Composite option valuation with smiles

Peter Jäckel

VTB Capital

Outline

The economic purpose of composite options

- Global Depository Receipts (GDR) are proxy securities that allow investors in one currency market (ZZZ) to participate in shares that are domestic in some other currency (YYY).
- The value of the GDR is solely generated by the value of the underlying share in its domestic currency YYY.
- Hence, the ZZZ-denominated GDR has a value (in principle) given by simply multiplying the share's YYY-denominated value with the corresponding FX rate, i.e.,
$$S_{\text{GDR}} = S \cdot Q^{\text{YYYZZZ}} \tag{2.1}$$
where Q^{YYYZZZ} represents the net present value of one YYY currency unit in terms of ZZZ currency units.

- Unsurprisingly, where GDRs trade on exchanges, there typically also are options on those GDRs.
- These options trade in their own right, but their link to the underlying actual shares is given in terms of the composite option valuation formula
$$v_{\text{composite}} = \mathrm{E}\big[(S(T) \cdot Q(T) - K)_+\big] \tag{2.2}$$
where we have dropped the FX subscript for brevity.
- This option value is in terms of currency ZZZ.

- Many commodities are quoted and sold worldwide in USD but produced in various countries around the world.
- Producers often wish to hedge their revenue streams

  denominated in their own domestic currency.

  This leads to composite put options that pay

$$(K - S(T) \cdot Q(T))_+ \ . \tag{2.3}$$

- On occasions, producers are also prepared to give up the potential upside of their revenues (denominated in their domestic currency) in return for a reduction of the interest on loans granted to them.

  This leads to composite call options:

$$(S(T) \cdot Q(T) - K)_+ \ . \tag{2.4}$$

1 Introduction | Economic purpose of composite options | Commodities

- As a variation of this theme, we also see this with (typically monthly) averaging features such as

$$\left(\sum_{i=1}^{n} S(T_i) \cdot Q(T_i)/n - K \right)_+ \ . \tag{2.5}$$

  Note that for the hedging of this latter case we may have access to options on $S(T)$, and to options on the FX rate, but we have no options on the composite underlying, unlike what we typically have with GDRs!

- Particularly for the (semi-)analytical valuation of composite Asian options, we'd ideally want to have a method for (semi-)analytical valuation of vanilla options on the composite

$$\tilde{S}(T) \ := \ S(T) \cdot Q(T) \ . \tag{2.6}$$

Multiplication becomes subtraction

- We denote the producer's domestic currency as DOM and the commodity quotation currency as FOR.
- We use the *FX net present value ratio* $Q(T)$ as the applied exchange rate in our discussion, but emphasize that the analysis is easily adjusted for the effect of the FX spot days lag[1], as well as small lags between the observation of the commodity fixing and the applicable FX fixing[2].

[1] The observable FX spot quote is in fact, in general, a short dated forward contract quote, and not equal to the actual current NPV of holding one foreign currency unit which we denote as $Q(t)$ at time t.

[2] A positive lag of the FX fixing leads to a small adjustment factor comprised by the ratios of FX forwards. A negative FX lag, i.e., the situation when the FX fixing is taken *before* the commodity fixing, leads to an additional small quanto adjustment.

Notation:-

$v^{\text{DOM}}(t)$: domestic composite option value at time t

$P_T^{\text{DOM}}(t)$: domestic zero coupon bond value for maturity T at time t

$P_T^{\text{FOR}}(t)$: foreign zero coupon bond value for maturity T at time t

$\mathrm{E}_t^{\aleph}[c(T)]$: expectation of $c(T)$ in measure induced by numéraire $\aleph$ as of filtration $\mathcal{F}_t$

$Q^{\text{FORDOM}}(t)$: one FOR currency unit's value in DOM at time t

$Q_T^{\text{FORDOM}}(t)$: par strike for T-forward contract on Q^{FORDOM} at time t

The composite call option value is

$$v^{\text{DOM}}(t) \quad = P_T^{\text{DOM}}(t) \cdot \mathrm{E}_t^{P_T^{\text{DOM}}}\left[(S(T) \cdot Q^{\text{FORDOM}}(T) - K)_+ \right]$$

which, by changing to the foreign T-forward measure, becomes

$$= Q^{\text{FORDOM}}(t) \cdot P_T^{\text{FOR}}(t) \cdot \mathrm{E}_t^{P_T^{\text{FOR}}}\left[\frac{(S(T) \cdot Q^{\text{FORDOM}}(T) - K)_+}{Q^{\text{FORDOM}}(T)} \right]$$

$$= Q^{\text{FORDOM}}(t) \cdot P_T^{\text{FOR}}(t) \cdot \mathrm{E}^{P_T^{\text{FOR}}}\left[\left(S(T) - \frac{K}{Q^{\text{FORDOM}}(T)} \right)_+ \right] . \quad (3.1)$$

This simplifies to the T-forward domestic value

$$\mathrm{E}^{P_T^{\text{DOM}}}\left[(S(T) \cdot Q^{\text{FORDOM}}(T) - K)_+ \right] = \quad (3.2)$$

$$Q_T^{\text{FORDOM}}(t) \cdot \mathrm{E}^{P_T^{\text{FOR}}}\left[\left(S(T) - K \cdot Q^{\text{DOMFOR}}(T) \right)_+ \right] .$$

NOTE:

Both $S(T)$ and Q^{DOMFOR} are martingales in the foreign T-forward measure!

➡ An option on the product of a (quantoed) asset price and a (martingale) FX rate turns into a *zero-strike option on the spread of two martingales*

$$\mathrm{E}\left[\left(S - K \cdot Q^{\text{DOMFOR}} \right)_+ \right] . \quad (3.3)$$

A multiplication becomes a subtraction.

Generic bilinear option valuation

In order to compute the value of the generic bilinear (call[3]) option

$$\mathrm{E}\big[(\alpha \cdot A + \beta \cdot B - \Gamma)_+\big] \tag{4.1}$$

we start with the following exact relationship for[4] $\alpha > 0$ and $\beta > 0$

$$(\alpha \cdot A + \beta \cdot B - \Gamma)_+ = \int_{-\infty}^{\infty} \mathbf{1}_{\{\alpha \cdot A \,\geq\, x \,\geq\, \Gamma - \beta \cdot B\}} \,\mathrm{d}x \;, \tag{4.2}$$

and hence

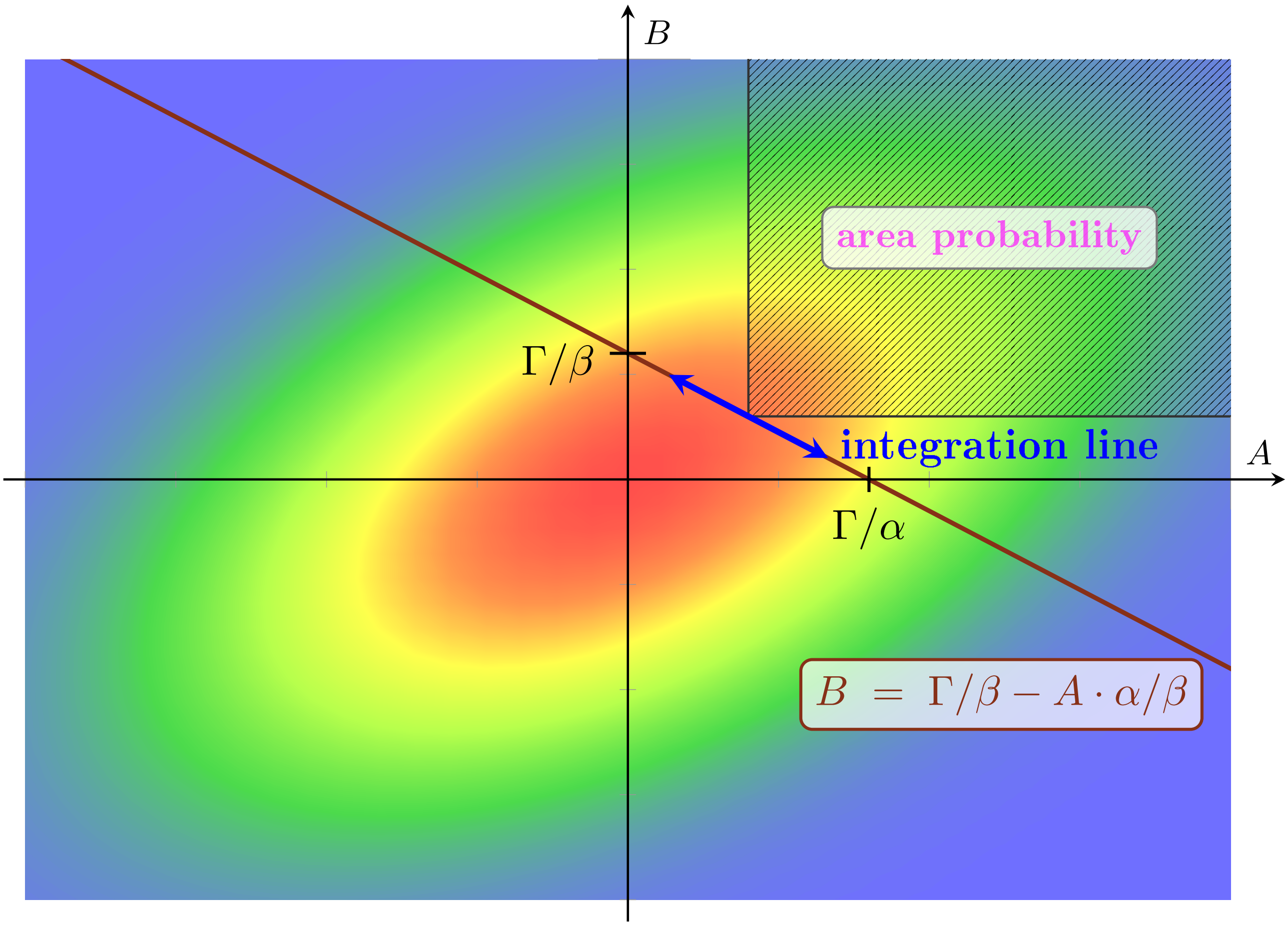
$$\mathrm{E}\big[(\alpha \cdot A + \beta \cdot B - \Gamma)_+\big] = \int_{-\infty}^{\infty} \overbrace{\mathrm{E}\big[\mathbf{1}_{\{A \,\geq\, x/\alpha\}} \cdot \mathbf{1}_{\{B \,\geq\, (\Gamma - x)/\beta\}}\big]}^{\textit{“quadrant-digital”}} \,\mathrm{d}x \;. \tag{4.3}$$

This is *a string of upper-right-quadrant-digitals* along the anti-diagonal-*esque*

$$B \;=\; \tfrac{\Gamma}{\beta} - \tfrac{\alpha}{\beta} \cdot A \;. \tag{4.4}$$

[3]The derivation for put options follows in complete analogy, though note that valuations should never be mapped from out-of-the-money to in-the-money!

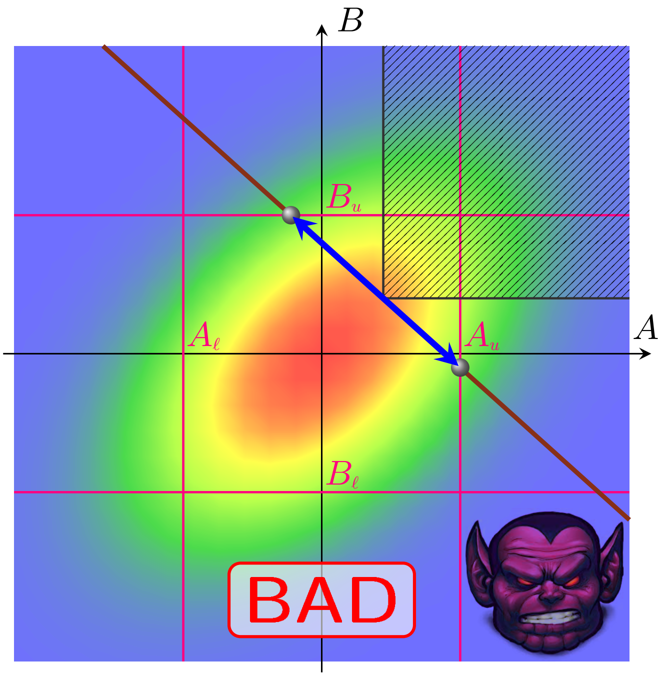
[4]Mutatis mutandis, the logic applies equally to all combinations of signs of α and β.

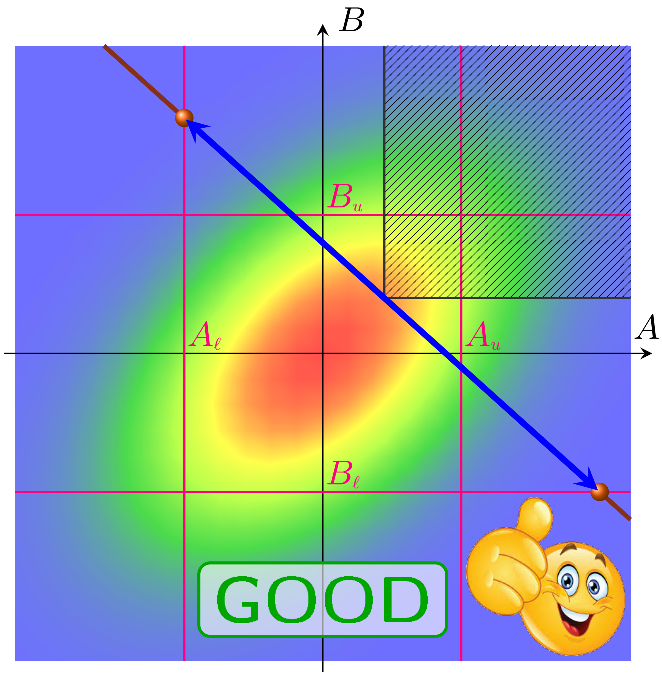


For the integration limits, find the quantiles A_ℓ, A_u, B_ℓ, and B_u such that

$$\mathrm{P}_{\{A<A_\ell\}} = p_{\mathsf{min}}, \quad \mathrm{P}_{\{A>A_u\}} = p_{\mathsf{min}}, \quad \mathrm{P}_{\{B<B_\ell\}} = p_{\mathsf{min}}, \quad \mathrm{P}_{\{B>B_u\}} = p_{\mathsf{min}},$$

where $p_{\mathsf{min}} := \sqrt{\texttt{DBL\_MIN}}$, and integrate to the *outermost intersection points* of the integration line with those univariate quantile levels:





All other cases, i.e., calls vs puts, $\alpha \lessgtr 0$, $\beta \lessgtr 0$, etc.

- Calls and puts (for the same α and β) use *opposite quadrant-digitals*.
- For $\alpha > 0$ and $\beta < 0$, we have

$$\mathrm{E}\big[(\alpha \cdot A - |\beta| \cdot B - \Gamma)_+\big] = \int_{-\infty}^{\infty} \mathrm{E}\big[\mathbf{1}_{\{A \,\geq\, x/\alpha\}} \cdot \mathbf{1}_{\{B \,\leq\, (x-\Gamma)/|\beta|\}}\big]\, \mathrm{d}x \;. \tag{4.5}$$

This is *a string of lower-right-quadrant-digitals* along the diagonal-*esque*

$$B \;=\; -\tfrac{\Gamma}{|\beta|} + \tfrac{\alpha}{|\beta|} \cdot A \;. \tag{4.6}$$

- For all other $\alpha \lessgtr 0$ and $\beta \lessgtr 0$, we use the invariance

$$\mathrm{E}\big[(\alpha \cdot A - \beta \cdot B - \Gamma)_+\big] = \mathrm{E}\big[((-\Gamma) - (-\alpha) \cdot A + (-\beta) \cdot B)_+\big] \;. \tag{4.7}$$

How do we evaluate the quadrant-digitals?

Take the example of the put option with $\alpha > 0$ and $\beta > 0$:

$$\mathrm{E}\big[(\Gamma - \alpha \cdot A - \beta \cdot B)_+\big] = \int_{-\infty}^{\infty} \mathrm{E}\big[\mathbf{1}_{\{A \le x/\alpha\}} \cdot \mathbf{1}_{\{B \le (\Gamma - x)/\beta\}}\big]\, \mathrm{d}x \; . \tag{4.8}$$

We approximate the quadrant-digital by the aid of the Gaussian copula

$$\mathrm{E}\big[\mathbf{1}_{\{A \le a\}} \cdot \mathbf{1}_{\{B \le b\}}\big] \approx G(\mathrm{P}_{\{A \le a\}}, \mathrm{P}_{\{B \le b\}}, \rho_{AB}) \tag{4.9}$$

where

$$\mathrm{P}_{\{X \le K\}} \;=\; \mathrm{E}\big[\mathbf{1}_{\{X \le K\}}\big] \;=\; \Phi(-d_2) + K \cdot \sqrt{T} \cdot \varphi(d_2) \cdot \frac{\mathrm{d}\hat{\sigma}(K)}{\mathrm{d}K} \tag{4.10}$$

with

$$d_2 \;:=\; \frac{\ln(\mathrm{E}[X]/K)}{\hat{\sigma}\sqrt{T}} - \frac{\hat{\sigma}\sqrt{T}}{2} \tag{4.11}$$

and G is the standard Gaussian copula function defined as

$$G(p_a, p_b, \rho) \;=\; \Phi_2\left(\Phi^{-1}(p_a), \Phi^{-1}(p_b), \rho\right) \; . \tag{4.12}$$

In order to avoid *catastrophic subtractive cancellation* (), use for the:-

- lower right quadrant with tail probabilities $\bar{p}_a := \mathrm{P}_{\{A>a\}}$ and $p_b := \mathrm{P}_{\{B<b\}}$,

$$\begin{aligned} \mathrm{P}_{\{A>a \,\wedge\, B<b\}} &= \mathrm{P}_{\{B<b\}} - \mathrm{P}_{\{A<a \,\wedge\, B<b\}} \\ p_b - G(1-\bar{p}_a, p_b, \rho_{AB}) &= \underline{\underline{G(\bar{p}_a, p_b, -\rho_{AB})}} \; , \end{aligned} \tag{4.13}$$

- upper left quadrant with tail probabilities $p_a := \mathrm{P}_{\{A<a\}}$ and $\bar{p}_b := \mathrm{P}_{\{B>b\}}$,

$$\begin{aligned} \mathrm{P}_{\{A<a \,\wedge\, B>b\}} &= \mathrm{P}_{\{A<a\}} - \mathrm{P}_{\{A<a \,\wedge\, B<b\}} \\ p_a - G(p_a, 1-\bar{p}_b, \rho_{AB}) &= \underline{\underline{G(p_a, \bar{p}_b, -\rho_{AB})}} \; , \end{aligned} \tag{4.14}$$

- upper right quadrant

$$\begin{aligned} \mathrm{P}_{\{A>a \,\wedge\, B>b\}} &= \mathrm{P}_{\{A>a\}} + \mathrm{P}_{\{B>b\}} - 1 + \mathrm{P}_{\{A<a \,\wedge\, B<b\}} \\ \bar{p}_a + \bar{p}_b - 1 + G(1-\bar{p}_a, 1-\bar{p}_b, \rho_{AB}) &= \underline{\underline{G(\bar{p}_a, \bar{p}_b, \rho_{AB})}} \end{aligned} \tag{4.15}$$

to evaluate all quadrants via the lower left quadrant of the Gaussian copula ***using only the quadrant-specific univariate tail probabilities.***

Solid bivariate cumulative normals

- Most implementations of analytical formulæ based on the bivariate cumulative normal probability function $\Phi_2(x, y, \rho)$ suffer severely from the use of an unreliable algorithm for Φ_2.

- Personally, I have distrusted any analytics based on Φ_2 for many for exactly that reason: either Φ_2 is not reliable enough to be universally usable, or, depending on the algorithm behind the scenes, so heavy that alternative methods are preferable.

- Graeme West [Wes05] also warns of this problem:

  "*Espen Haug relates a story to me of how his book [Hau97] has received a rather scathing review [...] option prices where the bivariate cumulative is used can be negative, under not absurd inputs!*"

- Graeme West [Wes05] took this as an incentive for a systematic investigation. He discusses various past reviews and algorithms, in particular comparing what he refers to as algorithms DW1 and DW2 from [DW89], and a refinement by Genz in [Gen04] based on DW2. He summarizes:

  "*So, this modified DW2 algorithm might be the algorithm of choice. **It is not as accurate as the Genz algorithm**, but does not have any material inaccuracies, and is certainly a lot more compact.*"

- Other studies [Mey09, Mey13] confirm the superiority of Genz's algorithm.

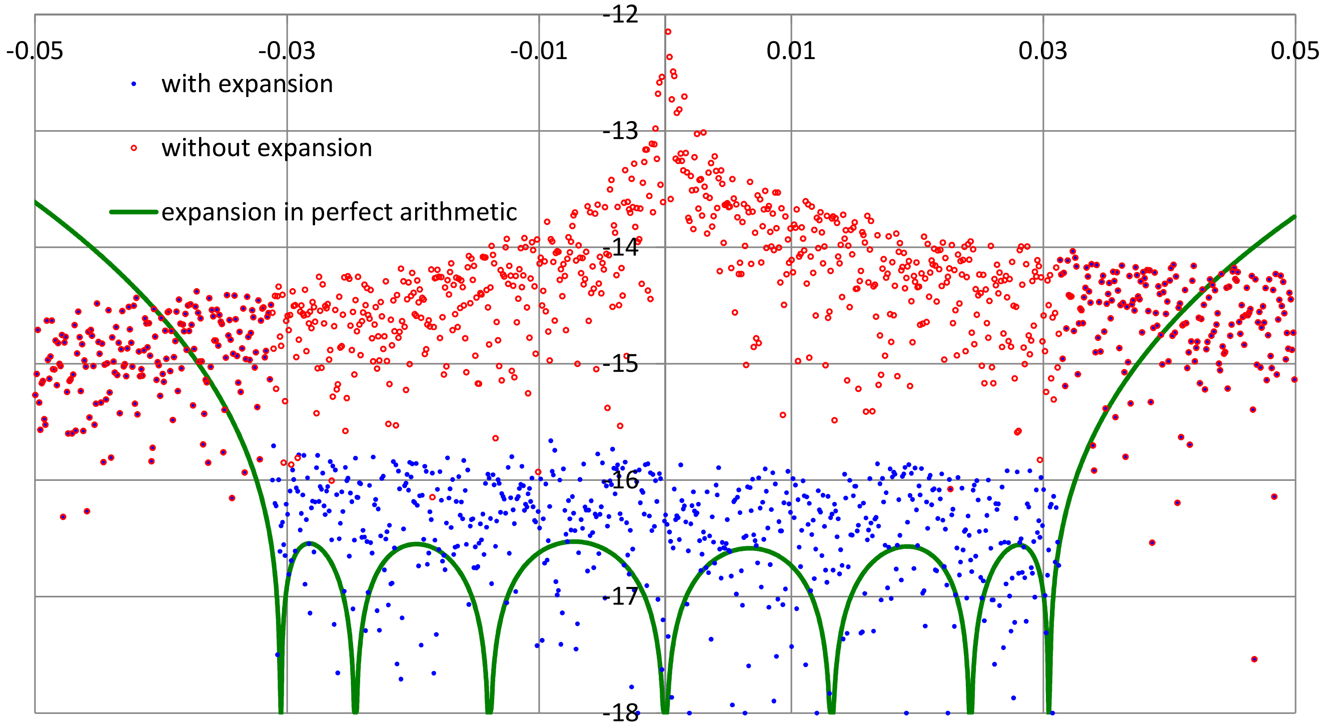
- **I found Genz's algorithm to be fast and compact.**

It's 32 lines of code, and only one of two branches is executed ...

```
double GenzBivariateCdf(double x, double y, double rho) {
  double ar = fabs(rho), H = -x, K = -y, HK = H * K, BVN = 0;
  const int n_GL = ar < 0.3 ? 6 : (ar < 0.75 ? 12 : 20);
  const double *roots = GaussLegendrePoints[n_GL-1], *weights = GaussLegendreWeights[n_GL-1];
  if (ar < 0.925) {
    if (ar > 0) {
      const double HS = 0.5 * (H * H + K * K), ASR = asin(rho);
      for (int i = 0; i < n_GL; ++i) {
        const double SN = sin(ASR * (roots[i] + 1) / 2);
        BVN += weights[i] * exp((SN * HK - HS) / (1 - SN * SN));
      }
      BVN *= 0.5 * ASR * ONE_OVER_TWO_PI;
    }
    BVN += UnivariateCdf(-H) * UnivariateCdf(-K);
  } else {
    if (rho < 0) { K = -K; HK = -HK; }
    if (ar < 1) {
      double AS = (1 - rho) * (1 + rho), A = sqrt(AS), tmp = H - K, BS = tmp * tmp,
             C = (4 - HK) / 8, D = (12 - HK) / 16, ASR = -(BS / AS + HK) / 2;
      if (ASR > -100) BVN = A * exp(ASR) * (1 - C * (BS - AS) * (1 - D * BS / 5) / 3 + C * D * AS * AS / 5);
      if (-HK < 100) {
        const double B = sqrt(BS);
        BVN -= exp(-HK / 2) * SQRT_TWO_PI * UnivariateCdf(-B / A) * B * (1 - C * BS * (1 - D * BS / 5) / 3);
      }
      A *= 0.5;
      for (int i = 0; i < n_GL; ++i) {
        tmp = A * (roots[i] + 1);
        const double x2 = tmp * tmp, RS = sqrt(1 - x2), rs_minus_one /* PJ */ = SqrtOnePlusXMinusOne(-x2);
        ASR = -(BS / x2 + HK) / 2;
        if (ASR > -100)
          BVN += A * weights[i] * exp(ASR) * ( exp(HK*rs_minus_one/(2*(1+RS)))/RS - (1+C*x2*(1+D*x2)) );
      }
      BVN *= -ONE_OVER_TWO_PI;
    }
    if (rho > 0) BVN += UnivariateCdf(-std::max(H, K));
    else BVN = (K > H) ? (UnivariateCdf(K) - UnivariateCdf(H)) - BVN : -BVN;
  }
  return BVN;
}

double SqrtOnePlusXMinusOne(double x) { // sqrt(1+x)-1 expanded for small x. © PJ, 2018.
  if (fabs(x) < 0.03125) // Relative accuracy (in perfect arithmetic) better than 3E-17 on its branch.
    return x * ( 0.5 - x *(0.1250000000000031+x*(0.12502244801304626+0.023447890920414075*x))
                        /(1+x*(1.5001795841045374+x*(0.6251729196189221 5+0.06253033898254554 6*x))) );
  return sqrt(1 + x) - 1;
}
```


Decadic logarithm of |relative accuracy of $\sqrt{1+x}-1$|.



Does it work?

- We compare the resulting composite option prices for BRENT·USDRUB as of 2018-09-06 (with $\rho = -40\%$) with a Monte Carlo simulation, converted to equivalent Black (implied) volatilities.
- We also include the smile-free (At-The-Forward) approximation

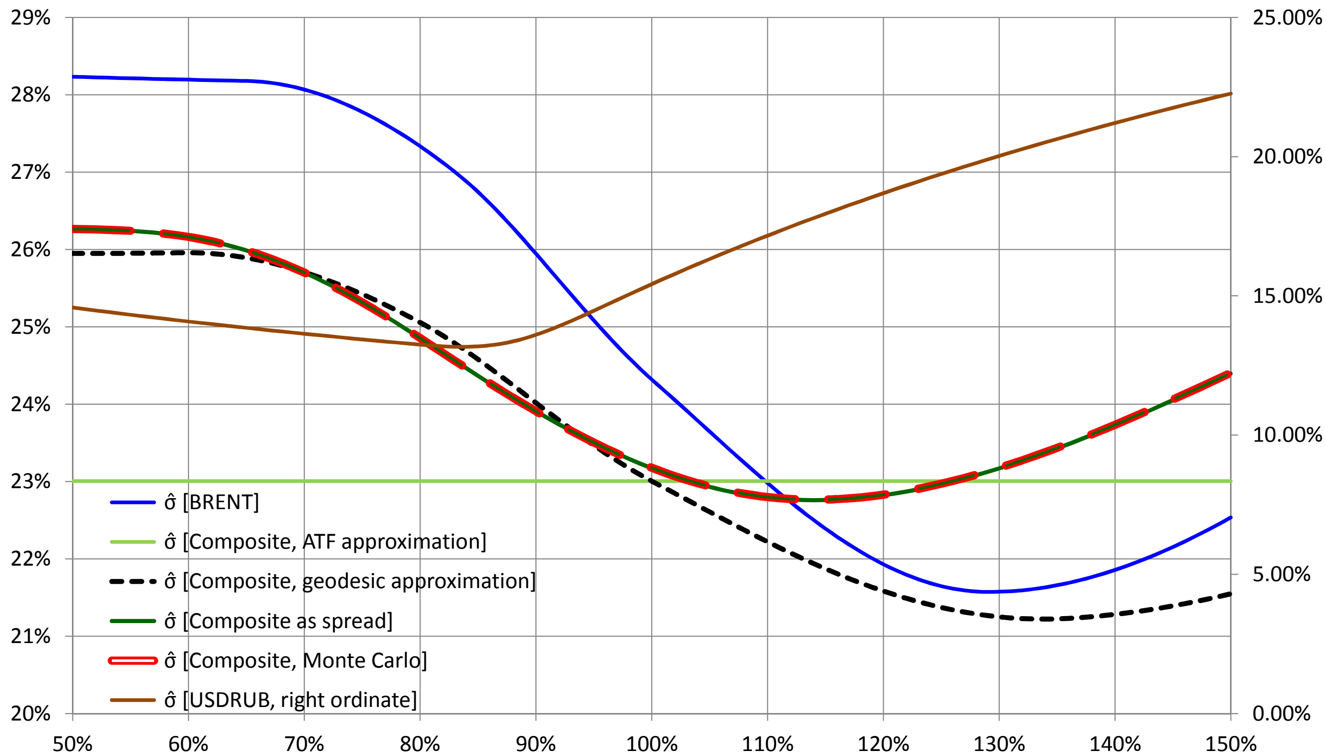
$$\hat{\sigma}_{\text{composite}} = \sqrt{\hat{\sigma}_S^2 + 2 \cdot \rho_{SQ} \cdot \hat{\sigma}_S \cdot \hat{\sigma}_Q + \hat{\sigma}_Q^2} \tag{6.1}$$

- and the *geodesic strikes* approximation [Jäc12] which uses (6.1) but with implied volatilities $\hat{\sigma}_S$ and $\hat{\sigma}_Q$ looked up at the strikes

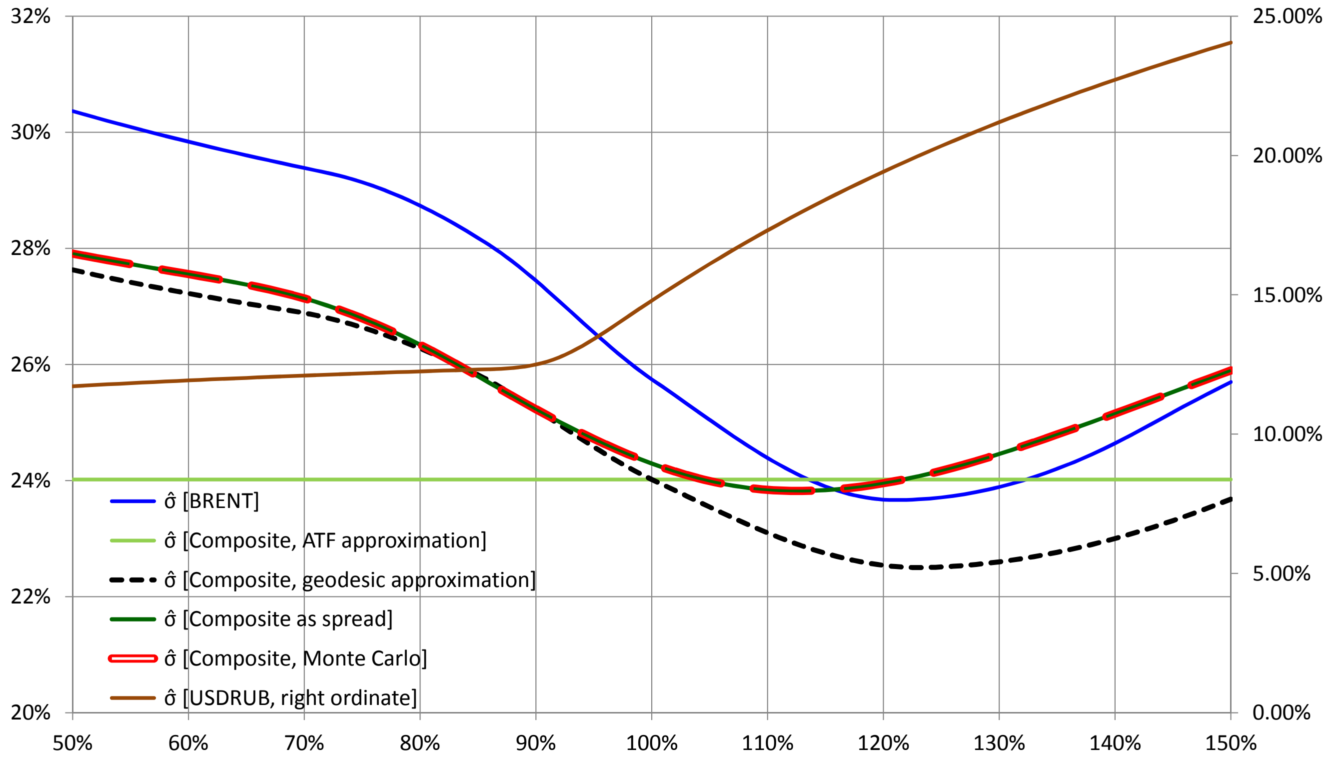
$$\begin{aligned} K_S^* &= \hat{S} \cdot \mathrm{e}^{\left(\ln\left(\frac{K}{\hat{S}\hat{Q}}\right)\cdot\frac{\hat{\sigma}_S\cdot(\hat{\sigma}_S+\rho_{SQ}\hat{\sigma}_Q)}{\hat{\sigma}_S^2+2\hat{\sigma}_S\rho_{SQ}\hat{\sigma}_Q+\hat{\sigma}_Q^2}\right)} \\ K_Q^* &= \hat{Q} \cdot \mathrm{e}^{\left(\ln\left(\frac{K}{\hat{S}\hat{Q}}\right)\cdot\frac{\hat{\sigma}_Q\cdot(\hat{\sigma}_Q+\rho_{SQ}\hat{\sigma}_S)}{\hat{\sigma}_S^2+2\hat{\sigma}_S\rho_{SQ}\hat{\sigma}_Q+\hat{\sigma}_Q^2}\right)} , \end{aligned} \tag{6.2}$$

where $\hat{S} := \mathrm{E}[S]$ and $\hat{Q} := \mathrm{E}[Q]$, in the comparison.

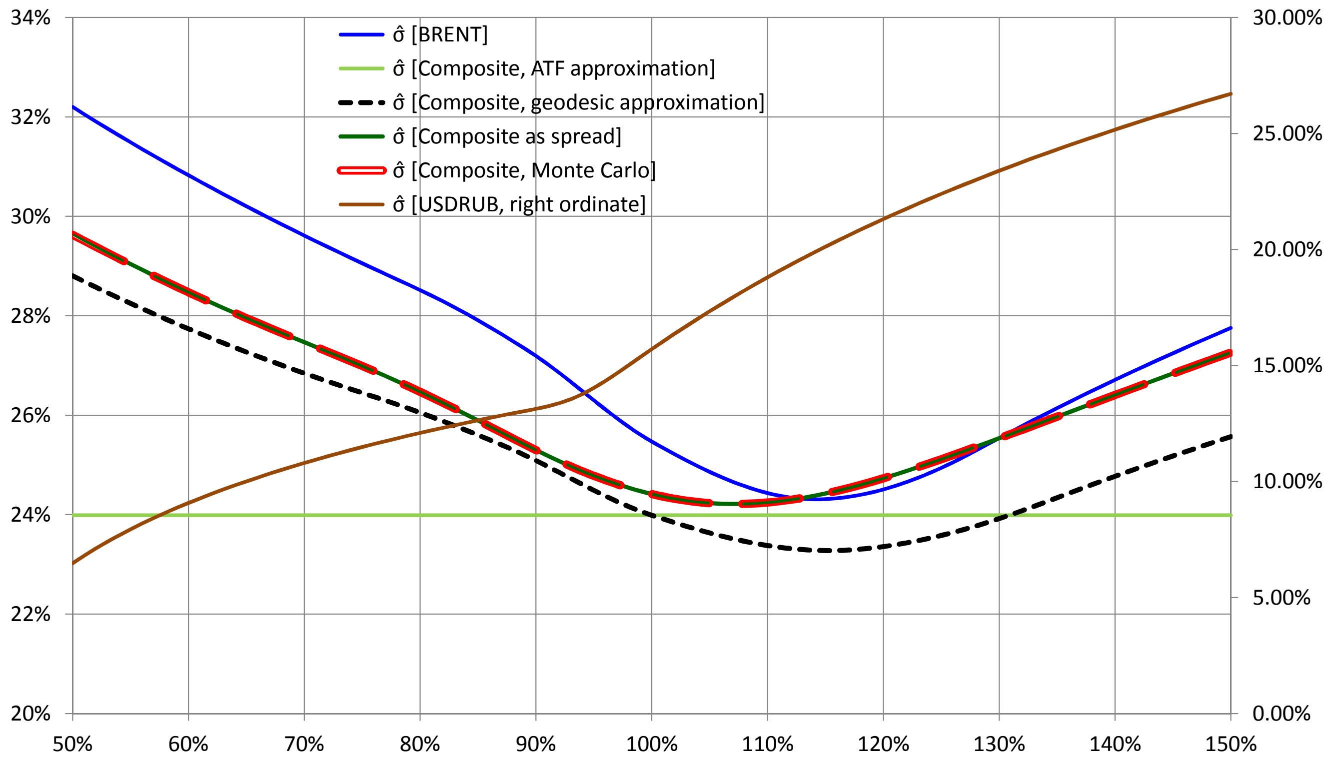
Expiry: 2Y. 1M Monte Carlo iterations.



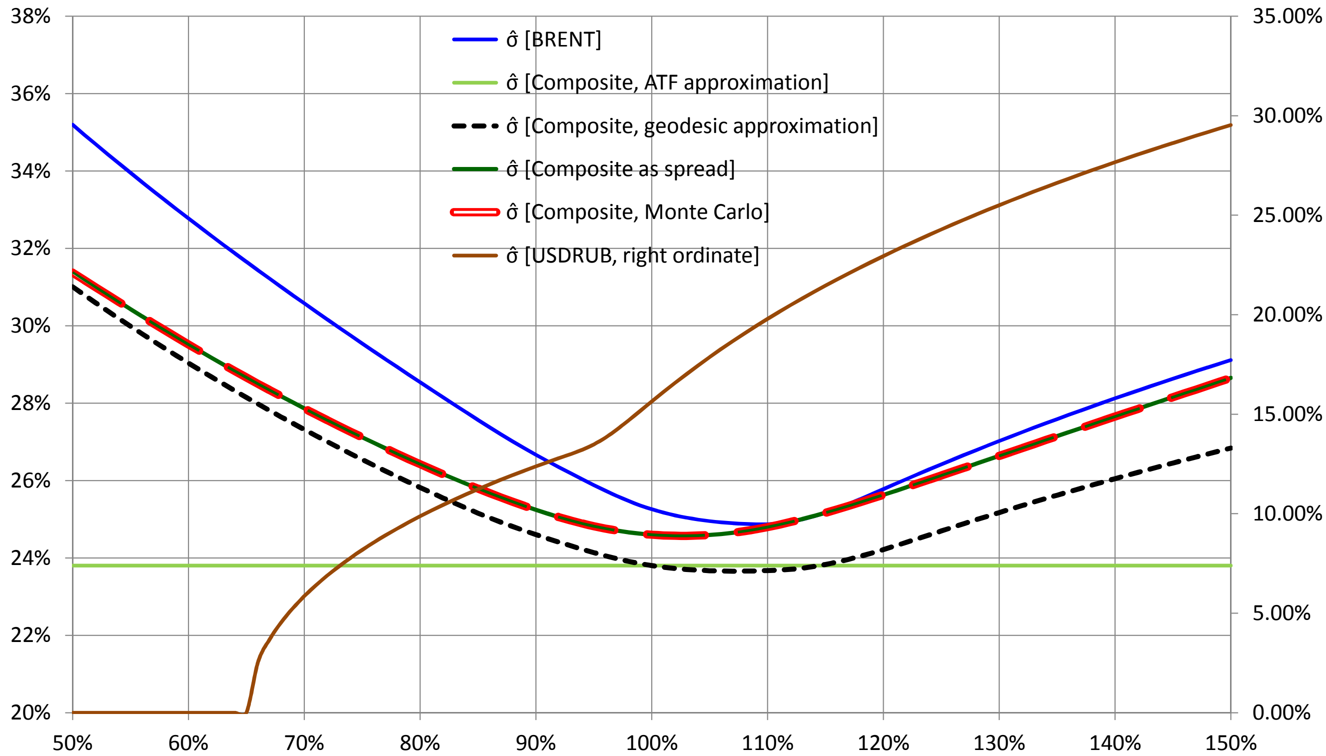
Expiry: 1Y. 1M Monte Carlo iterations.



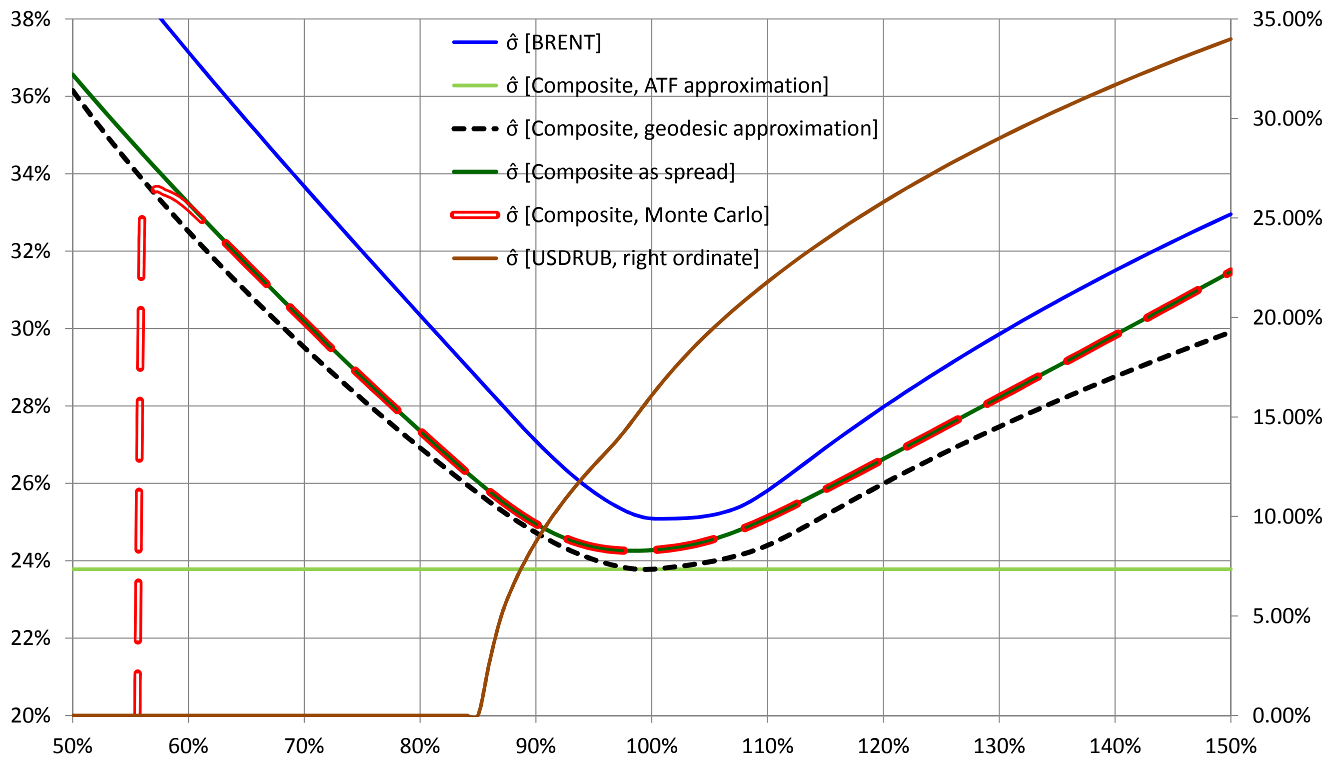
Expiry: 6M. 1M Monte Carlo iterations.



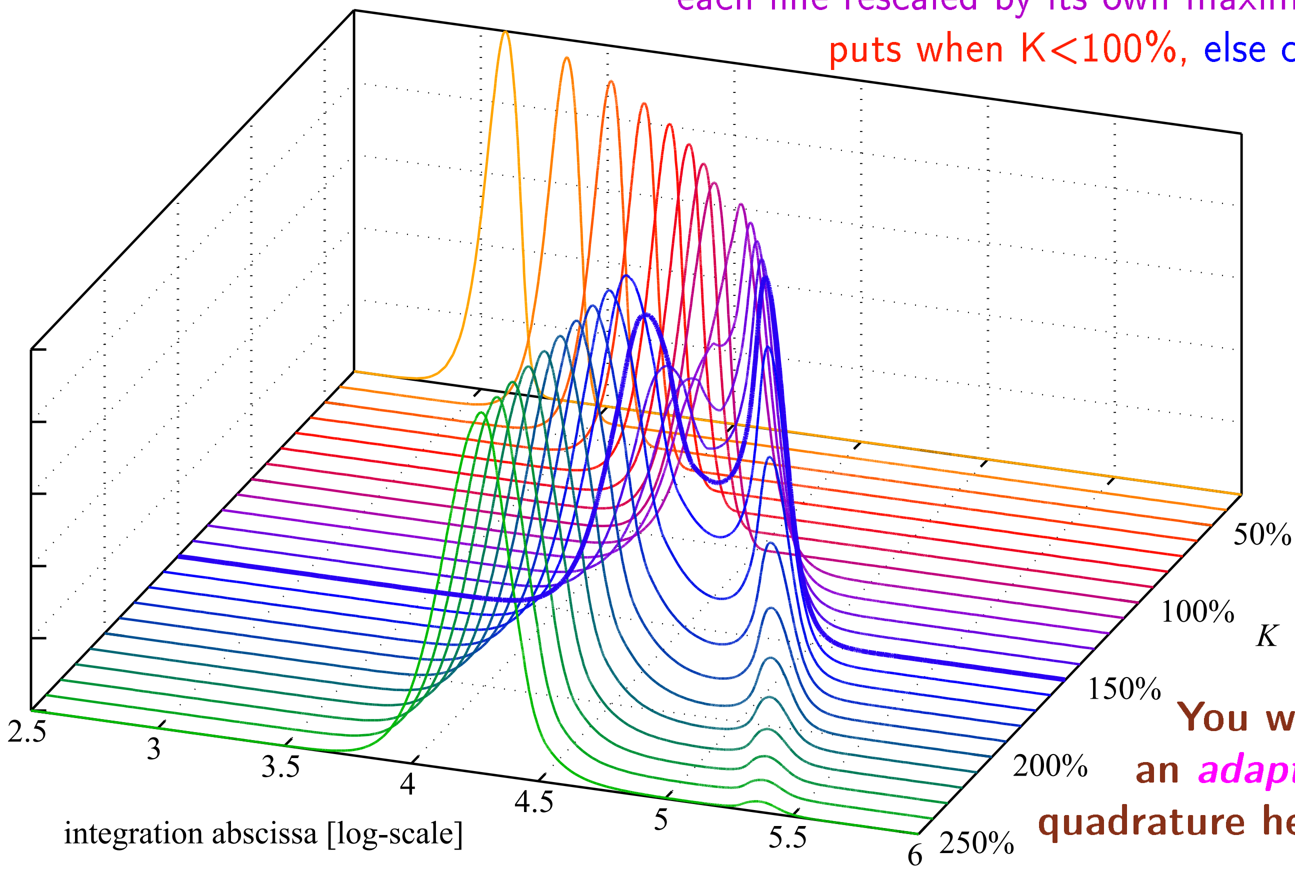
Expiry: 3M. 64M Monte Carlo iterations.



Expiry: 1M. 1G Monte Carlo iterations (but still not enough for low strikes).



The quadrant-digital integrand for BRENT·USDRUB at T=3M across strikes, each line rescaled by its own maximum, puts when K<100%, else calls.



You want an *adaptive* quadrature here!

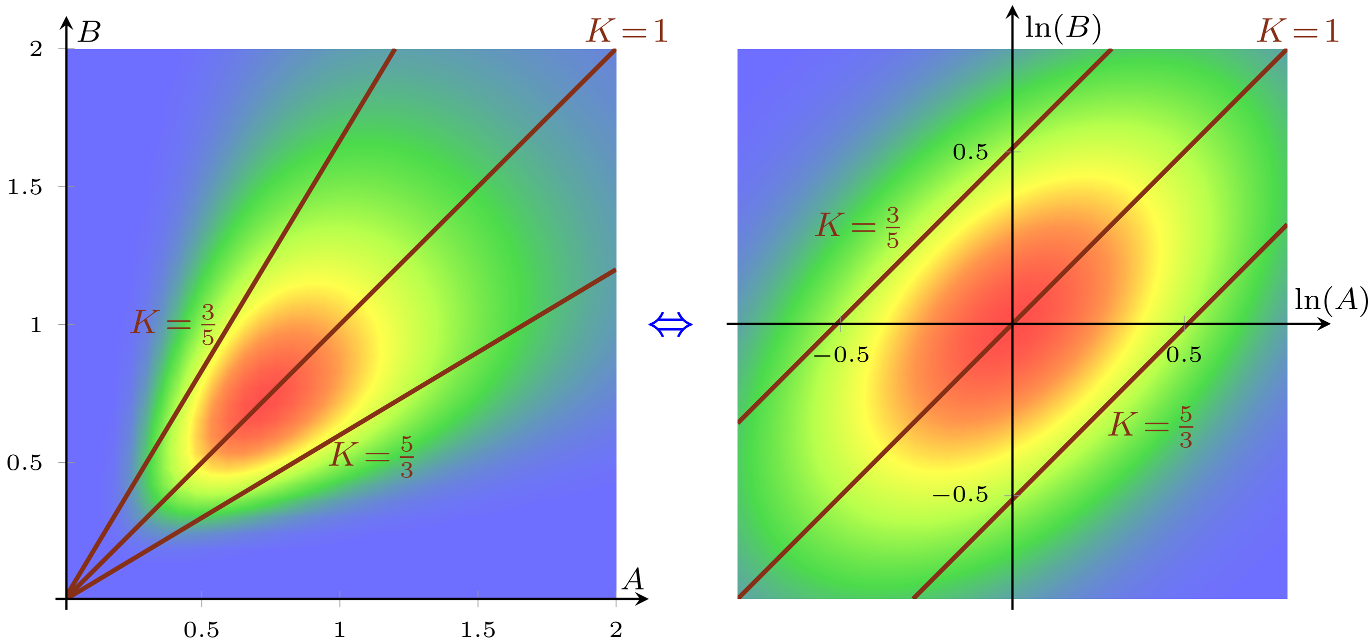
In the following, note that the integration line for $\mathrm{E}\left[(\pm(A - B\cdot K))_+\right]$ given by

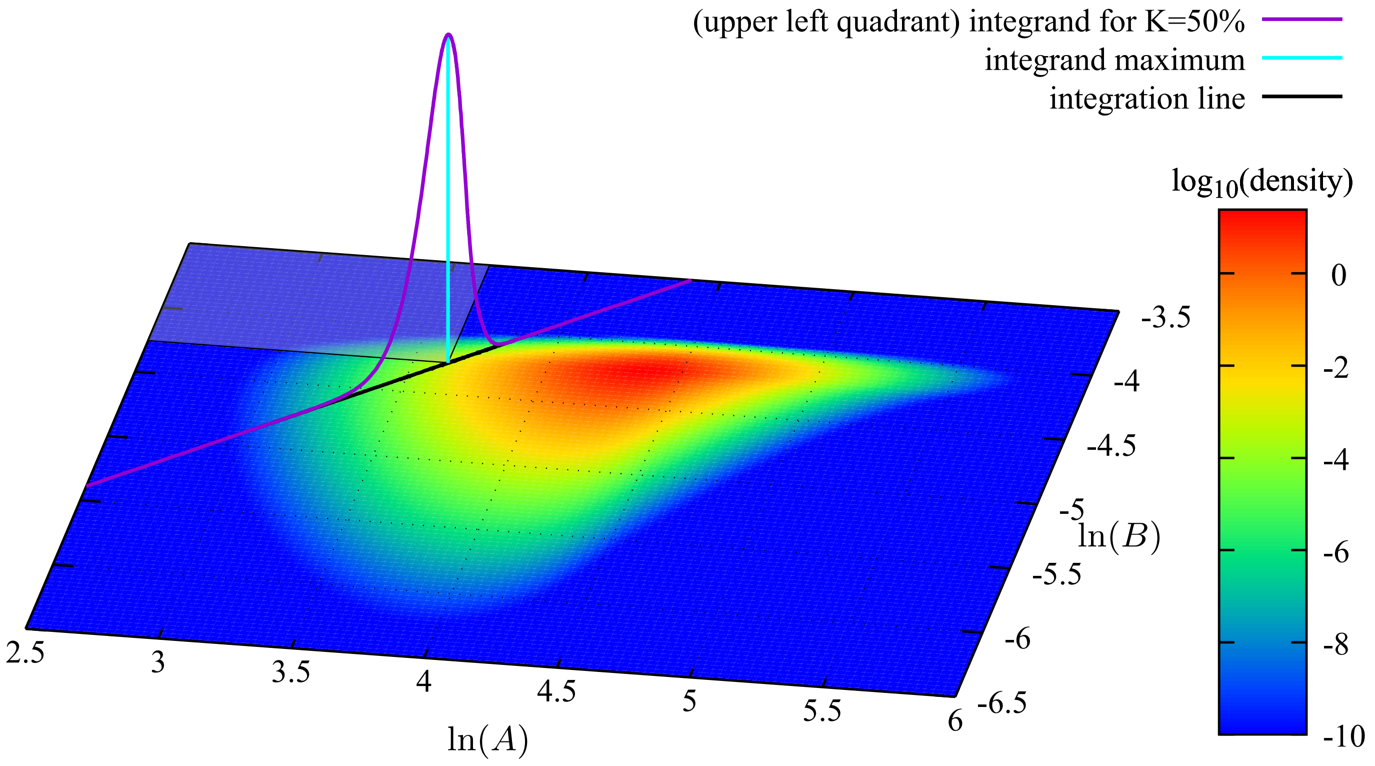
$$B = A/K\,, \tag{6.3}$$

in logarithmic coordinates, becomes

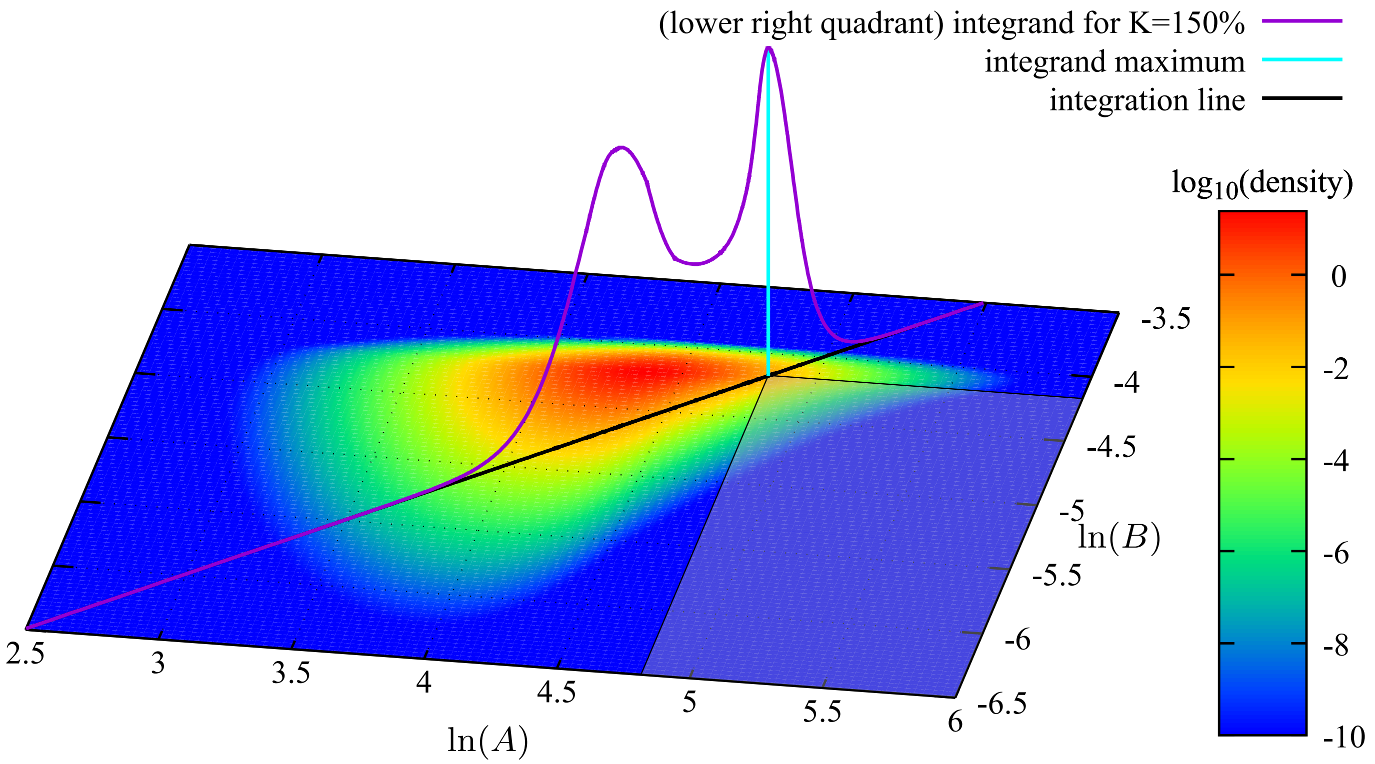
$$\ln(B) = \ln(A) - \ln(K). \tag{6.4}$$

The tilt becomes a shift:

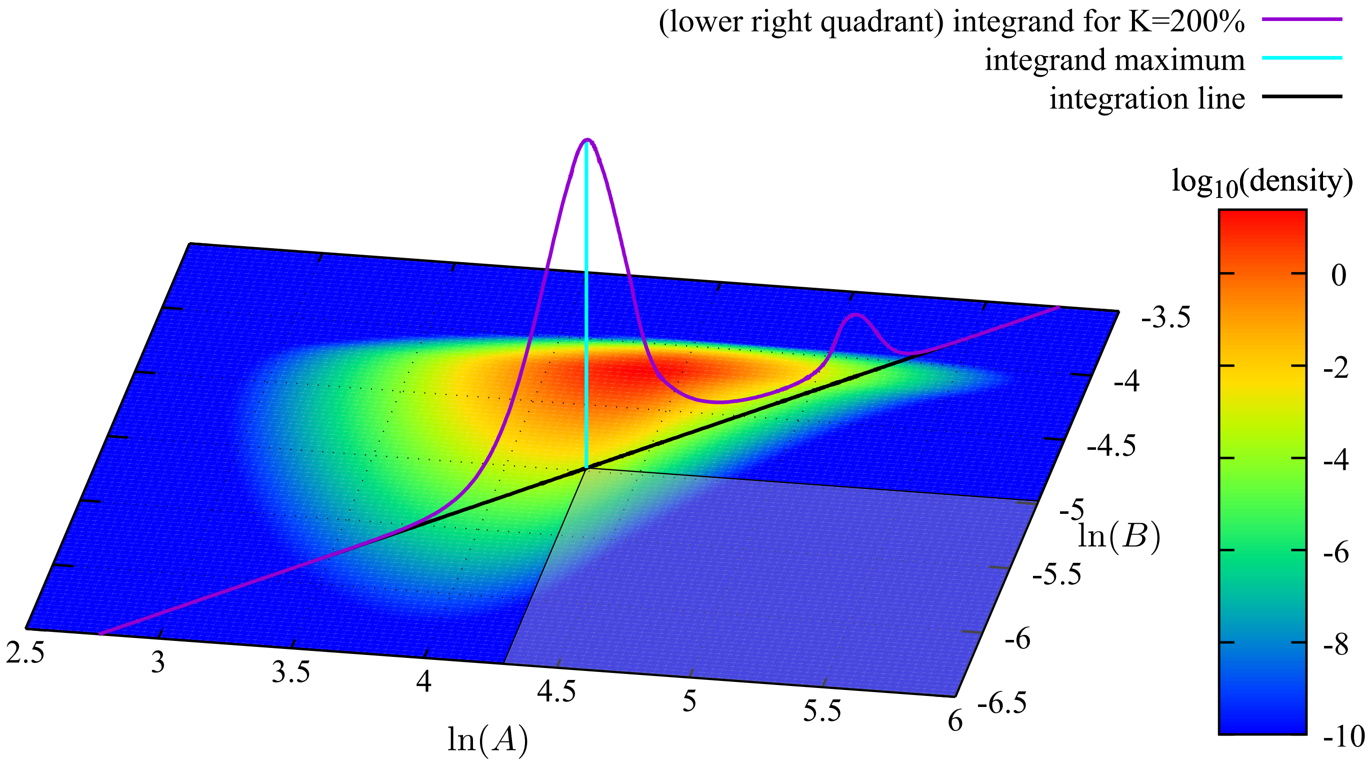




$A =$ Brent, $\qquad B =$ RUBUSD

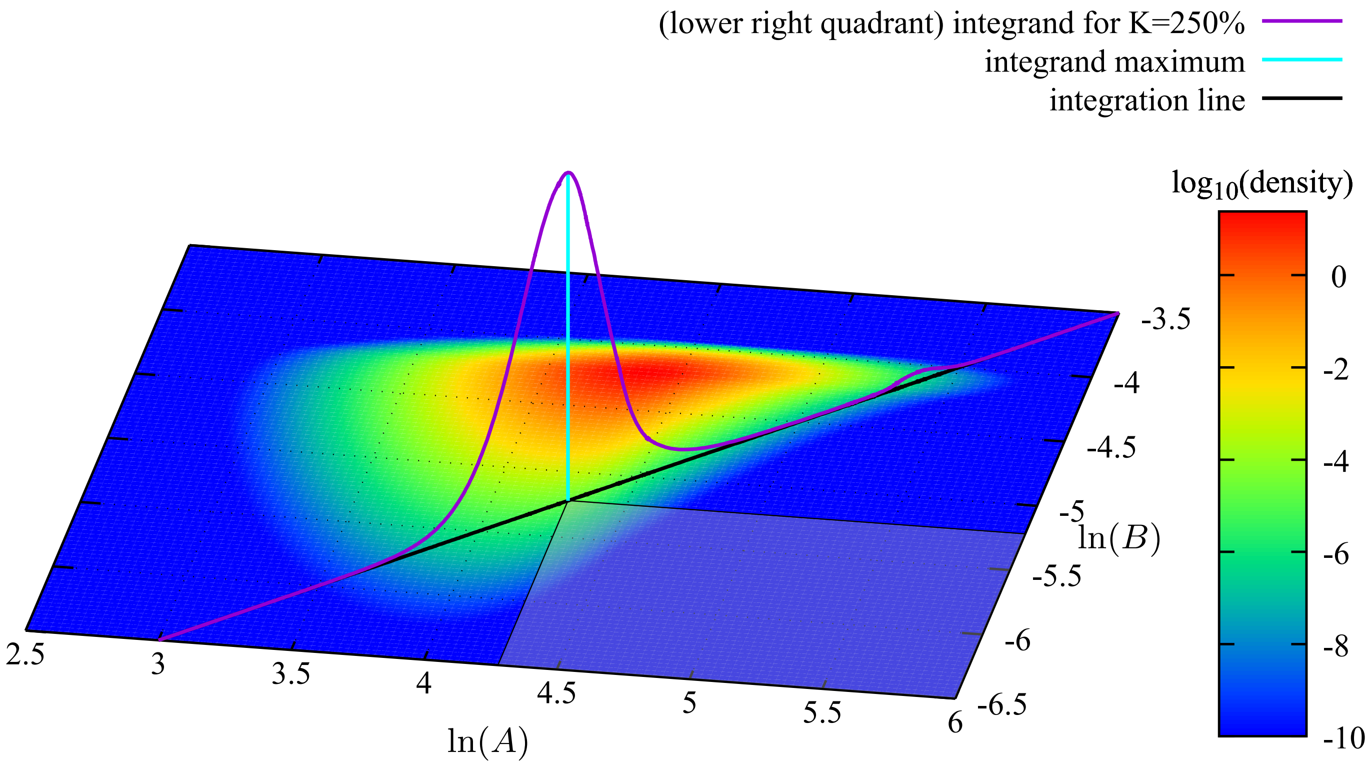


$A =$ Brent, $\qquad B =$ RUBUSD



$A = \text{Brent}, \qquad B = \text{RUBUSD}$

5 Does it work? The quadrant-digital integrand



$A = \text{Brent}, \qquad B = \text{RUBUSD}$

[DW89] Z. Drezner and G. Wesolowsky.
On the computation of the bivariate normal integral.
Journal of Statist. Comput. Simul., 35:101–107, 1989.

[Gen04] A. Genz.
Numerical computation of rectangular bivariate and trivariate normal and t probabilities.
Statistics and Computing, 14:151–160, 2004.

[Hau97] E. G. Haug.
The Complete Guide to Option Pricing Formulas.
McGraw-Hill, October 1997.
ISBN 0786312408.

[Jäc12] P. Jäckel.
Geodesic strikes for composite, basket, Asian, and spread options, July 2012.
www.jaeckel.org/GeodesicStrikesForCompositeBasketAsianAndSpreadOptions.pdf.

[Mey09] C. Meyer.
The Bivariate Normal Copula., 2009.

[Mey13] C. Meyer.
Recursive Numerical Evaluation of the Cumulative Bivariate Normal Distribution.
Journal of Statistical Software, 52, 2013.

[Wes05] G. West.
Better approximations to cumulative normal functions.
Wilmott Magazine, January:30–32, 2005.