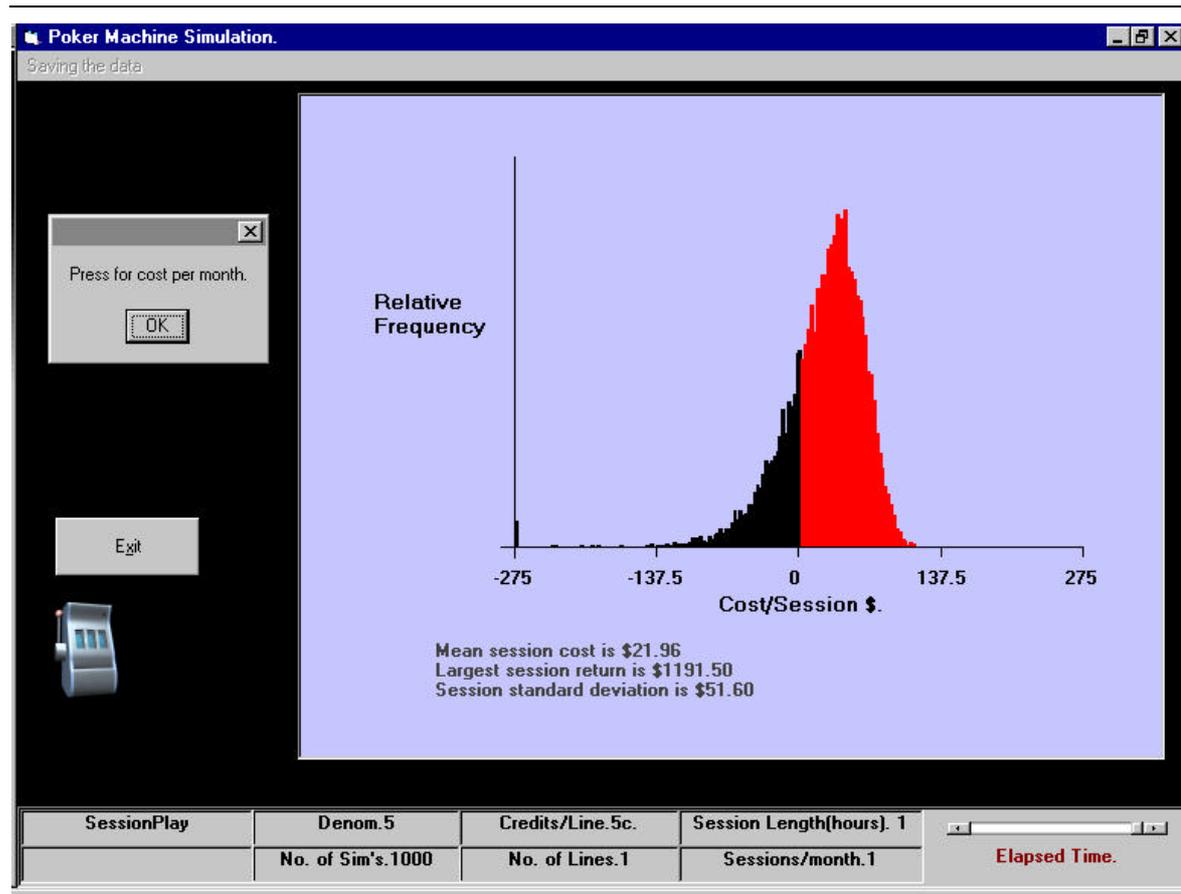
The first shows the outcome from 10 000 simulations of gamblers playing for just five minutes, and the second, the outcome from 1000 simulations of gamblers playing the same style of play, but for 8 continuous hours. It is

apparent that outcomes are much more variable for someone playing for five minutes than someone playing for a long period. A measure of this is the ratio of the standard deviation to the mean return (the coefficient of variation). This is 62 per cent in the first case and 6.3 per cent in the second case.

How does the distribution of returns vary from playing many lines versus just one?

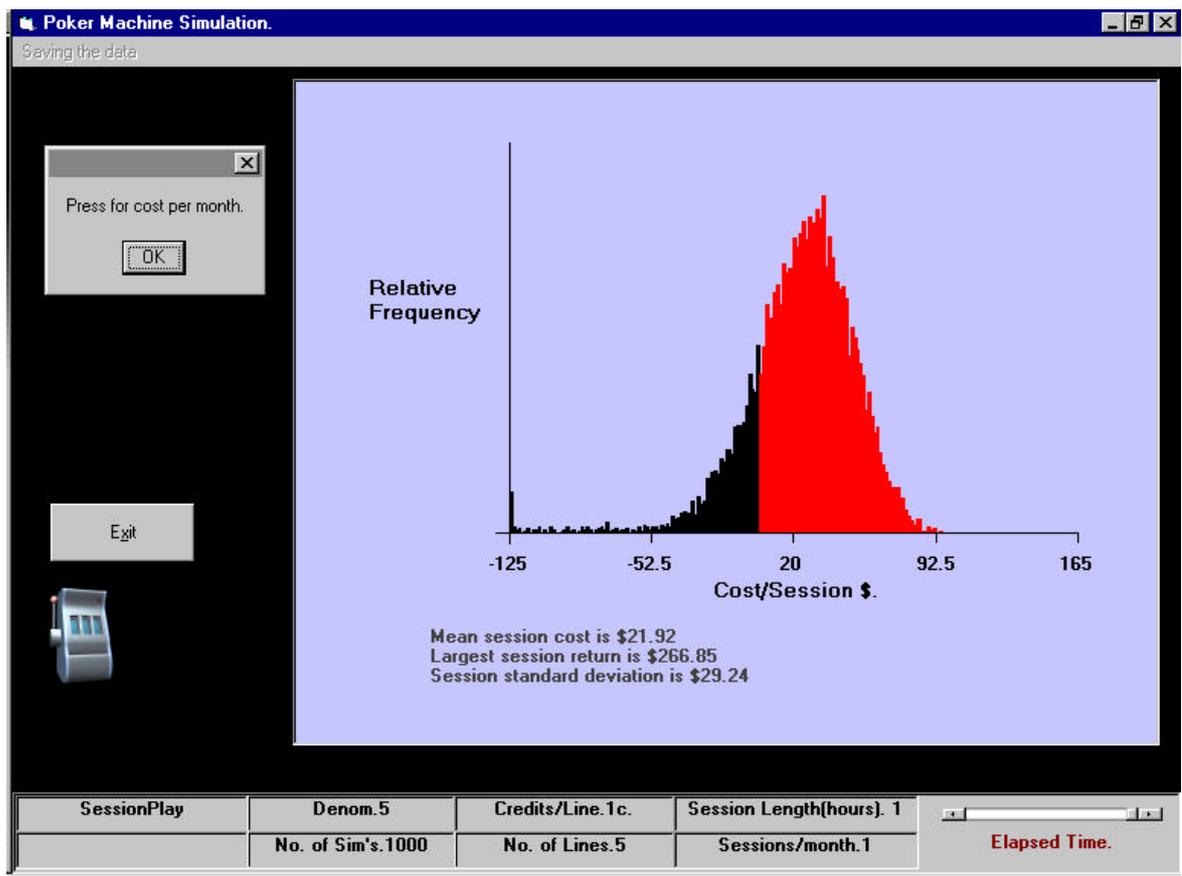
The distribution of returns is more volatile when someone plays a single line compared to multiple lines, in essence because playing more lines is like playing more games. As more games are played, the possibility of extreme outcomes (either big wins or long strings of losses) is significantly reduced. The relative variation (as measured by the ratio of the standard deviation to the mean cost) is about half when someone plays 5 multiple lines for a one hour session compared to just one line (figures 16 to 17).

Figure 16 One line only



Data source: Results from EGMSim.

Figure 17 5 lines simultaneously



Data source: Results from EGMSim.

References

Legato, F. 1999, 'RNGs for dummies, *Casino Center*, May, accessed from www.casinocenter.com/strictlyslots/archive/May99/html/behindthescenes.html.

Toneguzzo, S. 1996, 'Relationship between gaming machine prize limits and turnover. A mathematical model, 7th National Conference of the National Association for Gambling Studies, Stamford Grand Hotel, Glenelg, SA, 18-21 November.

Brawley, J. and Mateer, T. 1999, 'Video Poker in South Carolina: A Mathematical Study', Clemson University, South Carolina.