Interpolation of Discount Factors

Heinz Cremers
Willi Schwarz

Mai 1996
Abstract

This paper deals with the problem of interpolation of discount factors between time buckets. The problem occurs when price and interest rate data of a market segment are assigned to discrete time buckets. A simple criterion is developed in order to identify arbitrage-free robust interpolation methods. Methods closely examined include linear, exponential and weighted exponential interpolation. Weighted exponential interpolation, a method still preferred by some banks and also offered by commercial software vendors, creates several problems and therefore makes simple exponential interpolation a more logical choice. Linear interpolation provides a good approximation of exponential interpolation for a sufficiently dense time grid.

1 Introduction

Valuation and pricing of financial instruments generally requires knowledge of discount factors and/or zero bond prices. Fundamental to the calculation of discount factors is detailed information on interest rates, as well as on prices of fixed income securities in special market segments (Bond-, FRA-, Swap-market) at present time $t_0$. The procedure for calculating a discount structure $df$ from this information is as follows:

- Starting with market data we define a discrete time structure $t_1, t_2, \ldots, t_N$ and calculate the implied discount factor $df(t_0, t_n)$ for every time to maturity $t_n, n = 1, \ldots, N$ e.g. by using a bootstrapping technique.

- The calculation of the present value of a cash flow $CF(t)$ occurring at time $t$ requires the conversion of the discrete structure $df(t_0, t_1), df(t_0, t_2), \ldots, df(t_0, t_N)$ into a continuous discount curve $t \rightarrow df(t_0, t), t \in [t_0, t_N]$.

The complete set of empirical data is employed in order to derive the discrete discount structure, so that the second step of the problem is reduced to a pure interpolation problem. If the market data is incomplete then an interpolation problem may occur in the first step (e.g. this would be caused by a missing bond).

In this paper we study several widely used interpolation methods thereby confining ourselves to the study of those interpolation problems which require the knowledge of only two adjacent discount factors. McCulloch [1] has developed spline interpolation techniques by using the whole spectrum of market data. Spline interpolation offers a higher degree of smoothness, which has its price in terms of precision or even arbitrage-freeness. For a detailed discussion of this matter we
refer to Breckling, Dal Dosso [2], [3] and Shea [4]. In a forthcoming paper we will investigate interpolation methods using all available market information.

Let us state the problem in more precise terms:

**Problem**

Let \( t_0 \) denote the present time, \( t_1, t_2, \ldots, t_N \) the designated grid structure and let \( df(t_0, t_1), df(t_0, t_2), \ldots, df(t_0, t_N) \) be the discount factors. The problem is the valuation of a given cash flow \( CF(t) = (t, 1) \), which pays an amount 1 at a time \( t \) with \( t_0 \leq t_{n-1} < t < t_n \), considering only the discount factors \( df_{n-1} = df(t_0, t_{n-1}) \) and \( df_n = df(t_0, t_n) \) (without any restriction we assume \( df_{n-1} > df_n \)). Let the index \( n \) be fixed with \( n \in \{1, \ldots, N\} \).

The problem can be looked upon from two different points of view which are somehow “dual” to each other:

- **Interpolation**

  We calculate from discount factors \( df_{n-1} \) and \( df_n \) an interpolated value \( df(t_0, t) = Ip(t, df_{n-1}, df_n) \) and determine the present value (PV = Present Value) of the payment \( (t, 1) \) to be

  \[
  PV(t, 1) = 1 \cdot df(t_0, t) .
  \]  

- **Bucketing**

  Two functions \( B_1 = B_1(t, df_{n-1}, df_n) \) and \( B_2 = B_2(t, df_{n-1}, df_n) \) (bucketing functions) are to be determined in such a way that the cash flow \( (t, 1) \), which pays one unit in \( t \) can be replaced by the cash flows \( (t_{n-1}, B_1) \) and \( (t_n, B_2) \) (Buckets). The present value of the payment \( (t, 1) \) then is calculated as

  \[
  PV(t, 1) = 1 \cdot B_1 \cdot df(t_0, t_{n-1}) + 1 \cdot B_2 \cdot df(t_0, t_n) .
  \]

Tying together the two dual view points, i.e. equating (1) and (2) we obtain

\[
(*) \quad df(t_0, t) = B_1 \cdot df(t_0, t_{n-1}) + B_2 \cdot df(t_0, t_n);
\]

Therefore all bucketing methods can be considered as special interpolation methods. This formula and conditions resulting from bucket hedging will be the key point in our analysis. Bucket hedging has been extensively studied by Turnbull [5].
The paper is organized as follows. First, we set some notation and state a no
arbitrage condition suited for our purpose. In the second part, commonly applied
interpolation techniques such as linear, exponential and weighted exponential
interpolation are investigated in a qualitative manner. Their impact on zero rate
structures as well as on forward rate curves is discussed in connection with some
selected interest rate scenarios. It can be seen that the weighted exponential
interpolation already has remarkable drawbacks. The final section contains the
main results of this paper. A simple condition described by a system of differential
equations is imposed on equation (*) Solutions to this system include the linear
and exponential interpolation method. Interestingly, these two solutions are related
by the fact that linear interpolation is the first order term of the Taylor series
expansion of the exponential interpolation.

2 Notation

A continuous function
df = Ip(t, df_{n-1}, df_n), \quad t \in [t_{n-1}, t_n],
with boundary conditions
(3) \quad Ip(t_{n-1}, df_{n-1}, df_n) = df_{n-1} \quad \text{and} \quad Ip(t_n, df_{n-1}, df_n) = df_n
is called interpolation function. Let \( df_{n-1} > df_n \) for all \( n \in \{1, \ldots, N\} \). An interpolation
function \( Ip \) is called arbitrage-free, if \( Ip \) is strictly decreasing in \( t \), that means
(4) \quad Ip(s_1, df_{n-1}, df_n) > Ip(s_2, df_{n-1}, df_n) \quad \text{for} \quad t_{n-1} \leq s_1 < s_2 \leq t_n .
Furthermore, we assume that the variables \( df_n \) are independent given the above
restriction.

Remark 1

No arbitrage is equivalent to the fact that all forward interest rates \( r(t_0, s_1, s_2) \) with
\( t_{n-1} \leq s_1 < s_2 \leq t_n \) are positive.
Proof: For the forward interest rate \( r(t_0, s_1, s_2) \) one has
\[
\frac{df(t_0, s_2)}{df(t_0, s_1)} < 1 \quad \iff \quad Ip(s_2, df_1, df_2) < Ip(s_1, df_1, df_2).
\]

Since we are only interested in the relative distance of the time parameter \( t \) to the left boundary \( t_{n-1} \), we will use the parameter \( \lambda \) instead of \( t \) where
\[
\lambda = \lambda(t) = \frac{t - t_{n-1}}{t_n - t_{n-1}}.
\]

We denote by \( df_\lambda \) the following expression
\[
df_\lambda = Ip(\lambda, df_{n-1}, df_{n}), \quad \lambda \in [0,1]
\]

Then the above boundary conditions can be restated in terms of the new parameter \( \lambda \) as:
\[
(5) \quad Ip(0, df_{n-1}, df_{n}) = df_{n-1} \quad \text{and} \quad Ip(1, df_{n-1}, df_{n}) = df_{n}.
\]

3 Examples of interpolation functions

In the following section we look at different interpolation functions and discuss their qualitative behaviour. In analyzing the zero rate curve and the forward rate structure the following three zero rate scenarios are considered.

<table>
<thead>
<tr>
<th>Maturity</th>
<th>Scenario 1</th>
<th>Scenario 2</th>
<th>Scenario 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 yr</td>
<td>5,0 %</td>
<td>8,5 %</td>
<td>7,0 %</td>
</tr>
<tr>
<td>2 yrs</td>
<td>6,5 %</td>
<td>7,0 %</td>
<td>7,0 %</td>
</tr>
<tr>
<td>3 yrs</td>
<td>7,5 %</td>
<td>6,0 %</td>
<td>7,0 %</td>
</tr>
<tr>
<td>4 yrs</td>
<td>8,2 %</td>
<td>5,3 %</td>
<td>7,0 %</td>
</tr>
</tbody>
</table>

3.1 Linear Interpolation

Linear interpolation is obtained by assigning the relative distances \( 1 - \lambda \) and \( \lambda \) as weights to the discount factors \( df_{n-1} \) and \( df_{n} \), i.e.:
(6) \[ df_\lambda = Ip_{\text{lin}}(\lambda, df_{n-1}, df_n) = (1 - \lambda)df_{n-1} + \lambda df_n. \]

The boundary conditions (5) are easily verified. The no arbitrage condition follows from
\[ \frac{\partial Ip_{\text{lin}}}{\partial \lambda} = df_n - df_{n-1} < 0. \]

**Discount curve**

The resulting curve \( t \to df_t = df_{\lambda(t)} \) is a continuous piecewise linear function which is in general not differentiable at \((t_1, df_{t_1}), (t_2, df_{t_2}), \ldots, (t_N, df_{t_N})\).

**Zero rate curve**

If \( r_t \) denotes the continuously compounded zero rate of the discount factor \( df_t = df_{\lambda(t)} \), then the interpolated interest rate \( r_t \) is expressed as follows:
\[ r_t = -\frac{\ln(df_{\lambda(t)})}{t - t_0} = \frac{-\ln((1 - \lambda(t))e^{-r_{n-1}(t_{n-1} - t_0)} + \lambda(t)e^{-r_n(t_n - t_0)})}{t - t_0}. \]

For the period [1 yr, 4 yrs] we obtain, using a time interval of length \( \Delta = 0.1 \) yrs, and given scenario 1 the following zero rate curve.

![Graph 1. Zero rate curve with normal term structure (scenario 1)](image)

Similarly we obtain for an inverse term structure (scenario 2) a strictly decreasing zero rate curve with convex parts of the curve. In case of a flat term structure (scenario 3), linear interpolation yields a function which has slightly convex pieces.
**Forward rate curve**

Let \( df(t_0, s_1, s_2) \) denote the forward discount factor and \( r(t_0, s_1, s_2) \) its exponential forward interest rate for the time interval \([s_1, s_2] \subseteq [t_{n-1}, t_n]\). The discount factor, respectively the forward rate, can be expressed by the following formulas

\[
\begin{align*}
    df(t_0, s_1, s_2) &= \frac{df(t_0, s_2)}{df(t_0, s_1)} = \frac{(1 - \lambda(s_2))df_{n-1} + \lambda(s_2)df_n}{(1 - \lambda(s_1))df_{n-1} + \lambda(s_1)df_n} \\
    r(t_0, s_1, s_2) &= -\frac{\ln(df(t_0, s_1, s_2))}{s_2 - s_1}
\end{align*}
\]

For constant time intervals of length \( \Delta = s_2 - s_1 = 0.1 \) yrs and time interval \([1 \text{ yr}, 4 \text{ yrs}]\) the forward curve is as follows:
In both scenarios (normal term structure as well as inverse term structure) one obtains increasing forward rates within the interpolation interval; discontinuities appear at the boundary of the time intervals. The discontinuities are due to the method of interpolation chosen, which calculates discount factors as an average of adjacent discount factors. In a flat term structure scenario (scenario 3), forward rates are not only increasing but also show a periodic behaviour.

3.2 Exponential Interpolation

This form of interpolation is obtained by assigning certain exponents to the discount factors $df_{n-1}, df_n$:

\[
\alpha_n = I p_{\text{exp}}^\alpha (\lambda, df_{n-1}, df_n) = df_{n-1}^{1-\lambda} df_n^\lambda.
\]

The boundary conditions (5) are easily verified, the no arbitrage condition (4) follows from

\[
\frac{\partial p_{\text{exp}}}{\partial \lambda} = -(\ln df_{n-1}) df_{n-1}^{1-\lambda} df_n^\lambda + df_{n-1}^{1-\lambda} (\ln df_n) df_n^\lambda = df_n^\lambda (\ln df_n - \ln df_{n-1}) < 0.
\]
Discount curve

Since
\[
\frac{\partial^2 \ln \left[ \exp \left( \ln (df_n - \ln df_{n-1}) \right) \right]}{\partial \lambda^2} = df_{\lambda(t)} (\ln df_n - \ln df_{n-1})^2 > 0
\]

the exponential interpolation yields strongly convex pieces in the discount curve. The discount curve \( t \to df_t = df_{\lambda(t)} \) is a continuous function, but in general not differentiable at the points \( t_1, t_2, \ldots, t_N \).

Zero rate curve

Let \( r_t \) denote the continuously compounded zero rate of the discount factor \( df_t = df_{\lambda(t)} \). It is computed using the linearly interpolated value of the adjacent zero rates

\[
\exp(-r_t(t-t_0)) = df_{\lambda(t)} = df_{t_n-1}df_t = \exp(- (1 - \lambda) (t_{n-1} - t_0) r_{n-1}) \exp(-\lambda(t_n - t_0) r_n) = \exp\left(- \left( (1 - \lambda) \frac{t_{n-1} - t_0}{t - t_0} r_{n-1} + \lambda \frac{t_n - t_0}{t - t_0} r_n \right) (t - t_0) \right)
\]

and

\[
r_t = (1 - \lambda) \frac{t_{n-1} - t_0}{t - t_0} r_{n-1} + \lambda \frac{t_n - t_0}{t - t_0} r_n.
\]

Given scenario 1, the zero rate curve appears as follows, once again by using time intervals of \( \Delta = 0,1 \) yrs and time periods \([1 \text{ yr}, 4 \text{ yrs}]\):

![Graph 6. Zero rate curve with normal term structure (scenario 1)](image)

Given scenario 2, the zero rate curve decreases yielding convex curve pieces. Given a flat zero rate structure (scenario 3), the exponential interpolation maintains this property, which can be derived as follows: If \( r_{n-1} = r_n \) one obtains
\[ r_t = (1 - \lambda) \frac{t_{n-1} - t_0}{t - t_0} r_{n-1} + \lambda \frac{t_n - t_0}{t - t_0} r_n = \left( (1 - \lambda) \frac{t_{n-1} - t_0}{t - t_0} + \lambda \frac{t_n - t_0}{t - t_0} \right) r_n \]
\[ = \frac{(t_n - t)(t_{n-1} - t_0) + (t - t_{n-1})(t_n - t_0)}{(t_n - t_{n-1})(t - t_0)} r_n = r_n \quad \text{for all } t. \]

Graph 7. Zero rate curve with flat term structure (scenario 3)

**Forward rate curve**

Exponential interpolation implies constant forward rates \( r(t_0, s_1, s_2) \) for time intervals \([s_1, s_2]\) of equal length. Let \( s_1 \) and \( s_2 \) be such that \( t_{n-1} \leq s_1 < s_2 \leq t_n \). Then given \( \lambda_1 = \lambda(s_1) \), \( \lambda_2 = \lambda(s_2) \) and a forward discount factor \( df(t_0, s_1, s_2) \) it can be rewritten as

\[
df(t_0, s_1, s_2) = \frac{df(t_0, s_2)}{df(t_0, s_1)} = \frac{df^{1-\lambda_2} df^{\lambda_2}}{df^{1-\lambda_1} df^{\lambda_1}} = \frac{df^{\lambda_2-\lambda_1}}{df^{\lambda_2-\lambda_1}} = \left( \frac{df}{df_{n-1}} \right)^{\frac{s_2-s_1}{t_n-t_{n-1}}},
\]

i.e. \( df(t_0, s_1, s_2) \) and \( r(t_0, s_1, s_2) \) as well only depend on the distance \( s_2 - s_1 \). For time intervals with a length of \( \Delta = s_2 - s_1 = 0.1 \) yrs and time periods \([1 \text{ yr}, 4 \text{ yrs}]\) we obtain the following forward rate curve
3.3 Weighted Exponential Interpolation

This interpolation method is obtained by assigning additional time weights to the exponents in (7):

\[ df_i = \text{lp}^{\text{weight exp}} (t, df_{n-1}, df_n) = df_{n-1}^{\alpha_i(t)(1-\lambda(t))} \cdot df_n^{\alpha_i(t)\lambda(t)} \]
where: \( \lambda = \lambda(t) = \frac{t-t_{n-1}}{t_n-t_{n-1}} \) and \( \alpha_i = \alpha_i(t) = \frac{t-t_0}{t_i-t_0} \).

\( \lambda \) satisfies the boundary conditions (3), however the no arbitrage condition (4) does not hold.

**Counterexample**

Let \( df_1 = 0.91, \ df_2 = 0.89, \ t_0 = 0, \ t_1 = 1, \ t_2 = 2 \) and \( t = 1.8 \) then

\[ df(1.8) = 0.88882386 < 0.89 = df_2. \]

According to Remark 1 in Section I, negative or zero forward rates cannot be excluded by interpolation method (8).

**Remark 2**

In order to obtain a valid expression for the divisor \( t_{n-1} - t_0 = t_0 - t_0 = 0 \) in formula (8) for the first time interval \( [t_0, t_1] \) where \( n = 1 \), we set: \( t_1 = t_0 + 1 \) day and \( r_1 = \) overnight-rate.

**Discount curve**

The discount curve \( t \to df_t \) is a continuous function, but not necessarily differentiable at points \( t_1, t_2, ..., t_N \).

**Term structure**

Let \( r_t \) be the exponential interest rate with discount factor \( df_t = df_{\lambda(t)} \), then the interpolated rate \( r_t \) is given by

\[ r_t = (1-\lambda)r_{n-1} + \lambda r_n, \]

i.e. \( r_t \) is obtained by interpolating adjacent rates in a linear fashion. The term structure as defined by the previous scenarios yields the following shape:
Similar graphs are obtained for inverse (scenario 2) and flat (scenario 3) term structures using piecewise linear functions.

**Forward curve**

For the forward discount factor \( df(t_0, s_1, s_2) \) and its associated weighted exponential forward rate \( r(t_0, s_1, s_2) \) for the time period \([s_1, s_2] \in [t_{n-1}, t_n]\) we have

\[
\begin{align*}
\frac{df(t_0, s)}{df(t_0, s_1)} &= \frac{df(t_0, s_2)}{df(t_0, s_1)} \\
&= \frac{\frac{(s_2-t_n)(t_n-t_0)(t_n-t_0)}{1+t_n-t_0}}{\frac{(s_1-t_n)(t_n-t_0)(t_n-t_0)}{1+t_n-t_0}} \\
&\times \frac{\frac{(s_2-t_0)(t_0-t_n)(t_0-t_n)}{1+t_0-t_n}}{\frac{(s_1-t_0)(t_0-t_n)(t_0-t_n)}{1+t_0-t_n}} \\
r(t_0, s_1, s_2) &= -\frac{\ln(df(t_0, s_1, s_2))}{s_2 - s_1}.
\end{align*}
\]

For time intervals of equal length \( \Delta = s_2 - s_1 = 0.1 \) yrs and time periods \([1 \text{ yr}, 4 \text{ yrs}]\) we obtain the following forward rates, given the aforementioned scenarios:

![Graph 12. Forward interest rate curve with normal term structure (scenario 1)](image)

![Graph 13. Forward rate curve with inverse term structure (scenario 2)](image)
4 Results

A large class of interpolation methods is obtained by using so-called bucketing procedures. As mentioned in the introduction, «buckets» for a cash flow \((t, f)\) where \(t_0 \leq t_{n-1} < t < t_n\) are confined to the time period \(t_{n-1}\) and \(t_n\). Two continuous functions

\[
B_1 = B_1(t, df_{n-1}, df_n) \quad \text{and} \quad B_2 = B_2(t, df_{n-1}, df_n), \quad t \in [t_{n-1}, t_n]
\]

with \(0 \leq B_1 \leq 1\) and \(0 \leq B_2 \leq 1\) satisfying the boundary conditions

(9) \(t = t_{n-1}: \quad B_1(t_{n-1}, df_{n-1}, df_n) = 1\) and \(B_2(t_{n-1}, df_{n-1}, df_n) = 0\)

\(t = t_n: \quad B_1(t_n, df_{n-1}, df_n) = 0\) and \(B_2(t_n, df_{n-1}, df_n) = 1\)

are called bucketing functions or a bucketing procedure. As mentioned initially, every bucketing procedure defines an interpolation method. If \(B_1\) and \(B_2\) are bucketing functions, then

(10) \(I_p(t, df_{n-1}, df_n) = B_1(t, df_{n-1}, df_n) \cdot df_{n-1} + B_2(t, df_{n-1}, df_n) \cdot df_n\)

is the associated interpolation function. Given (9), the boundary conditions (3) are satisfied. A bucketing procedure \(B_1, B_2\) is called arbitrage-free, if the associated interpolation function \(I_p\) is arbitrage-free, i.e. if \(I_p\) is strictly decreasing in \(t\). A sufficient condition is

\[
\frac{\partial B_1(t, df_{n-1}, df_n)}{\partial t} df_{n-1} + \frac{\partial B_2(t, df_{n-1}, df_n)}{\partial t} df_n < 0
\]

provided \(B_1\) and \(B_2\) are differentiable in \(t\). Boundary conditions and the no arbitrage property of bucketing procedures have analogue concepts for the associated interpolation function. However, the concept of robustness which is discussed below, seems to have no apparent similarities to interpolation. Robustness is the essential ingredient in deriving “reasonable”
interpolation/bucketing procedures. Further, we assume that the function \( Ip \) is continuously differentiable in the variables \( df_{n-1} \) and \( df_n \).

A bucketing procedure is called \textit{robust}, if \( B_1, B_2 \), and its associated interpolation function satisfy the following system of partial differential equations

\[
\begin{align*}
(**): \quad & \frac{\partial Ip(t,df_{n-1},df_n)}{\partial df_{n-1}} = B_1(t,df_{n-1},df_n) \\
& \frac{\partial Ip(t,df_{n-1},df_n)}{\partial df_n} = B_2(t,df_{n-1},df_n) \quad \text{for all } t \in [t_{n-1},t_n].
\end{align*}
\]

\textbf{Interpretation}

The Taylor series of the associated interpolation function satisfying (**) is given for fixed \( t \in [t_{n-1},t_n] \) and \((df_{n-1}^0,df_n^0)\) by

\[
Ip(t,df_{n-1},df_n) = Ip(t,df_{n-1}^0,df_n^0) + \frac{\partial Ip}{\partial df_{n-1}}(t,df_{n-1}^0,df_n^0) \cdot (df_{n-1} - df_{n-1}^0) \\
+ \frac{\partial Ip}{\partial df_n}(t,df_{n-1}^0,df_n^0) \cdot (df_n - df_n^0) + R_1
\]

Consequently, small changes in discount factors \( df_{n-1} \) and \( df_n \) (\( \Rightarrow \) a small error term \( R_1 \)) will result in invariant bucketing functions \( B_1(t,df_{n-1},df_n) \) and \( B_2(t,df_{n-1},df_n) \). Therefore, a hedge based on bucketing does not have to be adjusted for small changes in market factors.

The main conclusion of the paper is

\textbf{Theorem:}

Let \( B_1, B_2 \) be as stated above, and \( Ip \) the associated interpolation function. Then

\[
\begin{align*}
(a) \quad & B_{1,\text{lin}}(t,df_{n-1},df_n) = 1 - \lambda(t) = \frac{t_n - t}{t_n - t_{n-1}} \quad \text{and} \quad B_{2,\text{lin}}(t,df_{n-1},df_n) = \lambda(t) = \frac{t - t_{n-1}}{t_n - t_{n-1}}
\end{align*}
\]

is an arbitrage-free solution to the system (**) where \( \lambda \) denotes the relative distance of \( t \) to \( t_{n-1} \). The associated interpolation function is linear and expressed by

\[
Ip_{\text{lin}}(t,df_{n-1},df_n) = (1 - \lambda(t))df_{n-1} + \lambda(t)df_n.
\]
(b) \[ B_1^{\text{exp}}(t, df_{n-1}, df_n) = (1 - \lambda(t)) \left( \frac{df_n}{df_{n-1}} \right)^{\lambda(t)} \] and
\[ B_2^{\text{exp}}(t, df_{n-1}, df_n) = \lambda(t) \left( \frac{df_n}{df_{n-1}} \right)^{\lambda(t)-1} \]
is an arbitrage-free solution to the system (**) where \( \lambda \) is as above. The associated interpolation function is as follows:
\[ I^\text{exp}_p(\lambda(t), df_{n-1}, df_n) = B_1^{\text{exp}}(t, df_{n-1}, df_n) \cdot df_{n-1} + B_2^{\text{exp}}(t, df_{n-1}, df_n) \cdot df_n \]
\[ = df_{n-1}^{1-\lambda(t)} \cdot d\lambda(t) \]

(c) The two bucketing procedures are approximately the same which can be seen from the first term of the Taylor series expansion of the exponential interpolation. Let \( \lambda \) be between 0 and 1 and \((df_{n-1}^0, df_n^0)\) be fixed. Then
\[ I^\text{exp}_p(\lambda, df_{n-1}^0, df_n^0) = I^\text{exp}_p(\lambda, df_{n-1}^0, df_n^0) + \frac{\partial}{\partial df_{n-1}} I^\text{exp}_p(\lambda, df_{n-1}^0, df_n^0)(df_{n-1}^0 - df_{n-1}^0) \]
\[ + \frac{\partial}{\partial df_n} (df_{n-1}^0, df_n^0)(df_n^0 - df_n^0) + R_1(df_{n-1}^0, df_n^0) \]
\[ = I^\text{exp}_p(\lambda, df_{n-1}^0, df_n^0) + (1 - \lambda)(df_{n-1}^0)^{\lambda-1}(df_n^0 - df_n^0) \]
\[ + \lambda (df_{n-1}^0)^{1-\lambda}(d\lambda) + R_1(df_{n-1}^0, df_n^0) \]
\[ = (1 - \lambda) \left( \frac{df_0}{df_{n-1}^0} \right)^{\lambda} \cdot df_{n-1}^0 + \lambda \left( \frac{df_n^0}{df_{n-1}^0} \right)^{\lambda-1} \cdot df_n + R_1(df_{n-1}^0, df_n^0) \]

For small values of \( t_n - t_{n-1} \) one has
\[ R_1(df_{n-1}^0, df_n^0) = 0 \quad \text{and} \quad \frac{df_n^0}{df_{n-1}^0} = 1, \]
and therefore,
\[ I^\text{exp}_p(\lambda, df_{n-1}^0, df_n^0) = (1 - \lambda) \cdot df_{n-1}^0 + \lambda \cdot df_n = I^\text{lin}_p(\lambda, df_{n-1}^0, df_n^0). \]

**Remark 3**

(1) The boundary conditions specified for our differential equations by no means guarantee a unique solution.

(2) The solution in (b) can be slightly generalized, if \( \lambda(t) \) is replaced by a strictly increasing continuous function with values between 0 and 1.
(3) The weighted exponential interpolation does not yield a robust bucketing procedure as

\[
\frac{\partial I^\text{weight exp}p(t,df_{n-1},df_n)}{\partial df_{n-1}} = \alpha_{n-1}(1-\lambda)\left(\frac{df_n}{df_{n-1}}\right)^{\alpha_{n-1}(1-\lambda)-1} \quad \text{and}
\]

\[
\frac{\partial I^\text{weight exp}p(t,df_{n-1},df_n)}{\partial df_n} = \alpha_n\lambda\left(\frac{df_n}{df_{n-1}}\right)^{\alpha_n\lambda-1}
\]

produces the following expressions for the bucketing functions \(B_1^{\text{weight exp}}\) and \(B_2^{\text{weight exp}}\):

\[
B_1^{\text{gew exp}}(t,df_{n-1},df_n) = \alpha_{n-1}(t)\cdot(1-\lambda(t))\cdot\left(\frac{df_n}{df_{n-1}}\right)^{\alpha_{n-1}(t)(1-\lambda(t))-1} \quad \text{and}
\]

\[
B_2^{\text{gew exp}}(t,df_{n-1},df_n) = \alpha_n(t)\cdot\lambda(t)\cdot\left(\frac{df_n}{df_{n-1}}\right)^{\alpha_n(t)\lambda(t)-1}
\]

The weighted exponential interpolation method, although still often used, satisfies neither the no arbitrage nor the robustness condition.

**References:**


## Arbeitsberichte der Hochschule für Bankwirtschaft

<table>
<thead>
<tr>
<th>Nr.</th>
<th>Autor/Titel</th>
<th>Jahr</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Moormann, Jürgen&lt;br&gt;Lean Reporting und Führungsinformationssysteme bei deutschen Finanzdienstleistern</td>
<td>1995</td>
</tr>
<tr>
<td>2.</td>
<td>Cremers, Heinz; Schwarz, Willi&lt;br&gt;Interpolation of Discount Factors</td>
<td>1996</td>
</tr>
</tbody>
</table>

### Bestelladresse:

Hochschule für Bankwirtschaft  
z. H. Frau Glatzer  
Sternstraße 8  
60318 Frankfurt/M.

Tel.: 069/95946-16  
Fax: 069/95946-28