

Optimization strategies in credit portfolio management

Benjamin Ivorra · Bijan Mohammadi ·
Angel Manuel Ramos

Received: 3 April 2007 / Accepted: 28 June 2007 / Published online: 4 August 2007
© Springer Science+Business Media, LLC 2007

Abstract This paper focuses on the application of an original global optimization algorithm, based on the hybridization between a genetic algorithm and a semi-deterministic algorithm, for the resolution of various constrained optimization problems for realistic credit portfolios. Results are analyzed from a financial point of view in order to confirm their relevance.

Keywords Credit portfolio management · Risk measure · Global optimization · Genetic algorithm · Semi-deterministic algorithm

1 Introduction

Continuous development of derivative credit products [1] makes risk management an important activity in asset allocation of financial structures. Credit risk is the risk of trading partners, called counterparties, not fulfilling their obligations on the due date resulting into losses for investors (this includes bankruptcies such as Enron (2001) and WorldCom (2002) cases [2]). The main objective of credit risk management is to provide models and tools allowing to estimate and eventually reduce amount of losses. One of the most important mathematical contribution in this field was the development of risk measures, such as Value at Risk [3]. But, risk measures are highly non-linear functions. In addition, credit losses are characterized by large probabilities of small earnings together with small probabilities of large losses. This makes difficult the approximation of the loss density function which is

B. Ivorra (✉) · A. M. Ramos
Departamento de Matemática Aplicada, Universidad Complutense de Madrid,
Plaza de Ciencias no 3, Madrid 28040, Spain
e-mail: benjamin.ivorra@mat.ucm.es

A. M. Ramos
e-mail: angel@mat.ucm.es

B. Mohammadi
Mathematics and Modelling Institute, Montpellier University, Montpellier, 34095, France
e-mail: mohamadi@math.univ-montp2.fr

necessary for the evaluation of the risk measure resulting in a non-convex and costly optimization problem. Therefore, one needs efficient global optimization techniques. In literature, many works deal with a convexified version of risk measures, resulting on an over-estimation of the risk [4,5].

In this paper, we focus on the application of a new optimization method for the improvement under constraints of credit portfolio performances, namely non-convex risk measures and income. This method is based on the hybridization between a genetic algorithm [6] and an original semi-deterministic method [7]. The portfolio considered here comes from a real case proposed by the “BNP-Paribas” Portfolio Management Team and belongs to a complex category of credit portfolio called Collateralized Loan Obligations (CLO) [1].

In Sects. 2 and 7, we describe the general CLO structure and present the model used to compute its risk measure and income. Section 3 gives a short introduction to our optimization algorithm. Finally, in Sect. 4 we present the considered optimization problems and analyze the results obtained with our algorithm.

2 Credit portfolio model

We present the general structure of a credit portfolio and explain how to evaluate its performances.

2.1 General structure

We focus on a portfolio (called master portfolio) compound by parts of other portfolios (called inner portfolio).

2.1.1 Inner portfolios: collateralized loan obligations

Collateralized Loan Obligations (CLO) are security interests in pools of assets, that usually comprise loans (also called facilities). The objective, for financial institutions, is to buy securitization to protect themselves from possible defaults of counterparties in the CLO. Investors, such as insurance companies, bear the credit risk (or a part of it) and receive until the CLO maturity date a periodic remuneration increasing with the level of risk.

Multiple slices of securities are issued by the CLO, offering various credit risk characteristics to investors. Slices are ranked, according to their degree of credit risk (the higher risk slice is called junior, the medium risk slices are called mezzanines and the lower risk slice is called senior). More precisely, if there is a default, investors on junior slice first cover the loss. In cases where the loss is superior to the junior slice amount of money, other investors, successively from mezzanines to senior, cover the remaining losses. However, the investor's earning increases with risk exposition.

2.1.2 Master portfolio—collateralized loan obligations²

During last years, the relative stability in corporate credit shifted investor interest further toward structured instruments to enhance yield, CLO² (CLO Square) emerged. The CLO² (or Master CLO) may be constructed on the basis of single slice CLOs. A CLO² simply consists into a repackaging transaction of one or several single slices of CLOs (the Inner CLOs (ICLO)). In addition to these ICLO slices, a set of additional stand-alone assets (here

loans), called Single-Names (SN), can be included in the portfolio. And eventually, the CLO² is divided in slices that will be proposed to investors.

A CLO² has several interesting features: it is a highly diversified portfolio, it proposes two layers of subordination (one at the level of the ICLOs and one at the level of the investment slices in the Master CLO) and it is more resilient to low/medium losses scenarios. On the other hand, CLO² clearly leaves investors exposed to more extreme systemic market conditions, in the sense that they experience no losses up to a point for about 90% of the cases, after which the loss deterioration is fairly fast and highly severe [1].

Analyzing a CLO² is difficult. There is an entire portfolio of credits to analyze combined with the complexity of the slicing. Sophisticated credit portfolio models, such as the one proposed in the next Subsection, should be used.

2.2 CLO² loss evaluation model

The present model aims at evaluating our portfolio and in particular, compute its risk measure and its income.

We consider a portfolio compound by $n_{\text{SICLO}} \in \mathbb{N}$ ICLO slices and $n_{\text{SN}} \in \mathbb{N}$ SNs. This portfolio contains $n \in \mathbb{N}$ different facilities (here loans), included in ICLOs and SNs, denoted by $(\text{Fac}_i)_{i=1, \dots, n}$.

Each facility $\text{Fac}_i, i = 1, \dots, n$, is characterized by its client Cl_i (the counterparty), nominal NO_i (the amount of money), maturity date T_i (the contract end date), spread Sp_i (the rate of interest), geographical zone C_i , industry sector I_i , rating Rat_i (An evaluation of a client’s relative safety from an investment standpoint), loss given default LGD_i (the percentage of money lost in case of default), R-square R_i (Represents the degree of correlation between the value of a client’s assets and the behavior of the global economy). All those informations are furnished by private institutions.

For each ICLO slice $i, i = 1, \dots, n_{\text{SICLO}}$, we know its nominal $\text{NO}_{\text{SICLO}_i}$, amount of subordination $\text{Sub}_{\text{SICLO}_i}$ (the position of the slice in the ICLO i) and rating $\text{Rat}_{\text{SICLO}_i}$.

These inputs are stocked in a set denoted by $\text{PORT}=(x, \text{PORT}_{\text{data}})$, where $x = (\text{NO}_{\text{SICLO}_1}, \dots, \text{NO}_{\text{SICLO}_{n_{\text{SICLO}}}}, \text{NO}_1, \dots, \text{NO}_{n_{\text{SN}}})$ and $\text{PORT}_{\text{data}}$ is the set of other data.

We denote by $\text{L}(\text{PORT})$ the random variable associated to the portfolio loss amount. In order to compute the portfolio risk measure, we need to determine its density function $\beta_{\text{L}(\text{PORT})}$. To do so, we introduce $\beta_{\text{L}(\text{PORT}), \Delta B}$ the discrete version of $\beta_{\text{L}(\text{PORT})}$, where $\Delta B \in \mathbb{N}$ is a discretization step size. $\beta_{\text{L}(\text{PORT}), \Delta B}$ is evaluated using $M \in \mathbb{N}$ iterations of a Monte-Carlo algorithm described in Sect. 7.

Then, using data stocked in PORT and $\beta_{\text{L}(\text{PORT}), \Delta B}$, we are able to compute:

Income The amount of money received by a person or organization because of return on investments. In our case, this is given by:

$$\text{IC}(\text{PORT}) = \sum_{i=1}^n \text{Sp}_i \times \text{NO}_i \tag{1}$$

Risk measure For the chance that return on a given investment is different than expected. This includes possible partial or total loss.

One can give a more formal description of what said above. Let Ω be a finite set of states of nature, \mathfrak{S} a σ -algebra and \mathbb{P} a risk measure. Any element X of the probability space $L^\infty(\Omega, \mathfrak{S}, \mathbb{P})$ is called risk and any mapping $\varpi : L^\infty(\Omega, \mathfrak{S}, \mathbb{P}) \rightarrow \mathbb{R}$ is called risk measure on $L^\infty(\Omega, \mathfrak{S}, \mathbb{P})$. Here, we focus on a particular and popular risk measure: the Value at Risk (VaR).

The VaR_α of a random variable $X \in L^\infty(\Omega, \mathfrak{F}, \mathbb{P})$, is defined as:

$$\text{VaR}_\alpha(X) = \inf \left[Z \in \mathbb{R} \mid \int_0^Z \beta_X(x) dx > (1 - \alpha) \right] \tag{2}$$

where β_X is the loss density function of X and $\alpha > 0$ is a given confidence level (i.e., a percent level).

$\text{VaR}_\alpha(\text{L(PORT)})$ can be interpreted as the nominal value of the smallest loss of the worst α % losses. A complete presentation of risk measures and VaR can be found in [3–5,8].

3 Global optimization methods

We consider the following minimization problem:

$$\min_{x \in \Omega} J(x) \tag{3}$$

where $J : \Omega \rightarrow \mathbb{R}$ is the cost function, x is the optimization parameter belonging to a compact space $\Omega \subset \mathbb{R}^N$, with $N \in \mathbb{N}$. We assume J is bounded on Ω .

In this section, we give a short presentation of an original optimization method used to solve (3). This algorithm is based on an hybridization between a semi-deterministic algorithm and a particular genetic algorithm to which one aims to provide suitable populations for global search.

3.1 Genetic algorithm

Genetic algorithms (GA) approximate the solution of (3) J through a stochastic process based on an analogy with the Darwinian evolution of species [6]: a first family, called ‘population’, $X^0 = \{x_l^0 \in \Omega, l = 1, \dots, N_p\}$ of N_p possible solutions of the optimization problem, called ‘individuals’, is randomly generated in the search space Ω . Starting from this population, we build recursively N_{gen} new populations, called generations, $X^i = \{x_l^i \in \Omega, l = 1, \dots, N_p\}$ with $i = 1, \dots, N_{\text{gen}}$ through three stochastic steps, called selection, crossover and mutation. Below, we present GA through an original matrix-form formulation.

We first rewrite X^i using the following (N_p, N) -real valued matrix form:

$$X^i = \begin{bmatrix} x_1^i(1) & \dots & x_1^i(N) \\ \vdots & \ddots & \vdots \\ x_{N_p}^i(1) & \dots & x_{N_p}^i(N) \end{bmatrix} \tag{4}$$

Selection Each individual, x_l^i is ranked with respect to its cost function value $J(x_l^i)$ (i.e., the lower is its value of $J(x_l^i)$ the higher is the ranking). Then N_p individuals are randomly selected (individuals with better ranking have higher chances to be selected), with eventual repetitions, to become ‘parents’.

Introducing a binary (N_p, N_p) -matrix S^i , generated according to previous ranking and selection processes, with $S_{j,k}^i = 1$ if the k th individual of X^i is the selected ‘parent’ number j and $S_{j,k}^i = 0$ otherwise, we define:

$$X^{i+1/3} = S^i X^i. \tag{5}$$

Crossover This process leads to a data exchange between two ‘parents’ and the apparition of two new individuals called ‘children’. We determine, with a probability p_c , if two consecutive

parents in $X^{i+1/3}$ should exchange data or if they are directly copied into the intermediate population $X^{i+2/3}$.

Introduce a real-valued (N_p, N_p) -matrix C^i where for each couple of consecutive lines $(2j - 1, 2j)$ ($1 \leq j \leq \frac{N_p}{2}$ in case N_p is even or $1 \leq j \leq \frac{N_p-1}{2}$ in case N_p is odd), the coefficients of the $2j - 1$ th and $2j$ th rows are given by:

$$C^i_{2j-1,2j-1} = \lambda_1, \quad C^i_{2j-1,2j} = 1 - \lambda_1, \quad C^i_{2j,2j-1} = \lambda_2, \quad C^i_{2j,2j} = 1 - \lambda_2$$

In this expression:

- $\lambda_1 = \lambda_2 = 1$ if parents are directly copied (with a probability $1 - p_c$).
- λ_1 and λ_2 are randomly chosen in $]0, 1[$ if a data exchange occurs between the two parents (with probability p_c).

Other coefficients of C^i are set to 0. If N_p is odd, the N_p th parent is directly copied, i.e., $C^i_{N_p, N_p} = 1$.

This step can be summarized as:

$$X^{i+2/3} = C^i X^{i+1/3} \tag{6}$$

Mutation This process leads to new parameter values for some individuals of the population. More precisely, each child is modified (or mutated) with a fixed probability p_m .

Introduce for instance a random perturbation matrix \mathcal{E}^i with an i th line equal to:

- A random vector $\epsilon_i \in \mathbb{R}^N$, according to the admissible space Ω , if a mutation is applied to the i th child (with probability p_m).
- 0 if no mutation is applied to the i th child (with probability $1 - p_m$).

This step can then take the following form:

$$X^{i+1} = X^{i+2/3} + \mathcal{E}^i \tag{7}$$

Therefore, the new population can be written as:

$$X^{i+1} = C^i S^i X^i + \mathcal{E}^i \tag{8}$$

With these three basic evolution processes, it is generally observed that the best obtained individual is getting closer after each generation to the optimal solution of the problem [9]. In practice, as final convergence is difficult with GA based algorithms, one should always complete GA iterations by a descent method for better accuracy.

Engineers like GAs because these do not require sensitivity computation, perform global and multi-objective optimization and are easy to parallelize. Their drawbacks remain their computational complexity, possible degeneracy and lack of accuracy already mentioned and cured coupling with descent methods. The hybridization with the semi-deterministic algorithm presented below aims to reduce computational complexity.

3.2 Hybrid optimization algorithm

Consider a core optimization algorithm $A_0(v, P, \epsilon)$ to solve (3) where $v \in V$ is an initial condition (V depends on the initial condition kind), $\epsilon \in \mathbb{R}$ quantifies a given stopping criterion and P is the set of all remaining optimization parameters. We assume the existence of $v_m \in V$ such that $A_0(v_m, P, \epsilon)$ approaches the solution of (3) with a precision $\epsilon \in \mathbb{R}$.

In this case, solving numerically (3) with the considered core optimization algorithm means to solve

$$\begin{cases} \text{Find } v \in V \text{ such that} \\ J(A_0(v, P, \epsilon)) < \epsilon, \end{cases} \tag{9}$$

where P and ϵ are fixed.

In order to solve (9), we propose to use a multi-layer semi-deterministic algorithm (here, we use the simplified notation SDA) based on the secant method coefficients [7]. In the sequel, we present a particular implementation of this algorithm in the case where a genetic algorithm is considered as core optimization algorithm.

The objective of a such hybridization is to reduce the GA computational complexity, by considering small population size N_p and generation number N_{gen} , keeping the efficiency of the method: SDA providing informations on the choice of the initial population and GA performing global optimization with this population. More precisely, we consider the following algorithms, denoted by $(A_i)_{i=1}^I$, corresponding to $I \in \mathbb{N}$ layers of the considered SDA and reading:

Step 1- Input: $X_{0,1}^i, J^1, \dots, J^i, P, \epsilon$

For l going from 1 to J^i

Step 2.1- $o_l = A_{i-1}(X_{0,l}^i, J^1, \dots, J^{i-1}, P, \epsilon)$

Step 2.2- If $\min\{J(o_k), k = 1, \dots, l\} < \epsilon$ **Go to Step 3**

Step 2.3- We construct $X_{0,l+1}^i = \{x_{l+1,j}^i \in \Omega, j = 1, \dots, N_p\}$ as following:

$$\forall j \in [1, N_p], x_{l+1,j}^i = x_l^i - J(o_l) \frac{o_l^j - x_{l,j}^i}{J(o_l) - J(x_{l,j}^i)} \text{ where } x_{l,j}^i \in X_{0,l}^i$$

End of the loop For

Step 3- Output: $A_i(X^i, J^0, \dots, J^i, \epsilon) = \operatorname{argmin}\{J(o^k), k = 1, \dots, i\}$

where $X_{0,1}^i = \{x_{1,j}^i \in \Omega, j = 1, \dots, N_p\} \in \Omega^{N_p}$ is an initial population for A_0 and $(J^k)_{k=1,\dots,I} \in \mathbb{N}^i$ are the iteration numbers of each algorithm A^k .

SDA intends to optimize the initial population of the considered GA. In cases where there is a significant evolution between the initial population and the best element found by GA, SDA generates a new population closer to this best element. In other cases, the secant method used in Step 2.3 allows to create a new population far from the GA solution. GA is then applied starting from this new population and we reiterate this complete process.

A complete description of this method can be found in [7,10,11]. We call this approach *HGSDA* (Hybrid Genetic/Semi-Deterministic Algorithm).

3.3 Parameters in algorithms

In this paper, HGSDA is applied using the two-layer SDA A_2 (i.e. $I = 2$) with $J^1 = J^2 = 5$ and $\epsilon = -\infty$ (i.e., the algorithm runs until the given complexity).

In addition, the GA parameters are turned as follows: the population size is set to $N_p = 10$ and the generation number is set to $N_{gen} = 10$. The selection is a roulette wheel type [6] proportional to the rank of the individual in the population. The crossover is barycentric in each coordinate with a probability of $p_c = 0.55$. The mutation process is non-uniform with a probability of $p_m = 0.55$. A one-elitism principle, that consists in keeping the current best individual in the next generation, has been imposed.

This set of parameters gives a good compromise between computational complexity and result accuracy. It has been applied and compared with a classical genetic algorithm on various benchmark test cases [7] and industrial applications [10–12].

4 Portfolio optimization problems

In this section we are interested in optimizing the allocation (i.e., the nominal of ICLO slices and SNs) of an initial portfolio, respecting given constraints, in order to improve its performance. We consider the two following optimization problems:

- P_1 : Reduce the portfolio risk measure keeping its income higher or equal to the initial value.
- P_2 : Maximize the portfolio income keeping its risk measure lower or equal to the initial value.

The risk measure considered here is the VaR with a confidence level set to $\alpha = 0.1\%$, this level is often used in banking system. It avoids too extreme, and thus non-realistic, risk scenarios.

The initial portfolio has a CLO² structure and is compound by 500 facilities dispatched in $n_{\text{SICLO}} = 40$ ICLO slices and $n_{\text{SN}} = 54$ SNs. The portfolio nominal is close to 2×10^9 Euros (E), its income near to 2×10^7 E and its $\text{VaR}_{0.1\%} = 1.9 \times 10^8$ E. It has been constructed in collaboration with “BNP-Paribas” Portfolio Management Team using recent market data, described in Subsect. 2.2 and stocked in $\text{PORT}_{\text{data}}$, in order to be as realistic as possible (those data can be found in [7]).

4.1 Parameterization

The parameters considered here are the nominal of ICLO slices and SNs compounding the initial portfolio and the nominal of other SNs owned by “BNP-Paribas” Portfolio Management Team that can be added to this portfolio (18 SNs).

Thus the set of parameters is represented by the following real vector:

$$x = (\text{NO}_{\text{SICLO}_1}, \dots, \text{NO}_{\text{SICLO}_{40}}, \text{NO}_{\text{SN}_1}, \dots, \text{NO}_{\text{SN}_{72}}) \tag{10}$$

In order to obtain a realistic optimized portfolio, respecting prescribed investment rules all parameters are subject to the following constraints:

- Avoid too much concentration in one facility: each nominal must be inferior to 10^8 E.
- Avoid small facility investment: if a nominal is lower than 5×10^6 E it is set to 0E.
- Use rating quality: each ICLO slice or SN have a certain rating. If the rating is good, the nominal can be raised or decreased. If the rating is fair, the nominal must be inferior to the initial portfolio value. Furthermore, for some problematic cases (fair rating and liquidity problem), the nominal is kept to the initial value.

Due to those constraints, the total number of parameters is $n_{\text{param}} = 65$. The admissible space is $\Omega = \prod_{i=1}^{n_{\text{param}}} 0 \times [5 \times 10^6, u_i]$, where u_i = initial value, in cases when the facility can only be reduced and $u_i = 10^8$ in other cases.

Constraints presented previously are common to P_1 and P_2 . In Subsect. 4.2, we introduce other constraints, specific to each problem, and integrate them directly in the formulation of the cost function. For other scenarios, with other kind of constraints, a similar approach could be carried out.

4.2 Cost function

Optimization problems P_1 and P_2 are of the form:

$$\min_{x \in \Omega_{\text{cons}}} J(x) \tag{11}$$

where the cost function $J(x)$ corresponds to the desired performance value of the portfolio associated with parameters x in $\Omega_{\text{cons}} = \{x \in \Omega / lc \leq C(x) \leq uc\}$ with C, uc and lc being, respectively, the problem specific constraint function, the upper and lower boundary values of the constraint. More precisely, for each problem:

- P_1 : $J(x) = \text{VaR}_{0,1\%}(\text{L}(x, \text{PORT}_{\text{data}}))$, $C(x) = \text{IC}(x, \text{PORT}_{\text{data}})$, $lc = 2 \times 10^7$ and $uc = +\infty$.
- P_2 : $J(x) = \text{IC}(x, \text{PORT}_{\text{data}})$, $C(x) = \text{VaR}_{0,1\%}(\text{L}(x, \text{PORT}_{\text{data}}))$, $lc = 0$ and $uc = 1.9 \times 10^8$.

where $\text{PORT}_{\text{data}}$ is given, $\text{VaR}_{0.1}(\cdot)$ and $\text{IC}(\cdot)$ are evaluated using the model described in previous Subsect. 2.2.

According to the work performed in [4], we reformulate the optimization problem (11) including the constraints in J using wall functions. To do so, we introduce a new function \tilde{J} in Ω to be minimized

$$\tilde{J}(x) = J(x) + \vartheta(\min(uc - C(x), 0) + \min(C(x) - lc, 0)) \tag{12}$$

where $\vartheta \gg 1$, and we rewrite problem (11) as:

$$\min_{x \in \Omega} \tilde{J}(x) \tag{13}$$

4.3 Results and discussion

The initial and optimized portfolio allocation structures are depicted by Fig. 1. All results presented in this Subsection are reported on Table 1.

The algorithm used to solve P_1 and P_2 is HGSDA, applied with parameters presented in Subsect. 3.3. During this work, we prefer to use a non-gradient based method as sensitivity analysis is difficult to perform. Indeed, gradient directions push the solution out of Ω_{cons} resulting on a slow convergence of the algorithm and gradient approximation is time consuming (because of intermediate Monte Carlo simulations). The use of HGSDA is also justified as we do not need too much precision on the optimized result (due to market evolution it is not possible to strictly respect the optimized allocation structure).

Table 1 Results obtained for problems P_1 and P_2

	Nominal	Nom. SICLO	Nom. SN	IC	$\text{VaR}_{0,1\%}$
Initial	$2. \times 10^9$	7.5×10^8	1.25×10^9	2.1×10^7	1.9×10^8
Sen.	–	–	–	3%	7%
P_1 -opt.	2×10^9	5.5×10^8	1.45×10^9	2.1×10^7	1.3×10^8
Evo.	0%	–26%	+11%	0%	–31%
Sen.	–	–	–	1%	1%
P_2 -opt.	3.2×10^9	7.5×10^8	2.7×10^9	$3. \times 10^8$	1.9×10^8
Evo.	+57%	+6%	+88%	+28%	0%
Sen.	–	–	–	2%	1%

From (Left) to (Right), main portfolio characteristics: nominal, ICLO slice nominal (Nom. SICLO), SN nominal (Nom. SN), Income (IC) and $\text{VaR}_{0,1\%}$. From (Top) to (Bottom), initial and optimized portfolios for P_1 (P_1 -opt.) and P_2 (P_2 -opt.). Evolution (Evo.) between initial and optimized portfolios and a sensitivity analysis (Sen.) are also reported

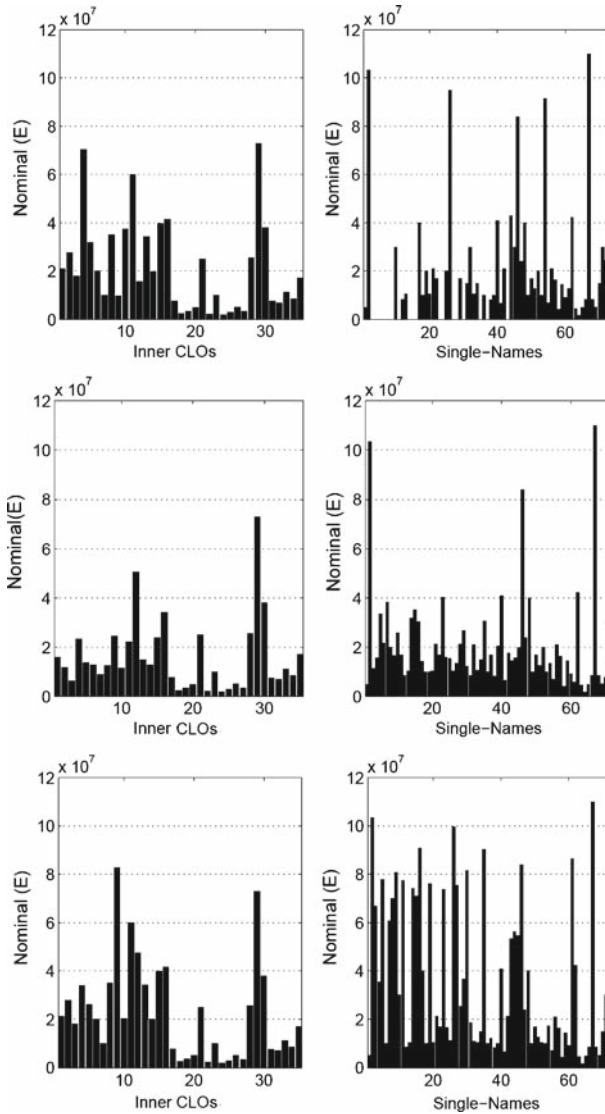


Fig. 1 Portfolios allocation structure: (Top) Initial, (Middle) P_1 -optimized and (Bottom) P_2 -optimized. (Left) nominal of ICLO slices (Right) nominal of SNs

Overall, one optimization process requires approximately 2000 evaluations of \tilde{J} and a computational time of 6 h on a 3 Giga-Hertz PC with 1 Giga-byte of memory. Convergence histories of the best element for P_1 and P_2 are presented in Fig. 2.

Results obtained for P_1 $\text{VaR}_{0,1\%}$ has been reduced by 30% of its initial value. The portfolio income is kept to the initial value. This is foreseeable as the result must be situated on the border of Ω_{cons} . Indeed, a portfolio having an income superior to 2×10^7 E can be improved by projecting it on the border of Ω_{cons} , the risk is then reduced as each nominal is decreased.

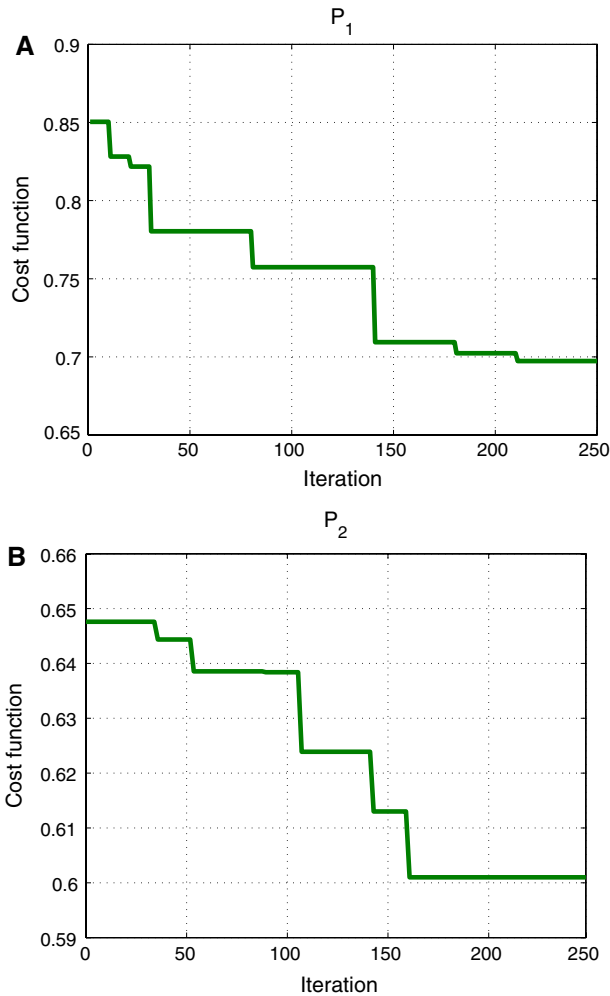


Fig. 2 Convergence histories (normalized cost function value vs. iteration number) of the best element during optimization process for problems (Top) P_1 and (Bottom) P_2

Result obtained by HGSDA suggests to choose a diversified allocation structure, with a high number of different facilities, each one having an average nominal of 2×10^7 E (less for higher risk products).

Due to the high correlation between each ICLO slice (the same product is present in various ICLOs), the total nominal invested on this kind of product is reduced by 26%. In fact, although a simple ICLO is robust to low loss scenarios, combining those ICLOs with a high nominal increase the high loss scenarios probability: if a default occurs in one ICLO other ICLOs have higher chances to be impacted as well.

In comparison, investing on diversified SNs with a reasonable nominal amount decreases the chance to encounter high loss scenarios. Defaults in various sectors and countries should occur in the same scenario to raise a critical loss amount (in fact, due to the diversification, we obtain a portfolio with a majority of low correlation coefficients between facilities). Thus, the total nominal of SNs in optimized portfolio is increased and equitably divided.

Results obtained for P_2 Income has been increased by 28%. As a consequence total portfolio nominal has augmented by 57%, but for the same reasons as previously, the SN nominal proportion has been raised (+88%). In order to improve the portfolio income and control its risk level, essentially SNs combining good spread and good rating have been privileged. Another consequence of the risk constraint is that the optimized portfolio still relatively diversified. As expected, optimized portfolio $\text{VaR}_{0,1\%}$ value is equal to the upper constraint boundary value.

Sensitivity analysis At the end of each optimization process, a sensitivity analysis is performed on the initial and optimized portfolios. To do so, nominals are randomly increased or decreased by 10^7 E. In both cases, the optimized result is more stable than the initial one on the Income and $\text{VaR}_{0,1\%}$ values. This behavior seems to be conformal with the fact that the optimized portfolios are more diversified than the original one and thus the impact of each facility on the portfolio performances is minor. This is an interesting characteristic because, as we said before, it will be difficult to construct the exact optimized portfolio.

5 Conclusion

An original hybrid genetic algorithm has been applied to optimize the performances of a complex and realistic portfolio. Obtained results are satisfactory and in adequacy with financial intuition. Furthermore, even if this intuition can help to generate a general portfolio structure, optimization method has furnished a precise allocation structure. In that sense, optimization algorithms are powerful tools that can help portfolio managers to improve their portfolio characteristics.

Acknowledgements This work was supported by the “Institut de Mathématiques et Modélisation de Montpellier”, “BNP-Paribas” and the “Dirección General de Universidades e Investigación de la Consejería de Educación de la Comunidad de Madrid y de la UCM”, in Spain, through the CCG06-UCM/ESP-1110 project. The authors would like also to thank Guillaume Quibel, Rim Tehraoui, Sebastien Delcourt and Veronique Ormezzano from “BNP-Paribas” for providing their professional point of view and knowledge during this work.

Appendix: algorithm evaluating $\beta_L(\text{PORT})$

The method presented in this section is based on two well known approaches [13–15] often used by financial institutions. We consider a portfolio with the set of inputs PORT described in Subsect. 2.2. In order to compute a discrete version of its loss density function, denoted by $\beta_{L(\text{PORT}),\Delta B}$ where $\Delta B \in \mathbb{N}$ is a discretization step size, we consider the following Monte–Carlo algorithm.

Step 1 Compute the covariance matrix Σ of facilities:

Σ represents the correlation between each facility client present in the portfolio. As it is often used, we decided to apply the Kealhofer, McQuown and Vasicek (KMV) correlation model (a complete description of this model can be found in [15]) in order to specify Σ :

The KMV model is a factor model [14]. It does not model the correlations directly, but in reference to 14 random variables $(E^k)_{1 \leq k \leq 14}$ that model the global economic trends and random variables P_{I_l} and P_{C_m} specific to the activity sector I_l and the geographical area C_m . Those factors E_{k_i} , P_{C_i} and P_{I_i} are supposed to be independent and normally distributed. Thus, the correlation $\text{cor}(i, j)$ between two clients associated to facilities i and j is given by:

$$\text{cor}(i, j) = R_i R_j \left[\sum_{k=1}^{14} \alpha_k^i \alpha_k^j + \beta^{P_{C_i}} \beta^{P_{C_j}} + \beta^{P_{I_i}} \beta^{P_{I_j}} \right] + \sqrt{1 - R_i^2} \sqrt{1 - R_j^2} \delta_{ij} \quad (14)$$

where $\delta_{ij} = 1$ if $i = j$, 0 if not. α_k^i, β^{C_i} and β^{I_i} are real coefficients that models the dependence of each client i to the factors E_i^k, P_{C_i} and P_{I_i} , respectively, and such that $\sum_{k=1}^{14} (\alpha_k^i)^2 + (\beta^{P_{C_i}})^2 + (\beta^{P_{I_i}})^2 = 1$.

For m from 1 to $M \in \mathbb{R}$

Step 2- Generate a default time vector Γ :

The default time τ_i of a particular facility Fac_i is a random variable representing the time from which the associated client Cl_i is in default (the time 0 corresponding to the creation date of the portfolio). The random vector $\Gamma = (\tau_1, \dots, \tau_n)$ is called default times vector. Each particular value of the default times vector Γ corresponds to a possible evolution scenario of the portfolio (See [14] for more details). In practice, τ_i is given by:

$$\tau_i = F_i^{-1}(\Theta(v_i)) \quad (15)$$

where $\Theta(\cdot)$ denotes the standard normal Gaussian density function. v_i is the i^{Th} component of a Gaussian vector $V = (v_1, \dots, v_n)$, with zero mean, covariance matrix Σ (computed in Step 1) and unit variances, given by:

$$V = \Xi G \quad (16)$$

where G is a standard Gaussian vector, Ξ is the Cholesky decomposition of Σ defined as $\Sigma = \Xi' \Xi$ [16]. F_i is the marginal default probability function of τ_i [17] defined by:

$$F_i(t) = \mathbb{P}(\tau_i \leq t). \quad (17)$$

In fact, we are able to approximate easily F_i thanks to the rating Rat_i (the rating gives the internal default probabilities associated with each annual horizon) [14].

Step 3- Compute L_m the loss amount of the scenario m :

We first compute the loss amount $L_{i,m}$ of each facility $\text{Fac}_i, i = 1, \dots, n$, given by:

$$L_{i,m} = (\text{LGD}_i \times \text{NO}_i) \chi_{[0, T_i]}(\tau_i), \quad (18)$$

where $\chi_{[0, T_i]}(\tau_i) = 1$ if $\tau_i \in [0, T_i]$, 0 elsewhere.

Then, we consider two cases: If Fac_i is a SN, we add $L_{i,m}$ to L_m . If Fac_i is included in the ICLO j , we add $L_{i,m}$ to the loss amount $L_{ICLO_j,m}$ of the ICLO j .

For each ICLO slice $j, j = 1, \dots, n_{\text{SICLO}}$, the loss $L_{\text{SICLO}_j,m}$ occurring into our master CLO is given by:

$$L_{\text{SICLO}_j,m} = \max\{\min\{L_{\text{ICLO}_j,m} - \text{Sub}_{\text{SICLO}_j}, \text{NO}_{\text{SICLO}_j}\}, 0\} \quad (19)$$

Thus L_m is given by

$$L_m = \sum_{k=1}^{n_{\text{SICLO}}} L_{\text{SICLO}_k,m} + \sum_{k=1, \text{Fac}_k \text{ is a SN}}^n L_{k,m} \quad (20)$$

Step 4- Complete $\beta_{L(\text{PORT}), \Delta B}$:

This step is done according to ΔB and L_m : We add $1/M$ to the value of the discrete interval of $\beta_{L(\text{PORT}), \Delta B}$ where L_m is included.

End of the loop For

References

1. Bruyere, R.: *Credit Derivatives and Structured Credit*. Wiley Finance (2005)
2. Benston, G., Bromwich, M., Litan, R.E., Wagenhofer, A.: *Following the Money: The Enron Failure and the State of Corporate Disclosure*. AEI-Brookings Joint Center For Regularity Studies, Washington, DC (2003)
3. Rockafellar, R.T., Uryasev, S.: Conditional value-at-risk for general loss distributions. *J. Bank.Finance* **26**, 1443–1471 (2002)
4. Artzner, P., Delbaen, D., Eber, J.M., Heath, D.: Coherent measures of risk. *Math. Finance* **9**, 203–228 (1999)
5. Uryasev, S. and Rockafellar, T: Conditional value-at-risk for general loss distributions *J. Bank. Finance* **26**(7), 1443–1471 (2002)
6. Goldberg, D.: *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison Wesley (1989)
7. Ivorra, B.: *Semi-deterministic global optimization*. PhD. University of Montpellier 2 (2006)
8. Artzner, P., Delbaen, D., Eber, J.M., Heath, D.: Thinking coherently. *RISK* **10**, 68–71 (1997)
9. Dumas, L., Herbert, V., Muyl, F.: Hybrid method for aerodynamic shape optimization in automotive industry. *Comput. Fluids* **33**(5), 849–858 (2004)
10. Ivorra, B., Mohammadi, B., Santiago, D.E., Hertzog, J.G.: Semi-deterministic and genetic algorithms for global optimization of microfluidic protein folding devices. *Int. J. Numer. Meth. Eng.* **66**(2), 319–333 (2006)
11. Ivorra, B., Mohammadi, B., Dumas, L., Durand, O., Redont, P.: Semi-deterministic vs. genetic algorithms for global optimization of multichannel optical filters. *Int. J. Comput. Sci. Eng.* **2**(3), 170–178 (2006)
12. Ivorra, B., Ramos Del Olmo, A.M., Mohammadi, B.: A semi-deterministic global optimization method. Application to a control problem of the burgers equation, comparing with other methods. *J. Optim. Theory Appl.* **135**(1), to be published (2007)
13. Glasserman, P.: *Monte Carlo Methods in Financial Engineering*. Springer-Verlag (2003)
14. Schönbucher, P.J.: *Credit Derivatives Pricing Models*. Wiley Finance (2003)
15. Sellers, M., Davidson, A.: *Modelling Default Risk: Private Firm Model*. KMV Corporation (1998)
16. Nash, J.C.: *Compact Numerical Methods for Computers: Linear Algebra and Function Minimization*, 2nd edn. Adam Hilger, Bristol, England (1980)
17. Nelsen, R.: *An Introduction to Copulas*. Springer-Verlag (1999)