Peter Jäckel[*]

The Implementation of the Discrete Gamma Pool model

First version: 19th September 2007
This version: 24th April 2008

Abstract

We discuss implementation details of the Discrete Gamma Pool model [Jäc07].

1 Introduction

The valuation of derivatives with the model proposed in [Jäc07] can be seen as a discrete approximation to a boundary value problem based on a partial integro differential equation, namely

$$\partial_t\tilde{V} - \kappa\cdot y\cdot\partial_y\tilde{V} + \frac{\sigma^2}{2}\cdot\partial_{yy}\tilde{V} + \mathsf{E}\left[\tilde{V}(t,y,\ell+\Delta\ell) + \tilde{\mathcal{P}}(t,\ell,\ell+\Delta\ell)\,\middle|\,\ell\right] - \tilde{V} \;=\; 0 \tag{1}$$

with $\tilde{V} = \tilde{V}(t,y,\ell)$ being the discounted value of the contingent claim at hand, $\tilde{\mathcal{P}}(t,\ell,\ell+\Delta\ell)$ standing for cashflows triggered by losses of size $\Delta\ell$ conditional on the current loss level in the overall portfolio being ℓ, and the expectation $\mathsf{E}[\cdot]$ being over the unknown Lévy measure associated with the dynamics of the loss ℓ. For any individual pricing problem, the contractual details determine what initial values are to accompany the PIDE (1), what the loss induced payment term $\tilde{\mathcal{P}}$ represents, and what further transition rules are to be applied.

2 The lattice implementation

The model's valuation is implemented as an explicit finite differencing scheme on a discretisation in the direction of time, the hazard rate driving process y, and the portfolio losses ℓ. Denote as $V_{i\,j\,k}$ the value of the contingent claim at the discretisation node associated with time t_i, hazard rate level y_j, and loss level ℓ_k. Denote the number of hazard rate levels in the lattice as n_y and the number of loss levels as n_ℓ. The hazard rate discretisation is done symmetrically (around zero) in terms of the driving Ornstein-Uhlenbeck process y, and the levels y_k are spaced equally in the y direction, i.e. $y_j - y_{j-1} = \Delta y$ for all $j > 1$ and $j \le n_y$. The loss discretisation is always laid out such that $\ell_1 = 0$ and $\ell_{n_\ell} = 1 - R$, but no assumption about homogeneity is made. In fact, the loss levels are deliberately aligned such that we always have a node on each of the attachment and detachment points of any involved tranches, as well as on all applicable trigger levels. We refer to all loss levels that we wish to have represented exactly under any circumstances as the set of *mandatory loss levels*. For instance, for a product involving a 5% loss trigger level, on a 10%–50%, tranche, for a portfolio with $R = 60\%$ recovery, this set would be $\{0, 0.05, 0.1, 0.5, 0.6\}$. In between any pair of adjacent mandatory loss discretisation levels, loss nodes are spaced homogeneously with the smallest number of intermediate nodes such that an externally specified maximum loss discretisation distance, e.g.

[*]Global head of Credit, Hybrid, Inflation, and Commodity Derivative Analytics, ABN AMRO

0.5%, is not exceed over any loss discretisation step. Since the family of products valued with this model includes loss triggered leveraged super-senior tranches with time varying trigger levels, the set of loss levels is also allowed to vary from one coupon period to the next, and a complete rebuilding of the loss level discretisation is invoked when necessary. The values $V_{i\,j\,k}$ are interpolated linearly over the nearest two loss levels prior to re-discretisation for any time t_i that is a transition between two periods where the set of mandatory loss levels changes.

In any given backwards induction step, the value at an earlier (interior) node is computed according to

$$\begin{aligned}\tilde{V}_{i-1\,j\,k} \;=\; & \sum_{k'=k}^{n_\ell} \pi_{k\,k'} \left(\tilde{V}_{i\,j\,k'} + \tilde{\mathcal{P}}_{i\,k\,k'} \right) \\ & + \Delta t \cdot \left(\frac{\sigma^2}{2} \cdot \frac{\tilde{V}_{i\,j+1\,k} - 2\tilde{V}_{i\,j\,k} + \tilde{V}_{i\,j-1\,k}}{\Delta y^2} - \kappa \cdot \frac{\tilde{V}_{i\,j+1\,k} - \tilde{V}_{i\,j-1\,k}}{2\Delta y} \right) . \end{aligned} \tag{2}$$

The term $\tilde{\mathcal{P}}_{i\,k\,k'}$ denotes the t_i-discounted value of any loss induced cashflows occurring at time t_i if portfolio losses jump from level ℓ_k to level $\ell_{k'}$. For nodes on the boundary in the y direction, we use

$$\begin{aligned}\tilde{V}_{i-1\,1\,k} \;=\; & \sum_{k'=k}^{n_\ell} \pi_{k\,k'} \left(\tilde{V}_{i\,1\,k'} + \tilde{\mathcal{P}}_{i\,k\,k'} \right) \\ & + \Delta t \cdot \left(\frac{\sigma^2}{2} \cdot \frac{\tilde{V}_{i\,3\,k} - 2\tilde{V}_{i\,2\,k} + \tilde{V}_{i\,1\,k}}{\Delta y^2} - \kappa \cdot \frac{4\tilde{V}_{i\,2\,k} - 3\tilde{V}_{i\,1\,k} - \tilde{V}_{i\,3\,k}}{2\Delta y} \right) \end{aligned} \tag{3}$$

$$\begin{aligned}\tilde{V}_{i-1\,n_y\,k} \;=\; & \sum_{k'=k}^{n_\ell} \pi_{k\,k'} \left(\tilde{V}_{i\,n_y\,k'} + \tilde{\mathcal{P}}_{i\,k\,k'} \right) \\ & + \Delta t \cdot \left(\frac{\sigma^2}{2} \cdot \frac{\tilde{V}_{i\,n_y\,k} - 2\tilde{V}_{i\,n_y-1\,k} + \tilde{V}_{i\,n_y-2\,k}}{\Delta y^2} - \kappa \cdot \frac{\tilde{V}_{i\,n_y-2\,k} + 3\tilde{V}_{i\,n_y\,k} - 4\tilde{V}_{i\,n_y-1\,k}}{2\Delta y} \right) \end{aligned} \tag{4}$$

which means that the scheme is of second order in the y direction throughout. The transition probabilities $\pi_{k\,k'}$ are determined by the Discrete Gamma loss process, and we discuss their derivation in more detail further on.

A technical inconvenience of the Discrete Gamma Pool model is that the matrix of transition probabilities $\pi_{k\,k'}$ is dense in its upper right triangle since all loss levels above the current one are attainable in any time step for any finite value of γ. In fact, it is this feature of the underlying Gamma-Phi model that was one of the main reasons why it was selected to govern the loss transition probabilities. Alas, a consequence of the dense transition probability matrix is a considerable computational burden. As it turns out, the main effort is the calculation of the transition probability coefficients $\pi_{k\,k'}$, and less so the subsequent valuation itself, even when taking into account the valuation of several contracts in parallel with all involved transition rules. Since the calculation of these transition probabilities, and the backwards induction (2), for any one time step, splits readily in a sequence of n_y calculations for each of the hazard rate levels, the model's implementation chooses to loop over j outside the loop over k, and parallelises the calculations for different j over a pool of threads that is, by default, the size of the number of CPUs on the respective computer. In practice, we found that even on dual-quad-core machines that have a total of eight effective CPUs with associated floating point unit, we achieve up to 90% utilisation of all the involved cores in this fashion.

3 The loss transition probabilities

The discrete backwards induction equation (2) consists of two parts, namely of a finite-differencing discretisation of the generator of the Ornstein-Uhlenbeck process y, and of a loss-discretised approximation to the expectation

$$\int_{\ell_{\text{min}}(\lambda,\ell)}^{(1-R)} \Big(\tilde{V}(t,y,\ell') + \tilde{\mathcal{P}}(t,\ell,\ell')\Big)\,\psi(\lambda,\ell,\ell')\,\mathrm{d}\ell' \;. \tag{5}$$

The loss transition density $\psi(\lambda,\ell,\ell')$ is given by

$$\psi(\lambda,\ell,\ell') \;=\; \frac{g\big(\gamma\phi\,,\,\Gamma^{-1}(\gamma,\mathrm{e}^{-\lambda\Delta t}) - \Gamma^{-1}(\gamma(1-\phi),\frac{1-R-\ell'}{1-R-\ell})\big)}{(1-R-\ell)\cdot g\big(\gamma(1-\phi)\,,\,\Gamma^{-1}(\gamma(1-\phi),\frac{1-R-\ell'}{1-R-\ell})\big)} \tag{6}$$

with

$$\ell_{\text{min}}(\lambda,\ell) \;=\; \ell + \big(1 - \Gamma(\gamma(1-\phi),\Gamma^{-1}(\gamma,\mathrm{e}^{-\lambda\Delta t}))\big)\cdot(1-R-\ell)\;. \tag{7}$$

and

$$g(\alpha,x) \;=\; x^{\alpha-1}\mathrm{e}^{-x}\big/\,\Gamma(\alpha)\;. \tag{8}$$

The transition density (6) is the core of the Discrete Gamma Pool model. Depending on the choice of parameters, it can be numerically difficult to manage, as becomes clear by the variety of shapes and scales of magnitude it can attain as discussed in [Jäc07]. For the loss-discretised evaluation in the lattice implementation, we need to choose non-negative discrete loss transition probabilities $\pi_{k\,k'}$ such that

$$\sum_{k'=k}^{n_\ell} \pi_{k\,k'}\Big(\tilde{V}_{i\,j\,k'} + \tilde{\mathcal{P}}_{i\,k\,k'}\Big) \;\approx\; \int_{\ell_{\text{min}}(\lambda(t_i,y_j),\ell_k)}^{(1-R)} \Big(\tilde{V}(t_i,y_j,\ell') + \tilde{\mathcal{P}}(t_i,\ell_k,\ell')\Big)\,\psi(\lambda(t_i,y_j),\ell_k,\ell')\,\mathrm{d}\ell' \tag{9}$$

with

$$\lambda(t_i,y_j) \;=\; \hat{\lambda}\cdot\mathrm{e}^{-\frac{1}{2}\mathsf{V}_0[y(t_{i-1})]+y_j}\;. \tag{10}$$

The problem of choosing suitable weights $\pi_{k\,k'}$ belongs to the more general family of quadratures. However, since the levels $\ell_{k'}$ are not to be varied on behalf of the quadrature (9), and since they are definitely not equally spaced, we are not dealing with a standard setting as for Gaussian, or Newton-Cotes quadratures. Notwithstanding the non-standard nature of the task, the choice of weights $\pi_{k\,k'}$ becomes unique once one selects a set of $\hat{n}_{\ell_k}$ specific functions $\chi_{k''}(\ell')$ for which one demands the integral to be exact, i.e.

$$\sum_{k'=k}^{n_\ell} \pi_{k\,k'}\cdot\chi_{k''}(\ell_{k'}) \;=\; \int_{\ell_{\text{min}}(\lambda(t_i,y_j),\ell_k)}^{(1-R)} \chi_{k''}(\ell')\,\psi(\lambda(t_i,y_j),\ell_k,\ell')\,\mathrm{d}\ell' \tag{11}$$

with

$$\hat{n}_{\ell_k} \;=\; n_\ell - \lfloor \ell_{\text{min}}(\lambda(t_i,y_j),\ell_k)\rfloor + 1 \tag{12}$$

and $k'' = 1,...,\hat{n}_{\ell_k}$. The addition of 1 on the right hand side of (11) represents the fact that we allow the sum on the left hand side in (9) to start at k, which means that we allow positive probability for the transition from $\ell_k \to \ell_k$, i.e. for no loss to incur. This is, strictly speaking, in contrast to the analytical properties of the transition density. However, this choice improves the overall accuracy of the numerical approximation. This can be understood intuitively: the bulk of the transition density is in the direct vicinity of the current loss level when spreads are at normal investment grade market levels, or lower, whence, as a discretised approximation, most of the transition probability must be on the same and the next loss level. When λ is so high that $\ell_{\min}(\lambda, \ell_k) > \ell_{k+1}$, we do, of course set all $\pi_{k\,k'}$ with $\ell_{k'+1} < \ell_{\min}(\lambda, \ell_k)$ to zero. This does indeed happen on some levels in y since the associated hazard rate levels $\lambda(y)$ can be very large when y is three, four, or even five standard deviations up from its mean, as is usual on a finite differencing lattice.

For quadratures of very smooth functions, the natural choice for the basis functions $\chi_{k''}(\ell')$ is often a set of monomials, either globally with increasing order (as for Gaussian quadratures), or, when the nodes are subdivided into groups, a set of low order local polynomials (as Newton-Cotes rules). However, in our setting, there are two reasons why we would not want to choose monomials as basis functions. The first reason is that we want the algorithm to match as closely as possible all functions that resemble what could be considered as effective payoff functions. For our target products, this typically entails linear terms interleaved with various $\min(\cdot,\cdot)$ and $\max(\cdot,\cdot)$ expressions, and functions of this type are notoriously ill represented by monomials due to their points of non-differentiability. In fact, we choose loss levels to be placed exactly on all such points of interest specifically so that we can represent the nonlinearity in the payoff accurately, hence it wouldn't make sense to abandon this advantage now. The second reason is that the inhomogeneity in the placement of loss levels can give rise to numerical instabilities in the solution of the resulting linear system for the weights $\pi_{k\,k'}$. In analogy to the set of basis functions that are used for standard linear finite element methods, we therefore choose as follows.

First, denote the *left half-hat function* as

$$\chi^{(-)}_{k'}(\ell) \;=\; \frac{\ell - \ell_{k'-1}}{\ell_{k'} - \ell_{k'-1}} \cdot \mathbf{1}_{\left\{\ell \in [\ell_{k'-1}, \ell_{k'})\right\}} , \tag{13}$$

and the *right half-hat function* as

$$\chi^{(+)}_{k'}(\ell) \;=\; \frac{\ell_{k'+1} - \ell}{\ell_{k'+1} - \ell_{k'}} \cdot \mathbf{1}_{\left\{\ell \in [\ell_{k'}, \ell_{k'+1})\right\}} . \tag{14}$$

Then, define the *asymmetric hat functions*

$$\chi_{k'}(\ell) \;=\; \chi^{(-)}_{k'}(\ell) + \chi^{(+)}_{k'}(\ell) . \tag{15}$$

The set of functions we demand to be integrated exactly by the discretisation for any given k is now defined by:-

- The right half-hat function $\chi^{(+)}_{k}(\ell)$.
- All $\chi_{k'}(\ell)$ for $k < k' < n_\ell$.

We show an example for this in figure 1. In addition, we demand that

$$\sum_{k'=k}^{n_\ell} \pi_{k\,k'} \;=\; 1 \tag{16}$$

which means that probability remains normalized. It is clear that this choice means

$$\pi_{k\,k'} \;=\; \int\limits_{\ell_{\min}(\lambda(t_i, y_j), \ell_k)}^{(1-R)} \chi_{k'}(\ell')\, \psi(\lambda(t_i, y_j), \ell_k, \ell')\, \mathrm{d}\ell' \tag{17}$$

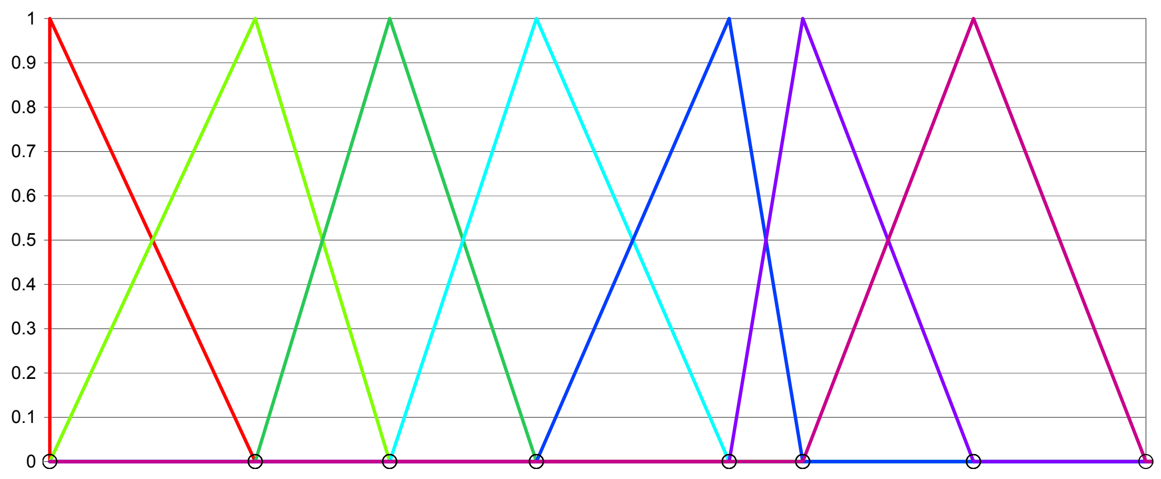

Figure 1: The linear finite element basis functions chosen to be matched exactly by the integration over the actual loss transition density as stated in equation (9) for an arbitrary inhomogeneous discretisation in the loss direction.

for all k' that satisfy $k \le k' < n_\ell$, and that the last weight is given by

$$\pi_{k\,n_\ell} \;=\; 1 - \sum_{k'=k}^{n_\ell-1} \pi_{k\,k'} \;. \tag{18}$$

For the first weight $\pi_{k\,k}$, we can substitute the definition of the right half-hat function, and integrate by parts to obtain

$$\pi_{k\,k} \;= \int\limits_{\ell_{\min}(\ell_k)}^{\ell_{k+1}} \tfrac{\ell_{k+1}-\ell}{\ell_{k+1}-\ell_k} \cdot \psi(\ell_k,\ell')\,\mathrm{d}\ell' \tag{19}$$

$$= \left[\tfrac{\ell_{k+1}-\ell'}{\ell_{k+1}-\ell_k} \cdot \Psi(\ell_k,\ell')\right]_{\ell'=\ell_{\min}(\ell_k)}^{\ell'=\ell_{k+1}} + \frac{1}{\ell_{k+1}-\ell_k} \int\limits_{\ell_k}^{\ell_{k+1}} \Psi(\ell_k,\ell')\,\mathrm{d}\ell' \tag{20}$$

where we have dropped the dependencies on λ for clarity. The first term on the right hand side of (20) is zero, and thus we have the loss transition transition probability given by an integral over the *cumulative loss transition probability*:

$$\pi_{k\,k} \;=\; \frac{1}{\ell_{k+1}-\ell_k} \int\limits_{\ell_k}^{\ell_{k+1}} \Psi(\ell_k,\ell')\,\mathrm{d}\ell' \tag{21}$$

For all $k' > k$ with $k' < n_\ell$, we can derive in complete analogy

$$\pi_{k\,k'} \;=\; \frac{1}{\ell_{k'+1}-\ell_{k'}} \int\limits_{\ell_{k'}}^{\ell_{k'+1}} \Psi(\ell_k,\ell')\,\mathrm{d}\ell' - \frac{1}{\ell_{k'}-\ell_{k'-1}} \int\limits_{\ell_{k'-1}}^{\ell_{k'}} \Psi(\ell_k,\ell')\,\mathrm{d}\ell' \;, \tag{22}$$

as well as finally

$$\pi_{k\,n_\ell} \;=\; 1 - \frac{1}{\ell_{n_\ell}-\ell_{n_\ell-1}} \int\limits_{\ell_{n_\ell-1}}^{\ell_{n_\ell}} \Psi(\ell_k,\ell')\,\mathrm{d}\ell' \;. \tag{23}$$

This means that we can compute all discrete loss transition probabilities from a sequence of integrals over the cumulative loss transition probability, and we discuss in the next section how this is done in a numerically robust fashion. An advantage of the above scheme for the calculation of the discrete loss transition probabilities is that the integration of the cumulative loss transition probability is numerically more tractable than integration of the density itself.

4 Integrating the cumulative loss transition probability

The cumulative loss transition probability is given by

$$\Psi(\ell,\ell') \;=\; \Gamma\left(\gamma\phi\,,\,\Gamma^{-1}(\gamma,\mathrm{e}^{-\lambda\Delta t}) - \Gamma^{-1}(\gamma(1-\phi),\tfrac{1-R-\ell'}{1-R-\ell})\right)\;. \tag{24}$$

The involved normalised incomplete Gamma function as defined by

$$\Gamma(\alpha,x) \;=\; \int_0^{x_+} g(\alpha,x')\,\mathrm{d}x' \tag{25}$$

and its inverse can be computed comparatively efficiently by the aid of the algorithm published by DiDonato and Morris [DM87]. In order to integrate over the cumulative loss transition probability from $\ell_{k'}$ to $\ell_{k'+1}$, for all interior intervals with $k' > k+1$ such that $\Psi(\ell_k,\ell_{k'-1}) > 0$, we use a *four-point rational interpolation* of the probability function through the four points

$$\begin{aligned}
x_0 &= 0 & f_0 &= 0\\
x_1 &= \ell_{k'+1}-\ell_{k'} & f_1 &= \Psi(\ell_k,\ell_{k'+1})-\Psi(\ell_k,\ell_{k'})\\
x_2 &= \ell_{k'+2}-\ell_{k'} & f_2 &= \Psi(\ell_k,\ell_{k'+2})-\Psi(\ell_k,\ell_{k'})\\
x_3 &= \ell_{k'-1}-\ell_{k'} & f_3 &= \Psi(\ell_k,\ell_{k'-1})-\Psi(\ell_k,\ell_{k'})
\end{aligned} \tag{26}$$

with the functional form

$$f(x) \;=\; \frac{bx+dx^2}{c+x}\;. \tag{27}$$

We selected this rational function since we found that polynomial interpolation results in significant approximation errors due to the fact that the cumulative probability function has domains of extremely steep and extremely flat slopes. Rational interpolation, whilst typically being slightly more complex in its implementation, is known to cope much better with functions of this type [PTVF92], and indeed we achieved excellent results with the method discussed here.

The coefficients b, c, and d in (27) are given by

$$b \;=\; \frac{-f_1f_3x_2^2(x_1-x_3)+f_2f_3x_1^2(x_2-x_3)+f_2f_1x_3^2(x_1-x_2)}{f_3x_1(x_1-x_2)x_2+f_1x_2(x_2-x_3)x_3-f_2x_1(x_1-x_3)x_3} \tag{28}$$

$$c \;=\; \frac{x_1x_2x_3(f_2(x_1-x_3)-f_1(x_2-x_3)-f_3(x_1-x_2))}{f_3x_1(x_1-x_2)x_2+f_1x_2(x_2-x_3)x_3-f_2x_1(x_1-x_3)x_3} \tag{29}$$

$$d \;=\; \frac{-f_2f_3x_1(x_2-x_3)+f_1f_3x_2(x_1-x_3)-f_1f_2x_3(x_1-x_2)}{f_3x_1(x_1-x_2)x_2+f_1x_2(x_2-x_3)x_3-f_2x_1(x_1-x_3)x_3}\;. \tag{30}$$

Note that the expressions for b, c, and d do not depend on the abscissa values x_1, x_2, and x_3 to be ordered. Also take notice that the benefit of the coordinate translation such that the integrand goes through the origin is not only that it simplifies the subsequent analytical expressions, it also turns out to *significantly stabilise the numerical implementation with respect to round-off errors*[1]. The integral over the cumulative probability is then approximated by

$$\int_{\ell_{k'}}^{\ell_{k'+1}} \Psi(\ell_k,\ell')\,\mathrm{d}\ell' \;\approx\; \Psi(\ell_k,\ell_{k'})\cdot(\ell_{k'+1}-\ell_{k'}) + \int_0^{x_1} f(x)\,\mathrm{d}x \;. \tag{31}$$

The integral over $f(x)$ is given by

$$\int_0^{x_1} f(x)\,\mathrm{d}x \;=\; \big(x_1^2/2 + D\cdot(\ln(1+e)-e)\big)\cdot d \tag{32}$$

with

$$D \;=\; c(c-b/d) \tag{33}$$

$$e \;=\; x_1/c\;. \tag{34}$$

In case the coefficient d is zero, the simpler formula

$$\left.\int_0^{x_1} f(x)\,\mathrm{d}x\right|_{d=0} \;=\; (x_1 - c\cdot\ln(1+e))\cdot b \tag{35}$$

is used. Attention is being paid to underflow of $\ln(1+e)$, which is approximated as

$$\ln(1+e) \;\approx\; e\cdot\left(1+e\cdot\left(\frac{e}{3}-\frac{1}{2}\right)\right) \tag{36}$$

whenever $|e|$ is less than half the machine precision on a logarithmic scale. The right hand side of (32) is then evaluated as

$$x_1{}^2\cdot\left(\frac{b}{2c}+\frac{d}{3c}\cdot x_1-\frac{b}{3c^2}\cdot x_1{}^2\right). \tag{37}$$

Equally, it is checked whether the function $f(x)$ is monotonic over the interval by evaluation as to whether the pole at $-c$ is inside the interval of integration, and by assessment as to whether $f(x)$ has any extrema inside the interval. When any of these conditions fail, the lower order rational function

$$\tilde{f}(x) \;=\; \frac{\tilde{b}x}{\tilde{c}+x} \tag{38}$$

is used for interpolation through the first three of the points in (26), i.e. x_0, x_1, and x_2, with the coefficients

$$\tilde{b} \;=\; \frac{f_1 f_2(x_2-x_1)}{f_1x_2 - f_2x_1} \tag{39}$$

$$\tilde{c} \;=\; \frac{(f_2-f_1)x_1x_2}{f_1x_2-f_2x_1}\;. \tag{40}$$

The integral then becomes

$$\int_0^{x_1}\tilde{f}(x)\,\mathrm{d}x \;=\; (x_1-\tilde{c}\cdot\ln(1+x_1/\tilde{c}))\cdot\tilde{b}. \tag{41}$$

[1]Without the translation of coordinates, we noticed frequently re-occurring round-off errors of prohibitive size for very low levels of λ. When the interpolation is done through points such that one is at $(0,0)$, no round-off errors occurred.

The logarithm may have to be substituted by expansion if $x_1/\tilde{c}$ is too small, as above, in which case the expression evaluated is

$$\frac{b}{c} \cdot \left(\frac{1}{2} - \frac{x_1}{3}\right) \cdot {x_1}^2 . \tag{42}$$

If this lower order rational function has its pole at $-c$ inside the interval, which we have yet not seen occurring in practice, the algorithm falls back to the simple trapezoidal rule

$$\int_0^{x_1} f(x)\,\mathrm{d}x \;\approx\; \frac{x_1 f_1}{2}\ . \tag{43}$$

It remains to be said how the integration is done for the intervals at either end of the loss discretisation. The last interval from $\ell_{n_\ell - 1}$to ℓ_{n_ℓ} is done with the three point rational form (38). The most difficult and most important one since it typically contains most of the probability mass, the integral from ℓ_k to ℓ_{k+1} is done using the four-point rational interpolation (27) through $\ell_{\text{min}}(\ell_k)$, ℓ_{k+1}, ℓ_{k+2}, and an additional point $\ell_{\text{mid}}(\ell_k) := (\ell_{\text{min}}(\ell_k) + \ell_{k+1})/2$ (note that this point is not part of the original loss level discretisation). Integration is from $\ell_{\text{min}}(\ell_k)$ with $\Psi(\ell_k, \ell_{\text{min}}(\ell_k))$=0 by definition. For consistency, the interval $[\ell_{k+1}, \ell_{k+2})$ is handled using the same interpolation points as the integral $[\ell_k, \ell_{k+1})$, after dropping the left-most used point, and adding ℓ_{k+1}. When $\ell_{\text{min}}(\ell_k) \geq \ell_{k+1}$, which can happen for high hazard rate levels, the algorithm finds the greatest m such that $\ell_{\text{min}}(\ell_k) \geq \ell_m$ and performs the above interpolation as if $k = m$ with $\ell_{\text{min}}(\ell_m) = \ell_{\text{min}}(\ell_k)$.

References

[DM87] A. R. DiDonato and A. H. Morris. Incomplete gamma function ratios and their inverse. *ACM TOMS*, 13:318–319, 1987. http://www.netlib.org/toms/654.

[Jäc07] P. Jäckel. The Discrete Gamma Pool Model. www.jaeckel.org/DiscreteGammaPoolModel.pdf, September 2007.

[PTVF92] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes in C*. Cambridge University Press, 1992. www.library.cornell.edu/nr/cbookcpdf.html.