A Baby Pricing Library and Uses in Caps Calibration

Chen Huang*

Columbia MSFE

December 20, 2019

Abstract

This report documents the instructions and practical use in caplets calibration of a C++ pricing library created while studying IEOR 4732 Computational Methods in Finance. This library for now has exported to python, thanks to project *pybind11*. It supports pricing for caplets and floorlets with several one factor classical models. In addition, fast fourier transformations for classical diffusions are also supported, not used in the calibration project though. A detailed introduction of the calibrations with Vasicek, CIR, Longstaff and LMM will be shown firstly. At the same time, the organization of the pricing library will become clear. Then there will be some simple more words for the library itself. The calibration used the data of 11/12/2019 for caplets within 2 year maturity with 4 different strikes. As a result, among all the one factor models used, CIR performs an outstanding matching between the theory and the market. And the behaviors of other models give us guidance on the improvement of single factor CIR. To achieve the further improvement efficiently, the pricing library should also be improved as well.

1 Introduction and Symbols

Caps and floors are popular fixed income derivatives and they provide ways to hedge the fluctuation of interest rate directly. Caps are nothing but combination of caplets, while floors are combination of floorlets. A caplet is a call option on interest rate while a floorlet is a put one. One classical caplet type is the call option on the forward LIBOR rate,usually with underlying asset related to Euro Dollar futures. There is a bias between the forward and future and this is not considered in this practice. One month LIBOR rates are used as forward rates here. And bond prices yield from forward rates. For caplets data, we break caps data with maturity 2 year on 1 month LIBOR into 1 month caplets, assuming that the implied volatility of same strike remains stable.

*email: ch3377@columbia.edu

Our goal is to try to use classical interest rates models to calibrate market caplets prices. There are various of short rates models, HJM models and market models popular for this aim. In this project, we picked short rate models: Vasicek, CIR and Longstaff; and a very simple version LIBOR Market Model. We use explicit form pricing formula under Vasicek and simple LMM. And Monte Carlo Simulation is used under CIR and Longstaff.

Here is the symbols table for those may not be explained further

Symbol	Meaning
r_t	Short rate at time t
$W_t^{\mathbb{P}}$	A Brownian motion in probability measure $\mathbb{P}$
$\mathbb{P}$	Historical measure
$\mathbb{Q}$	Risk neutral measure
$\mathbb{P}_T$	Forward measure with time for T
$\mathcal{F}_t$	The σ algebra with time t in filtration $\mathbb{F}$
$B(t,T)$	Zero coupon bond prices with maturity T at t
$d(t,T)$	Stochastic discounted ratio between t and T: $e^{-\int_t^T r_s ds}$
$\tau(t,T)$	Time interval between t and T with conventions used
$l(t,T,s)$	Simple forward rate from T to s at t
$l(t,T)$	Simple compounded rate from t to T

Table 1: Basic Symbols

2 Caplets Pricing Routine

This section introduces the algorithm used in the library for pricing caplets, together with methods on how to add new one factor models to it. Besides IEOR 4732, this part also quotes some results from this year's term structure course lecture notes and assignment 1&2.

2.1 Zero Coupon Bond Options

The pricing formula for caplets actually can be simply derived from the pricing formula for zero coupon bond option. The formula for zero coupon bond put option with expiry T, maturity s and strike K is

$$\begin{aligned} ZBP_t(T,s,K) &= E^{\mathbb{Q}}[d(t,T)(K - B(T,s))^+|\mathcal{F}_t] \\ &= B(t,T)E^{\mathbb{P}_{\mathbb{T}}}[(K - B(T,s))^+|\mathcal{F}_t] \end{aligned}$$

For caplets with same expiry, maturity, strike K and notion N we have

$$CAPL_t(T,s,N,K) = NE^{\mathbb{Q}}[d(t,T)(\tau(T,s)l(T,s) - K)^+|\mathcal{F}_t]$$

From some steps we can simply derive that

$$
\begin{aligned}
CAPL_t(T,s,N,K) &= N(1+\tau(T,s)K)E^{\mathbb{Q}}[d(t,T)(\frac{1}{1+\tau(T,s)K} - B(T,s))^{+}|\mathcal{F}_t] \\
&= N(1+\tau(T,s)K)ZBP_t(T,s,\frac{1}{1+\tau(T,s)K})
\end{aligned}
$$

We have similar result for floorlets. So we only need to get the pricing formula for zero coupon bond options for different models, then we can get caplets prices.

2.2 Explicit Pricing

From the short rate model of form

$$dr_t = u(t,r_t)dt + \sigma(t,r_t)dW_t^{\mathbb{Q}}$$

We can use the relationship $B(t,T) = E^{\mathbb{Q}}[d(t,T)|\mathcal{F}_t]$ to derive the dynamic of zero coupon bond prices

$$dB(t,T) = B(t,T)(r_t dt + b(t,T)dW_t^{\mathbb{Q}})$$

Where $b(t,T)$ represents the volatility term in bonds prices, which can be determined by the model we use. Once it is know and deterministic. We can generate explicit form zero coupon bond pricing formula.

$$ZBP_t(T,s,K) = B(t,s)N(d_1) - KB(t,T)N(d_0)$$

Where $N(.)$ is normal accumulated probability function and d_1, d_0 are generally known functions related to $b(t,T)$.

In Vasicek model whose dynamic is

$$dr_t = \kappa(\theta - r_t) + \sigma dW_t^{\mathbb{Q}}$$

We have

$$b(t,T) = -\sigma\frac{1-e^{-\kappa(T-t)}}{\kappa}$$

In the library, any short rate model with deterministic bond volatility can use the same explicit zero coupon bond pricer with only one function parameter change. The initial path for building the caplets pricing function $CAPL_t(T,s,N,K)$ for these explicit models is:

1. Create zero coupon bond put option pricer.
2. Create short rate models, ensuring that it has member function to return relative bonds volatility
3. Call zero coupon bond put option pricer with related bonds volatility, with strike $\frac{1}{1+\tau(T,s)K}$, then multiply by $N(1+\tau(T,s)K)$.

In practice, a base class for short rate is created and it asks all short rate models derived from the base must include some members. Here we know the first one is bond volatility $b(t,T)$. For now, we only write in Vasicek to have a deterministic bond volatility, because we only need this one in the calibration project. The library itself is open to any other new models with a deterministic bond vol. The procedure to add it is

1. Create new short rate model derived publicly from base *IShortRate*. Make sure it implements all members needed for explicitly pricing.
2. Add a new static member in *IShortRateFactory* for creating related model pointer without new command.
3. Call the explicit pricer in *ZBC* with this new one to create the bond option pricer for this model.
4. Add new bond put, bond call, caplet and floorlet pricing functions in *pypigs*, which is also the name for the library for now. Also make sure the functions to be interfaced with python added in the *pybind11* macro part.

2.3 Monte Carlo Pricing

For some other short rate models like CIR and Longstaff, the relative bonds volatility are not deterministic functions. Hence we can not easily use the Black-Scholes like formula to do the pricing. Actually, we have a little complex but still explicit form solution to their prices under some cases. We forget about this and try to use Monte Carlo simulation to price the zero coupon bond options and then caplets, which is more easy to derive and understand. It is very costing indeed, but thanks to the efficiency of C++, the calibration process does not cost forever.

Even we do not have the deterministic volatility, the bond price functions with respect to short rate r_t is still available. Therefore, we can simulate bond prices paths by simulating short rates paths. Then we use Monte Carlo to price the zero coupon bond options like

$$ZBP_t(T,s,K) = \frac{1}{N}\sum_{i=1}^{N}(K - B(T,s))^+$$

For CIR model whose dynamic is:

$$dr_t = k(\theta - r_t)dt + \sigma\sqrt{r_t}dW_t^{\mathbb{Q}}$$

The bond price is

$$B(t,T) = e^{\alpha(t,T)-\beta(t,T)r_t}$$

Where

$$\alpha(t,T) = \frac{2k\theta}{\sigma^2}\log\left(\frac{\sqrt{k^2+2\sigma^2}e^{k(T-t)/2}}{\sqrt{k^2+2\sigma^2}\cosh\left(\sqrt{k^2+2\sigma^2}(T-t)/2\right) + k\sinh\left(\sqrt{k^2+2\sigma^2}(T-t)/2\right)}\right)$$

$$\beta(t,T) = \frac{2}{\sqrt{k^2+2\sigma^2}\coth\left(\sqrt{k^2+2\sigma^2}(T-t)/2\right)+k}$$

For Longstaff model whose dynamic is:

$$dr_t = k\left(\frac{\sigma^2}{4k} - \sqrt{r_t}\right)dt + \sigma\sqrt{r_t}dW_t^{\mathbb{Q}}$$

The bond price is

$$B(t,T) = \alpha(t,T)e^{\beta(t,T)r_t+\gamma(t,T)\sqrt{r_t}}$$

Where

$$\alpha(t,T) = \frac{1}{2}\log\left(\frac{2}{1+e^{\sigma\sqrt{2}(T-t)}}\right) + \frac{T-t}{4}\left(\sigma\sqrt{2} - \frac{2k^2}{\sigma^2}\right) - \frac{k^2}{\sqrt{2}\sigma^3}\frac{1-e^{\sigma\sqrt{2}(T-t)}}{1+e^{\sigma\sqrt{2}(T-t)}}$$

$$\beta(t,T) = \frac{2k}{\sigma^2}\frac{(1-e^{\frac{\sigma}{\sqrt{2}}(T-t)})^2}{1+e^{\sigma\sqrt{2}(T-t)}}$$

$$\gamma(t,T) = \frac{\sqrt{2}}{\sigma}\frac{1-e^{\sigma\sqrt{2}(T-t)}}{1+e^{\sigma\sqrt{2}(T-t)}}$$

The initial path to build the caplets pricing function $CAPL_t(T,s,N,K)$ with Monte Carlo in the library is:

1. Create zero coupon bond put option simulation pricer, who depends on the short rate models' property: The dynamic of the short rate. The relationship of bond prices and short rates. The initial short rates.

2. Add more members that must be implemented in short rate models. With those we can get those needed in 1.

3. Price caplets with zero coupon bond put option pricing function.

Here are some comments for these implementation. Firstly, we do use Milstein Scheme for simulating CIR and Longstaff short rate dynamic. Secondly, asking to implement all member functions related to explicitly pricing and Monte Carlo pricing in each model class seems weird. For now, it just returns 0 for useless member functions in a model. These pattern issues could be improved later. Thirdly, sometime the initial short rate for simulation can be easily yield from bond prices(CIR), sometime not(Longstaff). In these no cases, we take r_0 as a model parameter.

Again, the library now only supports CIR and Longstaff. It is open to similar short rate models using Monte Carlo pricing and the procedure to add them is:

1. Add the model class derived publicly from base *IShortRate*. Make sure to correctly implement all members needed for Monte Carlo pricing: relationship between short rate and bond price, dynamic of short rate.

2. Add a new static member in *IShortRateFactory* for creating related model pointer without new command.

3. Call the simulation pricer in *ZBC* with the model pointer to create the bond option pricer for this model.

4. Add new bond put, bond call, caplet and floorlet pricing functions in *pypigs*. Also make sure the functions to be interfaced with python added in the *pybind11* macro part.

2.4 Simple LIBOR Market Model

LIBOR market model is supposed to be the most direct model for pricing caplets and floorlets. For simplicity, we use a very simple version of it.

We assume the volatility function for a forward LIBOR rate with maturity T $\lambda(t,T)$ to be constant. That is

$$dl(t,T,s) = l(t,T,s)\lambda_T dW_t^{\mathbb{P}_{\mathbb{T}}}$$

It is not very realistic to make such simple assumptions. Here we do this as a comparing group for other models. Once we have it, we can simply use Black-Scholes formula for pricng caplets.

Also this makes the number of parameters for calibration be the same as the number of time stages we have. In our case with maturity 2 years on 1 month LIBOR rates, it is 24. Therefore it causes troubles in optimization. We could improve this by using more normal LMM.

3 Calibration

3.1 Data and Parameter Sets

The data used for perform these calibrations are downloaded from website. They are 1 month LIBOR rates and 1 month LIBOR caps with maturity 2 year and notion 25000000. From the caps prices , we can get the implied vol for different strikes that fits better for caplets, under the assumption that the vol does not change for same strike. The results are

Strike	Price	Implied Vol
2	37000	35.32%
2.5	17000	37.77%
3	13000	42.92%
3.5	11000	47.24%

Table 2: 2 Year 1 Month LIBOR Caps Prices

After we have this, we can generate caplets prices with market Black-Scholes models and the 1 month LIBOR rates we have. As we mentioned, discounting bond prices yield from forward rates we have.

We have four models for calibration of the caplets prices. The related parameter sets are

Model	Parameters
Vasicek	(κ, θ, σ)
CIR	(k, θ, σ)
Longstaff	(r_0, k, σ)
Simple LMM	$(\lambda_T)_{T=1}^{24}$

Table 3: Model Parameters

The mission is to find good parameters to make model caplets prices match market prices we generated.

3.2 Target Function and Optimization

The target function for calibration is

$$\min_{\Theta} \sum_{1 \leq i \leq 4, 1 \leq j \leq 24} \omega_{i,j} \| CAPL_t(T_{j-1}, T_j, N, K_i; \Theta) - CAPL_t^M(T_{j-1}, T_j, N, K_i) \|^2$$

Where Θ is the model parameter set, $CAPL_t^M(.)$ is the market prices we have, and $\omega_{i,j}$s are weights we use for optimization.

We do not have a flexible enough parameter set for the one factor model. Fortunately, CIR can somehow well capture the prices we have. Also, it turns out that the calibration tends to match the caplets with low strike and neglect the caplets with high strike. To fix it, we choose the weights to be increasing with respect to strike. And the weights finally used are

Model	Weights
Vasicek	$\omega_{i,j} = K_i^{\frac{5}{2}}$
CIR	$\omega_{i,j} = K_i^{\frac{5}{2}}$
Longstaff	$\omega_{i,j} = 1$
Simple LMM	$\omega_{i,j} = K_i^3$

Table 4: Optimization Weights

3.3 Calibration Results

We do use Nelder-Mead optimization method to find the optimal parameter set. Here we will show the calibration results in figure for all the 4 models, together with the caplets price surface generated from it.

Start with Vasicek, the model prices and market prices are shown in the figure

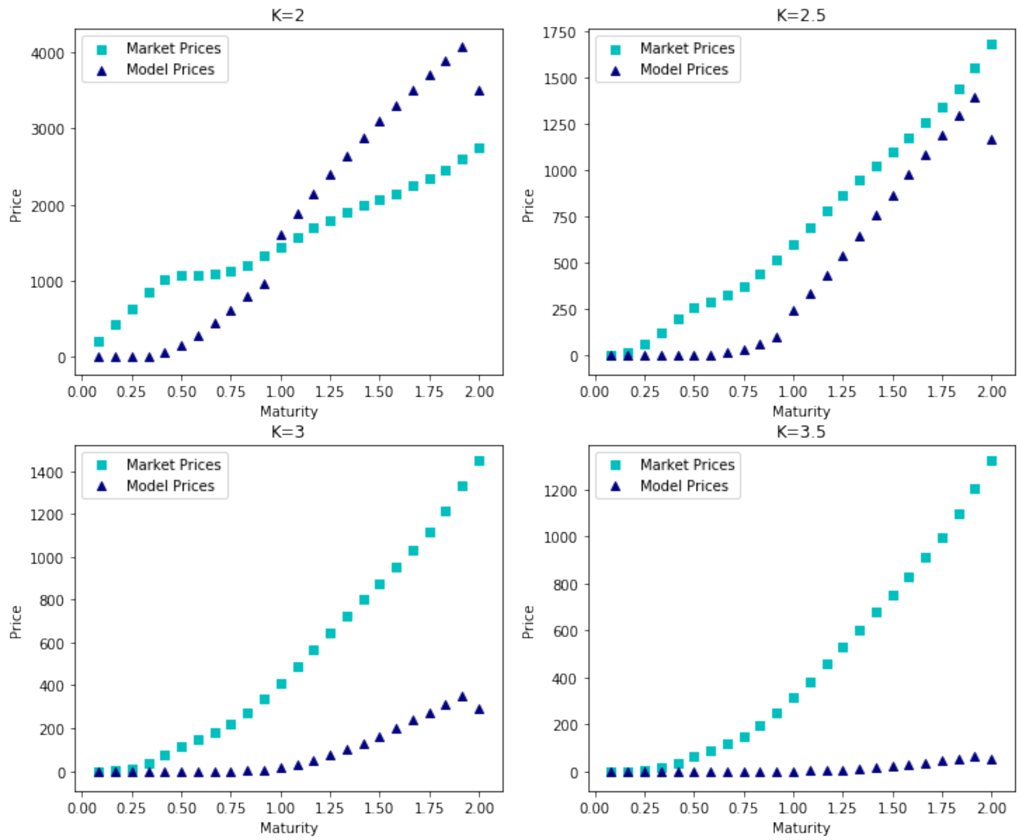

Figure 1: Vasicek Prices with Expiry Time

Here we can see the result is not so good. The shift of the price curve under Vasicek seems to be too naive. Even it tries hard to match the market prices and we put high weight on high strike caplets, there is still a bias between them.

And if we turn to the price shift with respect to strike, we can see the result is still not good. Please see figure 2. As a result, for the market prices under assumptions that implied vols have a flat term structure and a smile behavior to strike. The Vasicek fails to capture the caplets prices very well.

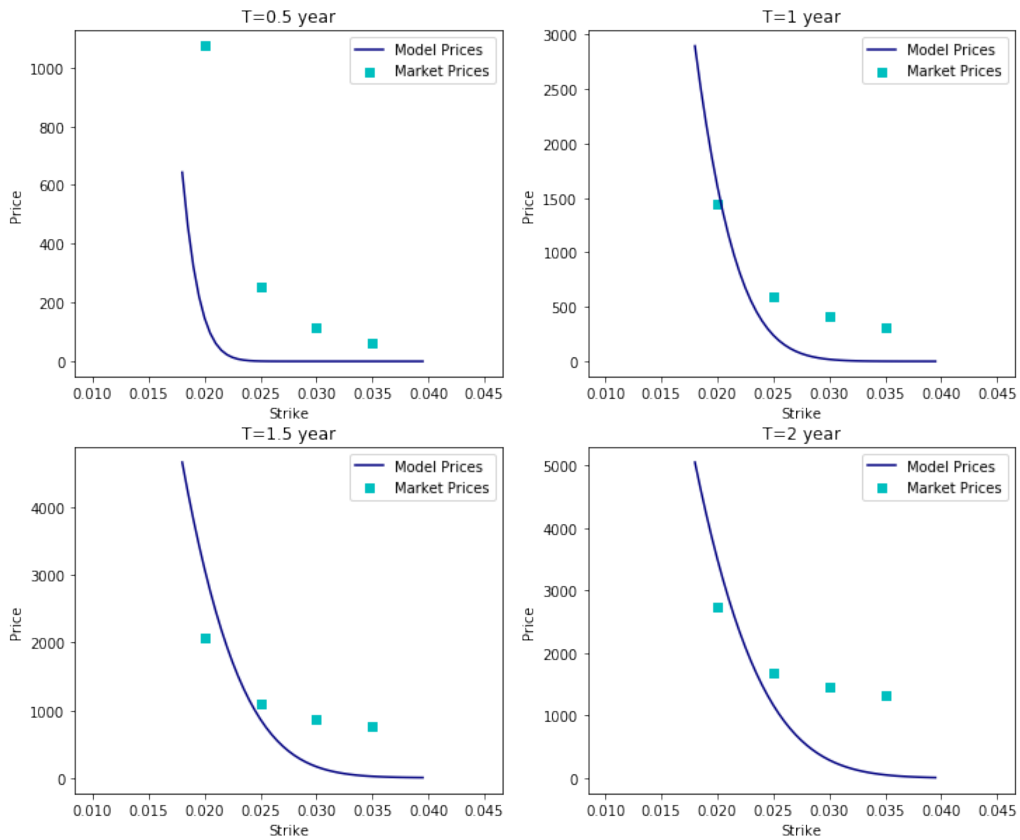

Figure 2: Vasicek Prices with Strike

The caplet price surface we get here is shown below. We can see that the surface is not smooth. And it becomes very close to zero in some place. This is not the ideal one we want for caplets prices.

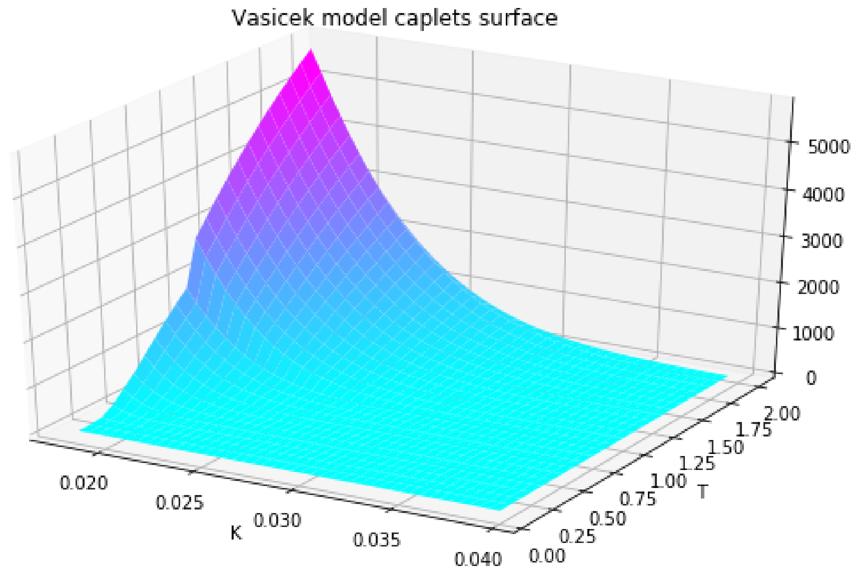

Figure 3: Vasicek Caplet Prices Surface

Now it comes to CIR, the comparing between model prices and market prices we have is shown in figure 4. We can see that CIR does a great work to match the caplets prices here. For different strike, the shape of the price curve can change to match the market price curve very well. Remember that this is only one factor CIR. If we try to use multi factors CIR model, I believe it can have a more beautiful calibration result.

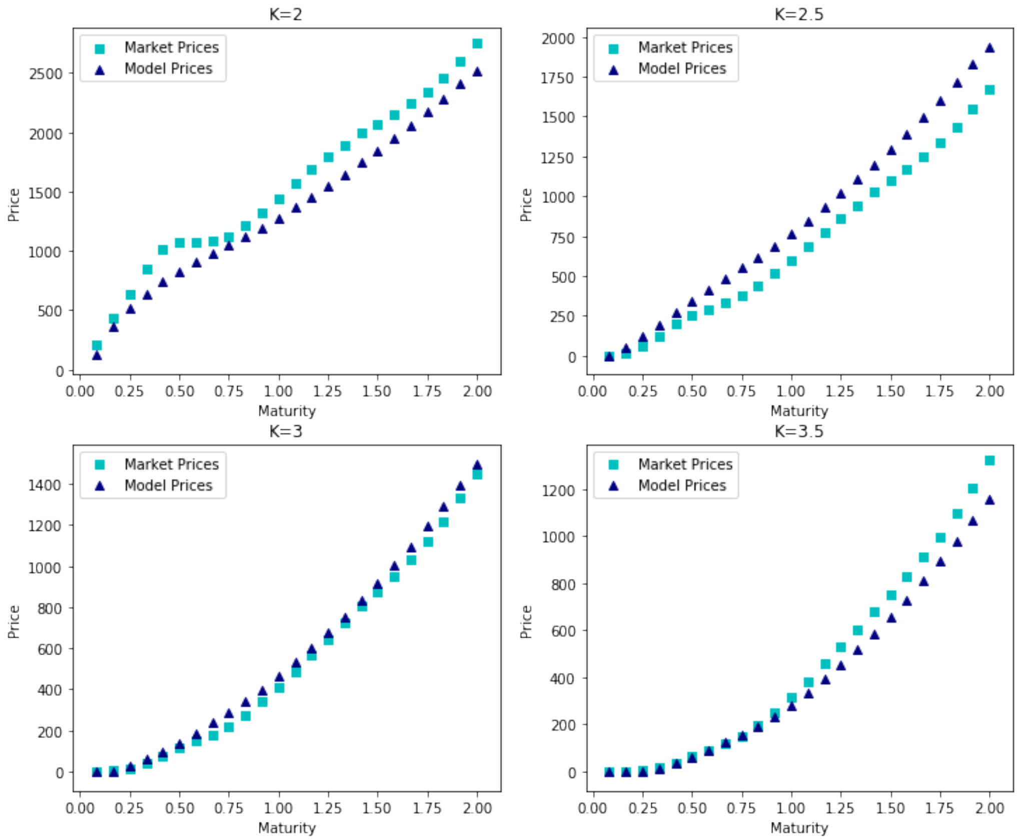

Figure 4: CIR Prices with Expiry Time

Again, we also turn to the relationship between price and strike. This is shown in figure 5 below. We can see that CIR also does a good work for matching the smile behavior. Which as mentioned, it has the special convexity behavior for each strike.

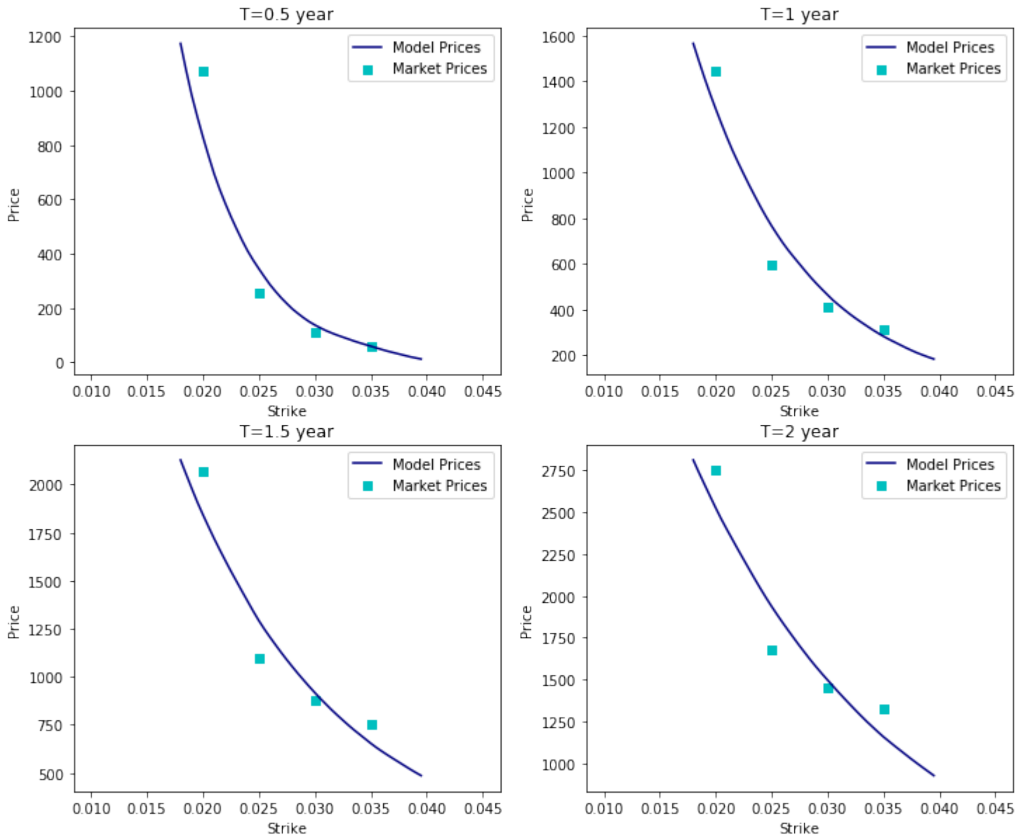

Figure 5: CIR Prices with Strike

The price surface here is also good, as we can see in figure 6. The surface is smooth on each direction and most of the part has effective caplets prices.

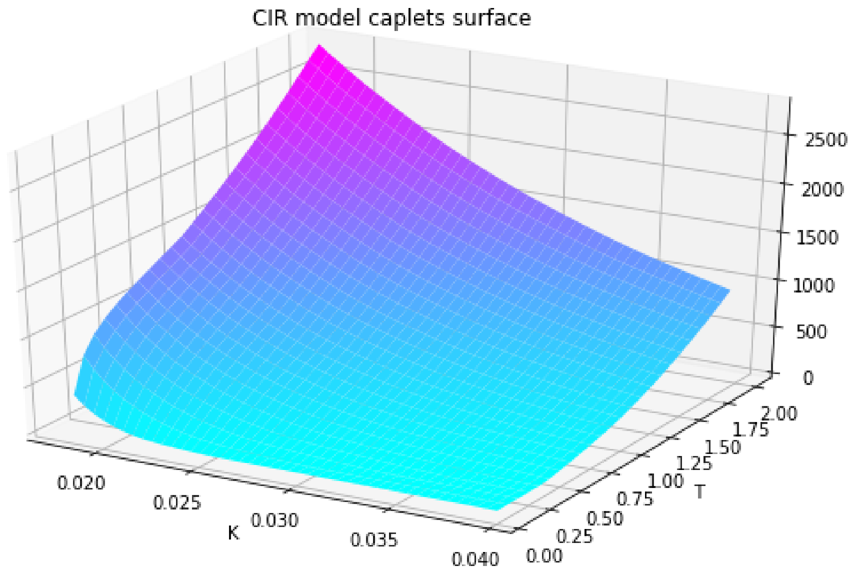

Figure 6: CIR Caplet Prices Surface

One potential problem in the CIR model is the monotonous behavior of price with respect to maturity. We can look figure 4 for strike $K = 2$. The dynamic of market prices is not monotonous but CIR model prices are increasing all the time. This does not affect the fact that CIR is the best model among the 4 in this project, but it raises the motivation to consider Longstaff model.

Longstaff model leads to a non monotonous relationship and somehow can capture the shape of the market prices.

We show this in figure 7 below. It does not match the market prices well, but in some degree it shows the non monotonous behavior. The relationship between price and strike is far from right here. And no figure will show for this. Even Longstaff does not work well in this project, it makes sense to consider combining Longstaff and CIR in a multi factors model. Hopefully it will give us a satisfying result, not included in this one factor project though.

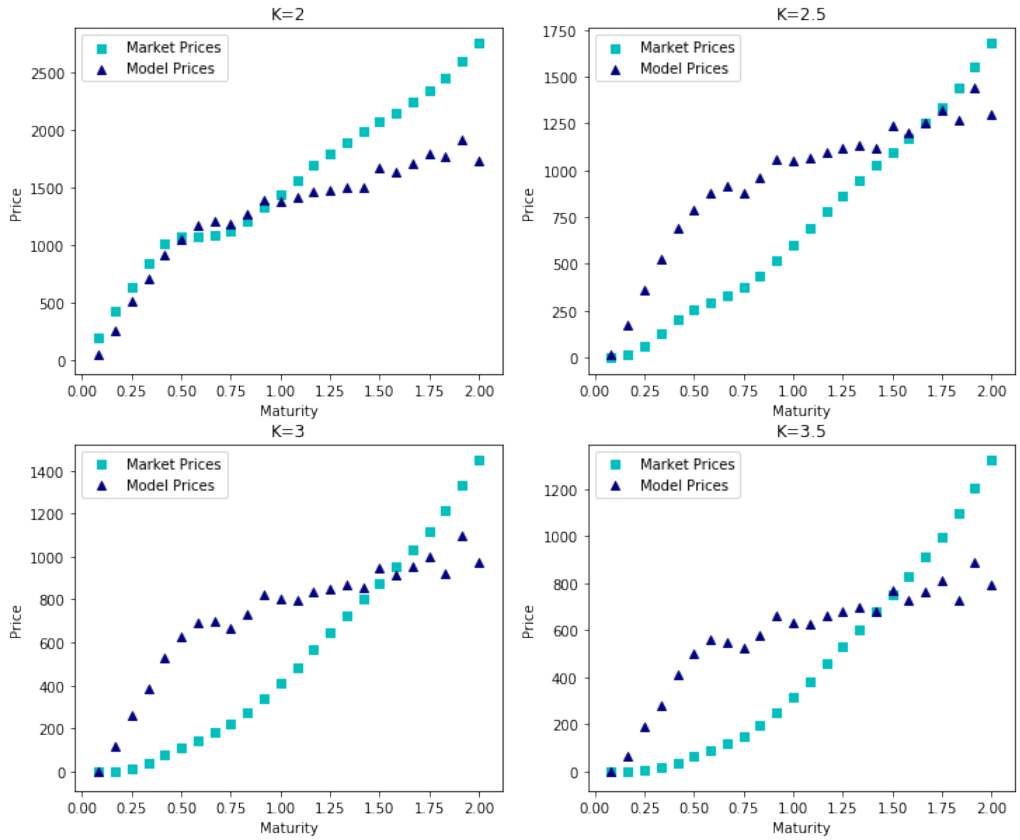

Figure 7: Longstaff Prices with Expiry Time

For comparing aim, we still show the price surface here, in figure 8. We can see this is not so good as the CIR surface. But the wave on it can reflect the non monotonous behavior of price.

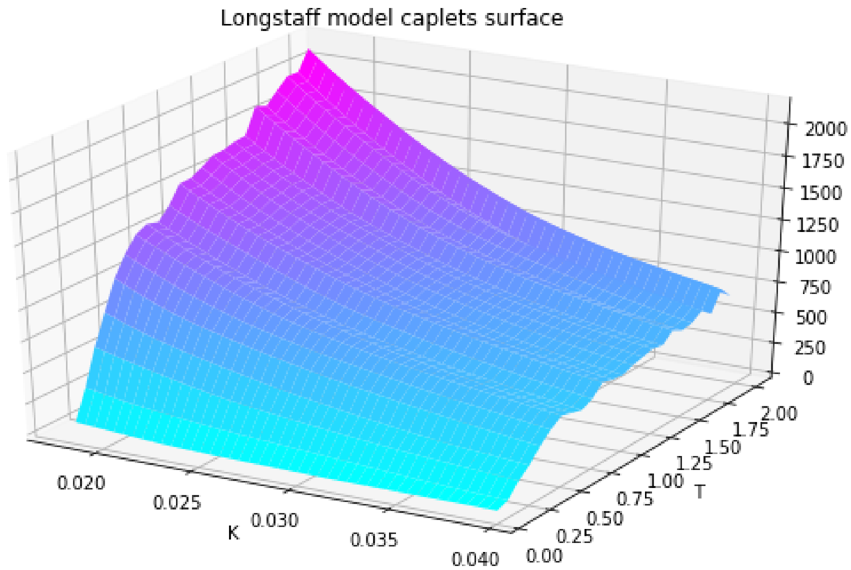

Figure 8: Longstaff Caplet Prices Surface

In addition to short rate models, we consider another class of models modeling the forward LIBOR rate directly. They should work very well with caplets pricing and calibration since they are directly modeling the underlying assets. The result shows that, even with very simple and naive settings, the LIBOR forward model can work for caplets prices calibration. It can capture the non monotonous behavior and not go too far from market prices we have.

Recall that our settings in LMM here is that the volatility of forward LIBOR rate of maturity T is a constant λ_T. The calibration result is below in figure 9.

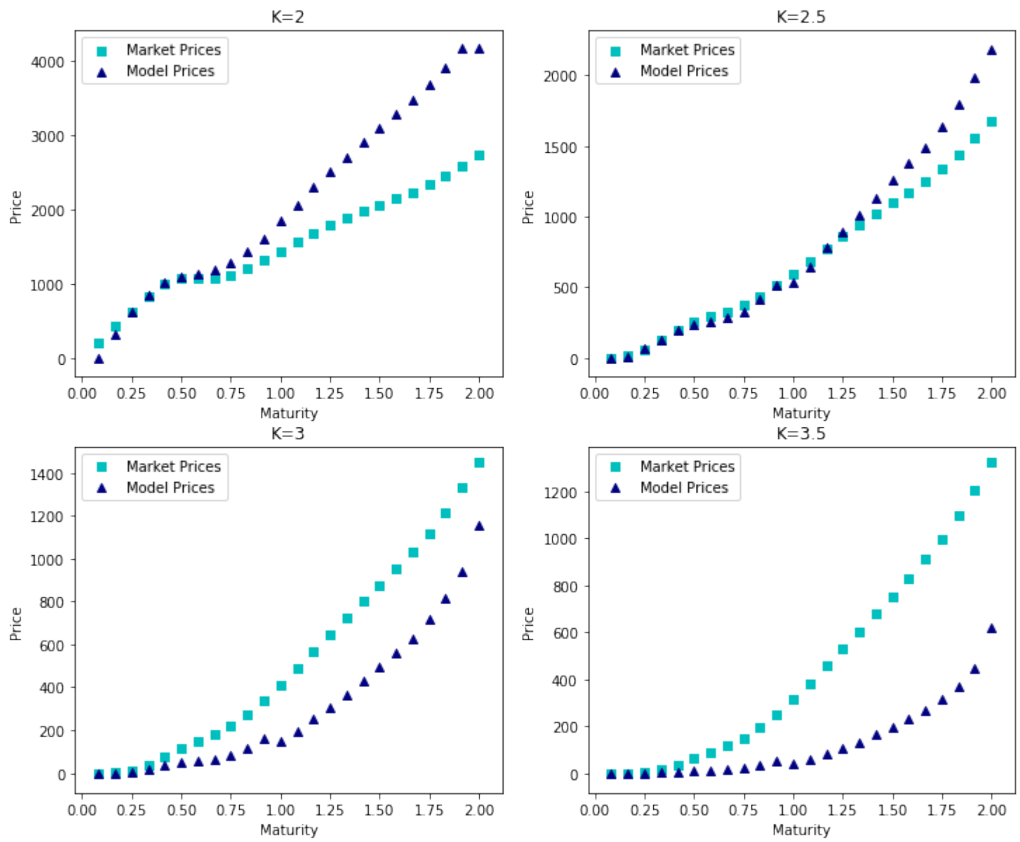

Figure 9: LMM Prices with Expiry Time

The LMM model can generate prices with non monotonous behavior. And compared with Longstaff, this is more close to the market caplets prices, not very well though.

We take a look for the model prices with respect to strike in figure 10 below, and we will see the problem in our settings. In the settings the volatility is deterministic under log normal dynamic, hence it cannot capture the volatility smile shape very well. The shape of the smile remains almost same at the average level. As a result, the price dynamic with different strikes cannot all work well, again it shows some average behavior. As in figure 9, for the 2 lower strike, model tends to over price the caplets, while for the 2 higher strike, model under prices them.

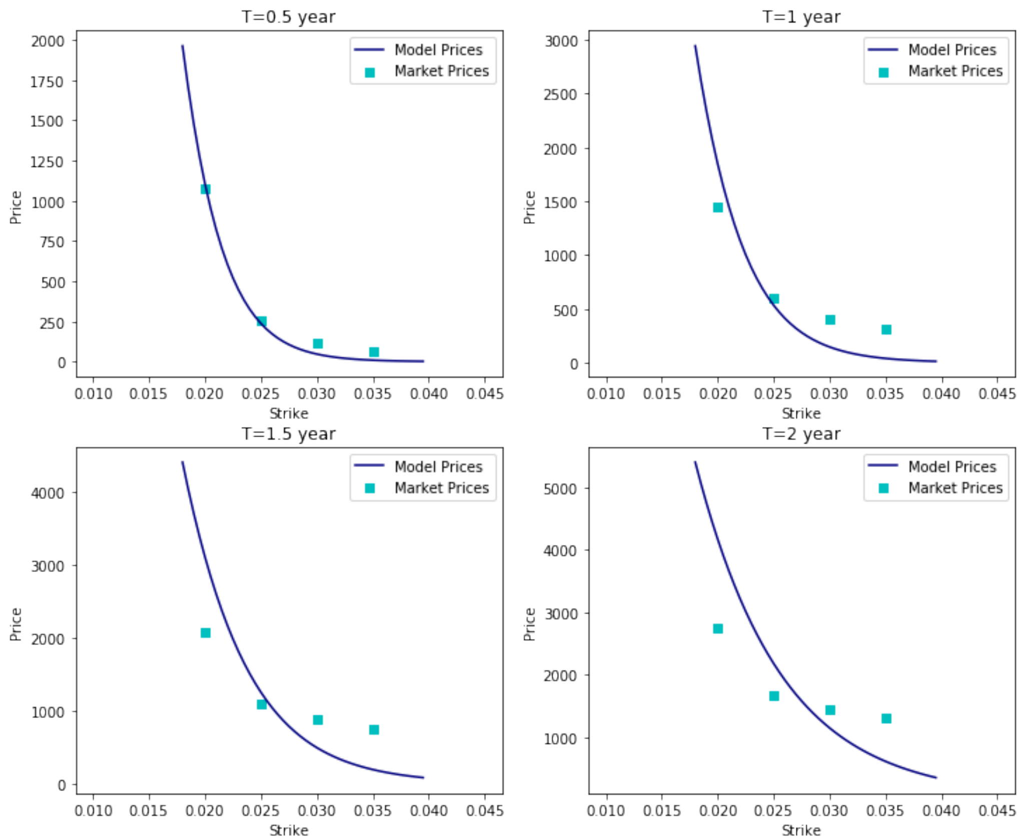

Figure 10: LMM Prices with Strike

The caplets prices surface here is shown in figure 11. It is smooth and non monotonous. The drawback is the change in the K direction cannot match market caplets behavior.

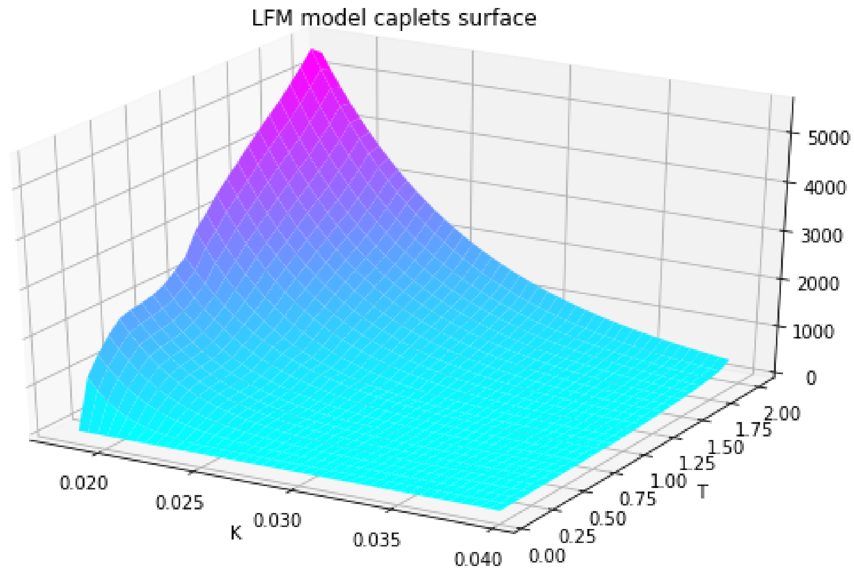

Figure 11: LMM Caplets Prices Surface

3.4 Conclusion and Improvement

Among the 4 models we used, the best one for calibrating caplets given our data is CIR. And we can see that it is not perfect. To improve it, we have following ideas from above section.

Firstly, we can consider multi factors models.

$$r_t = r_t^1 + r_t^2$$

Where r_t^1 is a single factor CIR dynamic and r_t^2 is another single factor CIR or Longstaff dynamic. The correlation ρ could be another new parameter.

Here we will try to provide some thoughts for the pricing under combined Longstaff and CIR. That is to say

$$r_t^1 = \kappa_1(\theta_1 - r_t^1)dt + \sigma_1\sqrt{r_t^1}dW_t^1$$

$$r_t^2 = \kappa_2(\frac{\sigma_2^2}{4\kappa_2} - \sqrt{r_t^2})dt + \sigma_2\sqrt{r_t^2}dW_t^2$$

$$dW_t^1 dW_t^2 = \rho dt$$

Again, we use Monte Carlo and we need the relationship

$$B(t,T) = v(t,T,r_t^1,r_t^2)$$

Then we can derive the PDE for $v(t, T, x, y)$

$$v_t + v_x\kappa_1(\theta_1 - x) + v_y\kappa_2(\frac{\sigma_2^2}{4\kappa_2} - \sqrt{y}) + \frac{1}{2}(xv_{xx}\sigma_1^2 + yv_{yy}\sigma_2^2) + v_{xy}\rho\sigma_1\sigma_2\sqrt{xy} = rv$$

$$v(T, T, x, y) = 1$$

At last, we can try solving this PDE explicitly or by any fancy methods for numerical solutions learned in class to do simulation.

Secondly, we can also consider more complex market models, namely

$$dl(t, T, s) = l(t, T, s)\lambda(t, T)dW_t^{\mathbb{P}_{\mathbb{T}}}$$

Where $\lambda(t, T)$ is a stochastic term.

4 More Words on the Library

Here are some instructions on the library *pypigs*.

Firstly, it is a baby library and only contains necessary functions for this project. But it is open to adding any new models.

To use it in MacOS, just make sure the *.so* file is in the same directory of python scripts, then run import commands in python. Please see the Jupyter Notebook for calibration. Using wheel to package it could make it more convenient for python users. The library was tested in MacOS systems and worked fine. Windows cannot recognize *.so* file so try to rebuild the C++ codes in a **'good'** environment to get new shared library file *.pyd*. For now, the only good IDE in Windows I found is Visual Studio with desktop C++ development tools installed.

To build it, the C++ project was developed in Jet Brain CLion IDE for MacOS and it will also work in Visual Studio for Windows. And all CMake commands for building FFT related things may be commented when submitted, since it depends on the place where fftw library is downloaded. The *.so* file submitted should have all functions.The *pyd.* for Windows only contains those necessary for calibration. The C++ project submitted will build a library without FFT until we uncomment the CMake list commands for FFT.

To be honest, the only thing I can make sure for anyone trying to build it is the library will be fine in MacOS. I tried to build it in Windows with *Cygwin64*, and failed to set the 'good' environment to bind it to python. Please do not try in Windows with any IDE other than Visual Studio 15(or later version) for building *pypigs*, since the GitHub project *pybind*11 also states this and *pybind*11 itself was compiled with MSVC.

In other words, if we do not care about the C++, things will be much easier. Please just open the jupyter notebook for caps calibration in the directory containing both the *.so* and *.pyd* files submitted. And hopefully it will run well in both MacOS and Windows. If that happens, these 2 files will be OK anywhere else in your computer too.