

ROYAUME DU MAROC
*_*_*_*_*_*_*_*_*_*
HAUT COMMISSARIAT AU PLAN
*_*_*_*_*_*_*_*_*_*
INSTITUT NATIONAL
DE STATISTIQUE ET D'ECONOMIE APPLIQUEE



INSEA

Projet de Fin d'Etudes

La modélisation hybride de la CVA par la méthode de Monte-Carlo

Préparé par : Mr BEN-HACHEM Marouane

Sous la direction de : Mr Yassine AL QALLI (INSEA)

Mme Saffae BADRAOUI (SG ATS)

Soutenu publiquement comme exigence partielle en vue de l'obtention du

Diplôme d'Ingénieur d'Etat

Filière : ACTUARIAT-FINANCE

Devant le jury composé de :

- *Mr Yassine AL QALLI (INSEA)*
- *Mr Mohammed EL HAJ TIRARI (INSEA)*
- *Mme Saffae BADRAOUI (SG ATS)*

Juin 2015

Résumé

Depuis la crise de 2008, se couvrir contre le risque de défaut de la contrepartie est devenu un challenge majeur pour les agents du marché financier. Toute contrepartie est " consciente " du risque qu'elle court en concluant un contrat avec une autre contrepartie à cause du risque de défaut de cette dernière. Ce risque qui a été longtemps négligé du fait que les montants engagés dans les portefeuilles étaient modestes et la probabilité de défaut des agents était faible.

Cependant, l'augmentation significative de ce risque avec le temps, a poussé à changer les pratiques dans le marché et à donner plus d'importance à ce risque.

La problématique de mon stage à SG ATS concerne directement ce concept. L'objectif était de définir et d'implémenter une application sous C# qui calcule cet ajustement de valeur qui s'effectue sur un portefeuille pour prendre en considération le risque de contrepartie à savoir la CVA.

Mots clés

- CVA
- Monte Carlo
- Simulation
- EDP
- C#

Dédicace

A mes respectueux parents :

J'espère être à la hauteur de l'image que vous faites de moi.

A Mehdi, Maria, Sabah et toute ma famille en témoignage de mon grand amour et affection.

A mes fidèles amis (es) en témoignage d'une amitié infinie et sérieuse.

Remerciements

Ma reconnaissance et mes remerciements s'adressent à toutes les personnes ayant contribué de près ou de loin à la réalisation et au bon déroulement de ce travail.

Qu'il me soit aussi permis de remercier mon Professeur à l'INSEA *M^r* EL QALLI Yassine pour son aide précieuse, ses conseils et l'intérêt qu'il a accordé à ce travail.

Je ne manquerai également d'exprimer mes remerciements à mes encadrants de stage, *M^{me}* BADRAOUI Safae, analyste quantitatif à Société Générale Africa Technologies & Services (SG ATS), *M^r* PIAU Benjamin, *M^r* JALLOULI Hatem et *M^{me}* BENTATA Amel analystes quantitatifs à Société Générale Corporate & Investment Banking (SG CIB), qui m'ont encadré durant toute cette période en me guidant grâce à leurs orientations et conseils et qui ont été très généreux dans leur partage d'information, et qui, malgré un emploi du temps très chargé, ont toujours été présents pour m'apporter les éclaircissements dont j'avais besoin tout au long de l'avancement dans mon stage.. Je vous remercie des efforts que vous avez déployés.

Mes remerciements s'adressent aussi à *M^{me}* NFISSI Amal, analyste quantitatif à SG ATS, pour sa sympathie, son aide et ses conseils. Je remercie également toute l'équipe SG ATS, d'avoir bien voulu de partager leurs connaissances et leur professionnalisme avec moi.

Je tiens également à remercier *M^r* ELHAJ TIRARI Mohammed d'avoir accepté d'évaluer ce modeste travail. Veuillez trouver ici le témoignage de mon respect le plus profond.

Table des matières

1	Présentation de l'Organisme	11
1.1	Le groupe Société Générale	11
1.2	Société Générale Corporate & Investment Banking	12
1.3	Société Générale Africa Technologies & Services	12
1.3.1	L'équipe R&D	13
2	Introduction au C# et au .NET Framework	14
2.1	Langage C#	14
2.2	Architecture de la plateforme du .NET Framework	15
3	Données de Marché :Concepts fondamentaux	17
3.1	Taux d'intérêt	18
3.1.1	Définitions	18
3.1.1.1	Taux nominal et taux réel	18
3.1.1.2	Le taux actuariel	18
3.1.1.3	Taux zéro-coupon	19
3.1.1.4	Taux moyen pondéré	19
3.1.1.5	Courbe de taux	19
3.1.2	Construction de la Courbe de Taux : Bootstrap	19
3.1.3	Modèle de Vasicek	24
3.1.3.1	Equation de la structure Par terme :	24
3.1.3.2	Modèle	26
3.1.3.3	Prix de zéro-coupon	26
3.1.3.4	Discrétisation du processus	27
3.1.3.5	Estimation des paramètres :	28
3.1.3.6	Application du modèle	31
3.2	Défaut de Crédit	32
3.2.1	Modèle à intensité :	32
3.2.2	Modèle de Black-Karasinski :	32
3.2.3	Simulation de temps de défaut (Méthode intégrale) :	32
3.3	Taux de change : FX	33
3.3.1	Modèle Garman	33
3.4	Simulation	33
3.4.1	Implémentation :	34
3.5	Pricing de Swaption défautable	41
3.5.1	Contrat Swap	41
3.5.2	Swaption	42
3.5.3	Swaption défautable	42
3.5.4	Implémentation :	42

4	Credit Valuation Adjustment (CVA)	45
4.1	Risque de contrepartie et mesures de risque	46
4.2	CVA : définition	46
4.3	Bilateral CVA	47
4.4	Calcul de CVA simple :	48
4.4.1	Simulation de défaut : <i>Méthode intégrale</i>	49
4.4.2	Simulation de défaut : <i>Méthode EDP</i>	50
4.4.2.1	Résolution : Méthode des différences finies :	51
4.5	Application : Calcul de la CVA pour un Swap :	55
4.5.1	Calcul théorique	55
4.5.2	Implémentation	56
	Bibliographie	59
	Webographie	60
	A Présentation de la méthode MonteCarlo :	61
	B Tutoriel Excel-dna :	62

Table des figures

1.1	Société Générale autour du monde	11
1.2	Logo de Société Générale ATS avec slogan	12
1.3	Organigramme de l'équipe ARD	13
2.1	Architecture de la plateforme du .NET Framework	16
3.1	Liaison Excel avec C# par Excel DNA	21
3.2	Méthode Bootstrap sur C#	22
3.3	Macros complémentaires Excel	22
3.4	Méthode Bootstrap sur Excel	23
3.5	Résultat de la méthode de Bootstrap	23
3.6	Test de Dickey-Fuller augmenté sur Eviews	28
3.7	Corrélogramme de la série des TMP	29
3.8	Résultat de la régression du processus AR1 des TMP	29
3.9	Test de Normalité des résidus	30
3.10	Estimation de la prime de risque de Vasicek	31
3.11	Comparaison entre la courbe des taux z_c de marché avec courbe taux z_c de Vasicek	31
3.12	Capture du Source Code MetaNumerics	35
3.13	Reference de la librairie Metanumerics Visual Studio	35
3.14	Classes Modèles	37
3.15	Méthode Next Value	38
3.16	Schéma du Simulateur	39
3.17	Simulateur Taux court	40
3.18	Simulation intensité et de FX	40
3.19	Représentation des flux d'un Swap sur LIBOR 6 mois	41
3.20	Classe Defautable Swaption	43
3.21	Classe Swap	43
3.22	Schéma du Pricer MC	44
3.23	Princing d'une swaption défautable par MC	44
4.1	Interpolation des solutions de l'EDP	55
4.2	Schéma du CVA pricer MC	56
4.3	Classe de pricer CVA	57
4.4	Résultat de calcul CVA pour un swap	57
A.1	Determination de la valeur Pi par MonteCarlo	61
B.1	Aperçue Visual Studio	62
B.2	Solution Explorer-Visual Studio	63
B.3	Exemple pour Excel DNA	63
B.4	Class First ADD IN	63
B.5	Propriétés Class First to Add In	64

B.6	Ajout du fichier Excel	64
B.7	FirstAddIn.dll Propriété	64
B.8	Exemple sur excel appel de C#	65
B.9	Fonction Exemple Excel	65

Liste des tableaux

3.1	méthode itérative de la transformation du taux actuariel en zéro-coupon maturité 2ans	20
3.2	méthode itérative de la transformation du taux actuariel en zéro-coupon maturité 3ans	21
3.3	Matrice de Corrélation	36
3.4	Données pour le modèle de Vasicek taux-court	36
3.5	Données pour le modèle de Black Karasinski de défaut	36
3.6	Données pour le modèle de Garman-Kohlhagen du FX	37
3.7	Présentation de la classe Vasicek	37
3.8	Présentation de la classe Karasinski	38
3.9	Présentation de la classe Garman	38
3.10	Donnée pour la swaption à pricer :	44

Liste des abréviations

- **TMP** : Taux Moyen Pondéré
- **CVA** : Credit Value Adjustment
- **ZC** : Zéro Coupon
- **EDS** : Equation Différentielle Stochastique
- **EDP** : Equation Différentielle Partielle
- **FX** : Forex
- **LGD** : Loss Given Default
- **MtM** : Mark to Market

Introduction

Tout Agent Financier dans le marché est exposé au risque de crédit, le risque que la contrepartie fasse défaut, après avoir été négligé pour longtemps, ce risque est devenu de nos jours beaucoup plus significatif à cause de plusieurs facteurs d'instabilité du marché et des impacts qu'a laissés la crise de 2008.

La CVA (Crédit Value Adjustment), un ajustement qui reflète ce risque et qu'on doit appliquer dans la valorisation d'un produit financier pour le prendre en compte, est le point majeur sur lequel nous nous sommes arrêtés dans le présent mémoire.

Après avoir présenté l'organisme d'accueil et avoir fait une brève introduction de l'outil de programmation utilisé dans le projet à savoir C#, nous allons traiter au niveau du 3eme chapitre les données de marché indispensables au calcul de cet ajustement. Nous allons nous limiter à la modélisation hybride des trois facteurs taux court d'intérêt grâce au modèle de Vasicek, taux de change par un modèle simple de Garman, ainsi que l'intensité par le modèle de Black Karasinski pour la modélisation du risque de défaut de la contrepartie, pour passer ensuite à la valorisation d'un produit defaultable de devise étrangère par une méthode de Monte-Carlo.

Nous allons ensuite entamer la CVA au niveau du 4eme chapitre, sa définition, sa formule de calcul ainsi que ses propriétés, en accordant un intérêt particulier à la simulation des probabilités de défaut avec les deux méthodes qui seront utilisées pour cet intérêt à savoir une méthode directe intégrale et une méthode plus complexe qui repose sur la résolution d'une EDP parabolique des probabilités de défaut, pour enfin procéder à un exemple de calcul de CVA pour un swap de devise étrangère.

Présentation de l'Organisme

Le stage s'est déroulé au sein de la SOCIÉTÉ GÉNÉRALE AFRICA TECHNOLOGIES & SERVICES (SG ATS), qui est une filiale du groupe SOCIÉTÉ GÉNÉRALE (SocGen). Elle regroupe les services Recherche & Développement des activités de marché de la BANQUE DE FINANCEMENT ET D'INVESTISSEMENT (SG CIB). Basée à Casablanca, Société Générale Africa Technologies & Services a ouvert ses portes début 2014.

1.1 Le groupe Société Générale

Depuis sa création en 1864, Société Générale met son expertise au service de ses clients et du financement de l'économie, avec l'ambition de devenir la banque relationnelle de référence. L'engagement, la responsabilité, l'esprit d'équipe et l'innovation sont les valeurs partagées par tous ses collaborateurs.

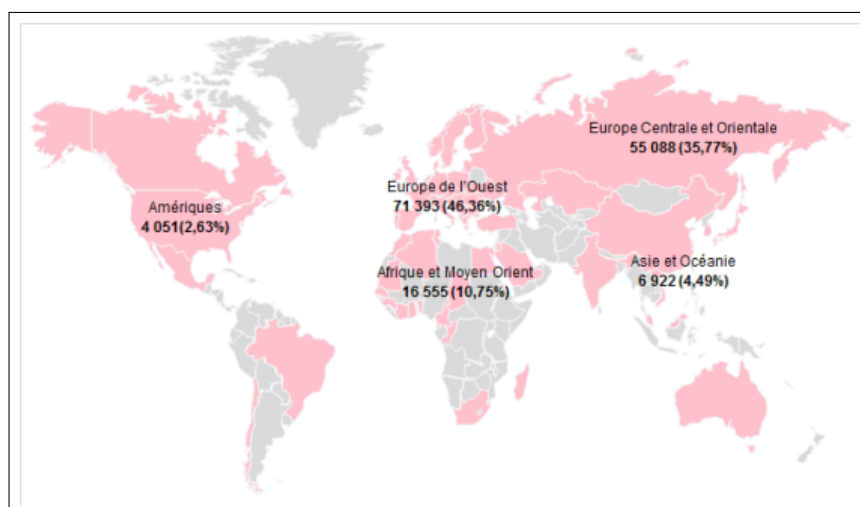


FIGURE 1.1 – Société Générale autour du monde

Société Générale est l'un des tout premiers groupes européens de services financiers. S'appuyant sur un modèle diversifié de banque universelle, le Groupe allie solidité financière et stratégie de croissance durable avec l'ambition d'être la banque relationnelle, référence sur ses marchés, proche de ses clients, choisie pour la qualité et l'engagement de ses équipes.

Acteur de l'économie réelle depuis 150 ans, Société Générale emploie plus de 148 000 collaborateurs, présents dans 76 pays, et accompagne au quotidien 32 millions de clients dans le monde entier en offrant une large palette de conseils et solutions financières sur mesure aux particuliers, entreprises et investisseurs institutionnels, qui s'appuie sur trois pôles métiers complémentaires :

- **La banque de détail en France** avec les enseignes Société Générale, Crédit du Nord et Boursorama qui offrent des gammes complètes de services financiers avec une offre multicanal à la pointe de l'innovation digitale.

- **La banque de détail à l'international, services financiers et assurances** avec des réseaux présents dans les zones géographiques en développement et des métiers spécialisés leaders dans leurs marchés.
- **La banque de financement et d'investissement, banque privée, gestion d'actifs et métier titres** avec leurs expertises reconnues, positions internationales clés et solutions intégrées.

1.2 Société Générale Corporate & Investment Banking

La banque de Financement et d'Investissement (SG CIB) est au coeur du modèle de banque universelle du Groupe Société Générale. Bien diversifiée, elle occupe une position de leader avec près de 12 000 employés présents dans 31 pays à travers l'Europe, l'Amérique et la région Asie Pacifique.

Accompagnant ses clients dans de nombreux secteurs, la banque de Financement et d'Investissement offre des solutions sur mesure en s'appuyant sur son expertise globale en matière de banque d'investissement, de financements et d'activités de marchés.

- Pour les **entreprises, les institutions financières** et le **secteur public** : une approche conseil globale (fusion-acquisitions, dette, actions, capital et gestion actif-passif) ; des solutions de levée de capitaux dette ou actions ; des financements optimisés et des réponses aux besoins de couverture, en particulier dans le domaine du change et des taux.
- Pour les **investisseurs** : des opportunités d'investissement adaptées et des solutions fiables en matière de gestion du risque grâce à une plateforme intégrée offrant un accès global aux marchés (actions, taux, crédit, change, matières premières et dérivés), des services de conseil et d'ingénierie financière, une qualité d'exécution - modèle d'agent et principal - , une recherche sur les différentes classes d'actifs et une offre leader en matière de compensation et de prime brokerage.

1.3 Société Générale Africa Technologies & Services



FIGURE 1.2 – Logo de Société Générale ATS avec slogan

Le groupe Société Générale, à travers sa banque de financement et d'investissement, a créé au Maroc, cette filiale de recherche et développement pour ses activités de marchés, cette filiale ayant obtenu le statut Casablanca Finance City (CFC).

SG ATS, basée à Casablanca et dirigée par Mme. Widad AZZAM LAHLOU, Directeur Général, est constituée d'une équipe d'ingénieurs. Son rôle est de développer pour la banque de financement et d'investissement des logiciels et librairies de formation de prix de produits financiers, ainsi que des outils de gestion des risques, sur l'ensemble des classes d'actifs et des produits de

marché. Cette nouvelle équipe vient compléter le dispositif de recherche et développement, dont l'organigramme ci-dessous, pour les activités de marchés de Société Générale, composé d'équipes basées à Paris, Londres, New-York et Hong-Kong afin de concevoir et développer de nouveaux outils toujours plus performants.

1.3.1 L'équipe R&D

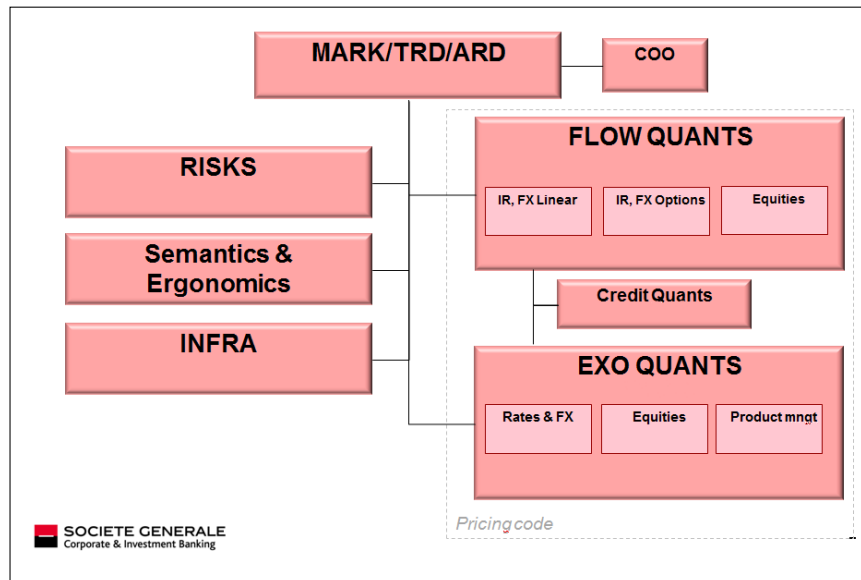


FIGURE 1.3 – Organigramme de l'équipe ARD

Les missions de l'équipe ARD¹ s'articulent autour de deux axes majeurs :

- **développement d'un noyau de calcul de qualité industrielle** : conception, développement, diffusion, maintenance, et amélioration continue.
- **exploitation optimale de ce noyau** : accompagnement des utilisateurs, veille technologique, conseil sur l'architecture du système et la mise en oeuvre effective du noyau.

Cette équipe doit remplir ces missions sous contraintes de temps et de ressources, en conservant toujours une forte adaptabilité :

- **orientation livrable** : elle doit proposer divers scénarios correspondant à des échelles de temps, de sophistication et de consommation de ressources différenciées, afin que le sponsor puisse implémenter sa stratégie globale ; elle doit tenir ses délais.
- **implication** : elle a un rôle de conseil auprès des sponsors sur le choix entre ces divers scénarios ; elle doit avoir un regard critique sur toutes les utilisations effectives de son code quantitatif.
- **flexibilité** : elle doit être en état de supporter des ajustements de priorités relativement fréquents.
- **lisibilité** : elle doit remonter une vision aussi claire que possible des projets et de l'allocation des ressources.
- **innovation** : elle doit proposer et/ou prototyper des améliorations par anticipation des besoins futurs.

1. Applied Research and Development

Introduction au C# et au .NET Framework

C# est un langage orienté objet de type sécurisé et élégant qui permet aux développeurs de générer diverses applications sécurisées et fiables qui s'exécutent sur le .NET Framework. Vous pouvez utiliser le langage C# pour créer entre autres des applications clientes Windows, des services Web XML¹, des composants distribués, des applications client-serveur et des applications de base de données. Visual C# fournit un éditeur de code avancé, des concepteurs d'interfaces utilisateur pratiques, un débogueur intégré et de nombreux autres outils pour faciliter le développement d'applications basées sur le langage C# et .NET Framework.

2.1 Langage C#

La syntaxe C# est très expressive, mais elle est également simple et facile à apprendre. La syntaxe de C# est facile à reconnaître à ses accolades si on connaît déjà les langages C, C++ ou Java. Grâce à la connaissance de l'un de ces langages, on peut devenir très vite productif en C#. La syntaxe C# permet de répondre à de nombreuses complexités de C++ en fournissant des fonctionnalités puissantes telles que des types valeur Nullable, des énumérations, des délégués, des expressions lambda et des accès directs à la mémoire qui n'existent pas en Java. C# prend en charge des méthodes et types génériques qui améliorent la cohérence et les performances des types, ainsi que des itérateurs, qui permettent aux implémenteurs de classes de collection de définir des comportements d'itération personnalisés simples à utiliser par le code client. Les expressions LINQ Language Integrated Query transforment les requêtes fortement typées en construction de langage de premier ordre.

En tant que langage orienté objet, C# prend en charge les concepts d'encapsulation, d'héritage et de polymorphisme. Toutes les variables et méthodes, y compris la méthode Main, point d'entrée de l'application, sont encapsulées dans des définitions de classe. Une classe peut hériter directement d'une classe parente, mais elle peut implémenter un nombre quelconque d'interfaces. Les méthodes qui substituent des méthodes virtuelles dans une classe parente requièrent le mot clé `override`, ce qui évite toute redéfinition accidentelle. En C#, le type `struct` ressemble à une classe légère ; alloué par pile, il peut implémenter des interfaces mais ne prend pas l'héritage en charge.

Outre ces principes orientés objet de base, C# permet de développer facilement des composants logiciel à travers plusieurs constructions de langage innovatrices, y compris les éléments suivants :

- Les signatures de méthodes encapsulées, appelées délégués, qui activent les notifications d'événement de type sécurisé.
- Les propriétés, utilisées comme accesseurs pour les variables membres privés.

1. Extensible Markup Language

- Les attributs, qui fournissent des métadonnées déclaratives à propos des types au moment de l'exécution.
- Les commentaires de la documentation XML inline.
- LINQ (Language Integrated Query) qui fournit des fonctions de requête intégrées à travers diverses sources de données.

En C#, le processus de génération est plus simple qu'en C et C++ et plus souple qu'en Java. Il n'y a pas de fichiers d'en-tête séparés et les méthodes et types peuvent être déclarés dans n'importe quel ordre. Un fichier source C# peut définir tout nombre de classes, structures, interfaces et événements.

2.2 Architecture de la plateforme du .NET Framework

Les programmes en C# s'exécutent sur le .NET Framework, composant intégral de Windows qui inclut un système d'exécution virtuel appelé CLR² et un ensemble unifié de bibliothèques de classes. Le CLR est l'implémentation commerciale de l'infrastructure du langage commun CLI³ de Microsoft, norme internationale constituant la base de toute création d'environnements d'exécution et de développement et assurant le fonctionnement homogène des langages et des bibliothèques.

Le code source écrit en C# est compilé dans un langage intermédiaire conforme à la spécification CLI. Le code IL de ce langage intermédiaire, ainsi que les ressources telles que les bitmaps et les chaînes, sont stockés sur le disque dans un fichier exécutable appelé assembly, dont l'extension est généralement .exe ou .dll. Un assembly contient un manifeste qui fournit des spécifications sur les types, la version, la culture et les conditions de sécurité de l'assembly. À l'exécution du programme C#, l'assembly est chargé dans le CLR, qui peut prendre différentes mesures sur la base des informations du manifeste. Ensuite, si les conditions de sécurité sont respectées, le CLR effectue une compilation juste-à-temps pour convertir le code du langage intermédiaire en instructions machine natives. Le CLR fournit également d'autres services en rapport avec les opérations automatiques de garbage collection et la gestion des exceptions et des ressources. Le code exécuté par le CLR est quelquefois appelé *code géré*, par contraste avec le *code non géré*, compilé dans le langage machine natif ciblant un système spécifique. Le diagramme suivant illustre les relations entre les temps de compilation et d'exécution des fichiers en code source C#, des bibliothèques de classes .NET Framework, des assemblies et du CLR.

2. Common Language Runtime

3. Common Language Infrastructure

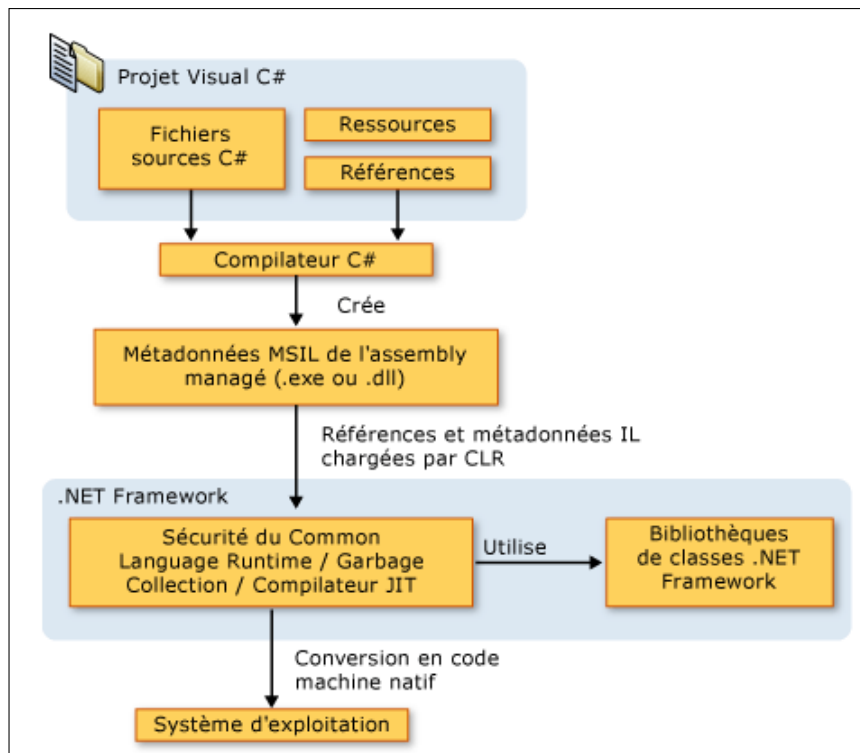


FIGURE 2.1 – Architecture de la plateforme du .NET Framework

L'interopérabilité des langages est une fonctionnalité clé du .NET Framework. Étant donné que le code IL⁴ produit par le compilateur C# respecte la spécification de type commun (CTS), il peut, lorsqu'il est généré à partir de C#, interagir avec le code généré à partir des versions .NET de Visual Basic, Visual C++, Visual J# ou d'un des vingt autres langages respectant la norme CTS. Un même assembly peut contenir plusieurs modules écrits dans différents langages .NET, et les types peuvent se référencer l'un l'autre exactement comme s'ils avaient été écrits dans le même langage. Outre les services d'exécution, le .NET Framework inclut également une bibliothèque étendue de plus de 4 000 classes organisées en espaces de noms, qui fournissent une large gamme de fonctionnalités couvrant de nombreuses utilisations, de l'entrée/sortie de fichiers aux contrôles Windows Forms, en passant par la manipulation de chaînes et l'analyse XML. Une application C# standard utilise largement la bibliothèque de classes .NET Framework pour assurer les tâches de maintenance courantes les plus fastidieuses.

4. Intermediate Language

Données de Marché : Concepts fondamentaux

Introduction

Le calcul de la CVA^a pour un produit financier dans ce travail est basé sur la simulation des scénarios possibles dans le futur aléatoire (Montecarlo) chose qui revient à la simulation des différents taux de marché par des modèles qui requièrent des paramètres et des données extraits des calibrations faites sur le marché en moyennant des modèles mathématiques.

Dans ce présent chapitre, nous nous arrêterons sur ces différents modèles, et nous procéderons par la suite à la valorisation d'un produit dont l'évaluation servira au calcul de la CVA.

a. Crédit Value Adjustment

3.1 Taux d'intérêt

3.1.1 Définitions

3.1.1.1 Taux nominal et taux réel

Prenons l'exemple d'un placement d'une année dans un instrument financier à échéance d'une année au taux nominal R un capital d'une unité monétaire génère, à l'échéance, une somme de $(1+R)$ unités monétaires.

Notons respectivement, P et π le niveau du prix initial et le taux d'inflation durant la durée de portage de l'instrument. Un capital réel de $\frac{1}{P}$ génère donc une valeur réelle de $\frac{1+R}{P(1+\pi)}$. Le taux de rendement réel du r du placement est donc le suivant :

$$\frac{1+R}{1+\pi} - 1$$

3.1.1.2 Le taux actuariel

Le taux de rendement actuariel (d'une obligation par exemple) est le taux qui annule la différence entre le prix d'émission et la valeur actuelle des flux futurs qu'elle génère. Ce taux est calculé au jour du règlement et figure obligatoirement dans les brochures d'émission. Pour l'acheteur de l'obligation, le taux actuariel représente le taux de rentabilité qu'il obtiendrait en gardant l'obligation jusqu'à son remboursement et en réinvestissant les intérêts au même taux actuariel. Le taux de rendement actuariel y est donc la solution de l'équation :

$$P(t, T) = \sum_{i=1}^T \frac{\phi_i}{(1+y(t, T))^i}$$

Avec :

- P : Le prix de l'obligation
- ϕ_i : Le flux reçu en i
- T : la maturité de l'obligation
- $y(t, T)$: le taux de rendement actuariel

Dans ce qui suit nous avons appliqué la formule suivante pour calculer le taux actuariel à partir du TMP¹ des bons de trésors :

$$y = \left[\left(1 + TMP * \frac{n}{360} \right)^{\frac{365}{n}} \right] - 1$$

Où y est le taux actuariel, n la maturité .

1. Taux moyen pondéré du bon

3.1.1.3 Taux zéro-coupon

La construction d'une structure par terme des taux d'intérêt fait le plus souvent référence à la notion de bon zéro-coupon. Un zéro-coupon de maturité T est un contrat qui garantit au détenteur 1 unité de monnaie payée à la date T . Le prix du bon zéro-coupon de maturité T est noté par $P(t, T)$.

Une relation simple lie le prix du bon à son taux noté $R(t, T)$, ce qui permet de travailler indifféremment sur l'un ou l'autre. À un instant donné, la structure par terme des taux d'intérêt se définit comme la valeur prise par $R(t, T)$ pour différentes valeurs de T .

Cependant, pour un prix donné, le taux d'intérêt du bon zéro-coupon est :

$$R(t, T) = -\frac{\log P(t, T)}{T - t}$$

3.1.1.4 Taux moyen pondéré

Le TMP² est un taux au jour le jour du marché monétaire. C'est un taux moyen pondéré par les montants des transactions déclarées, pour un échantillon représentatif d'établissements admis au marché interbancaire. Il est publié par la banque centrale et constitue un des principaux taux de référence du marché monétaire.

3.1.1.5 Courbe de taux

On appelle la structure par terme des taux d'intérêt (ou courbe des taux ou encore gamme des taux) l'ensemble des taux au comptant, un taux constaté immédiatement sur le marché et qui peut donner lieu à transaction (emprunt ou prêt), selon leur maturité. Cette gamme des taux permet de construire une courbe dite courbe des rendements au comptant. La courbe des taux n'est que la représentation graphique des rendements en fonction de la maturité. On distingue les courbes dites positives : les taux sont d'autant plus élevés que l'échéance est lointaine et les courbes négatives ou décroissantes dans le cas inverse.

3.1.2 Construction de la Courbe de Taux : Bootstrap

Étant le seul titre représentatif de sa maturité, l'obligation zéro-coupon présente les meilleures caractéristiques pour construire une structure par terme. Ainsi, dans ce qui suit, nous n'allons prendre en considération que les taux à coupon zéro. Mais on doit d'abord savoir comment passer des obligations à coupons aux obligations zéro-coupon.

Bootstrap³

- Le taux zéro-coupon d'un instrument de maturité un an et distribuant un coupon $C1$ au bout d'un an est égal au taux de ce placement puisque celui-ci est déjà un zéro-coupon. On a donc : $y_1 = C1$

2. taux moyen pondéré

3. Méthode Analytique itérative

- Le taux zéro-coupon d'un instrument de maturité de 2ans et distribuant un coupon $C2$ annuellement se calcule en remplaçant le coupon intermédiaire $C2$ distribué au bout d'un an au taux zéro-coupon 1 an y_1
 Si l'on procède aux deux opérations suivantes : Prêt d'un montant unitaire sur 2 ans (MTS=1)
 Emprunt d'un montant $\frac{C2}{(1+y_1)^2}$ au taux y_1 sur 1 an.
 Le tableau suivant contient le bilan des flux de ces deux opérations :

Date	0	1an	2an
emprunt	$\frac{C2}{(1+y_1)}$	-C2	0
Prêt	-1	C2	1+C2
Total	$\frac{C2}{(1+y_1)} - 1$	0	1+C2

TABLE 3.1 – méthode itérative de la transformation du taux actuariel en zéro-coupon maturité 2ans

Par cette opération on a créé un instrument synthétique ne détachant pas des coupons intermédiaires .le taux de rendement interne de cette opération est donc le même que celui d'un prêt zéro-coupon. On en déduit donc le taux zéro-coupon y_2 de 2ans :

$$1 - \frac{C2}{(1+y_1)} = \frac{1+C2}{(1+y_2)^2}$$

D'où :

$$y_2 = \left[\frac{1+C2}{1 - \frac{C2}{(1+y_1)}} \right]^{\frac{1}{2}} - 1 \quad (3.1)$$

En utilisant les taux zéro-coupon 1 an et 2 an, on peut calculer le taux zéro-coupon 3 ans a partir du taux couponé 3 ans par une stratégie similaire à la précédente :

- Prêt d'un montant unitaire distribuant des coupons $C3$ annuellement et remboursant une unité monétaire au bout de 3 ans
- Emprunt en zéro-coupon de $\frac{C3}{(1+y_2)^2}$ sur 2 ans (cet emprunt zéro-coupon permet de compenser le flux $C3$ reçu l'année 2).
- Emprunt en zéro-coupon de $\frac{C3}{(1+y_1)}$ sur 1 an (cet emprunt zéro-coupon permet de compenser le flux $C3$ reçu à l'année 1).

Le tableau suivant contient le bilan des flux de ces trois opérations :

Date	0	1an	2an	3ans
emprunt	$\frac{C3}{(1+y_1)}$	-C3	0	0
	$\frac{C3}{(1+y_2)^2}$	0	-C3	0
Prêt	-1	C3	C3	1+C3
Total	$\frac{C3}{(1+y_1)} + \frac{C3}{(1+y_2)^2} - 1$	0	0	1+C3

TABLE 3.2 – méthode itérative de la transformation du taux actuariel en zéro -coupon maturité 3ans

On vérifie que la stratégie ainsi réalisée est bien une stratégie de prêt zéro-coupon sur 3 ans y_3 :

$$y_3 = \left[\frac{1 + C3}{1 - \frac{C3}{(1+y_1)} + \frac{C3}{(1+y_2)^2}} \right]^{\frac{1}{3}} - 1 \quad (3.2)$$

On procède ainsi par itération successive pour les années suivantes en utilisant chaque fois les taux zéro-coupon déterminés précédemment. Ce mécanisme peut être résumé par la formule récursive du taux zéro-coupon en fonction des $n-1$ taux zéro-coupon précédents :

$$y_n = \left[\frac{1 + Cn}{1 - Cn \sum_{i=1}^{n-1} \frac{1}{(1+y_i)^i}} \right]^{\frac{1}{n}} - 1 \quad (3.3)$$

Ainsi, nous obtenons une nouvelle base de données qui est la courbe des taux zéro-coupon

Ce calcul de taux zéro-coupon est développé dans C# en utilisant l'ADDIN Excel DNA (voir ANNEXE) pour prendre les inputs depuis excel et donner les résultats sur une autre feuille.



FIGURE 3.1 – Liaison Excel avec C# par Excel DNA

Pour ce faire, nous avons créé une méthode statique Bootstrap dans la classe ZeroCouponCurve (Ci dessous le code sur C#).

```

using System;
using ExcelDna.Integration;

namespace Vasicek
{
    public class ZeroCouponCurve
    {
        [ExcelFunction] public static double[,] Bootstrap(double[,] Tauxcoupon)
        {
            double[,] TauxZeroCoupon = new double[Tauxcoupon.GetLength(0), Tauxcoupon.GetLength(1)];
            for (int i = 0; i < Tauxcoupon.GetLength(0); i++)
            {
                TauxZeroCoupon[i,0] = Tauxcoupon[i,0];

                for (int j = 1; j < Tauxcoupon.GetLength(1); j++)
                {
                    TauxZeroCoupon[i,j] = Math.Pow((1 + Tauxcoupon[i,j]) / (1 - Tauxcoupon[i,j]*Sommev(i,j, TauxZeroCoupon))), (double)1/(j+1))-1;
                }
            }

            return TauxZeroCoupon;
        }

        private static double Sommev(int i, int j, double[,] T)
    }
}
    
```

FIGURE 3.2 – Méthode Bootstrap sur C#

Maintenant, on a plus qu'à ajouter notre projet d'ExcelDNA dans les Macros complémentaires sur Excel :

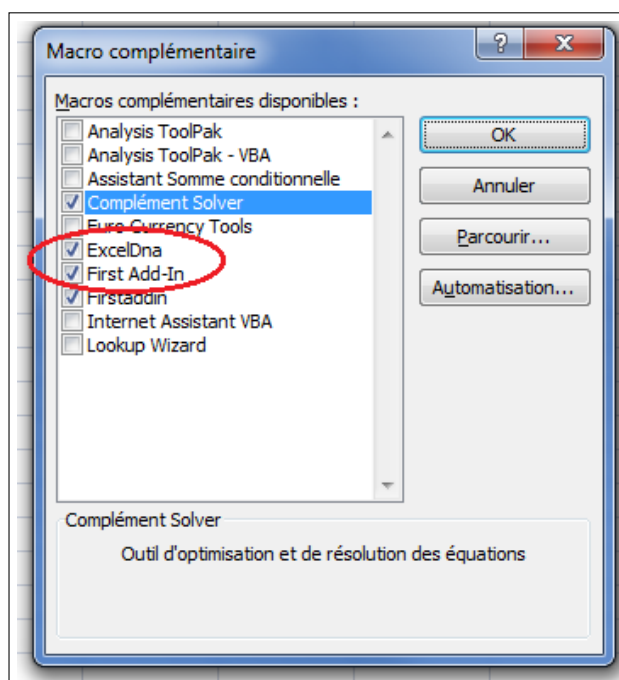


FIGURE 3.3 – Macros complémentaires Excel

A ce stade on peut faire appel à notre méthode "Bootstrap" qu'on trouve parmi les fonctions définies dans Excel :

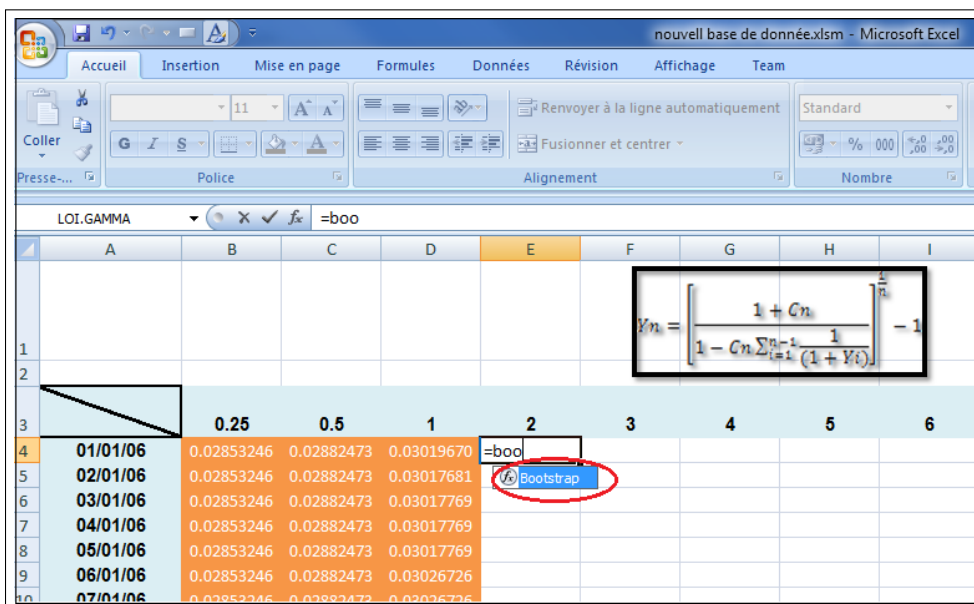


FIGURE 3.4 – Méthode Bootstrap sur Excel

Après avoir exécuté la méthode "Bootstrap" qui prend en paramètre un tableau de données, à savoir les obligations couponnées (annuels) à chaque date pour chaque maturité.

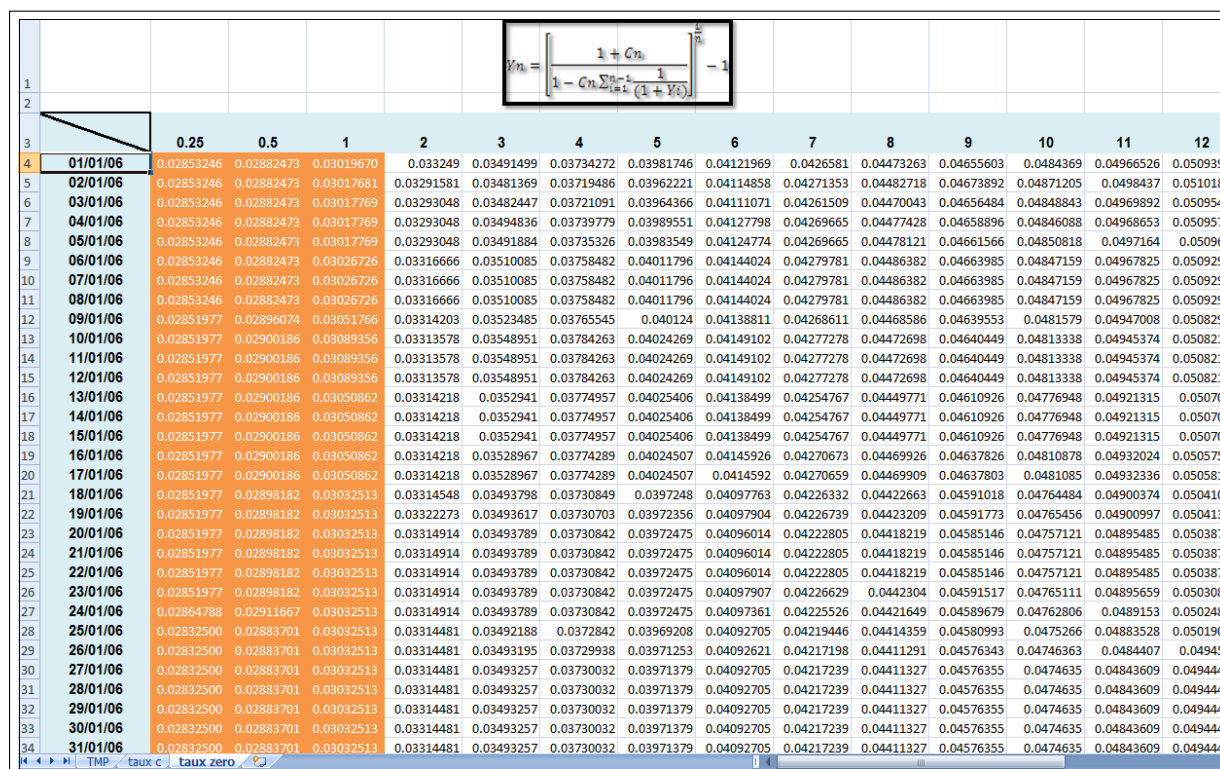


FIGURE 3.5 – Résultat de la méthode de Bootstrap

On obtient une base de taux zéro-coupon de différentes maturités allant de 3mois jusqu'à 20 ans, ces taux ZC qu'on comparera par la suite avec les ZC estimés par le modèle Vasicek.

3.1.3 Modèle de Vasicek

3.1.3.1 Equation de la structure Par terme :

Nous avons repris l'évaluation, à l'aide de modèles d'équilibre inter temporel, des obligations zéro-coupon pour retrouver l'EDP⁴ vérifiée par le prix d'une obligation zéro-coupon à une date t . Supposant une dynamique générale de taux spot sous la probabilité historique \mathbb{Q} de la forme :

$$dr(t) = \mu dt + \sigma dW_t \quad (3.4)$$

Avec W_t un mouvement Brownien standard et μ et σ et successivement la moyenne et la volatilité du taux court.

Soit la dynamique du prix zéro-coupon :

$$\frac{dP(t, T, r)}{P} = \mu_P dt + \sigma_P dW_t \quad (3.5)$$

Proposition 1 On peut écrire :

$$\mu_P = P_t + \mu P_r + \frac{1}{2} \sigma^2 P_{rr} \quad (3.6)$$

$$\sigma_P = \frac{1}{P} \sigma P_r \quad (3.7)$$

Avec :

- $P_t = \frac{\partial P(t, T, r)}{\partial t}$: la dérivée du prix zéro-coupon par rapport au temps
- $P_r = \frac{\partial P(t, T, r)}{\partial r}$: la dérivée du prix zéro-coupon par rapport au taux court
- $P_{rr} = \frac{\partial^2 P(t, T, r)}{\partial r^2}$: la dérivée seconde du prix zéro-coupon par rapport au taux court

Démonstration :

Le prix zéro-coupon à l'instant t est une fonction de t , r et de T , $P(t, T) = P(t, T, r)$. On applique le lemme d'Ito⁵ à $P(t, T) = P(t, T, r)$:

$$dP(t, T, r) = (P_t + \mu P_r + \frac{1}{2} \sigma^2 P_{rr}) dt + \sigma P_r dW \quad (3.8)$$

On peut bien écrire :

$$\frac{dP(t, T, r)}{P} = \frac{1}{P} \left(P_t + \mu P_r + \frac{1}{2} \sigma^2 P_{rr} \right) dt + \frac{1}{P} \sigma P_r dW$$

En utilisant (3.5), on aura directement :

$$\mu_P = P_t + \mu P_r + \frac{1}{2} \sigma^2 P_{rr} \quad (3.9)$$

$$\sigma_P = \frac{1}{P} \sigma P_r \quad (3.10)$$

4. Equation dérivée partielle

5. voir Annexe

Proposition 2 *Le prix zéro-coupon satisfait l'EDP suivante :*

$$P_t + (\mu - \sigma R_m)P_r + \frac{1}{2}\sigma^2 P_{rr} - r(t)P = 0$$

Démonstration :

Soit W le portefeuille obtenu en date t , en vendant n_1 obligation ZC de maturité T1 noté $P_1(\sigma_1; \mu_1)$ et en achetant n_2 obligation ZC⁶ d'échéance T2 noté $P_2(\sigma_2; \mu_2)$:

$$W = n_2 P_2 - n_1 P_1 \quad (3.11)$$

On pose :

$$w_1 = \frac{n_1 P_1}{W} \quad (3.12)$$

Et

$$w_2 = \frac{n_2 P_2}{W} \quad (3.13)$$

On a

$$\frac{dW}{W} = w_2 \frac{dP_2}{P_2} - w_1 \frac{dP_1}{P_1} \quad (3.14)$$

D'où :

$$\frac{dW}{W} = (w_2 \mu_2 - w_1 \mu_1) dt + (w_1 \sigma_1 - w_2 \sigma_2) dW_t \quad (3.15)$$

L'astuce consiste à prendre des proportions bien particulières pour w_1 et w_2 de telle sorte que le terme aléatoire disparaisse. c.à.d :

$$w_1 \sigma_1 - w_2 \sigma_2 = 0 \quad (3.16)$$

Ce choix est :

$$w_1 = \frac{\sigma_2}{\sigma_1 - \sigma_2}$$

Et

$$w_2 = \frac{\sigma_1}{\sigma_1 - \sigma_2}$$

Le portefeuille est maintenant sans risque, L'AOA⁷ dans un marché parfait assure à ce portefeuille de rapporter le taux sans risque r .

On obtient :

$$\frac{dW}{W} = (w_2 \mu_2 - w_1 \mu_1) dt = r(t) dt$$

Puisque $w_2 - w_1 = 1$

On peut écrire :

$$w_2(\mu_2 - r(t)) - w_1(\mu_1 - r(t)) = 0$$

D'où la relation :

6. zéro-coupon

7. Absence d'opportunité d'arbitrage

$$\frac{\mu_2 - r(t)}{\sigma_2} = \frac{\mu_1 - r(t)}{\sigma_1} \quad (3.17)$$

Cette relation étant vérifiée quel que soit le choix des échéances T_i , la valeur $\frac{\mu_P - r(t)}{\sigma_P}$ est indépendante de la maturité T de l'obligation ZC.

Notons (prime de risque) :

$$R_m(t, r) = \frac{\mu(t, r, T) - r(t)}{\sigma}$$

En remplaçant μ_P et σ_P par leur expression en (3.6) et (3.7) on retrouve l'EDP :

$$P_t + (\mu - \sigma R_m)P_r + \frac{1}{2}\sigma^2 P_{rr} - r(t)P = 0 \quad (3.18)$$

3.1.3.2 Modèle

Ce modèle a été proposé par Vasicek(1977). On suppose que sous une proba risque-neutre \mathbb{Q} le taux court instantané r suit un processus d'Ornstein-Uhlenbeck à coefficients constants :

$$dr(t) = \alpha(\beta - r(t))dt + \sigma dW_t \quad (3.19)$$

avec α, β et σ des constantes positives et W un mouvement brownien F_t -adapté. Le terme en dt du processus d'Ornstein-Uhlenbeck s'interprète classiquement comme un retour à la moyenne long-terme β avec une vitesse de retour à la moyenne α . σ est la volatilité du taux court.

Proposition 3 *La solution de l'EDS (3.19) s'écrit :*

$$r(t) = r(s)e^{-\alpha(t-s)} + \beta(1 - e^{-\alpha(t-s)}) + \sigma \int_s^t e^{-\alpha(t-u)} dW_u \quad (3.20)$$

Démonstration : En appliquant le lemme d'Ito à $Y_t = r(t)e^{\alpha t}$

3.1.3.3 Prix de zéro-coupon

Dans le cadre du modèle de Vasicek L'EDP (3.18) du prix zéro-coupon est de la forme :

$$P_t + (\mu - R_m(t))P_r + \frac{1}{2}\sigma^2 P_{rr} - r(t)P = 0 \quad (3.21)$$

On cherche une solution de la forme :

$$P(t, T) = A(t, T) * e^{-B(t, T)r(t)} \quad (3.22)$$

Si on remplace dans l'EDP (3.21) on retrouve les solutions :

$$\left\{ \begin{array}{l} B(t, T) = \frac{1 - e^{-\alpha(T-t)}}{\alpha} \\ A(t, T) = \exp \left[R(\infty)(B(t, T) - (T - t)) - \frac{\sigma^2}{4\alpha} B(t, T)^2 \right] \end{array} \right\}.$$

Avec :

$$R_\infty = \beta + \frac{m}{\alpha} - \frac{\sigma^2}{2\alpha^2} = \lim_{T \rightarrow \infty} R(t, T) \quad (3.23)$$

Pour plus de détail voir [14].

Du prix d'un zéro-coupon on peut extraire directement le taux zéro-coupon. puisque :

$$R(t, T) = \frac{-\ln(P(t, T))}{T - t}$$

Donc :

$$R(t, T) = R_\infty + (r(t) - R_\infty) \frac{1 - e^{-\alpha(T-t)}}{\alpha(T-t)} + \frac{\sigma^2}{4\alpha^3(T-t)} (1 - e^{-\alpha(T-t)})^2 \quad (3.24)$$

3.1.3.4 Discrétisation du processus

D'après la solution (3.20) de l'EDS⁸ de VASICEK, Les propriétés de l'intégrale d'une fonction déterministe par rapport à un mouvement brownien nous conduisent à la discrétisation exacte suivante :

$$r(t) - r(t-1) = \beta(1 - e^{-\alpha}) + (e^{-\alpha} - 1)r(t-1) + \epsilon \quad (3.25)$$

Avec :

$$\epsilon_t \rightarrow N(0, \frac{\sigma^2}{2\alpha}(1 - e^{-2\alpha}))$$

Ce qui montre que nous pouvons estimer les paramètres du processus par la méthode des moindres carrés ordinaires, en utilisant des données discrètes du taux à court terme dans l'équation du processus autorégressif d'ordre (1) suivante :

$$r_t = a + br_{t-1} + \epsilon \quad (3.26)$$

Avec :

- $a = \beta(1 - e^{-\alpha})$
- $b = e^{-\alpha}$
- ϵ : le résidu de l'équation de régression

Dont l'écart type : $\sigma_\epsilon^2 = \frac{\sigma^2}{2\alpha}(1 - e^{-2\alpha})$

Dans la suite du travail nous allons estimer les paramètres $a; b; \sigma_\epsilon$ et on retrouve les paramètres $\alpha; \beta; \sigma$.

- $\alpha = -\ln(b)$
- $\beta = \frac{a}{1-b}$
- $\sigma = \sigma_\epsilon \sqrt{\frac{-\ln(b^2)}{1-b^2}}$

8. Equation différentielle stochastique

3.1.3.5 Estimation des paramètres :

La régression décrite ci-dessus sera effectuée à partir des observations quotidiennes du Taux Moyen Pondéré (TMP) du marché monétaire interbancaire, sur la période allant du 1er janvier 2006 au 08 avril 2012.

- **Test de Stationnarité :**

Tout d'abord il est indispensable de vérifier l'hypothèse de stationnarité de la série des taux courts. Nous appliquons alors le test de Dickey-Fuller augmenté.

L'application de ce test sur Eviews donne le tableau suivant :

Null Hypothesis: TMP has a unit root		
Exogenous: Constant, Linear Trend		
Lag Length: 9 (Automatic - based on SIC, maxlag=26)		
	t-Statistic	Prob.*
Augmented Dickey-Fuller test statistic	-5.530235	0.0000
Test critical values:		
1% level	-3.961955	
5% level	-3.411723	
10% level	-3.127742	
*Mackinnon (1996) one-sided p-values.		

FIGURE 3.6 – Test de Dickey-Fuller augmenté sur Eviews

La série des TMP est donc stationnaire pour un niveau de confiance de 95%, car nous avons : $-5,530235 < -3,411723$.

- **L'ordre du processus autorégressif (1) :**

Soit X un processus autorégressif et b sa fonction d'auto-corrélation partielle. Il est d'ordre p si et seulement si $b(q) = 0$ pour tout $q > p$. voir [10]

Testons : $H_0 : PAC_i = 0$ VS $H_1 : PAC_i \neq 0 \forall i = 1, 2, \dots, 12$

Avec PAC_i : coefficient d'auto corrélation partielle d'ordre i.

Pour un échantillon de grande taille n (notre cas n= 2357) le coefficient PAC_i tend de manière asymptotique vers une loi normale de moyenne 0 et d'écart type $\frac{1}{\sqrt{n}}$.

L'intervalle de confiance de PAC_i correspondant à ce test avec un niveau de confiance de 95% est le suivant :

$$PAC_i \in 0 \pm t_{\frac{\alpha}{2}} \frac{1}{\sqrt{n}} = [-0.04037; 0.04037]$$

La figure suivante représente le corrélogramme de la série des taux courts .

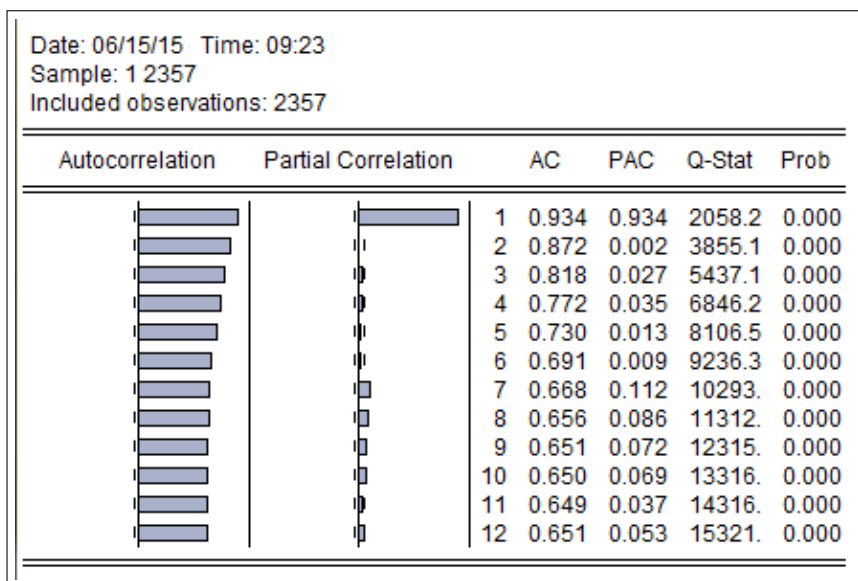


FIGURE 3.7 – Corrélogramme de la série des TMP

Pour un niveau de confiance de 95%, tous les PAC d'ordre 2 et plus rentrent dans l'intervalle de confiance. Donc, nous ne rejetons pas l'hypothèse de la nullité des PAC, et par conséquent, notre processus se rapproche d'un processus auto régressif d'ordre 1. Alors nous retenons le modèle (3.26).

- **Résultat de la régression Eviews :**

Nous avons choisi d'estimer les paramètres a et b dans l'équation de discrétisation par le biais d'une régression linéaire à l'aide du logiciel E-views. Les résultats de l'estimation sont représentés dans le tableau suivant :

Dependent Variable: TMP Method: Least Squares Date: 06/15/15 Time: 09:21 Sample (adjusted): 2 2357 Included observations: 2356 after adjustments				
Variable	Coefficient	Std. Error	t-Statistic	Prob.
C	0.200279	0.023170	8.643723	0.0000
TMP(-1)	0.937056	0.007226	129.6736	0.0000
R-squared	0.877199	Mean dependent var		3.182961
Adjusted R-squared	0.877147	S.D. dependent var		0.386868
S.E. of regression	0.135599	Akaike info criterion		-1.157382
Sum squared resid	43.28315	Schwarz criterion		-1.152489
Log likelihood	1365.396	Hannan-Quinn criter.		-1.155600
F-statistic	16815.24	Durbin-Watson stat		1.947185
Prob(F-statistic)	0.000000			

FIGURE 3.8 – Résultat de la régression du processus AR1 des TMP

Nous avons des statistiques de Student de respectivement 8.64 et 129.67 qui sont largement supérieures à la valeur critique de Student qui est de 1.96 (pour un niveau de confiance de

95%). Donc on rejette l'hypothèse de la nullité des coefficients a et b.

De plus, La statistique de Fisher permet de rejeter également l'hypothèse de la nullité simultanée des deux paramètres a et b.

- **Etude des résidus**

Nous allons étudier les résidus de la régression que nous avons effectuée :

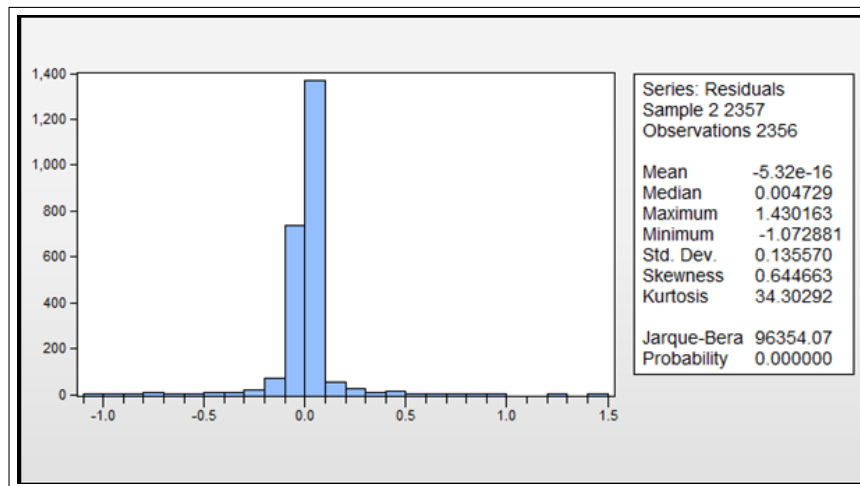


FIGURE 3.9 – Test de Normalité des résidus

Le Skewness étant de 1,43, le Kurtosis est loin de la valeur 3 et la probabilité de Jarques-Berra est très grande. Ainsi, on s'éloigne du cas de la normalité des résidus, et par conséquent, l'hypothèse de normalité est rejetée. Néanmoins, nous avons décidé de poursuivre l'ajustement, tout en étant conscient de cette source d'erreur.

- **Paramètres de Vasicek :**

- $\alpha = -\ln(b) = 0.06501223$

- $\beta = \frac{a}{1-b} = 3.18186007$

- $\sigma = \sigma_\epsilon \sqrt{\frac{-\ln(b^2)}{1-b^2}} = 0.1400299$

Maintenant que les paramètres du processus de diffusion sont estimés, il reste un seul paramètre à estimer afin de déterminer les prix zéro-coupon à savoir la prime de risque R_m mais c'est un paramètre qui n'est pas directement observé sur le marché. Ainsi, différentes méthodes sont employées pour l'estimer. Dans ce travail, ce paramètre sera retrouvé comme la valeur qui minimise l'écart (carré des erreurs) entre les prix zéro-coupons réels (obligation ZC du marché) et les prix théoriques calculés avec Vasicek pour une date donnée.

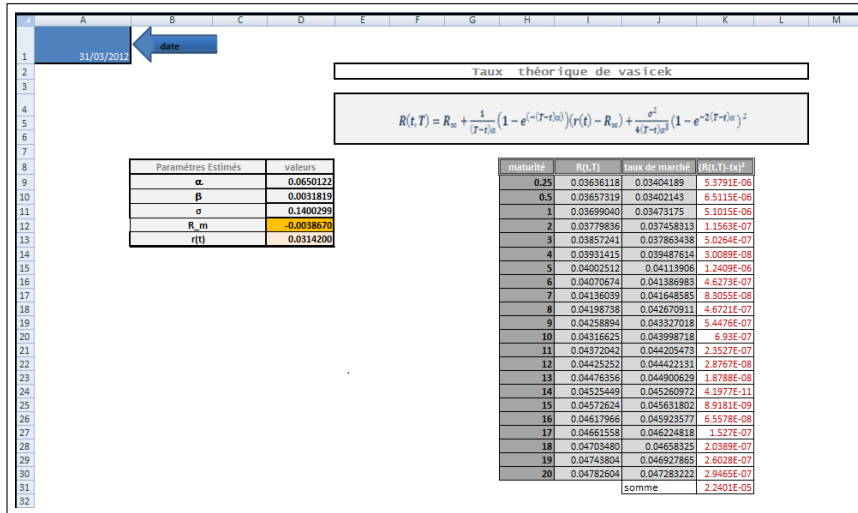


FIGURE 3.10 – Estimation de la prime de risque de Vasicek

3.1.3.6 Application du modèle

Maintenant que tous les paramètres qu'on a dans l'expression du taux zéro-coupon dans le modèle de Vasicek sont estimés, on peut calculer le taux zéro-coupon à une date t de maturité T à partir du taux court à la date t .

Enfin, et afin de valider le modèle, on compare la courbe de zéro-coupon estimée par vasicek avec celle calculée à partir des obligations couponnées de marché (Bootstrap).

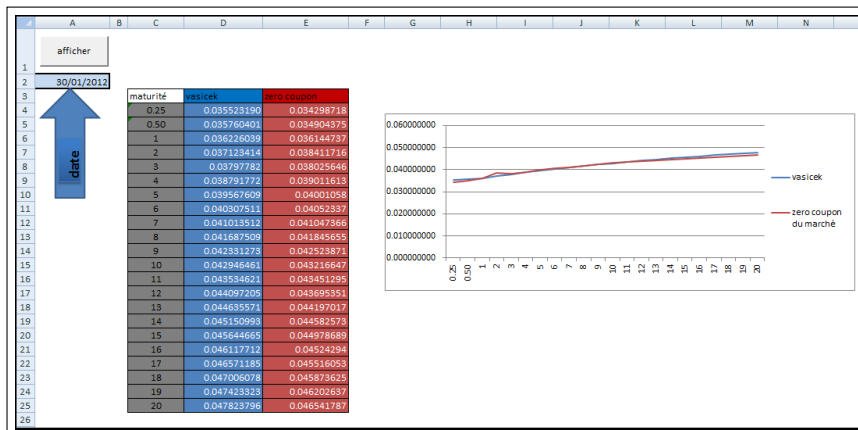


FIGURE 3.11 – Comparaison entre la courbe des taux zc de marché avec courbe taux zc de Vasicek

3.2 Défaut de Crédit

Risque de défaut de la contrepartie : C'est l'incapacité du débiteur à faire face à ses obligations de paiement (paiement des intérêts ou remboursement du capital). Ce risque concerne les activités traditionnelles d'établissement de crédit, mais intervient encore dans le cadre de son activité de marché. Les banques réalisent, en effet, un certain nombre d'opérations sur le marché interbancaire et se trouvent, de ce fait, engagées les unes vis-à-vis des autres, ce qui génère des risques de contrepartie.

3.2.1 Modèle à intensité :

Les temps de défaut sont décrits par des processus de Poisson dont l'intensité, elle-même, est un processus de Black Karasinski. Le calcul de cette probabilité est alors donné pour une discrétisation temporelle $T_1; T_2 \dots T_N$ par :

$$\forall i, \mathbb{Q}(\tau < T_i) = 1 - \exp\left(-\int_0^{T_i} \lambda(t) dt\right) \quad (3.27)$$

Avec τ : l'instant du défaut .

Pour plus de details, voir [4]

3.2.2 Modèle de Black-Karasinski :

Dans le modèle de black-Karasinski de temps de défaut, soit λ_t l'intensité de défaut, on pose $\lambda_t = \exp(Y_t)$. On suppose que sous une proba risque-neutre \mathbb{Q} , Y est un processus d'Ornstein-Uhlenbeck :

$$dY(t) = \alpha_y(\beta_y - Y(t))dt + \sigma_y dW_t \quad (3.28)$$

Avec τ : l'instant de défaut.

Dans ce travail, nous avons pris les résultat d'une calibration des paramètres de ce modèle.

Proposition 4 *La solution de l'EDS s'écrit donc :*

$$Y(t) = Y(s)e^{-\alpha_y(t-s)} + \beta_y(1 - e^{-\alpha_y(t-s)}) + \sigma_y \int_s^t e^{-\alpha_y(t-u)} dW_u \quad (3.29)$$

Donc, cette EDP admet comme solution en forme discrétisée (que nous allons utiliser pour la simulation) :

$$Y(t) = \beta_y(1 - e^{-h\alpha_y}) + e^{-h\alpha_y} Y(t-h) + \sigma \sqrt{\frac{(1 - e^{-2h\alpha_y})}{2\alpha_y}} G^Y \quad (3.30)$$

Avec G^Y est une variable gaussienne standard.

3.2.3 Simulation de temps de défaut (Méthode intégrale) :

Modèle à intensité :

Dans le modèle à intensité, la relation entre probabilité de défaut (ou de survie) et intensité est donnée par :

$$\mathbb{Q}(\tau > t) = \mathbb{E}^{\mathbb{Q}}\left(e^{-\int_0^t \lambda(s) ds}\right) \quad (3.31)$$

Comme le temps de défaut dans le modèle à intensité suit une loi exponentielle, la méthode intégrale consiste à utiliser l'inversion de sa fonction de répartition pour la simulation (l'intensité est supposée déterministe ici) :

Donc le temps de défaut est donné par :

$$\tau = \inf \left\{ u \in \mathbb{R}^+ / \int_0^u \lambda(Y_s) ds > -\ln(U) \right\} \quad (3.32)$$

Nous simulons d'abord une variable uniforme U indépendante. τ est donc T_i si on a :

$$\int_0^{T_i} \lambda(Y_s) ds > -\ln(U)$$

Si un tel T_i n'existe pas, on considère qu'il n'y a pas de défaut avant T .

3.3 Taux de change : FX

3.3.1 Modèle Garman

Le modèle de Garman-Kohlhagen est une adaptation du modèle de Black & Scholes au marché des devises qui a été proposée en 1983.

Ce modèle suppose que les cours de change évoluent selon une distribution log-normale.

$$dS_t = S_t([r_t^d - r_t^e]dt + \sigma^{FX} dW_t^{FX}) \quad (3.33)$$

Avec r_t^d : Taux court de la devise domestique.

r_t^e : Taux de court étranger.

1 unité de devise étrangère = S unités de devise domestique

Donc on a la solution de l'EDS :

$$S_t = S_0 \exp \left(([r_t^d - r_t^e] - \sigma^{FX2}/2)t + \sigma^{FX} W_t^{FX} \right) \quad (3.34)$$

Sous forme discrétisée :

$$S_t = S_0 \exp \left(([r_t^d - r_t^e] - \sigma^{FX2}/2)t + \sigma^{FX} \sqrt{t} G^{FX} \right) \quad (3.35)$$

3.4 Simulation

La simulation des scénarios du Pricer Monte-Carlo repose sur la simulation des trois facteurs de marché Taux intérêt, intensité ainsi que le taux de change.

Sur la même discrétisation temporelle T_1, T_2, \dots, T_N nous souhaitons faire notre simulation.

On suppose de plus que $\forall i, T_{i+1} - T_i = h$

- Le modèle de Vasicek du taux d'intérêt nous donne :

$$r(t) = \beta_r(1 - e^{-h\alpha_r}) + (e^{-h\alpha_r} - 1)r(t-h) + \sigma_r \sqrt{\frac{(1 - e^{-2h\alpha_r})}{2\alpha_r}} G^r \quad (3.36)$$

- Le modèle de Black Karasinski de l'intensité nous donne :

$$Y(t) = \beta_y(1 - e^{-h\alpha_y}) + e^{-h\alpha_y}Y(t-h) + \sigma\sqrt{\frac{(1 - e^{-2h\alpha_y})}{2\alpha_y}}G^Y \quad (3.37)$$

$$\lambda(t) = \exp(Y(t)) \quad (3.38)$$

- Le modèle Garman-Kohlhagen Pour le taux de change nous donne :

$$S_t = S_0 \exp\left(\left([r_t^d - r_t^e] - \sigma^{FX2}/2\right)t + \sigma^{FX}\sqrt{t}G^{FX}\right) \quad (3.39)$$

Le problème de cette simulation étant la corrélation entre ces trois facteurs, une corrélation qui est bien expliquée par une corrélation entre les variables gaussiennes : G^r, G^{FX}, G^Y .

Soit le vecteur Gaussien X :

$$X = \begin{pmatrix} G^r \\ G^{FX} \\ G^Y \end{pmatrix}$$

X est caractérisé par son vecteur espérance mathématique nul et sa matrice de corrélation $Corr_x$.

Maintenant, pour simuler une réalisation de X, on simule U à composantes gaussiennes centrées réduites et indépendantes, on construit cette réalisation en calculant $X = AU$ tel que A est la matrice qui vérifie $A^t A = Corr_x$. (transformé de **Cholesky**)

3.4.1 Implémentation :

En vue de simuler les différents taux de marché, nous devons simuler des gaussiens standards, c'est pour ce faire que nous avons utilisé la librairie Meta Numerics qu'on va présenter brièvement avant de passer à l'implémentation.

Librairie "Meta Numerics" :

Une librairie mathématique et statistique qui supporte les calculs scientifiques en particulier sous C#, elle offre des API (interfaces de programmation) pour l'algèbre des matrices, fonction de répartition et de densité statistique ou toute fonction de nombre réel ou complexe.

Exemple :

Pour générer une réalisation d'une variable gaussienne nous avons utilisé la classe NormalDistribution définie sur Metanumerics :

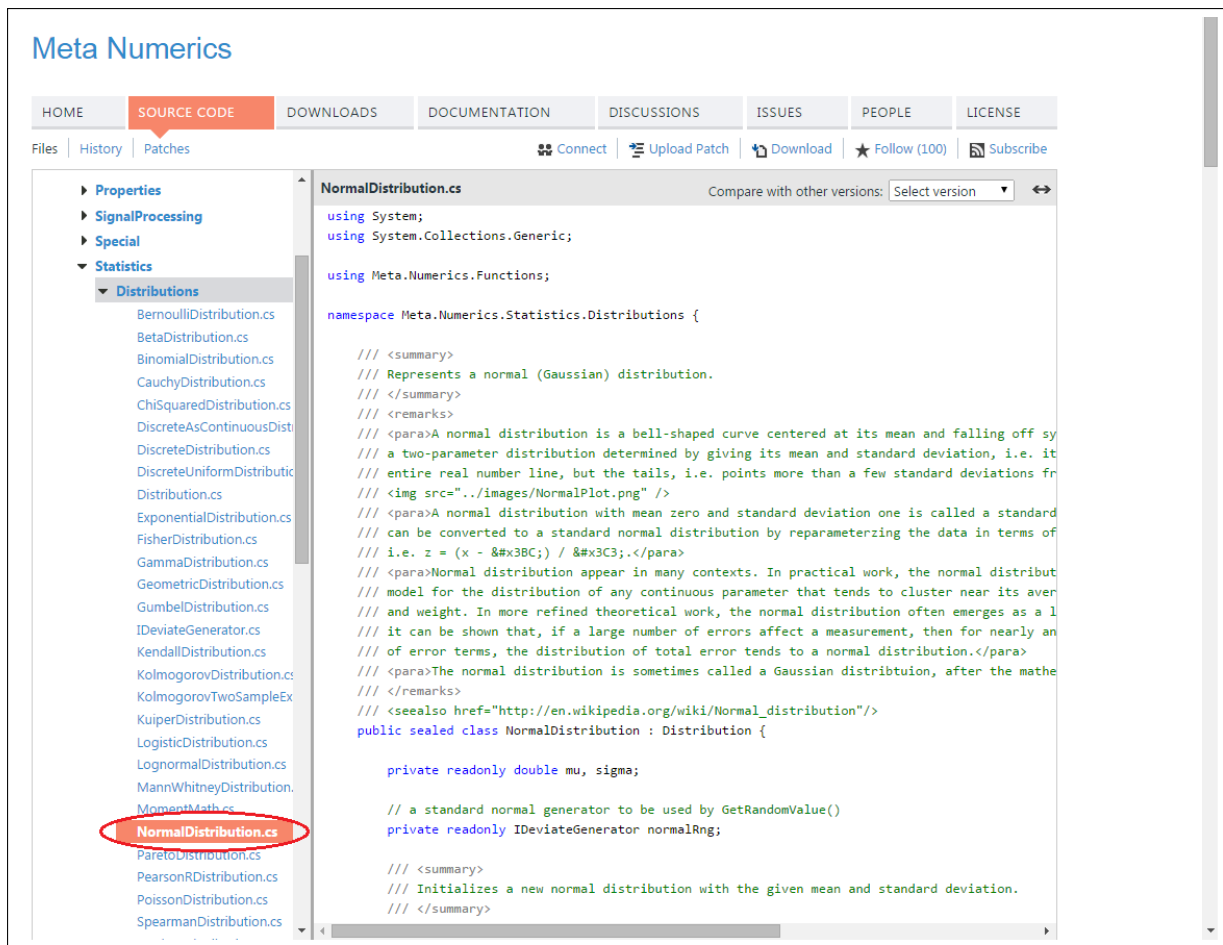


FIGURE 3.12 – Capture du Source Code MetaNumerics

Il suffit d'ajouter la référence MetaNumerics dans le projet sur Visual Studio :

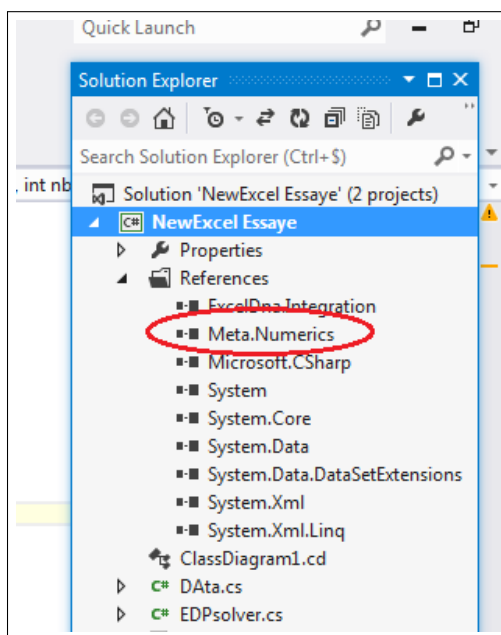


FIGURE 3.13 – Reference de la librairie Metanumerics Visual Studio

Générer un gaussien standard

On génère un nombre aléatoire sur $[0, 1]$ avec par exemple `Random.Next()` .
 Ensuite, on inverse la fonction de répartition de la loi normale prédéfinie sur `Meta Numerics` :

On crée une nouvelle instance de la Classe `NormalDistribution` :

- `NormalDistribution law = new NormalDistribution()`

Et enfin :

- `x= law.InverseLeftProbability(uniformSimulator.Next());`

Présentation des données disponibles :

G^r	G^Y	G^{FX}	
1	0.3	0.4	G^r
0.3	1	0.2	G^r
0.4	0.2	1	G^r

TABLE 3.3 – Matrice de Corrélation

r(0)	0.033
Vol	0.1400299
Mean	3.18186007
Mean-Reversion	0.06501223

TABLE 3.4 – Données pour le modèle de Vasicek taux-court

Y(0)	1.0986
Vol	0.03
Mean	1.097
Mean-Reversion	4

TABLE 3.5 – Données pour le modèle de Black Karasinski de défaut

S(0)	1.3777
r^e	0.0252
Vol	0.01285

TABLE 3.6 – Données pour le modèle de Garman-Kohlhagen du FX

Le simulateur développé sous C# est très générique, dans la suite nous essayerons de présenter l'architecture du code.

Les trois modèles (Intérêt / défaut / change) sont définis dans les classes :

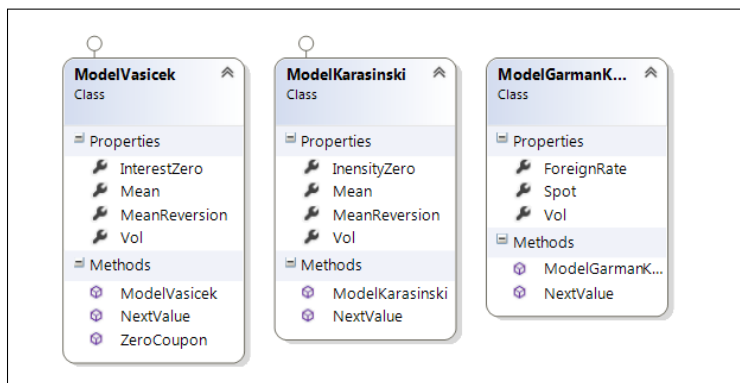


FIGURE 3.14 – Classes Modèles

	Vasicek
Paramètre	<ul style="list-style-type: none"> - interestzero (double) : la valeur du taux court aujourd'hui r_0 - vol (double) : volatilité - double moyenne - meanReversion (double) : le retour à la moyenne
Méthode	<ul style="list-style-type: none"> - NextValue(double PreviousValue, double Pas, double G) : Qui prend la réalisation précédant à t, une réalisation d'une variable gaussienne ainsi qu'un Pas et retourne la réalisation à t+PAs - ZeroCoupon(double Txcourt_t, DateTime t, DateTime T) : Prend t et T ainsi que le taux court à t, et retourne $P(t,T)$

TABLE 3.7 – Présentation de la classe Vasicek

	Karasinski
Paramètre	- intensitzero (double) : la valeur de l'intensité aujourd'hui λ_0 - vol (double) : volatilité - double moyenne - meanReversion (double) : le retour à la moyenne
Méthode	- NextValue (double PreviousValue, double Pas, double G) : Qui prend la réalisation précédant à t , un Pas ainsi qu'une réalisation d'une variable gaussienne et retourne la réalisation suivante à $t + \text{Pas}$

TABLE 3.8 – Présentation de la classe Karasinski

	Garman
Paramètre	- Spot (double) : la valeur du taux de change aujourd'hui FX_0 - vol (double) : volatilité - ForeignRate (double) : taux court de la devise extérieur
Méthode	- NextValue (double PreviousValue, double Pas, double G) : Qui prend la réalisation précédant à t , un Pas ainsi qu'une réalisation d'une variable gaussienne et retourne la réalisation suivante à $t + \text{Pas}$

TABLE 3.9 – Présentation de la classe Garman

Chacune des classes modèles respecte une certaine interface Discretisable, c-à-d qu'elle a une méthode `NextValue` qui prend les paramètres du modèle, une réalisation à t , un Gaussien et un Pas, et retourne une réalisation à $t+h$, par exemple pour le modèle de taux court :

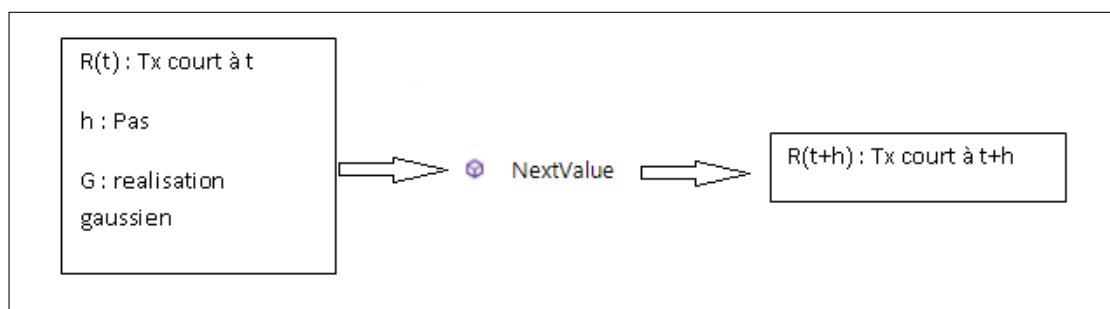


FIGURE 3.15 – Méthode Next Value

Alors, le Simulateur développé prend les trois modèles et simule trois courbe de données Y , r et FX de $t=0$ jusqu'à une date future T .

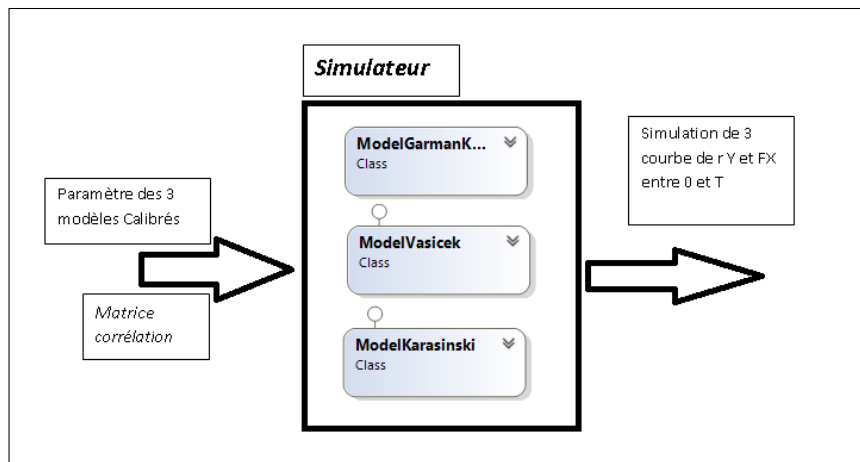


FIGURE 3.16 – Schéma du Simulateur

Algorithm Simulateur :

Algorithm 1 Simulateur

REQUIRE $NbrSimulation > 0$; $Corr$: $double[3,3]$

Result: Simuler Taux court, Intensité et FX

initialization :

$U = double[3]$

$choleskyCorr = double[3,3]$

$G = double[3]$

$tauxcourt = double[NbrSimulation]$

$yIntensite = double[NbrSimulation]$

$FX = double[NbrSimulation]$

$CholeskyCorr \leftarrow TransformeCholesky(Corr)$

for $i \leftarrow 1$ **to** $NbrSimulation$ **do**

 On génère 3 gaussiens indépendante $U[1]; U[2]; U[3]$

$G \leftarrow CholeskyCorr * U$

$tauxcourt[i] \leftarrow ModeleVasicek.NextValue(tauxcourt[i-1], G[1])$

$yIntensite[i] \leftarrow ModeleBK.NextValue(yIntensite[i-1], G[2])$

$FX[i] \leftarrow Garman.NextValue(FX[i-1], G[3])$

end

Résultat : Le simulateur implémenté nous permet de faire une simulation du triplet corrélé (taux court, intensité de défaut, forex). Ci dessous un exemple d'un scénario simulé :

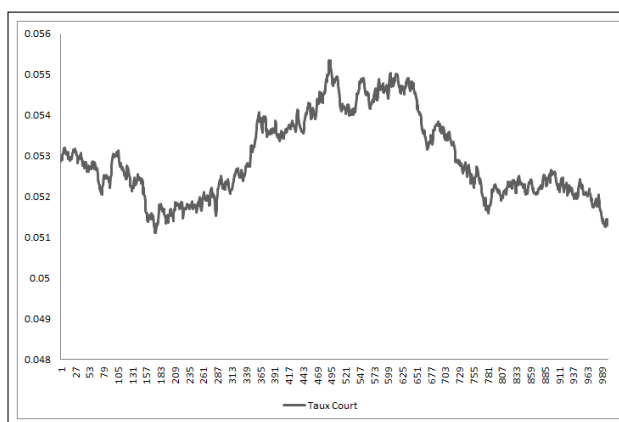


FIGURE 3.17 – Simulateur Taux court

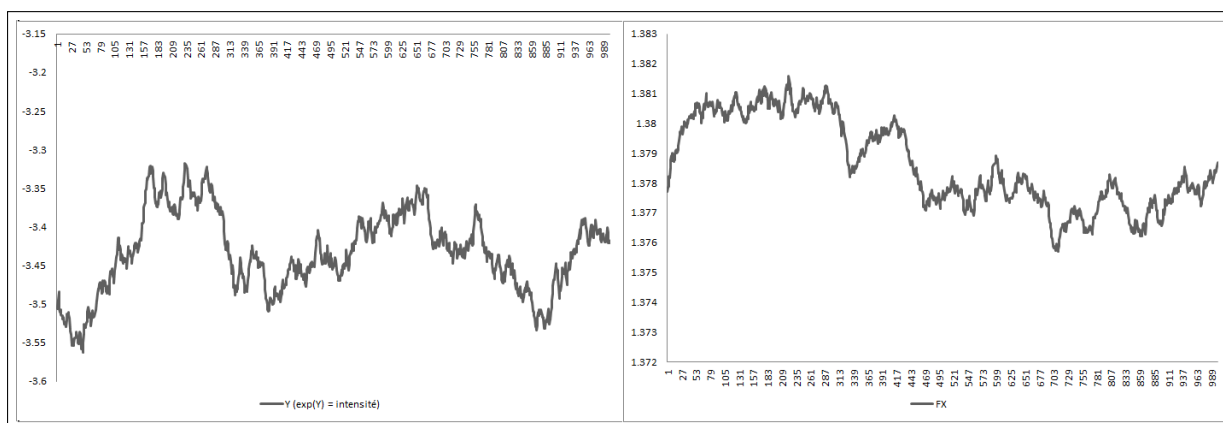


FIGURE 3.18 – Simulation intensité et de FX

Test de la simulation : Afin de tester la simulation, nous avons implémenté des tests qui permettent de s'assurer de la nature des chiffres que donne le code. En effet, ce test est sous forme d'un projet Unit Test⁹ dont on définit des inputs et on fait appel au simulateur. Par exemple pour le taux court, on compare le zéro-coupon calculé comme l'intégrale des taux court jusqu'à la maturité et le ZC calculé à partir de la formule close de Vasicek.

9. voir [web5]

3.5 Pricing de Swaption défectable

3.5.1 Contrat Swap

Soit un calendrier de dates $T_i | i = 0, \dots, N$. On suppose $h = T_i - T_{i-1}$. Le swap de nominal N_m et de Strike K sur la période d'investissement (T_1, T_N) est un contrat échangeant à chaque date $T_i, i = 1, \dots, N$ entre deux contreparties A, B un flux fixe $N_m * K * h$ (jambe fixe) et un flux variable $h N_m L(T_{i-1}, T_i)$ (jambe variable). Le flux fixe est déterminé à l'avance tandis que le flux flottant (variable) est déterminé à chaque date de fixing. Il correspond typiquement au taux LIBOR, EURIBOR ...

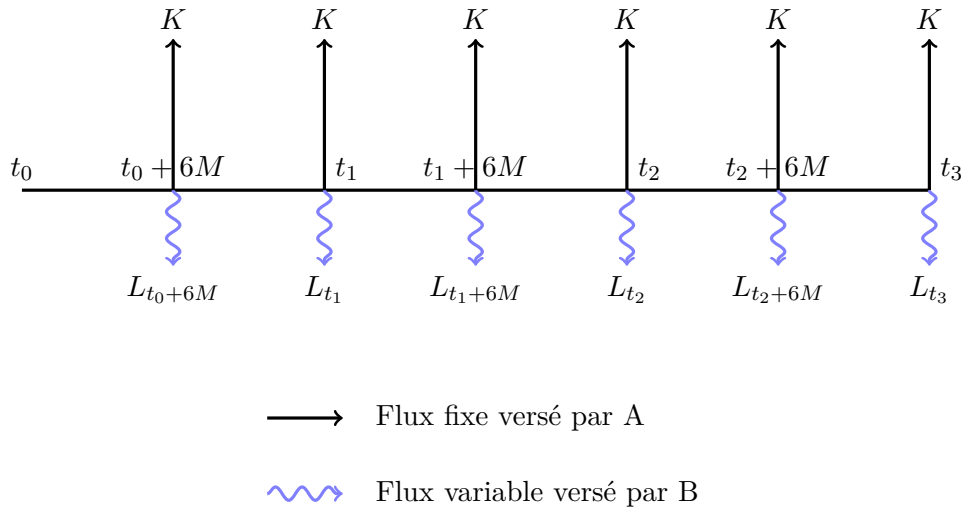


FIGURE 3.19 – Représentation des flux d'un Swap sur LIBOR 6 mois

Le swap est dit "receveur" si l'on reçoit la jambe fixe et on paie la jambe variable, et "payeur" dans le cas contraire. le swap peut donc s'écrire comme une somme de FRA. Ainsi, le prix du swap sur Libor payeur de paramètres (T_0, T_N, N, K) à la date t vaut :

$$\sum_{i=1}^N \text{Prix}^{FRa}(t, T_{i-1}, T_i, N, K) = \sum_{i=1}^N N_m P(t, T_i) (T_i - T_{i-1}) [L(t, T_{i-1}, T_i) - K] \quad (3.40)$$

$L(t, T_1, T_2)$: taux Libor forward expirant à la date T_1 et de maturité T_2 par définition :

$$L(t, T_1, T_2)(T_2 - T_1) + 1 = \frac{P(t, T_1)}{P(t, T_2)} = \frac{1}{P(t, T_1, T_2)} \quad (3.41)$$

On peut donc écrire le prix de swap sous la forme :

$$\text{swap}(t, T_0, T_N) = N_m \sum_{i=1}^N P(t, T_i) h \left[\frac{1}{\frac{P(t, T_{i-1}, T_i)}{h}} - 1 - K \right] \quad (3.42)$$

3.5.2 Swaption

Une swaption européenne est une option permettant d'entrer, à une date T_0 , dans un swap payeur pour la période (T_0, T_N) de nominal N et de Strike K . Le Payoff de la swaption payeuse reçu à la date T_0 :

$$[swap(T_0, T_0, T_N)]^+ = N_m \left(\sum_{i=1}^N P(T_0, T_i) N h [L(T_0, T_{i-1}, T_i) - K] \right)^+ \quad (3.43)$$

Et en fonction des zéro-coupons :

$$[swap(T_0, T_0, T_N)]^+ = \left(N_m \sum_{i=1}^N [P(T_0, T_i) h \left[\frac{\left(\frac{1}{P(T_0, T_{i-1}, T_i)} - 1 \right)}{h} - K \right]] \right)^+ \quad (3.44)$$

Finalement, le prix (en devise d) d'une swaption (en devise e) payeuse à $t=0$:

$$Swaption(0, T_0, T_N) = \mathbb{E} \left[e^{-\int_0^{T_0} r(s) ds} \left(N_m \sum_{i=1}^N [P(T_0, T_i) h \left[\frac{\left(\frac{1}{P(T_0, T_{i-1}, T_i)} - 1 \right)}{h} - K \right]] \right)^+ FX(T_0) \right] \quad (3.45)$$

3.5.3 Swaption défectable

Maintenant, nous voulons calculer le prix d'une swaption défectable c-à-d le prix de la swaption est égal à 0 si la contrepartie fait défaut avant la maturité de la swaption (T_0 dans notre cas). Donc, le nouveau prix de la swaption défectable est :

$$Swaption(0, T_0, T_N) = \mathbb{E} \left[e^{-\int_0^{T_0} r(s) ds} \left(N_m \sum_{i=1}^N [P(T_0, T_i) h \left[\frac{\left(\frac{1}{P(T_0, T_{i-1}, T_i)} - 1 \right)}{h} - K \right]] \right)^+ FX(T_0) 1_{\tau > T_0} \right] \quad (3.46)$$

Avec τ : l'instant de défaut de la contrepartie.

3.5.4 Implémentation :

Le pricer développé sous C# est un outil générique indépendant du type de produit qu'on veut pricer, dans lequel un moteur MonteCarlo fait appel au simulateur qui lui fournit une simulation des variables (taux court/intensité/FX) pour calculer le prix de la swaption défectable pour chaque scénario et retourne la moyenne de ces prix sur le nombre total de tous les scénarios (10000).

Le temps de défaut est donné par :

$$\tau = \inf \left\{ u \in \mathbb{R}^+ / \int_0^u \lambda(Y_s) ds > -\ln(U) \right\} \quad (3.47)$$

La swaption défaultable est définie comme une classe qui hérite d'une classe abstraite Option :

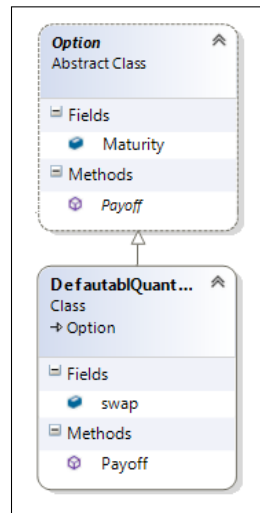


FIGURE 3.20 – Classe Defaultable Swaption

Avec Swap est une Classe qui hérite de la classe Product :

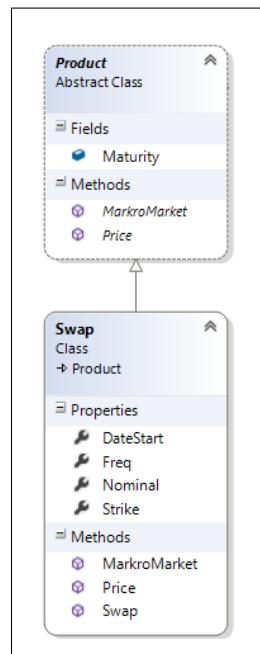


FIGURE 3.21 – Classe Swap

Ci-dessous le résultat de cette application pour des données de swaption :

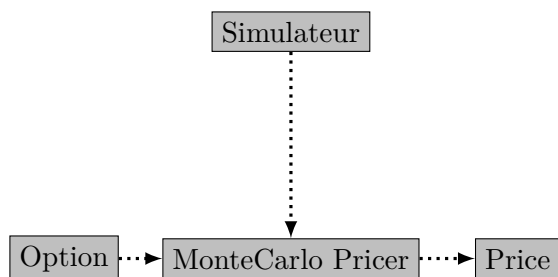


FIGURE 3.22 – Schéma du Pricer MC

Maturité swaption	01/03/2016
Ténon	01/03/2020
Frequence	3 mois
Nominale	10
Strike	0.01

TABLE 3.10 – Donnée pour la swaption à pricer :

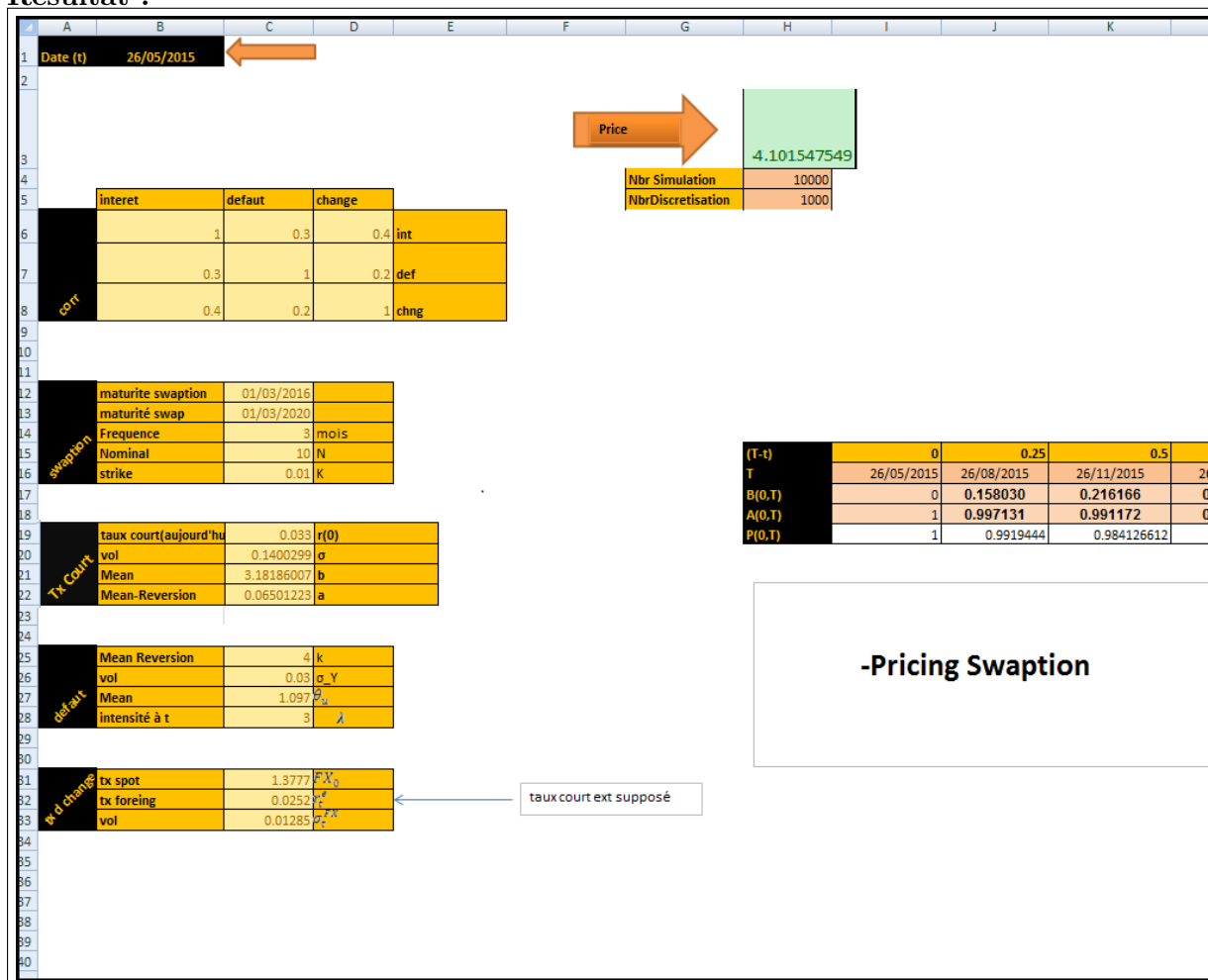
Résultat :


FIGURE 3.23 – Pringing d'une swaption défautable par MC

Credit Valuation Adjustment (CVA)

Introduction

La CVA est l'ajustement qu'il faut faire sur le prix d'un portefeuille avec une contrepartie pour prendre en compte son risque de défaut.

Après avoir défini différents facteurs de marchés ainsi que la méthode utilisée pour les simuler, nous allons définir et mettre en place la méthode Monte-Carlo pour le calcul de la CVA pour un produit financier de devise étrangère, en simulant un grand nombre de MC^a de scénarios futurs possibles pour la perte, avant de finir par un exemple de calcul de CVA pour un swap.

a. Monte-Carlo

4.1 Risque de contrepartie et mesures de risque

C'est le risque couru par une entité qui conclut un contrat avec des contreparties ayant un risque de défaut. On distingue généralement deux types de risque de contrepartie :

- Les mesures pour déterminer les fonds propres réglementaires (Bale II) : Lorsqu'une banque prête de l'argent ou investit dans une entité ayant un risque de défaut, elle fait face à un risque de contrepartie. Elle doit alors fixer son capital afin de prendre en compte les pertes qu'elle pourrait voir survenir lors du défaut des parties avec qui elle traite; ce sont les fonds propres réglementaires.
- le risque de contrepartie dans un point de vue de pricing (le prix ajusté d'un produit en prenant en considération ce risque).

Value at Risk (VaR) : la VaR au niveau de confiance α à l'instant T est la plus petite valeur x telle que la probabilité que la perte LOSS sur la position tenue par la banque dépasse x soit moins de $(1 - \alpha)$:

$$VaR_{\alpha}(T) = \inf\{x \in \mathbb{R} : \mathbb{Q}(LOSS(T) > x) \leq 1 - \alpha\}$$

C'est la pire perte attendue jusqu'à T.

CPE¹ : C'est la différence entre zéro et la valeur du marché du portefeuille de position avec une contrepartie, qui serait perdue si elle faisait défaut à une date future avec un taux de recouvrement nul.

CE² : C'est la valeur courante de la CounterParty Exposure soit la valeur courante du portefeuille si elle est positive ou 0 sinon. C'est typiquement la partie positive de l'espérance des flux futurs sous la probabilité risque-neutre.

PFE³ : C'est l'exposition maximale à une future date atteinte avec un certain niveau de confiance. La courbe de la PFE dans le temps est appelée le profil PFE jusqu'à la maturité finale du portefeuille de positions avec la contrepartie. Cette mesure est calculée à l'aide d'une simulation. Alors que la Credit VaR répond à la question : quelle est la perte finale qui ne soit pas dépassée avec une probabilité donnée sur un horizon donné ?, la PFE répond à la question : si le défaut se produit combien d'argent va-t-on perdre ?

EE⁴ : C'est l'exposition moyenne sous la probabilité historique à une date future.

4.2 CVA : définition

Credit Valuation Adjustment (CVA) est l'ajustement qu'il faut faire sur le prix d'un portefeuille avec une contrepartie pour prendre en considération son risque de défaut. On peut alors la voir comme la différence entre le prix de ce portefeuille " sans-risque " (la contrepartie est supposée sans risque de défaut) et le prix de ce portefeuille calculé avec le risque de défaut de la contrepartie.

Si la contrepartie fait défaut et que le portefeuille, à ce moment, est de valeur positive pour l'autre contrepartie alors elle ne reçoit qu'une fraction (un taux de recouvrement défini) du prix du portefeuille par la partie qui a fait défaut. Ce taux sera noté par la suite R.

Si la valeur du portefeuille noté MtM " Mark to Market ", au moment du défaut est négative pour la contrepartie restante, alors, elle doit payer cette somme entière aux liquidateurs de

1. Counter Party Exposure
 2. Current exposure
 3. Potential Futur Exposure
 4. Expected Exposure

l'entité qui a fait défaut.

C'est cette asymétrie qui fait que l'on peut considérer la CVA comme une option sur la valeur résiduelle d'un portefeuille avec une maturité aléatoire, donnée par le temps de défaut de la contrepartie. Le calcul de la CVA dépend donc du choix du modèle. Etant donné que l'on doit pricer une option, dans le cas d'un swap par exemple, cela nécessite un modèle de taux d'intérêt. Les volatilités des sous-jacents de portefeuille et des spreads de crédit (permettant de quantifier le risque de défaut d'une entité) de la contrepartie ont donc un fort impact sur la valorisation.

4.3 Bilateral CVA

Jusqu'à maintenant, nous nous sommes uniquement placés du point de vue d'une seule contrepartie. La CVA est la valeur positive qu'il faut retrancher du prix d'un deal pour prendre en compte le risque de défaut de sa contrepartie avant la maturité du deal. Dès lors, on peut aussi se placer du point de vue de l'autre contrepartie. Prenons l'exemple d'une opération entre une banque et une entreprise. Si les 2 parties sont d'accord sur le fait que la banque a un risque de défaut nul, alors, elle va faire un ajustement négatif sur le prix sans risque de l'opération avec l'entreprise cliente et l'entreprise va faire un ajustement positif (l'opposé de l'ajustement négatif) sur le prix sans risque. À ce moment, les 2 parties sont d'accord sur le prix. Cet ajustement considéré par l'entreprise cliente est positif et est appelé Débit Valuation Adjustment (DVA). Il est positif car le défaut de l'entreprise amènerait à une dévalorisation de son obligation ce qui signifie un gain dans un certain sens. Dans le cas où la banque est sans risque de défaut, on parle de la CVA et de la DVA unilatéral (UCVA & UDVA) car seul le risque de défaut de l'entreprise est pris en compte. On a alors :

$$UCVA(banque) = UDVA(entreprise)$$

Et

$$UDVA(banque) = UCVA(entreprise)$$

Si l'entreprise n'accepte pas que la banque soit perçue comme sans risque de défaut, alors, la seule façon pour que les 2 parties soient d'accord sur le prix est qu'elles y incluent chacune sa CVA et sa DVA. On a alors :

$$Prix(banque) = Prixsansrisque(banque) + DVA(banque) - CVA(banque)$$

On parle alors de Bilatéral Valuation Adjustment (BVA) ou de CVA bilatérale que l'on note $BVA = DVA - CVA$. On dira que lorsque l'on calcule une CVA bilatérale, on calcule en fait la BVA qui prend en compte le risque de défaut de chacune des deux contreparties. Cependant, il faut faire attention car la BVA n'est pas juste la différence entre la DVA et la CVA, chacune calculée séparément comme si un seul nom peut faire défaut (pour l'exemple du point de vue de la banque; DVA : banque fait défaut et CVA : entreprise fait défaut). En calculant la DVA et la CVA, on doit prendre en compte le défaut des deux entités à la fois; il y a donc une notion de premier à faire défaut. Si on fait les calculs du point de vue de la banque, si la banque fait défaut en premier, alors la DVA est activée tandis que la CVA disparaît, et réciproquement si l'entreprise fait défaut en premier. On doit donc prendre en compte qui fait défaut en premier. L'erreur faite en négligeant cet aspect peut être assez conséquente.

Les méthodologies de calcul de la CVA sont symétriques à celles de la DVA, par conséquent, nous nous limiterons dans le présent rapport au le calcul de la CVA.

4.4 Calcul de CVA simple :

La perte subie à cause d'un événement de défaut à T d'une contrepartie de taux de recouvrement R pour un portefeuille de valeur MtM_t est

$$(1 - R)MtM_t$$

Donc la perte financière au moment du défaut τ :

$$LGD.MtM_{\tau}^{+}.1_{\tau \leq T}$$

Telle que : LGD : Loss Given Default = 1-R.

$1_{\tau \leq T}$: est égale à 1 s'il y a défaut avant T et zéro sinon.

$MtM_{\tau}^{+} = \max\{0, MtM_{\tau}\}$: La partie positive de MtM⁵ à $t=\tau$.

T : La maturité du contrat.

L'idée est d'actualiser tous les flux futurs envoyés d'une contrepartie à une autre en faisant intervenir la probabilité de défaut de la première. Alors, si on se place sous la probabilité risque neutre \mathbb{Q} (un actif échangeable actualisé est une martingale).

Donc :

$$CVA = \mathbb{E}_t^{\mathbb{Q}}[DF_{\tau}.LGD.MtM_{\tau}^{+}.1_{\tau \leq T}] \quad (4.1)$$

$DF_{\tau} = \exp^{-\int_0^{\tau} r(t)dt}$ où $r(t)$ est le taux sans risque.

Si on a MtM_{τ}^{+} en devise étrangère, on introduit une composante de change FX en τ :

$$CVA = \mathbb{E}_t^{\mathbb{Q}}[DF_{\tau}.LGD.MtM_{\tau}^{+}.FX(\tau).1_{\tau \leq T}] \quad (4.2)$$

La valorisation de la CVA est, donc, équivalente à la détermination de la loi jointe de plusieurs variables aléatoires.

Sous la forme d'intégrale on obtient :

$$CVA = \mathbb{E}_t^{\mathbb{Q}}\left[\int_0^T DF_{\tau}.LGD.MtM_{\tau}^{+}.FX(\tau).dH_t\right] \quad (4.3)$$

Avec $H_t = .1_{\tau \leq T}$ le processus de défaut.

On suppose qu'une discrétisation des dates des paiements et des temps de défaut possibles de la contrepartie est choisie à l'avance telle qu'ils peuvent uniquement se produire aux dates T_1, \dots, T_N . En pratique, si la contrepartie fait défaut à une date T_i , cela signifie que le défaut a réellement eu lieu entre T_i et T_{i+1} .

5. Mark to Market : valeur du portefeuille

Sous la forme discrétisée on obtient :

$$CVA = LGD. \sum_{i=1}^{N-1} \mathbb{E}_t^{\mathbb{Q}} \left[\int_{T_i}^{T_{i+1}} DF_{\tau} . MtM_{\tau}^{+} . FX(\tau) . dH_t \right] \quad (4.4)$$

$$CVA = LGD. \sum_{i=1}^{N-1} \mathbb{E}_t^{\mathbb{Q}} \left[DF_{T_i} . MtM_{T_i}^{+} . FX(T_i) . (H_{T_{i+1}} - H_{T_i}) \right] \quad (4.5)$$

$$CVA = LGD. \sum_{i=1}^{N-1} \mathbb{E}_t^{\mathbb{Q}} \left[DF_{T_i} . MtM_{T_i}^{+} . FX(T_i) . 1_{T_i < \tau < T_{i+1}} \right] \quad (4.6)$$

$$CVA = LGD. \sum_{i=1}^{N-1} \mathbb{E}_t^{\mathbb{Q}} \left[DF_{T_i} . MtM_{T_i}^{+} . FX(T_i) . \mathbb{Q}(T_i < \tau < T_{i+1} / F_{T_i}) \right] \quad (4.7)$$

F_t représente l'information sur les facteurs de marché à l'instant t .

Procédure de Monte-Carlo :

- Nous simulons N^{mc} scénarios de MonteCarlo⁶ et la formule de la CVA devient pour (4.1) :

$$CVA = \frac{1}{N^{mc}} \sum_{w=1}^{N^{mc}} DF_{\tau_w} . LGD . MtM_{\tau_w}^{+} . 1_{\tau_w \leq T} \quad (4.8)$$

Donc, dans ce cas, on procède en simulant l'instant de défaut.

- Et sous la forme dicrétisée (4.7) :

$$CVA = \frac{1}{N^{mc}} \sum_{w=1}^{N^{mc}} LGD. \sum_{i=1}^{N-1} DF_{T_i} . MtM_{T_i}^{+} . FX(T_i) . \mathbb{Q}^w(T_i < \tau < T_{i+1} / F_{T_i}) \quad (4.9)$$

Par contre, pour cette formule, on simule des Probas de défaut dans la discrétisation T_1, T_2, \dots, T_{N-1} .

4.4.1 Simulation de défaut : *Méthode intégrale*

Modèle à intensité :

Dans le modèle à intensité, la relation entre probabilité de défaut (ou de survie) et intensité est donnée par :

$$\mathbb{Q}(\tau > t) = \mathbb{E}^{\mathbb{Q}} \left(e^{-\int_0^t \lambda(s) ds} \right) \quad (4.10)$$

Donc, on peut simuler la probabilité de faire défaut entre T_i et T_{i+1} par ce calcul simple . Comme le temps de défaut dans le modèle à intensité suit une loi exponentielle , nous la simulons en utilisant l'inversion de sa fonction de répartition :

Donc, le temps de défaut est donné par :

$$\tau = \inf \left\{ u \in \mathbb{R}^+ / \int_0^u \lambda(Y_s) ds > -\ln(U) \right\}$$

Nous simulons, d'abord une variable uniforme U indépendante. τ est donc T_i si on a :

$$\int_0^{T_i} \lambda(Y_s) ds > -\ln(U)$$

Si un tel T_i n'existe pas, on considère qu'il n'y a pas défaut avant T .

6. voir Annexe

4.4.2 Simulation de défaut : Méthode EDP

$$CVA = LGD. \sum_{i=1}^{N-1} \mathbb{E}_t^{\mathbb{Q}} \left[DF_{T_i} \cdot MtM_{T_i}^+ \cdot FX(T_i) \cdot \mathbb{Q}(T_i < \tau < T_{i+1}/F_{T_i}) \right] \quad (4.11)$$

$$CVA = LGD. \sum_{i=1}^{N-1} \mathbb{E}_t^{\mathbb{Q}} \left[DF_{T_i} \cdot MtM_{T_i}^+ \cdot FX(T_i) (1 - \mathbb{E}_t^{\mathbb{Q}} \left[\exp \left(- \int_{T_i}^{T_{i+1}} \lambda(t) dt \right) / Y_{T_i} \right]) \right] \quad (4.12)$$

Les Y_t seront simulés dans les scénarios de Monte-Carlo comme des gaussiens corrélés au taux d'intérêt et au taux de change.

Dans le modèle de black-Karasinski de temps de défaut, soit λ_t l'intensité de défaut, on pose $\lambda_t = \exp(Y_t)$ où :

$$dY(t) = \alpha_y(\beta_y - Y(t))dt + \sigma_y dW_t \quad (4.13)$$

On veut étudier le défaut entre $[T_1, T_N]$, on admet une discrétisation temporelle $T_1; \dots; T_N$ (pour l'implémentaion de notre solveur, nous avons utilisé la même discrétisation utilisée dans la partie simulation).

Et ça pour $\forall i \in [0, T - 1]$.

Soit la probabilité conditionnelle à Y_t de défaut sur $[t, T_{i+1}]$:

$$\mathbb{Q}(t < \tau < T_{i+1} | Y_t) = 1 - \mathbb{E} \left[\exp \left(- \int_t^{T_{i+1}} \lambda(Y_s) ds \right) \middle| Y_t \right] \quad (4.14)$$

On pose $\forall i \in [1, N - 1]$

$$f^i(t, Y_t) = \mathbb{E} \left[\exp \left(- \int_t^{T_{i+1}} \lambda(Y_s) ds \right) \middle| Y_t \right] \quad (4.15)$$

Proposition 5 f^i satisfait l'EDP suivante :

$$\begin{cases} \frac{\partial f^i}{\partial t} + \alpha_y(\beta_y - Y(t)) \cdot \frac{\partial f^i}{\partial Y} + \frac{1}{2} \sigma_y^2 \frac{\partial^2 f^i}{\partial Y^2} - \exp(Y) f^i(t, Y) = 0 \\ f^i(T_{i+1}, Y) = 1 \end{cases} \quad (4.16)$$

Démonstration :

On définit la fonction :

$$\begin{aligned} \tilde{f}^i(t, Y_t) &= \exp \left(- \int_0^t \lambda(Y_s) ds \right) f^i(t, Y_t) \\ &= \mathbb{E} \left[\exp \left(- \int_0^{T_{i+1}} \lambda(Y_s) ds \right) \middle| Y_t \right] \end{aligned}$$

Alors $\tilde{f}^i(t, Y_t)$ est une martingale et :

$$d\tilde{f}^i(t, Y_t) = \exp\left(-\int_0^t \lambda(Y_s) ds\right) df^i(t, Y_t) - \lambda(t) \exp\left(-\int_0^t \lambda(Y_s) ds\right) dt \cdot f^i(t, Y_t) \quad (4.17)$$

$$= \tilde{f}^i(t, Y_t) \frac{df^i(t, Y_t)}{f^i(t, Y_t)} - \lambda(t) \tilde{f}^i(t, Y_t) dt \quad (4.18)$$

$$= \tilde{f}^i(t, Y_t) \left(\frac{df^i(t, Y_t)}{f^i(t, Y_t)} - \lambda(t) dt \right) \quad (4.19)$$

On applique le lemme d'Ito à la fonction f^i :

$$df^i = \frac{\partial f^i}{\partial t} dt + \frac{\partial f^i}{\partial Y} dY + \frac{1}{2} \frac{\partial^2 f^i}{\partial Y^2} \sigma_y^2 dt \quad (4.20)$$

Et ça donne :

$$df^i = \frac{\partial f^i}{\partial t} dt + \frac{\partial f^i}{\partial Y} [\alpha_y(\beta_y - Y(t)) dt + \sigma_y dZ_t] + \frac{1}{2} \frac{\partial^2 f^i}{\partial Y^2} \sigma_y^2 dt \quad (4.21)$$

Ou bien :

$$df^i = \left(\frac{\partial f^i}{\partial t} + \frac{\partial f^i}{\partial Y} \alpha_y(\beta_y - Y(t)) + \frac{1}{2} \frac{\partial^2 f^i}{\partial Y^2} \sigma_y^2 \right) dt + \sigma_y \frac{\partial f^i}{\partial Y} dZ_t$$

Ainsi $\tilde{f}^i(t, Y_t)$ est une martingale donc , son terme à variation finie est nul :

$$\frac{\frac{\partial f^i}{\partial t} + \frac{\partial f^i}{\partial Y} \alpha_y(\beta_y - Y(t)) + \frac{1}{2} \frac{\partial^2 f^i}{\partial Y^2} \sigma_y^2}{f^i(t, Y_t)} - \lambda(t) = 0 \quad (4.22)$$

Donc, pour tout intervalle, on simule d'abord une variable uniforme U_i , on calcule $\mathbb{Q}(t_i < \theta < t_{i+1} | Y_{t_i})$ en utilisant un "backward" de cette EDP et enfin on a défaut dans cet intervalle si $U_i < \mathbb{Q}(t_i < \theta < t_{i+1} | Y_{t_i})$.

En effet, pour chaque $i \in [0, n-1]$ on calcule $f^i(T_i, Y_{T_i})$ par la résolution de (4.16) et par suit :

$$\mathbb{Q}(t_i < \tau < t_{i+1} | Y_{t_{i+1}}) = 1 - f^i(T_i, Y_{T_i}) \quad (4.23)$$

4.4.2.1 Résolution : Méthode des différences finies :

Lorsqu'on n'a pas une formule close de la solution d'une EDP (comme notre cas), on peut utiliser une méthode numérique pour calculer des valeurs approximatives à la solution.

En analyse numérique, la méthode des différences finies est une technique courante de recherche de solutions approchées d'équations aux dérivées partielles qui consiste à résoudre un système de relations (schéma numérique) liant les valeurs de fonctions inconnues en certains points suffisamment proches les uns des autres.

Une différence finie est un quotient qui donne une valeur proche d'une dérivée partielle, exemple :

$$\frac{\partial f(t, x)}{\partial t} \approx \frac{f(t, x) - f(t - \delta t, x)}{\delta t} \quad (4.24)$$

$$\frac{\partial f(t, x)}{\partial t} \approx \frac{f(t + \delta t, x) - f(t, x)}{\delta t} \quad (4.25)$$

$$\frac{\partial f(t, x)}{\partial x} \approx \frac{f(t, x) - f(t, x - \delta x)}{\delta x} \quad (4.26)$$

$$\frac{\partial f(t, x)}{\partial x} \approx \frac{f(t, x + \delta x) - f(t, x)}{\delta x} \quad (4.27)$$

$$\frac{\partial f(t, x)}{\partial x} \approx \frac{f(t, x + \delta x) - f(t, x - \delta x)}{2\delta x} \quad (4.28)$$

$$\frac{\partial^2 f(t, x)}{\partial x^2} \approx \frac{f(t, x + \delta x) - 2f(t, x) + f(t, x - \delta x, t)}{\delta x^2} \quad (4.29)$$

Pour ce faire, on construit un maillage en espace et en temps, en construisant une grille où chaque point ou noeud (t_i, x_j) est repéré sur la grille par :

$$\left\{ \begin{array}{l} T_j = j * \delta t + T_1 \\ x_i = i * \delta' x + x_0 \end{array} \right\}$$

(toujours même discrétisation pour le temps)

Avec $[x_0, x_M]$ l'intervalle d'espace sur lequel on veut résoudre l'équation. Et on suppose que le temps final (maturité) $T = T_n = n\delta t + T_1$.

Nous nous plaçons dans l'intervalle $[T_i, T_{i+1}]$:

On cherche alors $f^i(T_i, Y_{T_i})$

Pour ce faire, on rediscrétise notre intervalle $[T_i, T_{i+1}]$ en $T_{i,1}; T_{i,2} \dots T_{i,K}$ tel que $T_{i,1} = T_i$, $T_{i,K} = T_{i+1}$ et aussi $T_{i,k+1} - T_{i,k} = \delta x$.

On cherche une approximation de $f_{k,j}^i \approx f^i(T_{i,k}, x_j)$.

Avec cette notation, on a la condition finale :

$$\forall j, f_{K,j}^i = 1$$

Schéma Crank-Nicholson :

Si on utilise (4.25) :

$$\begin{aligned} \frac{\partial f^i(T_{i,k}, x_j)}{\partial t} &\approx \frac{f^i(T_{i,k+1}, x_j) - f^i(T_{i,k}, x_j)}{\delta t} \\ &\approx \frac{f_{k+1,j}^i - f_{k,j}^i}{\delta t} \end{aligned} \quad (4.30)$$

De même on peut utiliser (4.28) :

$$\frac{\partial f^i(T_{i,k}, x_j)}{\partial x} \approx \frac{1}{2} \left(\frac{f_{k,j+1}^i - f_{k,j-1}^i}{2\delta x} + \frac{f_{k+1,j+1}^i - f_{k+1,j-1}^i}{2\delta x} \right) \quad (4.31)$$

Ainsi pour (4.29)

$$\frac{\partial^2 f^i(t_{i,k}, x_j)}{\partial x^2} \approx \frac{1}{2} \left(\frac{f_{k,j+1}^i - 2f_{k,j}^i + f_{k,j-1}^i}{\delta x^2} + \frac{f_{k+1,j+1}^i - 2f_{k+1,j}^i + f_{k+1,j-1}^i}{\delta x^2} \right) \quad (4.32)$$

Et finalement l'EDP nous donne pour chaque $i \in [1, n-1]$:

$$\begin{aligned} & \frac{f_{k+1,j}^i - f_{k,j}^i}{\delta t} + \alpha_y(\beta_y - x_j) \frac{1}{2} \left(\frac{f_{k,j+1}^i - f_{k,j-1}^i}{2\delta x} + \frac{f_{k+1,j+1}^i - f_{k+1,j-1}^i}{2\delta x} \right) \\ & + \frac{1}{2} \sigma_y^2 \frac{1}{2} \left(\frac{f_{k,j+1}^i - 2f_{k,j}^i + f_{k,j-1}^i}{\delta x^2} + \frac{f_{k+1,j+1}^i - 2f_{k+1,j}^i + f_{k+1,j-1}^i}{\delta x^2} \right) - \exp(x_j) \frac{1}{2} (f_{k,j}^i + f_{k+1,j}^i) = 0 \end{aligned} \quad (4.33)$$

On a comme condition finale :

$$\forall j \in [0, M] / f_{K,j}^i = 1 \quad (4.34)$$

$$\begin{aligned} & f_{k+1,j}^i + \delta t \frac{\alpha_y(\beta_y - x_j)}{4\delta x} (f_{k+1,j+1}^i - f_{k+1,j-1}^i) + \frac{1}{2} \frac{\delta t}{2} \frac{\sigma_y^2}{\delta x^2} (f_{k+1,j+1}^i - 2f_{k+1,j}^i + f_{k+1,j-1}^i) - \frac{1}{2} \exp(x_j) f_{k+1,j}^i \delta t \\ & = f_{k,j}^i - \delta t \frac{\alpha_y(\beta_y - x_j)}{4\delta x} (f_{k,j+1}^i - f_{k,j-1}^i) - \frac{1}{2} \frac{\delta t}{2} \frac{\sigma_y^2}{\delta x^2} (f_{k,j+1}^i - 2f_{k,j}^i + f_{k,j-1}^i) + \frac{1}{2} \exp(x_j) f_{k,j}^i \delta t \end{aligned}$$

Donc, on peut écrire :

$$\begin{aligned} & \frac{1}{2} \underbrace{\left(-\frac{\alpha_y(\beta_y - x_j)\delta t}{2\delta x} + \frac{\delta t}{2} \frac{\sigma_y^2}{\delta x^2} \right)}_{-p_d^j} f_{k+1,j-1}^i + \underbrace{\left(1 - \frac{1}{2} \delta t \frac{\sigma_y^2}{\delta x^2} - \frac{1}{2} \exp(x_j)\delta t \right)}_{2-p_m^j} f_{k+1,j}^i + \frac{1}{2} \underbrace{\left(\delta t \frac{\alpha_y(\beta_y - x_j)}{2\delta x} + \frac{\delta t}{2} \frac{\sigma_y^2}{\delta x^2} \right)}_{-p_u^j} f_{k+1,j+1}^i \\ & = \frac{1}{2} \underbrace{\left(\frac{\alpha_y(\beta_y - x_j)\delta t}{2\delta x} - \frac{\delta t}{2} \frac{\sigma_y^2}{\delta x^2} \right)}_{p_d^j} f_{k,j-1}^i + \underbrace{\left(\frac{1}{2} \delta t \frac{\sigma_y^2}{\delta x^2} + \frac{1}{2} \exp(x_j)\delta t + 1 \right)}_{p_m^j} f_{k,j}^i + \frac{1}{2} \underbrace{\left(-\delta t \frac{\alpha_y(\beta_y - x_j)}{2\delta x} - \frac{\delta t}{2} \frac{\sigma_y^2}{\delta x^2} \right)}_{p_u^j} f_{k,j+1}^i \end{aligned}$$

D'une façon simplifiée :

$\forall j$

$$-\frac{1}{2} p_d^j \times f_{k+0,j-1}^i + \frac{1}{2} (2 - p_m^j) \times f_{k+1,j}^i - \frac{1}{2} p_u^j \times f_{k+1,j+1}^i = \frac{1}{2} p_d^j \times f_{k,j-1}^i + \frac{1}{2} p_m^j \times f_{k,j}^i + \frac{1}{2} p_u^j \times f_{k,j+1}^i \quad (4.35)$$

Remarque : En général on peut montrer que pour un schéma mixte⁷ de coefficient λ , on a :

7. le schéma de Crank-Nicholson est un schéma mixte avec $\lambda = 0.5$

$$-\lambda p_d^j \times f_{k+0,j-1}^i + \lambda(2-p_m^j) \times f_{k+1,j}^i - \lambda p_u^j \times f_{k+1,j+1}^i = (1-\lambda)p_d^j \times f_{k,j-1}^i + (1-\lambda)p_m^j \times f_{k,j}^i + (1-\lambda)p_u^j \times f_{k,j+1}^i \quad (4.36)$$

Avec la condition finale (4.34) qui nous donne :

$$\forall j, f_{K,j}^i = 1$$

Donc, l'écriture matricielle suivante résume notre méthodologie de résolution :

$$\begin{aligned} & \lambda \begin{pmatrix} 2-p_m^1 & -p_u^1 & 0 & \cdot & \cdot & \cdot & 0 \\ -p_d^2 & 2-p_m^2 & -p_u^2 & \cdot & \cdot & \cdot & \cdot \\ 0 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 0 \\ \cdot & \cdot & \cdot & \cdot & -p_d^{M-1} & 2-p_m^{M-1} & -p_u^{M-1} \\ 0 & \cdot & \cdot & \cdot & 0 & -p_d^M & 2-p_m^M \end{pmatrix} * \begin{pmatrix} f_{K,1}^i = 1 \\ f_{K,2}^i = 1 \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ f_{K,M}^i = 1 \end{pmatrix} \\ & = (1-\lambda) \begin{pmatrix} p_m^1 & p_u^1 & 0 & \cdot & \cdot & \cdot & 0 \\ p_d^2 & p_m^2 & p_u^2 & \cdot & \cdot & \cdot & \cdot \\ 0 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 0 \\ \cdot & \cdot & \cdot & \cdot & p_d^{M-1} & p_m^{M-1} & p_u^{M-1} \\ 0 & \cdot & \cdot & \cdot & 0 & p_d^M & p_m^M \end{pmatrix} * \begin{pmatrix} f_{K-1,1}^i \\ f_{K-1,2}^i \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ f_{K-1,M}^i \end{pmatrix} \quad (4.37) \end{aligned}$$

On note :

$$A' C_K^i = A C_{K-1}^i$$

Où :

$$A^{-1} A' C_K^i = C_{K-1}^i$$

Donc le calcul de C_{K-1}^i revient au calcul de A^{-1} l'inverse de la matrice A .

On fait un backward de K itérations pour trouver $f_{1,j}^i = f^i(T_i, x_j)$ pour $j \in [0, M]$ et ensuite on n'a qu'à trouver un certain I tel que : $x_I \leq Y(T_i) < x_{I+1}$ et on trouve $f^i(T_i, Y_{T_i})$ par interpolation.

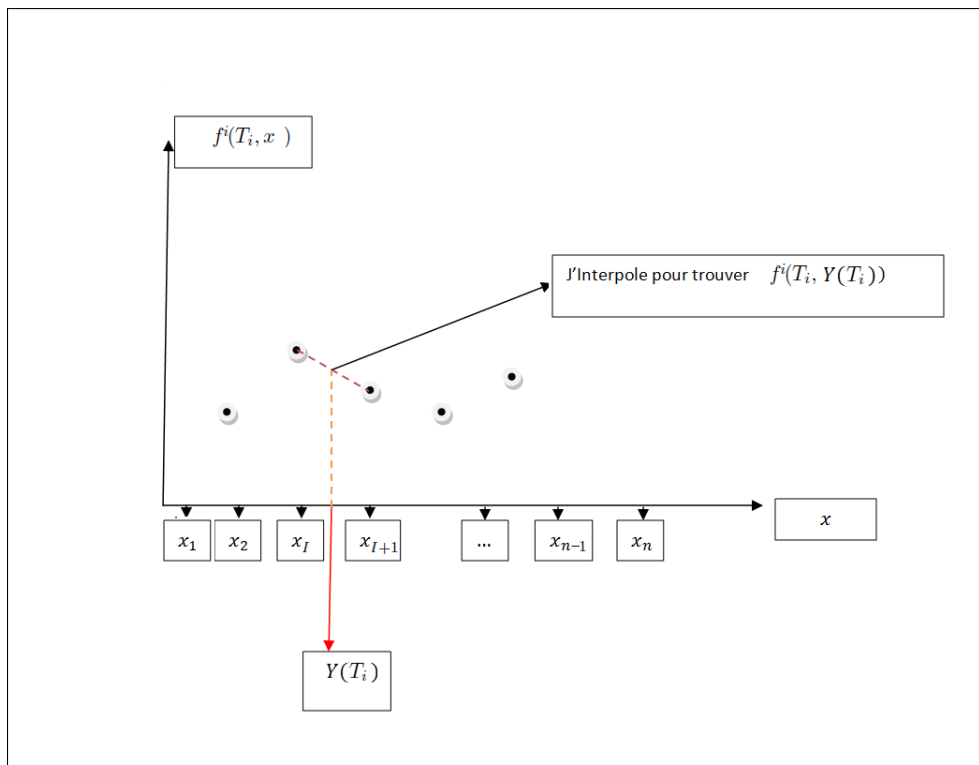


FIGURE 4.1 – Interpolation des solutions de l'EDP

Algorithm 2 calculer $f^i(T_i, Y(T_i))$

Require : $A^{-1}A'$
 $C \leftarrow 1, 1, 1, 1, \dots, 1$
for $i \leftarrow K$ **to** 1 **do**
 $C \leftarrow A^{-1}A' * C$
end
return Interpolate($C, y[T_i]$)

 Interpolate trouve $f^i(Y(T_i))$ par interpolation

Pour l'application, nous avons pris $x_{min} = \theta - 3 * \sigma_y$ et $x_{max} = \theta + 3 * \sigma_y$ cela est justifié par le fait que Y est gaussienne et alors la probabilité de tomber hors de l'intervalle $[\theta - 3 * \sigma_y, \theta + 3 * \sigma_y] < 0.01$ chose qui est satisfaisante comme précision .

4.5 Application : Calcul de la CVA pour un Swap :

4.5.1 Calcul théorique

Etant donné qu'un swap⁸ peut être de valeur positive ou négative, à certains moments, il peut être vu comme un actif et être exposé au risque de défaut de la contrepartie, tandis qu'à d'autres moments, il peut être vu comme un passif et ne pas y être exposé.

Soit R_S le taux de recouvrement du swap, supposé constant. Au moment du défaut de la contrepartie τ , si le swap a une valeur positive ($swap(\tau, \beta(\tau), T_N) > 0$), alors l'investisseur reçoit

8. voir définition chapitre précédent

$R_s swap(\tau, \beta(\tau), T_N)$ donc il perd $(1 - R_s) swap(\tau, \beta(\tau), T_N)$.
 Avec $\beta(t) = \max\{j | T_j \leq t\}$

Si la valeur du swap est négative, l'investisseur ne perd rien.
 La perte perçue par l'investisseur en τ est donc :

$$(1 - R_s) [swap(\tau, \beta(\tau), T_N)]^+ \quad (4.38)$$

Le terme $swap(\tau, \beta(\tau), T_N)^+$ n'est que le payoff d'un call sur le swap (non defaultable) au temps de défaut τ c.-à-d. le payoff d'une swaption de ténor T_N vu de τ .

En prenant l'espérance sous la probabilité risque neutre de la perte, la valeur de la CVA vaut :

$$(1 - R_s) \mathbb{E}[(swap(\tau, \beta(\tau), T_N))^+ DF_\tau 1_{\tau < T_N}] \quad (4.39)$$

En introduisant le taux de change :

$$(1 - R_s) \mathbb{E}[(swap(\tau, \beta(\tau), T_N))^+ FX_\tau DF_\tau 1_{\tau < T_N}] \quad (4.40)$$

Et sous la forme discrétisée :

$$CVA = LGD. \sum_{i=1}^{N-1} \mathbb{E}_t^{\mathbb{Q}} \left[DF_{T_i} \cdot (swap(T_i, \beta(T_i), T_N))^+ \cdot FX(T_i) \cdot \mathbb{Q}(T_i < \tau < T_{i+1} / F_{T_i}) \right] \quad (4.41)$$

Avec DF : Discount Factor⁹

Avec le prix d'un swap comme est donné par (3.42) :

$$swap(t, T_I, T_N) = N_m \sum_{i=I+1}^N P(t, T_i) h \left[\frac{\frac{1}{P(t, T_{i-1}, T_i)} - 1}{h} - K \right] \quad (4.42)$$

4.5.2 Implémentation

La majeure partie du travail a été de créer deux outils qui prennent en paramètre l'intensité entre T_i et T_{i+1} ainsi qu'une discrétisation et fournit la probabilité de faire défaut entre $[T_i, T_{i+1}]$, le premier outil en utilisant la méthode intégrale et l'autre en utilisant un solveur EDP.

Le pricer implémenté est générique et indépendant du type du produit, dont le moteur MonteCarlo, après l'exécution du simulateur (vu au chapitre précédent), il fait appel à l'un des deux outils (selon le choix) qui fournissent les probabilités de défaut et enfin retourne le prix de la CVA du produit dans ce scénario.

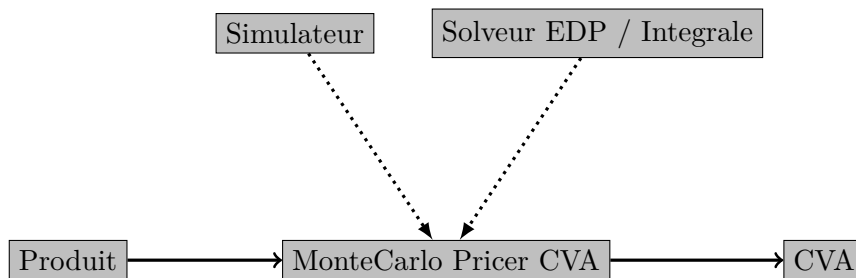


FIGURE 4.2 – Schéma du CVA pricer MC

9. le Discount Factor est calculé comme une intégrale des taux court simulés

Une classe qui calcul la CVA d'un produit :

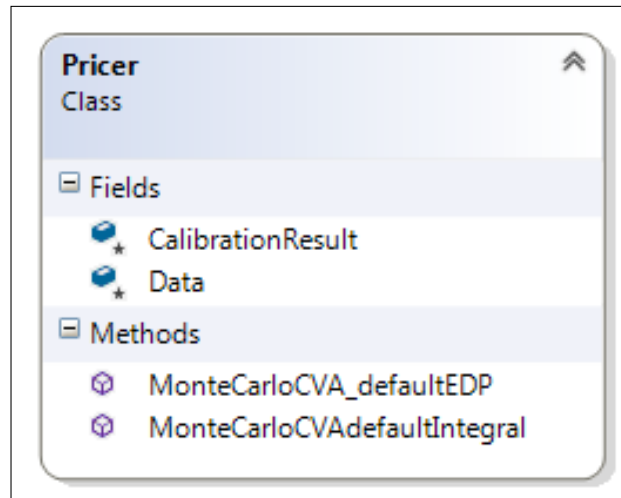


FIGURE 4.3 – Classe de pricer CVA

Résultat de Calcul de la CVA pour le même Swap présenter dans le chapitre précédent :

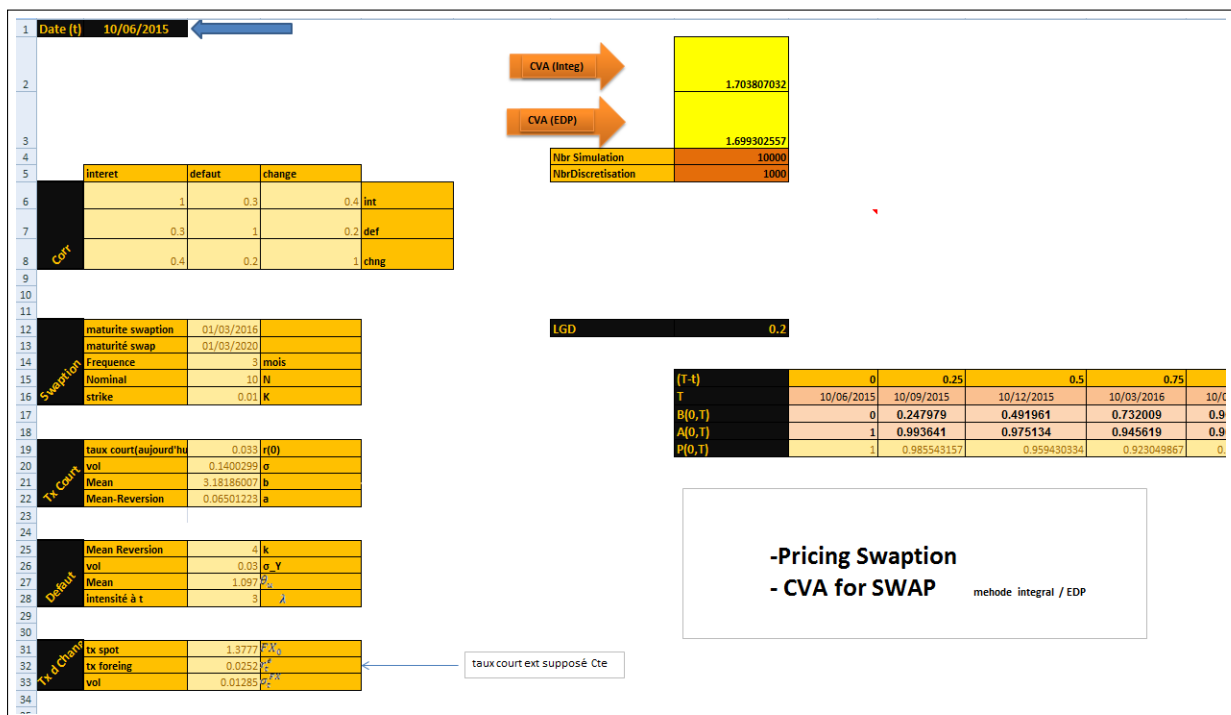


FIGURE 4.4 – Résultat de calcul CVA pour un swap

Conclusion

L'objectif de ce projet était de réaliser un " pricer " sous C# qui calcule la valeur CVA pour un produit de devise étrangère, en utilisant une méthode Monte-Carlo. En effet le pricer calcule le prix de la CVA comme la moyenne des valeurs de la perte possible dans chaque scénario simulé.

Dans le 3ème chapitre, nous nous sommes concentrés sur la méthode de simulation de ces scénarios, une simulation qui revient à la modélisation hybride de plusieurs facteurs corrélés entre eux. Ensuite, nous avons développé un outil de valorisation qui valorise un produit défautable, à savoir, une swaption défautable qui, à la maturité, donne le droit d'entrer en contrat swap s'il n'y a pas eu de défaut.

Dans le 4ème chapitre, nous avons introduit la notion de la CVA et ses formules de calculs. La partie majeure de ce chapitre était d'estimer les probabilités de défaut à partir de l'intensité simulée, c'est pour ce faire qu'on a introduit les deux méthodes, intégrale et EDP avec une résolution numérique.

Après avoir établi les formules permettant de retrouver la CVA d'un produit, nous avons développé un outil qui calcule la CVA pour un produit par une Méthode de Monte-Carlo. Nous avons ainsi présenté un exemple de calcul de CVA pour un swap.

Bibliographie

- [1] Albahari, J. B. Albahari, 2012, « *C# 5.0 IN A NUTSHELL* », 5th Edition, O'REILLY.
- [2] Benayad, M. D.DAHAN, 2008/2009, « *Modélisation et évaluation des produits dérivés de crédit* », Document interne SG.
- [3] Bennehou, S. , September 18, 2007 , « *Default time simulation : state of the art* » ,(Internal document SG)
- [4] Brigo, D., 10 nov 2011, « *Counterparty risk faq* » , Pearson education international, defaultrisk.com.
- [5] Brigo, D., 2006, « *Interest Rate- Models Theory and Practice* » , 2nd Edition, Springer Finance, 2006.
- [6] Fauth, A., 2014/2015, « *Surface de Volatilité et Introduction au Risque de Crédit* », Spécialités Mathématiques du Risque & Finance Computationnelle , Mathématique et Finance
- [7] Glasserman , P., 2003, *Carlo Methods in Finance Springer* » .
- [8] Hull, J., 2009, « *Options, Futures and other derivatives* » , 7th edition, Pearson education international.
- [9] Lange, L., 24 octobre 2012, « *Méthodes de calcul de CVA* » , Modélisation Aléatoire, Paris 7.
- [10] Marri F., « *Séries chronologiques* » , INSEA Rabat , 2014
- [11] Meyre, T. & F. Comet, 2006, « *Calcul stochastique et modeles de diffusions.* » Dunod.
- [12] Millet, September 18, 2007, 2012, « *Methodes de monte-carlo,* » ,
- [13] Taik, A., 2008, « *Equations aux Dérivées Partielles Méthodes des Différences Finies,* » , Département de Mathématiques FST-Mohammedia.
- [14] Toke, I.M. 11 janvier 2011, « *Modèles stochastiques de taux d'intérêts* » , Option Mathématiques Appliquées, Ecole Centrale Paris.

Webographie

[w1] <http://africa-technologies-services.sgcib.com/>

[w2] <http://www.investopedia.com>

[w3] <http://www.iotafinance.com/Article-Construction-d-une-Courbe-de-Taux-zéro-coupon-par-Bootstrapping.html>

[w4] <http://sites.google.com/site/calculstochastiqueetmodeles/construction-de-la-courbe-des-taux>

[w5] <http://pandas.pydata.org/pandas-docs/version/0.12.0/>

[w5] <http://finance.sia-partners.com/20140708/cva-dva-fva-ou-lextrême-sophistication-de-la-valorisation-des-derives-de-credit/>

[w5] <https://msdn.microsoft.com/fr-fr/library/ms182532.aspx/>

Présentation de la méthode MonteCarlo :

La méthode de simulation de Monte Carlo est une méthode algorithmique visant à calculer une valeur numérique approchée en utilisant des procédés aléatoires, c'est-à-dire des techniques probabilistes. Le nom de cette méthode, qui fait allusion aux jeux de hasard pratiqués à Monte-Carlo, a été inventé en 1947 par Nicolas Metropolis, et publié pour la première fois en 1949 dans un article coécrit avec Stanislaw Ulam. La méthode de Monte Carlo est une généralisation de la méthode de l'aiguille de Buffon, la méthode par laquelle Louis Buffon a pu rapprocher la probabilité qu'une aiguille jetée d'une même hauteur tombe à cheval sur une rainure d'un parquet composé de planches parallèles de même largeur (cas " favorable ") par rapport au nombre total de lancers. Au fur et à mesure que le nombre de lancers augmente, le quotient se rapproche d'un certain nombre .

Détermination de la valeur de p (π) : Soit un point M de coordonnées (x, y) , où $0 < x < 1$ et $0 < y < 1$. On tire aléatoirement les valeurs de x et y . Le point M appartient au disque de centre $(0,0)$ de rayon 1 si et seulement si $x^2+y^2 \leq 1$. La probabilité que le point M appartienne au disque est $\pi/4$ (l'aire du quart de cercle de Rayon 1).

En faisant le rapport du nombre de points dans le disque au nombre de tirages, on obtient une approximation du nombre $p/4$ si le nombre de tirages est grand.

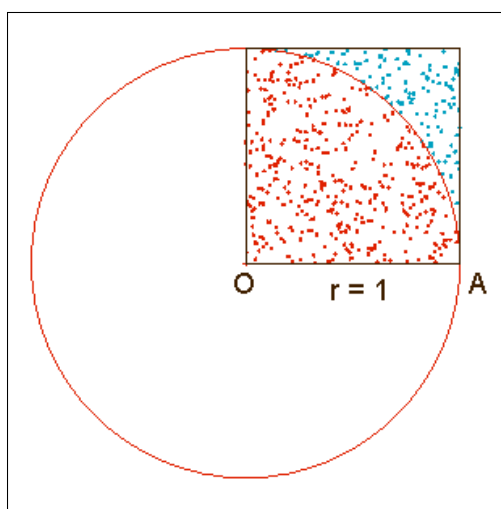


FIGURE A.1 – Détermination de la valeur Pi par MonteCarlo

Tutoriel Excel-dna :

Excel-DNA est un projet open-source indépendant qui intègre .NET dans Excel. Le runtime Excel-DNA est distribué sous une licence libérale qui autorise explicitement l'utilisation commerciale. Excel-DNA offre la possibilité d'intégrer des modules de calcul développer su C# à Excel suivant l'interface la plus rapide celle des fonctions de feuille de calcul.

Création d'une nouvelle bibliothèque complément dans C# :

Dans cette section, nous créons un addin simple de C#, qu'on intégrera à Excel avec le runtime Excel - DNA.

- Dans Visual Studio, créez un nouveau projet de bibliothèque de classes

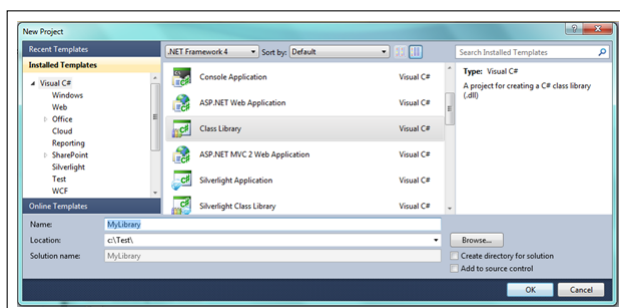


FIGURE B.1 – Aperçue Visual Studio

- Ajouter une référence à l'assembly ExcelDna.Integration.dll (qui est dans le répertoire de distribution Excel - DNA. Cochez cette référence copie localeFaux :

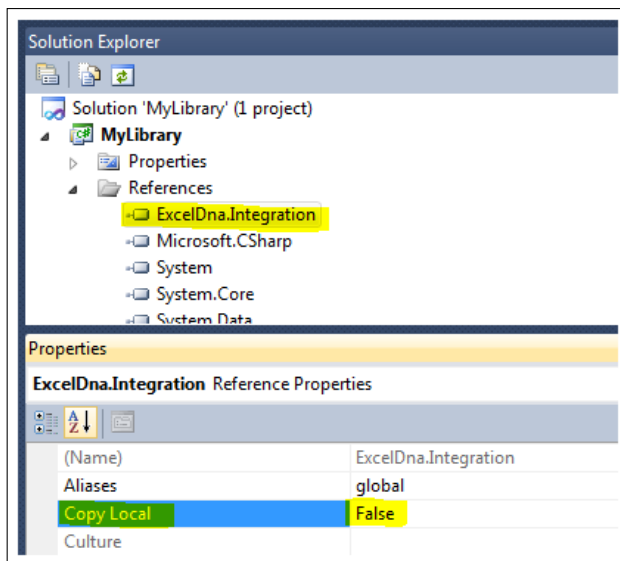


FIGURE B.2 – Solution Explorer-Visual Studio

- Dans le fichier Class1.cs, ajoutez une déclaration d'espace de noms et une fonction publique statique de la classe Class1 :

```

using System;
using ExcelDna.Integration;

namespace MyLibrary
{
    public class Class1
    {
        [ExcelFunction(Description="My first Excel-DNA function")]
        public static string MyFirstFunction(string name)
        {
            return "Hello " + name;
        }
    }
}

```

FIGURE B.3 – Exemple pour Excel DNA

- Ajouter un fichier au projet appelé " FirstAddIn.dna " avec le contenu suivant :

```

<DnaLibrary Name="First Add-In" RuntimeVersion="v4.0">
  <ExternalLibrary Path="MyLibrary.dll" />
</DnaLibrary>

```

FIGURE B.4 – Class First ADD IN

- Dans les propriétés de ce fichier, faites que ce fichier soit copié dans le répertoire de sortie :
- Faites une copie de la ExcelDna.xll dans le répertoire du projet, et le renommer en FirstAddIn.xll :
- Ajouter ce fichier dans le projet, et dans les propriétés du fichier, s'assurer qu'il sera copié dans le répertoire de sortie :
- Double - cliquez sur le FirstAddIn.xll à ouvrir dans Excel (sélectionnez Fichier- > Ouvrir une session Excel).
- Dans une cellule, entrez = MyFirstFunction ("Daniel")
- Vérifiez la description de fonction dans l'assistant de fonction :

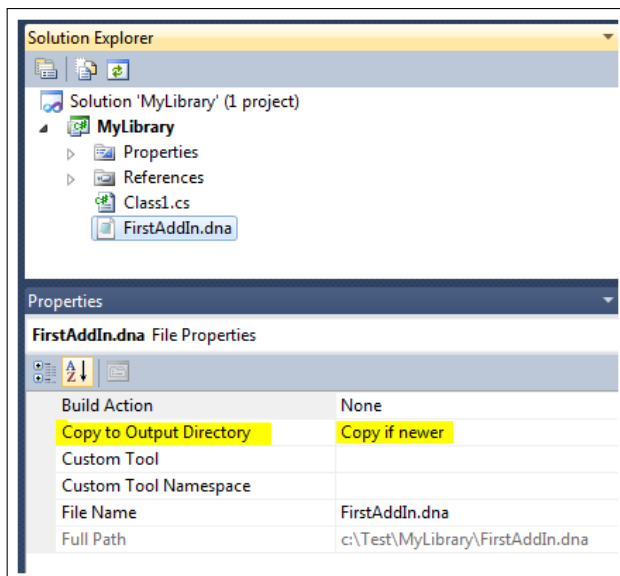


FIGURE B.5 – Propriétés Class First to Add In

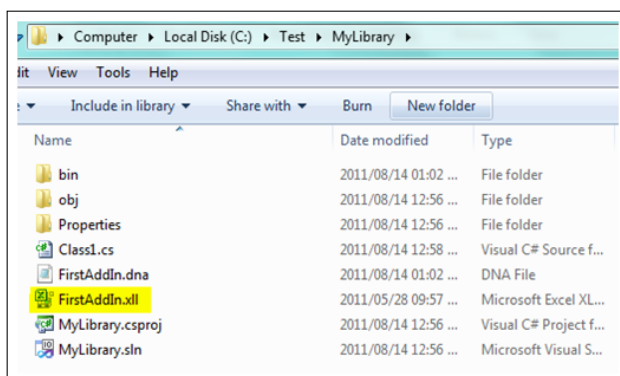


FIGURE B.6 – Ajout du fichier Excel

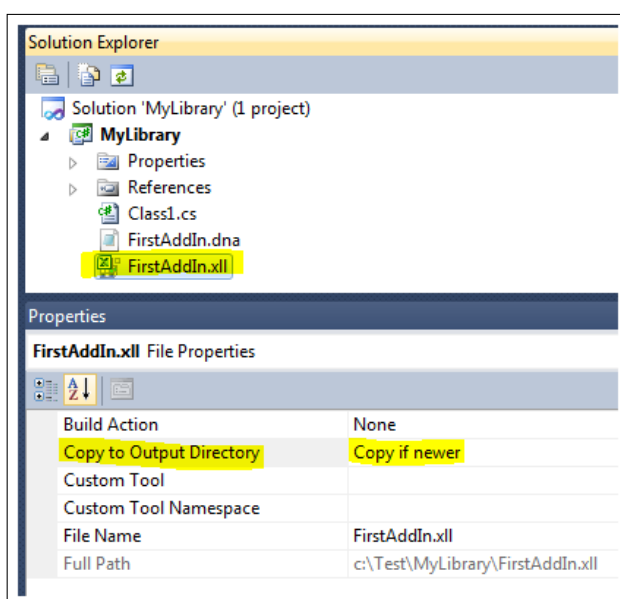


FIGURE B.7 – FirstAddIn.dll Propriété

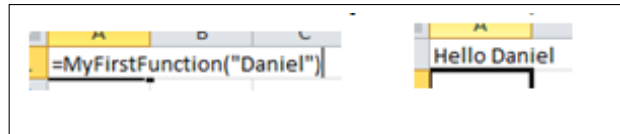


FIGURE B.8 – Exemple sur excel appel de C#

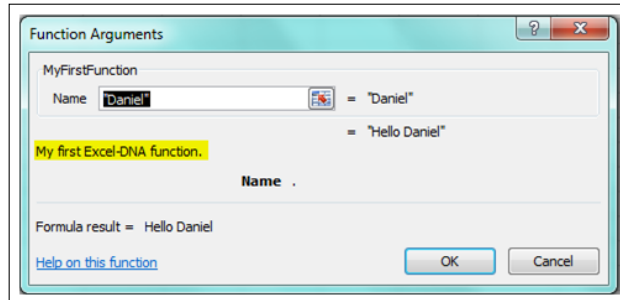


FIGURE B.9 – Fonction Exemple Excel

F I N

 E i U