

ÉCOLE NATIONALE SUPÉRIEURE D'INFORMATIQUE ET DE MATHÉMATIQUES APPLIQUÉES DE GRENOBLE

RAPPORT DE PROJET D'ÉVALUATION DE PRODUITS STRUCTURÉS

VERSION BETA

Multimonde 2021

26 JANVIER 2018



Bernard Julien
Berthon Christophe
Herbadji Alexandra
Moine Yoann
Namasse Soufiane

Enseignant : Ollivier TARAMASCO

Table des matières

1	Introduction	3
2	Cadre de l'étude	3
3	Description du produit MultiMonde 2021	3
4	Analyse des flux financiers	3
4.1	Description des flux	3
4.2	Définition mathématique des flux	4
5	Analyse des risques	4
5.1	Différents types de risques	5
5.2	Analyse concrète des Risques	5
5.2.1	Risque de marché	5
5.2.2	Risque de Change	5
6	Modèle Mathématiques	6
6.1	Dynamique des sous-jacents	6
6.2	Modélisation des taux de changes	6
6.3	Modélisation de la volatilité	6
7	Portefeuille de couverture	6
7.1	Construction du Portefeuille de couverture	7
8	Architecture logicielle	7
8.1	Architecture globale	7
8.1.1	PricerDll	7
8.1.2	PricerConsole	8
8.1.3	PricerDll.CustomTests	9
8.1.4	PEPS-Beta et PEPS-Beta.Tests	10
8.2	Instructions d'utilisation	10
8.3	Base de données financières	10
8.4	Fonctionnalités implémentées	11

9	Moteur de calcul	11
9.1	Calcul des prix	11
9.2	Calcul du prix	11
9.3	Prix en 0	12
9.4	Prix en t	12
9.5	Calcul des deltas	12
9.6	Simulation de la couverture	12
9.7	Dates choisies	13
9.8	Dates de rebalancement	13
10	Tests et Validation	13
10.1	Call Vanille	13
10.1.1	Prix en $t = 0$	13
10.2	Prix en $0 \leq t$	13
10.3	Calcul des deltas à $t=0$ et $t \geq 0$	14

1 Introduction

2 Cadre de l'étude

Dans le cadre du Projet d'Évaluation de Produits Structurés, l'objectif est de concevoir une interface informatique permettant au gérant d'un produit structuré d'évaluer et de couvrir son produit. Le produit à charge est le MultiMonde 2021 dont la société de gestion est CIC Banque. Ce rapport a pour objectif de fournir une analyse financière du produit à partir de son Document d'Information Clé pour l'Investisseur (DICI), disponible auprès de l'institut nancier qui le commercialise. Pour ce faire, nous étudierons d'abord les flux financiers puis nous analyserons les risques auxquels s'expose la banque qui émet ce produit ainsi que ceux auxquels s'expose l'acheteur du produit.

3 Description du produit MultiMonde 2021

4 Analyse des flux financiers

4.1 Description des flux

Dans le cas d'une souscription à l'émission du titre, deux flux monétaires majeurs ont lieu :

- Lors de la souscription, le client paie 101,50 au CIC, dont 1,5 représente les frais de souscription qui paieront les frais relatifs à la gestion de l'actif et ne seront donc pas pris en compte dans les flux versés par l'actif pour son pricing. Ainsi, le capital investi est de 100.
- Le 18 novembre 2021, le porteur est remboursé et touche le payoff du Multimonde 2021, que nous allons expliciter par la suite.

Des frais de gestion annuels variables allant jusqu'à 1,20% de la valeur du fond sont appliqués.

Le payoff du produit est calculé de la manière suivante :

Les dates de constatations associées à cet actif sont le 7 octobre 2016, le 13 octobre 2017, le 19 octobre 2018, le 25 octobre 2019, le 30 octobre 2020 et le 5 novembre 2021.

A chaque date de constatation, on enregistre la meilleure performance réalisée par les indices pour leurs valeurs réelles (c'est à dire prenant en compte l'inflation), ce, par rapport à la valeur liquidative initiale constatée au 1er octobre 2015. On retire alors le sous-jacent ayant le mieux performé du panier d'actifs. Les indices sont pris dividendes non réinvestis.

La performance est ramenée à +15% si elle est supérieure à +15%, et -15% si elle est inférieure à -15%.

Lors de la dernière date de constatation, on calcule la valeur liquidative finale du fond en multipliant la valeur liquidative initiale (de 100 pour un actif) par la somme des performances retenues.

La valeur liquidative finale est donc incluse entre 10 et 190.

4.2 Définition mathématique des flux

Notons VLI_i la valeur liquidative initiale de l'indice i avec $i \in \llbracket 1, 6 \rrbracket$,

$VLC_{t,i}$, la Valeur Liquidative de l'indice i à la date de constatation t avec (i,t) corrigée de l'inflation par rapport à la valeur liquidative initiale $\in \llbracket 1, 6 \rrbracket$,

VLF_i la valeur liquidative finale pour l'indice i du panier, avec $i \in \llbracket 1, 6 \rrbracket$.

On a alors :

$$VLF_i = \max\{-15\%, \min\{15\%, \max_{S \in I_{t-1}} \left\{ \frac{VLC_{i,t-1} - VLI_i}{VLI_i} \right\}\}\}$$

On associe les indices 0 à 6 de façon croissante aux dates 1/10/15; 7/10/16; 13/10/17; 19/10/18; 25/10/19; 30/10/20; 5/11/21.

Notons I_1 l'ensemble des indices de départ.

Mathématiquement :

$$I_1 = \{\text{EuroStoxx 50, S\&P 500, Nikkei 225, HangSeng, FTSE100, S\&P ASX200}\}$$

$$= \{S_0, S_1, S_2, S_3, S_4, S_5, S_6\}$$

De plus, à chaque date de constatation t , on a :

$$\forall t \in \llbracket 2, 6 \rrbracket, I_t = I_{t-1} \setminus \left\{ \underset{S_i \in I_{t-1}}{\operatorname{argmax}} \left\{ \frac{VLC_{i,t-1} - VLI_i}{VLI_i} \right\} \right\}$$

On a donc la formule de Pay-off suivante :

$$\text{Payoff} = 100 * \left(1 + \sum_{t=1}^6 \max\{-15\%, \min\{15\%, \max_{i \in \llbracket 1, 6 \rrbracket, S_i \in I_{t-1}} \left\{ \frac{VLC_{i,t-1} - VLI_i}{VLI_i} \right\}\}\} \right)$$

5 Analyse des risques

Dans le Projet d'Évaluation de Produit Structurés, nous sommes amenés à travailler sur des produits soumis à des risques, et qui sont complexes, c'est à dire des produits qui ne se trouvent pas sur le marché mais qui résultent de différents produits provenant de classes d'assets différentes.

Un des intérêts majeurs à recourir aux produits structurés est de fournir des produits financiers nouveaux qui sachent répondre à des besoins plus spécifiques. Ils permettent d'obtenir des profils de rendement et de risque plus fins. Nous allons nous intéresser aux risques qui entourent notre produit structuré et nous allons étudier des moyens pour se prémunir contre ces risques.

5.1 Différents types de risques

L'émetteur du produit structuré est exposé à nombre de risques. Ils comprennent :

- Le risque de Marché : Le risque de marché couvre l'ensemble des activités et des variations des différents marchés mondiaux et pouvant impacter la performance de notre portefeuille et a fortiori les produits structurés comme le MultiMonde : de fortes variations dans un des indices du panier sont en mesure de faire baisser l'ensemble de la performance du produit structuré.
- Le risque de taux d'intérêt : Les taux d'intérêt du marché sont essentiels pour déterminer le drift des taux de change. Leurs variations sont donc impactantes sur l'ensemble de nos modèles de calculs et par conséquent sur le pricing du MultiMonde. De plus, les taux d'intérêt disponibles sur les marchés ont des conséquences sur l'ensemble des instruments liés au marché de taux : swaps – futures – forwards. item Le risque de Liquidité : Il permet de prendre en compte des éventuels impayés de la part de nos contreparties. En eet, étant donné qu'on sera amenés à utiliser des assets de la classe xed-income obligations, futures...etc.).
- Le risque de Change : Le multimonde comprend des titres de différentes devises (deux en euros (EUR), un en dollar américain (USD), un en livre-sterling (GBP), un en yen (JPY), un en Hong-Kong Dollar (HKD) et un dollar australien (AUD)). Or les flux liés au portefeuille sont calculés en euros (EUR), la devise de référence du contrat. Ainsi, de grandes variations dans les taux de change des six monnaies en jeu peuvent altérer les performances du produit structuré.
- Le risque Opérationnel : Ce risque est lié aux erreurs de reporting ou alors des erreurs de gestion. Il est difficile à prendre en compte.

5.2 Analyse concrète des Risques

5.2.1 Risque de marché

Pour gérer ce risque, on va adopter une couverture en delta (delta-hedging). Il s'agit de se couvrir contre des variations des prix de sous-jacents (au premier ordre du moins).

5.2.2 Risque de Change

Avec une proportion de 80% de nos indices côtés dans une devise étrangère, nous nous devons d'établir différents scénarios pour les marchés de change des devises suivantes : EUR, USD, GBP, YUP, YEN. La gestion de ce risque va consister à considérer les sous-jacents en devise étrangère comme des sous-jacents en devise domestique en rajoutant des dynamiques de taux de change . On va s'y prendre de la même façon que pour le risque de marché, à ceci près que les taux de change sont eux-même vus comme des actifs. Un delta-hedging sur un sous-jacent en devise étrangère exprimé avec un strike en euro permet de se couvrir contre les variations des taux de change. Il s'agit donc d'adopter la même approche qu'une option Quanto (qui est couverte contre le risque de taux de change car elle délivre le payoff dans la monnaie de référence).

6 Modèle Mathématiques

6.1 Dynamique des sous-jacents

La valeur liquidative finale de notre produit est basée sur les performances du panier composé des 6 indices. Pour modéliser le comportement du cours des indices, nous utiliserons le modèle de Black-Scholes.

Ce modèle repose sur les hypothèses suivantes :

- L'absence d'opportunité d'arbitrage
- Les taux sans risques sont constants et connus
- Le temps est continu
- pas de coûts de transaction
- les ventes à découvert sont possibles

Sous ces hypothèses, le prix de l'actif $S_{i,t}$ a pour dynamique :

$$S_{i,t} = \mu_i S_{t,i} dt + \sigma_i S_{t,i} dW_{i,t}$$

où $W_{t,i}$ est un processus de Wiener, σ_i la volatilité de l'actif, constante, et μ_i une dérive, constante. L'évolution du sous-jacent vérifie alors :

$$S_{i,t} = S_{0,i} \exp\left(\left(\mu_i - \frac{\sigma_i^2}{2}\right)t + \sigma_i W_{t,i}\right)$$

6.2 Modélisation des taux de changes

Les taux de changes sont modélisés comme des actifs risqués versant un dividende. Ce dividende est le taux sans risque du marché considéré par rapport au marché domestique. En notant $X_{t,i}$ le taux de change EUR/USD, on a la dynamique suivante :

$$X_{i,t} = X_{t,i}[(r - r_{\$})dt + \sigma_i dW_{i,t}]$$

Les autres taux de change peuvent s'exprimer de manières similaires.

6.3 Modélisation de la volatilité

Le modèle à volatilité constante est celui qui est adopté pour le moment. Il sera possible de choisir une volatilité historique en effectuant l'opération suivante sur les données des sous-jacents :

$$r_{i,j} = \log\left(\frac{S_{i,j}}{S_{i,j-1}}\right)$$

7 Portefeuille de couverture

Notre outil permettra de réaliser un portefeuille de couverture pour se couvrir contre les risques de change et de marché.

7.1 Construction du Portefeuille de couverture

En utilisant le simulateur, nous obtenons les prix des sous-jacents et des deltas aux dates de rebalancement. Grâce à ces valeurs, l'application construit le portefeuille de couverture destiné à répliquer le payoff du MultiMonde2021. Les valeurs des deltas permettent d'estimer la quantité à investir dans les indices, en zéro coupon étranger et en zéro coupon domestique. Ainsi, on récupère les quantités à investir dans chaque indice du portefeuille à chaque date de rebalancement.

8 Architecture logicielle

8.1 Architecture globale

La solution PEPS-Beta est divisée en 5 sous-projets. Elle est destinée à être ouverte et lancée avec Visual Studio 2017, avec une configuration externe minimale à effectuer, décrite ultérieurement dans cette section.

8.1.1 PricerDll

PricerDll est une **Dll C++**, qui contient l'intégralité des fonctionnalités calculatoires du projet. Le C++ a été utilisé pour une vitesse de simulation maximale.

Le code est organisé autour d'un couple de fichiers **PricerDll.h** et **PricerDll.cpp**, qui exposent similairement à une API les fonctions accessibles aux projets extérieurs. Ces fichiersinstancient dans leurs fonctions les autres entités nécessaires à la logique métier des fonctionnalités, notamment les trois classes suivantes : **Option**, **MonteCarlo** et **BlackScholesModel**.

Option est une classe abstraite dont héritent les options que PricerDll peut effectivement pricer. Pour ce rendu, PricerDll peut pricer les options d'achat sur paniers d'actifs (**BasketOption**) et l'option Multimonde2021 (**Multimonde2021**).

Pour ces 2 options, 3 types d'appels sont disponibles dans l'API de la Dll.

Obtenir le prix, obtenir les deltas relativement à l'Euro (par convention, le premier élément des tableaux correspond à l'actif européen), et obtenir les deltas de l'actif quant à ses sous-jacents dans leurs monnaies respectives, ce qui est une étape intermédiaire au calcul des deltas précédents.

Ces trois types d'appels peuvent être fait en 0 ou en t, comme conséquence du développement incrémental du pricer.

Le schéma de nommage des fonctions exposées via API est le suivant :
(Price/DeltasSingleCurrency/DeltasMultiCurrency)(Basket/Multimonde2021)
(\emptyset /AnyTime)

Pour un total donc, de 12 fonctions. Les appels Price renvoient également la largeur de l'intervalle de confiance à 95%.

8.1.2 PricerConsole

PricerConsole est une **application console C#** interactive permettant d'effectuer les appels du Pricer relatif au Multimonde, ou de simuler une couverture du Multimonde avec calcul de la tracking error.

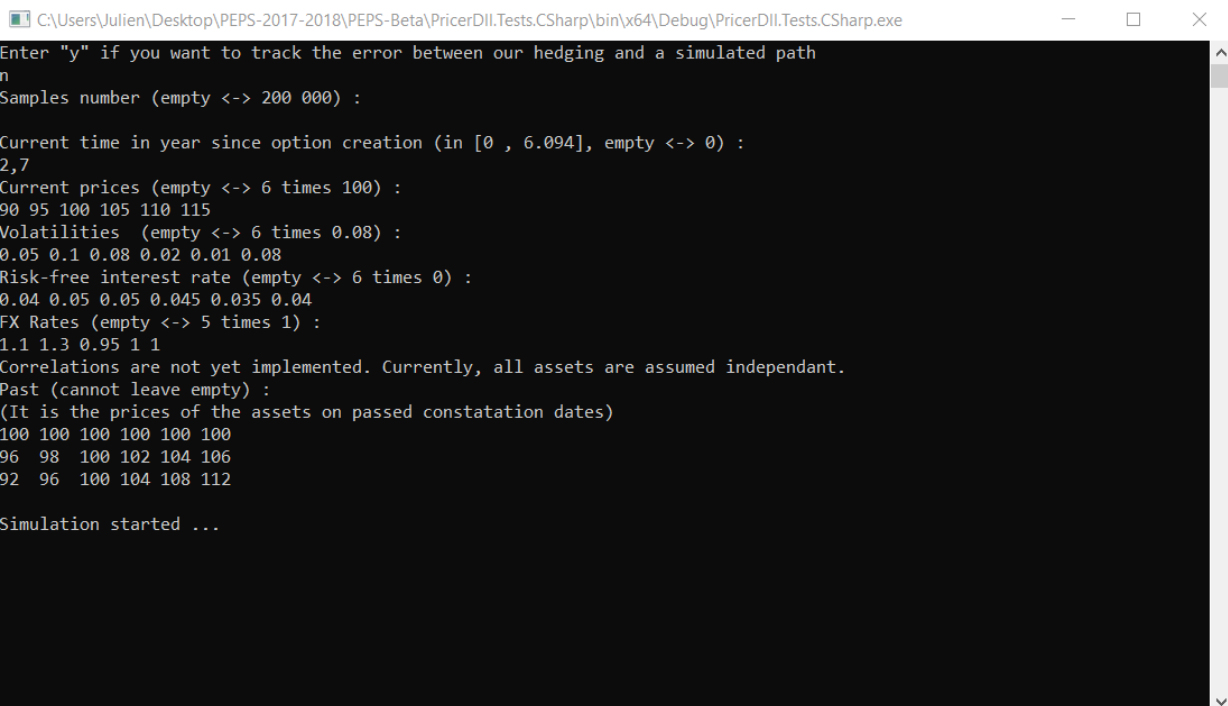
Cette application consiste donc en une boucle infinie de demandes d'entrées console, d'appels au Pricer, et de présentation de résultats.

Au démarrage de l'application, vous aurez le choix, par le biais d'une entrée de touche, entre une simulation de couverture ("y") et les appels standards au Pricer.

Les tableaux doivent être rentrés avec des valeurs séparées par des espaces. La virgule des nombres peut être un point ou une virgule.

Le temps t sera initialement demandé. On considère que 0 est la date de constatation initiale de l'actif, c'est à dire le 1^{er} octobre 2015. 1 signifie alors le 1^{er} octobre 2016. Le Multimonde ayant des dates de constatations espacées de 371 jours, le paramètre t est donc inclus entre 0 et 6,094. Dans le cas où un temps supérieur à 0 est rentré, il faudra rentrer la trajectoire jusqu'à présent. L'application déterminera seule en fonction du temps rentré le nombre de lignes à décrire. La première ligne est la ligne en temps 0 (typiquement 100 6 fois), tandis que la 2ème est celle en date 1,016, c'est à dire le 7 octobre 2016.

Un exemple d'utilisation de l'application console interactive est donné ci-dessous :



```
C:\Users\Julien\Desktop\PEPS-2017-2018\PEPS-Beta\PricerDII.Tests.CSharp\bin\x64\Debug\PricerDII.Tests.CSharp.exe
Enter "y" if you want to track the error between our hedging and a simulated path
n
Samples number (empty <-> 200 000) :

Current time in year since option creation (in [0 , 6.094], empty <-> 0) :
2,7
Current prices (empty <-> 6 times 100) :
90 95 100 105 110 115
Volatilities (empty <-> 6 times 0.08) :
0.05 0.1 0.08 0.02 0.01 0.08
Risk-free interest rate (empty <-> 6 times 0) :
0.04 0.05 0.05 0.045 0.035 0.04
FX Rates (empty <-> 5 times 1) :
1.1 1.3 0.95 1 1
Correlations are not yet implemented. Currently, all assets are assumed independant.
Past (cannot leave empty) :
(It is the prices of the assets on passed constatation dates)
100 100 100 100 100 100
96 98 100 102 104 106
92 96 100 104 108 112
Simulation started ...
```

```
C:\Users\Julien\Desktop\PEPS-2017-2018\PEPS-Beta\PricerDll.Tests.CSharp\bin\x64\Debug\PricerDll.Tests.CSharp.exe
Prix Multimonde : 117,896144535128
Intervalle de confiance Multimonde : 0,126763428560232

Deltas intermédiaires (indicatif) :
0,921658297770089
0,727072004221966
0,786306180433315
0,739803362457056
0
0

Nombre d'actifs à acheter :
0,921658297770089
0,660974549292696
0,604850908025627
0,778740381533743
0
0

Quantité de monnaie étrangère à acheter :
-62,7925821828062
-60,4850908025627
-81,767740061043
0
0

Euros restants à mettre au taux sans risque européen :
34,9468977358204

==== New entry ====
```

Les corrélations sont en fait implémentées, mais actuellement non modifiables de façon interactive.

8.1.3 PricerDll.CustomTests

PricerDll.CustomTests est une **application console C#** non interactive. Ce projet contient les tests pré-configurés du projets, destinés à vérifier que le Pricer fonctionne sur des options simples, puis à servir de tests anti-régression.

Ces tests comparent les prix et les deltas trouvés par le Pricer aux prix et aux deltas fournis par les formules fermées disponibles pour des options simples.

Actuellement, 4 tests sont lancés automatiquement au lancement de l'application PricerDll.CustomTests. Tous portent sur le Call Vanille sur un sous-jacent, c'est à dire une forme spécifique du Call sur un panier d'options. Sont testés :

1. Le prix en 0
2. Le prix en t
3. Le delta en 0
4. Le delta en t

Les valeurs en t sont arbitraires et non-extrêmes, pour ne pas représenter de cas particulier.

Les tests demandent une différence de prix ou de delta de moins de 2% avec le prix donné par la formule théorique. Le nombre de tirages est de 1 million par test.

Ce projet contient également la logique de récupération de données et d'estimation du prix basée sur des valeurs historiques, fonctionnelle en quasi-totalité, à l'exception que les taux sans risques ne sont pas encore récupérés, or ils sont nécessaires à l'estimation du prix sous probabilité risque-neutre.

Li'appel commenté renvoie par conséquent une valeur fausse du fait d'utiliser les tendances des actifs et non le taux sans risque du marché associé.

8.1.4 PEPS-Beta et PEPS-Beta.Tests

Le projet PEPS-Beta a été créé en vue de l'utilisation ultérieure du pricer via une application Web. Il consiste en l'interfaçage du moteur de calcul avec une page Web en Asp.net. Actuellement, ce projet est capable de faire appel à une fonction de pricing et l'afficher sur une page vide. Ce projet a été mis en pause à l'approche du rendu Beta, afin de se concentrer sur le moteur de calcul. Suite aux modifications effectuées sur le pricer, il est possible que des bugs surviennent sur les pages générées. Il n'est pas conseillé actuellement de chercher à utiliser le projet.

Le projet PEPS-Beta.Tests correspond au projet de tests unitaires associés au projet précédent. Il est actuellement vide, puisque PEPS-Beta n'est qu'un prototype de départ. Dans la suite, il servira à effectuer des tests sur l'application web et les appels aux moteurs de calculs sur celle-ci.

8.2 Instructions d'utilisation

Une fois le dossier téléchargé, avant de lancer PEPS-Beta.sln avec Visual Studio 2017 et choisir le projet à démarrer (soit PricerConsole, soit PricerDll.CustomTests), une action est nécessaire : il faut ajouter le chemin vers le dossier lib de la pnl utilisée (1.9.2) au PATH de l'ordinateur.

Pour lancer le projet ASP.NET, il est nécessaire que le debugger ASP.NET soit réglé en 64 bits, ce qui est spécifique au Visual Studio de l'ordinateur. Pour changer cela, Aller dans Outils/Tools > Options > Solutions et Projets / Projects and Solutions > Projets Web / Web Projects > Utiliser la version 64 bits d'IIS Express pour les sites et les projets web / Use the 64 bit version of IIS Express for web sites and projects.

Il faut aller dans les propriétés de la solution (Clic droit Solution PEPS-Beta, en haut à droite de la configuration par défaut de Visual Studio), puis changer le projet de démarrage à **PEPS-Beta**.

8.3 Base de données financières

Les données financières nécessaires au projet ont été récupérées de différentes manières. Tous les indices à l'exception du FTSE100 ont pu être récupéré sur YahooFinance[1], en prenant les données sur 5 ans entre le 30/09/2010 et le 30/09/2015. Les données du FTSE100 ont été récupérée sur www.tr4der.com[2]. Un traitement a dû être effectué sur l'indice australien ASX200 car le fichier de données téléchargé présentait un problème de décalage de dates, avec des valeurs le dimanche et pas le vendredi. Ces fichiers sont ensuite lus par l'application et stockés dans le programme. Pour obtenir le même nombre de dates pour chaque indice, ce nombre variant notamment en fonction des jours fériés, toute absence de valeur à une date située entre le lundi et le vendredi a été comblée en prenant la valeur du jour précédent. Toutes les valeurs des indices ont été prises à la clôture.

Pour les données des taux de changes nous avons utilisés le site www.oxf.com[3]. Les données ont été prises sur 3 ans entre le 26/01/2015 et le 26/01/2018. Ces données prennent en compte les valeurs également le samedi et le dimanche.

Pour les actifs nous avons voulu prendre les valeurs jusqu'à la date de début du produit pour obtenir une simulation plus proche de la tendance au lancement du produit.

8.4 Fonctionnalités implémentées

Pour cette version, le projet affiche une interface console qui donne :

- la composition du portefeuille en quantité de sous-jacents
- la composition du portefeuille en quantité de zéro-coupon étranger
- la composition du portefeuille en quantité de zéro-coupon euro
- Le prix à une date donnée du produit
- le choix du nombre de simulations pour l'obtenir

9 Moteur de calcul

Le moteur de calcul correspond à l'implémentation du pricer pour le calcul du prix du produit tout au long de sa durée de vie et la construction de son portefeuille de couverture. Nous allons nous pencher sur l'architecture adoptée pour le calculateur avec les différentes méthodes implémentées.

9.1 Calcul des prix

C'est le modèle de BlackScholes constant qui a été adopté. On considère N actifs qui suivent chacun un modèle de BS de dimension 1 et corrélés entre eux. La dynamique de ces actifs s'écrit :

$$S_{i,t} = S_{0,i} \exp\left((r - \frac{\sigma_i^2}{2})t + \sigma_i W_t, i\right)$$

avec r le taux d'intérêt du marché domestique, $(\sigma_1, \dots, \sigma_d)$ sont les volatilités des indices, (W_1, \dots, W_d) vecteur de mouvements browniens corrélés. La matrice de corrélation $\Gamma_{i,j} = \rho$ pour $i \neq j$ et pour $i=j$, $\Gamma_{i,j} = 1$. En utilisant la factorisation de Cholesky de $\Gamma = L'L$ avec L triangulaire inférieure. La dynamique du sous-jacent s'écrit alors :

$$S_{d,t} = S_{d,0} \exp\left((r - \frac{\sigma_d^2}{2})t + \sigma_d L_d B_t\right)$$

avec L_d correspondant à la ligne d de la matrice L . On en déduit :

$$S_{d,t_{i+1}} = S_{d,t_i} \exp\left((r - \frac{\sigma_d^2}{2})(t_{i+1} - t_i) + \sigma_d \sqrt{(t_{i+1} - t_i)} L_d G_{i+1}\right)$$

avec $d \in \llbracket 1, N \rrbracket$

Pour simuler le processus S sur la grille (t_i) il suffit de savoir simuler des vecteurs gaussiens centrés de matrice de covariance identité.

9.2 Calcul du prix

Afin de calculer le prix du produit à l'aide du simulateur, en posant $\Phi(S_{t_0}, \dots, S_{t_N})$, où les (t_i) correspondent à une grille de dates de constatation. Soit r le taux d'intérêt instantané qu'on supposera constant. Le prix en $0 < t < T$ est tel que :

$$v(t, S_{t_0}, \dots, S_{t_N}) = \exp(-r(T-t))E(\Phi(S_{t_0}, \dots, S_{t_N}))$$

9.3 Prix en 0

En utilisant la formule précédente, on a le prix en 0 :

$$v(t, S_0) = \exp(-rT)E(\Phi(S_0, \dots, S_N))$$

On approche $v(0, S_0)$ en utilisant la méthode de Monte Carlo par l'estimateur :

$$\hat{v}_M(0, S_0) = \exp(-rT) \frac{1}{M} \sum_{i=0}^M \Phi(S_0, \dots, S_N)$$

où M correspond au nombre d'itérations de la méthode de Monte Carlo.

9.4 Prix en t

Pour déterminer le prix en t, il s'agit de passer par une espérance conditionnelle. Il s'agit alors d'écrire S_{t+u} à l'instant $t+u$ pour $u > 0$ en fonction de la valeur S_t à l'instant t et d'une quantité indépendante de F_t :

$$S_{t+u} = S_t \exp\left((r - \frac{\sigma^2}{2})t + \sigma(B_{u+t} - B_t)\right) = S_t s_u$$

où s_u est **indépendant** du passé jusqu'à t. Sa dynamique est donc la même que vu précédemment avec un brownien indépendant de la filtration F_t .

9.5 Calcul des deltas

Notre modèle de couverture est entièrement déterminé par les quantités d'actifs à détenir à chaque date t de rebalancement. On obtient ces quantités en prenant la dérivée du prix par rapport à l'actif. On l'approche par une méthode de différences finies. Ainsi, pour un produit contenant N sous-jacents, la quantité d'actifs "i" à détenir est :

$$\frac{\delta v(t, S_0, \dots, S_t)}{\delta S_{t,i}}$$

9.6 Simulation de la couverture

Pour estimer la qualité de notre moteur de calcul, nous faisons recours à un calcul de tracking error entre une trajectoire simulée et une couverture rebalancée de manière journalière depuis la date 0 jusqu'à la maturité du produit.

Afin de calculer cette tracking error, nous simulons une première trajectoire des actifs. À chaque

jour correspond alors 6 prix d'actifs et 5 valeurs de taux de change.

Un premier portefeuille initial est construit, à partir du prix en 0 et des deltas calculés en 0. Ensuite, chaque jour, le portefeuille est auto-financé et constitué des deltas nécessaires en actif risqués (indices sous-jacents et monnaies étrangères). L'argent excédentaire est alors placé sur le taux sans-risque domestique.

Une fois la maturité atteinte, la tracking error correspond à l'écart entre le payoff du produit sur la trajectoire premièrement simulée et la valeur du portefeuille.

9.7 Dates choisies

Le calcul du payoff de notre produit est basé sur des dates de constatations.

Pour le calcul du prix du produit en t , il s'agit de récupérer les valeurs des sous-jacents à cette date puis d'appliquer la formule du payoff.

9.8 Dates de rebalancement

Par défaut, on choisit de rebalancer au quotidien. A chaque date de rebalancement, on récupère les données simulées puis on génère le reste de la trajectoire, qui sera utilisé pour calculer le nouveau prix du produit mais aussi les deltas.

10 Tests et Validation

Pour valider le pricer, nous l'avons appliqué à un call Vanille pour lequel nous possédons une formule fermée.

10.1 Call Vanille

10.1.1 Prix en $t = 0$

L'idée est de valider le pricer en calculant le prix d'un Call Vanille à $t=0$, en utilisant le pricer et la formule fermée. Ce test est effectué dans le fichier C# **TestsCall.cs**. Il détermine l'erreur entre le prix issu de la formule fermée et le prix du pricer. On rappelle la formule fermée de Black-Scholes du prix d'une option Call de maturité T et de Strike K sur un actif S ne versant pas de dividende :

$$C(0,K,S,T) = S_0 N(d_1) - K B(t, T) N(d_0)$$

avec :

$$d_0 = \frac{1}{\sigma\sqrt{T}} \left[\ln\left(\frac{S_0}{K}\right) + \left(r + \frac{1}{2}\sigma^2\right)T \right]$$

$$d_1 = d_0 + \sigma\sqrt{T}$$

10.2 Prix en $0 \leq t$

Toujours dans le test C# **TestsCall.cs**, on calcule l'erreur entre le prix issu de la formule fermée et le prix du pricer en t . On utilise la formule fermée de Black-Scholes du prix d'une option Call de maturité T et de Strike K sur un actif S ne versant pas de dividende en t :

$$C(t,K,S,T) = S_t N(d_1) - K B(t, T) N(d_0)$$

avec :

$$d_0 = \frac{1}{\sigma\sqrt{T-t}} \left[\ln\left(\frac{S_0}{K}\right) + \left(r + \frac{1}{2}\sigma^2\right)(T-t) \right]$$

$$d_1 = d_0 + \sigma\sqrt{T-t}$$

10.3 Calcul des deltas à $t=0$ et $t \geq 0$

Enfin nous comparons les deltas obtenus par la formule fermée et ceux obtenus par notre pricer. On utilise la formule fermée suivante en 0 :

$$D(0,K,S,T) = N(d_1)$$

avec :

$$d_0 = \frac{1}{\sigma\sqrt{T}} \left[\ln\left(\frac{S_0}{K}\right) + \left(r + \frac{1}{2}\sigma^2\right)T \right]$$

$$d_1 = d_0 + \sigma\sqrt{T}$$

Puis en t :

$$D(t,K,S,T) = N(d_1)$$

avec :

$$d_0 = \frac{1}{\sigma\sqrt{T-t}} \left[\ln\left(\frac{S_0}{K}\right) + \left(r + \frac{1}{2}\sigma^2\right)(T-t) \right]$$

$$d_1 = d_0 + \sigma\sqrt{T-t}$$

Références

- [1] Données pour les indices :
<https://finance.yahoo.com/world-indices>
- [2] Données pour le FTSE100 :
<http://www.tr4der.com/historical-prices/%5EFTSE/>
- [3] Données pour les valeurs de changes :
<https://www.ofx.com/en-au/forex-news/historical-exchange-rates/>