CHAPTER 3
GENERAL ISSUES IN MANAGEMENT

Real Options Valuation in Energy Investment Projects: Modeling Hedging Strategies Using Genetic Algorithm Software

Marco Antônio Guimarães Dias[1], Mario Antonio Rivera[2]

Abstract: Complex real options models for project economics simulation and modeling must contend with several sources of uncertainty, and the arrays of options to be considered typically include several options to invest in information generation and evaluation. Monte Carlo simulation is considered a good way to address these modeling challenges, but it does not lend itself well to optimization problems. This paper presents a model of dynamic hedging aimed at optimization and using genetic algorithms and Monte Carlo simulation. This approach offers new possibilities for energy investment applications of dynamic hedging based on real options theory.

The modeling summarized in this study relied on the Excel-based commercial software *RiskOptimizer* for a simple case (with known values) and for a more complex real options case involving investment in information. The results from several experimental runs are presented, with suggestions for improvement of the software – the strengths and weaknesses of RiskOptimizer are pointed out in this context – and for new directions for research.

Keywords: real options, genetic algorithms, Monte Carlo simulation, optimization under uncertainty, economic valuation of projects.

1. Introduction

The practical problems associated with the *complex dimensionality*[3] of *decisional modeling*[4] incorporating the evaluation of options have directed some recent real options research[5] to the Monte Carlo simulation approach, due to its modeling flexibility. The major problem is the difficulty of performing optimization analysis together with simulation projections, which is necessary for American-type[6] options. Among the papers that use new optimization techniques in an effort to tackle these difficulties are Cortazar & Schwartz (1998) and, Longstaff & Schwartz (1998), Broadie, Glasserman & Jain (1997), and Ibáñez & Zapatero (1999).

---

[1] Ph.D., Pontifical Catholic University, Brazil

[2] Ph.D., University of New Mexico, USA

[3] The so-called "curse of dimensionality" is the exponential computational time explosion associated with the growth of problem dimensionality. Some real options papers, such as Dias (1997), consider several sources of uncertainty (economic, technical and strategic), but use simplified models to get practical results.

[4] The so-called "curse of modeling" is associated with problem formulation with an explicit system. Changing some aspects of the model makes it necessary to change all of the optimization procedure. The use of a simulator to avoid the traps of complex modeling is one goal of this paper, accounting for its use of genetic algorithms together with Monte Carlo simulation.

[5] See Dias (1999) for a real options literature review and a consideration of recent trends. See Dixit & Pindyck (1994) and Trigeorgis (1996) for the general theory of real options and applications.

[6] American options can be exercised before expiration, so there is the problem of finding the optimal earlier. European options can only be exercised at the expiration date.

This study presents another possibility for a Monte Carlo simulation of real options problems: the use of an evolutionary computing approach, specifically the use of *genetic algorithms* (GA) as the optimizing tool. The expectation is that initial estimated values with high "fitness" will transfer their genetic characteristics to their offspring. It is assumed that high fitness means high real options value.

The use of computational intelligence techniques in real options applications is rare. A working paper by Taudes & Natter & Trcka (1996) used neural networks for the purpose of real options valuation of a flexible manufacturing system. These authors settled on dynamic programming under uncertainty, with *simulated annealing* determining the network weights of the neural net, which are taken to approximate the real option value function. The authors addressed uncertainty through a Monte Carlo simulation. Such combinations of Bellman's dynamic programming and neural networks have been termed *neuro-dynamic programming* (for the underlying theory, see Bertsekas & Tsitsiklis, 1996).

The cases examined in this study involve a real option to invest in petroleum field development. The case analysis is informed by several papers applying computational intelligence to petroleum production problems, for example Rocha & Morooka & Alegre (1996, which applies *fuzzy logic* to the drilling, pumping, and processing of offshore oil, in a project designed to cut costs). Bittencourt & Home (1997) applied genetic algorithms to the dynamic valuation of petroleum reservoirs. Two cases are explored in the present study, one without information investment and a more complex one with such investment.

The study is divided as follows. Section 2 presents the real options case studies. Section 3 presents the strength and weakness of the software product *RiskOptimizer* from a genetic algorithms perspective. Section 4 presents simulations using RiskOptimizer, while Section 5 provides the results of the simulation. Section 6 presents conclusions and suggestions based on the foregoing.

2. The Real Options Case: Petroleum Field Development

An oilfield that has been discovered and delineated remains undeveloped. The estimated reserve is 120 million barrels. The investment required to develop[1] the oilfield is estimated at US$ 480 million (all present values). The investment is not obligatory but optional, and this option expires in two years. Two variants are considered, expiration in ten years and expiration in two years. Standard value estimation for the petroleum field is given by the static *net present value* (NPV). The NPV expression can be written as follows:

$$NVP = qPB - D\,, \tag{1}$$

where q = economic quality of the reserve[2], estimated in this instance at 20%;
P = petroleum price, supposing that the current value is US$ 20/bbl;
B = reserve size, estimated at 120 million barrels[3]; and
D = development costs, assumed to be US$ 480 million.
(Note that for the base case value given above, the NPV is zero [= 0.2 x 20 x 120 – 480]).

One may first consider oil prices as the only source of uncertainty, with probability distributions over time as stochastic price fluctuations. For the sake of simplicity, it may be assumed

[1] In order to develop the oilfield is necessary to drill development wells, to build/buy/rent a production platform and its processes facilities, to launch pipelines, etc.

[2] Introduced by Dias in Stavanger's Workshop on Real Options, May 1998. The reserve is more valuable as higher its q. The value of q depends of several factors: the permo-porosity properties of the reservoir-rock; the quality and properties of the oil and/or gas; reservoir inflow mechanism; operational cost; country taxes; cost of capital; etc. For details about q see: http://www.puc-rio.br/marco.ind/quality.html

[3] All the oil reserves have associated gas. Consider the 120 million barrels as *total* equivalent barrels.

that oil prices follow *Geometric Brownian Motion*[1] in the form of a *risk-neutralized* stochastic process[2]

$$dP = \left(r - \delta - 0.5\sigma^2\right)Pdt + \sigma Pdz\ , \qquad (2)$$

where r = interest rate, assumed to be 8% per annum (hereafter p.a.);

δ = convenience yield of the oil, also assumed to be 8% p.a.;

= volatility rate for oil prices, assumed to be 25% p.a.; and

dz = Wiener increment = $\varepsilon\sqrt{dt}$, where $\varepsilon \sim N(0, 1)$.

Given the aforementioned market uncertainty and assuming that the option will take ten years to expire (so that there is a large timing flexibility to invest), how much value can be assigned to real option rights? What is the *decision rule*?

An *investment threshold* gives the decision rule. The threshold is the critical oil price level that renders immediate investment optimal. The threshold level maximizes the real options value, i.e., threshold value determines the optimal exercise of the real option. The option exercise strategy involves hedging, such that one exercises the option only at or above this threshold level, thereby maximizing the real options value. The threshold *curve* will be estimated by GAs; it is a theoretical curve calculated by finite differences, as shown in Figure 1 below (projected using the "Extendible" software product [Dias & Rocha, 1998]).

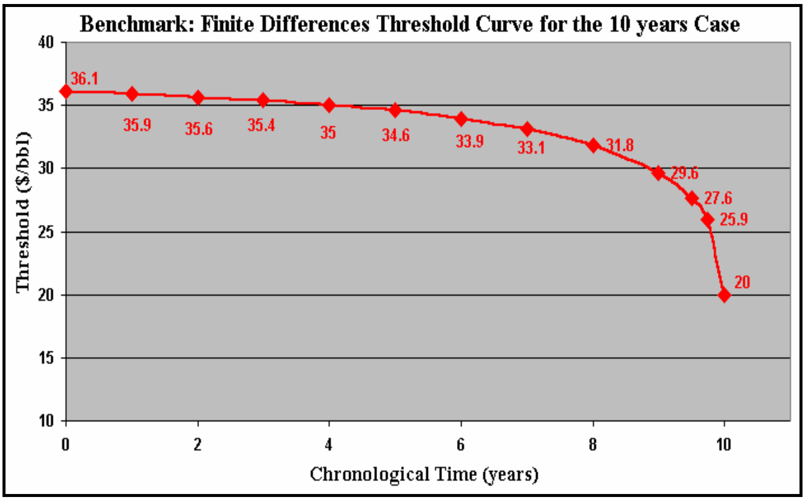


Fig. 1. The Theoretical Threshold Curve Calculated with Finite Differences

There are at least two ways to use GA to get the threshold optimal level. One is letting GA work freely. Another is to approximate the threshold with one or more functions, with or without free points at the extremes. By changing the time variable (using time to expiration instead of chronological time), several GA experiments modeled the threshold curve with two free points and two logarithm functions[3]. This is illustrated by Figure 2 immediately following:

[1] The simulation of a more realistic model is also possible and straightforward. For example the mean-reversion with jumps, as in Dias & Rocha (1998). The idea here is the simplicity for comparison purpose.

[2] For a discussion and the discretization of this equation in order to perform the Monte Carlo simulation, see for example Clewlow and Stickland (1998, mainly the chapter on Monte Carlo simulation).

[3] For American *put*, Ju (1998) uses a *multipiece exponential function* to approximate the early exercise boundary (threshold curve). For an American *call*, exponential approximation is the analogue.

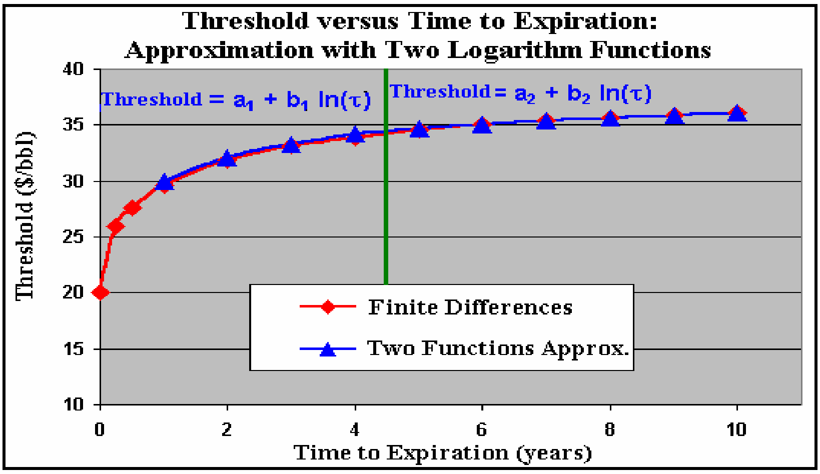


Fig. 2. Threshold Approximation with Two Logarithm Functions

The second variant analyzed is very similar to the first one, but with earlier expiration at two years. Reducing the time horizon allowed experimental simulation with greater precision.

A distinct second case was developed that was rendered significantly more complex by incorporating investment in information and revealing the uncertainties associated with two technical parameters, reserve size (B) and quality (q). The case therefore presents three sources of uncertainty in all: information costs, reserve size, and quality.

The information in this case is assumed to cost US$10 million (cost of an *information-acquisition well*[1]). After the investment in information, there are several possible scenarios for B and q (though expected values remain the same as with the previous variations). The information acquired is assumed to reveal these scenarios with the following *triangular probability distributions* (minimum, probable, maximum):

B ~ Triang (90, 120, 150) [million barrels]

q ~ Triang (15%, 20%, 25%)

These distributions are sufficiently large so that a given investment in information is considered essential in this particular case. Comparing cases with and without investment in information confirms the expectation that investment in information is by far the best course of action. A yet more complex case could entail choosing both the optimal time for investment in information and the optimal timing of the petroleum field development investment as such.

3. RiskOptimizer and the Genetic Algorithms

As previously indicated, this study uses genetic algorithms (GA) in order to optimize the real option value. Genetic algorithm computational modeling was developed by John Holland (see Holland, 1975); excellent descriptions of GA are found in Goldberg (1989), Davis (1991), and Michalewicz (1996).

There are several applications of evolutionary computation in financial options pricing which can be adapted to real options applications. One powerful tool involves the application of symbolic regression, which is considered a branch of genetic programming (see Koza, 1992, chapter 10). The work of Keber (1999) is another good example, using symbolic regression to get an analytical approximation for an American *put option value*. Another alternative that could be applied to options valuation entails solving a stochastic partial differential equation with analytical approximations to both threshold value and the option value itself.

[1] Lower-cost wells drilled with lower diameter to gain requisite information while preserving the option of identifying producer wells.

The present study applies GA to the task of estimating the threshold curve of a real option application. Customarily, the threshold curve is determined together with the real options value by solving a stochastic *partial differential equation* (PDE). In this study, the GA application projects possible threshold curves. For each trial, a Monte Carlo simulation is performed in order to evaluate "fitness." In GA terms, each trial is equivalent to an *organism* (or *chromosome*), and the "fitness" function is the *real options value* (also called *dynamic net present value*[1]). The total number of simulations is equal to the number of valid[2] "organisms." For a run using one thousand such organisms, with one thousand iterations for each simulation, the equivalence is to a simulation with one million iterations, with an associated computational time cost.

The way selected to set the initial threshold projection (tantamount to an informed guess) as well as to run the Monte Carlo simulation was informed in part by Winston (1999), who developed a case of dynamic hedging using options to make investment decisions in the expansion of a gold mine.

The software utilized was the Excel-based add-in *RiskOptimizer*, from Palisade. RiskOptimizer incorporates and integrates two popular Palisade software packages: @Risk, which performs Monte Carlo simulation, and Evolver, which performs complex optimization using genetic algorithms. This section focuses on both the simulation and optimization aspects of RiskOptimizer.

RiskOptimizer, as Evolver, uses *steady state reproduction* with parameter one. In other words, after the creation of the initial "population" of GA trials, each new solution created in the reproduction phase using the genetic operators replaces the worst solution (lower real option value).

Consideration of whether steady-state reproduction is better or worse than *generational replacement*[3] is beyond the scope of this paper, but the practical insight given by Davis (1991) is very pertinent to an analysis of RiskOptimizer's performance potential. Davis pointed out that steady-state reproduction has been used with success for cases *without_duplicates*. It is a "reproduction technique that discards children that are duplicates of current chromosomes in the population rather than inserting them into the population" (Davis, 1991, p. 37).

Unfortunately, RiskOptimizer does not permit the elimination of duplicates, weakening the steady state reproduction technique. Without the elimination of duplicates, the user may be faced with a *premature convergence problem*, due the loss of diversification in the population, before reaching the optimal solution. Introducing more diversity into the population can reduce the GA problem of premature convergence behavior, though still without guarantee of optimality. The elimination of duplicates is intended to increase diversification.

The Figure below illustrates the problem of premature convergence due to the loss of diversification. The genes panel (right bottom chart) and the panel of gene values (left bottom chart) show that the population retains the same genes. The performance panel (left upper chart) and the organisms-in-population panel (right upper chart) indicate that the few remaining organisms are "evolving" to become identical genes.

The obvious recommendation is that the software developer (Palisade) includes, at least as a user option, the capability of eliminating duplicates. Even greater flexibility could be attained with the capability of setting the maximum duplicates percentage, so that a limited number of duplicates would be permitted, with the limit defined by the user.

---

[1] The NPV is calculated under uncertainty by running a Monte Carlo simulation, given the optimal exercise of the options available (by exercising the option at or above the threshold line). Therefore, this is a *dynamic NPV* estimation, in contrast with *static NPV* calculations that fail to consider (a) decisional uncertainty and (b) the timing option.

[2] Valid means "satisfies the restrictions." For the not-valid organisms, no simulation is performed in the program, saving time; therefore, the generation of non-valid organisms is not critical in practice.

[3] *Generational replacement* means that all the population will be replaced with the new organisms created by the evolutionary process. Some of them could be identical inter-generationally, since in most cases the previous best is preserved.

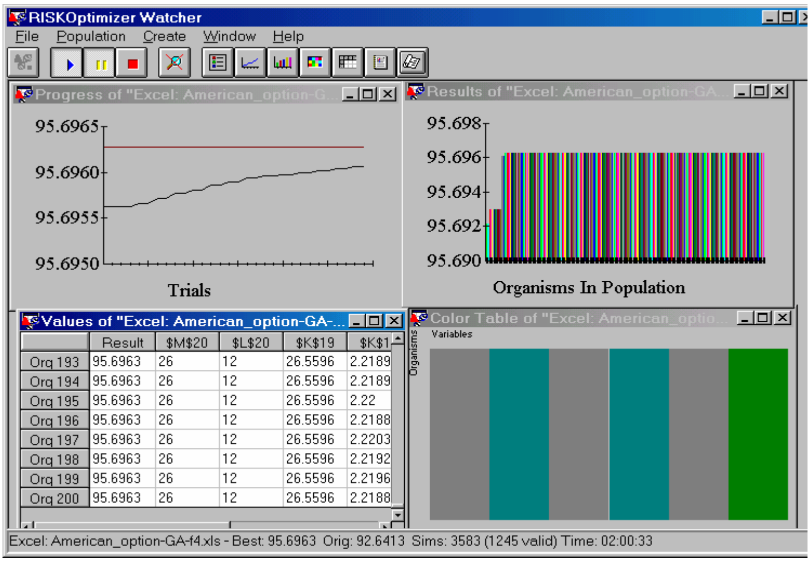


Fig. 3. Premature Convergence by Lost Diversification (case 10 years expiration, two functions)

As noted by Bittencourt & Home (1997, p.547), some researchers prevent premature convergence in GA simulations by using strategies such as *niching* (niche-creation), *increasing the mutation rate*, and *incest-prevention* (limiting endogamous interactions among similar elements). RiskOptimizer permits increases in the mutation rate during the optimization process. However, in general this device was insufficient if population diversification was insufficient.

Hybridization or combination of genetic algorithms with other optimization algorithms is another suggestion for improvement in the GA simulation performance. One variant is the hybridization of GA with *simulated annealing* (SA). The performance of SA is often frustrated by slow convergence, but in combination with GA it introduces more diversity into the population, avoiding premature GA convergence[1], as well as avoiding the long computation time required by SA (see Sait & Youssef, 1999, pp.164, 349-350).

Another hybrid GA is applied in Bittencourt & Home (1997) to petroleum reservoir optimization; GA is hybridized with the optimization algorithms *tabu search* (TS) and *polytope search*. Tabu search is a generalization of local search that considers, in addition to an objective function value, the given search history, the region being searched, and similar factors (Sait & Youssef, 1999, pp. 240-241).

This approach precludes revisiting the current search region for some finite number of iterations, as it provides a mechanism that forces the search into other regions of the domain (Bittencourt & Home, 1997, p.548). Polytope2 search is intended to accelerate search convergence, reducing computation time, but it has the drawback of excessive sensitivity to the initial size and location of the polytope.

Another interesting GA hybrid involves both simulated annealing and tabu search, which is described as a reasonable combination of local search and global search, with TS incorporated to prevent local optima convergence. Here the search domain is globally searched with GA, the GA offspring are introduced in the population using as criteria the acceptance probability of a SA, and the neighborhood of the accepted chromosome is searched by TS. In a maintenance-scheduling problem, the hybrid GA + SA + TS found better results than the pure GA or the hybrid GA + SA (see Sait & Youssef, 1999, p.350).

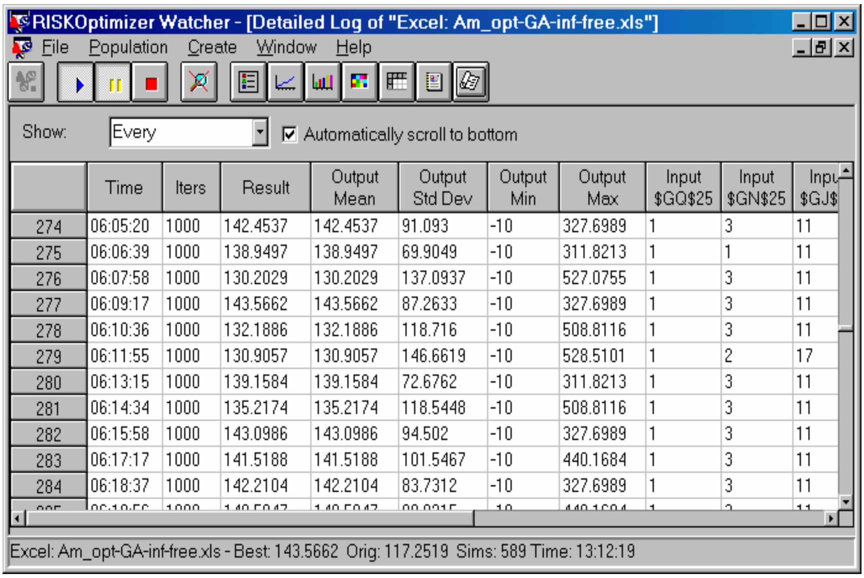
---

[1] An instance might entail improving the convergence of a GA by using the acceptance probability of SA as the acceptance criterion of GA's new chromosomes trials (see Sait & Youssef, 1999, p.349-350).

[2] Polytope is a geometric figure bounded by hyperplanes. In this algorithm, the polytope flips about the centroid towards the optimum (see Bittencourt & Home, 1997, pp.547-548).

The second problem with steady-state reproduction is the case when the evaluation function is *noisy*, as pointed out by Davis (1991). Noisy evaluation means that the fitness function (here the simulated real option) returns different values each time the organism is evaluated. Davis indicates that, in steady-state reproduction, "a lucky good evaluation for a chromosome will be forever associated with it, and, since a good chromosome won't be deleted from the population using this technique, the spurious evaluation will cause the bad genetic material to disseminate throughout the population" (Davis, 1991, p.40).

In the present case, which is less complex than Davis', the evaluation function is only somewhat noisy, due specifically to the precision level specified for the Monte Carlo simulation. Simulation takes up most of the computation time in the present study, and increasing precision by way of additional iterations can become a very time-consuming process. The Figure 4 suggests one of RiskOptimizer's relative strengths: the comparatively short time spent for each trial[1].



	Time	Iters	Result	Output Mean	Output Std Dev	Output Min	Output Max	Input $GQ$25	Input $GN$25	Inpu GJ
274	06:05:20	1000	142.4537	142.4537	91.093	-10	327.6989	1	3	11
275	06:06:39	1000	138.9497	138.9497	69.9049	-10	311.8213	1	1	11
276	06:07:58	1000	130.2029	130.2029	137.0937	-10	527.0755	1	3	11
277	06:09:17	1000	143.5662	143.5662	87.2633	-10	327.6989	1	3	11
278	06:10:36	1000	132.1886	132.1886	118.716	-10	508.8116	1	3	11
279	06:11:55	1000	130.9057	130.9057	146.6619	-10	528.5101	1	2	17
280	06:13:15	1000	139.1584	139.1584	72.6762	-10	311.8213	1	3	11
281	06:14:34	1000	135.2174	135.2174	118.5448	-10	508.8116	1	3	11
282	06:15:58	1000	143.0986	143.0986	94.502	-10	327.6989	1	3	11
283	06:17:17	1000	141.5188	141.5188	101.5467	-10	440.1684	1	3	11
284	06:18:37	1000	142.2104	142.2104	83.7312	-10	327.6989	1	3	11

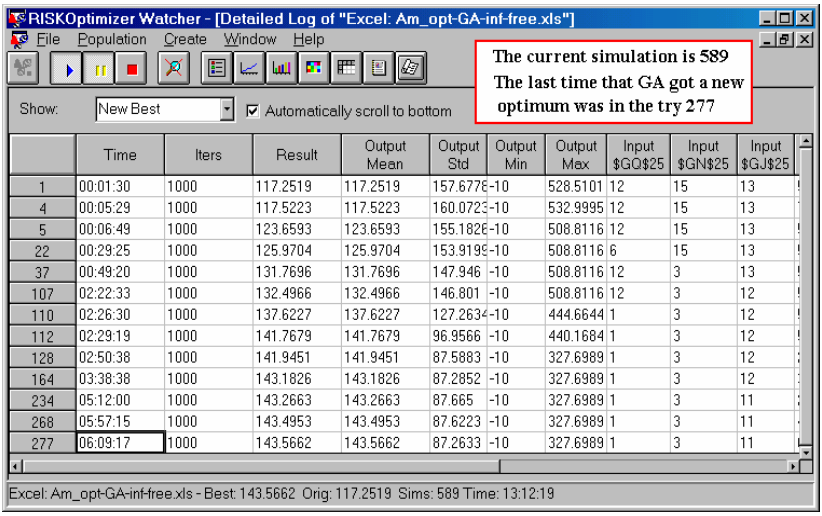
Fig. 4. Risk Optimizer's relative strength

Another GA problem faced with some experiments is the problem of *epistasis*. This occurs when there are interactions between genes (one gene value affects other gene values), bringing additional difficulties to the search for optimal or near-optimal solutions. Some authors suggest a means toward the reduction of the epistasis problem, namely the use of *inversion operators*, which consist of a position-switch for two randomly chosen genes (see Pacheco, 2000, p.3). The approach best suited for strong epistasis problems may be the use of so-called "messy" GA, which permits the given gene to change its position inside the chromosome, for a more convenient position for purposes of crossover operations, tending to preserve good *schemata*.

The next Figure presents the problem of evolution to the best solution using RiskOptimizer. The table shows only trials which became a "new best." Note that after the last "new best," in the 277th trial, more than 300 new trials were performed (for more than 7 hours) without any new successes, even with an increase in the mutation ratio (from 1% up to 30%) during the last

[1] If 1,000 iterations were not sufficient and more precision were necessary, the required computation time would be exponentially higher and computationally intractable. The evaluation of *every* alternative in this example with investment in information, even with a computer eighty times faster than the one used in the case, would take about 1.5 billion years for all alternatives in the search space! GA can be viewed as a global search technique that intends to reduce the search time using an evolutionary approach to get a proximate, *near- optimal* solution.

100 trials. This could be an indication of global optimum or could be a "lucky" simulation that put iteration number 277 at the top, so that even stronger alternatives did not match the result obtained in the 277th case.



	Time	Iters	Result	Output Mean	Output Std	Output Min	Output Max	Input $GQ$25	Input $GN$25	Input $GJ$25
1	00:01:30	1000	117.2519	117.2519	157.6778	-10	528.5101	12	15	13
4	00:05:29	1000	117.5223	117.5223	160.0723	-10	532.9995	12	15	13
5	00:06:49	1000	123.6593	123.6593	155.1826	-10	508.8116	12	15	13
22	00:29:25	1000	125.9704	125.9704	153.9199	-10	508.8116	6	15	13
37	00:49:20	1000	131.7696	131.7696	147.946	-10	508.8116	12	3	13
107	02:22:33	1000	132.4966	132.4966	146.801	-10	508.8116	12	3	12
110	02:26:30	1000	137.6227	137.6227	127.2634	-10	444.6644	1	3	12
112	02:29:19	1000	141.7679	141.7679	96.9566	-10	440.1684	1	3	12
128	02:50:38	1000	141.9451	141.9451	87.5883	-10	327.6989	1	3	12
164	03:38:38	1000	143.1826	143.1826	87.2852	-10	327.6989	1	3	12
234	05:12:00	1000	143.2663	143.2663	87.665	-10	327.6989	1	3	11
268	05:57:15	1000	143.4953	143.4953	87.6223	-10	327.6989	1	3	11
277	06:09:17	1000	143.5662	143.5662	87.2633	-10	327.6989	1	3	11

Fig. 5. RiskOptimizer Watcher Showing the New Best Occurrence (case with information)

Even with the limitations of the GA method used by RiskOptimizer, this flexible software tool allowed for new approaches to decisional modeling not previously found in the real options literature. This point will be revisited in sections 5 and 6.

4. RiskOptimizer and Monte Carlo Simulation

In the previous section it was suggested that one problem associated with the steady-state reproduction device used by RiskOptimizer is the noisy evaluation of fitness. This section examines an extension of the noise problem by considering the precision of Monte Carlo simulation in combination with RiskOptimizer for the purposes of options valuation.

Wayne Winston, who has published several books on decision analysis, and who has carried out simulations using Palisade's products (both @Risk and RiskOptimizer), raised the following question in the process of evaluating an European put option: "After 10,000 iterations why [are we no closer] to the price of the put? The reason is that this put only pays off *on very extreme results* [author's emphasis]. It takes many iterations to accurately represent extreme values of a Lognormal variable" (Winston, 1998, p.335).

RiskOptimizer, as @Risk, uses a *variance reduction*[1] technique named *Latin hypercubic sampling*. This increases precision as compared with the pure Monte Carlo sampling. Press et al. (1992) provide the algorithm for this sampling application, while Moro (1995) undertakes comparison of the computational efficiency of Latin hypercubic sampling and the efficiency levels characteristic of other techniques. Latin hypercubic is a special type of stratified sampling for higher dimensions (see the discussion in Boyle et. al. [1997]). Another variance-reduction technique that can be combined with stratified sampling is *importance sampling*, which concentrates samples in the area where they are most effective by incorporating *a priori* information. It is thus possible to generate uniformly distributed samples with Latin hypercubic.

[1] Variance reduction techniques are tools that use known information about the problem in order to estimate parameters of simulated distribution with lower variance (therefore resulting in an improved estimator). Among the best-established techniques are stratified sampling, importance sampling, conditional Monte Carlo, moment-matching, correlated sampling; and the reduction of dimensionality.

Number Theory can help reduce variance with *low-discrepancy sequences*[1], in so-called *quasi-random simulation*. In this vein, Birge (1995), among others, proposes Quasi Monte Carlo (QMC) methods for options pricing valuation with complex contingencies. This alternative relies on the numerical properties of special sequences, namely quasi-random numbers or low-discrepancy sequences[2] based purely on numerical (not statistical) analysis.

Clewlow & Strickland (1998, pp.129-133) present charts with plots comparing random and quasi-random numbers. They show clearly that the first one leaves "several 'clumps' of points close together and empty spaces," whereas the quasi-random plots "are much more evenly distributed but still appear somewhat random." They present the quasi-random Faure[3] numbers approach, which requires very few iterations for near-zero error rates. Press et al. (1992) present an algorithm that generates Sobol's sequences, while Moro (1995) provides an efficient method to obtain standard normal sequences from uniform sequences, which permits a more efficient use of Quasi Monte Carlo approaches.

While the anticipated Monte Carlo simulation error is $O(1/\sqrt{N})$, in quasi-random sequences error generally reduces to *O(1/N)*, with N being the number of observations. This means that pure Monte Carlo requires the square of the number of iterations demanded by Quasi Monte Carlo to obtain the same error rate. While there are still difficulties attendant to high-dimensionality cases,[4] there is promise in recent research applications of QMC in such instances. Willard (1997) demonstrates that in some cases effective dimensionality can be reduced[5]. Paskov & Traub (1995) present an efficient QMC simulation for a 360-dimension valuation of a financial derivative (mortgage valuation—cf. Galanti & Jung [1997]). In general, variance-reduction techniques can improve simulation performance (in importance sampling, for example); however, some reduction-variance techniques are not suited for use with Quasi Monte Carlo[6].

5. Results: Theoretical versus Genetic Algorithms

The next Figure presents the problem of optimization in terms of projected levels of possible threshold curves in relation to the theoretical threshold curve (finite differences). Different threshold curves (above and under the theoretical level, with distance from 0% to +or - 30%) were simulated and the values plotted in the chart. It should be noted that there is a near-plane region

---

[1] "Discrepancy" measures the extent to which the plot points are uniformly dispersed inside a region. With low-discrepancy sequences, the given sequence of points remains fairly evenly dispersed throughout the region as points are added (Boyle et al., 1997).

[2] The main type is called "*Sobol's quasi-random sequence.*" There are many ways of producing quasi-random numbers—see Clewlow & Strickland (1998, pp.129-133) for a discussion. Birge (1995) prefers Sobol rather than Faure's quasi-random numbers, arguing that the second is not as apt for high- dimensionality problems. Brotherton-Ratcliffe (1995) describes a new technique to generate Sobol's sequences, claiming that it is faster and more precise.

[3] Willard (1997), however, indicates that Faure sequences are not as accurate and require significantly more computation time than Sobol and Halton ones.

[4] Bouleau & Lépingle (1994), Aragão & de La Roque (2000, pp. 4-5) among others, point out that Sobol's sequence has problems with high dimensions. Papageorgiou (1999) indicates that although QMC's worst-case convergence is $(\log n)^d/n$, this can be improved with some special integral approximations, with the worst case becoming $(\log n)^{1/2}/n$ (both convergence and error). Ökten (1999) used a hybrid Monte Carlo sequence (or *mixed sequence)* in an application to options pricing, with low-discrepancy sequences for the "low dimensional part" and random vectors for the "high-dimension part."

[5] Willard (1997) uses the technique of "Brownian bridge," from a stochastic calculus toolkit. He argues that it substantially reduces errors and improves the convergence of price estimates for high-dimensional pricing problems. Morokoff (1997) presents two techniques for a 360-dimensional case, *generalized Brownian bridge* and *particle reordering*. Acworth et al. (1996) uses a technique called *principal components analysis* with Sobol's sequences, claiming better results than Brownian-bridge.

[6] Galanti & Jung (1997) report studies indicating that antithetic variables and stratified sampling alter the order of low-discrepancy sequences, and so may not help in simulation performance.

close to the optimum value (in red)[1]. This region is here called an *optima region* because (given the simulation's degree of precision) every point could be considered an optimum.

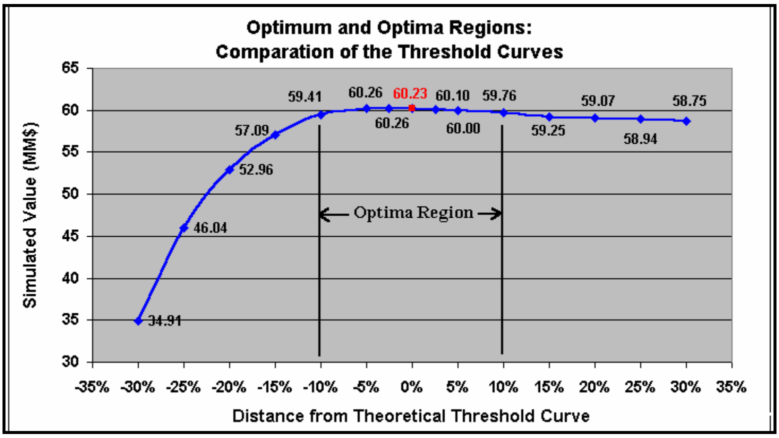


Fig. 6. Optima Region for the Real Options Problem (two–year case)

An interesting feature of this particular chart is found in the regions neighboring the optima region. Considering that there is always some imprecision in the model parameters, and looking at the inclination of the curve plotted here, the higher neighboring region (adjacent to the right) is less risky than the lower neighboring region (adjacent to the left) in relation to possible loss of value.

Some results of GA experiments for the ten-year case are presented in the Figure that follows. There are some cases below and others above the theoretical optimum. Some simulations showed GA solutions that are under the theoretical level for a time horizon interval and above the theoretical level in the rest of the time horizon (see the f6a case in the figure). Even crossing the theoretical optimum curve, the solution was evidently within the just-discussed "optima region."

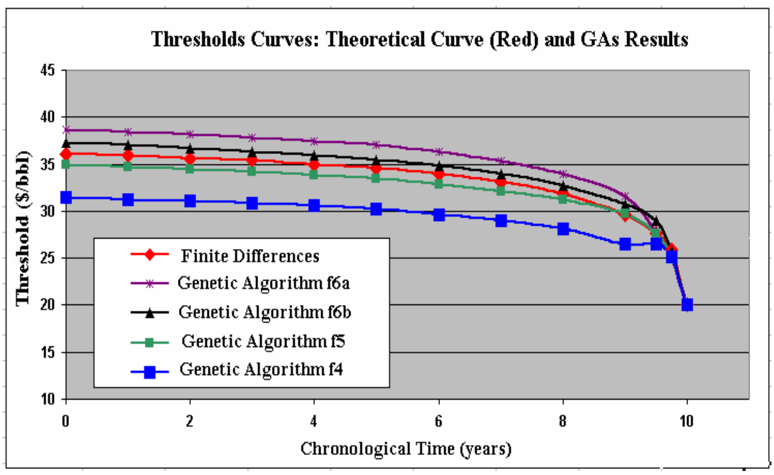


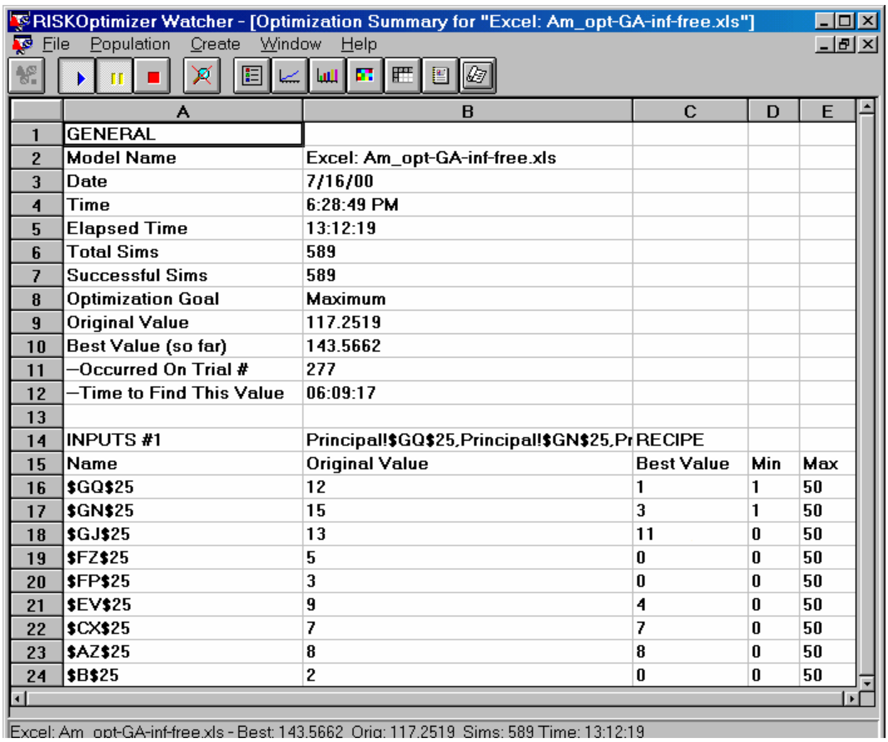
Fig. 7. Threshold Curves: Theoretical x GAs (ten-year case)

[1] This (red) value was simulated (50,000 iterations; $\Delta t = 0.01$ y.) using the theoretical *threshold*. The correct value (60.17) is very close of the simulated (60.23). Due to the range of simulation error, two lower levels (2.5% and 5% below the theoretical level) attained slightly better values (60.26, 0.05% higher).

The optima region makes the close competition problem even more complicated due the noisy evaluation function that derives from simulation precision. This factor changes the fitness order of the trials during optimization.

For the ten-year case, the theoretical real options value is estimated in US$ 96.68 million, using the finite differences method[1] (and "Extendible" software, see Dias & Rocha, 1998). Several GA trials got fitness results in the precision range of the simulation, between US$ 95 million and US$ 97 million. The creation of simulation noise in the course of steady-state reproduction in RiskOptimizer becomes a practical problem when there is reliance on a low number of iterations due to computational time restrictions.

The more complex case with investment in information uses two years as time to the option's expiration. In this case, instead the threshold being tied to the oil price P level, it refers to the whole operating project value V (= q P B), a sufficiently high value for exercise of the option. For more general modeling, the threshold is normalized as the ratio V/D. The traditional NPV rule directs one to invest if this ratio is above 1, whereas the real options threshold will demand a higher V/D level in order to proceed. In the present instance, the threshold curve is not known *a priori*. The Figure below presents the RiskOptimizer summary for the case with investment in information. The window shows the range of values for all the inputs, and some highlights of the optimization operation performed up to that point.



	A	B	C	D	E
1	GENERAL				
2	Model Name	Excel: Am_opt-GA-inf-free.xls			
3	Date	7/16/00			
4	Time	6:28:49 PM			
5	Elapsed Time	13:12:19			
6	Total Sims	589			
7	Successful Sims	589			
8	Optimization Goal	Maximum			
9	Original Value	117.2519			
10	Best Value (so far)	143.5662			
11	—Occurred On Trial #	277			
12	—Time to Find This Value	06:09:17			
13					
14	INPUTS #1	Principal!$GQ$25,Principal!$GN$25,Pr	RECIPE		
15	Name	Original Value	Best Value	Min	Max
16	$GQ$25	12	1	1	50
17	$GN$25	15	3	1	50
18	$GJ$25	13	11	0	50
19	$FZ$25	5	0	0	50
20	$FP$25	3	0	0	50
21	$EV$25	9	4	0	50
22	$CX$25	7	7	0	50
23	$AZ$25	8	8	0	50
24	$B$25	2	0	0	50

Fig. 8. Summary for the GA Modeling – Case with Information Investment

Note that the maximum real options value here (US$ 143.566 million) is much higher than the comparison value without investment in information about the reserve size and the quality of the reserve (US$ 60.17 million). The information in this case is much more valuable than the US$ 10 million that it costs to acquire it.

[1] Using the analytic approximation considered one of most precise for American options, Bjerksund & Stensland, the value is very close: US$ 96.38 million.

The Figure below shows the GA results for the case with investment in information (red line) and the normalized threshold (in terms of V/D instead of P) of the previous case (i.e., the one without investment in information).

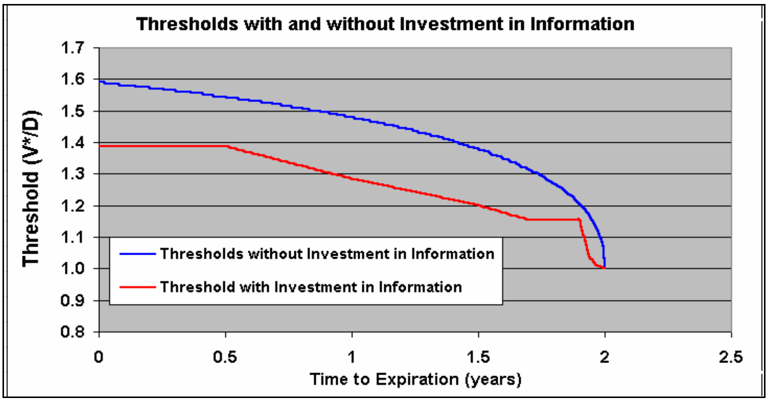


Fig. 9. The Thresholds Curves with and without Option to Buy Information

Note that, in light of the information acquired, the petroleum concession owner should be more willing to invest in development of the oilfield (due to a lower threshold) than otherwise.

6. Conclusions and Suggestions

The foregoing analysis resulted from a genetic algorithm application as an optimization tool for dynamic investment decisions under uncertainty, and it afforded new insights about real options evaluation not previously addressed in the literature. The close competition of several threshold alternative curves drove the search to the "optima region," within which any one solution is indistinguishably close to an absolute-optimum solution. These findings have implications for real options evaluations and in general for dynamic hedging and the management of real options portfolios.

Several strengths and weakness of RiskOptimizer software were noted in this paper. The most crucial suggestion is the inclusion of an option for the elimination of duplicates, in order to preserve population diversification, thereby preventing premature convergence of the GA. Another RiskOptimizer limitation concerned the problems associated with efforts to increase simulation precision by increasing the number of iterations. Increasing the number of iterations was found to bring noise to the evaluation function, making the problem of close competition critical[1].

These caveats notwithstanding, modeling flexibility, an informative and helpful windows interface, and other user-friendly features make RiskOptimizer a very good tool for a first-cut analysis of complex optimization problems. It would be valuable to continue development of this research, using RiskOptimizer but moving to a faster environment (such as C+ programming, for instance), and using Quasi Monte Carlo simulation together with variance-reduction techniques, in order to get faster and more reliable solutions. In this context, a more specific GA could be developed, allowing for the use of *hybrid genetic algorithms* for the modeling of a wide range of dynamic optimization problems under conditions of uncertainty.

[1] The software review of Bergey & Ragsdale (1999) does not mention the steady-state feature, and so also fails to mention the problems associated with duplicates and with noisy functions, even though the review uses Davis (1991) as reference. The lead author (Dias) tested Evolver 4.0 (the same optimization tool incorporated in RiskOptimizer) with the De Jong functions and with the traveling salesman problem, with good results

References

1. Acworth P. & M. Broadie & P. Glasserman (1996): "A Comparison of Some Monte Carlo and Quasi-Monte Carlo Techniques for Option Pricing," in Niederreiter et al. (Eds.), *Monte Carlo and Quasi-Monte Carlo Methods 1996* - Springer-Verlag/New York, Lectures Notes in Statistics, 1998, pp.1-18
2. Aragão C.S.L. & E. de La Roque (2000): "Simulação de Monte Carlo com Volatilidade Estocástica para a Análise do Risco de uma Carteira de Opções" ("Monte Carlo Simulation with Stochastic Volatility for a Risk Analysis of a Portfolio of Options"), *Working Paper*, Banco BBM, 2000, 16 pp. (in Portuguese).
3. Bergey P.K. & C.T. Ragsdale (1999): Software Review "Evolver 4.0," *OR/MS Today*, August 1999, pp.44-47
4. Birge, J.R. (1995): "Quasi-Monte Carlo Approaches to Option Pricing," Technical Report 94-19, DIOE, University of Michigan, revision of June 1995, 28 pp.
5. Bertsekas D.P. & J.N. Tsitsiklis (1996): *Neuro-Dynamic Programming*, Athena Scientific, 1996, 491 pp.
6. Bittencourt A.C. & R.N. Home (1997): "Reservoir Development and Design Optimization," *Proceedings* of the 1997 SPE Annual Technical Conference and Exhibition, San Antonio, Texas, October 1997, pp.545-558.
7. Bouleau N. & D. Lépingle (1994): *Numerical Methods for Stochastic Processes*, John Wiley & Sons, 1994, 359 pp.
8. Boyle P. & M. Broadie & P. Glasserman (1997): "Monte Carlo Methods for Security Pricing," *Journal of Economic Dynamics and Control*, June 1997, vol.21, nº 8-9, pp.1267-1321
9. Broadie M. & P. Glasserman & G. Jain (1997): "Enhanced Monte Carlo Estimates for American Option," *Journal of Derivatives*, vol.5, pp.25-44
10. Brotherton-Ratcliffe R. (1994): "Monte Carlo Monitoring," *Risk*, vol.7 nº 12, December 1994, pp.53-57
11. Clewlow L. & C. Strickland (1998): *Implementing Derivatives Models*, John Wiley & Sons Ltd., 1998, 309 pp.
12. Cortazar G. & E.S. Schwartz (1998): "Monte Carlo Evaluation Model of an Undeveloped Oil Field," *Journal of Energy Finance & Development*, vol.3, nº 1, pp.73-84; and also *Working Paper* UCLA #1-98, PUC-Chile and UCLA, January 1998, 20 pp.
13. Davis L. (Eds.) (1991): *Handbook of Genetic Algorithms*, Van Nostrand Reinhold, New York, 1991, 385 pp.
14. Dias M.A.G. (1999): "A Note on Bibliographical Evolution of Real Options," in L. Trigeorgis, Eds., *Real Options and Business Strategy – Applications to Decision Making* – Risk Books, 1999, pp. 357-362
15. Dias M.A.G. (1997): "The Timing of Investment in E&P: Uncertainty, Irreversibility, Learning, and Strategic Consideration," *SPE Paper* nº 37949, presented at 1997 SPE Hydrocarbon Economics and Evaluation Symposium, Dallas 16-18 March 1997, Proceedings pp.135-148
16. Dias M.A.G. & K.M.C. Rocha (1998/9): "Petroleum Concessions with Extendible Options Using Mean Reversion with Jumps to Model Oil Prices," *Working Paper*, 23 pp., presented at the to the 3rd Annual International Conference on Real Options, June 1999, Netherlands.
17. Dixit A.K. & R.S. Pindyck (1994): *Investment under Uncertainty*, Princeton University Press, Princeton, N.J., 1994, 468 pp.
18. Galanti S. & A. Jung (1997): "Low-Discrepancy Sequences: Monte Carlo Simulation of Option Prices," *Journal of Derivatives*, Fall 1997, pgs.63-83
19. Goldberg D.E. (1989): *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison- Wesley Longsman, Inc., 1989, 412 pp.

20. Holland J.H. (1975): *Adaptation in Natural and Artificial Systems*, MIT Press Edition, 1992 (original 1975), 211 pp.
21. Ibáñez A. & F. Zapatero (1999): "Monte Carlo Valuation of American Options Through Computation of the Optimal Exercise Frontier," *Working Paper*, Instituto Tecnológico Autónomo de México and University of Southern California, August 1999, 30 pp.
22. Ju N. (1998): “Pricing an American Option by Approximating Its Early Exercise Boundary as a Multipiece Exponential Function,” *Review of Financial Studies*, vol.11, nº 3, Fall 1998, pp.627-646.
23. Keber C. (1999): “Option Valuation with the Genetic Programming Approach,” in *Computacional Finance 1999*, Y.S. Abu-Mostafa, B. LeBaron, A.W. Lo, & A.S. Weigend, Eds., MIT Press, 2000, pp. 689-703
24. Koza J.R. (1992): *Genetic Programming – On the Programming of Computers by Means of Natural Selection*, MIT Press, 1992, 819 pp.
25. Longstaff F.A. & E.S. Schwartz (1998): “Valuing American Options By Simulation: A Simple Least-Square Approach," *Working Paper* nº 25-98, UCLA, November 1998, 38 pp.
26. Michalewicz Z. (1996): *Genetic Algorithms + Data Structures = Evolution Programs*, Springer-Verlag, New York, Third Edition, 1996, 387 pp.
27. Moro B. (1995): “The Full Monte,” *Risk*, vol.8, nº 2, February 1995.
28. Morokoff W.J. (1997): “Generating Quasi-Random Paths for Stochastic Processes,” *Working Paper*, UCLA, Mathematics Dept., February 1997, 28 pp.
29. Ökten G. (1999): “Applications of a Hybrid-Monte Carlo Sequence to Option Pricing,” in Niederreiter & Spanier (Eds.), *Monte Carlo and Quasi-Monte Carlo Methods 1998* - Springer-Verlag Berlin Heidelberg New York, 2000, pp. 391-406
30. Pacheco M.A.C. (2000): “Algoritmos Genéticos: Princípios e Aplicações” (“Genetic Algorithms: Principles and Applications”), unpublished paper, ICA, PUC-Rio, 2000, 6 pp. (in Portuguese)
31. Papageorgiou A. (1999): “Fast Convergence of Quasi-Monte Carlo for a Class of Isotropic Integrals,” *Working Paper*, Columbia University, Dept. of Computer Science, June 1999, 11 pp.
32. Paskov S. & J. Traub (1995): “Faster Valuation of Financial Derivatives,” *Journal of Portfolio Management*, Fall 1995, pp.113-120.
33. Press W.H. & S.A. Teukolsky & W.H. Vetterling & B.P. Flannery (1992): *Numerical Recipes in C*, Cambridge University Press, Second Edition, 1992, 994 pp.