UNIVERSITY PARIS-DAUPHINE

MASTER THESIS - M1 203

Equity Structured products : Goals and Pricing of Autocallable Structures

Anthony NADJARI-FINCKELSTEIN

Tutor : M. Zhenjie REN

January 2021 - June 2021

Context & Abstract

Currently enrolled in an internship at Crédit Agricole CIB in the Equity Solutions department as an Exotic Equity Derivatives Structuring Intern, I have encountered multiple products and developed interest for autocallable structures. This thesis aims at giving a general view about structured products and autocallable, and about the classic way to price these options.

Acknowledgements

I would like to thank my tutor Mr. Ren for his expertise and advice, and also Mrs. Le Fol for helping me finding this subject and its stakes. Further, I would like to sincerely thank my Structuring team for giving me their point of view, but also for having trained me during this internship and for having transmitted their knowledge to me.

Contents

I Introduction **5**

II Structured equity products: goals and functions of auto-calls **6**

1 Short leg: the way to finance coupons **6**
1.1 Put options: European and American 7
1.2 European Put Down And In 7
1.3 Geared put: an alternative to a put option 8
1.4 Put-spread: a way to drastically reduce risk 9

2 Long Leg: benefiting from coupon **10**

3 Bullet Products: a first step to autocallable structures **10**
3.1 Reverse convertible: a simple structured product 10
3.2 Twin Win with European PDI: betting on volatility 11

4 Autocallable Structures **12**
4.1 Calibration of an Autocall: parameters 13
4.2 Athena Autocall 13

5 Optimizing the product to maximize the coupon and reduce the risk **14**
5.1 Choosing the right underlying 14
5.1.1 The Forward value of an underlying 14
5.1.2 Optimizing the forward 15
5.1.3 Choosing the right type of basket 17
5.2 Optimizing the barrier and duration of the products 18
5.3 Dividends: another tool to play with 20
5.3.1 The exposition to dividend 20
5.3.2 An exposure that had a huge impact during Covid-19 crisis 20
5.3.3 Decrement indices: a way to hedge the risk 21
5.3.4 Small review: custom indexes 22

III Pricing of autocall - litteral and systematic pricing **23**

6 Black-Scholes diffusion: the basic model of diffusion **24**
6.1 Diffusion equations and few prices 24
6.2 A problem to Black-Scholes diffusion 25
6.3 Volatility: Heston model 25

7 Computational pricing of structured products autocalls **26**
7.1 Monte-Carlo generation: the most common and used tool to price options and autocall . . . 26
7.2 Euler's Discretization . . . 27
7.3 computational pricing comparison . . . 32
7.4 Comments on the results . . . 34
7.5 Provisions: a way to improve the pricing . . . 34
7.6 The funding: what changes our prices a lot . . . 35

8 Algorithmic pricing - presentation of the pricer **37**
8.1 Generating random vectors . . . 37
8.2 Compute the forward . . . 38
8.3 Generate Black-Scholes Paths . . . 39
8.4 Generating Heston Paths . . . 40
8.5 The pricing function . . . 40
8.6 Calling the functions and returning the greeks . . . 42
8.7 Vanilla Pricing . . . 43

IV Conclusion **43**

Bibliography **45**

V Appendix: quick guide to use the pricer **46**

9 Sheet "main" **46**

10 Sheet "Data" **48**

11 Sheet "Vanilla" **48**

Part I
Introduction

Structured products have known a huge increase of interest since the crisis of 2008. People have had different objectives for their money. Speculation has let its place to hedging and coverage, and to avoid another crisis like this, regulations have been created, reducing arbitrage opportunities in the market. Clients have also been seeking more safety: the subprime crisis showed that unlikely scenarios can happen at any time, and that therefore, a security is needed.
Structured products are called so, because they depend on a stock, an index, a rate, or more globally: an underlying. The principle of a structured product can be basically compared to a sport bet: the match itself is the underlying, and the bet (which team is going to win) is the structured product.

These products can be said to be tailor-made: clients are looking for both innovative and custom payoffs. This is a big advantage, as it creates a huge diversity and allows every client to have its own product and better meet his need. However, this flexibility had an impact on the sell-side. As the number of possibilities increases, it was urgent to find processes to adapt the pricing of these products.

This thesis aims at giving a summary of basic products that we can find on the market and their functioning. We will then describe the diffusion models that are the most used to generates simulations via stochastic calculus.It will also describe a general but detailed enough solution to price structured products with and without autocallable feature. This is globally allowed by Monte-Carlo methods, that allows to custom the payoff as much as the client desires. Finally, an autocall pricer will be available with this thesis, and a quick guide can be found in appendix.

Part II

Structured equity products: goals and functions of auto-calls

First, there are two big categories of structured products: bullet and auto-callables. Bullet products are products that end at maturity, unlike autocallable products that can end whenever. Before introducing auto-callable products, it is necessary to present particular products that are used in those structures.
The functioning of structured product is the following: the client gives the bank a nominal expressed in percentage (if the client wants to invest 5 millions of euros, then 5,000,000 will be 100 %. At the end of the product, the owner of the structured product will receive back the nominal, plus or minus a certain amount depending on the product and the performance of the underlying.

First, an equity structured product is composed of two legs. The first leg is called the short leg, because it is composed of options that the client shorts, and the long leg, composed of options that the owner buys. Usually, the short leg is used to be able to finance (partially) the long leg: in a simple case scenario, the owner sells a structured put option to be able to finance digits, which are assimilated to coupons. The aim of this leg is also to ensure a certain safety, as we mentioned before. This safety is often called "Guaranteed Capital" (denominated "KG").

1 Short leg: the way to finance coupons

The short leg aims at selling an option to the bank that bets on the downside of the stock: the lower the stock (or the forward) will be, the more "in the money"[1]" the underlying will be, and thus, the more expensive will the option be. Finally, this will lead to a bigger coupon, financed by a product called "digital", on which we will come back later.
There are different kinds of products that we can find on the short leg. As a reference, the level of the strike of these options is expressed in a percentage of the spot, which is the current level of the underlying: if the spot is at 203, a strike at 100(%) means that the value of the strike is 203. Usually, short leg options that we find in structured products are striked at 100%, because it is the best equilibrium. Having a lower strike would increase the guaranteed capital but decrease the possible gain, and having a higher strike would have the exact opposite effect.

[1]The moneyness of an put option depends on the value of

$$(K - S_t)$$

at any time t. If it is positive, the option is "in-the-money", if it is negative the option is "out-of-the-money", and if it equals 0, it is "at-the-money".

1.1 Put options: European and American

A put option is the most classic option to bet on the downside of an underlying. The owner pays the price of the option called "the premium", and benefits in return of the negative performance of the underlying. There are two types of put options. The European[1] put option can be exercised only at maturity, whereas American put options can be exercised at any moment between the strike date (i.e. the beginning) and the maturity date.[2] The payoff[3] of this option at any time t is:

$$Payoff = (K - S_t)^+$$

where x^+ is the positive part of x (i.e. $max(0, x)$), K represents the strike, and S_T the value of the underlying at time T, the maturity. Both strike level and value of underlying can be represented either in percentage, or in value, but they must be of the same type.

The payoff can be represented as follows:

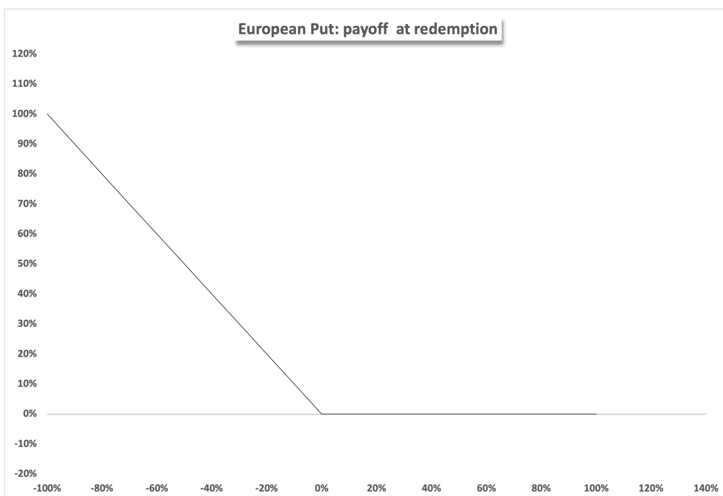

Figure 1: European Put Payoff at maturity

1.2 European Put Down And In

The European put down and in is part of a family of options called "barrier options". It is basically a put option, to which we add another parameter called the "barrier". There are 4 types of barriers. Once the barrier is breached (at maturity for a European option and during the lifetime of the product for an

[2] In practice, American Put options can be exercised "only" every day.

[3] The payoff of an option represents the gain/loss at maturity. It does not take into account the premium.

American option), the product either activates or deactivates. For and up-and-out barrier for instance, the product is deactivated ("out") if the underlying goes above the barrier ("up"). For a "down-and-in" barrier, the product is activated ("in") if the underlying goes below ("down") the barrier. Thus, the European PDI (for "Put-Down-and-In") will activate the put option if the underlying is below the barrier. The payoff is given by the formula:

$$Payoff = \begin{cases} (K - S_T)^+ & \text{if} \quad S_T \leq B \\ 0 & \text{otherwise} \end{cases}$$

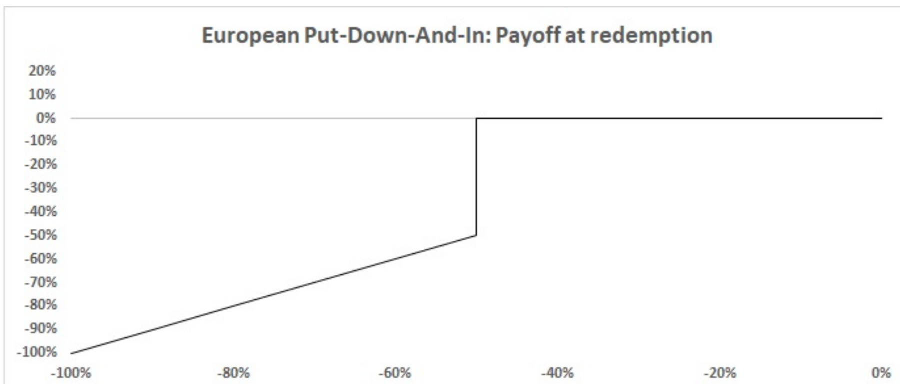

Figure 2: European Put Down And In Payoff

Compared to a simple put option, this product is less expensive : when the client is long PDI, he does not win anything as long as the barrier is not breached. However, this product is the most used because it allows the client to have its capital protected as long as the barrier is not breached.

1.3 Geared put: an alternative to a put option

Geared put is a common name for leveraged put. Before explaining the principle, let us first introduce the notion of "leverage". The objective of this product is to benefit from the performance of an At-The-Money put (we will denote that ATM Put), by financing only Out-Of-The-Money Puts (we will denote that OTM Put).
To illustrate that, let's take some concrete data: first, take an ITM Put, so a Put with a strike 100%. The maximum gain that the owner can have is called the cap: by default, the cap of a put is its strike, because the stock cannot go below 0 (theoretically):

$$\max_{S_T \in [0;+\infty[} (K - S_T)^+ = K$$

In comparison, the cap of a call is + ∞, because :

$$\max_{S_T \in [0;+\infty[} (S_T - K)^+ = +\infty$$

Then, take a put with a 85% gearing; the cap of the put is thus 85%. In order to get a 100% cap, we must then either buy one ATM put, or buy 100% / 85% = 1,25 OTM put. This number is called the leverage of the put. When the owner of a structured product shorts a geared put, he benefits from the cap of an ATM put, but also has his capital guaranteed until the stocks decreases

1.4 Put-spread: a way to drastically reduce risk

Another great alternative to both finance coupons and reduce risk exposure is the Put-spread. The principle of this product is to finance a put of Strike K_1 by short-selling another put of strike K_2, with $K_2 < K_1$. When an agent owns this strategy, it leads to the following result: the owner benefits from the negative performance of the underlying from K_1 to K_2, but his benefits are capped at $K_1 - K_2$ from K_2 to 0. However, using this strategy in a short leg allow the owner to have a maximum loss of $K_1 - K_2$. It is therefore one of the less risky strategies to use in a short leg, because it is the only one that guarantees a fixed capital at maturity compared to the other strategies we have seen before. However, it is not the one that will allow the user to finance the more coupons: since it is less risky, it is worth less. We can also see that, compared to a put, if we are short a put-spread, we buy a put with strike K_2, which reduces the amount we get.

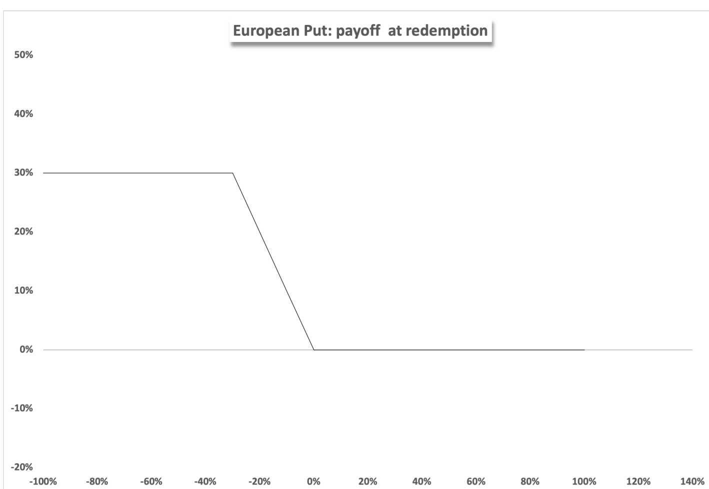

Figure 3: European Put Spread Payoff

2 Long Leg: benefiting from coupon

As we said earlier, the aim of the short leg is to finance coupons. The name of this product is a "digit", and is part of the family of barrier options like the Put-Down-And-In. This time, the option is an Up-And-In option. If the underlying goes above this barrier, it redeems a fixed coupon. Thus, the payoff can be expressed by this formula:

$$Payoff = \begin{cases} c & \text{if} \quad S_T \geq B \\ 0 & \text{otherwise} \end{cases}$$

It can also be written as:

$$Payoff = c * \mathbb{1}_{\{S_T \geq B\}}$$

3 Bullet Products: a first step to autocallable structures

Bullet products are structured products that do not have an autocallable feature. The observations can be at anytime from the strike date to the maturity date, but the payoff will always occur at maturity. To illustrate that, let us review a few products.

3.1 Reverse convertible: a simple structured product

The reverse convertible is probably the most classic and the simplest bullet structured product. At a given observation frequency, the client receives a fixed coupon regardless of the performance of the underlying. However, his invested capital is still exposed to the performance of the underlying. Indeed, the capital amount that is redeemed to the investor at maturity will be assessed by whether or not the PDI barrier was activated. In other words, the PDI aims at financing the coupons. The payoff at any time before maturity is therefore c, and the payoff at maturity can be written as below:

$$Payoff = \begin{cases} c & \text{if} \quad S_T \geq B \\ c + K - S_T & \text{otherwise} \end{cases}$$

An additional barrier can be added for the coupon: it is not guaranteed as before, and will be redeemed if the underlying goes above the coupon barrier (which is then higher than the PDI barrier).

The payoff of a Barrier Reverse Convertible at maturity is then:

$$Payoff = \begin{cases} c & \text{if} \quad S_T \geq B_{coupon} \\ K - S_T & \text{if} \quad S_T \leq B_{PDI} \end{cases}$$

3.2 Twin Win with European PDI: betting on volatility

The Twin-Win is the best product when we want to bet on the volatility of an underlying: when the owner thinks that the underlying will be moving a lot on the upside and/or on the downside, he could be inclined to buy either a put or a call, but buying both would be quite expensive. Therefore, the owner has the possibility to buy both by shorting a PDI. We say that we benefit from the "absolute performance" of the underlying. Therefore, the call will be uncapped, but the put will be capped from the PDI strike: it is therefore a Down-And-Out put, because it will deactivate if the PDI activates. The call will also deactivate if the PDI barrier is breached at or before maturity. Another variation would be to buy an Up-And-Out call: this would be less expensive than an uncapped call, and in case of medium-high volatility, it would also provide benefits if the underlying increases.

$$Payoff = \begin{cases} |S_T - K| & \text{if} \qquad S_T \geq B \\ K - S_T & \text{otherwise} \end{cases}$$

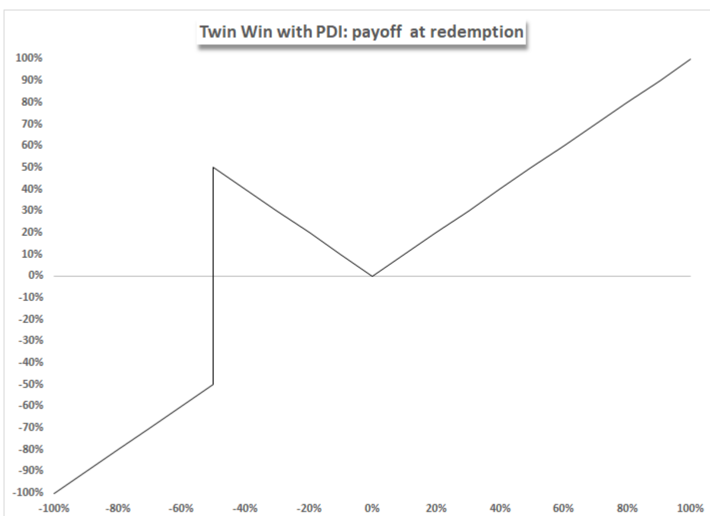

Figure 4: Twin Win with PDI Payoff

4 Autocallable Structures

Now that we detailed the principles of an equity structured product, it is time to introduce autocalls, a nickname for "autocallable structures". These are structured products composed with two legs as we saw before, but with an additional feature: the product can end before its maturity date.
Additional features go along with additional barrier: the product automatically ends when a barrier is breached. This barrier is called "autocall barrier", and is also an important parameter to play with to perfect the product. Once the product is recalled, the owner usually receives a coupon but this also depends on the product. For instance, product can have a coupon barrier different from the autocall barrier. This feature is also very useful, because as soon as it is recalled, the PDI deactivates and the owner locks the coupon. However, it is very important to understand that this autocall features makes the product unable to be expressed with other vanilla / barrier options: since we do not know in advance when the product is going to be recalled, it is impossible to say that the product is made of digits. There is of course the notion of digitals, because the client is redeemed a coupon when recalled (or when above a barrier), but we do not know the maturity of these. This complexity makes the autocall un-pricable with Black-Scholes model: we cannot deduce, from the payoff, a perfect written formula. Hence, other methods have to be applied to be able to price these options. We will come back to these methods later in this thesis.
When an autocall is ordered, parameters are taken into account depending on the product.

4.1 Calibration of an Autocall: parameters

Firstly, one or multiple underlyings shall be chosen. In this case, the type of basket has to be determined too. There are three types of basket in the market:

- the worst-of basket: at each observation date, the underlying with the worst performance determines the level of the basket.
- the best-of basket: at each observation date, the underlying with the best performance determines the level of the basket.
- the equally-weighted basket: at each observation date, the level of the basket is determined by taking the weighted average of the performances of each underlying.

Then, the maturity has to be chosen: even is the product can be ended before maturity thanks to the autocallable feature, if no autocall event has happened, the product will end like a bullet product depending on its performance and structure.
Finally, the client should choose the frequency of the observations. It is usually daily, quarterly, semi-annually, annually or bi-annually.

4.2 Athena Autocall

Let us introduce this structure by giving a classic example: the Athena autocall. This structure is the most basic autocall that we find on the market, but also the most traded in Equity Structured Products. The short leg of the autocall is composed of a PDI. When the underlying hits the autocall barrier, the products ends and the owner receives the nominal and a coupon. The coupon is said to be "snowball", because it increases as time goes.
For instance, take a classic Athena which has a maturity of 8 years, and yearly observations. The coupon that the autocall delivers can be written as:

$$\forall t \in [\![1 \ ; \ 8]\!], c_t = t * c_1 = c_{t-1} + c_1 \text{ (assuming } c_0 = 0)$$

The payoff cannot be expressed in one formula, but it is possible to express the payoff depending on the scenario that happens. Taking an athena with maturity n, observed yearly with an annual coupon of c gives us:

$$\forall t \in [\![1 \ ; \ n-1]\!], \ Payoff_t = (100\% + c_t) * \mathbb{1}_{\{S_t \geq B_{autocall}\}}$$

Then, at maturity, the payoff ends up being:

$$Payoff = \begin{cases} 100\% + n * c & \text{if} \quad S_T \geq B_{autocall} \\ 100\% - (K - S_T) & \text{if} \quad S_T \leq B_{PDI} \end{cases}$$

(Note that $K = 100\%$ for almost every PDI)

5 Optimizing the product to maximize the coupon and reduce the risk

Now that we have described the parameters that influence the price of an autocall, the aim is to see which parameters can be optimized in order to better meet the client's needs and ensure a good revenue by suiting the risk profile of the investor. First, we will determine which parameters specific to the underlying can be improved.

5.1 Choosing the right underlying

As we mentioned before, the first thing that the owner looks for in an autocall is the guaranteed capital. Depending on the underlying, the client will adapt his short leg.

5.1.1 The Forward value of an underlying

It is now time to present a very important notion that will help us all along this thesis: the forward value of an underlying. This value represents an estimation of the future value of an underlying, given its actual level (called spot), the dividend yield of the underlying, and the risk-free interest rate. More precisely, given a spot S_0, an interest rate r and a dividend yield q, the forward at any time t is:

$$S_t = S_0 \, \mathrm{e}^{(r-q)t}$$

In practice, the risk free interest rate does not exist. We thus use the forward rates curves. This corresponds to the fixed rate in a swap.

Figure 5: Euro Swap Rates Curve

But why is the forward so important? First, the forward value is the only indicator that tells us a possible future value of the underlying. For instance, if we want to study the possibility of the underlying to go above an autocall barrier in one year, we cannot refer to the spot level, because it would not take into account the evolution of the stock. First, the forward value of a stock will have an impact regarding the barriers. The price of an autocall is in fact the discounted payoff of the product. This result is not direct, and requires some stochastic calculus knowledge. The proof can be found in Bouzoubaa's book[2]. Thus, the price of the autocall is given by the following formula:

$$P_x = \mathbb{E}[e^{-rt} CF(t)],$$

where $CF(t)$ is the cash-flow at time t. For a n-year maturity autocall, it can also be rewritten as:

$$P_x = \sum_{t=1}^{n} p_t * CF(t)\, \mathrm{e}^{-rt},$$

where p_t is the probability to get recalled: $p_t := \mathbb{P}[S_t \geq B_{autocall}]$
We can see in this formula that, for a same price, an infinite number of combinations of probability and cash-flows exists. Thus, for a same price, increasing the probability of being recalled will mechanically decrease the cash flow. Thus, for a given nominal, a possibility to redeem better coupon would be to decrease the probability of being recalled. Looking only at the underlying, this implies choosing an underlying that has a low probability to breach the autocall barrier, hence: an underlying with a low forward. The lower the forward, the lower the probability, thus the higher the cash-flows (so the coupon). In addition, having a low forward would make the PDI deeper in-the-money, hence increase its value and therefore finance more coupons. However, choosing to put the barrier to an absurdly high level would be a non-sense because the client would lower a lot his guaranteed capital. Also, having a very low forward would increase the possibility to activate the PDI at maturity, and therefore to engage losses. This is all the difficulty of autocall products: playing with the parameters to both meet great possible redemption and probable / concrete evolution scenario. It also depends a lot on the analysis of the client: his view on the market will decide his risk appetite.

5.1.2 Optimizing the forward

Let us now come back to the forward. The aim of the structured product is therefore to have a low forward, and hoping that the realisation of the forward does not happen. Buying an autocall is thus betting on the non-occurrence of the forward, and depends on the analysis of the buy-side. If the investors are bullish on a stock, they will be even more delighted to find a low forward.
Our objective is now to minimize the forward decently. By simply taking a look at the formula above, choosing a stock with a high dividend yield would be an opportunity. Indeed, with the hypothesis of absence of arbitrage, if a stock pays a dividend, its value will decrease buy the amount the dividend, otherwise we could by a stock just before the dividend is paid, get the dividend, and sell back the stock.

To illustrate that, we can do some quick calculations and compute the forward for two french stocks: Axa and Airbus. Axa is a very popular stock because it pays high dividends, whereas Airbus does not pay any.

	Forward		Dividend Yields	
Maturity	CS FP	AIR FP	CS FP	AIR FP
07/05/22	99.54%	99.54%	6.29%	0.00%
07/05/23	93.66%	99.23%	6.26%	0.00%
07/05/24	88.36%	99.23%	6.36%	0.00%
07/05/25	83.76%	99.52%	6.23%	0.00%
07/05/26	80.00%	100.22%	5.93%	0.00%

Figure 6: Forward and Dividends of Axa and Airbus

As we can see in the data-frame above, a high-maturity autocall on Axa would be very interesting, because it has a very low forward: around 80% of its initial value, so in other words, a loss of 20% of its value.

5.1.3 Choosing the right type of basket

As explained above, three types of basket exist, and provide different levels of coupon depending on the forward of the underlyings. Usually, the worst-of basket is the most used, because low-forward stocks imply an even lower basket forward. In fact, since the worst-of basket takes the lowest performing underlying, the forward of this basket will always be lower than (or equal to) each stock individually. However, it is mostly seen as a double-edge sword; on the one hand, this would result in a lower forward, hence much higher coupons. On the other hand, the probability to breach the autocall barrier would be highly reduced. Another factor has to be taken into account when choosing a basket of stocks: the correlation. In a nutshell, correlation represents the evolution of a stock relatively to another. It is measured by the correlation coefficient:

$$\rho_{i,j} = \frac{cov(S_i, S_j)}{\sigma_i \ \sigma_j}$$

where σ_i is the standard deviation of S_i.
When dealing with several underlyings, it is necessary to represent their correlations with the correlation matrix:

$$M_\rho = \begin{pmatrix} \rho_{1,1} & \rho_{1,2} & \cdots & \rho_{1_n} \\ \rho_{2,1} & \ddots & & \vdots \\ \vdots & & \ddots & \rho_{n-1,n} \\ \rho_{n,1} & \cdots & \rho_{n,n-1} & \rho_{n,n} \end{pmatrix}) = \begin{pmatrix} 1 & \rho_{1,2} & \cdots & \rho_{1_n} \\ \rho_{2,1} & \ddots & & \vdots \\ \vdots & & \ddots & \rho_{n-1,n} \\ \rho_{n,1} & \cdots & \rho_{n,n-1} & 1 \end{pmatrix}$$

Correlation has a big impact on the basket. When using a worst-of basket, the owner is always "short-correl" (for short correlation), meaning that he wants the correlation to be as low as possible. Since $\rho_{i,j} \in [-1, 1]$, the ideal situation

6M\5Y	TOTF.PA	AXAF.PA	BOUY.PA	BNPP.PA
TOTF.PA	100.00%	77.09%	71.02%	76.05%
AXAF.PA	77.09%	100.00%	72.86%	85.39%
BOUY.PA	71.02%	72.86%	100.00%	75.87%
BNPP.PA	76.05%	85.39%	75.87%	100.00%

Figure 7: Correlation matrix between Total, Axa, Bouygues and BNP Paribas

is when the correlation is equal to -1. Indeed, if we take the basic example of two stocks, a -1 correlation means that when an underlying moves up by 1%, the other one moves down by 1%. Since we are taking the lowest underlying in a worst-of basket, it guarantees that, in this case, no matter in which way the underlying moves, it will always benefit to the basket. To be more precise, the correlation impact on the option price can be described by a greek [4]: the cega, also called correlation delta. This sensitivity is computed by taking the difference of the price with the normal correlation matrix, and the price with a shift of correlation, meaning:

$$\frac{P_{M_{\rho+\epsilon}} - P_{M_\rho}}{\epsilon}$$

with

$$M_{\rho+\epsilon} = M_\rho + \epsilon * \begin{pmatrix} 1 & 1 & \cdots 1 & \\ 1 & \ddots & & \vdots \\ \vdots & & \ddots & 1 \\ 1 & \cdots & 1 & 1 \end{pmatrix}$$

What is to remind is that the sign of cega tells whether or not we "want" correlation on the basket: for a worst-of basket, cega will be negative. As the correlation decreases, the coupon paid will be higher (hence, the price would be higher). We can use symmetrical reasoning for best-of and equally weighted baskets.

5.2 Optimizing the barrier and duration of the products

As mentioned in the beginning, one of the main goals of the autocall is to ensure a quick roll: For the client, it allows him to invest in a new product faster, hence to benefit from new coupons. For the bank, it allows it to sell another product, and therefore to take some margin for it again.
The concept of duration intervenes there. The duration of an autocall is the average time interval at which it is going to be recalled. A duration of 3 means

[4] Greeks represent the sensitivities of the option price to the modification of one or several parameters: the spot (delta), volatility (vega) or sensitivity to delta (gamma)

the autocall will, on average, end in three years from the strike date.
Further, as we saw before, the stocks we chose have a decreasing forward, hence a lower chance to breach the autocall barrier as time to maturity reduces. It also means that the product has a great chance to end after the first period of time, and as a consequence, the coupon would be reduced. A trick that is often used is to define a "non-call period": this means that during this period, the autocall can not end, and the coupon is still being incremented. To illustrate that, let us have a look at the following figures.This shows the probabilities of recall for different barriers, as time to maturity reduces (this example is taken on a classic Athena with quarterly observations), with the feature of a non-call for 1 year ("NC1Y"):

	Autocall Barrier (in % of initial level)						
Trimester	85	90	95	100	105	110	115
T1	0.000%	0.000%	0.000%	0.000%	0.000%	0.000%	0.000%
T2	0.000%	0.000%	0.000%	0.000%	0.000%	0.000%	0.000%
T3	0.000%	0.000%	0.000%	0.000%	0.000%	0.000%	0.000%
T4	67.198%	57.939%	47.960%	37.983%	29.084%	21.530%	15.539%
T5	5.079%	5.772%	6.269%	6.452%	6.177%	5.586%	4.729%
T6	2.782%	3.351%	3.821%	4.074%	4.086%	3.940%	3.626%
T7	1.734%	2.121%	2.484%	2.781%	2.981%	2.969%	2.796%
T8	1.155%	1.430%	1.719%	1.976%	2.095%	2.181%	2.136%
T9	0.948%	1.186%	1.429%	1.672%	1.825%	1.878%	1.880%
T10	0.780%	0.969%	1.187%	1.373%	1.493%	1.564%	1.624%
T11	0.629%	0.782%	0.949%	1.090%	1.229%	1.307%	1.293%
T12	0.504%	0.648%	0.793%	0.951%	1.107%	1.157%	1.171%

Figure 8: Probabilities of Recall evolving with Autocall Barrier and Time

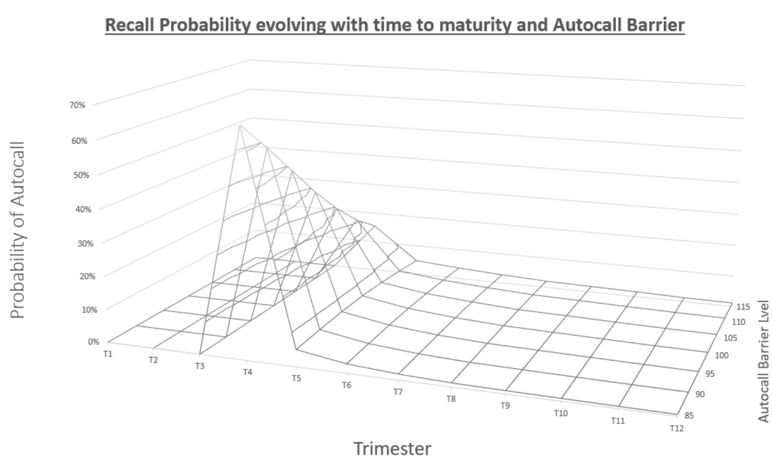

Figure 9: Probabilities of Recall evolving with Autocall Barrier and Time

Obviously, as the barrier increases, the probability to be recalled decreases. We can also see that the recall probability is the highest at the first autocall period, and this is true no matter how long the non-call period is. It is then crucial to wonder when the product is going to end. In a good approximation, following the results that we found, we can have two scenarios. The first one would be that the product gets recalled after the first observation. The second one would be that the product is not recalled at maturity, but still ends. In this case, the client would either receives his nominal back, or suffer a loss from the PDI. Hence, the probability of the autocall to last until maturity plays an important role in the pricing, because it can add a lot of value to the PDI.
Another tool to play with is the observation frequency of the product: the more frequent we observe the product, the more chance of being recalled we have.

5.3 Dividends: another tool to play with

Dividends are the factor that impacts the most the forward. As we saw, it depends on the rates and dividend, but in practice, the variations of rates do not impact the forward a lot. However, the dividends is subject to the most variations.

5.3.1 The exposition to dividend

It is now time to introduce precisely the first greek, i.e. the first sensibility of an autocall price to a factor. This greek is called "epsilon", and refers to the variation of the price for a unit of a variation of the dividend yield of the underlying.

$$\epsilon = \frac{\partial P_x}{\partial q}$$

Thus, for a move of 1% of the dividend yield, the price of the autocall should move by ϵ. Indeed, this ratio is negative, because an increase of the dividend means a lower forward, thus a lower autocall price (or higher coupon).

5.3.2 An exposure that had a huge impact during Covid-19 crisis

Beyond all that, dividends represent an exposure to risk for the bank. A high dividend stock represents a possibly high coupon, but also represents a possible increase of the forward. In fact, if the dividends paid are lower than expected, the stock will not go as low as expected, and the possibility to breach the autocall barrier will increase. This happened during the Covid-19 crisis. Banks had a lot of issues because dividends were cut[3], meaning that the companies did not pay any and traders were not prepared enough for this change in dividends.

5.3.3 Decrement indices: a way to hedge the risk

As discussed before, dividends are truly useful, because they can guarantee a very good ratio benefit/risk, but are also very difficult to hedge for the bank. It is then interesting to find a compromise between the bank and the client, a tool that would ease the hedging for the bank, and also guarantee a decent coupon level for the client. An underlying that is often use to overcome this is called a "decrement indices", or a "synthetic dividends indices".
The principle of these underlying is the following. Let's say that the client is long the EuroStoxx 500 (SX5E). It means that he owns a basket of European stocks, which all have a weight. Some of these stocks can pay a dividend, and the owner of the indice will receive the weights of the stocks multiplied by the dividends. Thus, the index will lose a bit of its value due to the dividends, but the owner will compensate that by earning dividends. However, the fact that the dividends are not fixed make the decrease of the index variable, and it is therefore another parameter to take into account. On the contrary, a decrement index will lose a fixed value each period (fixed in advance). But where are the dividends going? Of course, the index can be composed of stocks that pay dividends. But these dividends will be reinvested in the index, to compensate the fixed decrement.

Figure 10: Computation of a decrement index

<u>Pros of Decrement indices:</u>

- The ultimate pro of these index is the neutralization of the dividend risk. As mentioned before, these indexes are also called "synthetic dividend" because the decrement plays the role of a dividend (even if it does not directly go in the pocket of the investor). Therefore, there will not be uncertainty to epsilon anymore.
- These decrements are often in the range of 3% to 6%, which is lower than the index itself. It means a lower forward, therefore a better participation to coupon. From another point of view, it means that the PDI Barrier can be lowered for an equivalent coupon, which increase the guaranteed capital.

<u>Cons of decrement indices:</u>

- If the decrement is higher than the dividends paid, it will underperfom the index and therefore reduce the probability to be autocalled.

5.3.4 Small review: custom indexes

Decrement index are part of a family of indices called "custom indexes". The principle of this indices is to have a chosen composition of the index and repartition of the weights.
Some indices will simply replicate the stocks of another one, and modify the type of basket. The ISXEC50 (EURO iSTOXX Equal Weight Constant 50) ensures a synthetic dividend and modifies the weight of the basket to become an equally weighted index (which changes a lot). The ISXE50D index (EURO iSTOXX Environmental 50 EqualWeight NR Decrement 5%) gathers environmental-caring banks and companies with a fixed decrement.
This type of indexes offers a very good opportunity for autocalls, but also represents a very good marketing tool, even more nowadays where environmental issues are present.

Part III

Pricing of autocall - litteral and systematic pricing

The first step of our journey was to express the different versions of autocalls, their features, and the possibilities to custom them. It is now time to begin the second phase when emitting a product, which is providing a product to the client.

But what is providing a price? Before moving on to the pricing, we have to clearly define these term, meaning what we are solving.

As mentioned before, the client comes with a nominal of 100%, and the goal of the product is to give back this nominal plus eventual coupons, minus eventual loss. Usually, clients are solving the level of coupons, but it is totally possible to solve other parameters, such as the barrier of autocall (or coupon). In this case, the client comes for a definite level of coupon and is interested in seeing which barrier the bank can propose. It is also possible to solve the "Upfront", which, in a nutshell, represents the margin of both the bank and the client, if this client is a broker for instance (he buys the product to sell it again to a private client or an asset manager). Upfront can be solved by making the difference between the nominal and the product value.

The first part of this section will be dedicated to the use of stochastic calculus and models to compute prices. The second part will compare the different models of algorithmic pricing through a hand-made pricer that aims at getting as close as possible to prices that can be found using a real pricer. We will also explain the discrepancies between those results, and the pros and cons.

6 Black-Scholes diffusion: the basic model of diffusion

6.1 Diffusion equations and few prices

We start by introducing Black-Scholes Model. This model at first aims at describing the evolution of an underlying regarding volatility and interest rate. It describes the variation of the underlying during a short period of time. The fundamental equation is the following:

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

where W_t represents a Brownian motion. For a given filtration $(\mathcal{F}_t)_{t\geq 0}$, a Brownian Motion is defined as follows:

- $t \longmapsto B_t(\omega)$ is continuous for all $\omega \in \Omega$
- $\forall t \geq s, B_t - B_s$ is independant from $\mathcal{F}_s$
- $\forall t \geq s, B_t - B_s \sim \mathcal{N}(0, t-s)$

Therefore, the variation can be expressed by two factors. The first one is called the trend, and does not depend on random movements: it is the path that the stock would "normally" follow. The second component is the Brownian Motion[5], that explains the random moves of the stock. Using Itô Lemma on a function f describing the payoff of an option, we can deduce that:

$$df = \left(\frac{\partial f}{\partial S}\mu S + \frac{\partial f}{\partial t} + \frac{1}{2}\frac{\partial^2 f}{\partial S^2}\sigma^2 S^2\right) dt + \frac{\partial f}{\partial S}\sigma S dW_t.$$

We can then use an non-arbitrage argument and end by getting the famous formulas for the price of a call and a put, that are given by:

Price of a call option:

$$C = S_0 N(d_1) - Ke^{-rt} N(d_2)$$

Price of a Put option:

$$C = Ke^{-rt} N(-d_2) - S_0 N(-d_1)$$

with $d_1 = \frac{\ln(S_0/K)+(r-\sigma^2/2)T}{\sigma\sqrt{T}}$ and $d_2 = d_1 - \sigma\sqrt{T}$
This precious diffusion equation can now give us all the tools to price options with a define payoff.

[5]It is originally used to describe the movement of a pollen particle on water.

6.2 A problem to Black-Scholes diffusion

Although Black-Scholes is a very powerful tool, there is a big issue in this model, that basically lies in its hyopthesis: the constantness of its parameters.
It is obvious that in practice, most of the parameters than we take as endogenous, are indeed market parameters that fluctuate as well as the stock does. A static model like Black-Scholes' is therefore not enough to fit well reality. We can therefore introduce, for each parameters, a model to fit as much as possible reality.

6.3 Volatility: Heston model

One of the most important tools for autocallable pricing is the volatility. When deciding which stock to use, it is major to check its volatility, because it will determine the easiness to breach the barrier. It will also determine the value of the PDI, because it gives an indication about how hard it is to breach the PDI barrier. The principle of Heston model is that it is a stochastic volatility model. As a nutshell, it allows to give a random distribution to the volatility, and to be variable, unlike Black-Scholes model.
We begin by defining the diffusions of stock and volatility as follows:

$$\begin{cases} dS_t = \mu S_t dt + \sqrt{v_t} S_t \; dW_t^S \\ dv_t = \kappa(\theta - v_t)dt + \sigma\sqrt{v_t} dW_t^V \end{cases}$$

We have a first equation to describe the stock like in Black-Scholes model, but we add another one for the volatility. It is very important to notice that the two Brownian Motions are different and that, in addition, they are correlated, meaning:

$$d\langle W_t^S, W_t^V \rangle = \rho dt$$

This type of model is called "mean reverting" process. In a long term, the underlying is supposed to come back to its mean (to revert), at a certain speed. This model is part of a mean-reverting processes family called "Cox-Ingersoll-Ross" models. Let us explain the above parameters taken into account in the model:

- v_t is the stochastic volatility at time t
- θ is the long term mean:
- κ is the speed at which the volatility is going to return to its mean (mean-reversion rate)
- σ is the "volvol": it is the volatility (constant) of the volatility (stochastic)

To these two equations, we must add a condition. This was proven by Feller[4] in 1951, and it is a necessary condition for the volatility process to be positive :

$$2\kappa\theta > \sigma^2$$

7 Computational pricing of structured products autocalls

In the previous part, we detailed the models that are used to compute prices of structured products and vanillas. However, in practice, we are confronted to another issue: there is no closed formula to describe the price of an autocall. It is always possible to compute the equation of diffusion, but it is not enough: in fact, the price of an autocall depends on τ^*. It is part of the family of stopping times, and represents the time at which the autocall is going to end. It is defined as:
Indeed, since τ^* is not fixed, we can not take the discounted payoff and price the option. We therefore have to find other ways to approach as best as possible the price of the product.
First, we must notice that we have a fixed payoff for each observation. We will use this to develop our pricer. Further, we will determine how we can improve our method to reduce complexity and calculation time.

7.1 Monte-Carlo generation: the most common and used tool to price options and autocall

Monte-Carlo method is the best tool to price exotic options which payoffs are complex.
Originally, Monte-Carlo methods are used to approximate quantities by simulating random variables. For a given quantity P that we want to approximate (for instance, the price of an option), we rewrite the quantity as an expectation of random variables $S = \left(S^{(i)}\right)_{i \geq 0}$ following the same probability law (in our case, payoffs):

$$P = \mathbb{E}^{\mathbb{P}}[h(S^{(i)})] = \int_{\mathbb{R}} h(S^{(i)})\, \mathbb{P}(\mathrm{d}S^{(i)})$$

From now, we know from the law of large numbers that the unbiased estimator of the mean will converge to the undiscounted price of the option:

$$\tilde{P} = \frac{1}{n} \sum_{i=1}^{m} h(S^{(i)})$$

Models that we studied before allow us to give the diffusion of the underlying, and therefore to get a forecast value S_t for any possible time t. From that, we can decompose for each observation the payoff that will result from this generation: it is called a path. From that, we know that the price of a product is given by the formula:

$$P(t, S_t) = \mathrm{e}^{r(T-t)}\, \mathbb{E}^{\mathbb{P}}[h(S_t)|\mathcal{F}_t]$$

At each observation and for each path, we are able to deduce a possible value of the path. We can do the following reasoning for a number m of steps, and at the end, we will dispose of a vector :

$$\left(S^{(1)}_{\tau_1}, S^{(2)}_{\tau_2}, \ldots, S^{(m)}_{\tau_m}\right)$$

with τ_i being the stopping time of the i^{th} path result. We can then apply Monte-Carlo method to get the price of the option.
In the above section, we also mentioned the discounting factor, used to compute the present value of the payoff. We must notice that this the discounting will not happen at the end of the simulation, as we did for bullet option. Since the aim of the discounting is to get the value of the cash from now to the autocall time, we must discount the payoffs for each observation when they are autocalled, because the autocall time will almost surely not be all equal to each other [6].
We are however exposed to a problem: the diffusion equation allow us to have underlying values in a continuous way, whereas our observations are discrete. We will now see how to discretize our processes by Euler's discretization.

7.2 Euler's Discretization

In this part, we will follow Leif Andersen's research paper[5] steps in order to develop a use-able process for algorithmic pricing. We first begin by using the "naive" method of the discretization. Starting from the equation of the model, we first compute the discretization of $ln(S_t)$. Taking the logarithm will ease the computations, but it will also improve the algorithm. The quantity can be written with integrals by:

$$\begin{aligned} d\ln(S_t) &= \frac{1}{S_t}\mathrm{d}S_t - \frac{1}{S_t^2}d\langle S\rangle_t \\ &= \frac{1}{S_t}(rS_t dt + \sqrt{v}_t S_t dW_t^S) - \frac{1}{S_t^2}(\sqrt{v}_t)^2 S_t^2 dt \\ &= (r - \frac{v_t}{2})dt + v_t dW_t^S \end{aligned}$$

Now that we have the expression that we wanted, we can move to the discretization part. We first write the SDE in its integral form:

$$\ln(S_{t+\Delta t}) = \ln(S_t) + \int_t^{t+\Delta t} (r - \frac{v_s}{2})\ ds + \int_t^{t+\Delta t} \sqrt{v_s}\ dW_s^S$$

We can now proceed to approximate each term by the discretization scheme:

$$\int_t^{t+\Delta t} (r - \frac{v_s}{2})\ ds \approx (r - \frac{v_t}{2})\Delta t$$

[6]This means that our previous function h will include this discounting factor.

We also have the following approximation for the stochastic part:

$$\int_t^{t+\Delta t} \sqrt{v_s}\; dW_s^S \approx \sqrt{v_t}(W_{t+\Delta t}^S - W_t^S)$$

We also know that the difference of the two Brownian motions from above follows a gaussian law with the following parameters:

$$\begin{aligned} W_{t+\Delta t}^S - W_t^S &\sim \mathcal{N}(0, t + \Delta t - t) \quad \sim \mathcal{N}(0, \Delta t) \\ &= \sqrt{\Delta t}\; Z^S \end{aligned}$$

with $Z_S \sim \mathcal{N}(0,1)$ a gaussian variable. We end up by getting the full Euler's discretization for the diffusion of the underlying:

$$\ln(S_t + \Delta t) = \ln(S_t) + (r - \frac{v_t}{2})\Delta t \; + \sqrt{v_t \Delta t}\; Z^S$$

To end the computations, we switch to the "normal" form of the diffusion (without the logarithms):

$$S_{t+\Delta t} = S_t \; \exp((r - \frac{v_t}{2})dt + \sqrt{v_t \Delta t} Z^S)$$

With the previous reasoning, we can also deduce a discretization for the process of the volatility:

$$v_{t+\Delta t} = v_t + \int_t^{t+\Delta t} \kappa(\theta - v_s)\; ds \; + \int_t^{t+\Delta t} \sigma \sqrt{v_s} dW_s^V$$

Approximate the first integral by:

$$\int_t^{t+\Delta t} \kappa(\theta - v_s)\; ds \approx \kappa(\theta - v_t)\Delta t$$

Approximate the second term by:

$$\int_t^{t+\Delta t} \sigma \sqrt{v_u} dW_s^V \approx \sigma \sqrt{v_t}(W_{t+\Delta t}^V - W_t^V) \; = \sigma \sqrt{v_t \Delta t}\; Z^V$$

With $Z^V \sim \mathcal{N}(0,1)$.
To conclude, we end up by having the following discretization for the processes:

$$\begin{cases} S_{t+\Delta t} & = S_t \; \exp((r - \frac{v_t}{2})dt + \sqrt{v_t \Delta t} Z^S) \\ v_{t+\Delta t} & = v_t + \kappa(\theta - v_t)\Delta t + \sigma \sqrt{v_t \; \Delta t} Z^V \end{cases}$$

Finally, we can proceed to the simulation of the correlated Brownian motions. We start by generating 2 Brownian motions independently W^S and W_2. Then, we compute $W^V = \rho W^S + \sqrt{1-\rho^2}W_2$. However, we can improve this discretization by getting rid of a problem. In our scheme, the discretization of the volatility can lead to a negative volatility even if Feller's condition is not breached. There are several ways to fix the problem. Broadie[6] and Kaya have first proposed to simply disregard the negative values and set them to 0. After more investigation, they proposed an alternative method to Euler's discretization. First, we can rewrite Heston's model as follows:

$$\begin{cases} dS_t & = \mu S_t dt + \sqrt{v_t} S_t \ (\rho dW_t^S + \ \sqrt{1-\rho^2} dW_t^V) \\ dv_t & \qquad = \kappa(\theta - v_t)dt + \sigma\sqrt{v_t}dW_t^V \end{cases}$$

with, this time, W^S and W^V being sampled independently. In fact, we can easily prove that these two motions have a correlation of ρ. First, the sum of these Brownian motions is of course a Brownian motion by linearity. For the correlation, a few lines lead to the result. We begin by recalling the formula of the cross quadratic variation of two processes X and Y:

$$\langle X, Y \rangle_t := \frac{1}{4}[\langle X+Y \rangle_t - \langle X - Y \rangle_t]$$

And we can observe that:

$$\langle X+Y \rangle_t = \langle X \rangle_t + 2\langle X, Y \rangle_t + \langle Y \rangle_t$$

We can then derive from the cross quadratric variation of the two motions that:

$$\begin{aligned} \langle \rho W^S + \sqrt{1-\rho^2} W^V, W^S \rangle_t &= \frac{1}{4}[\langle \rho W^S + \sqrt{1-\rho^2} W^V + W^S \rangle_t - \langle \rho W^S + \sqrt{1-\rho^2} W^V - W^S \rangle_t] \\ &= \frac{1}{4}[\langle (1+\rho) W^S + \sqrt{1-\rho^2} W^V \rangle_t - \langle (\rho - 1) W^S + \sqrt{1-\rho^2} W^V \rangle_t] \end{aligned}$$

To ease the computations, we decompose the two quadratic variations separately with the formula of the sum:

$$\langle (1+\rho) W^S + \sqrt{1-\rho^2} W^V \rangle_t = \langle (1-\rho) W^S \rangle_t + 2\langle (1+\rho) W^S, \sqrt{1-\rho^2} W^V \rangle_t + \langle \sqrt{1-\rho^2} W^V \rangle_t$$

The second term of the equations equals 0 by independance of the two Brownian motions. We end up with:

$$\begin{aligned} \langle W^S(1+\rho) + W^V \sqrt{1-\rho^2} \rangle_t &= (1+\rho)^2 \langle W^S \rangle_t + \sqrt{1-\rho^2}^2 \langle W^V \rangle_t \\ &= (1+\rho)^2 \langle W^S \rangle_t + (1-\rho^2) \langle W^V \rangle_t \end{aligned}$$

Further, we know that for a Brownian motion B, $\langle B \rangle_t = t$. So, finally:

$$\langle W^S(1+\rho) + W^V \sqrt{1-\rho^2} \rangle_t = t(1 + 2\rho + \rho^2 + 1 - \rho^2) \quad = t(2+2\rho)$$

We can do the same reasoning for the other term:

$$\begin{aligned}\langle W^S(\rho-1)+W^V\sqrt{1-\rho^2}\rangle_t &= (\rho-1)^2\langle W^S\rangle_t+\sqrt{1-\rho^2}^2\langle W^V\rangle_t\\ &= (\rho-1)^2\langle W^S\rangle_t+(1-\rho^2)\langle W^V\rangle_t\\ &= t(1-2\rho+\rho^2+1-\rho^2)\\ &= t(2-2\rho)\end{aligned}$$

Finally,

$$\begin{aligned}d\langle\rho W^S+\sqrt{1-\rho^2}W^V,W^S\rangle_t &= \frac{dt}{4}[2+2\rho-(2-2\rho)]\\ &= \frac{dt}{4}(4\rho)\\ &= \rho dt\end{aligned}$$

Now that we have our alternative Heston, we can give the value of S_t knowing S_u (with $u \leq t$), with an integral form:

$$S_t = S_u\,\mathrm{e}^{r(t-u)-\frac{1}{2}\int_u^t v_s ds+\rho\int_u^t\sqrt{v_s}dW_s^S+\sqrt{1-\rho^2}\int_u^t\sqrt{v_s}dW_s^V}$$

Then, we do the same for the process of the volatility:

$$\begin{aligned}v_t &= v_u+\int_u^t\kappa(\theta-v_s)\ ds\ +\int_u^t\sigma\sqrt{v_s}dW_s^S\\ &= v_u+\kappa\theta(t-u)-\kappa\int_u^t v_s ds\ +\int_u^t\sigma\sqrt{v_s}dW_s^S\end{aligned}$$

Then, we process to sampling the variables by a method that can be summarized by theses steps:

- Knowing v_u, we generate a sample of v_t
- Knowing v_t and v_u, we generate a sample of $\int_u^t v_s\ ds$
- Using the values above and the equation of v_t, we find $\int_u^t\sqrt{v_s}dW_s^S$
- We generate a sample of S_t.

This method thus gives exact simulations for Heston model, and solves our problem. It is however more complex to implement, so our pricer will not rely on this method.
We will conserve our discretization, and propose a less complex method that is more easily implementable. Instead of taking the negative values, we can replace the function of the volatility by other functions that will better fit the scheme. We can translate this problem by the following: we are looking for 3 stable functions g_1,g_2 and g_3 that take v_t as an argument:

$$v_{t+1} = g_1(v_t)+\kappa(\theta-g_2(v_t))\Delta t+\sqrt{g_3(v_t)\Delta t}Z^S$$

As we want to maintain positive values for the volatility, we can directly think of two functions: the positive part function, and the absolute values function. Depending on the function that we use, we can classify the methods and report their creators in the below dataframe:

Method Name	**Paper**	$g_1(x)$	$g_2(x)$	$g_3(x)$
Absorption	?	x^+	x^+	x^+
Reflection	Diop [2003], Bossy and Diop [2004], Berkaoui [2008]	$\mid x \mid$	$\mid x \mid$	$\mid x \mid$
Higham – Mao method	Higham and Mao [2005]	x	x	$\mid x \mid$
Partial truncation	Deelstra and Delbaen [1998]	x	x	x^+
Full truncation	Lord, Koekkoek and Van Dijk [2007]	x	x^+	x^+

Figure 11: Summary of the different methods

The last function that we mentioned is more radical: instead of considering only positive values, we inverse the sign of the negative one, which doubles the movement of change in our truncation. Hence, this methods, though it is wildly used, is less stable. Choosing the full truncation leads us to the following discretization scheme for the volatility simulation:

$$v_{t+1} = v_t + \kappa(\theta - v_t^+)\Delta t + \sigma\sqrt{v_t^+ \ \Delta t} \ Z^V$$

We now have our steps to generate paths and compute the price of autocallable structures. We first need to set up the Monte-Carlo method before computation. As a brief summarizing:

- choose the model to diffuse the spots and eventually parameters
- discretize the model in order to have views for each autocall period
- choose n the number of observations, and m the number of Monte-Carlo paths we want to generate.[7]

Now that we have our prepared scheme, we can proceed to the computations:

- generate our m stochastic paths
- for each of the n observations, we look at the payoff and if it's not called (or if it's maturity), we continue to the next one. At the end, we keep the payoff result in memory (or add it to a variable). Note that we discount the payoff differently depending on the autocall date.
- after the m paths have been checked, we take the average of the payoffs and it gives us the price.

[7] We need to find the equilibrium between:

– generating lots of observations: better precision but higher pricing time

– generating less observations: lower pricing time but less precise

. This steps usually depends on the calculation structures that the bank has access to

Our algorithm is now complete. We have a functional Monte-Carlo pricing method that can give us good approximations of prices for the calculation power that we have at our disposal.

7.3 computational pricing comparison

Recall Barrier
95%

PDI Barrier	50%
Price	93,05%
Sto. Rates Provision	0,00%
Sto. Vol Provision	0,44%
Skew Corr Provision	0,00%
Trading Adjustments	0,00%
Scheme Impact	0,00%
Licence	
Misc. (DV/Listing, etc.)	
TM	
CM	
Reoffer	93,48%

Heston	
Price	94,76

Black Scholes	
Price	102,76

PDI Barrier	60%
Price	90,57%
Sto. Rates Provision	0,00%
Sto. Vol Provision	0,33%
Skew Corr Provision	0,00%
Trading Adjustments	0,00%
Scheme Impact	0,00%
Licence	
Misc. (DV/Listing, etc.)	
TM	
CM	
Reoffer	90,90%

Heston	
Price	89,78

Black Scholes	
Price	98,69

Figure 12: Prices for recall barrier at 95% and PDI barrier at 50% and 60%

Recall Barrier
95%

PDI Barrier	70%
Price	88,92%
Sto. Rates Provision	0,00%
Sto. Vol Provision	0,19%
Skew Corr Provision	0,00%
Trading Adjustments	0,00%
Scheme Impact	0,00%
Licence	
Misc. (DV/Listing, etc.)	
TM	
CM	
Reoffer	89,11%

Heston	
Price	86,22

Black Scholes	
Price	90,77

PDI Barrier	80%
Price	86,07%
Sto. Rates Provision	0,00%
Sto. Vol Provision	0,15%
Skew Corr Provision	0,00%
Trading Adjustments	0,00%
Scheme Impact	0,00%
Licence	
Misc. (DV/Listing, etc.)	
TM	
CM	
Reoffer	86,22%

Heston	
Price	85,09

Black Scholes	
Price	83,37

Figure 13: Prices for recall barrier at 95% and PDI barrier at 70% and 80%

Recall Barrier
100%

PDI Barrier	50%
Price	91,48%
Sto. Rates Provision	0,00%
Sto. Vol Provision	0,41%
Skew Corr Provision	0,00%
Trading Adjustments	0,00%
Scheme Impact	0,00%
Licence	
Misc. (DV/Listing, etc.)	
TM	
CM	
Reoffer	91,89%

Heston	
Price	94,54
Black Scholes	
Price	103,63

PDI Barrier	60%
Price	88,37%
Sto. Rates Provision	0,00%
Sto. Vol Provision	0,36%
Skew Corr Provision	0,00%
Trading Adjustments	0,00%
Scheme Impact	0,00%
Licence	
Misc. (DV/Listing, etc.)	
TM	
CM	
Reoffer	88,73%

Heston	
Price	88,49
Black Scholes	
Price	100,60

Figure 14: Prices for recall barrier at 100% and PDI barrier at 50% and 60%

Recall Barrier
100%

PDI Barrier	70%
Price	86,22%
Sto. Rates Provision	0,00%
Sto. Vol Provision	0,27%
Skew Corr Provision	0,00%
Trading Adjustments	0,00%
Scheme Impact	0,00%
Licence	
Misc. (DV/Listing, etc.)	
TM	
CM	
Reoffer	86,49%

Heston	
Price	84,03
Black Scholes	
Price	87,79

PDI Barrier	80%
Price	84,50%
Sto. Rates Provision	0,00%
Sto. Vol Provision	0,22%
Skew Corr Provision	0,00%
Trading Adjustments	0,00%
Scheme Impact	0,00%
Licence	
Misc. (DV/Listing, etc.)	
TM	
CM	
Reoffer	84,72%

Heston	
Price	82,00
Black Scholes	
Price	77,64

Figure 15: Prices for recall barrier at 100% and PDI barrier at 70% and 80%

Note: prices were made the 23/06/2021 with Total's data on that day. The forward value of the firm at this time at 67.29% for 2026, which is also the maturity year of the autocallables used.

7.4 Comments on the results

Our results are globally consistent regarding the basics of autocalls: as the PDI barrier increases, the price lowers. Comparing our prices with the real prices leads us to think that Heston model is more consistent and precise, and it is totally normal. In fact, Total's stock has a forward close to 60% and from our results, it is were our Heston prices are the most precise. Since the volatility is way less stable approaching the forward, Heston model is much more appropriate in this case as the stochastic volatility will adapt itself. Black-Scholes model tends to have huge differences when the PDI barriers are too close or too far from the forward value. In fact, Black-scholes model is very irregular when the volatility of the product is really different from the input volatility. A very low volatility, like when we are very far from the forward, will be very different from the input volatility, hence it will change the price a lot. On the contrary, a PDI barrier being close to the forward will lead to a very high volatility, much higher than the one in input.

As a conclusion, we can say that our pricer is a good indication of the prices that we can find in a bank. Further, the difference of precision between the two models is explained by the lack of flexibility of Black-Scholes model, that cannot adapt itself to big moves of volatility.

Now that we have made the comparison, we will see some aspects that can explain more in details the difference between the price.

7.5 Provisions: a way to improve the pricing

Choosing the right model was our first step, yet it is the most crucial one. We have seen stock-only diffusion models, and a stock volatility diffusion model. However, we can imagine that all the other parameters can themselves be stochastic. We would therefore need a model that reunites diffusion for all parameters: stochastic rates, stochastic dividends, or even stochastic correlation. In practice, it is extremely hard to create/calibrate a model that takes multi-parameters into account. Only few banks can do it, but the result is not very convincing. The difference of price is largely degrade by the huge complexity of computations.

Thus, banks have created the notion of "provisions", synonym for "impact". Instead of choosing a multi-factor model, we chose our main model, for instance a stochastic spot and volatility model like Heston's. From that, we compute the first price P_1. After that, we compute the price by adding a stochastic parameters to the diffusion, let's say the rates. We compute the price P_2, that takes into account stochastic rates. Our Stochastic Rates Provision will be the difference of the two prices. Yet, it is still an approximation and not usable for professional purposes. We can thus wonder what are the discrepancies between a "professional pricer", and our hand-made pricer. From that, we can see the solutions to bring more precision to our pricer (and when it is possible), or also to reduce the calculation time. The aim of this thesis would more likely be to enhance the precision, since we do not have any time restrictions.

7.6 The funding: what changes our prices a lot

We have noticed that beyond the changes of models and other parameters such as rates, or Heston model parameters, a difference of pricing still exists. For this, we must add to our pricing the notion of funding.
When the client comes to the Equity Department with his notional (100 %), the Equity department will give this amount to the Asset and Liabilities Management department ("ALM"). Having a lot a treasury is a really great added value for the bank. Thus, as an exchange, the ALM will remunerate the Equity Department by giving it a rate and a spread. The rate depends on the currency of the product, and if the note is a swap or not (we will not get into details). For a classic note, the rate is almost surely Euribor3M if the currency is the Euro. For other currencies, it will depend on the funding of the bank: some do not have the treasury to receive other currencies, or do not have the liquidity, and will therefore be funded in other major currencies (usually in Dollars). For instance, if the product is in Swedish kronas (SEK), the bank does not have much advantages to keep some. It will thus realise an FX swap and pay SEK to receive USD at the rate of Libor-USD.
The second component is the most important, because it is what differentiates all the bank during the eternal competition of prices: the funding. As described, the funding is the second part of the remuneration of the ALM, and is different in every bank. For a given period of time, the ALM will pay the rate + funding. In fact, the bank will receive the notional, and keep it as long as the product is not recalled. Therefore, the longer the duration (or maturity), the higher the spread is. This rule can be applied for 95% of the maturities, but for long term maturities, it is a bit different because it can be a constraint to keep money very long time.
We will not be able to compute them directly and impact is as a comparison in our pricer for the following reason. Each bank has a different funding grill, that is changing each period of time (for instance each month). We will therefore not be able to use Crédit Agricole's funding grill, as it has to stay very confidential. It would be useless to use an artificial grill as it would not be coherent as a way of comparison. However, we will compute the impact on our prices using the artificial grill below. On our "real" prices, we will also not use the funding, since the impact would be the same on both pricers.
The numbers are presented in "bips" [8].

[8] 1 bp=0.01%

Duration	Structured products	
	EUR	USD
371 days	4	7
15 months	7	10
18 months	7	14
21 months	10	15
2 years	15	21
3 years	20	27
4 years	34	39
5 years	39	46
6 years	41	50
7 years	42	54
8 years	46	57
9 years	46	54
10 years	45	57
11 years	43	57
12 years	46	55
13 years	43	54
14 years	44	55
15 years	42	52
16 years	43	52
17 years	42	52
18 years	40	53
19 years	37	48
20 years	36	48
25 years	38	50
30 years	37	48
40 years	32	41

Figure 16: Funding grill (artificial)

This funding grill is used to compute the fixed funding: each period of time (depends on the bank, we will take the example of quarterly payments), the ALM will give Euribor3m + fixed funding to the department. The fixed funding is thus an average of all the fundings. In fact, the weights that will be used are the probabilities of being recalled, since the money will leave the ALM when the product is recalled. We thus have the following formula for the fixed funding spread s_f:

$$s_f = \sum_{i=1}^{n} p_t * s_t,$$

where p_t is the probability of recall at time t and s_t is the funding for a maturity t. From this formula, it is thus possible to recover the impact I of the funding spread on our prices, by using the duration $DV01$:

$$I = DV01 * s_f$$

8 Algorithmic pricing - presentation of the pricer

This last part is dedicated to the functioning of the pricer, to explain the code and guide the user to understand how to use it. We first begin by introducing related functions that will be useful for the rest of the pricing. We then present the core functions that are used in pricing: generating paths and using these paths in a pricing function. This coding was highly simplified using Fabrice Douglas Rouah's guide[7] on how to price options using VBA. This books gives methodical techniques on how to improve the convergence speed of the algorithm, and also its simplicity.

8.1 Generating random vectors

We compute Gaussian variables and store them in a matrix:

```
'----- Create a matrix of random gaussian variables -----
Public Function GetAlea(n, m) As Double()
Dim i As Long
Dim j As Long
Dim W() As Double

ReDim W(n, m)

For i = 1 To n
    For j = 1 To m
        Randomize
        W(i, j) = Application.WorksheetFunction.NormSInv(Rnd())
    Next j
Next i
GetAlea = W
End Function
```

The matrix contains $n * m$ variables, which is the number of periods multiplied by the number of paths. This function will be used later for Black-Scholes, but also for Heston where we will have to correlate the vectors.

8.2 Compute the forward

The function returns a vector that contains the forward of a stock, using the repo-dividend-rate formula. The terms of the vector are expressed in %.

```
'----- Compute the forward values of the underlying -----'
Public Function GetForward(s0 As Double, r As Double, d As Double, q As Double, _
                           Freq As Double, Mat As Double) As Double()

Dim i As Integer
Dim dlns As Double
Dim dt As Double
Dim n As Integer
Dim Fwd() As Double
Dim Perf() As Double

dt = Freq
n = Mat / Freq

ReDim Fwd(n + 1)
ReDim Perf(n + 1)

For i = 1 To n + 1  'loop on dates

    If i = 1 Then
        Fwd(i) = s0
        Perf(i) = 0
    Else
        dlns = (r - q - d) * dt
        Fwd(i) = Fwd(i - 1) * Exp(dlns)
        Perf(i) = Log(Fwd(i) / s0)
    End If

Next i

GetForward = Perf

End Function
```

8.3 Generate Black-Scholes Paths

This function takes as arguments the classic parameters of Black-Scholes model, and also the frequency of the paths: this will be very usefull as we want autocalls with different frequencies. It also calls the previous function to generate random gaussian vectors. The function returns the values of one trajectory, and also the performance in %.

```
'----- Compute paths with Black-Scholes diffusion -----
Public Function BS_trajectoires(s0 As Double, Vol As Double, r As Double, d As Double, q As Double, _
                                Freq As Double, Mat As Double, Nbsimul As Double, Alea() As Double) As Double()


'Generates a trajectory matrix andd computes the logs returns

Application.DisplayStatusBar = True
Application.StatusBar = "Simulating trajectories..."


Dim i As Integer
Dim j As Long
Dim dt As Double
Dim dlns As Double
Dim n As Integer
Dim Perf() As Double
Dim Results() As Double

dt = Freq
n = Mat / Freq

ReDim Results(2, Nbsimul, n + 1)   'results(1) = spot / results(2) = perf

For j = 1 To Nbsimul 'loop on simulated vectors
    For i = 1 To n + 1  'loop on dates

            If i = 1 Then
                Results(1, j, i) = s0
                Results(2, j, i) = 1
            Else
                dlns = (r - q - d - (0.5 * (Vol ^ 2))) * dt + Vol * Alea(i - 1, j) * (dt ^ 0.5)
                Results(1, j, i) = Results(1, j, i - 1) * Exp(dlns)
                Results(2, j, i) = Results(1, j, i) / s0
            End If
    Next i

Next j

BS_trajectoires = Results

End Function
```

8.4 Generating Heston Paths

We generates at the same time the stochastic volatility and stock paths, using Euler's discretization and the truncation. The function returns the stock and volatility paths but also the performance in %.

```
Public Function Heston_trajectoires(s0 As Double, Vol As Double, r As Double, d As Double, q As Double, _
                                    Recall_force As Double, Var_avg As Double, Volvol As Double, CorrVolSpot As Double, _
                                    Freq As Double, Mat As Double, Nbsimul As Double, AleaSpot() As Double, _
                                    AleaVol() As Double) As Double()


Dim i As Integer
Dim j As Long
Dim dt As Double
Dim dlns As Double
Dim n As Integer
Dim Perf() As Double
Dim Results() As Double
Dim Var As Double

'Feller Condition
If 2 * Recall_force * Var_avg <= Volvol ^ 2 Then
    MsgBox "La variance n'est pas positive"
    Exit Function
End If

dt = Freq
n = Mat / Freq

ReDim Results(3, Nbsimul, n + 1)   'results(1) = spot / results(2) = perf / results(3) = variance

For j = 1 To Nbsimul
    For i = 1 To n + 1  'loop on the dates
 'loop on the simulated vectors

        If i = 1 Then
            Results(1, j, i) = s0
            Results(2, j, i) = 1
            Results(3, j, i) = Vol
        Else
            'vol diffusion
            Results(3, j, i) = Recall_force * (Vol_avg ^ 2 - Results(3, j, i - 1)) * dt _
            + Volvol * (Results(3, j, i - 1) ^ 0.5) * AleaVol(j, i - 1) + Results(3, j, i - 1)

            Var = Results(3, j, i)

            'spot diffusion
            dlns = (r - q - d - 0.5 * Abs(Var)) * dt + (Abs(Var) ^ 0.5) * AleaSpot(j, i - 1) * dt ^ 0.5
            Results(1, j, i) = Results(1, j, i - 1) * Exp(dlns)

            'levels computations
            Results(2, j, i) = Results(1, j, i) / s0
        End If
    Next i
Next j
Heston_trajectoires = Results

End Function
```

8.5 The pricing function

We now introduce the most important function. This function is independent from the model used, because it directly takes in argument the trajectories. The function also has for arguments the parameters of the autocalls. It can produce the price of Athena Autocalls, where the barrier of coupon is the same as the recall barrier, but also Pheonix Memory autocalls: in this structure, the coupon is paid if the underlying is above a coupon barrier, lower than the autocall barrier. However, the coupon not paid are counted (kept in "memory"), and paid

when the underlying reaches the coupon barrier[9].
The function returns a single price for one trajectory. To discount, we use the Euribor3M value of −0.5432%.

```
Public Function GetPrice(SimPerf() As Double, Recall_Barrier As Double, Pdi_barrier As Double, _
    Cpn As Double, Cpn_barrier As Double, _
                    n As Double, Freq As Double, Nbsimul As Double) As Double()

    Dim i As Long
    Dim j As Long, Recall_count As Long
    Dim k As Integer
    Dim Payoff() As Double
    Dim cpn_adj As Double, nb_coupon As Double, DF As Double
    Dim Res() As Double
    Dim RecallDate() As Double
    Dim Duration As Double

    DF = 1.00544953649

    ReDim Res(3)
    ReDim Payoff(Nbsimul)
    ReDim RecallDate(n)
    Recall_count = 0

    Application.StatusBar = "Computing Price..."

    For j = 1 To Nbsimul    'loop on simulations
        nb_coupon = 0
        For i = 1 To n + 1  'loop on dates

            'Barriere coupon
            If SimPerf(2, j, i) >= Cpn_barrier And i <> 1 Then 'do not take 1 because it is the spot
                nb_coupon = nb_coupon + 1
            End If

            If i > 1 And i <= n Then
            'Early redemption
                If SimPerf(2, j, i) >= Recall_Barrier Then
                    RecallDate(i - 1) = RecallDate(i - 1) + 1
                    Payoff(j) = (100 + (100 * Cpn * nb_coupon * Freq)) * DF ^ i
                    Recall_count = Recall_count + 1
                    Exit For
                End If

            'Redemption at maturity
            ElseIf i = n + 1 Then
                RecallDate(i - 1) = RecallDate(i - 1) + 1
                'Case where it is > recall barrier
                If SimPerf(2, j, i) >= Pdi_barrier Then
                    Payoff(j) = (100 + (100 * Cpn * nb_coupon * Freq)) * DF ^ i

                'Activation of the PDI
                ElseIf SimPerf(2, j, i) < Pdi_barrier Then
                    Payoff(j) = 100 * SimPerf(2, j, i) + (100 * Cpn * nb_coupon * Freq) * DF ^ (i - 1)
                End If

            End If

        Next i

    Next j
```

[9]An Athena is a Phoenix Memory where the autocall barrier equals the coupon barrier.

8.6 Calling the functions and returning the greeks

Now that we detailed how to compute the prices, it remains the final step which is computing the greeks. For that, we only need to compute two prices and variate one of the parameters by a small amount:

```
'--------------- Greeks ----------------------

Dim Delta As Double, Delta2 As Double

Dim Gamma As Double
Dim Vega As Double

Dim Alea1() As Double

If Range("BSGreeks?").Value = "Yes" Then

    Alea1 = GetAlea(n, Nbsimul)

'----- Getting trajectories and variating spot, vol to compute greeks -----
    Simul1 = BS_trajectoires(s0 + 1, Vol, Tx, Dvd, Repo, Freq, Mat, Nbsimul, Alea1)
    Simuld = BS_trajectoires(s0 + 2, Vol, Tx, Dvd, Repo, Freq, Mat, Nbsimul, Alea1)

    Simulv = BS_trajectoires(s0, Vol * (1.01), Tx, Dvd, Repo, Freq, Mat, Nbsimul, Alea)

    BS1 = GetPrice(Simul1, Recall_Barrier, Pdi_barrier, Cpn, Cpn_barrier, n, Freq, Nbsimul)(1)
    BSd = GetPrice(Simuld, Recall_Barrier, Pdi_barrier, Cpn, Cpn_barrier, n, Freq, Nbsimul)(1)
    BSv = GetPrice(Simulv, Recall_Barrier, Pdi_barrier, Cpn, Cpn_barrier, n, Freq, Nbsimul)(1)

    Delta = BS1 - BS0
    Delta2 = BSd - BS1

    Gamma = Delta2 - Delta
    Vega = (BSv - BS0) / 0.1

    Range("Delta").Value = Delta
    Range("Gamma").Value = Gamma
    Range("Vega").Value = Vega

End If
```

What is to notice is that we did not discuss about the model used to compute the greeks and directly used Black-Scholes. In fact, the sensitivities are not impacted a lot by the model. What is interesting to see when we compute the greeks is their sign and an approximation. Using a different model like Heston would have led us to have an impact of maybe a few bips, which is negligible. However, in practice, this step is very important for the hedging costs. When the bank sells a products, the price takes into account the hedging costs, which is basically what the traders need to cover the position of the bank, which is, for example, long a PDI and short digits. The hedging costs are decided regarding the greeks: usually, traders will cover their position by taxing a fixed number of the vega and the epsilon, and sometimes the skew. This of course depends on the maturity, the underlying, and all the other parameters specific to the underlying.

8.7 Vanilla Pricing

Pricing autocallable structures often implies to have a look at the different legs separately. For more simplicity, a vanilla pricer is included, an allows to price a Call, a Put and a PDI. The model used is Black-Scholes, which allows an even lower time of pricing.

```
Public Sub PricingVanilla()

Application.ScreenUpdating = False
Set wsv = ThisWorkbook.Worksheets("Vanille")

    'Getting the imputs
Nbsimul = 100000

ReDim Payoff(Nbsimul)
l = Range("A" & Rows.Count).End(xlUp).Row

For m = 1 To wsv.Range("A" & Rows.Count).End(xlUp).Row - 1

    'Generating random matrix
    Alea = GetAlea(Mat, Nbsimul)

    'Simulating trajectories and calculating performances

    Simul = BS_trajectoires(s0, Vol, Tx, Dvd, Repo, 1, Mat, Nbsimul, Alea)

    Price = 0

    For j = 1 To Nbsimul    'loop on simulations

        If iType = "Call" Then
            Payoff(j) = WorksheetFunction.Max(0, Simul(2, j, Mat + 1) - k)
        ElseIf iType = "Put" Then
            Payoff(j) = WorksheetFunction.Max(0, k - Simul(2, j, Mat + 1))
        ElseIf iType = "PDI" Then
            If Simul(2, j, Mat + 1) <= B Then
                Payoff(j) = WorksheetFunction.Max(0, k - Simul(2, j, Mat + 1))
                    Else
                Payoff(j) = 0
            End If
        End If

        Price = Price + Payoff(j)

    Next j

    Price = Price / Nbsimul

    wsv.Rows(1).Find("MC Price", lookat:=xlWhole).Offset(m, 0).Value = Price

Next m
End Sub


Application.ScreenUpdating = True
```

We basically detailed the most important functions of our pricer. The rest of the code is mostly some formatting, and can be found in the tab "Developer" of VBA. The detailed use of the pricer can be found in the appendix.

Part IV
Conclusion

After presenting the basic bullet and autocallable structures that we can find on the market, we introduced two stochastic models that are used to diffuse the spot and volatility of an asset. Then, we used these models and adapted them to produce an algorithmic pricing method to compute structured products prices. Our comparison led us to identify some differences, and we further analyzed methods to reduce the discrepancies between our prices and prices that can be found on the market.
Monte-Carlo methods are the basics to autocallable and bullet prices, but we also find many products on the market that, in fact, are both autocallable and bullet: we can for instance present the Twin-Win Autocall, which is an Athena Autocall with a Twin-Win at maturity if the product has not been recalled. Monte-Carlo methods are thus in the center of the expansion of pricing methods, and is nowhere near to be replaced by another method, given its flexibility that allows to price every existing payoff.

Bibliography

[1] John Hull. *Options, Futures and other Derivatives.* Pearson, 2017.

[2] Mohamed Bouzoubaa and Adel Osseiran. *Exotic Options and Hybrids: A Guide to Structuring, Pricing and Trading.* Wiley Finance, 2010.

[3] Kevin Krieger. The impact of the COVID-19 pandemic on dividends. *Finance Research Letters*, 2021.

[4] William Feller. Two Singular Diffusion Problems. *Annals of Mathematics*, 54(1), 1951.

[5] Leif Andersen. Simple and efficient simulation of the Heston stochastic volatility model. *Journal of Computational Finance*, 11(3), 2008.

[6] Mark Broadie and Özgür Kaya. Exact Simulation of Stochastic Volatility and Other Affine Jump Diffusion Processes. *OPERATIONS RESEARCH*, 54(2), 2006.

[7] Fabrice Douglas Rouah and Gregory Vainberg. *Option Pricing Models and Volatility Using Excel-VBA*. Wiley Finance, 2007.

Part V

Appendix: quick guide to use the pricer

9 Sheet "main"

This sheet allows the user to compute all the prices. The input are underlined in yellow:

Underlying	
Name	FP FP
Spot	40,36
Vol	5,00%
Taux	-0,54%
Div Yield	5,0%
Repo	1,0%

The name of the underlying was originally coded to receive Bloomberg functions and retrieve the datas by itself. The other component have to be put directly. We then have to indicate the parameters of the autocall:

Autocall	
Recall Barrier	100%
Frequency (y)	1,00
Coupon Barrier	100%
Coupon	5,00%

The frequency can be chosen between 1 (yearly observations), 0.5 (semi-annual observations) and 0.25 (quarterly observations).
We can then choose the parameters of the PDI:

PDI	
PDI strike	100%
PDI Barrier	50,00%

The number of Monte-Carlo simulations can be chosen too:

Models Parameters	
Nb simulations	200000

Finally, we can input the parameters of Heston model (if we do not use Black-Scholes):

Volatility diffusion

Recall Force	1
Average Vol	15,00%
Vol of Vol	2,00%
Correl Vol Spot	-0,6

We can now proceed to pricing, by chosing which model we want to use. On the right, we can find the prices and also the greeks of the products.

Strike Date	vendredi 25 juin 2021
Issue Date	vendredi 9 juillet 2021
Last obs date	jeudi 25 juin 2026
Maturity Date	jeudi 9 juillet 2026

Compute BS Price

Compute Heston Price

Pricer ready

Get recall proba

Refresh Param

Outputs

Black Scholes

Price	103,59
Duration	4,61
Early Redem Prob	10,16%
Sensibilties	
Delta	0,01
Gamma	-0,01
Vega	0,05
Execution time	
Pricing	3,78 sec
Greeks computation	4,64 sec
Calcul Grecques ?	Yes

Heston

Price	95,52
Duration	8,17
Early Redem Prob	66,42%
Sensibiltiés	
Delta	
Gamma	
Vega	
Execution time	
Pricing	30,11 sec
Greeks computation	
Calcul Grecques ?	No

10 Sheet "Data"

This sheet is very simple, and reunites the spot paths, performance, forward and recall probabilities:

Volatility diffusion	
Recall Force	1
Average Vol	15,00%
Vol of Vol	2,00%
Correl Vol Spot	-0,6

11 Sheet "Vanilla"

For this sheet as well, the parameters will have to be input in the yellow boxes.

Underlying	Reference	Spot	Vol	Repo	Div	Rate	Instr. Type	Strike	Barrier	Maturity	Mat (y)	Proxy BS	Close Form	MC Price
SX5E	65654841Z11Z	4120,66	20,00%	-0,43%	3,00%	-0,55%	PDI	4 120,66	60%	25/06/2026	5	41,12%	0,00%	12,38%

Price MC BS