European Option Pricing Using a Combined Inversive Congruential Generator*

Yered Hammurabi Pita-Juarez †
Steven Melanson ‡

Abstract

One of the main problems in mathematical finance is to find the fair price of various contracts that convey a right, known as options, which depend on the price of other financial assets like stocks, known as the underlying assets. A "fair" price for some of these contracts may not be obtained analytically. In this manner, Monte Carlo simulations offer a convenient way to compute the fair price numerically, relying on the approximation of an expected value by the average of the simulated values. We biefly discuss some common random number generators, including a combined inverse congruential random number generator, in Monte Carlo simulations to compute the fair price of a European call option and to analyze the sensitivity of the price with respect to the changes in the key model parameters.

Faculty Advisor: Coskun Cetin
Department of Mathematics and Statistics,
California State University, Sacramento
e-mail:cetin@csus.edu

1 Introduction

1.1 Options

An option is a financial contract that gives the holder of the contract the right to sell or buy an asset, but the holder does not have to exercise this right. Moreover, the purchase of an option requires an upfront payment. There are two basic types of options: call and put

*This work was supported by the SUMMA program of MAA during the summer-fall of 2010 when both the authors were undergraduate students at CSU, Sacramento.

†Harvard University, Department of Biostatistics, e-mail: yered.h@gmail.com

‡Columbia University, Industrial Engineering and Operations Research Department, e-mail: spm2139@columbia.edu

options. A call option gives the holder the right to *buy* an asset by a certain date (expiration date) for a certain price (strike price). On the other hand, a put option gives the holder the right to *sell* an asset by a certain date for a certain price. European options are options that can be only exercised on the expiration date [4]. Even though there are many analytical formulas and online tools to compute the fair price (premium) for such contracts under various simplifying conditions (e.g. constant model parameters, complete markets, etc), it becomes a difficult task to compute the price of an option when more realistic assumptions are used. There are no explicit formulas available in such cases and the computations should be done numerically. In this paper, by using simulation-based probabilistic methods, we describe and analyze how such a price can be computed numerically.

In sections two and three, we briefly describe and compare some well-known random number generators and their statistical properties. The section four is about the stochastic modeling of the financial market structure for our problem. We show how the numerical results compare with the values computed from the well-known Black-Scholes formula. We then conclude in section 5.

2 Random Number Generators

Since many statistical and stochastic systems in science, engineering, cryptography and economics rely on Monte Carlo simulations, which use (pseudo)random numbers, researchers in such areas need reliable "random number generators" (RNGs for short) for their work. There are various methods to generate such pseudo-random numbers. We discuss some well known RNGs below before starting the Monte Carlo simulations for financial applications.

2.1 Linear Congruential Generator

The linear congruential generator (LCG) is one of the oldest and best known pseudo-random number generators. The LCG was first proposed by D.H. Lehmer in 1948. The LCG uses a recursive formula: $x_i \equiv (ax_{i-1} + c)\text{mod}m$ to generate a series of pseudo-random numbers x_i, where $0 \leq x_i < m$. The first number in the series, x_0, is the seed, a is the multiplier, c is the increment and m is the modulus [2]. LCG algorithms have faster and simpler computations compared to other reasonable alternatives, but they are tractable.

In 1988, S.K. Park and K.W. Miller proposed a "minimal standard" LCG. The main aim of the Park and Miller was to create a LCG algorithm that could be ported to all systems available at that time. The minimal standard LCG is also known as the Park–Miller random number generator, which uses the parameters $a = 16087$, $c = 0$, and $m = 2^{31} - 1$ in the LCG formula. The modulus, $2^{31} - 1 = 2,147,483,647$, is the period of the generator, and is also a Mersenne number that is prime [7]. This period is considered too small for serious simulation-based work in science, engineering and financial economics applications. Besides being tractable and having relatively small period, this generator and in general LCGs are also criticized due to the lattice patterns and serial correlations issues that are not good indicators of ideal RNGs (e.g. see [2, p. 15-16], and the references there). Many of the alternative (non-linear) generators are much slower compared to LCGs but thanks to the advances in technology in regards to the computational power and efficiency, such alternatives are becoming more and more popular for more complex applications. Due to the desirable "randomness" properties, we discuss and use only inversive congruential generators (ICGs in short) together with their simple modifications (explicit and combined ICGs) in this paper. The reader may consult to [5, p. 15-16] and [2] for more information about other linear and nonlinear alternatives.

2.2 Inversive Congruential Generator

As a first nonlinear alternative to the LCGs, we considered the inversive congruential generators (ICGs, in short). After the ICGs were first proposed by Eichenauer and Lehn in 1986, various research papers were published on their good uniformity properties. An ICG doesn't show the regular lattice patterns or apparent serial correlations like LCGs but some spacing issues were still reported in some randomness tests (see [2, p. 36-37], [6] and the references there for more information). Another alternative that shares some of the similar nice theoretical and empirical properties is explicit (EICG) inversive congruential generators. For both of these generators, some extensive tables of parameters [3], and empirical comparisons are available [6].

The ICG uses the modular multiplicative inverse (when it exists) to generate next number in the sequence of pseudo-random numbers. The formula for the ICG is $x_i \equiv (ax_{i-1}^{-} + c) \bmod m$, where $0 < x_i < m$, x^{-} denotes the modular multiplicative inverse, a is the multiplier, c is the increment and m is the modulus. If x^{-} does not exists, then $x_i = c$. The ICG has a slow computation time and the choices for the modulus are restricted. Therefore, ICG is not a very common choice for a random number generator [2]. The choices for the modulus are restricted in order to achieve the full period m [1]. Hellekalek (1995) proposes the following parameters: $(a = 55, m = 1031, c = 1)$, $(a = 103, m = 1033, c = 1)$, $(a = 481, m = 1039, c = 1)$, and $(a = 66, m = 2027, c = 1)$ [3].[2]

2.3 Combined Inversive Congruential Generators

A way to improve the period and the randomness of certain RNGs (and to make them look less tractable) is to combine more than one generator, e.g. by applying a module reduction to a linear combination of a pool of RNGs. Intuitively, such "combined" generators are expected to be more "random" with nice uniformity properties since they may produce numbers from a richer finite set than each single generator with a more complex operation. However, the combination of multiple generators doesn't guarantee that the resulting generator is much better than each generator in the combination, especially if the generators are correlated with each other, possibly magnifying their "bad" properties. Below, we consider a pool of generators from ICGs. This approach has some certain advantages: First of all, since the moduli of the ICG generators are relatively small, the computational time resulting from the complexity of the modular operations can be improved, especially in parallel computational systems where each ICG generator can run in a different computer, independently of the others. Secondly, combined ICGs still preserve the desired statistical properties of ICGs

[1] The maximal period is obtained if and only if $x^2 - cx - a$ is a primitive polynomial in the finite field of $\{0, 1, ...p - 1\}$. See e.g. [6].

[2] See also the pLab project group website for similar combinations: http://random.mat.sbg.ac.at/

that are discussed in section [?]. So such generators have a good potential to be more popular in more complex (multiscale) simulation applications, for example when a quick and precise price quote for a certain complex financial contract or insurance premium is needed.

2.3.1 Three-Source ICG

We first start with a three-source ICG where each (source) ICG generator has a prime (full) period. The formula that we use for the three-source ICG in our simulations is $u_i = (\frac{x_i}{1031} + \frac{y_i}{1033} + \frac{z_i}{2027}) mod 1$, where x_i is generated by $ICG(a = 55, m = 1031, c = 1)$, y_i is generated by $ICG(a = 103, m = 1033, c = 1)$, and z_i is generated by $ICG(a = 66, m = 2027, c = 1)$. Since we want to simulate a uniform distribution from (0,1) (in short $U(0, 1)$), we divide each ICG by its own modulus. Assuming that all the variables x, y and z have a $U(0, 1)$ distribution and are independent, the resulting variable u has also a (theoretically) $U(0, 1)$ distribution. The period for the three-source ICG is the product of the three prime moduli [3]: $1031 \times 1033 \times 2027 = 2, 158, 801, 621$, which is slightly larger than that of the minimal standard LCG.

2.3.2 Four-Source ICG

Later on, we decided to improve the combined ICG by adding a fourth ICG. In this manner, we came up with a four-source ICG. The formula for the four-source ICG is $u_i = (\frac{x_i}{1031} + \frac{y_i}{1033} + \frac{z_i}{1039} + \frac{v_i}{2027}) mod 1$, where x_i is generated by $ICG(a = 55, m = 1031, c = 1)$, y_i is generated by $ICG(a = 103, m = 1033, c = 1)$, z_i is generated by $ICG(a = 481, m = 1039, c = 1)$ and v_i is generated by $ICG(a = 66, m = 2027, c = 1)$. The period for the four-source ICG is approximately 2.242995×10^{12}, which is greater than the three-source ICG period.

3 Empirical Tests

In our statistical tests or the heuristic discussions for the random number generators, the null hypothesis states that the output from a certain generator consists of independently and identically distributed (i.i.d.) observations from $U(0, 1)$. The alternative says that the sequence of such numbers doesn't have this distribution. After comparing the basic statistical summary (mean and standard deviation) of the sequence of the numbers generated, we mainly consider two types of tests: the χ^2 goodness-of-fit test and the Kolmogorov-Smirnov test. There are many other tests and measures to assess the quality of the random number generators, including spectral tests, lattice tests, Anderson-Darling tests, serial tests and runs tests, among others (see e.g. [2] and [5] for these and other well-known tests including the theoretical ones). Each test may be sensitive to some certain aspects of the alternative hypotheses. Some publically available test batteries or suites consider a rich set of tests that cover some common forms of the alternative hypotheses. Among them, the DIEHARD tests of G. Marsaglia, the NIST Test Suite and TestU01 of L'Ecuyer are well known. The TestU01 package includes most of the tests from the other suites and is usually preferred to others [2].

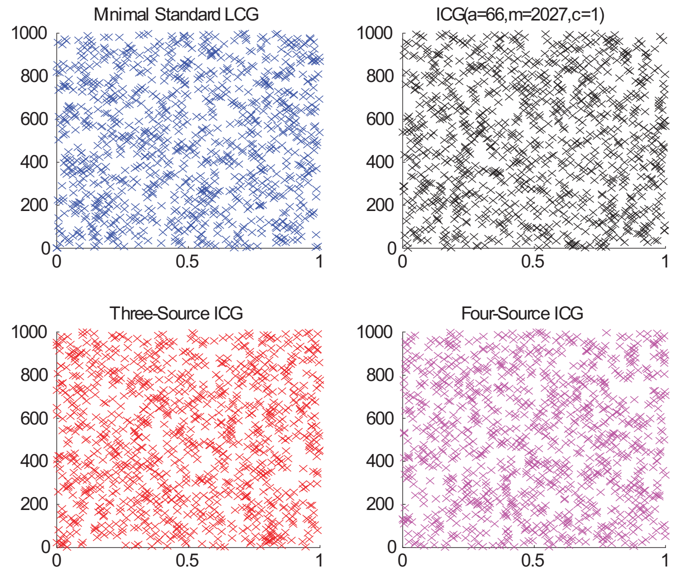


Figure 1: LCG and ICG scatterplots

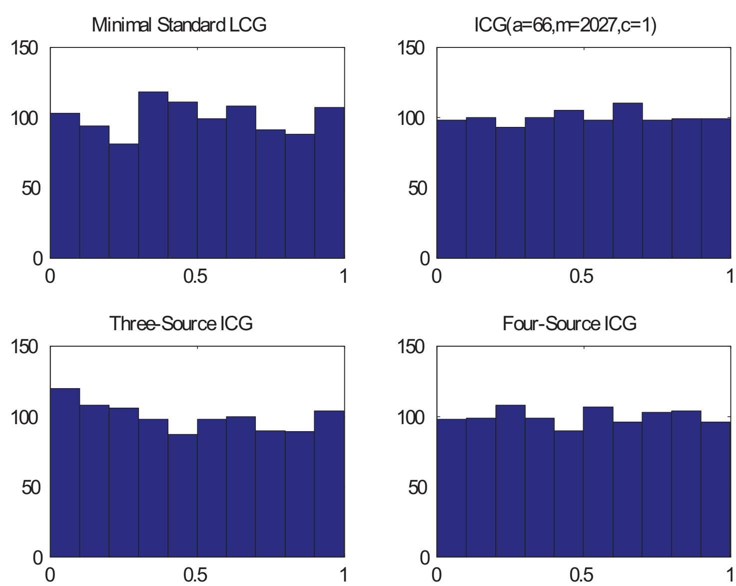


Figure 2: LCG and ICG histograms.

3.1 Computing Time

One of the biggest drawbacks of the ICG compared to the LCG alternatives is the slow computing time, mostly due to the operations to find the modular multiplicative inverse. So the implementation of the three-source and the four-source ICGs are also slow.

Table 1: Comparison of computing time with improved version

Name	n=100	n=1000	n=10,000
LCG(a=16807,m=$2^{31}-1$)	2.0589×10^{-4}	3.8077×10^{-4}	0.0021
Three-Source ICG	0.1702	0.4112	3.9843
Four-Source ICG	0.0861	0.5481	5.4486

*seconds

3.2 Mean and Standard Deviation

Since our implementations of the minimal standard LCG and the combined ICGs are supposed to resemble a $U(0,1)$ distribution, a way to assess the performance of a pseudo-random number generator is to compare the mean and the standard deviation of a sample generated by the generators with the mean of $U(0,1)$ distribution. Recall that for a $U(0,1)$ distribution, $\mu = 0.5$ and $\sigma = \sqrt{\frac{1}{12}} = 0.288675135$. Overall, the performance of the ICGs and combined ICGs were much better than the minimal standard LCG. In our simulations, $ICG(a = 66, m = 2027, c = 1)$ generator had the best performance in terms of being closer to the theoretical mean and the standar deviation (see the tables below).

Table 2: Comparison of LCG and ICGs(n=100)

Name	Mean	Standard Deviation
LCG(a=16807,m=$2^{31}-1$)	0.4778	0.3129
ICG(a=55,m=1031,c=1)	0.4930	0.2981
ICG(a=103,m=1033,c=1)	0.5119	0.2981
ICG(a=481,m=1039,c=1)	0.5058	0.2834
ICG(a=66,m=2027,c=1)	0.5039	0.3057
Three-Source ICG	0.5093	0.2907
Four-Source ICG	0.5057	0.2883

3.3 χ^2 Goodness of Fit Test and Kolmogorov-Smirnov Test

We performed two statistical tests, the χ^2 goodness-of-fit test and the Kolmogorov-Smirnov test, to assess the quality of our combined ICGs and compare it to the minimal standard LCG (with about the same period). For these tests, we obtained 10,000 random numbers from each RNG and divided them in 10 intervals: $(0, 0.1]$, $(0.1, 0.2]$, $(0.2, 0.3]$, $(0.3, 0.4]$, $(0.4, 0.5]$, $(0.5, 0.6]$, $(0.6, 0.7]$, $(0.7, 0.8]$, $(0.8, 0.9]$, $(0.9, 1]$. In these manner, we obtained a sample of size ten:

Table 3: Comparison of LCG and ICGs(n=1000)

Name	Mean	Standard Deviation
LCG(a=16807,m=$2^{31}-1$)	0.4971	0.2899
ICG(a=55,m=1031,c=1)	0.4992	0.2894
ICG(a=103,m=1033,c=1)	0.5004	0.2886
ICG(a=481,m=1039,c=1)	0.4992	0.2880
ICG(a=66,m=2027,c=1)	0.5002	0.2826
Three-Source ICG	0.4973	0.2889
Four-Source ICG	0.5050	0.2898

Table 4: Comparison of LCG and ICGs(n=10000)

Name	Mean	Standard Deviation
LCG(a=16807,m=$2^{31}-1$)	0.4950	0.2866
ICG(a=55,m=1031,c=1)	0.5001	0.2885
ICG(a=103,m=1033,c=1)	0.4990	0.2888
ICG(a=481,m=1039,c=1)	0.4994	0.2888
ICG(a=66,m=2027,c=1)	0.4999	0.2885
Three-Source ICG	0.5021	0.2879
Four-Source ICG	0.4993	0.2903

$X_1, X_2, X_3, X_4, X_5, X_6, X_7, X_8, X_9, X_{10}$. Each random variable, X_i, represents the frequency of the random numbers in each interval. If we assume that this is a $U(0,1)$ distribution then $X_i = 1000$ for $i = 1, 2, 3, 4, 5, 6, 7, 8, 9, 10$. For both tests we choose a significance level $\alpha = 0.05$.

In both cases, the hypotheses are H_0: The sample belongs to the uniform distribution $U(0,1)$, and H_1: The sample does not belong to the uniform distribution $U(0,1)$. The test statistic in χ^2 goodness-of-fit test is given by $\chi_p^2 = \sum_{i=1}^{10} \frac{(o_i - e_i)^2}{e_i}$ where o_i and e_i are, respectively, the observed and expected numbers of X_i. The decision rule is if $\chi_p^2 > \chi_{9,0.05}^2$ we reject H_0 and accept H_1. Otherwise we retain H_0. For Kolmogorov-Smirnov Test, the test statistic is given by $D_n = \sup_x |F_n(x) - F(x)|$, $F_n(x)$ and $F(x)$ are, respectively, the empirical and the hypothesized cumulative distribution functions of X. The decision rule is if $D_n > D_{0.05,10}$ we reject the null hypothesis H_0 and accept the alternate hypothesis H_1. Otherwise we retain H_0.[3]

All of the p-values are large and the statistical results are the same for the minimal standard LCG, the Three-Source ICG and the Four-Source ICG: there was a very weak evidence to reject the null hypothesis that these samples belonged to a $U(0,1)$ distribution.

[3] Since all of these generators are sufficiently good to pass the standard tests (with very large p-values), it is advised to use second order tests, e.g. another one based on the observed p-values of the generator.[2, p. 71-72]

Table 5: Minimal Standard LCG

Interval	1	2	3	4	5	6	7	8	9	10
o_i	970	981	983	1008	1055	1029	989	983	1009	993
e_i	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000
$F_n(x)$	0.0970	0.1951	0.2934	0.3942	0.4997	0.6026	0.7015	0.7998	0.9007	1.0000
$F(x)$	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0

- $\chi_p^2 = 6.0200 < \chi^2_{9,0.05} = 16.92$, keep H_0
- p-value = 0.737915
- $D_{10} = 0.0066 < D_{0.05,10} = 0.410$, keep H_0
- Thus, the test fails to reject the hypothesis that the sequence comes from the distribution $U(0, 1)$.

Table 6: Three-Source ICG

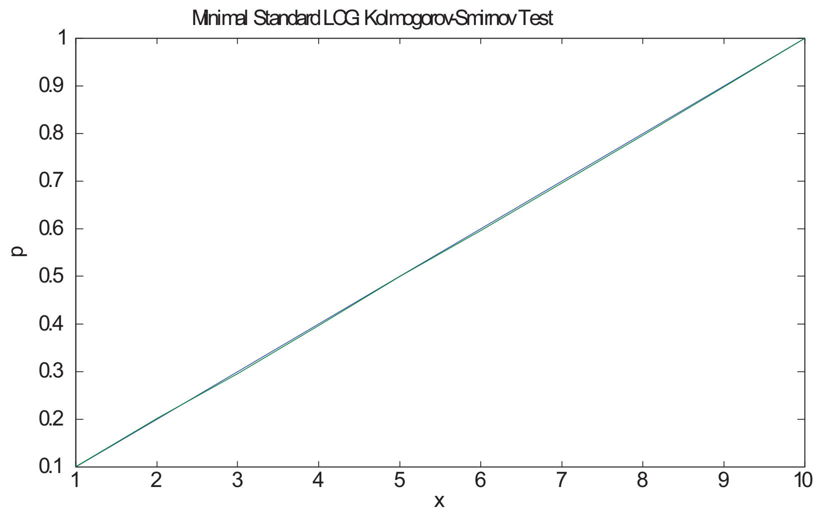
Interval	1	2	3	4	5	6	7	8	9	10
o_i	1014	968	1010	963	983	944	1020	1011	1021	1066
e_i	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000
$F_n(x)$	0.1014	0.1982	0.2992	0.3955	0.4938	0.5882	0.6902	0.7913	0.8934	1.0000
$F(x)$	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0

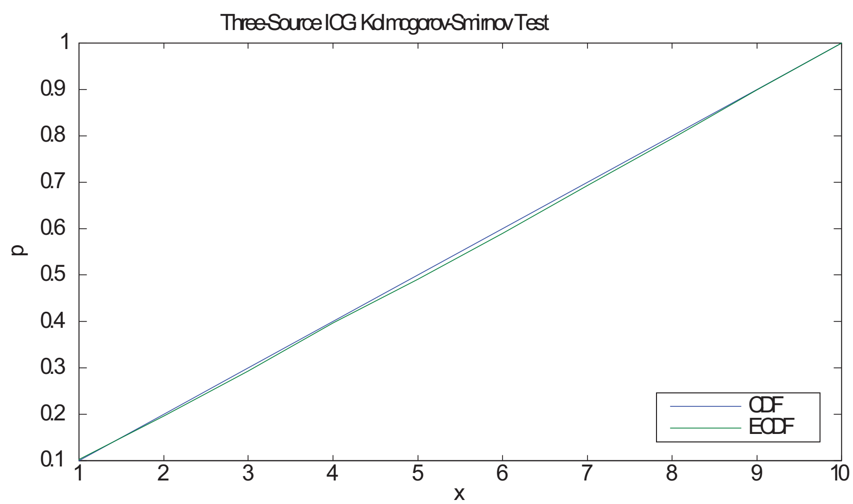
- $\chi_p^2 = 11.4320 < \chi^2_{9,0.05} = 16.92$, keep H_0
- p-value = 0.7527448
- $D_{10} = 0.0118 < D_{0.05,10} = 0.410$, keep H_0
- Thus, the test fails to reject the hypothesis that the sequence comes from the distribution $U(0, 1)$.

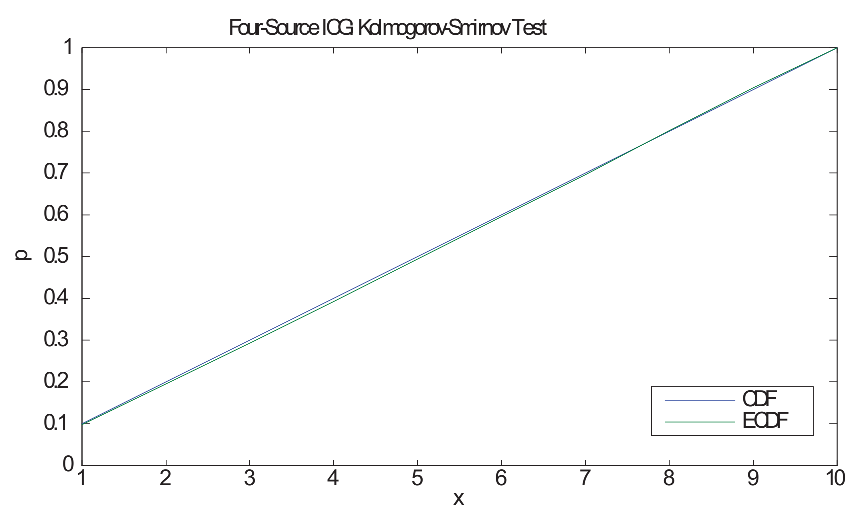
Table 7: Four-Source ICG

Interval	1	2	3	4	5	6	7	8	9	10
o_i	997	986	971	1025	1027	1091	961	977	987	978
e_i	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000
$F_n(x)$	0.0997	0.1983	0.2954	0.3979	0.5006	0.6097	0.7058	0.8035	0.9022	1.0000
$F(x)$	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0

- $\chi_p^2 = 13.3840 < \chi^2_{9,0.05} = 16.92$, keep H_0
- p-value = 0.8540135
- $D_{10} = 0.0097 < D_{0.05,10} = 0.410$, keep H_0
- Thus, the test fails to reject the hypothesis that the sequence comes from the distribution $U(0, 1)$.


Minimal Standard LCG Kolmogorov-Smirnov Test
p
x


Three-Source ICG Kolmogorov-Smirnov Test
p
x
CDF
ECDF


Four-Source ICG Kolmogorov-Smirnov Test
p
x
CDF
ECDF

4 Stock Pricing and Interest Rate Model

4.1 Stochastic Equations

The stock price is modeled by the stochastic differential equation (SDE, in short)

$$dS(t) = S(t)r(t)dt + S(t)\sigma dW(t), \tag{1}$$

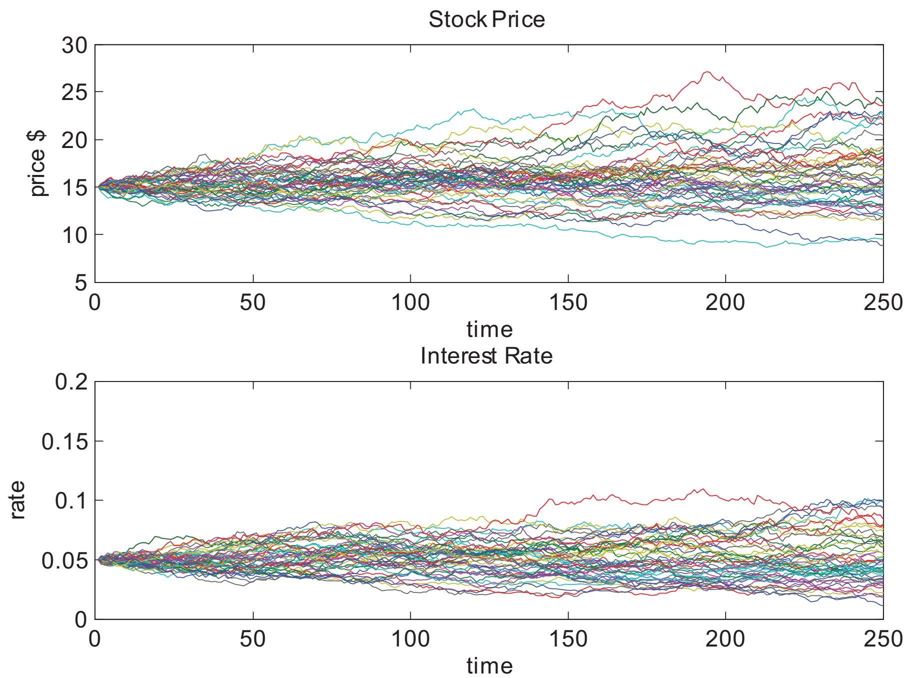
where $r(t)$ represents the interest rate, σ the stock price volatility, and $W(t)$ represents a Brownian motion. Moreover, the interest rate is also modeled by a stochastic differential equation $dr(t) = a[b - r(t)]dt + \hat{\sigma}\sqrt{r(t)}d\hat{W}(t)$,where a represents how fast the interest rate reverts to the long term interest rate, b represents the long term interest rate process, $\hat{\sigma}$ is the interest rate volatility, and $\hat{W}(t)$ represents another Brownian motion process that is correlated with $W(t)$. The details of the correlation structure is given below.

4.2 Discrete Versions

In order to use Monte Carlo simulations, we use the discretized versions of the SDEs for the interest rate and the stock price processes. The discrete version of the stock price differential equation is $\Delta S(t) = S(t)r(t)\Delta t + S(t)\sigma\Delta W(t)$, where $\Delta W(t) = \sqrt{\Delta t}Z_1$ represents an increment of the Brownian motion process W, and Z_1 represents a standard normal random variable. Similarly, the discrete version of the interest rate differential equation is $\Delta r(t) = a(b - r(t))\Delta t + \hat{\sigma}\sqrt{r(t)}\Delta\hat{W}(t)$, where $\Delta\hat{W}(t) = \sqrt{\Delta t}\hat{Z}$ represents an increment of the correlated Brownian motion process $\hat{W}$, and $\hat{Z} = \rho Z_1 + \sqrt{1-\rho^2}Z_2$ is a standard normal random variable correlated with i.i.d. standard normally distributed variables Z_1 and Z_2, and $\rho = \rho(Z_1, \hat{Z})$ is the correlation coefficient. The i.i.d. standard normal variables Z_1 and Z_2 are generated using the Box-Muller method: $Z_1 = \sqrt{-2\ln u_1}\cos(2\pi u_2)$ and $Z_2 = \sqrt{-2\ln u_1}\sin(2\pi u_2)$, where u_1 and u_2 are both generated by the Four-Source ICG. Since the sequences generated by the ICG methods and their combined generators don't have serial correlation or lattice structure issues [6], we can safely assume that the normally distributed pairs obtained through combined ICGs and Box-Muller method are also independent. Even though theoretical results justify the assumption of independence, it is always a good idea to test for independence of the generated sequence of numbers. Our empirical observations with the ICG-based computations didn't reveal any correlation issues.[4]

In our model, the expiration date, T, was set to a year for simplicity, and the following model parameters are chosen for the initial computations: $a = 0.25$, $b = 0.06$, $\hat{\sigma} = 0.1$, $r_0 = 0.04$, and $\sigma = 0.2$. Later, we change these parameters to assess the sensitivity of the option price with respect to key parameters.

[4] One way to get an even safer set of independent normal sequences would be perhaps using two different combined ICGs; one for u_1 (e.g. three-source ICG) and one for u_2 (e.g. four-source ICG) to generate a pair (z_1, z_2) of pseudo-random numbers from a normal distribution..



4.3 Call Option Prices

The terminal payoff of an European Call Option is given by

$$C(T) = \begin{cases} S(T) - K, & \text{if } S(T) > K \\ 0, & \text{if } S(T) \leq K. \end{cases}$$

where K is the strike price of the contract (the holder exercises it only when the terminal stock price is larger than K). See [4] for details. Using the terminal payoff and applying a discounting term, we can calculate the option price as $C_0 = E\left[e^{-\int_0^T r(t)dt}(S_T - K)^+\right]$. In our model, we also use a discretized version of the Riemann integral to approximate the integral $\int_0^T r(t)dt$ for each realization (simulation) of the interest rate process r: $\int_0^T r(t)dt \approx \Sigma_{i=0}^{n-1} r(t_i)\Delta t$. Since the expected value to find C_0 above cannot be computed explicitly, it will be estimated through the Monte Carlo simulations numerically.

4.3.1 Black-Scholes Model

The Black-Scholes model is used to calculate the value of an option, by considering the stock price, the strike price and the expiration date, the risk-free return, and the standard deviation of the stock's return on the basis of an assumed stochastic process for stock prices. This model was created by Myron Scholes and Fischer Black. However, Robert Merton was the first to publish the paper expanding the understand of the mathematics of option pricing. Merton and Scholes received the Nobel Prize in economics in 1997, but unfortunately, Black had already passed away. The model develops a partial differential equation whose explicit solution, known as the Black-Scholes formula, is widely used in the pricing of European-style options, usually for comparison purposes. It should be noted that the assumptions of the Black-Scholes model (constant model parameters, complete markets and no borrowing constraints etc.) don't hold in real stock markets and shouldn't be used to price the options directly.

4.3.2 Comparison of Black-Scholes Model and Our Model

The call option price can be calculated using the closed form of Black-Sholes formula assuming that the interest rate is fixed. $C_0 = S_0\Phi(d_1) - Ke^{rT}\Phi(d_2)$, where

$$\begin{aligned}
\Phi &: \text{ standard normal cumulative distribution function} \\
d_1 &= \frac{\ln(\frac{S_0}{K}) + \left(r + \frac{1}{2}\sigma^2\right)T}{\sigma\sqrt{T}} \\
d_2 &= d_1 - \sigma\sqrt{T}
\end{aligned}$$

Comparing our model with Black-Scholes formula gave us an idea if our model offered reasonable estimates since Black-Scholes formula is still widely used. For our model, we used 1000 simulations, $\Delta t = 50$ and the parameters metioned previously. Moreover, we tried to see how the correlation ρ would affect the option price. A positive correlation of $\rho = 0.5$ gave us the closest estimate to Black-Scholes formula.

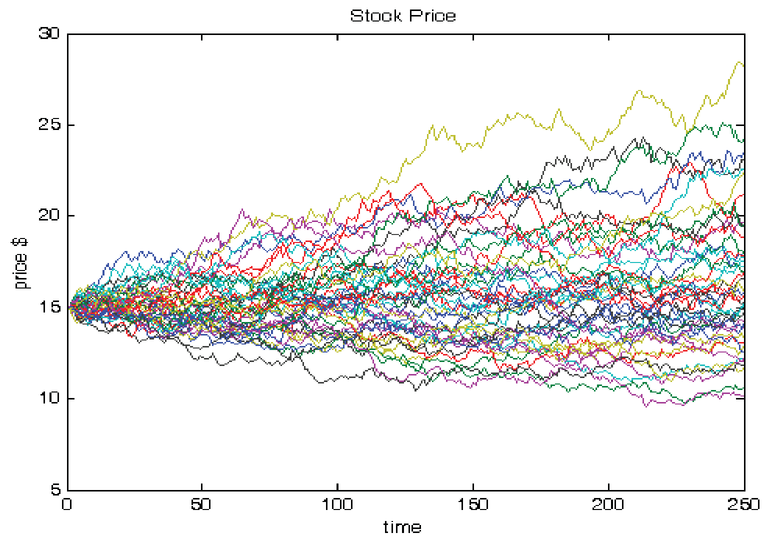
Table 8: Comparison with Black-Scholes Formula

Black-Scholes Formula	$C_0 = 0.1934$
$\rho = -1$	$C_0 = 0.1275$
$\rho = -0.5$	$C_0 = 0.1399$
$\rho = 0$	$C_0 = 0.1611$
$\rho = 0.5$	$C_0 = 0.2049$
$\rho = 1$	$C_0 = 0.2345$

In order to get a better comparison with Black-Scholes formula, we have to fix the interest rate r (take it as a constant) since it is one of the key assumptions of the Black-Scholes model. Fixing the interest rate changes the SDE for the stock price from $dS(t) = S(t)r(t)dt + S(t)\sigma dW(t)$ to $dS(t) = S(t)rdt + S(t)\sigma dW(t)$. Again, we use the discrete version, $\Delta S(t) = S(t)r\Delta t + S(t)\sigma\Delta W(t)$, for the simulation purposes. Moreover, the formula for the fair price also changes to $C_0 = E\left[e^{-rT}(S(T) - K)^+\right]$. The estimate of our modified model is close to Black-Scholes formula.

Table 9: Comparison with Black-Scholes Formula

Black-Scholes Formula	$C_0 = 0.1934$
Modified Model	$C_0 = 0.1832$



4.4 Sensitivity Analysis

4.4.1 Correlation Coefficient (ρ)

A negative correlation between the stock price and the interest rate dynamics, (or equivalently between the Brownian motion processes W and $\hat{W}$), yields a lower call option price because if ρ is negative the "dt" part of stock price that depends on $r(t)$ will be moving in the opposite direction of the "dW" (uncertainty) terms of the stock price (see equation 1) on the average. However, if they are positively correlated, the combined effect will move in the same (up) direction, hence driving the call option price up. Our simulation results confirmed this intuition, showing a monotone increasing relationship between the corelation and the call option price (Table 10). For the numerical computations, we used 1000 simulations and the time increment $\Delta t = 1/250$ (representing roughly 250 trading days/year).

Table 10: Results

ρ	C_0
-1.00	1.0898
-0.75	1.0894
-0.50	1.1158
-0.25	1.1221
0	1.1523
0.25	1.1567
0.50	1.1621
0.75	1.1980
1.00	1.2127

4.4.2 Volatility (σ)

Increasing the stock price volatility, σ, will increase the call option price. This is an expected result, since an increased volatility means a higher chance of surpassing the strike price. In the same manner, an increase in the initial stock price, S_0, will also increase the call option price in the long run.

Table 11: Results

σ	C_0
0.2000	0.2043
0.3000	0.7565
0.4500	1.9591

5 Conclusion

In general, the inversive congruential generators (ICGs) are suitable candidates for the financial applications due to their desirable uniformity and serial-independence advantages, despite their slower computing time compared to linear congruential generators. The combined ICGs are useful extensions of ICGs for ore advanced applications due to their computational advantages. The Four-Source ICGs that we used passed two widely used empirical tests: the χ^2 goodness-of-fit test and the Kolmogorov-Smirnov test.

Our proposed European option pricing model considers the interest rate as an stochastic process rather than a constant. Accordingly, the stock price and the interest rate are correlated in this model. Since the well-known analytical methods for "complete" markets don't apply, we use numerical methods based on Monte Carlo simulations to compute a reasonable price for the European options. The call option price increases with the correlation between the interest rate and the stock price. Furthermore, an increase in the stock price volatility also results in an increase in the call option price as expected.

References

[1] Eichenauer, J. and Lehn, J. (1986). A non-linear congruential pseudo random number generator. Statist. Papers 27, 315-326.

[2] Gentle, J. E. (2003). Random Number Generation and Monte Carlo Methods. Springer, second edition.

[3] Hellekalek, P. (1995). Inversive Pseudorandom Number Generators: Concepts, Results and Links. In Alexopoulos, C. and Kang, K. and Lilegdon, W.R. and Goldsman, D., editor(s), Proceedings of the 1995 Winter Simulation Conference, pp. 255–262. , 1995

[4] Hull, J. C. (2006) Options, Futures and Other Derivatives. Prentice Hall, sixth edition.

[5] Knuth, D.E. (1997). The Art of Computer Programming, volume 2: Seminumerical Algorithms. Addison-Wesley, Reading, MA, 3rd edition.

[6] Leeb, H. and Wegenkittl, S. (1997). Inversive and linear congruential pseudorandom number generators in empirical tests. ACM Transactions on Modeling and Computer Simulation, Vol 7, No. 2, 272-286.

[7] Park, S. K. and Miller, K. W. (1988). Random number generators: good ones are hard to find. Commun. AC31(10):1192-1201.