

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

RAPPORT DE PROJET D'ÉVALUATION DE PRODUITS STRUCTURÉS

---

## Multimonde 2021

---

26 JANVIER 2018



Bernard Julien  
Berthon Christophe  
Herbadji Alexandra  
Moine Yoann  
Namasse Soufiane

*Enseignant : Ollivier TARAMASCO*

# Table des matières

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Cadre de l'étude</b>	<b>3</b>
<b>3</b>	<b>Description du produit MultiMonde 2021</b>	<b>3</b>
<b>4</b>	<b>Analyse des flux financiers</b>	<b>3</b>
4.1	Description des flux . . . . .	3
4.2	Définition mathématique des flux . . . . .	4
<b>5</b>	<b>Analyse des risques</b>	<b>5</b>
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
<b>6</b>	<b>Modèle Mathématiques</b>	<b>6</b>
6.1	Dynamique des sous-jacents . . . . .	6
6.2	Modélisation des taux de changes . . . . .	6
6.3	Modélisation des zéros coupons . . . . .	6
6.4	Modélisation de la volatilité . . . . .	7
6.5	Améliorations possibles . . . . .	7
<b>7</b>	<b>Portefeuille de couverture</b>	<b>7</b>
7.1	Couverture théorique . . . . .	8
7.2	Simulation du portefeuille de couverture . . . . .	9
<b>8</b>	<b>Architecture logicielle</b>	<b>9</b>
8.1	Architecture globale . . . . .	9
8.1.1	PricerDll . . . . .	10
8.1.2	PricerConsole . . . . .	10
8.1.3	PricerDll.CustomTests . . . . .	12
8.1.4	PEPS-Beta et PEPS-Beta.Tests . . . . .	12
8.2	Instructions d'utilisation . . . . .	12
8.3	Base de données financières . . . . .	13
8.4	Fonctionnalités implémentées . . . . .	13
8.5	Architecture du projet web . . . . .	14
8.6	Utilisation de l'outil web . . . . .	14
8.7	Extensions voulues mais non réalisées . . . . .	14

<b>9 Moteur de calcul</b>	<b>15</b>
9.1 Calcul des prix . . . . .	15
9.2 Calcul du prix . . . . .	15
9.3 Prix en $t=0$ . . . . .	16
9.4 Prix en $t$ . . . . .	16
9.5 Calcul des deltas . . . . .	16
9.6 Simulation de la couverture . . . . .	17
9.7 Dates choisies . . . . .	17
9.8 Dates de rebalancement . . . . .	17
<b>10 Tests et Validation</b>	<b>17</b>
10.1 Call Vanille . . . . .	17
10.1.1 Prix en $t=0$ . . . . .	17
10.2 Prix en $0 \leq t$ . . . . .	18
10.3 Calcul des deltas à $t=0$ et $t \geq 0$ . . . . .	18
10.4 Call Quanto . . . . .	19
10.4.1 Prix et delta $t$ . . . . .	19
<b>11 Remarques sur le fonctionnement du pricer</b>	<b>19</b>
11.1 Accélérations du pricer . . . . .	19
11.1.1 Initialisation de boucles de simulation . . . . .	19
11.1.2 Variables antithétiques . . . . .	20
11.2 Subtilités de la simulation des deltas . . . . .	21
11.3 Remarques sur la simulation de la tracking error . . . . .	22
11.3.1 Possibilités . . . . .	22
11.3.2 Façon de calculer . . . . .	22
11.3.3 Calibrage des dates de rebalancement . . . . .	22
11.3.4 Minorant de la tracking error . . . . .	22

# 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

Le produit MultiMonde est un produit structuré portant sur 6 indices boursiers, vivant sur des marchés boursiers différents :

- EuroStoxx 50
- S&P500
- Nikkei 225
- HANG Seng Index
- FTSE 100
- ASX200

Le fond débute le 01 octobre 2015 et paie à échéance uniquement le 18 novembre 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 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,  $\sigma_i$  la volatilité de l'actif, constante, et  $\mu_i$  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 entre la monnaie domestique (indiquée par  $d$  et une monnaie étrangère, indiquée par  $f$ ), on a la dynamique suivante :

$$X_{i,t} = X_{t,i}[(r_d - r_f)dt + \sigma_i dW_{d,i,t}]$$

Ainsi, sous le brownien étranger, cela donne :

$$X_{i,t} = X_{t,i}[(r_d - r_f + \sigma_x^2)dt + \sigma_i dW_{f,i,t}]$$

Tous les taux de change peuvent s'exprimer de manières similaires.

### 6.3 Modélisation des zéros coupons

Afin de permettre une utilisation directe des dynamiques simulées, il s'est trouvé intéressant de travailler avec la simulation des zéros-coupons étrangers convertis en €, notons-les de façon générique  $ZC_{f,i}$ , où  $f$  est la devise d'origine. Ainsi les dynamiques des 5 zéros coupons sont déterminées de la façon suivante :

Notons  $X_f$ , le taux de change qui fournit le prix d'une unité de devise étrangère en euros. Maintenant, considérons  $Z = F(X_f, ZC_f)$  où  $ZC_f$  correspond uniquement à la dynamique du zéro-coupon dans sa monnaie de référence. Alors la fonction  $F$  est telle que  $F : (X, ZC_f, t) \rightarrow X * ZC_f$  est mesurable sur l'espace des réels. On peut donc appliquer la formule d'Itô à  $Z$ . On a alors :

$$\begin{aligned} Z_t &= \frac{\partial F(X, ZC, t)}{\partial X} dX + \frac{\partial F(X, ZC, t)}{\partial t} dt + \frac{\partial F(X, ZC, t)}{\partial ZC} dZC + \frac{(\partial)^2 F(X, ZC, t)}{(\partial X)^2} d\langle X \rangle \\ &= ZC_f dX + 0 + X dZC + 0 \\ &= ZC_f X [(r - r_f)dt + \sigma_x dW_t] + X ZC(r_f dt) \end{aligned}$$

D'où la dynamique du zéro-coupon étranger en euros :

$$dZ = Z(r dt + \sigma_x dW_t)$$

Procéder en utilisant les zéros-coupons convertis en euros c'est se garantir de récupérer des proportions de couvertures exactes directement sans procéder à des opérations de change supplémentaires.

## 6.4 Modélisation de la volatilité

Le modèle à volatilité historique est celui qui est actuellement adopté. Cette volatilité historique est déterminée en déterminant la log-rentabilité et en déterminant l'estimateur de sa volatilité l'opération suivante sur les données des sous-jacents :  $r_{i,j} = \log\left(\frac{S_{i,j}}{S_{i,j-1}}\right)$  où  $r_{i,j}$  correspond à la rentabilité logarithmique du titre  $i$  entre les dates  $j$  et  $j-1$ . L'estimateur  $s$  de la volatilité annuelle est le coefficient diagonale de la matrice de covariance multiplié par  $\sqrt{252}$ .

$$s = \sqrt{\sum_{i=1}^n \frac{1}{n-1} (R_{i+1} - E) * (R_{i+1} - E)}$$

avec  $E = \sum_{i=1}^n \frac{1}{n} R_i$

## 6.5 Améliorations possibles

Nous nous sommes renseignés sur les possibles améliorations du modèle, notamment sur l'utilisation d'une volatilité stochastique. En effet, il se trouve qu'une volatilité historique n'est pas un facteur constant. Il existe ce qu'on appelle le smile de volatilité, c'est à dire une variation de la volatilité avec le Strike et la maturité. Cela n'est pas pris en compte dans le modèle de volatilité historique. Il faudra donc adopter un modèle de volatilité stochastique.

## 7 Portefeuille de couverture

Le projet permet de réaliser un portefeuille de couverture pour se couvrir contre les risques de change et de marché auxquels est soumis les Multimonde.



## 7.1 Couverture théorique

On se place dans un espace de probabilité filtré  $(\Omega, \mathcal{F}, P)$ , et un mouvement brownien standard  $W = W_t$  tel que  $F = \sigma(W)$ . On considère un actif risqué, un horizon temporel  $T > 0$  fini, et un taux sans risque  $r > 0$  et constant. De plus, l'actif risqué  $S_t$  suit un mouvement brownien géométrique de dynamique :

$$S_t = \mu S_t dt + \sigma S_t dW_t$$

Construisons maintenant la couverture mathématiquement déterminée. Soit  $\delta_{0,t}$  la proportion d'actif sans risque  $S_0$  et  $\delta_t$  la proportion d'actif risqué  $S$ . On note alors la valeur du portefeuille à l'instant  $t$ , avec  $0 \leq t \leq T, V_t$  :

$$V_t = \delta_{0,t} dS_{t,0} + \delta_t dS_t$$

Notons le flux final :

$$\mathbb{F}_T = g(T)$$

La valeur actualisée du portefeuille se note :

$$\tilde{V}_t = \frac{V_t}{S_{t,0}}$$

On note également l'actif actualisé :

$$\tilde{S}_t = \frac{S_t}{S_{t,0}} = e^{-rt} S_t$$

On a alors, en appliquant la formule d'Itô :  $d(\tilde{V}_t) = \delta_t d[\frac{S_t}{S_{t,0}}]$

Appliquons la formule d'Itô à l'actif actualisé, qui est un processus d'Itô :

$$\tilde{S}_t = \frac{dS_t}{S_{t,0}} + S_t \frac{1}{S_{t,0}} + d < S_t, \frac{1}{S_{t,0}} = \tilde{S}_t [(\mu - r)dt + \sigma dW_t]$$

Posons maintenant :

$$\lambda = \frac{\mu - r}{\sigma}$$

On reconnaît la prime de risque qui va nous permettre d'introduire une probabilité équivalente à  $P$  en utilisant le théorème de Girsanov. Et on définit  $Q$  la probabilité risque-neutre, qu'on suppose unique, localement équivalente à  $P$  par :

$$\frac{dQ}{dP} = e^{-\frac{\lambda^2 T}{2} - \lambda W_T}$$

Posons alors

$$\widetilde{W}_t = W_t + \lambda t$$

Ainsi l'actif actualisé, sous  $Q$  a pour dynamique :

$$d\tilde{S}_t = \sigma \tilde{S}_t d\widetilde{W}_t$$

Sous  $Q$ ,

$$\widetilde{W}_t = W_t + \lambda t$$

est un mouvement brownien. Ainsi, sous  $Q$ ,

$$\tilde{S}_t = \sigma \delta_t \tilde{S}_t d\widetilde{W}_t$$

La condition de Novikov est à vérifier : sur  $[0,T]$  l'actif est un processus continu sur un intervalle fini, donc uniformément continu, donc borné. Ainsi :

$$E_Q[\int_0^T |\delta_t \tilde{S}_t|^2 dt] \leq$$

Dès lors, on a la valeur du portefeuille à l'instant  $t$  :

$$\mathbb{V}_t = v(t, S_t) = e^{-r(T-t)} E_Q[g(S_T)|F_t]$$

Supposons l'existence de cette probabilité risque neutre. Alors le portefeuille peut être répliqué. Notons ce portefeuille autofinancé  $P_t$  qui réplique la valeur de  $v(t, S_t)$ . Alors :

$$\mathbb{P}_t = \phi_{t,0} S_{t,0} + \phi_t S_t$$

$$P_t = dv(t, S_t)$$

On a par unicité :

$$\phi_t = \frac{\partial v}{\partial S}$$

$$\phi_{0,t} = \frac{v(t, S_t) - \phi_t S_t}{S_{t,0}}$$

Ce principe de couverture est celui de "delta hedging". Il s'agit de se couvrir en suivant exactement les mêmes variations que l'actif.

## 7.2 Simulation 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.

La composition du portefeuille de couverture comprend, avant la première date de constatation tous les indices du produit. Puis, à chaque date de constatation on enlève du portefeuille celui qui a le mieux performé.

# 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.

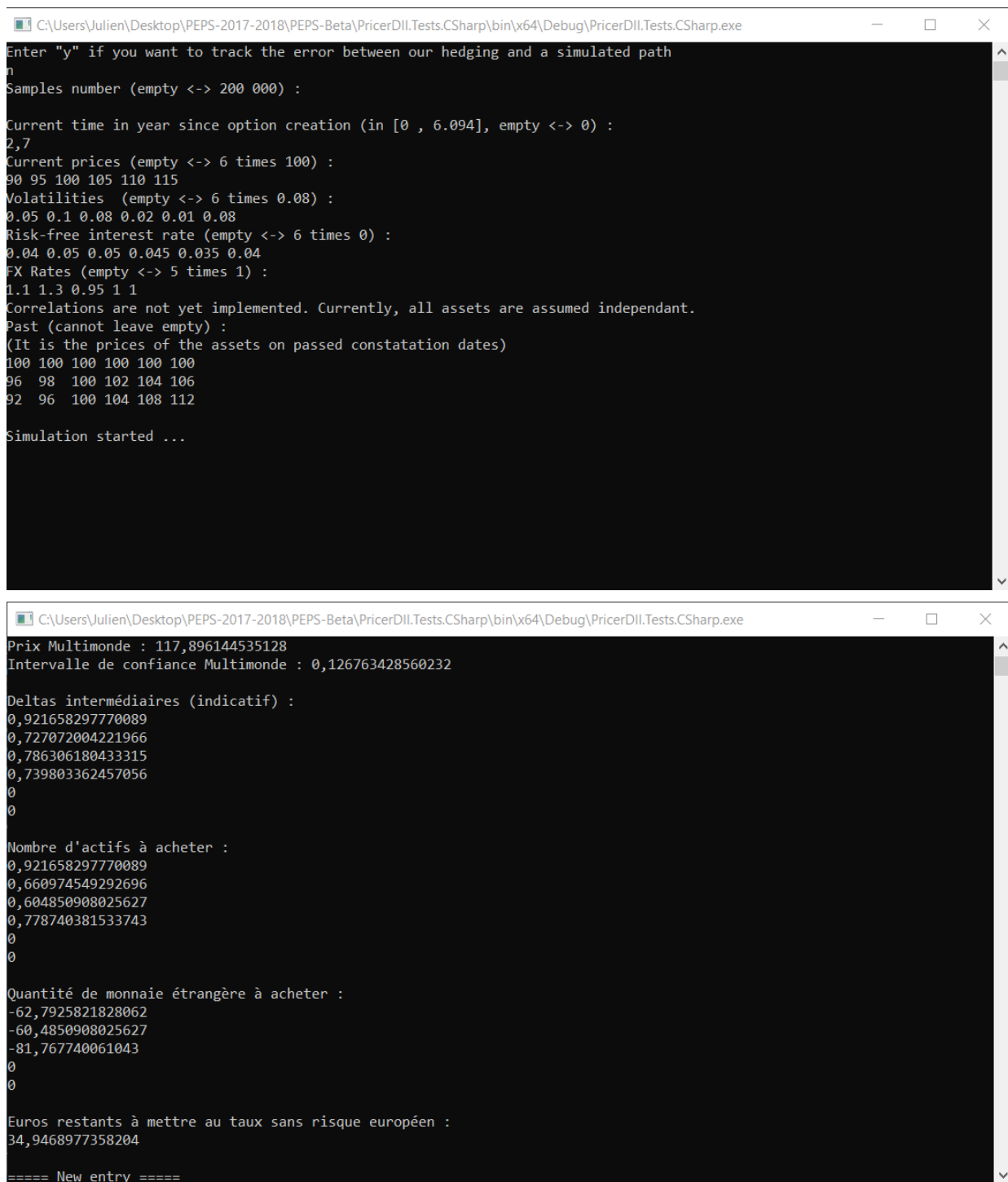
Cette application consiste donc en une boucle infinie de demandes d'entrées console, d'appels au Pricer, et de présentation de résultats.

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<sup>er</sup> octobre 2015. 1 signifie alors le 1<sup>er</sup> 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 ...

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

Initialement le projet utilisait des données financières contenues dans des fichiers CSV. Cette solution présentait plusieurs défauts. Tout d'abord elle demandait à l'utilisateur de rajouter quotidiennement les données du jour aux fichiers. Ensuite à chaque démarrage de l'application les données étaient parsées, puis stockées dans une classe. Cette opération présentait un coût conséquent d'opérations, et bloquait en plus de la mémoire du processus. Enfin l'accès à une donnée particulière était difficile.

Nous avons donc mis en place une base de données peuplées par des requêtes WEB. Pour parer aux absences de valeurs, nous avons décidé de remplacer les données manquantes par la dernière donnée présente de l'indice. Cette opération génère un coût conséquent lors de l'initialisation de la base de données. Cependant une fois l'initialisation faite, l'utilisateur n'a plus qu'à se servir des fonctions de mises à jour pour remplir la base de données en un temps très faible. Le deuxième avantage de notre implémentation est la facilité d'accès aux données.

La base de données se compose des tables suivantes :

- TauxDeChanges : contient les valeurs des taux de change
- IndexesAtDates : contient les valeurs des marchés
- Indices : contient des informations propres à chaque indice (volatilité, corrélation,...)
- PortefeuilleIdeals : contient les deltas du portefeuille idéal
- PortefeuilleCouvert : contient les deltas de la couverture du Multimonde
- MultiMondeParam : contient le prix, la valeur du profitLoss, le nombre d'échantillon, d'indices et les dates origine, actuelle et fin

Les accès à la base de données se font par requête SQL, sur une base de données locale SQLServerExpress.

### 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

## 8.5 Architecture du projet web

La partie web de l'application de gestion du produit consiste en une application Modèle-Vue-Contrôleur ASP.Net. La vue principale est découpée en vues partielles fortement typée associées aux modèles. Ce découpage est un point fort de l'application, car cela permet de rapidement ajouter de nouvelles fonctionnalités. Une fois que la structure est comprise, cela permet de répartir des tâches indépendantes rapidement.

Le modèle a été explicité plus haut. Les échanges entre modèle et vue sont effectués au travers du principal HomeController. Lorsqu'un échange est nécessaire avec la base de données, tout est fait au travers d'une classe DAL (Data Access Layer), qui permet d'éviter tout effet de bord sur la base de données. Le contrôleur associés aux vues fortement typés permet de contrôler toute entrée sur l'application web.

L'échange de données entre la vue et les contrôleurs est exclusivement effectué au travers de formulaires Ajax. Bien que cela soit parfois de trop, cela permet de ne pas avoir à rafraîchir la page à chaque envoi/réception de données et de limiter la bande passante de l'utilisateur. Cependant, l'omniprésence de formulaires Ajax pousse à un grand nombre de boutons, et d'appels asynchrone. L'application n'est actuellement pas robuste, et ne supporterait pas un grand nombre de clics et d'appels aux contrôleurs. Il est fortement conseillé d'attendre la fin de chaque requête Ajax avant d'effectuer toute nouvelle requête. Une barre de chargement est visible pour chaque requête non instantanée. De plus, l'application ne fonctionne actuellement que pour une utilisation simultanée. L'utilisation de l'application par 2 utilisateurs provoque actuellement un comportement non prévisible, et est fortement déconseillée. Il est cependant possible de fermer la page et la rouvrir plus tard, ou bien couper l'application et la relancer plus tard.

## 8.6 Utilisation de l'outil web

- Au premier lancement de l'application, la BDD locale se crée, et est initialisée au jour de fermeture du produit.
- Les paramètres liés aux indices n'ont pas encore été calculés. Pour ce faire, il faut sélectionner un intervalle de temps. Une fois ceci-fait, on peut ré-estimer des paramètres, mais ils sont conservés en BDD jusqu'à ce qu'ils soient re-modifiés.
- Pour créer le portefeuille de couverture, il faut calculer le portefeuille optimal, et cliquer sur "Initialisation". Ce bouton permet de récupérer le portefeuille optimal à tout instant.
- On peut ensuite naviguer modifier la "date actuelle" utilisée par le moteur de calcul. On ne peut naviguer qu'entre le 21 octobre 2015 Afin de voir l'évolution du portefeuille sur 1 jour, on peut par exemple placer la date au 2 Octobre 2015.
- Si l'on souhaite ré-organiser le portefeuille de couverture pour qu'il soit adapté à l'état actuel du marché, il faut recalculer le portefeuille optimal, et cliquer sur "Mettre à jour". Cela a pour effet de re-composer le portefeuille de couverture avec l'argent disponible.

## 8.7 Extensions voulues mais non réalisées

Par manque de temps, plusieurs fonctionnalités n'ont pas été réalisées. En voici la liste.

- Pouvoir se déplacer dans le temps avec une fréquence de rebalancement spécifiée. La fonctionnalité est présente et fonctionnelle (testée) dans le moteur de calcul, mais pas interfacée avec l'application Web.
- Afficher les performances passées et enregistrées des indices, afin que le gérant puisse avoir un visuel clair et rapide de l'état du produit.

- Simuler une trajectoire historique des indices et taux de change du marché jusqu'en novembre 2021, afin de pouvoir se déplacer jusqu'à échéance du produit, et avoir accès à un suivi complet de PnL sur la durée totale du produit. Seule la partie côté Web manque pour cette fonctionnalité.
- Afficher sous forme de graphiques les variations des indices, taux de change et PnL.
- Afficher de manière plus ergonomique la composition du portefeuille de couverture.

## 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 Black-Scholes 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 :

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

On calcule le prix par une méthode de Monte-Carlo.



### 9.3 Prix en t= 0

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

$$v(0, S_0) = \exp(-rT) E(\Phi(S_0, \dots, S_N))$$

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 :

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

ce qui correspond à l'estimateur suivant :

$$\frac{e^{-r(T-t)}}{M 2 s_t h} \sum_{i=0}^M \phi(s_{t0}, \dots, s_{t_i}, s_t(h+1)S_{t_{i+1}}, \dots, s_t(1+h)S_{t_{N-t}}) - \phi(s_{t0}, \dots, s_{t_i}, s_t(1-h)S_{t_{i+1}}, \dots, s_t(1+h)S_{t_N})$$

## 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

La validation du pricer est effectuée au travers d'un projet de tests automatiques. Ces tests couvrent les fonctionnalités de pricing et de hedging sur différents produits :

- Call Vanille
- Call Quanto

De plus, nous avons pricé le produit MultiMonde dans des états qui nous permettaient de prévoir des résultats.

## 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 :

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

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

Les tests correspondants aux prix des Call Vanille en 0 et en t fonctionnent et sont tous validés.

## 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 :

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

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

Les tests des deltas du Call Vanille ne passent pas la validation. En effet, ils utilisent des fonctions du pricer dépréciés. Le débog et mise à jour de ces fonctions n'a pas été une priorité, car le pricer fournissait des valeurs cohérentes de pricing et de hedging pour les Calls Quanto.

## 10.4 Call Quanto

### 10.4.1 Prix et delta t

On rappelle la formule fermée de Black-Scholes du prix d'une option Call Quanto de maturité  $T$ , et de Strike  $K$  sur un actif  $S$  ne versant pas de dividende, sur un marché étranger, avec  $X$  symbolisant le taux de change entre la monnaie domestique, à la date  $t$ . On note  $r^d$  ce qui correspond au taux d'intérêt domestique, et  $r^f$  ce qui correspond au taux d'intérêt étranger.

$$C(t, K, S, X, T) = S_t \exp [-(r^d - r^f - \rho \sigma_X \sigma_S)(T - t)] N(d_1) - K B^d(t, T) N(d_0)$$

avec :

$$d_0 = \frac{1}{\sigma \sqrt{T-t}} [\ln(\frac{S_t}{K}) + (r^f + \rho \sigma_X \sigma_S + \frac{1}{2} \sigma_X^2)(T - t)]$$

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

Les deltas correspondants :

$$\Delta_S = \frac{1}{X_t} \exp [-(r^d - r^f - \rho \sigma_X \sigma_S)(T - t)] N(d_1)$$

$$\Delta_{B^d} = S_t \exp [r^f + \rho \sigma_X \sigma_S)(T - t)] N(d_1) - K N(d_0)$$

Nous comparons les valeurs théoriques de ces formules fermées avec les valeurs retournées par le pricer, dans l'intervalle de confiance associé. Des tests en 0 et à un instant  $t$  sont réalisés.

Voici la dynamique proba risque-neutre domestique utilisée pour la diffusion des actifs, en notant  $S^d$  l'actif étranger vu en monnaie domestique, et  $B^{fd}$  le Zéro-Coupon étranger vu en monnaie domestique.

$$\frac{dS_t^d}{S_t^d} = r^d * dt + \sqrt{\sigma_{S^f}^2 + \sigma_X^2 - 2\rho\sigma_{S^f}\sigma_X} dW_t^0$$

$$\frac{d(B^{fd}/S_t)}{S_t} = r^d * dt + \sigma_X dW_t^1$$

En corrélant les deux mouvements Brownien :

$$\text{corr}(W_t^0, W_t^1) = -\frac{\rho\sigma_{S^f}\sigma_X}{\sqrt{\sigma_{S^f}^2 + \sigma_X^2 - 2\rho\sigma_{S^f}\sigma_X}}$$

Le pricer retourne en réalité la quantité  $\Delta_{B^f}$ . Nous transformons donc les quantités théoriques précédentes pour comparer à cette valeur. Tous les tests correspondants fonctionnent avec succès.

## 11 Remarques sur le fonctionnement du pricer

### 11.1 Accélérations du pricer

#### 11.1.1 Initialisation de boucles de simulation

Pour améliorer la vitesse du pricer, l'appel à une boucle de simulation de trajectoires est séparés en deux fonctions : une fonction d'initialisation à appeler (initAsset) une fois, et la fonction de génération d'une trajectoire (postInitAsset), appellable en boucle dès lors que la fonction d'initialisation a été appelée.

Ceci permet de ne calculer qu'une seule fois la matrice de corrélations, et de n'allouer qu'une seule fois la matrice de bonne taille qui servira à la génération des browniens. La matrice recevant la trajectoire est également allouée une seule fois, mais dans la fonction appelante. Ces améliorations permettent d'économiser sur le temps d'allocation et de désallocation mémoire.

### 11.1.2 Variables antithétiques

La technique de la variable antithétique dans le cadre du Monte-Carlo consiste à, lors de l'estimation de l'espérance de  $E[f(X)]$  avec  $X$  une variable aléatoire par Monte-Carlo, simuler une portion de l'échantillon de façon identiquement distribué par rapport à la loi de  $X$ , mais en renonçant à l'indépendance avec le reste de l'échantillon, l'idée étant de construire cette portion de l'échantillon telle que sa covariance avec le reste de l'échantillon soit négative.

D'un point de vue purement qualitatif, le Multimonde rapporte d'autant plus que les browniens intervenant dans la simulation de la trajectoire sont positifs (c'est à dire, que les sous-jacents gagnent de la valeur). J'ai donc intuité que la covariance du payoff associé à une matrice de brownien avec le payoff associé à l'opposé de cette matrice devrait être négative, de plus, l'échantillon des matrices "opposés" reste indépendant entre eux et identiquement distribué par rapport à la loi qui les génère (notamment au niveau des covariances). Il ne restait plus qu'alors à tester s'il y avait, ou non, accélération, c'est à dire si la covariance de l'ensemble de l'échantillon des matrices opposés est effectivement négativement corrélé à l'échantillon initial.

On trouve une réduction moyenne de l'écart-type lors du calcul du prix du Multimonde de :

- -12,7% pour 100 tirages
- -2,4% pour 1000 tirages

L'assomption est vérifiée, cependant l'impact de l'utilisation de variables antithétiques sur l'écart-type décroît rapidement avec le nombre de tirages. Ce dernier tend vers 0 selon mes mesures. Or, pour obtenir des prix précis en pratique, un million de tirage est un minimum ; dans notre cas, les variables antithétiques n'ont donc pas d'intérêt vis-à-vis de la réduction de la variance.

Cependant, ne pas faire augmenter la variance est suffisant pour justifier l'utilisation de variables antithétiques, car le calcul de l'opposé d'une matrice de brownien est moins coûteux que d'en générer une nouvelle.

Ainsi, en mesurant les temps d'exécution en millisecondes des appels non managés à l'estimation du prix du Multimonde 2021 dans les tests associés, avec un million de tirages, on obtient les résultats suivants :

Sans	Avec
15516	14750
14575	12251
14937	12809
15735	14154
4242	3909
4223	3934
4150	4013
15326	13317
15364	13201
15546	13521

On obtient donc une réduction moyenne du temps d'exécution de **-10,34%** pour des appels importants lors de l'utilisation de variables antithétiques.

## 11.2 Subtilités de la simulation des deltas

On se réfère par la suite aux actifs dont les performances ont été enregistrés par la formule de calcul du payoff du Multimonde 2021 à un stade d'une trajectoire comme *actifs sélectionnés*.

Le paramètre "h" est le shifting relatif dans le calcul des deltas. Il a un impact négatif quand poussé aux extrêmes, **dans les deux sens**.

Dans le cas d'un paramètre h trop grand, l'explication est simple : l'impact de la "barrière" des -15% et 15%, c'est-à-dire l'arrêt de l'augmentation ou de la diminution du payoff quand le sous-jacent s'éloigne trop de sa valeur d'origine, grandit alors que ces dernières n'ont théoriquement pas d'impact sur les deltas. Un h de 0,15 est absurde dû au fait qu'il atteint directement les barrières depuis la valeur initiale. h devrait idéalement être infime vis-à-vis de ces dernières.

L'explication inverse est plus subtile. Le cas classique d'impact du shifting de la trajectoire par h est que l'actif shiftée a une valeur shiftée par h quand il est "sélectionné". Dans ce cas, l'impact de h sur le prix est linéaire. Cependant il est possible que l'impact de h soit non linéaire : ceci se produit dans le cas où **l'ordre de sélection des actifs est modifié par le shifting**. Le shifting peut causer l'inversion de l'ordre de sélection entre deux (ou plus) actifs, ce qui a un impact fixe, indépendant de h (shifting de la valeur d'un des deux actifs à part), dû à la différence de trajectoire de ces actifs entre la date de sélection du premier de ces actifs, et celle de l'autre. Or cet impact fixe est alors amplifiée par la division par h.

Réduire la valeur de h a donc pour conséquence d'augmenter l'impact de ces occurrences non h-linéaires, mais elle a aussi pour conséquence d'en augmenter la rareté. Dans une optique de compensation de ces occurrences non h-linéaires entre elles, la valeur de h est donc doublement inversement linéaires à l'impact global de ces dernières.

Avec  $h=0.0001$ , même pour 100.000 tirages, on peut obtenir constamment des deltas de 0,15 à 0,45 pour une vraie valeur de deltas à 0,3. Le choix de la valeur de h est donc **la clé de la précision du calcul des deltas de ce produit**.

L'expérimentation m'a conduit à régler une valeur de h à 0,005.

## 11.3 Remarques sur la simulation de la tracking error

### 11.3.1 Possibilités

La fonction exposée de simulation de couverture permet actuellement les options suivantes : - Il est possible de fournir un passé, et il est possible de fournir un scénario partiel et ne simuler qu'à partir de la fin de ce scénario, même si ce scénario commence après la date de début de couverture. - Il est possible de spécifier qu'on ne simule la couverture que sur une partie des 6 et quelques années qui composent la fenêtre temporelle du Multimonde. - A noter que le scénario fourni doit fournir des prix pour chaque date de rebalancement, mais pas nécessairement pour chaque date de constatation. Si ces dernières ne coïncident pas, les prix seront interpolés. - La fonction renvoie la liste des différences de rendements du portefeuille. L'appelant peut alors l'utiliser pour calculer la PL de stratégies concrètes qui auraient été effectuées dans une optique de couverture.

### 11.3.2 Façon de calculer

La tracking error est calculé sur la différence des logarithmes des rendements.

De surcroît, nous mensualisons les rendements avant de calculer la tracking error (ce qui revient à multiplier le logarithme par la bonne valeur). Ceci est nécessaire pour pouvoir comparer des tracking error de fréquence de rebalancement différentes. Nous reviendrons sur ces comparaisons par la suite.

$$\text{C'est donc } \frac{1}{N} \sum_0^{N-1} \frac{r_{produit}}{r_{portefeuille}}$$

### 11.3.3 Calibrage des dates de rebalancement

Importance de faire coïncider les dates de constatations à des dates de rebalancement Il est fortement bénéficiaire à la tracking error de rebalancer le portefeuille aux dates de constatation. La raison à cela étant que le shifting étant fait à partir de la date de rebalancement suivant celle à laquelle on rebalance, si une date de constatation se trouve entre ces deux dates de rebalancement, la quantité d'actif en portefeuille entre la première date de constatation et la date de rebalancement sera erronée au sens où le sous-jacent qui s'apprête à être sélectionné a encore un impact relatif à ses variations fort sur le payoff.

Ainsi, par exemple, à paramètres identiques par ailleurs (5000 tirages pour le prix ; 500 pour les deltas ; simulation sur toute la durée ; pas de scénario fournie, monde standard, volatilités à 2%), on trouve une augmentation de 20% (pas points) de la tracking error lorsqu'on rebalance 52\*6 fois (synchronisation parfaite) ou 52\*6-3 fois (3 dates de constatations qui ne coïncident pas) :

312 rebalancements : 0,203% de tracking error. 309 rebalancements : 0,258% de tracking error.

### 11.3.4 Minorant de la tracking error

Il m'est apparu en travaillant sur la tracking error que lors de la simulation de cette dernière qu'en simulant un scénario, puis en calculant de façon répétée le prix, les deltas, et la valeur du portefeuille de couverture à la date de rebalancement suivante par rapport à celle du produit, l'estimation de la tracking error était sujette à un bruit très important : l'erreur d'estimation du prix.

En effet, le rendement (non mensualisé) du produit est linéaire à la période de rebalancement, tandis que l'écart-type relatif de l'estimation du prix ne l'est pas. La conséquence de ceci est que, même en supposant que le portefeuille de couverture suive parfaitement les variations du produit ; alors le terme mesuré (auquel nous appliquerons un logarithme, une mensualisation, puis calculerons l'écart-type de la série) est

$$\begin{aligned} & ((P + a) * x - (P + a)) - ((P + b) * x - (P + a)) \\ &= P * x + a * x - P - a - P * x - b * x + P + a \\ &= (a - b) * x \end{aligned}$$

En notant  $P$  le prix à une date de constatation,  $a$  la différence telle que  $P + a$  soit le prix estimé à cette date,  $x$   $1 +$  le rendement entre le vrai prix à la date de constatation suivante et  $P$ , et  $b$  la différence telle que  $(P + b) * x$  soit le prix estimé à la date de constatation suivante.

Le tracking error est donc bruitée au minimum par cette série des  $(a - b) * x$ . L'écart-type de cette série n'est pas cependant un minorant puisque les erreurs de couverture peuvent compenser accidentellement les erreurs d'estimation de telle sorte à réduire la tracking error.

Ci dessous sont certaines des tracking error observées et l'écart-type relatif moyen d'estimation du prix durant ces couvertures (couvertures totales, monde basique, volatilités 2%) :

**371\*6 rebalancements (journaliers); 5 000 tirages de prix, 500 tirages de deltas**

Tracking error mensuelle	1,33522%
Ecart-type relatif mensuel moyen d'estimation du prix	1,6130%

**36\*6 rebalancements (tous les 10 jours), 50 000 tirages de prix, 20 000 tirages de deltas**

Tracking error mensuelle	0,0571%
Ecart-type relatif mensuel moyen d'estimation du prix	0,0502%

**12\*6 rebalancements ( mensuel), 200 000 tirages de prix, 20 000 tirages de deltas**

Tracking error mensuelle	0,0442%
Ecart-type relatif mensuel moyen d'estimation du prix	0,0072%

On remarque sur ces valeurs que la tracking error mensuelle pour un rebalancement journalier est énorme par rapport à ce qu'on peut voir pour les autres fréquences de rebalancement. En particulier, bien qu'elle soit inférieure à l'écart-type mensuel moyen d'estimation du prix, on peut inférer qu'elle y est totalement due : les variations d'un jour sur l'autre de la valeur de l'actif et du portefeuille sont extrêmement faibles par rapport aux fausses variations dues à l'estimation variable du prix.

Cet écart-type relatif mensuel moyen d'estimation du prix est également nettement supérieur à ceux observés dans les 2 autres exemples, ceci est dû au fait que l'écart-type relatif **non**-mensuel moyen d'estimation du prix ne soit pas proportionnel à la période de rebalancement, mais qu'il impact linéairement la différence de rendement et que cette dernière soit effectivement mensualisée pour avoir un sens financier pratique.

De surcroît, pour simuler la couverture dans les mêmes délais, le nombre de tirages doit être inversement proportionnel à la période de rebalancement, or la l'écart-type du prix est inversement proportionnel à la racine carrée du nombre de tirages. Par conséquent, à temps de simulation fixé, **l'impact de l'écart-type relatif mensuel moyen d'estimation du prix sur la tracking error est proportionnel à la fréquence de rebalancement à la puissance 1,5ème.**

On peut donc voir que la significativité du résultat obtenu pour la première simulation est en fait lamentable, que celle obtenue pour la 2ème est très faible, et qu'il faut atteindre 200.000 tirages de



prix et un rebalancement seulement mensuel pour avoir un résultat qui commence à être significativement plus élevé que le bruit qui l'affecte. La conséquence étant que pour avoir un test significatif de tracking error dans des délais raisonnables, il est obligatoire de réduire le nombre de dates de rebalancements.

Rebalancer le plus souvent possible n'en est pas moins optimal financièrement, de même que cela le serait pour le calcul de la PL, cependant en respectant la façon de calcul la tracking error, c'est à dire l'écart-type de la différence des rendements entre le portefeuille de couverture et l'indice de référence (qui ici est notre produit), les résultats de tracking error obtenu avec des périodes de rebalancement courtes sont **non significatifs**.

En particulier, un rebalancement journalier renvoie une tracking error sans aucune valeur.

## 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/>