Master in Finanza Quantitativa

Valutazione dei derivati

Docente: M. Marena

Gruppo 05:

De Witt Tommaso
Maironi da Ponte Pietro
Prestidonato Antonio
Sigona Mattia
Taschetta Latteri Saverio

Modello GH – Generalized Hyperbolic

1. Tecniche di pricing delle plain vanilla option con carr-madan

To price a derivative we have different techniques, in particular we focus our attention on Carr-Madan Formula, using algorithm's power (Fast Fourier Transformation).

1.1. Prezzo di un derivato con evoluzione del sottostante con modello GH

We need to find the European Call Option price thanks to Carr-Madan Formula. This formula shows how the analytical solution for a European call option price, and more generally of a European Option, could be obtained thanks to FFT of characteristic function of $\log S_t$.

Given C_t as European Call price expressed by the formula:

$$C_t(k) = e^{-r(T)}\mathbb{E}^Q[(S_T - K)^+] \ (2)$$

This is the payoff of expected present value under risk neutral probability measure.

Now we consider log-prices and related log-strikes, so $x_t = \log S_T$ e $k = \log K$

Then, is true that:

$$C_t(k) = e^{-rT}\int_k^{+\infty} \left(e^x - e^k\right) f(x)\, dx$$

where $f(x)$ is the risk-neutral density of log price.

Idea: if calculate Fourier Transformation of an option price would be possible, If we know this transformation, then doing reversed Fourier Transformation we will obtain option price:

$$\text{if } \exists\, \mathcal{F}(C_t) :\ \exists\, \mathcal{F}^{-1}\big(\mathcal{F}(C_t)\big) = C_t$$

In reality, $\mathcal{F}(C_t)$ is not defined because $C_t(-\infty) = e^{-rT}\mathbb{E}[S_T] = S_0$ but $C_t \notin L^1(\mathbb{R})$, consequently, is it possible make it integrable simply multiplying it for a damping factor α so that $c_t := e^{\alpha k} C_t(k)$, $c_t(-\infty) = 0$ such that $c_t \in L^2(\mathbb{R})$.

Therefore, instead of using a Fourier Transformation of an option price, we use a modified transformation for Call price c_t. That is given by modified call price $c(k) = e^{\alpha k} C(k)$, then $\Psi = \mathcal{F}(c)$ is a Fourier Transformation of modified Call price, that is defined for a fixed α, where a call $\alpha > 0$.

So, we have:

$$\Psi(u) = \int_{-\infty}^{+\infty} e^{iuk}\, c(k)\, dk$$

$\Psi(u)$ is known, then:

$$\Psi = \mathcal{F}(c) \Rightarrow c(k) = \mathcal{F}^{-1}(\Psi) = \frac{1}{2\pi}\int_{-\infty}^{+\infty} e^{-iuk}\ \Psi(u)\, du$$

We can demonstrate that is true:

$$c(k) = \frac{1}{\pi}\int_0^{+\infty} e^{-iuk}\ \Psi(u)\ du$$

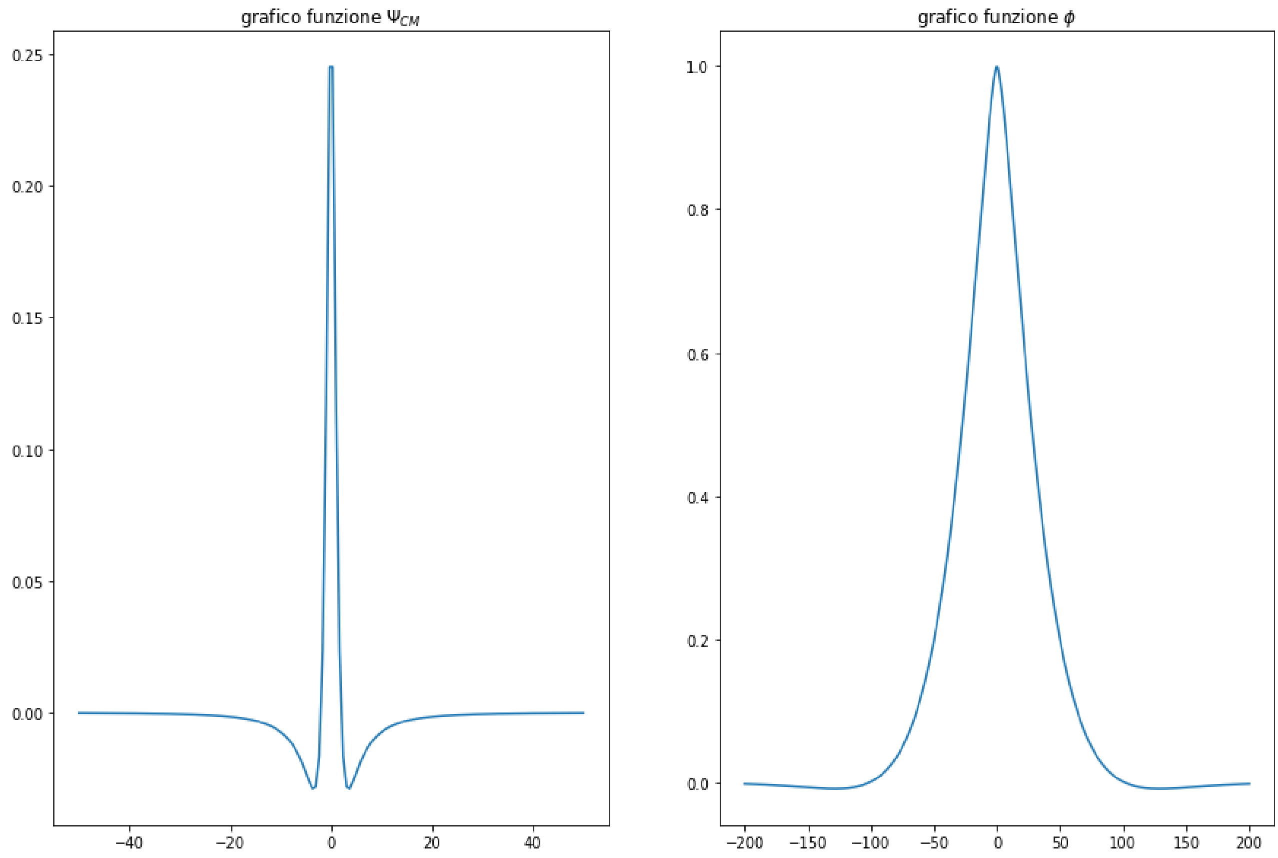
$c(k)$ is the modified Call price, then is true the following relationship:

$$C(k) = \frac{e^{-\alpha k}}{\pi}\int_0^{+\infty} e^{-iuk}\ \Psi(u)\ du \ (3)$$

This is Carr-Madan formula. Where $\Psi(u)$ is:

$$\Psi(u) = \frac{e^{-rT}\varphi(u-(\alpha+1)i}{\alpha^2+\alpha-u^2+i(2\alpha+1)u}$$

This formula depends on φ, that is the characteristic function of $\log S_T$, and depends on α.



1.2. Implementazione di carr-madan via FFT

We want to calculate numerically $I(k) = \int_0^{+\infty} e^{-iuk}\ \Psi(u)\ du$, to do that we need to choose a numerical method. In our case, we want to use Trapezoid method. Therefore, we rewrite the integral as:

$$I(k_n) \simeq \sum_{j=0}^{N-1} e^{-iu_jk_n}\ \Psi(u_j)w_j$$

Then, we will have N points and N-1 ranges. We need to choose the right grid for $u_j = [u_0, u_1,, \dots, u_{n-1}]$ where $u_{n-1} = u_{max}$, and $w_j = [w_0, w_1, \dots, w_{n-1}]$.

With η the range width $u_j = [0, u_{max}]$, where $u_j = \eta j$, with $j = 0, \dots, N-1$; and $u_{max} = (N-1)\eta$. For Trapezoid method, vector w_j will have following values:

- $w_0 = w_{N-1} = 0{,}5\eta$
- $w_1 = \dots = w_{N-2} = 1\eta$

Integral depends on k, so we will have call prices for N levels of k in range $[-b, b]$ with $b > 0$, this range will be centered at the origin because $S_0 = \log 1 = 0$. Log-strike's range k has width $\lambda = \frac{2b}{N}$, whereas strike's grid will be $k_n = -b + \lambda n$, with $n = 0, \dots, N-1$.

So, integral formula is:

$$I(k_n) = \sum_{j=0}^{N-1} e^{-i\eta j(-b+\lambda n)}\, \Psi(u_j) w_j$$

Consequent we have:

$$I(k_n) = \sum_{j=0}^{N-1} e^{i\eta j b} e^{-\eta\lambda j n}\, \Psi(u_j) w_j$$

We have to calculate previous integral N times with $n = 0, \dots, N-1$, we use FFT (Fast Fourier Transformation) algorithm, through which we can solve integral as follow:

$$h_n = \frac{1}{N} \sum_{j=0}^{N-1} e^{-i\frac{2\pi}{N} jn} H_j\,, n = 0, \dots, N-1\,;$$

Fixing $\frac{2\pi}{N} = \eta\lambda$, and $H_j = e^{-\eta\lambda j n}\, \Psi(u_j)(w_j)$ we have:

$$h_n = I(k_n) = \sum_{j=0}^{N-1} e^{i u_j b} e^{-i\eta\lambda j n}\, \Psi(u_j) w_j$$

From this formula we obtain strikes k of FFT; so, we need to come back to market strikes and to market prices through rescaling and separating the value obtained from FFT and from logarithm:

$$C(k_n) = \frac{e^{-\alpha k} e^{-rT}}{\pi} h_n$$

At the end, to obtain market strike prices, we need to interpolate, considering the mistake made choosing λ, if small/large we make a mistake small/large, furthermore we need to find the right trade-off between λ and η. Decreasing mistakes on λ, decrease mistakes on η and vice versa.

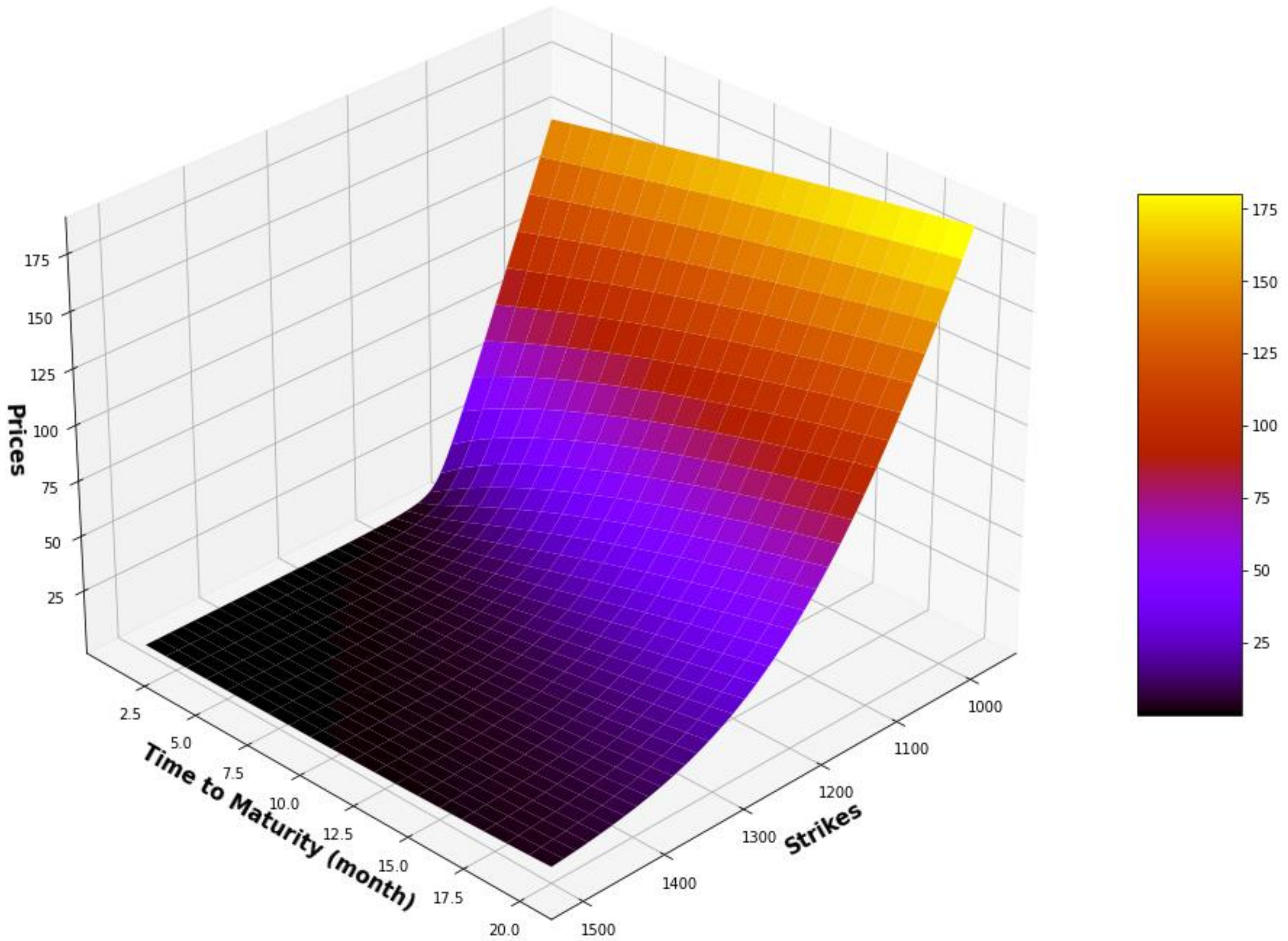
1.3. Esercizio 1

Given a call option, with parameters given by Schoutens:

- $S_0 = 1124.47, r = 1.9\%, q = 1.2\%$
- $T = 1$
- $\alpha = 3.8288$
- $\beta = -3.8286$
- $\delta = 0.2375$
- $v = -1.7555$

We assumed that underlying develops following GH "Generalized Hyperbolic" model. Following information have been obtained using Python coding language.

Using the function **get_price** on Python, we can show the price's surface that has been obtained from input α, β, δ, v given by Schoutens.



This surface shows prices development by varying TTM[1] and strikes.

Goal is to calculate parameters α, β, δ, v, starting from that one given by Schoutens, using RMSE as target function.

Because the difficulties in model implementation, we have chosen to use prices omitting implicit volatility.

Therefore, we want minimize following amount:

$$RMSE(\alpha, \beta, \delta, v) = \sqrt{\sum_{options} \frac{(market\ price - model\ price(\alpha, \beta, \delta, v))^2}{\#options}}$$

In this regard the function used was **scipy.optimize.least_squares** to obtain optimized parameters.

[1] Have been used maturity with one month interval, using months given by columns of price's table in Schoutens file.

Notes about **least_squares** use:

- As starting values vector x_0 we have used the Schoutens parameters vector $(\alpha, \beta, \delta, v)$.
- Constraint $\|\alpha\| > \beta$ is a result difficult to implement within the function **least_squares**, and this constraint can't be removed because values $\|\alpha\| \leq \beta$ bring to a division by zero. Therefore, we have chosen to consider following fixed bounds on α, β

$$lb_\alpha = 3.8288 \, , ub_\alpha = +\infty$$

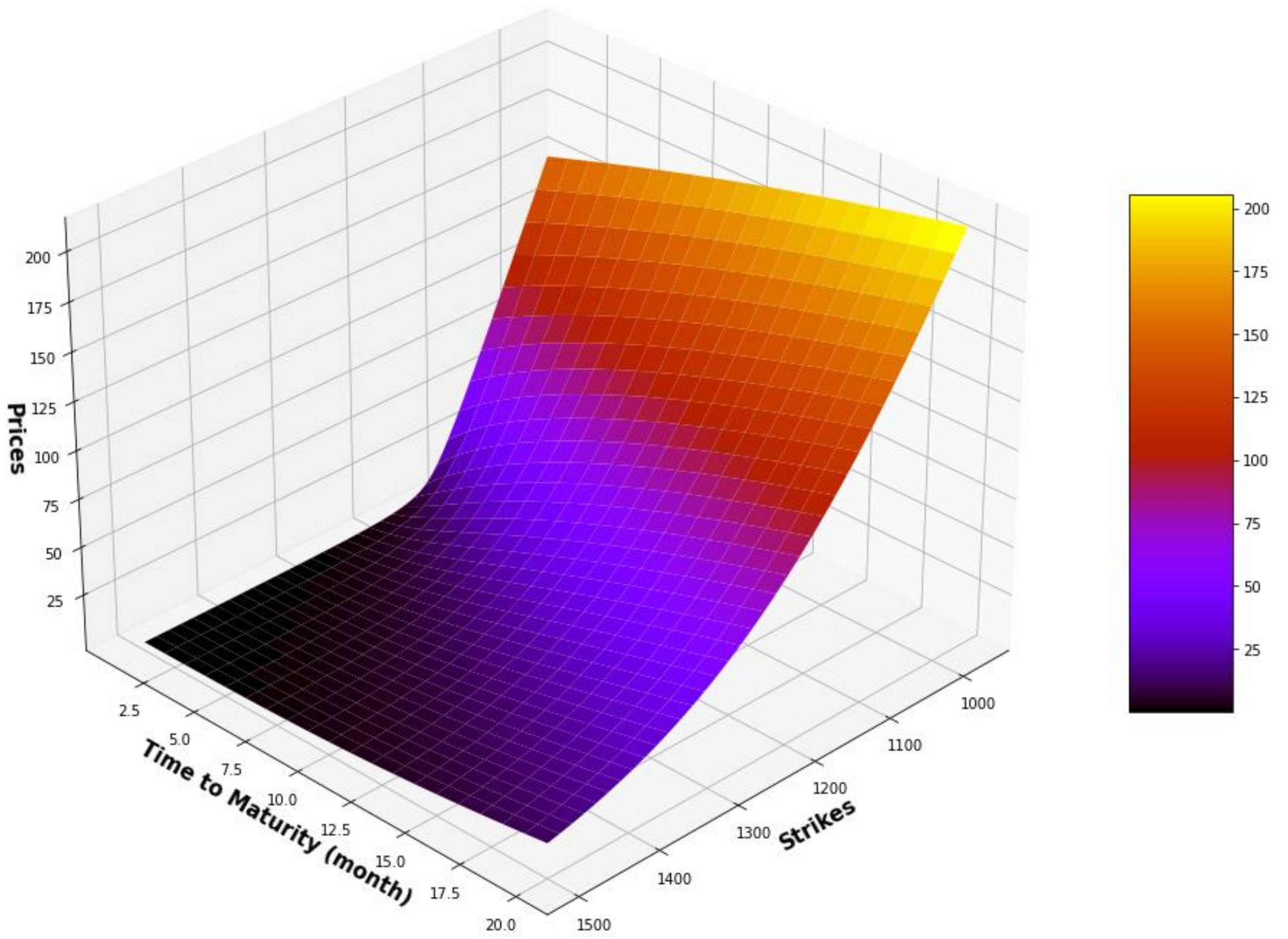
$$lb_\beta = -3.8287 \, , ub_\beta = 3.8287$$

In this way we can use function symmetry EX_psi[2] compared to α and we avoid the condition $\|\alpha\| \leq \beta$. So, in our conditions we can maximize the optimization.

Through the optimization we have following values for calibrated parameters:

$$\alpha = 3.8288 \, , \beta = -3.8287 \, , \delta = 0.37105 \, , v = -2.04555$$

Following is reported the new price surface:



[2] $EX_{psi}(u, \alpha, \beta, \delta, v) = \log\left(\left(\frac{\alpha^2-\beta^2}{\alpha^2-(\beta+i*u)^2}\right)^{\frac{v}{2}} * \frac{K_v\left(\delta\sqrt{\alpha^2-(\beta+i*u)^2}\right)}{K_v\left(\delta\sqrt{\alpha^2-\beta^2}\right)} \right)$

Note how for a fixed given strike, the curve obtained is more sloped compared to that one obtained using Schoutens parameters.

Prices obtained are reported at the end of the paper.

2. Grafico della funzione di densità di probabilità

Now we see how to move from log-price characteristic function to probability density function, using FFT algorithm. From this we obtain cumulative distribution function (CDF), calculating the derivative of this, we obtain the probability density function (PDF).

2.1. Inversione della funzione caratteristica

Given S_T that evolve following GH model, the log-price characteristic function under the risk neutral measure Q is:

$$\varphi_T(u) = \mathbb{E}_t^Q\left[e^{iux_T}\right] = e^{iux_t+iu\left(r_f-q-\psi(-i)\right)(T-t)+\psi(u)(T-t)} \quad (*)$$

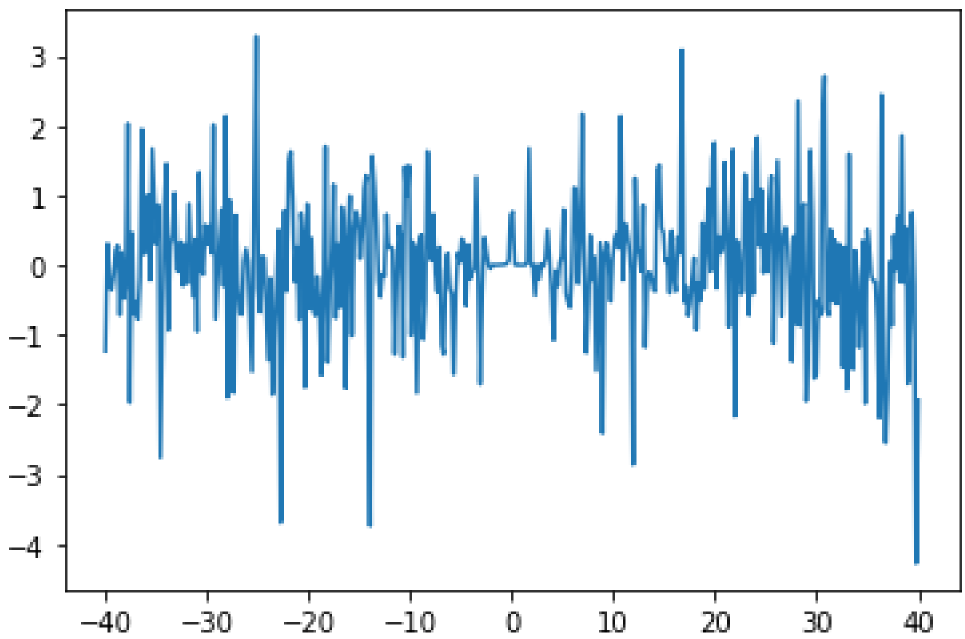
We want obtain from (*) the equivalent CDF, that can be obtained inverting the formula:

$$F(x) = \frac{1}{2} - \frac{1}{\pi}\int_0^{+\infty} Re\left(\frac{e^{-iux}\varphi(u)}{iu}\right) du$$

If $\int_{\mathbb{R}}|\varphi(u)|du < \infty$, so X has the continues probability function and can be recovered from inversion of Fourier Transformation:

$$f(x) = \frac{1}{2\pi}\int_{-\infty}^{+\infty} e^{-iux}\varphi(u)du = \frac{1}{\pi}\int_0^{+\infty} e^{-iux}\varphi(u)du$$

Using different algorithms to calculate the integral, we have obtained some "divide by zero" mistakes within EX_{psi} function, because of presence of modified Bessel formula K_v.



Previous graph is the result of PDF calculus, clearly it can't be correct for several reasons: mainly it can't be negative. To obtain it we have used scipy.integrate.quad function, that permits to integrate in relation to a single variable.

Exactly the message error shown is:

divide by zero encountered in log, return np.log(f1*(f2_num/f2_den))

In spite of CF e surfaces are rights, "divide by zero" error within the logarithm bring us to this graph about PDF.

We have this error for inputs grater than $\|input\| > 2938$ in EX_{psi} function, within we have Bessel function K_v.

Table of first 40 strike prices obtained through calibration

Strike	may2002	june2002	sep2002	dec2002	march2003	june2003	dec2003
975	153,195	156,9104	167,7679	177,9723	187,4648	196,2903	212,2128
980	148,3232	152,1615	163,342	173,8023	183,4993	192,4924	208,677
985	143,4573	147,4234	158,9366	169,6571	179,5599	188,7206	205,1662
990	138,5976	142,697	154,5529	165,5374	175,6472	184,9755	201,6807
995	133,7447	137,9829	150,1918	161,4443	171,7621	181,2578	198,2209
1000	128,8992	133,2823	145,8548	157,3789	167,9055	177,5681	194,7872
1005	124,0616	128,596	141,543	153,3423	164,0781	173,907	191,38
1010	119,2325	123,9251	137,2579	149,3355	160,2808	170,2752	187,9997
1015	114,4129	119,2711	133,001	145,3598	156,5145	166,6734	184,6467
1020	109,6036	114,6352	128,7737	141,4164	152,7802	163,1023	181,3214
1025	104,8056	110,0189	124,5779	137,5066	149,0787	159,5625	178,0241
1030	100,02	105,4241	120,4153	133,6316	145,411	156,0547	174,7554
1035	95,24811	100,8527	116,2877	129,7929	141,778	152,5796	171,5154
1040	90,49149	96,30686	112,1973	125,9918	138,1807	149,1379	168,3048
1045	85,75188	91,78909	108,1462	122,2298	134,6201	145,7303	165,1237
1050	81,03134	87,3022	104,1367	118,5083	131,0972	142,3575	161,9727
1055	76,33229	82,84939	100,1713	114,829	127,613	139,0201	158,852
1060	71,6576	78,43435	96,25247	111,1933	124,1685	135,7189	155,762
1065	67,01074	74,06126	92,38304	107,603	120,7647	132,4546	152,7032
1070	62,39586	69,73495	88,56588	104,0596	117,4027	129,2279	149,6758
1075	57,81807	65,461	84,80406	100,5648	114,0835	126,0394	146,6802
1080	53,28366	61,24586	81,1008	97,12045	110,8081	122,8898	143,7167
1085	48,80055	57,09701	77,45945	93,72813	107,5776	119,7798	140,7857
1090	44,37871	53,02311	73,88354	90,38964	104,3929	116,7101	137,8875
1095	40,03096	49,03422	70,3767	87,10674	101,2552	113,6813	135,0224
1100	35,77409	45,142	66,9427	83,88119	98,16537	110,6941	132,1908
1105	31,63027	41,35987	63,58539	80,71475	95,12444	107,749	129,3928
1110	27,6289	37,70312	60,30867	77,60914	92,13338	104,8467	126,6289
1115	23,80915	34,18889	57,1165	74,56608	89,19312	101,9878	123,8992
1120	20,2228	30,83603	54,0128	71,58724	86,30458	99,17276	121,204
1125	16,93383	27,66434	51,00141	68,67424	83,46864	96,40223	118,5437
1130	14,01197	24,69362	48,08605	65,82864	80,68615	93,6767	115,9184
1135	11,51618	21,94205	45,27024	63,05191	77,95791	90,99666	113,3283
1140	9,464406	19,4239	42,55722	60,34544	75,28468	88,36257	110,7736
1145	7,828764	17,14775	39,94988	57,71052	72,66716	85,77488	108,2547
1150	6,542628	15,1149	37,45071	55,14831	70,10602	83,23398	105,7715
1155	5,531623	13,31841	35,06166	52,65984	67,60183	80,74025	103,3244
1160	4,729983	11,7446	32,78415	50,246	65,15513	78,29402	100,9134
1165	4,085938	10,37469	30,61897	47,90752	62,76637	75,89559	98,53866
1170	3,561649	9,186879	28,56626	45,64495	60,43596	73,54521	96,20033