

FAST APPROXIMATE PRICING FOR FX TARGET REDEMPTION FORWARDS

PETER CASPERS

First Version May 14, 2015 - This Version May 14, 2015

ABSTRACT. We use the ideas of American Monte Carlo methods to develop a fast approximate pricing of FX target redemption forwards. One possible application for this are XVA simulations. We give numerical examples using the QuantLib pricing library.

1. INTRODUCTION

An FX TaRF is a path dependent FX exotic that is usually priced using Monte Carlo methods or partial differential equations. Both approaches are too slow to be applied in a typical simulation for XVA calculations. It is common practice to rely on regression methods to approximate exotic instruments' npvs under market scenarios and time decay. We develop such a framework and test the approximation against full Monte Carlo pricings. We use QuantLib[1] to do the actual computations. A less formal introduction to the topic can be found in the author's blog[2] (search for FX TaRF).

2. PRODUCT DESCRIPTION

A typical FX TaRF works as follows: A bank sells a series of FX call options with a common strike K and foreign notional N_f to a client and buys at the same time FX put options with the same strike and a possibly leveraged notional λN_f back from the client. The option expiry dates are for example monthly spaced with the last option expiry in one year, so we would have a series of 12 options in this case. The options are cash settled with the usual delay of two business days after their expiry, i.e. the i th payout (in domestic units) for the call side is (as seen from the bank)

$$(2.1) \quad -N_f \max(F(t_i) - K, 0)$$

and accordingly for the put side

$$(2.2) \quad \lambda N_f \max(K - F(t_i), 0)$$

where t_i denotes the i th fixing time and F the value of the underlying FX index, expressed in domestic units to be paid for one foreign unit. If the total sum of payouts exceeds a given target S at the i th fixing, i.e.

$$(2.3) \quad \sum_{j=1}^i \max(F(t_i) - K, 0) \geq S$$

the structure terminates, i.e. all coming options are killed without generating any further payments. If this event occurs, the i th payout itself may take place as a full coupon payment, a capped coupon payment (such that the target is exactly reached) or no coupon payment for the i th fixing at all.

Since obviously the upside potential for the client is limited by the target while the downside is not, and the downside being possibly leveraged on top, the strike will be in the money from the client's perspective to get a fair structure at trade date.

There are further complexities that can be introduced in the structure like a (european or american) knock in feature with an in the money barrier for the puts. We do not discuss these additional features in the following.

3. EXAMPLE TERMS

The trade date is October 13, 2014. The reference Spot EUR-USD is 1.2640. The payment schedule is displayed in table 1. The strike K is set to 1.2100, the target level is $S = 0.25$. In case of a target event the current coupon is paid partially to exactly match the target. The leverage is $\lambda = 2$.

TABLE 1. Example schedule for a FX TaRF, the column Calendar Days contains the number of days from the trade to the fixing date

Fixing Date	Payment Date	Calendar Days
November 13th, 2014	November 17th, 2014	31
December 11th, 2014	December 15th, 2014	59
January 13th, 2015	January 15th, 2015	92
February 12th, 2015	February 17th, 2015	122
March 12th, 2015	March 16th, 2015	150
April 13th, 2015	April 15th, 2015	182
May 13th, 2015	May 15th, 2015	212
June 11th, 2015	June 15th, 2015	241
July 13th, 2015	July 15th, 2015	273
August 13th, 2015	August 17th, 2015	304
September 11th, 2015	September 15th, 2015	333
October 13th, 2015	October 15th, 2015	365

4. VALUATION MODELS

From the common literatur on target redemption structures we know that the valuation depends on the n dimensional common distribution of the underlying fixings

$$(4.1) \quad (F(t_1), \dots, F(t_n))$$

or in other words on

- (1) the (entire) FX smile on expiry times t_i representing the marginal distribution of the underlying, and
- (2) the intertemporal correlations of the underlying fixings on the different t_i

In this paper we will start with a standard Garman-Kohlagen model with constant volatility (i.e. in particular a flat smile and no explicit control over the correlation structure).

The hope is that the methods developed for this case will also be applicable to more involved models. A final answer to this is due to future work and tests.

5. REGRESSION ANALYSIS

We fix the setup by specifying the model to be a Garman-Kohlagen process with $\sigma = 20\%$ lognormal volatility, $r_f = 1\%$ foreign (EUR) rate and $r_d = 2\%$ domestic (USD) rate. The process for the underlying in the domestic bank account measure thus reads

$$(5.1) \quad dF = (r_d - r_f)Fdt + \sigma FdW$$

The goal is to compute future npvs

$$(5.2) \quad \nu(t) = E \left(\sum D(t, p_i) \Pi(F(t_i)) \middle| \mathcal{F}_t \right)$$

with D the discount bond, p_i the i th payment time, Π the TaRF payoff and \mathcal{F}_t the filtration at time t . As it turns out it is convenient to compute forward npvs ν' as of the last payment date p_n and then discount them back to t when an approximation for $\nu(t)$ is required.

The filtration in our model setup is effectively generated by the underlying F ; rates are static as well as the volatility. Note that when using $\nu(t)$ in a XVA simulation, this npv is consistent with the model implied rate and volatility structure 5.1, i.e. should the XVA scenarios come with an own rate or even volatility different from 5.1, this is ignored when retrieving ν , which creates an inconsistency in the setup.

5.1. Main ideas. Our strategy is now to use the data generated during a Monte Carlo simulation to compute $\nu(0)$ (as it would be used for a pricing of the structure as of today) and derive a deterministic function Φ taking the valuation time t , the already accumulated amount s and the current FX spot $F(t)$ and returning the desired (forward) npv $\nu'(t)$.

$$(5.3) \quad \nu'(t) \approx \Phi(t, s, F(t))$$

We now specify the functional form of Φ . This is inspired by looking at generated data for several valuation times t and scenarios for the already accumulated amount s and the FX spot $F(t)$ and tests against full monte carlo pricings under these scenarios. See [2] for some motivation.

Below we will introduce some further refinements to the function Φ , here we present only the main direction.

In time dimension t we choose Φ to be piecewise constant between the fixing times t_i . Although there is a dependency of ν on t even within $[t_{i-1}, t_i)$ we ignore

this since it is rather mild (as the numerical examples below will prove). We denote $T_i := [t_i, t_{i+1})$ with $t_0 := 0$.

For the s dimension we bucket the whole possible interval $[0, S]$ into n equally sized pieces

$$(5.4) \quad S_0 = [0, S/n), S_1 = [S/n, 2S/n), \dots, S_{n-1} = [(n-1)S/n, S)$$

For a start we assume Φ to be piecewise constant on each S_i as well. In the numerical examples below we use $n = 5$.

On each segment $T_i \times S_j$ we parametrize $\Phi = \Phi(F)$ now being only dependent on F as

$$(5.5) \quad \Phi(F) := \begin{cases} a_1 F^2 + b_1 F + c_1 & x \leq c \\ a_2 F^2 + b_2 F + c_2 & x > c \end{cases}$$

dropping the dependency on (i, j) in this notation.

We call c the cutoff point and estimate two quadratic polynomials using the data sets

$$(5.6) \quad D_1 = \{(F, \nu') | F \in (-\infty, c]\} \text{ and } D_2 = \{(F, \nu') | F \in (c, \infty)\}$$

c is determined by asking for $|D_2| \approx 0.2|D_1 \cup D_2|$.

5.2. Refinements. We apply certain refinements to the scheme described above:

- (1) It turns out that within the buckets for the already accumulated amount S_i the npv varies significantly due to the different upside potential for future coupons (in case of a capped coupon type) and the different distance to the target event. The results can be improved by assuming the value for Φ as computed above to be valid for the midpoint of each bucket S_i and then linearly interpolate them in s direction. Below the first midpoint of S_0 and above the last midpoint of S_{n-1} we extrapolate linearly as well, using the first resp. last slope. In the implementation the interpolation can be switched on and off by a flag `interpolate`. Below we always use this interpolation option.
- (2) To make sure that in each bucket S_i is enough data to do the regression we require to have a certain ration of the total number of generated data points in each bucket and merge neighboured buckets in case this criterion is violated. Below we require this ratio to be 10%.
- (3) Similarly if the spot segmentation D_1, D_2 leads to a set D_2 with too few data points we amend the cutoff ratio of 0.2 until D_2 is big enough. Below we require D_2 to have at least the size $0.2 \cdot 0.25 \cdot |D_1 \cup D_2|$, otherwise we iteratively multiply the initial ratio of $1 - 0.2$ for D_1 by 0.99 until D_2 has the desired size.
- (4) For low spot values the quadratic function is replaced by a linear function. The linear function has the same value and slope as the quadratic function at the glue point c_l , which is defined such that only a certain ratio (below we use 0.05) of the data points lies left from this point.
- (5) We only use the ascending branches of the parabolas and extrapolate them flat from their extreme point.

- (6) We allow the cutoff point c to be moved within a certain range around its original value (below we use a relative threshold of 5%) to meet the intersection point of the two parabolas. This smoothes the resulting approximation function.

6. NUMERICAL EXAMPLES

We give numerical examples using [1]. We refer to the example terms and market data given above.

We use 50 time steps per year for the simulation and 40000 paths for the Monte Carlo simulation. We run the test cases described in table 2

TABLE 2. Test cases for the comparison of proxy pricing and full monte carlo pricing, the original trade date is shifted forward by dayshift calendar days, an accumulated amount of acc is assumed

Test case #	dayshift	acc	rationale
1	15	0.0	all fixings open
2	36	0.0	one fixing, start of coupon period
3	45	0.0	one fixing, middle of coupon period
4	58	0.0	one fixing, end of coupon period
5	75	0.0	two fixings, no accumulated amount
6	75	0.01	two fixings, lower end of 1st acc bucket
7	75	0.025	two fixings, middle of 1st acc bucket
8	75	0.04	two fixings, upper end of 1st acc bucket
9	220	0.0	seven fixings, no accumulated amount
10	220	0.10	seven fixings, 0.1 out of 0.25 target reached
11	220	0.23	seven fixings, 0.23 out of 0.25 target reached
12	320	0.04	two fixings open, 0.04 out of 0.25 target reached
13	320	0.09	two fixings open, 0.09 out of 0.25 target reached
14	320	0.11	two fixings open, 0.11 out of 0.25 target reached
15	320	0.18	two fixings open, 0.18 out of 0.25 target reached
16	320	0.21	two fixings open, 0.21 out of 0.25 target reached
17	320	0.23	two fixings open, 0.23 out of 0.25 target reached
18	350	0.10	one fixing open, 0.10 out of 0.25 accumulated
19	350	0.24	one fixing open, 0.24 out of 0.25 accumulated

The results per test case are displayed in a range of $[0.60, 1.75]$ for the spot. The black horizontal line marks the region where 99.8% of the generated data points lie, i.e. we cut off the 0.1% biggest and smallest values of simulated spot values. The circles represent full Monte Carlo prices, the line the proxy pricing.

We can not expect to be able to extrapolate outside this region too much with high precision, because that would mean that we can extrapolate the price to market scenarios that were never seen in the simulation. However if the XVA simulation uses a comparable dynamics (volatility and drift) as the pricing model itself, this will not cause huge problems: The low ratio of paths walking outside the trusted region (which would be 0.2% in our example) will probably not affect the result much. Also in the XVA simulation typically fewer paths will be used compared to the pricing. Finally, the extrapolation outside the trusted region is still reasonable and not producing too extreme errors in our experiments.

7. OUTLOOK / OPEN POINTS

How does the method developed in this paper translate to

- (1) models including a smile and different correlation structures ?
- (2) product features like knock ins ?

REFERENCES

- [1] QuantLib A free/open-source library for quantitative finance, <http://www.quantlib.org>
 [2] Fooling around with QuantLib, <http://quantlib.wordpress.com>

FIGURE 1. Test case #1 (all fixings open)

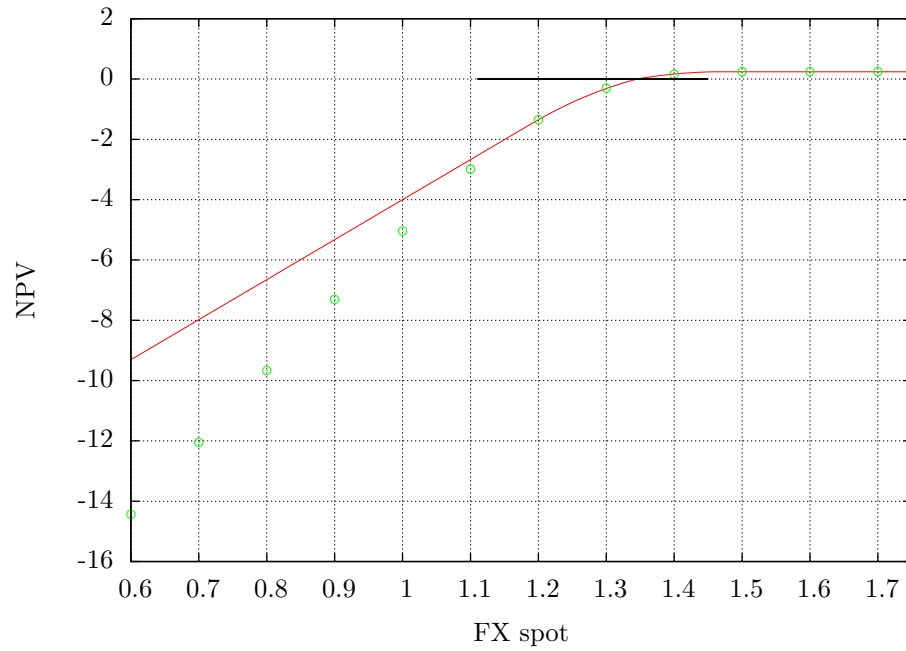


FIGURE 2. Test case #2 (one fixing, start of coupon period)

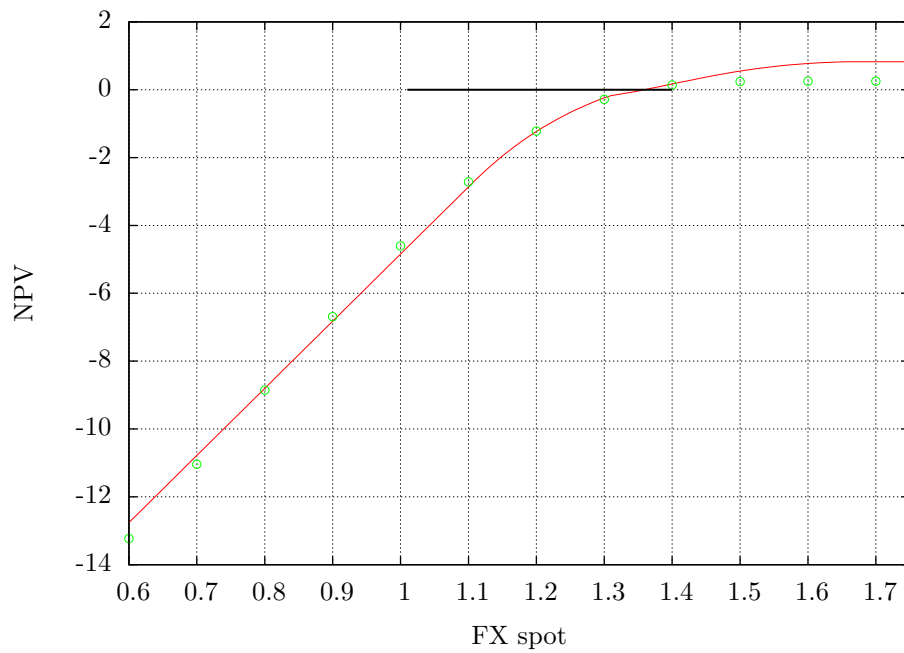


FIGURE 3. Test case #3 (one fixing, middle of coupon period)

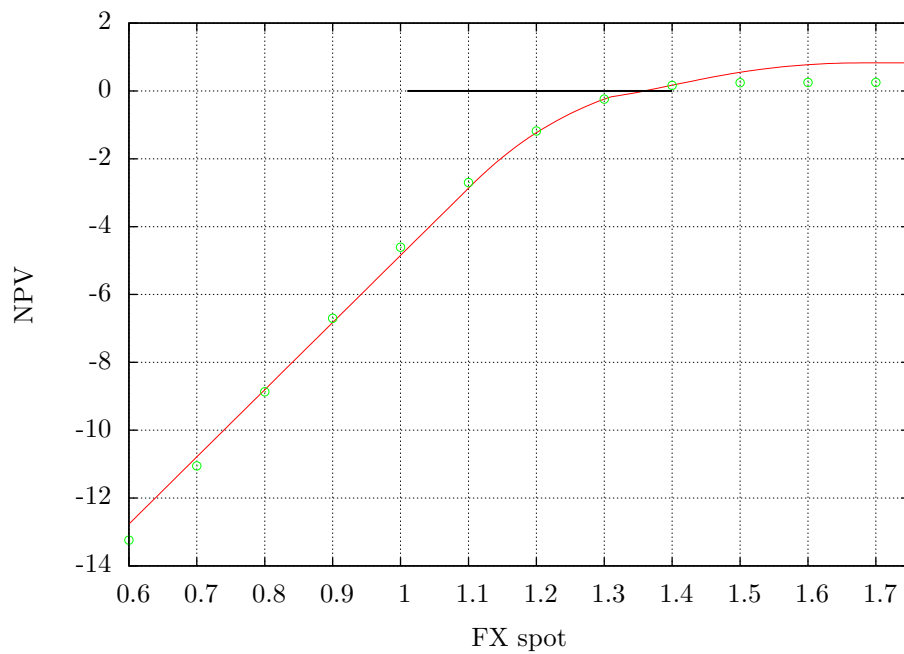


FIGURE 4. Test case #4 (one fixing, end of coupon period)

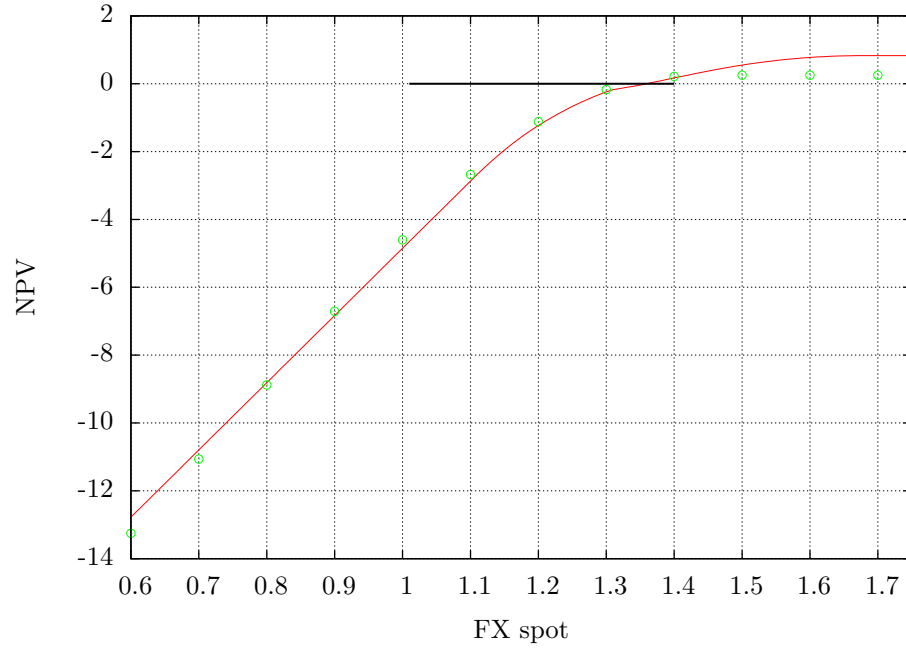


FIGURE 5. Test case #5 (two fixings, no accumulated amount)

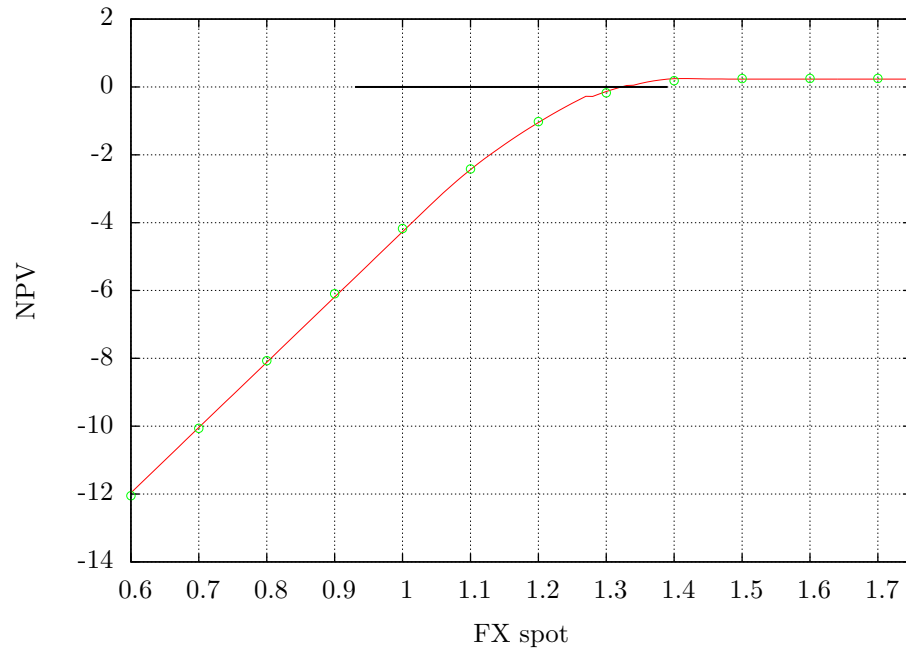


FIGURE 6. Test case #6 (two fixings, lower end of 1st acc bucket)

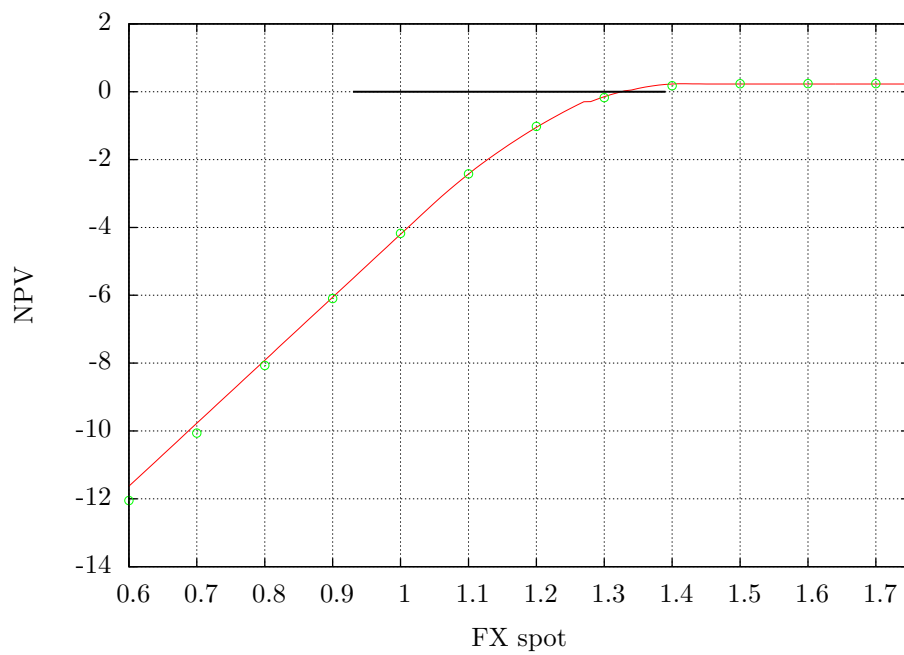


FIGURE 7. Test case #7 (two fixings, middle of 1st acc bucket)

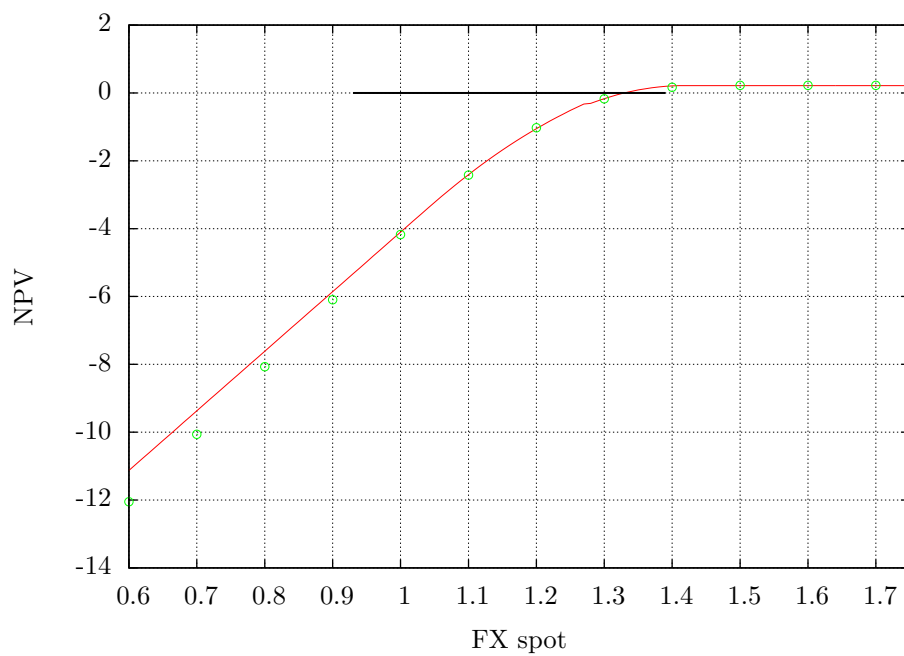


FIGURE 8. Test case #8 (two fixings, upper end of 1st acc bucket)

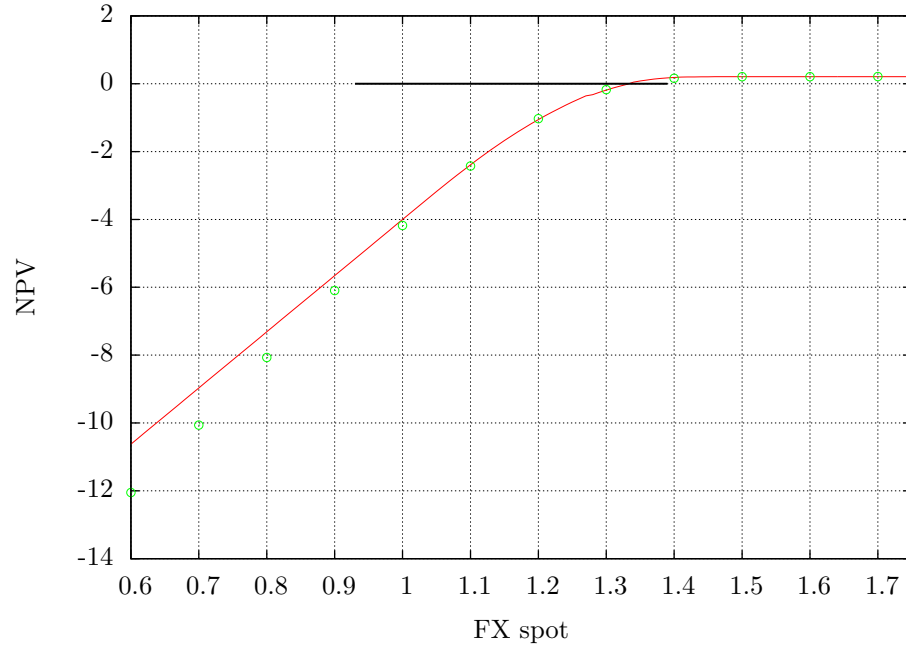


FIGURE 9. Test case #9 (seven fixings, no accumulated amount)

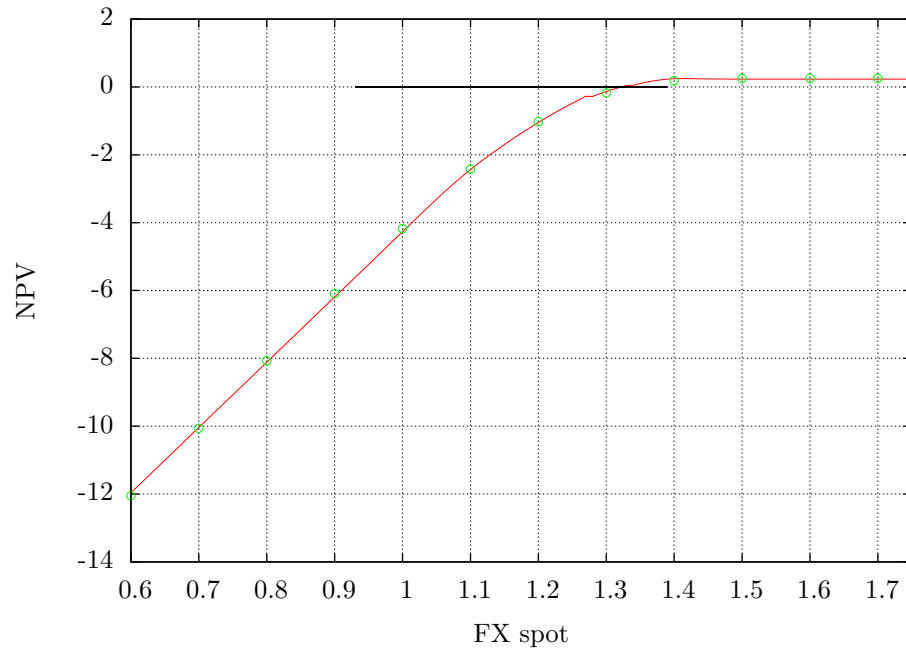


FIGURE 10. Test case #10 (seven fixings, 0.1 out of 0.25 target reached)

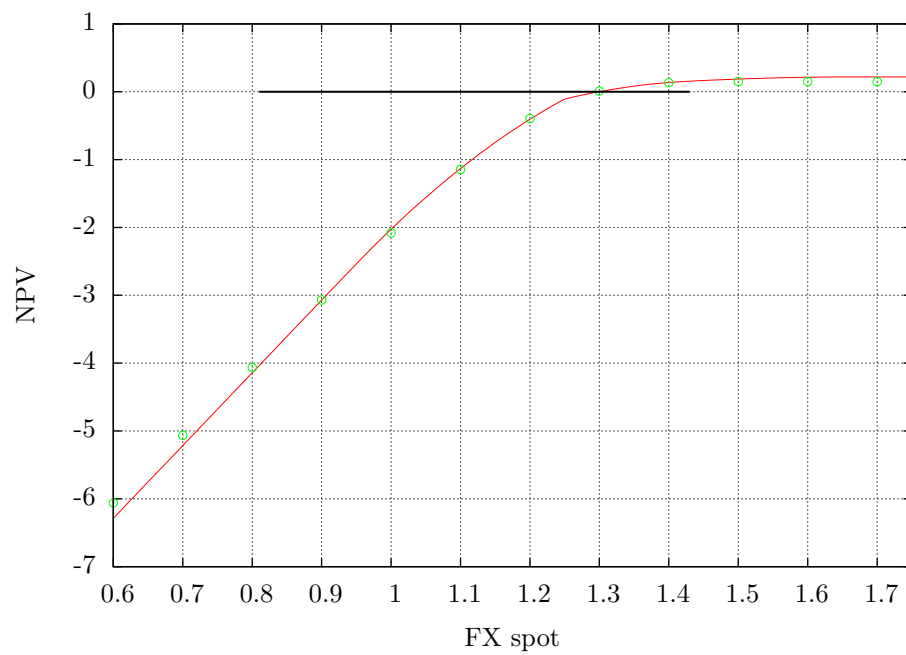
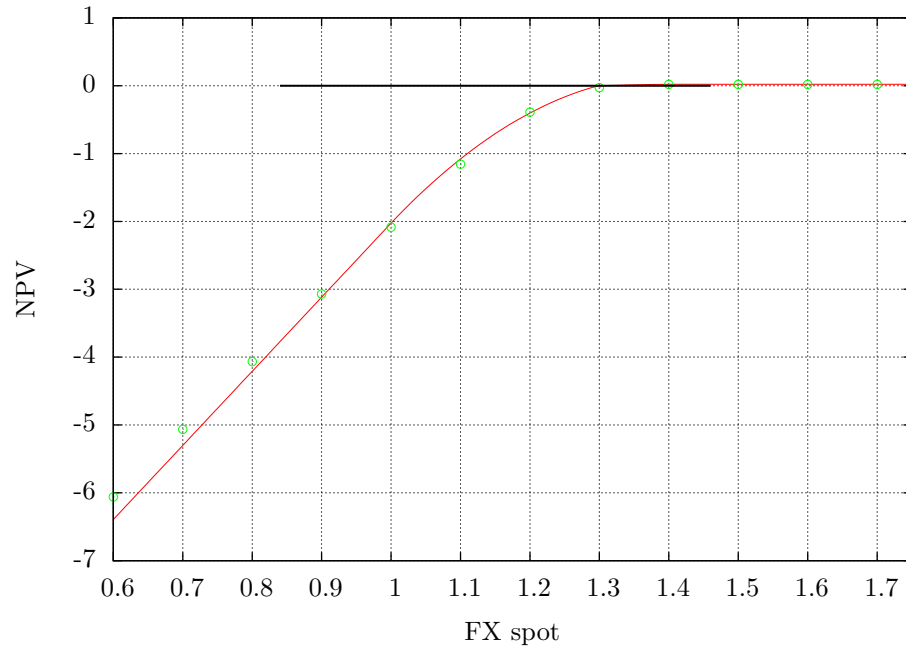


FIGURE 11. Test case #11 (seven fixings, 0.23 out of 0.25 target reached)



E-mail address, Peter Caspers: pcaspers1973@gmail.com

FIGURE 12. Test case #12 (two fixings open, 0.04 out of 0.25 target reached)

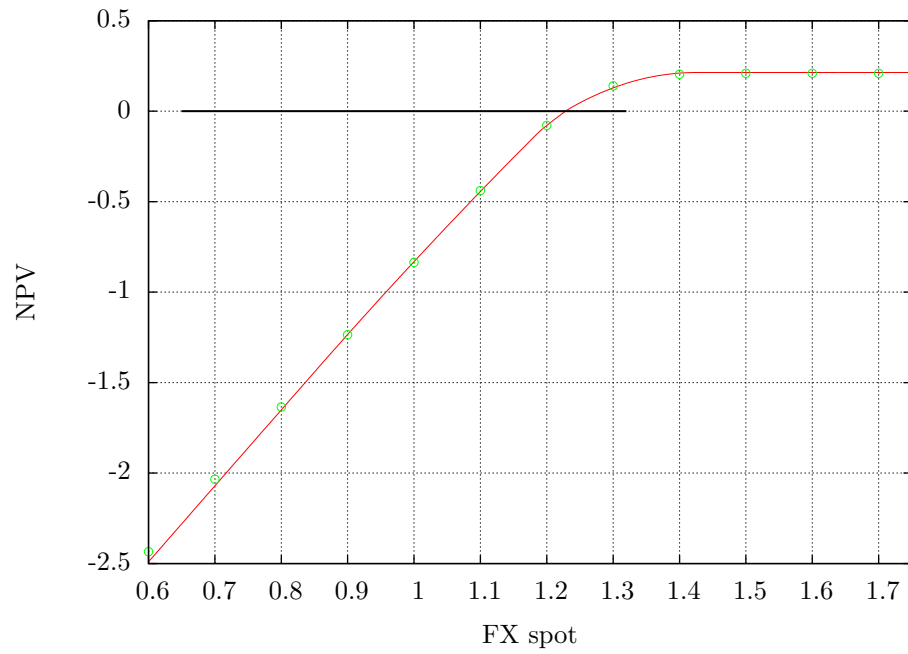


FIGURE 13. Test case #13 (two fixings open, 0.09 out of 0.25 target reached)

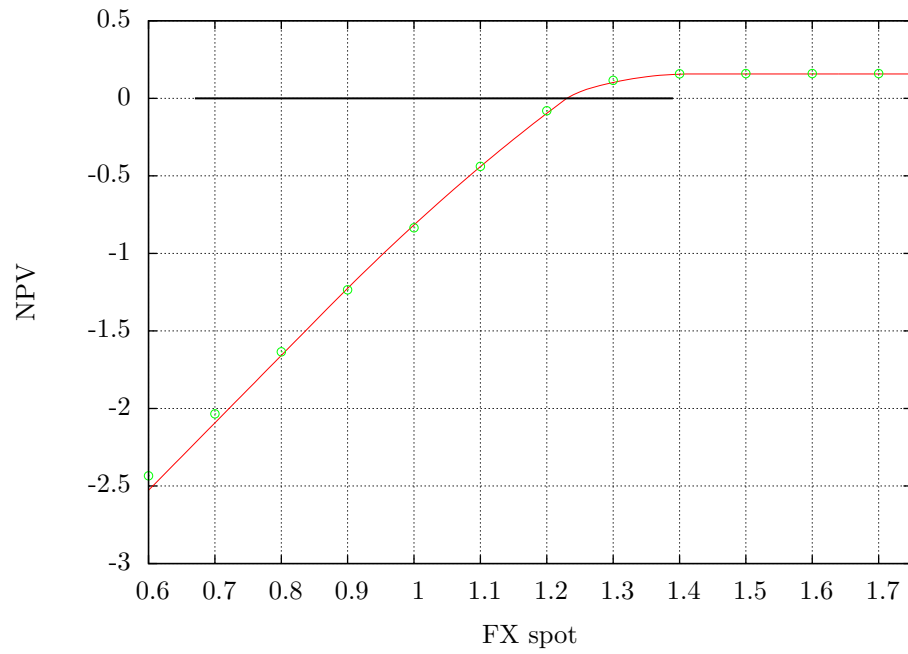


FIGURE 14. Test case #14 (two fixings open, 0.11 out of 0.25 target reached)

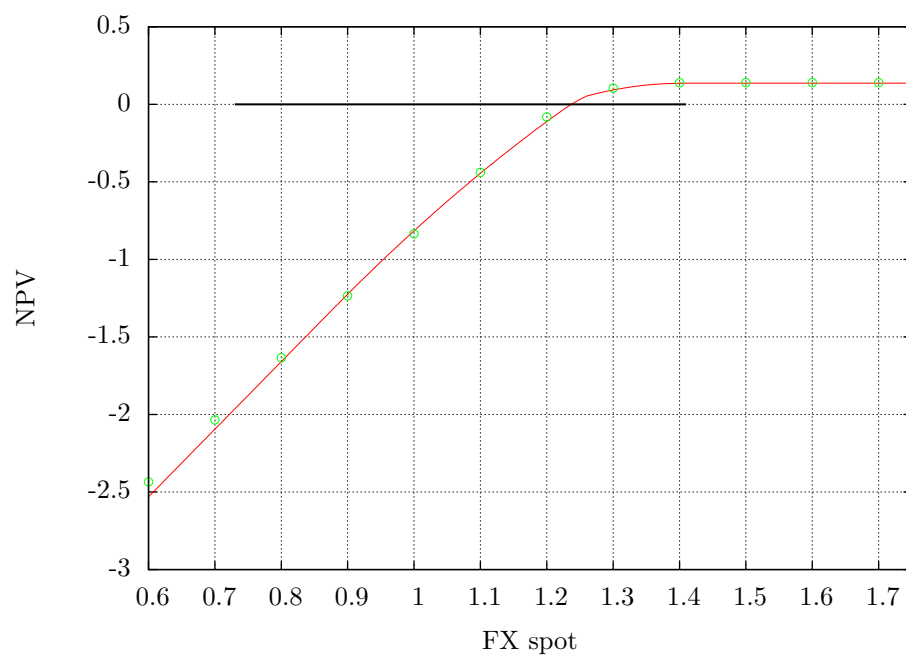


FIGURE 15. Test case #15 (two fixings open, 0.18 out of 0.25 target reached)

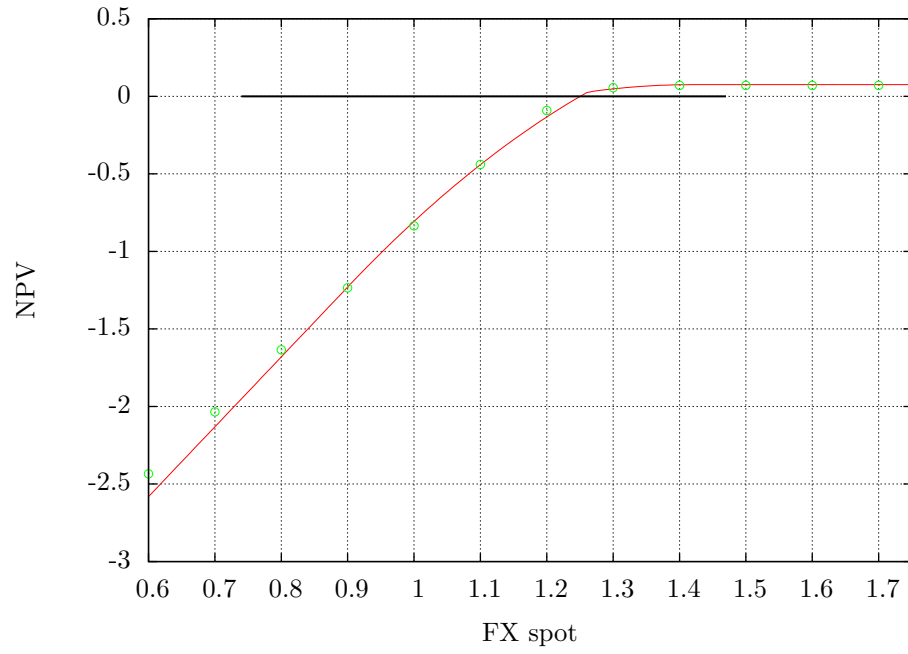


FIGURE 16. Test case #16 (two fixings open, 0.21 out of 0.25 target reached)

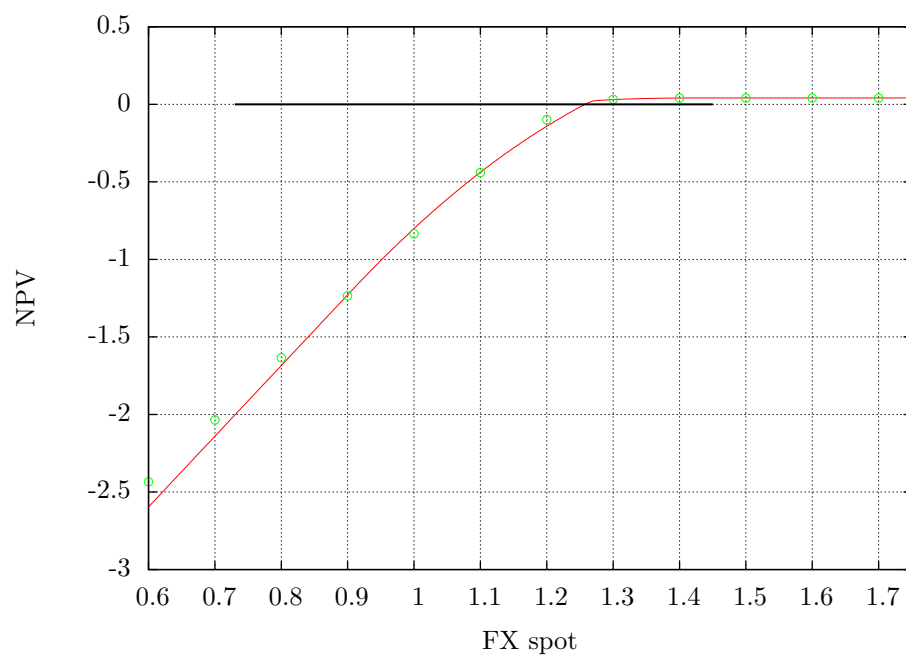


FIGURE 17. Test case #17 (two fixings open, 0.23 out of 0.25 target reached)

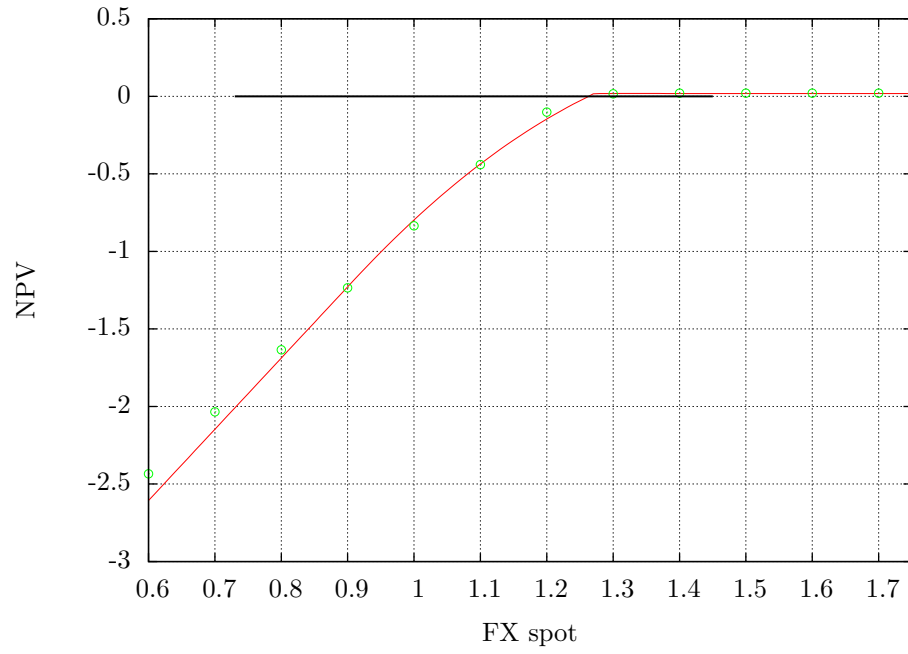


FIGURE 18. Test case #18 (one fixing open, 0.10 out of 0.25 accumulated)

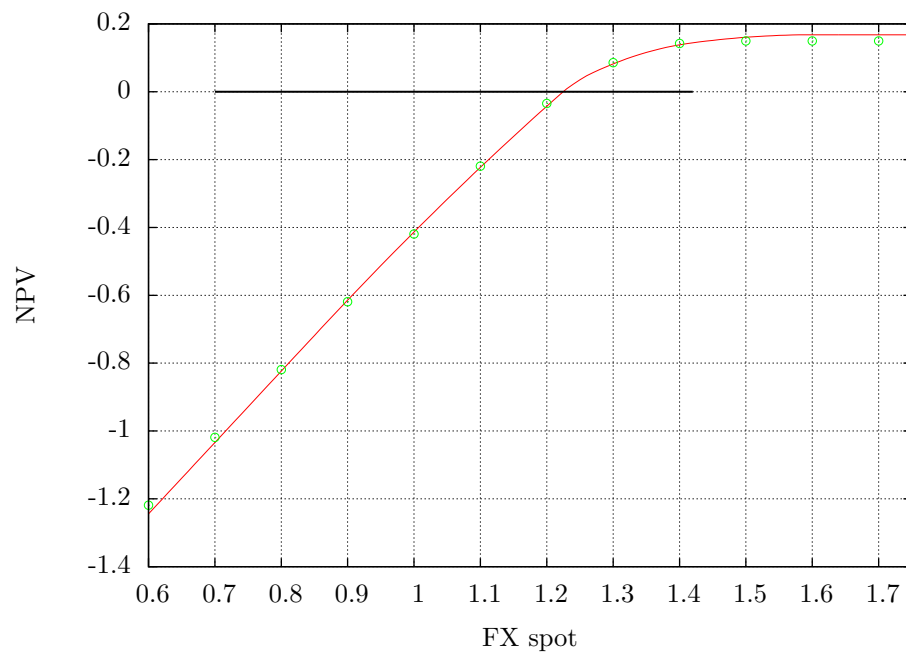


FIGURE 19. Test case #19 (one fixing open, 0.24 out of 0.25 accumulated)

