

DFN Generator: Technical Notes

Michael Welch and Michael Lüthje

07/09/2021

1 Introduction

This document provides an overview of the structure of the DFN Generator code, that will be useful to those intending to modify or enhance the DFN Generator tool. In addition, it contains some details of the calculation algorithm that are not included elsewhere.

It is not however intended as a complete description of derivation of the algorithm used by the DFN Generator tool. This is described in detail in the monograph “Modelling the Evolution of Natural Fracture Networks” by Welch et al. (2020) (<https://link.springer.com/book/10.1007/978-3-030-52414-2>). This publication includes a full description of both the conceptual model and the detailed algorithm, as well as the results of a sensitivity study detailing the effects of the main input parameters, and finally some examples of the application of the algorithm to outcrops and subsurface reservoirs. Some additional results are also given in the Petroleum Geoscience paper “Influence of fracture nucleation and propagation rates on fracture geometry: insights from geomechanical modelling” by Welch et al. (2019) (<https://doi.org/10.1144/petgeo2018-161>). We recommend obtaining a copy of these publications before making any modifications or enhancements to the code.

Finally, we welcome any comments, feedback or suggestions for future enhancements. If you are planning to make modifications to the code and would like to discuss these, or need help, please feel free to contact us as mwelch@dtu.dk or mikael@dtu.dk.

1.1 Structure of this document

In this document, we will first describe two specific aspects of the algorithm that are not fully covered in the publications referenced above: calculation of fracture deactivation probabilities for multiple fracture dipsets (described in Section 2), and calculation of mean lengths of residual fractures (described in Section 3). In both cases, the algorithms described here build on the algorithms described in Welch et al. (2020), so it will be necessary to refer to this publication.

In Section 4, we describe in more detail a workflow that uses the algorithm to generate combined implicit and explicit fracture models. This workflow is summarised in Welch et al. 2020, but in this document we describe in more detail some of the methods that are used to streamline the calculation of the explicit DFN, such as local spatial coordinate systems and weighted time coordinates.

Finally in Section 5 we describe the structure of the code. This section includes a description of the principal classes, the methods used to implement the algorithm and how they interact, and the front ends provided.

1.2 Other code enhancements

The current version of the DFN Generator code contains significant enhancements in the calculation of fracture driving stress, stress shadow volume and in situ stress, compared with the algorithm documented in Chapter 6 of Welch et al. (2020). Amongst other benefits, these enhancements allow for:

- Modelling of strike-slip fractures and fractures striking oblique to the principal stress directions.
- Inclusion of multiple fracture sets, allowing modelling of polygonal fracture patterns.
- Full 3D tensor representation of in situ stress and fracture strain.
- Modelling stress shadow interactions between oblique fractures.
- Calculation of the full 3D compliance tensor for a fractured rockmass.

These enhancements are not described in this document, but will be the subject of a forthcoming paper.

The code also contains some “stubs” designed for use in future enhancements. These include new techniques for modelling fractures controlled by ductile layer boundaries, and the capability of filtering fractures in the explicit DFN based on fracture size. At present we have no specified date for completing these enhancements. If you are interested in working on them, or on any other enhancements, please contact us at the email addresses given above.

2 Fracture sets with multiple dipsets

The algorithm described in Chapter 5 of Welch et al. (2020) incorporates the interactions between different fractures as the fracture network develops. This is done through two steps:

- Firstly, we calculate the probability that different fractures will become deactivated due to interaction with other fractures during each timestep (described in Section 5.1 of Welch et al. 2020).
- Then we use this data to calculate the population of both active and static fractures at the end of the timestep (described in Sections 5.2, 5.3 and 5.4 of Welch et al. 2020).

To calculate the probability of fracture interactions, we define fracture sets comprising all fractures with a specified fracture strike, typically perpendicular to the principal horizontal strain directions. Each fracture is considered to be surrounded by a stress shadow, in which applied strain is accommodated by displacement on the fracture rather than by elastic strain in the rockmass. We then consider three mechanisms by which fractures may cease propagating:

- Microfractures may cease propagating if they fall into the stress shadow of a nearby parallel propagating macrofracture.
- Macrofractures may also cease propagating if they fall into the stress shadow of another parallel propagating macrofracture.
- Finally, macrofractures may cease propagating if they intersect an orthogonal or oblique macrofracture.

In all three cases, we must consider how the width of the stress shadow around individual fractures will affect the spatial distribution of fractures. This is described in Appendix 2 of Welch et al. 2020. However the algorithm described in Welch et al. 2020 assumes that all fractures in a given set have the same stress shadow width. This assumption will be valid if the fractures all have the same dip and mode (e.g. if they are all vertical Mode 1 dilatant fractures, or if they are all optimally-inclined Mode 2 shear fractures). However in order to fully generalise the model, we must also be able to model fracture sets containing fractures with different dips and modes. We cannot just split them into separate fracture sets, since fractures with the same strike will interact with each other by stress shadow overlap, not by intersection, which is the interaction mechanisms that applies between different fracture sets. Instead we can subdivide the fracture sets into multiple “dipsets”, each of which has a different dip, mode and stress shadow width, but which interact with each other by stress shadow overlap rather than by intersection.

The spatial distribution for such a fracture set will still be described by a single spacing distribution function that must be calculated for the entire set; however the calculation is more complex as it must take into account the different stress shadow widths surrounding fractures from different dipsets. A method for calculating this is given in Section 2.1. We then show how to use this function to calculate the probability of stress shadow interaction and intersection for the different dipsets, in Sections 2.2, 2.3 and 2.4.

Once we have calculated the fracture deactivation probabilities of the different dipsets, we can use the same algorithm as described Sections 5.2, 5.3 and 5.4 of Welch et al. 2020 to calculate the population of active and static fractures at the end of each timestep. Note that in the DFN Generator code we include 2 dipsets in each fracture set: the first comprising vertical Mode 1 fractures and the second comprising optimally-inclined Mode 2 shear fractures. However the algorithms described in Sections 2.2, 2.3 and 2.4 are valid for any number of dipsets.

2.1 Macrofracture distribution with stress shadows of different widths

The arguments above assume that all the macrofracture stress shadows have the same width, W . However it is possible to envisage a scenario in which the macrofractures have different stress shadow widths, for example if we have Mode 1 and Mode 2 macrofractures, or if we have cemented and uncemented macrofractures with different mechanical properties.

In this section we will consider a scenario in which a fracture set can be subdivided into n dipsets, $M1, M2, M3, \dots, M_n$, with different macrofracture stress shadow widths $W_1, W_2, W_3, \dots, W_n$, such that $W_1 \geq W_2 \geq W_3 \geq \dots \geq W_n$. Note that the macrofracture stress shadow widths will not change through time.

2.1.1 Cumulative macrofracture spacing distribution

We can divide the cumulative macrofracture spacing distribution function $S(x)$ into segments, from 0 to W_n , from W_n to W_{n-1} , \dots , from W_2 to W_1 , and greater than W_1 (see *Figure 1*). However, as we will see, it is not possible to come up with an exact expression for the cumulative macrofracture spacing distribution function $S(x)$ in any of these segments except the final one ($x \geq W_1$). This was also the case for the uniform stress shadow scenario, where there was no exact expression for $S(x)$ where $x < W$; however in the current scenario it presents a more serious problem, since we will need to integrate $S(x)$ across many of these segments. To get around this problem, we will approximate these segments as offset exponential functions, so that the cumulative macrofracture spacing distribution function $S(x)$ will take the form

$$[1] \quad S(x) = \begin{cases} A_1 e^{-B_1(x-W_1)} & \text{for } x \geq W_1 \\ (A_2 + C_2) e^{-B_2(x-W_2)} - C_2 & \text{for } W_2 \leq x \leq W_1 \\ (A_3 + C_3) e^{-B_3(x-W_3)} - C_3 & \text{for } W_3 \leq x \leq W_2 \\ \vdots & \vdots \\ (A_n + C_n) e^{-B_n(x-W_n)} - C_n & \text{for } W_n \leq x \leq W_{n-1} \\ (C_{MF32} + C_{n+1}) e^{-B_{n+1}x} - C_{n+1} & \text{for } x \leq W_n \end{cases}$$

Note that $C_1 = 0$, and from [1], we can see that for any dipset q ,

$$[2] \quad \begin{aligned} S(W_q) &= A_q = (A_{q+1} + C_{q+1}) e^{-B_{q+1}(W_q - W_{q+1})} - C_{q+1} \\ \Rightarrow \frac{A_q + C_{q+1}}{A_{q+1} + C_{q+1}} &= e^{-B_{q+1}(W_q - W_{q+1})} \\ \Rightarrow B_{q+1} &= \frac{\ln(A_{q+1} + C_{q+1}) - \ln(A_q + C_{q+1})}{W_q - W_{q+1}} \end{aligned}$$

Thus the only independent variables we need to define the cumulative macrofracture spacing distribution function $S(x)$ (apart from the stress shadow widths $W_1, W_2, W_3, \dots, W_n$) are the A values $A_1, A_2, A_3, \dots, A_n$, the first B value B_1 , and the C values.

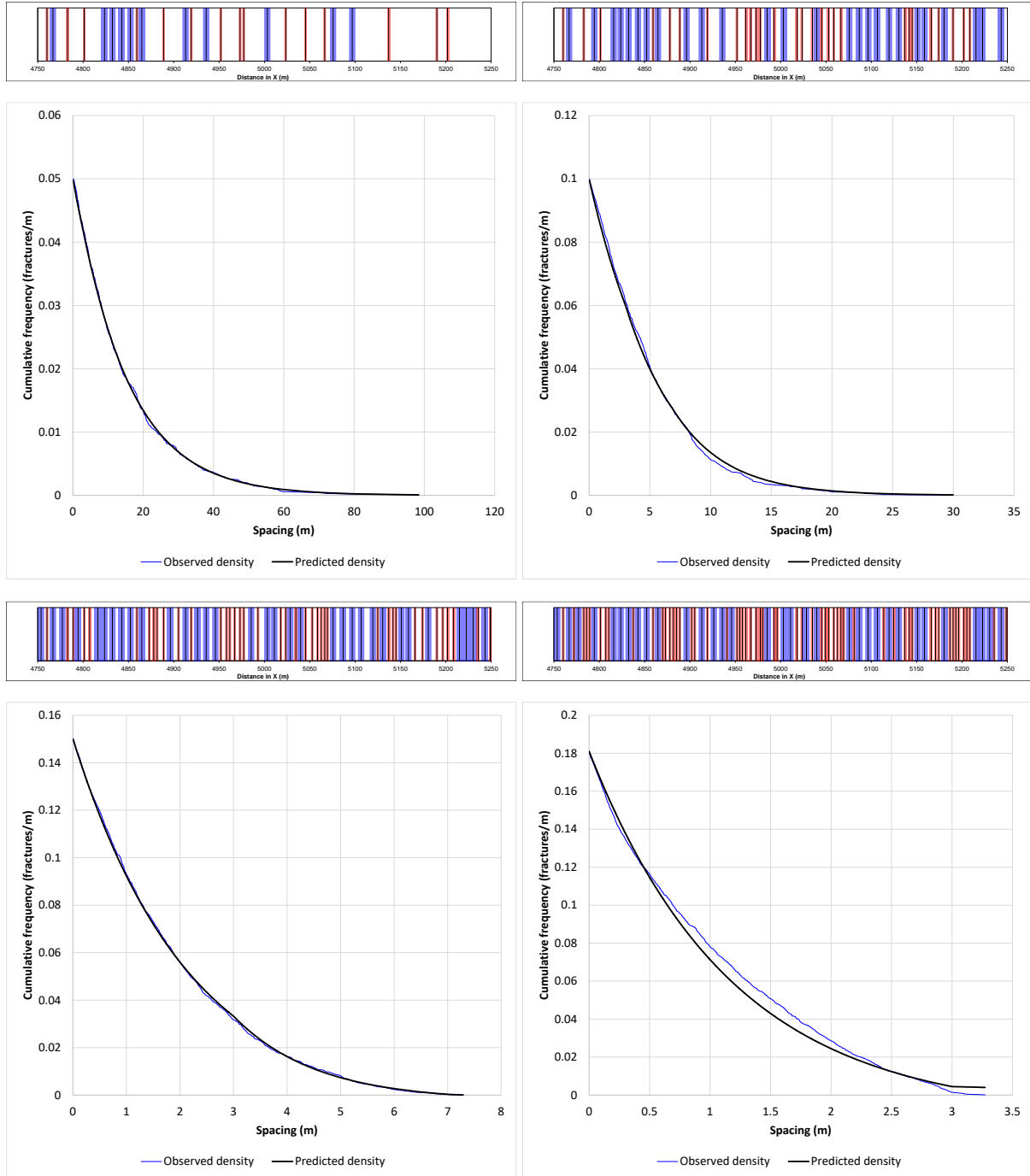


Figure 1: Cumulative macrofracture spacing distributions obtained for a fracture set with two dipsets, of stress shadow width $W_1=7$ and $W_2=3$. New macrofractures are added at random locations along a transect of length 10000; they are randomly assigned to one of the two dipsets, with equal probability, and the random location selected is checked to see if the new macrofracture will overlap the stress shadow of an existing macrofracture. Initially, therefore, new macrofractures from both sets are added at an equal rate ($d^1_{MF}P_{32}/dt = d^2_{MF}P_{32}/dt$); however these rates diverge as the space available to add new macrofractures from dipset 1 ($1-\chi_1$) decreases more rapidly than the space available to add new macrofractures from dipset 2 ($1-\chi_2$). After 1808 new macrofractures (of both sets) had been added, there was no space to add any more dipset 1 macrofractures; however it was possible to add a further 407 dipset 2 macrofractures. The top left chart shows the spacing distribution after 500 fractures were added, when 24.5% of the transect was covered by stress

shadows; the top right chart shows the spacing distribution after 1000 fractures were added, when 48.7% of the transect was covered by stress shadows; the bottom left chart shows the spacing distribution after 1500 fractures were added, when 72.2% of the transect was covered by stress shadows; the right hand chart shows the final spacing distribution stage, reached after 1808 macrofractures had been added and 81.6% of the transect was covered by stress shadows. The blue line represents the cumulative spacing distribution of the actual macrofractures, while the black line represents the theoretical cumulative spacing distribution derived from [1]. Above each chart is shown a small portion of each transect; the red and blue shading represents the 3m and 7m wide stress shadows, respectively.

2.1.2 Total exclusion zone volume

Before we derive expressions to describe the evolution of these independent variables, we can use [1] directly to derive expressions for the exclusion zone volume χ and the clear zone volume $1-\chi$. As we noted previously, the exclusion zone volume depends on the stress shadow width of the new macrofractures we are adding, as well as on the stress shadow width of the existing macrofractures. Thus we can define a series of exclusion zone volumes, one for each fracture dipset, labelled $\chi_1, \chi_2, \chi_3, \dots, \chi_n$. Note that the fracture dipset identifier represents the dipset of the fracture that is nucleating or propagating.

We can calculate the clear zone volume $1-\chi_q$ for a specific dipset q in the same way as for the uniform stress shadow scenario (see Figure 2). Since new macrofractures with a stress shadow width W_q can only be placed in spacings with length greater than W_q , and within these spacings they cannot be placed within a distance $W_q/2$ of each endpoint, the proportion of the transect in which such macrofractures can be placed is given by

$$\begin{aligned}
 [3] \quad 1 - \chi_q &= \int_{W_q}^{\infty} -\frac{dS(x_n)}{dx_n} (x_n - W_q) dx_n = \int_0^{\infty} -\frac{dS(x_w + W_q)}{dx_w} x_w dx_w \\
 &= \int_0^{W_{q-1}-W_q} -\frac{dS(x_w + W_q)}{dx_w} x_w dx_w + \int_{W_{q-1}-W_q}^{W_q-2-W_q} -\frac{dS(x_w + W_q)}{dx_w} x_w dx_w + \dots \int_{W_1-W_q}^{\infty} -\frac{dS(x_w + W_q)}{dx_w} x_w dx_w \\
 &= \sum_{s=1}^q \left[\int_{W_s-W_q}^{W_{s-1}-W_q} -\frac{dS(x_w + W_q)}{dx_w} x_w dx_w \right]
 \end{aligned}$$

where $x_w = x_n - W_q$ and $W_0 = \infty$. We can solve [3] by noting that, for $W_s \leq x \leq W_{s-1}$,

$$[4] \quad S(x) = (A_s + C_s)e^{-\mathbb{B}_s(x-W_s)} - C_s = (A_s + C_s)e^{\mathbb{B}_s(W_s-W_q)}e^{-\mathbb{B}_s(x-W_q)} - C_s$$

and hence

$$[5] \quad \frac{dS(x)}{dx} = -(A_s + C_s)\mathbb{B}_s e^{\mathbb{B}_s(W_s-W_q)}e^{-\mathbb{B}_s(x-W_q)} = -(A_s + C_s)\mathbb{B}_s e^{\mathbb{B}_s(W_s-W_q)}e^{-\mathbb{B}_s x_w}$$

Inserting [4] into [3] gives us

$$\begin{aligned}
 [6] \quad 1 - \chi_q &= \sum_{s=1}^q \left[\int_{W_s-W_q}^{W_{s-1}-W_q} (A_s + C_s)\mathbb{B}_s e^{\mathbb{B}_s(W_s-W_q)}e^{-\mathbb{B}_s x_w} x_w dx_w \right] \\
 &= \sum_{s=1}^q \left[(A_s + C_s)x_w e^{\mathbb{B}_s(W_s-W_q)}e^{-\mathbb{B}_s x_w} + \frac{(A_s + C_s)}{\mathbb{B}_s} e^{\mathbb{B}_s(W_s-W_q)}e^{-\mathbb{B}_s x_w} \right]_{W_s-W_q}^{W_{s-1}-W_q} \\
 &= \sum_{s=1}^q \left[(A_s + C_s)(W_s - W_q) + \frac{(A_s + C_s)}{\mathbb{B}_s} - (A_s + C_s)(W_{s-1} - W_q)e^{-\mathbb{B}_s(W_{s-1}-W_s)} - \frac{(A_s + C_s)}{\mathbb{B}_s} e^{-\mathbb{B}_s(W_{s-1}-W_s)} \right]
 \end{aligned}$$

From [2] we get

$$[7] \quad A_{s-1} + C_s = (A_s + C_s)e^{-B_s(W_{s-1}-W_s)}$$

and hence [6] can be simplified to

$$[8] \quad 1 - \chi_q = \sum_{s=1}^q \left[(A_s + C_s)(W_s - W_q) + \frac{(A_s + C_s)}{B_s} - (A_{s-1} + C_s)(W_{s-1} - W_q) - \frac{(A_{s-1} + C_s)}{B_s} \right] \\ = \sum_{s=1}^q \left[A_s(W_s - W_q) - A_{s-1}(W_{s-1} - W_q) - C_s(W_{s-1} - W_s) + \frac{(A_s - A_{s-1})}{B_s} \right]$$

The intermediate $A_s(W_s - W_q)$ terms in [8] will cancel out, as will the end terms since $A_0=0$ and $W_q=0$. Thus [8] can be simplified to

$$[9] \quad 1 - \chi_q = \sum_{s=1}^q \left[\frac{(A_s - A_{s-1})}{B_s} - C_s(W_{s-1} - W_s) \right]$$

where we assume that $A_0 = C_1=0$ and W_{s-1} is undefined. [9] can also be written as

$$[10] \quad 1 - \chi_q = \begin{cases} \frac{A_1}{B_1} & \text{for } q = 1 \\ \frac{A_q}{B_q} + \sum_{s=1}^{q-1} \left[\frac{A_s}{B_s} - \frac{A_s}{B_{s+1}} - (C_{s+1} - C_s)(W_s - W_q) \right] & \text{for } q > 1 \end{cases}$$

However, as *Figure 1* shows, that the population distribution for the smaller dipsets can tend towards a linear function in the later stages of fracture growth. In these cases, B_s tends to 0 and C_s tends to infinity. Thus neither [9] nor [10] will give correct results; however the appropriate terms of [9] can easily be modified to a linear form, to give

$$[11] \quad 1 - \chi_q = \sum_{s=1}^q \begin{cases} \left[\frac{1}{2} (A_s + A_{s-1})(W_{s-1} - W_s) \right] & \text{as } B_s \rightarrow 0, C_s \rightarrow \infty \\ \left[\frac{(A_s - A_{s-1})}{B_s} - C_s(W_{s-1} - W_s) \right] & \text{otherwise} \end{cases}$$

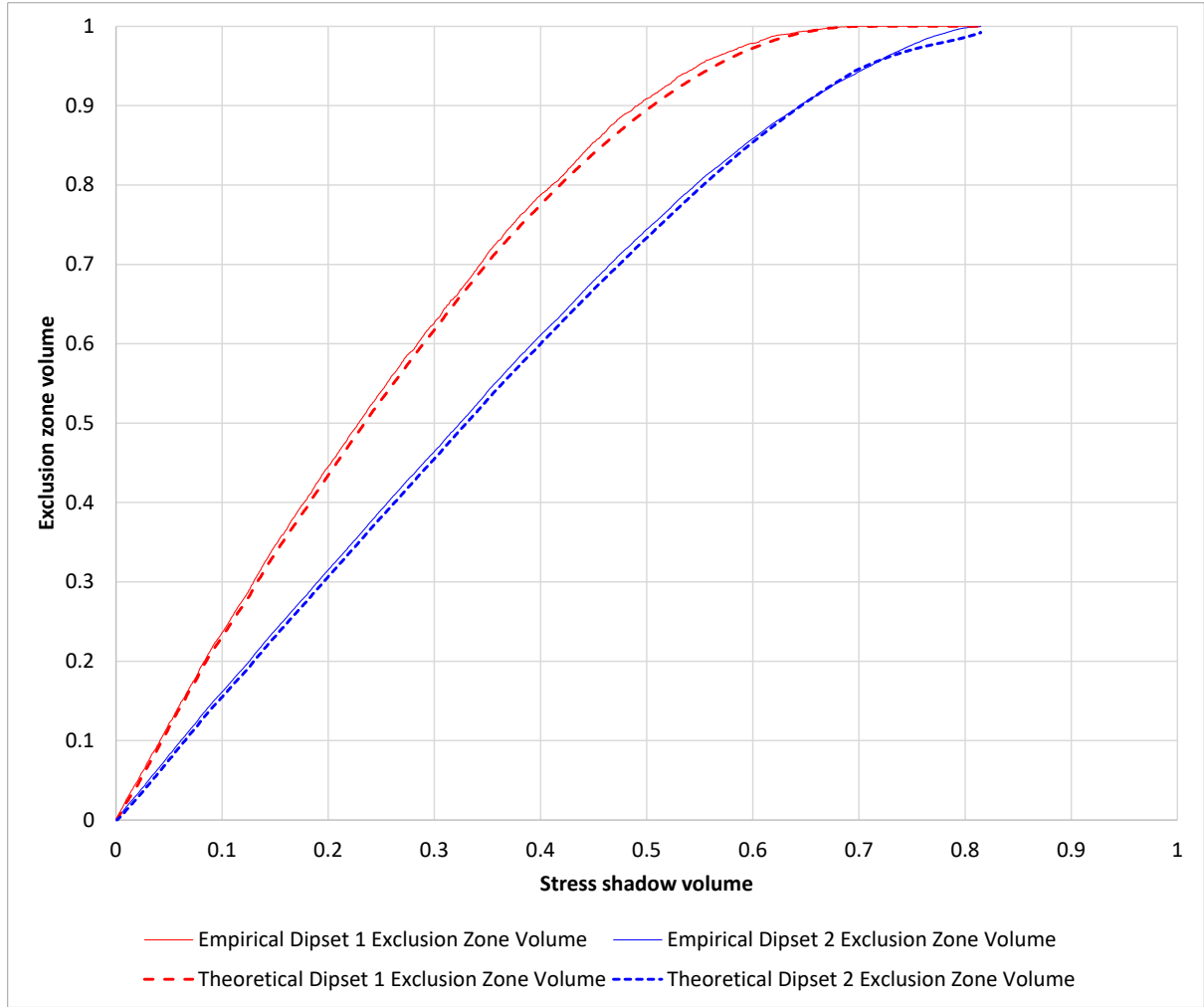


Figure 2: Total exclusion zone volume χ as a function of total stress shadow volume ψ for a fracture set with two dipsets, of stress shadow width $W_1=7$ and $W_2=3$, as shown in Figure 1. The exclusion zone volume for new macrofractures from dipset 1 (χ_1 , shown by the dashed red line), and the exclusion zone volume for new macrofractures from dipset 2 (χ_2 , shown by the dashed blue line) are both calculated using [9]. These are compared with empirical values for χ_1 (solid red line) and χ_2 (solid blue line), measured as new macrofractures from both dipsets were added one by one to a transect of length 10000 (see Figure 1).

2.1.3 Calculating the spacing distribution parameters A, B and C

Before we can use [9], we must derive expressions for the changes to $A_1, A_2, A_3, \dots, A_n, B_1$ and C_{n+1} when we add new macrofractures from any dipset q . We can do this using the same method as for the uniform stress shadow scenario: deriving two independent expressions for $dS(x)/d^q_{MF}P_{32}$, and equating the terms. Note that $^q_{MF}P_{32}$ is the total mean linear density of all macrofractures from dipset q , so $dS(x)/d^q_{MF}P_{32}$ represents the rate of change of $S(x)$ when we add macrofractures from dipset q . We can start by deriving an expression for $dS(x)/d^q_{MF}P_{32}$ that is valid for any dipset q when $x \geq W_1$, using the same procedure as previously:

$$[12] \quad \frac{dS(x)}{d^q_{MF}P_{32}} = \frac{\int_x^{x+W_q} \frac{dS(x_n)}{dx_n} (x_n - W_q) dx_n - \int_{x+W_q}^{\infty} \frac{dS(x_n)}{dx_n} (x_n - W_q) \left(1 - \frac{2x}{x_n - W_q}\right) dx_n}{\int_{W_q}^{\infty} -\frac{dS(x_n)}{dx_n} (x_n - W) dx_n}$$

$$= \frac{\int_{x-W_q}^x \frac{dS(x_w+W_q)}{dx_w} x_w dx_w - \int_x^\infty \frac{dS(x_w+W_q)}{dx_w} x_w dx_w - 2xS(x+W_q)}{\int_0^\infty -\frac{dS(x_w+W_q)}{dx_w} x_w dx_w}$$

where $x_w = x_n - W_q$. Using [4] and [9], we can write [12] as

$$\begin{aligned} [13] \quad \frac{dS(x)}{d_{\text{MFP}_{32}}^q} &= \frac{\int_{x-W_q}^x -A_1 \mathbb{B}_1 e^{\mathbb{B}_1(W_1-W_q)} e^{-\mathbb{B}_1 x_w} x_w dx_w - \int_x^\infty -A_1 \mathbb{B}_1 e^{\mathbb{B}_1(W_1-W_q)} e^{-\mathbb{B}_1 x_w} x_w dx_w - 2x A_1 e^{-\mathbb{B}_1 x}}{\sum_{s=1}^q \left[\int_{W_{s-1}-W_q}^{W_s-1-W_q} (A_s + C_s) \mathbb{B}_s e^{\mathbb{B}_s(W_s-W_q)} e^{-\mathbb{B}_s x_w} x_w dx_w \right]} \\ &= \frac{\left[A_1 x_w e^{\mathbb{B}_1(W_1-W_q)} e^{-\mathbb{B}_1 x_w} + \frac{A_1}{\mathbb{B}_1} e^{\mathbb{B}_1(W_1-W_q)} e^{-\mathbb{B}_1 x_w} \right]_{x-W_q}^x - \left[A_1 x_w e^{\mathbb{B}_1(W_1-W_q)} e^{-\mathbb{B}_1 x_w} + \frac{A_1}{\mathbb{B}_1} e^{\mathbb{B}_1(W_1-W_q)} e^{-\mathbb{B}_1 x_w} \right]_x^\infty - 2x A_1 e^{-\mathbb{B}_1 x}}{\sum_{s=1}^q \left[(A_s + C_s) x_w e^{\mathbb{B}_s(W_s-W_q)} e^{-\mathbb{B}_s x_w} + \frac{(A_s + C_s)}{\mathbb{B}_s} e^{\mathbb{B}_s(W_s-W_q)} e^{-\mathbb{B}_s x_w} \right]_{W_{s-1}-W_q}^{W_s-1-W_q}} \\ &= \frac{2 \frac{A_1}{\mathbb{B}_1} e^{-\mathbb{B}_1 W_q} e^{-\mathbb{B}_1(x-W_1)} - A_1(x-W_q) e^{-\mathbb{B}_1(x-W_1)} - \frac{A_1}{\mathbb{B}_1} e^{-\mathbb{B}_1(x-W_1)}}{\frac{A_1}{\mathbb{B}_1} + \sum_{s=2}^q \left[\frac{(A_s - A_{s-1})}{\mathbb{B}_s} - C_s(W_{s-1} - W_s) \right]} \end{aligned}$$

Differentiating [1] directly with respect to $q_{\text{MFP}_{32}}$, when $x \geq W_1$, gives

$$\begin{aligned} [14] \quad \frac{dS(x)}{d_{\text{MFP}_{32}}^q} &= \frac{dA_1}{d_{\text{MFP}_{32}}^q} e^{-\mathbb{B}_1(x-W_1)} - A_1(x-W_1) \frac{d\mathbb{B}_1}{d_{\text{MFP}_{32}}^q} e^{-\mathbb{B}_1(x-W_1)} \\ &= \frac{dA_1}{d_{\text{MFP}_{32}}^q} e^{-\mathbb{B}_1(x-W_1)} + A_1(W_1 - W_q) \frac{d\mathbb{B}_1}{d_{\text{MFP}_{32}}^q} e^{-\mathbb{B}_1(x-W_1)} - A_1(x-W_q) \frac{d\mathbb{B}_1}{d_{\text{MFP}_{32}}^q} e^{-\mathbb{B}_1(x-W_1)} \end{aligned}$$

First, we can equate the $(x-W_q)e^{-\mathbb{B}_1(x-W_q)}$ terms to get

$$\begin{aligned} [15] \quad -A_1(x-W_q) \frac{d\mathbb{B}_1}{d_{\text{MFP}_{32}}^q} e^{-\mathbb{B}_1(x-W_1)} &= \frac{-A_1(x-W_q) e^{-\mathbb{B}_1(x-W_1)}}{\frac{A_1}{\mathbb{B}_1} + \sum_{s=2}^q \left[\frac{(A_s - A_{s-1})}{\mathbb{B}_s} - C_s(W_{s-1} - W_s) \right]} \\ \Rightarrow \frac{d\mathbb{B}_1}{d_{\text{MFP}_{32}}^q} &= \frac{1}{\frac{A_1}{\mathbb{B}_1} + \sum_{s=2}^q \left[\frac{(A_s - A_{s-1})}{\mathbb{B}_s} - C_s(W_{s-1} - W_s) \right]} \end{aligned}$$

Next, we can equate the $e^{-\mathbb{B}_1(x-W_q)}$ terms to get

$$\begin{aligned} [16] \quad \frac{dA_1}{d_{\text{MFP}_{32}}^q} e^{-\mathbb{B}_1(x-W_1)} + A_1(W_1 - W_q) \frac{d\mathbb{B}_1}{d_{\text{MFP}_{32}}^q} e^{-\mathbb{B}_1(x-W_1)} &= \frac{\frac{A_1}{\mathbb{B}_1} (2e^{-\mathbb{B}_1 W_q} - 1) e^{-\mathbb{B}_1(x-W_1)}}{\frac{A_1}{\mathbb{B}_1} + \sum_{s=2}^q \left[\frac{(A_s - A_{s-1})}{\mathbb{B}_s} - C_s(W_{s-1} - W_s) \right]} \\ \Rightarrow \frac{dA_1}{d_{\text{MFP}_{32}}^q} &= \frac{\frac{A_1}{\mathbb{B}_1} (2e^{-\mathbb{B}_1 W_q} - 1) - A_1(W_1 - W_q)}{\frac{A_1}{\mathbb{B}_1} + \sum_{s=2}^q \left[\frac{(A_s - A_{s-1})}{\mathbb{B}_s} - C_s(W_{s-1} - W_s) \right]} \end{aligned}$$

We can also derive an expression for $dS(W_r)/d_{\text{MFP}_{32}}^q$ that is valid for any two dipsets q and r . If $r \leq q$ (i.e. $W_r \geq W_q$) then

$$\begin{aligned} [17] \quad \frac{dS(W_r)}{d_{\text{MFP}_{32}}^q} &= \frac{\int_{W_r}^{W_r+W_q} \frac{dS(x_n)}{dx_n} (x_n - W_q) dx_n - \int_{W_r+W_q}^\infty \frac{dS(x_n)}{dx_n} (x_n - W_q) \left(1 - \frac{2W_r}{x_n - W_q}\right) dx_n}{\int_{W_q}^\infty -\frac{dS(x_n)}{dx_n} (x_n - W) dx_n} \\ &= \frac{1}{1-\chi_q} \left(\sum_{r=R+1}^r \left[\int_{W_{s-1}-W_q}^{W_s-1-W_q} \frac{dS(x_w+W_q)}{dx_w} x_w dx_w \right] + \int_{W_R-W_q}^{W_r} \frac{dS(x_w+W_q)}{dx_w} x_w dx_w \right. \\ &\quad \left. - \int_{W_r}^{W_{R-1}-W_q} \frac{dS(x_w+W_q)}{dx_w} x_w dx_w - \sum_{s=1}^{R-1} \left[\int_{W_{s-1}-W_q}^{W_s-1-W_q} \frac{dS(x_w+W_q)}{dx_w} x_w dx_w \right] - 2W_r S(W_r + W_q) \right) \end{aligned}$$

$$\begin{aligned}
&= \frac{1}{1-\chi_q} \left(\begin{aligned} &\sum_{s=R+1}^r \left[\int_{W_s-W_q}^{W_{s-1}-W_q} -(\mathbb{A}_s + \mathbb{C}_s) \mathbb{B}_s e^{\mathbb{B}_s(W_s-W_q)} e^{-\mathbb{B}_s x_w} x_w dx_w \right] \\ &+ \int_{W_R-W_q}^{W_r} -(\mathbb{A}_R + \mathbb{C}_R) \mathbb{B}_R e^{\mathbb{B}_R(W_R-W_q)} e^{-\mathbb{B}_R x_w} x_w dx_w \\ &- \int_{W_r}^{W_{R-1}-W_q} -(\mathbb{A}_R + \mathbb{C}_R) \mathbb{B}_R e^{\mathbb{B}_R(W_R-W_q)} e^{-\mathbb{B}_R x_w} x_w dx_w \\ &- \sum_{s=1}^{R-1} \left[\int_{W_s-W_q}^{W_{s-1}-W_q} -(\mathbb{A}_s + \mathbb{C}_s) \mathbb{B}_s e^{\mathbb{B}_s(W_s-W_q)} e^{-\mathbb{B}_s x_w} x_w dx_w \right] - 2W_r \left((\mathbb{A}_R + \mathbb{C}_R) e^{\mathbb{B}_R(W_R-W_q)} e^{-\mathbb{B}_R W_r} - \mathbb{C}_R \right) \end{aligned} \right) \\
&= \frac{1}{1-\chi_q} \left(\begin{aligned} &\sum_{s=R+1}^r \left[(\mathbb{A}_s + \mathbb{C}_s) x_w e^{\mathbb{B}_s(W_s-W_q)} e^{-\mathbb{B}_s x_w} + \frac{(\mathbb{A}_s + \mathbb{C}_s)}{\mathbb{B}_s} e^{\mathbb{B}_s(W_s-W_q)} e^{-\mathbb{B}_s x_w} \right]_{W_s-W_q}^{W_{s-1}-W_q} \\ &+ \left[(\mathbb{A}_R + \mathbb{C}_R) x_w e^{\mathbb{B}_R(W_R-W_q)} e^{-\mathbb{B}_R x_w} + \frac{(\mathbb{A}_R + \mathbb{C}_R)}{\mathbb{B}_R} e^{\mathbb{B}_R(W_R-W_q)} e^{-\mathbb{B}_R x_w} \right]_{W_R-W_q}^{W_r} \\ &- \left[(\mathbb{A}_R + \mathbb{C}_R) x_w e^{\mathbb{B}_R(W_R-W_q)} e^{-\mathbb{B}_R x_w} + \frac{(\mathbb{A}_R + \mathbb{C}_R)}{\mathbb{B}_R} e^{\mathbb{B}_R(W_R-W_q)} e^{-\mathbb{B}_R x_w} \right]_{W_r}^{W_{R-1}-W_q} \\ &- \sum_{s=1}^{R-1} \left[(\mathbb{A}_s + \mathbb{C}_s) x_w e^{\mathbb{B}_s(W_s-W_q)} e^{-\mathbb{B}_s x_w} + \frac{(\mathbb{A}_s + \mathbb{C}_s)}{\mathbb{B}_s} e^{\mathbb{B}_s(W_s-W_q)} e^{-\mathbb{B}_s x_w} \right]_{W_s-W_q}^{W_{s-1}-W_q} \\ &- 2W_r (\mathbb{A}_R + \mathbb{C}_R) e^{\mathbb{B}_R(W_R-W_q)} e^{-\mathbb{B}_R W_r} + 2W_r \mathbb{C}_R \end{aligned} \right) \\
&= \frac{1}{1-\chi_q} \left(\begin{aligned} &-\mathbb{A}_r (W_r - W_q) - \sum_{s=R+1}^r \left[\frac{(\mathbb{A}_s - \mathbb{A}_{s-1})}{\mathbb{B}_s} - \mathbb{C}_s (W_{s-1} - W_s) \right] \\ &- \frac{(\mathbb{A}_R + \mathbb{C}_R)}{\mathbb{B}_R} (1 - e^{-\mathbb{B}_R(W_q + W_r - W_R)}) + \frac{(\mathbb{A}_{R-1} + \mathbb{C}_R)}{\mathbb{B}_R} (e^{-\mathbb{B}_R(W_q + W_r - W_{R-1})} - 1) \\ &+ \mathbb{C}_R (W_q + W_r - W_R) + \mathbb{C}_R (W_q + W_r - W_{R-1}) \\ &+ \sum_{s=2}^{R-1} \left[\frac{(\mathbb{A}_s - \mathbb{A}_{s-1})}{\mathbb{B}_s} - \mathbb{C}_s (W_{s-1} - W_s) \right] + \frac{\mathbb{A}_1}{\mathbb{B}_1} \end{aligned} \right) \\
&= \frac{1}{1-\chi_q} \left(\begin{aligned} &\sum_{s=1}^{R-1} \left[\frac{(\mathbb{A}_s - \mathbb{A}_{s-1})}{\mathbb{B}_s} - \mathbb{C}_s (W_{s-1} - W_s) \right] - \sum_{s=R+1}^r \left[\frac{(\mathbb{A}_s - \mathbb{A}_{s-1})}{\mathbb{B}_s} - \mathbb{C}_s (W_{s-1} - W_s) \right] - \mathbb{A}_r (W_r - W_q) \\ &+ 2 \frac{(\mathbb{A}_R + \mathbb{C}_R)}{\mathbb{B}_R} e^{-\mathbb{B}_R(W_q + W_r - W_R)} + 2\mathbb{C}_R (W_q + W_r) - \frac{\mathbb{A}_R + \mathbb{A}_{R-1} + 2\mathbb{C}_R}{\mathbb{B}_R} - \mathbb{C}_R (W_{R-1} + W_R) \end{aligned} \right)
\end{aligned}$$

where R is the fracture dipset for which $W_R \leq W_r + W_q < W_{R-1}$. Note that R may be the same as r, in which case the term in [17] containing the sum from $s=R$ to $r-1$ will be redundant. As $\mathbb{B}_s \rightarrow 0$ and $\mathbb{C}_s \rightarrow \infty$, we can use the linear forms for the terms of the two sums as given in [11], but when $\mathbb{B}_R \rightarrow 0$ and $\mathbb{C}_R \rightarrow \infty$, we must replace the R terms in [17] with

$$\begin{aligned}
[18] \quad &\frac{dS(W_r)}{d_{\text{MF}}^{\text{qP}}}_{32} = \\
&\frac{1}{1-\chi_q} \left(\begin{aligned} &\sum_{s=1}^{R-1} \left[\frac{(\mathbb{A}_s - \mathbb{A}_{s-1})}{\mathbb{B}_s} - \mathbb{C}_s (W_{s-1} - W_s) \right] - \sum_{s=R+1}^r \left[\frac{1}{2} (\mathbb{A}_s + \mathbb{A}_{s-1}) (W_{s-1} - W_s) \right] - \mathbb{A}_r (W_r - W_q) \\ &+ \frac{\mathbb{A}_R W_R - \mathbb{A}_{R-1} W_{R-1}}{W_{R-1} - W_R} (W_{R-1} + W_R - 2(W_r + W_q)) - \frac{1}{2} \frac{\mathbb{A}_R - \mathbb{A}_{R-1}}{W_{R-1} - W_R} (W_{R-1}^2 + W_R^2 - 2(W_r + W_q)^2) \end{aligned} \right)
\end{aligned}$$

(assuming $\mathbb{B}_s \rightarrow 0$ and $\mathbb{C}_s \rightarrow \infty$ for $s > R$).

If $r > q$ (i.e. $W_r < W_q$) then we get a slightly different form:

$$\begin{aligned}
[19] \quad &\frac{dS(W_r)}{d_{\text{MF}}^{\text{qP}}}_{32} = \frac{\int_{W_q}^{W_r+W_q} \frac{dS(x_n)}{dx_n} (x_n - W_q) dx_n - \int_{W_r+W_q}^{\infty} \frac{dS(x_n)}{dx_n} (x_n - W_q) \left(1 - \frac{2W_r}{x_n - W_q} \right) dx_n}{\int_{W_q}^{\infty} \frac{dS(x_n)}{dx_n} (x_n - W) dx_n} \\
&= \frac{1}{1-\chi_q} \left(\begin{aligned} &\sum_{s=R+1}^q \left[\int_{W_s-W_q}^{W_{s-1}-W_q} \frac{dS(x_w+W_q)}{dx_w} x_w dx_w \right] + \int_{W_R-W_q}^{W_r} \frac{dS(x_w+W_q)}{dx_w} x_w dx_w \\ &- \int_{W_r}^{W_{R-1}-W_q} \frac{dS(x_w+W_q)}{dx_w} x_w dx_w - \sum_{s=1}^{R-1} \left[\int_{W_s-W_q}^{W_{s-1}-W_q} \frac{dS(x_w+W_q)}{dx_w} x_w dx_w \right] - 2W_r S(W_r + W_q) \end{aligned} \right)
\end{aligned}$$

$$\begin{aligned}
&= \frac{1}{1-\chi_q} \left(\begin{aligned} &\sum_{s=R+1}^q \left[\int_{W_s-W_q}^{W_{s-1}-W_q} -(\mathbb{A}_s + \mathbb{C}_s) \mathbb{B}_s e^{\mathbb{B}_s(W_s-W_q)} e^{-\mathbb{B}_s x_w} x_w dx_w \right] \\ &+ \int_{W_R-W_q}^{W_r} -(\mathbb{A}_R + \mathbb{C}_R) \mathbb{B}_R e^{\mathbb{B}_R(W_R-W_q)} e^{-\mathbb{B}_s x_w} x_w dx_w \\ &- \int_{W_r}^{W_{R-1}-W_q} -(\mathbb{A}_R + \mathbb{C}_R) \mathbb{B}_R e^{\mathbb{B}_R(W_R-W_q)} e^{-\mathbb{B}_s x_w} x_w dx_w \\ &- \sum_{s=1}^{R-1} \left[\int_{W_s-W_q}^{W_{s-1}-W_q} -(\mathbb{A}_s + \mathbb{C}_s) \mathbb{B}_s e^{\mathbb{B}_s(W_s-W_q)} e^{-\mathbb{B}_s x_w} x_w dx_w \right] - 2W_r \left((\mathbb{A}_R + \mathbb{C}_R) e^{\mathbb{B}_R(W_R-W_q)} e^{-\mathbb{B}_1 W_r} - \mathbb{C}_R \right) \end{aligned} \right) \\
&= \frac{1}{1-\chi_q} \left(\begin{aligned} &\sum_{s=R+1}^q \left[(\mathbb{A}_s + \mathbb{C}_s) x_w e^{\mathbb{B}_s(W_s-W_q)} e^{-\mathbb{B}_s x_w} + \frac{(\mathbb{A}_s + \mathbb{C}_s)}{\mathbb{B}_s} e^{\mathbb{B}_s(W_s-W_q)} e^{-\mathbb{B}_s x_w} \right]_{W_s-W_q}^{W_{s-1}-W_q} \\ &+ \left[(\mathbb{A}_s + \mathbb{C}_s) x_w e^{\mathbb{B}_s(W_s-W_q)} e^{-\mathbb{B}_s x_w} + \frac{(\mathbb{A}_s + \mathbb{C}_s)}{\mathbb{B}_s} e^{\mathbb{B}_s(W_s-W_q)} e^{-\mathbb{B}_s x_w} \right]_{W_R-W_q}^{W_r} \\ &- \left[(\mathbb{A}_R + \mathbb{C}_R) x_w e^{\mathbb{B}_R(W_R-W_q)} e^{-\mathbb{B}_R x_w} + \frac{(\mathbb{A}_R + \mathbb{C}_R)}{\mathbb{B}_R} e^{\mathbb{B}_R(W_R-W_q)} e^{-\mathbb{B}_R x_w} \right]_{W_r}^{W_{R-1}-W_q} \\ &- \sum_{s=1}^{R-1} \left[(\mathbb{A}_s + \mathbb{C}_s) x_w e^{\mathbb{B}_s(W_s-W_q)} e^{-\mathbb{B}_s x_w} + \frac{(\mathbb{A}_s + \mathbb{C}_s)}{\mathbb{B}_s} e^{\mathbb{B}_s(W_s-W_q)} e^{-\mathbb{B}_s x_w} \right]_{W_s-W_q}^{W_{s-1}-W_q} \\ &- 2W_r (\mathbb{A}_R + \mathbb{C}_R) e^{\mathbb{B}_R(W_R-W_q)} e^{-\mathbb{B}_R W_r} + 2W_r \mathbb{C}_R \end{aligned} \right) \\
&= \frac{1}{1-\chi_q} \left(\begin{aligned} &- \sum_{s=R+1}^q \left[\frac{(\mathbb{A}_s - \mathbb{A}_{s-1})}{\mathbb{B}_s} - \mathbb{C}_s (W_{s-1} - W_s) \right] \\ &- \frac{(\mathbb{A}_R + \mathbb{C}_R)}{\mathbb{B}_R} (1 - e^{-\mathbb{B}_R(W_q + W_r - W_R)}) + \frac{(\mathbb{A}_{R-1} + \mathbb{C}_R)}{\mathbb{B}_R} (e^{-\mathbb{B}_R(W_q + W_r - W_{R-1})} - 1) \\ &+ \mathbb{C}_R (W_q + W_r - W_R) + \mathbb{C}_R (W_q + W_r - W_{R-1}) \\ &+ \sum_{s=2}^{R-1} \left[\frac{(\mathbb{A}_s - \mathbb{A}_{s-1})}{\mathbb{B}_s} - \mathbb{C}_s (W_{s-1} - W_s) \right] + \frac{\mathbb{A}_1}{\mathbb{B}_1} \end{aligned} \right) \\
&= \frac{1}{1-\chi_q} \left(\begin{aligned} &\sum_{s=1}^{R-1} \left[\frac{(\mathbb{A}_s - \mathbb{A}_{s-1})}{\mathbb{B}_s} - \mathbb{C}_s (W_{s-1} - W_s) \right] - \sum_{s=R+1}^q \left[\frac{(\mathbb{A}_s - \mathbb{A}_{s-1})}{\mathbb{B}_s} - \mathbb{C}_s (W_{s-1} - W_s) \right] \\ &+ 2 \frac{(\mathbb{A}_R + \mathbb{C}_R)}{\mathbb{B}_R} e^{-\mathbb{B}_R(W_q + W_r - W_R)} + 2 \mathbb{C}_R (W_q + W_r) - \frac{\mathbb{A}_R + \mathbb{A}_{R-1} + 2\mathbb{C}_R}{\mathbb{B}_R} - \mathbb{C}_R (W_{R-1} + W_R) \end{aligned} \right)
\end{aligned}$$

Again, as $\mathbb{B}_s \rightarrow 0$ and $\mathbb{C}_s \rightarrow \infty$ we can use the linear forms given in [11] for the terms of the two sums in [19], but when $\mathbb{B}_R \rightarrow 0$ and $\mathbb{C}_R \rightarrow \infty$, we must replace the R terms with

$$\begin{aligned}
[20] \quad \frac{dS(W_r)}{d_{\text{MFP}_{32}}^q} &= \\
\frac{1}{1-\chi_q} \left(\begin{aligned} &\sum_{s=1}^{R-1} \left[\frac{(\mathbb{A}_s - \mathbb{A}_{s-1})}{\mathbb{B}_s} - \mathbb{C}_s (W_{s-1} - W_s) \right] - \sum_{s=R+1}^q \left[\frac{1}{2} (\mathbb{A}_s + \mathbb{A}_{s-1}) (W_{s-1} - W_s) \right] \\ &+ \frac{\mathbb{A}_R W_R - \mathbb{A}_{R-1} W_{R-1}}{W_{R-1} - W_R} (W_{R-1} + W_R - 2(W_r + W_q)) - \frac{1}{2} \frac{\mathbb{A}_R - \mathbb{A}_{R-1}}{W_{R-1} - W_R} (W_{R-1}^2 + W_R^2 - 2(W_r + W_q)^2) \end{aligned} \right)
\end{aligned}$$

Finally, we can calculate \mathbb{C}_{n+1} from the total stress shadow volume ψ iteratively. To do this, we will assume that the distribution of the offset values is based on the distribution of macrofractures between the dipsets, so that for any dipset s ,

$$[21] \quad \widehat{\mathbb{C}}_s = \frac{\mathbb{C}_s}{\mathbb{C}_{n+1}} = \sum_{u=1}^{u=q-1} \left[\frac{\mathbb{M}_{\text{FP}_{32}}^u}{\mathbb{M}_{\text{FP}_{32}}} \right]$$

By substituting [2] and [21] into [9], we can get an expression $1-\chi_{n+1}$, that excludes any of the unknown \mathbb{B}_s factors (i.e. where $s>1$):

$$[22] \quad 1 - \chi_{n+1} = \frac{\mathbb{A}_1}{\mathbb{B}_1} + \sum_{s=2}^{n+1} \left[\left(\frac{(\mathbb{A}_s - \mathbb{A}_{s-1})}{\ln(\mathbb{A}_s + \widehat{\mathbb{C}}_s \mathbb{C}_{n+1}) - \ln(\mathbb{A}_{s-1} + \widehat{\mathbb{C}}_s \mathbb{C}_{n+1})} - \widehat{\mathbb{C}}_s \mathbb{C}_{n+1} \right) (W_{s-1} - W_s) \right]$$

This represents the “clear zone volume for new fractures with stress shadow width zero”, and thus should be equal to the inverse stress shadow volume $1-\psi$. We can therefore solve [22] using a Newton-Raphson iteration. The differential of [22] is given by

$$\begin{aligned}
[23] \quad \frac{d(1-\chi_{n+1})}{d\mathbb{C}_{n+1}} &= \sum_{s=2}^{n+1} \left[\left(-\frac{(\mathbb{A}_s - \mathbb{A}_{s-1}) \left(\frac{\widehat{\mathbb{C}}_s}{\mathbb{A}_s + \mathbb{C}_s \mathbb{C}_{n+1}} - \frac{\widehat{\mathbb{C}}_s}{\mathbb{A}_{s-1} + \mathbb{C}_s \mathbb{C}_{n+1}} \right)}{(\ln(\mathbb{A}_s + \widehat{\mathbb{C}}_s \mathbb{C}_{n+1}) - \ln(\mathbb{A}_{s-1} + \widehat{\mathbb{C}}_s \mathbb{C}_{n+1}))^2} - \widehat{\mathbb{C}}_s \right) (W_{s-1} - W_s) \right] \\
&= \sum_{s=2}^{n+1} \left[\widehat{\mathbb{C}}_s \left(\frac{(\mathbb{A}_s - \mathbb{A}_{s-1})^2}{(\mathbb{A}_s + \mathbb{C}_s)(\mathbb{A}_{s-1} + \mathbb{C}_s)(\ln(\mathbb{A}_s + \mathbb{C}_s) - \ln(\mathbb{A}_{s-1} + \mathbb{C}_s))^2} - 1 \right) (W_{s-1} - W_s) \right] \\
&= \sum_{s=2}^{n+1} \left[\widehat{\mathbb{C}}_s \left(\frac{(\mathbb{A}_s - \mathbb{A}_{s-1})^2}{(\mathbb{A}_s + \mathbb{C}_s)(\mathbb{A}_{s-1} + \mathbb{C}_s)(W_{s-1} - W_s) \mathbb{B}_s^2} - (W_{s-1} - W_s) \right) \right]
\end{aligned}$$

so we can iterate with

$$[24] \quad \mathbb{C}_{n+1} = \mathbb{C}_{n+1} - \frac{\frac{\mathbb{A}_1}{\mathbb{B}_1} + \sum_{s=2}^{n+1} \left[\frac{(\mathbb{A}_s - \mathbb{A}_{s-1})}{\mathbb{B}_s} - \mathbb{C}_s (W_{s-1} - W_s) \right] - (1 - \psi)}{\sum_{s=2}^{n+1} \left[\widehat{\mathbb{C}}_s \left(\frac{(\mathbb{A}_s - \mathbb{A}_{s-1})^2}{(\mathbb{A}_s + \mathbb{C}_s)(\mathbb{A}_{s-1} + \mathbb{C}_s)(W_{s-1} - W_s) \mathbb{B}_s^2} - (W_{s-1} - W_s) \right) \right]}$$

recalculating \mathbb{B}_s and \mathbb{C}_s between iterations for all dipsets $1 < s \leq n+1$, using [21] and [2].

We can therefore calculate all the parameters we need to describe the cumulative macrofracture spacing distribution function $S(x)$ numerically, as we add macrofractures from any dipset q , using variants of [2], [15], [16], [17], [19] and [21]:

$$[25] \quad \frac{d\mathbb{A}_1}{d_{\text{MFP}_{32}}^q} = \frac{\frac{\mathbb{A}_1}{\mathbb{B}_1} (2e^{-\mathbb{B}_1 W_q} - 1) - \mathbb{A}_1 (W_1 - W_q)}{\frac{\mathbb{A}_1}{\mathbb{B}_1} + \sum_{s=2}^q \left[\frac{(\mathbb{A}_s - \mathbb{A}_{s-1})}{\mathbb{B}_s} - \mathbb{C}_s (W_{s-1} - W_s) \right]} \quad \text{for } r = 1$$

$$[26] \quad \frac{d\mathbb{A}_r}{d_{\text{MFP}_{32}}^q} = \frac{\left(\sum_{s=1}^{R-1} \left[\frac{(\mathbb{A}_s - \mathbb{A}_{s-1})}{\mathbb{B}_s} - \mathbb{C}_s (W_{s-1} - W_s) \right] - \sum_{s=R+1}^r \left[\frac{(\mathbb{A}_s - \mathbb{A}_{s-1})}{\mathbb{B}_s} - \mathbb{C}_s (W_{s-1} - W_s) \right] - \mathbb{A}_r (W_r - W_q) \right) + 2 \frac{(\mathbb{A}_R + \mathbb{C}_R)}{\mathbb{B}_R} e^{-\mathbb{B}_R (W_q + W_r - W_R)} + 2 \mathbb{C}_R (W_q + W_r) - \frac{\mathbb{A}_R + \mathbb{A}_{R-1} + 2\mathbb{C}_R}{\mathbb{B}_R} - \mathbb{C}_R (W_{R-1} + W_R)}{\frac{\mathbb{A}_1}{\mathbb{B}_1} + \sum_{s=2}^q \left[\frac{(\mathbb{A}_s - \mathbb{A}_{s-1})}{\mathbb{B}_s} - \mathbb{C}_s (W_{s-1} - W_s) \right]} \quad \text{for } 1 < r \leq q$$

$$[27] \quad \frac{d\mathbb{A}_r}{d_{\text{MFP}_{32}}^q} = \frac{\left(\sum_{s=1}^{R-1} \left[\frac{(\mathbb{A}_s - \mathbb{A}_{s-1})}{\mathbb{B}_s} - \mathbb{C}_s (W_{s-1} - W_s) \right] - \sum_{s=R+1}^q \left[\frac{(\mathbb{A}_s - \mathbb{A}_{s-1})}{\mathbb{B}_s} - \mathbb{C}_s (W_{s-1} - W_s) \right] \right) + 2 \frac{(\mathbb{A}_R + \mathbb{C}_R)}{\mathbb{B}_R} e^{-\mathbb{B}_R (W_q + W_r - W_R)} + 2 \mathbb{C}_R (W_q + W_r) - \frac{\mathbb{A}_R + \mathbb{A}_{R-1} + 2\mathbb{C}_R}{\mathbb{B}_R} - \mathbb{C}_R (W_{R-1} + W_R)}{\frac{\mathbb{A}_1}{\mathbb{B}_1} + \sum_{s=2}^q \left[\frac{(\mathbb{A}_s - \mathbb{A}_{s-1})}{\mathbb{B}_s} - \mathbb{C}_s (W_{s-1} - W_s) \right]} \quad \text{for } r > q$$

$$[28] \quad \frac{d\mathbb{B}_1}{d_{\text{MFP}_{32}}^q} = \frac{1}{\frac{\mathbb{A}_1}{\mathbb{B}_1} + \sum_{s=2}^q \left[\frac{(\mathbb{A}_s - \mathbb{A}_{s-1})}{\mathbb{B}_s} - \mathbb{C}_s (W_{s-1} - W_s) \right]} \quad \text{for } r = 1$$

$$[29] \quad \frac{d\mathbb{B}_r}{d_{\text{MFP}_{32}}^q} \approx \frac{\left(\frac{d\mathbb{A}_r}{d_{\text{MFP}_{32}}^q} + \frac{d\mathbb{C}_r}{d_{\text{MFP}_{32}}^q} \right) \frac{1}{\mathbb{A}_r + \mathbb{C}_r} - \left(\frac{d\mathbb{A}_{r-1}}{d_{\text{MFP}_{32}}^q} + \frac{d\mathbb{C}_r}{d_{\text{MFP}_{32}}^q} \right) \frac{1}{\mathbb{A}_{r-1} + \mathbb{C}_r}}{W_{r-1} - W_r} \quad \text{for } r > 1$$

Note that in [29], we can usually ignore the change in \mathbb{C}_r when we add additional macrofractures, as this is likely to be small compared with the change in \mathbb{A}_r and \mathbb{A}_{r-1} . The evolution of the cumulative macrofracture spacing distribution function $S(x)$ through time, when adding macrofractures from two dipsets with different stress shadow widths, is illustrated in *Figure 1*.

Finally we can differentiate [9] to obtain an expression for the rate of change of the exclusion zone volume as new macrofractures are added to the transect:

$$\begin{aligned}
[30] \quad \frac{d\chi_r}{d_{\text{MFP}_{32}}^q} &= -\frac{d}{d_{\text{MFP}_{32}}^q} \left(\sum_{s=1}^q \left[\frac{(\mathbb{A}_s - \mathbb{A}_{s-1})}{\mathbb{B}_s} - \mathbb{C}_s (W_{s-1} - W_s) \right] \right) \\
&= \frac{\mathbb{A}_1}{\mathbb{B}_1^2} \frac{d\mathbb{B}_1}{d_{\text{MFP}_{32}}^q} - \mathbb{B}_1 \frac{d\mathbb{A}_1}{d_{\text{MFP}_{32}}^q} + \sum_{s=2}^q \left[\frac{(\mathbb{A}_s - \mathbb{A}_{s-1})}{\mathbb{B}_s^2} \frac{d\mathbb{B}_s}{d_{\text{MFP}_{32}}^q} - \mathbb{B}_s \left(\frac{d\mathbb{A}_s}{d_{\text{MFP}_{32}}^q} \frac{d\mathbb{A}_{s-1}}{d_{\text{MFP}_{32}}^q} \right) \right]
\end{aligned}$$

$$\approx \frac{\frac{A_1}{B_1} \left(\frac{2}{B_1} (1 - e^{-B_1 W_q}) + (W_1 - W_q) \right)}{\frac{A_1}{B_1} + \sum_{s=2}^q \left[\frac{(A_s - A_{s-1})}{B_s} - C_s (W_{s-1} - W_s) \right]} + \sum_{s=2}^q \left[\left(\frac{A_s - A_{s-1}}{B_s^2} \right) \frac{dB_s}{d_{MF}^q P_{32}} - \frac{1}{B_s} \left(\frac{dA_s}{d_{MF}^q P_{32}} - \frac{dA_{s-1}}{d_{MF}^q P_{32}} \right) \right]$$

ignoring the changes in C values when we add additional macrofractures. The values of the differentials $dA_s/d_{MF}^q P_{32}$ and $dB_s/d_{MF}^q P_{32}$ can be obtained from [25], [26], [27] and [29] respectively.

2.1.4 Proximity zone volume

Calculating the total volume of a set of macrofracture proximity zones of arbitrary width X_p is more complex when the macrofractures have variable stress shadow widths, as the proportion of the proximity zone that lies within the stress shadow will vary between different macrofractures. Furthermore, the minimum distance between adjacent macrofractures will vary, depending on the stress shadow width of both macrofractures. To account fully for this, we would need to determine the probability of each potential combination of macrofracture stress shadow widths, and then calculate the probability that the proximity zone will exceed the macrofracture spacing in each case.

However we can get an approximate result much more easily by taking the average macrofracture stress shadow width W_{av} , which can be obtained by dividing the total macrofracture stress shadow volume ψ by the total mean linear macrofracture density $_{MF}P_{32}$. This will also give the average minimum distance between two adjacent macrofractures. Using this approximation, we can assume that:

- If the macrofracture proximity zone width X_p is less than W_{av} , the total proximity zone volume is given by $_{MF}P_{32}X_p$ (although this will not always be accurate, as some proximity zone overlap may occur if two adjacent macrofractures both have narrow stress shadows).
- If the proximity zone width X_p is greater than W_{av} , we can insert W_{av} into Equation B.40 from Welch et al. 2020 to calculate the total volume not lying within the proximity zone, $1 - \chi'(X_p)$:

$$\begin{aligned} [31] 1 - \chi'(X_p) &\approx \int_{X_p - W_{av}}^{\infty} -\frac{dS(x_n)}{dx_n} (x_n - (X_p - W_{av})) dx_n \\ &= \int_0^{\infty} -\frac{dS(x_w + (X_p - W_{av}))}{dx_w} x_w dx_w \\ &= \int_0^{W_{R-1} - (X_p - W_{av})} -\frac{dS(x_w + (X_p - W_{av}))}{dx_w} x_w dx_w + \sum_{s=1}^{R-1} \left[\int_{W_s - (X_p - W_{av})}^{W_{s+1} - (X_p - W_{av})} -\frac{dS(x_w + (X_p - W_{av}))}{dx_w} x_w dx_w \right] \end{aligned}$$

where $x_w = x_n - (X_p - W_{av})$, and where R is the fracture dipset for which $W_R \leq X_p - W_{av} < W_{R+1}$. We can now solve [31] in the same way that [3] was solved to calculate the clear zone volume $1 - \chi_q$. First we note that, for $W_s \leq x \leq W_{s+1}$,

$$\begin{aligned} [32] S(x) &= (A_s + C_s) e^{-B_s(x - W_s)} - C_s \\ &= (A_s + C_s) e^{-B_s((X_p - W_{av}) - W_s)} e^{-B_s(x - (X_p - W_{av}))} - C_s \end{aligned}$$

Inserting [32] into [31] gives us

$$\begin{aligned} [33] 1 - \chi'(X_p) &\approx \int_0^{W_{R-1} - (X_p - W_{av})} (A_R + C_R) B_R e^{-B_R((X_p - W_{av}) - W_R)} e^{-B_R x_w} x_w dx_w \\ &\quad + \sum_{s=1}^{R-1} \left[\int_{W_s - (X_p - W_{av})}^{W_{s+1} - (X_p - W_{av})} (A_s + C_s) B_s e^{-B_s((X_p - W_{av}) - W_s)} e^{-B_s x_w} x_w dx_w \right] \end{aligned}$$

$$\begin{aligned}
&= - \left[(\mathbb{A}_R + \mathbb{C}_R) x_w e^{-\mathbb{B}_R((X_p - W_{av}) - W_R)} e^{-\mathbb{B}_R x_w} + \frac{(\mathbb{A}_R + \mathbb{C}_R)}{\mathbb{B}_R} e^{-\mathbb{B}_R((X_p - W_{av}) - W_R)} e^{-\mathbb{B}_R x_w} \right]_0^{W_{R-1} - (X_p - W_{av})} \\
&\quad - \sum_{s=1}^{R-1} \left[(\mathbb{A}_s + \mathbb{C}_s) x_w e^{-\mathbb{B}_s((X_p - W_{av}) - W_s)} e^{-\mathbb{B}_s x_w} + \frac{(\mathbb{A}_s + \mathbb{C}_s)}{\mathbb{B}_s} e^{-\mathbb{B}_s((X_p - W_{av}) - W_s)} e^{-\mathbb{B}_s x_w} \right]_{W_s - (X_p - W_{av})}^{W_{s-1} - (X_p - W_{av})} \\
&= \frac{(\mathbb{A}_R + \mathbb{C}_R)}{\mathbb{B}_R} e^{-\mathbb{B}_R((X_p - W_{av}) - W_R)} - (\mathbb{A}_R + \mathbb{C}_R) (W_{R-1} - (X_p - W_{av})) e^{-\mathbb{B}_R(W_{R-1} - W_R)} - \\
&\quad \frac{(\mathbb{A}_R + \mathbb{C}_R)}{\mathbb{B}_R} e^{-\mathbb{B}_R(W_{R-1} - W_R)} \\
&\quad + \sum_{s=1}^{R-1} \left[(\mathbb{A}_s + \mathbb{C}_s) (W_s - (X_p - W_{av})) - (\mathbb{A}_s + \mathbb{C}_s) (W_{s-1} - (X_p - W_{av})) e^{-\mathbb{B}_s(W_{s-1} - W_s)} \right. \\
&\quad \quad \left. + \frac{(\mathbb{A}_s + \mathbb{C}_s)}{\mathbb{B}_s} - \frac{(\mathbb{A}_s + \mathbb{C}_s)}{\mathbb{B}_s} e^{-\mathbb{B}_s(W_{s-1} - W_s)} \right] \\
&= \frac{(\mathbb{A}_R + \mathbb{C}_R)}{\mathbb{B}_R} e^{-\mathbb{B}_R((X_p - W_{av}) - W_R)} - \frac{(\mathbb{A}_{R-1} + \mathbb{C}_R)}{\mathbb{B}_R} - (\mathbb{A}_{R-1} + \mathbb{C}_R) (W_{R-1} - (X_p - W_{av})) \\
&\quad + \sum_{s=1}^{R-1} \left[\mathbb{A}_s (W_s - (X_p - W_{av})) - \mathbb{A}_{s-1} (W_{s-1} - (X_p - W_{av})) \right. \\
&\quad \quad \left. - \mathbb{C}_s (W_{s-1} - W_s) + \frac{\mathbb{A}_s - \mathbb{A}_{s-1}}{\mathbb{B}_s} \right] \\
&= \frac{(\mathbb{A}_R + \mathbb{C}_R)}{\mathbb{B}_R} e^{-\mathbb{B}_R((X_p - W_{av}) - W_R)} - \frac{(\mathbb{A}_{R-1} + \mathbb{C}_R)}{\mathbb{B}_R} - \mathbb{C}_R (W_{R-1} - (X_p - W_{av})) \\
&\quad + \sum_{s=1}^{R-1} \left[\frac{\mathbb{A}_s - \mathbb{A}_{s-1}}{\mathbb{B}_s} - \mathbb{C}_s (W_{s-1} - W_s) \right]
\end{aligned}$$

As previously, when $\mathbb{B}_s \rightarrow 0$ and $\mathbb{C}_s \rightarrow \infty$ we can use the linear forms given in [11] for the terms of the sum in [33], but when $\mathbb{B}_R \rightarrow 0$ and $\mathbb{C}_R \rightarrow \infty$, we must replace the R terms with

$$\begin{aligned}
[34] \quad 1 - \chi'(X_p) &\approx \frac{1}{2} \frac{\mathbb{A}_R - \mathbb{A}_{R-1}}{W_{R-1} - W_R} (W_{R-1} - (X_p - W_{av}))^2 + \mathbb{A}_{R-1} (W_{R-1} - (X_p - W_{av})) \\
&\quad + \sum_{s=1}^{R-1} \begin{cases} \left[\frac{1}{2} (\mathbb{A}_s + \mathbb{A}_{s-1}) (W_{s-1} - W_s) \right] & \text{as } \mathbb{B}_s \rightarrow 0, \mathbb{C}_s \rightarrow \infty \\ \left[\frac{(\mathbb{A}_s - \mathbb{A}_{s-1})}{\mathbb{B}_s} - \mathbb{C}_s (W_{s-1} - W_s) \right] & \text{otherwise} \end{cases}
\end{aligned}$$

A general expression for the total proximity zone volume $\chi'(X_p)$ is therefore given by

$$[35] \quad \chi'(X_p) \approx \begin{cases} MF P_{32} X_p & \text{if } X_p \leq W_{av} \\ \left(1 - \frac{(\mathbb{A}_R + \mathbb{C}_R)}{\mathbb{B}_R} e^{-\mathbb{B}_R((X_p - W_{av}) - W_R)} + \frac{(\mathbb{A}_{R-1} + \mathbb{C}_R)}{\mathbb{B}_R} \right. \\ \quad \left. + \mathbb{C}_R (W_{R-1} - (X_p - W_{av})) - \sum_{s=1}^{R-1} \left[\frac{\mathbb{A}_s - \mathbb{A}_{s-1}}{\mathbb{B}_s} - \mathbb{C}_s (W_{s-1} - W_s) \right] \right) & \text{if } X_p > W_{av} \end{cases}$$

As in the uniform stress shadow width scenario, if we want to calculate the proportion of the transect that lies within an arbitrary distance X_d on either side of the nearest macrofracture, we must set $X_p = 2X_d$, whereas if we want to calculate the proportion of the transect that lies within an arbitrary distance X_d on one side (e.g. the southernmost side) of the nearest macrofracture (the “one sided proximity zone volume”), we must set $X_p = X_d$.

2.1.5 Direct calculation of \mathbb{B}_r and \mathbb{C}_r

The expressions for the spacing distribution function $S(x)$ given by equation [1] are approximations; in general $S(x)$ will not always follow a single exponential curve for values of x between W_{r-1} and W_r when $r > 1$. We can show this by demonstrating that the gradient of $dS(x)/dx$ will not always be continuous as we add additional macrofractures from fracture dipset q .

If $q \geq r$, so that $x \geq W_q$, then $S(x)$ for $W_r \leq x < W_r$ will change following

$$\begin{aligned}
[36] \quad \frac{dS(x)}{d_{\text{MF}^{\text{P}}_{32}}^{q_2}} &= \frac{\int_x^{x+W_q} \frac{dS(x_n)}{dx_n} (x_n - W_q) dx_n - \int_{x+W_q}^{\infty} \frac{dS(x_n)}{dx_n} (x_n - W_q) \left(1 - \frac{2x}{x_n - W_q}\right) dx_n}{\int_{W_q}^{\infty} \frac{dS(x_n)}{dx_n} (x_n - W) dx_n} \\
&= \frac{\int_{x-W_q}^x \frac{dS(x_w + W_q)}{dx_w} x_w dx_w - \int_x^{\infty} \frac{dS(x_w + W_q)}{dx_w} x_w dx_w - 2xS(x + W_q)}{\int_0^{\infty} \frac{dS(x_w + W_q)}{dx_w} x_w dx_w} \\
&= \\
&\frac{1}{1-\chi_q} \left(\int_{x-W_q}^{W_{r-1}-W_q} \frac{dS(x_w + W_q)}{dx_w} x_w dx_w + \sum_{s=R+1}^{r-1} \left[\int_{W_s-W_q}^{W_{s-1}-W_q} \frac{dS(x_w + W_q)}{dx_w} x_w dx_w \right] + \int_{W_R-W_q}^x \frac{dS(x_w + W_q)}{dx_w} x_w dx_w \right. \\
&\quad \left. - \int_x^{W_{R-1}-W_q} \frac{dS(x_w + W_q)}{dx_w} x_w dx_w - \sum_{s=1}^{R-1} \left[\int_{W_s-W_q}^{W_{s-1}-W_q} \frac{dS(x_w + W_q)}{dx_w} x_w dx_w \right] - 2xS(x + W_q) \right) \\
&= \\
&\frac{1}{1-\chi_q} \left(\int_{x-W_q}^{W_{r-1}-W_q} -(\mathbb{A}_r + \mathbb{C}_r) \mathbb{B}_r e^{\mathbb{B}_r(W_r-W_q)} e^{-\mathbb{B}_r x_w} x_w dx_w \right. \\
&\quad + \sum_{s=R+1}^{r-1} \left[\int_{W_s-W_q}^{W_{s-1}-W_q} -(\mathbb{A}_s + \mathbb{C}_s) \mathbb{B}_s e^{\mathbb{B}_s(W_s-W_q)} e^{-\mathbb{B}_s x_w} x_w dx_w \right] \\
&\quad + \int_{W_R-W_q}^x -(\mathbb{A}_R + \mathbb{C}_R) \mathbb{B}_R e^{\mathbb{B}_R(W_R-W_q)} e^{-\mathbb{B}_R x_w} x_w dx_w \\
&\quad \left. - \int_x^{W_{R-1}-W_q} -(\mathbb{A}_R + \mathbb{C}_R) \mathbb{B}_R e^{\mathbb{B}_R(W_R-W_q)} e^{-\mathbb{B}_R x_w} x_w dx_w \right. \\
&\quad \left. - \sum_{s=1}^{R-1} \left[\int_{W_s-W_q}^{W_{s-1}-W_q} -(\mathbb{A}_s + \mathbb{C}_s) \mathbb{B}_s e^{\mathbb{B}_s(W_s-W_q)} e^{-\mathbb{B}_s x_w} x_w dx_w \right] - 2x \left((\mathbb{A}_R + \mathbb{C}_R) e^{\mathbb{B}_R(W_R-W_q)} e^{-\mathbb{B}_R x} - \mathbb{C}_R \right) \right) \\
&= \frac{1}{1-\chi_q} \left(\left[(\mathbb{A}_r + \mathbb{C}_r) x_w e^{\mathbb{B}_r(W_r-W_q)} e^{-\mathbb{B}_r x_w} + \frac{(\mathbb{A}_r + \mathbb{C}_r)}{\mathbb{B}_r} e^{\mathbb{B}_r(W_r-W_q)} e^{-\mathbb{B}_r x_w} \right]_{x-W_q}^{W_{r-1}-W_q} \right. \\
&\quad + \sum_{s=R+1}^{r-1} \left[(\mathbb{A}_s + \mathbb{C}_s) x_w e^{\mathbb{B}_s(W_s-W_q)} e^{-\mathbb{B}_s x_w} + \frac{(\mathbb{A}_s + \mathbb{C}_s)}{\mathbb{B}_s} e^{\mathbb{B}_s(W_s-W_q)} e^{-\mathbb{B}_s x_w} \right]_{W_s-W_q}^{W_{s-1}-W_q} \\
&\quad + \left[(\mathbb{A}_R + \mathbb{C}_R) x_w e^{\mathbb{B}_R(W_R-W_q)} e^{-\mathbb{B}_R x_w} + \frac{(\mathbb{A}_R + \mathbb{C}_R)}{\mathbb{B}_R} e^{\mathbb{B}_R(W_R-W_q)} e^{-\mathbb{B}_R x_w} \right]_{W_R-W_q}^x \\
&\quad - \left[(\mathbb{A}_R + \mathbb{C}_R) x_w e^{\mathbb{B}_R(W_R-W_q)} e^{-\mathbb{B}_R x_w} + \frac{(\mathbb{A}_R + \mathbb{C}_R)}{\mathbb{B}_R} e^{\mathbb{B}_R(W_R-W_q)} e^{-\mathbb{B}_R x_w} \right]_x^{W_{R-1}-W_q} \\
&\quad - \sum_{s=1}^{R-1} \left[(\mathbb{A}_s + \mathbb{C}_s) x_w e^{\mathbb{B}_s(W_s-W_q)} e^{-\mathbb{B}_s x_w} + \frac{(\mathbb{A}_s + \mathbb{C}_s)}{\mathbb{B}_s} e^{\mathbb{B}_s(W_s-W_q)} e^{-\mathbb{B}_s x_w} \right]_{W_s-W_q}^{W_{s-1}-W_q} \\
&\quad \left. - 2x(\mathbb{A}_R + \mathbb{C}_R) e^{\mathbb{B}_R(W_R-W_q)} e^{-\mathbb{B}_R x} + 2x\mathbb{C}_R \right) \\
&= \\
&\frac{1}{1-\chi_q} \left((\mathbb{A}_{r-1} + \mathbb{C}_r)(W_{r-1} - W_q) + \frac{(\mathbb{A}_{r-1} + \mathbb{C}_r)}{\mathbb{B}_r} - (\mathbb{A}_r + \mathbb{C}_r)(x - W_q) e^{-\mathbb{B}_r(x-W_r)} - \frac{(\mathbb{A}_r + \mathbb{C}_r)}{\mathbb{B}_r} e^{-\mathbb{B}_r(x-W_r)} \right. \\
&\quad - \sum_{s=R+1}^{r-1} \left[\frac{(\mathbb{A}_s - \mathbb{A}_{s-1})}{\mathbb{B}_s} + (\mathbb{A}_s + \mathbb{C}_s)(W_s - W_q) - (\mathbb{A}_{s-1} + \mathbb{C}_s)(W_{s-1} - W_q) \right] \\
&\quad + (\mathbb{A}_R + \mathbb{C}_R) x e^{-\mathbb{B}_R(x+W_q-W_R)} + \frac{(\mathbb{A}_R + \mathbb{C}_R)}{\mathbb{B}_R} e^{-\mathbb{B}_R(x+W_q-W_R)} - (\mathbb{A}_R + \mathbb{C}_R)(W_R - W_q) - \frac{(\mathbb{A}_R + \mathbb{C}_R)}{\mathbb{B}_R} \\
&\quad - (\mathbb{A}_{R-1} + \mathbb{C}_R)(W_{R-1} - W_q) - \frac{(\mathbb{A}_{R-1} + \mathbb{C}_R)}{\mathbb{B}_R} + (\mathbb{A}_R + \mathbb{C}_R) x e^{-\mathbb{B}_R(x+W_q-W_R)} + \frac{(\mathbb{A}_R + \mathbb{C}_R)}{\mathbb{B}_R} e^{-\mathbb{B}_R(x+W_q-W_R)} \\
&\quad + \sum_{s=1}^{R-1} \left[\frac{(\mathbb{A}_s - \mathbb{A}_{s-1})}{\mathbb{B}_s} + (\mathbb{A}_s + \mathbb{C}_s)(W_s - W_q) - (\mathbb{A}_{s-1} + \mathbb{C}_s)(W_{s-1} - W_q) \right] \\
&\quad \left. - 2x(\mathbb{A}_R + \mathbb{C}_R) e^{\mathbb{B}_R(W_R-W_q)} e^{-\mathbb{B}_R x} + 2x\mathbb{C}_R \right) \\
&= \frac{1}{1-\chi_q} \left(-(\mathbb{A}_r + \mathbb{C}_r)(x - W_q) e^{-\mathbb{B}_r(x-W_r)} - \frac{(\mathbb{A}_r + \mathbb{C}_r)}{\mathbb{B}_r} e^{-\mathbb{B}_r(x-W_r)} + \frac{\mathbb{C}_r}{\mathbb{B}_r} \right. \\
&\quad - \sum_{s=R}^{r-1} \left[\frac{\mathbb{A}_s}{\mathbb{B}_s} - \frac{\mathbb{A}_s}{\mathbb{B}_{s+1}} - (\mathbb{C}_{s+1} - \mathbb{C}_s)(W_s - W_q) \right] \\
&\quad + 2 \frac{(\mathbb{A}_R + \mathbb{C}_R)}{\mathbb{B}_R} e^{-\mathbb{B}_R(x+W_q-W_R)} + 2x\mathbb{C}_R \\
&\quad \left. - 2 \frac{\mathbb{C}_R}{\mathbb{B}_R} + \sum_{s=1}^{R-1} \left[\frac{\mathbb{A}_s}{\mathbb{B}_s} - \frac{\mathbb{A}_s}{\mathbb{B}_{s+1}} - (\mathbb{C}_{s+1} - \mathbb{C}_s)(W_s - W_q) \right] \right)
\end{aligned}$$

Therefore

$$\begin{aligned}
[37] \quad \frac{d^2 S}{d_{\text{MFP}_{32}} dx} &= \frac{1}{1-\chi q} \begin{pmatrix} (\mathbb{A}_r + \mathbb{C}_r) \mathbb{B}_r (x - W_q) e^{-\mathbb{B}_r(x-W_r)} \\ -(\mathbb{A}_r + \mathbb{C}_r) e^{-\mathbb{B}_r(x-W_r)} \\ +(\mathbb{A}_r + \mathbb{C}_r) e^{-\mathbb{B}_r(x-W_r)} \\ -2(\mathbb{A}_R + \mathbb{C}_R) e^{-\mathbb{B}_R(x+W_q-W_R)} + 2\mathbb{C}_R \end{pmatrix} \\
&= \frac{1}{1-\chi q} \left((\mathbb{A}_r + \mathbb{C}_r) \mathbb{B}_r (x - W_q) e^{-\mathbb{B}_r(x-W_r)} - 2(\mathbb{A}_R + \mathbb{C}_R) e^{-\mathbb{B}_R(x+W_q-W_R)} + 2\mathbb{C}_R \right)
\end{aligned}$$

The fracture dipset R is defined such that $W_R \leq x+W_q < W_{R-1}$. Clearly this may vary between $x = W_r$ and $x = W_{r-1}$, independently of the values of W_r and W_{r-1} , in which case there will be a break in the gradient $dS(x)/dx$, and $S(x)$ cannot follow a single exponential curve between $x = W_r$ and $x = W_{r-1}$.

Similarly, when $q < r$ and $x < W_q$ then W_{r-1} must also be $\leq W_q$, so

$$\begin{aligned}
[38] \quad \frac{dS(x)}{d_{\text{MFP}_{32}}} &= \frac{\int_{W_q}^{x+W_q} \frac{dS(x_n)}{dx_n} (x_n - W_q) dx_n - \int_{x+W_q}^{\infty} \frac{dS(x_n)}{dx_n} (x_n - W_q) \left(1 - \frac{2x}{x_n - W_q}\right) dx_n}{\int_{W_q}^{\infty} \frac{dS(x_n)}{dx_n} (x_n - W) dx_n} \\
&= \frac{\int_0^x \frac{dS(x_w + W_q)}{dx_w} x_w dx_w - \int_x^{\infty} \frac{dS(x_w + W_q)}{dx_w} x_w dx_w - 2xS(x+W_q)}{\int_0^{\infty} \frac{dS(x_w + W_q)}{dx_w} x_w dx_w} \\
&= \frac{1}{1-\chi q} \left(\begin{aligned} &\sum_{s=R+1}^q \left[\int_{W_s - W_q}^{W_{s-1} - W_q} \frac{dS(x_w + W_q)}{dx_w} x_w dx_w \right] + \int_{W_R - W_q}^x \frac{dS(x_w + W_q)}{dx_w} x_w dx_w \\ &- \int_x^{W_{R-1} - W_q} \frac{dS(x_w + W_q)}{dx_w} x_w dx_w - \sum_{s=1}^{R-1} \left[\int_{W_s - W_q}^{W_{s-1} - W_q} \frac{dS(x_w + W_q)}{dx_w} x_w dx_w \right] - 2xS(x+W_q) \end{aligned} \right) \\
&= \frac{1}{1-\chi q} \left(\begin{aligned} &\sum_{s=R+1}^q \left[\int_{W_s - W_q}^{W_{s-1} - W_q} -(\mathbb{A}_s + \mathbb{C}_s) \mathbb{B}_s e^{\mathbb{B}_s(W_s - W_q)} e^{-\mathbb{B}_s x_w} x_w dx_w \right] \\ &+ \int_{W_R - W_q}^x -(\mathbb{A}_R + \mathbb{C}_R) \mathbb{B}_R e^{\mathbb{B}_R(W_R - W_q)} e^{-\mathbb{B}_R x_w} x_w dx_w \\ &- \int_x^{W_{R-1} - W_q} -(\mathbb{A}_R + \mathbb{C}_R) \mathbb{B}_R e^{\mathbb{B}_R(W_R - W_q)} e^{-\mathbb{B}_R x_w} x_w dx_w \\ &- \sum_{s=1}^{R-1} \left[\int_{W_s - W_q}^{W_{s-1} - W_q} -(\mathbb{A}_s + \mathbb{C}_s) \mathbb{B}_s e^{\mathbb{B}_s(W_s - W_q)} e^{-\mathbb{B}_s x_w} x_w dx_w \right] \\ &- 2x \left((\mathbb{A}_R + \mathbb{C}_R) e^{\mathbb{B}_R(W_R - W_q)} e^{-\mathbb{B}_R x} - \mathbb{C}_R \right) \end{aligned} \right) \\
&= \frac{1}{1-\chi q} \left(\begin{aligned} &\sum_{s=R+1}^q \left[(\mathbb{A}_s + \mathbb{C}_s) x_w e^{\mathbb{B}_s(W_s - W_q)} e^{-\mathbb{B}_s x_w} + \frac{(\mathbb{A}_s + \mathbb{C}_s)}{\mathbb{B}_s} e^{\mathbb{B}_s(W_s - W_q)} e^{-\mathbb{B}_s x_w} \right]_{W_s - W_q}^{W_{s-1} - W_q} \\ &+ \left[(\mathbb{A}_R + \mathbb{C}_R) x_w e^{\mathbb{B}_R(W_R - W_q)} e^{-\mathbb{B}_R x_w} + \frac{(\mathbb{A}_R + \mathbb{C}_R)}{\mathbb{B}_R} e^{\mathbb{B}_R(W_R - W_q)} e^{-\mathbb{B}_R x_w} \right]_{W_R - W_q}^x \\ &- \left[(\mathbb{A}_R + \mathbb{C}_R) x_w e^{\mathbb{B}_R(W_R - W_q)} e^{-\mathbb{B}_R x_w} + \frac{(\mathbb{A}_R + \mathbb{C}_R)}{\mathbb{B}_R} e^{\mathbb{B}_R(W_R - W_q)} e^{-\mathbb{B}_R x_w} \right]_x^{W_{R-1} - W_q} \\ &- \sum_{s=1}^{R-1} \left[(\mathbb{A}_s + \mathbb{C}_s) x_w e^{\mathbb{B}_s(W_s - W_q)} e^{-\mathbb{B}_s x_w} + \frac{(\mathbb{A}_s + \mathbb{C}_s)}{\mathbb{B}_s} e^{\mathbb{B}_s(W_s - W_q)} e^{-\mathbb{B}_s x_w} \right]_{W_s - W_q}^{W_{s-1} - W_q} \\ &- 2x(\mathbb{A}_R + \mathbb{C}_R) e^{\mathbb{B}_R(W_R - W_q)} e^{-\mathbb{B}_R x} + 2x\mathbb{C}_R \end{aligned} \right) \\
&= \frac{1}{1-\chi q} \left(\begin{aligned} &-\sum_{s=R+1}^q \left[\frac{(\mathbb{A}_s - \mathbb{A}_{s-1})}{\mathbb{B}_s} + (\mathbb{A}_s + \mathbb{C}_s)(W_s - W_q) - (\mathbb{A}_{s-1} + \mathbb{C}_s)(W_{s-1} - W_q) \right] \\ &+ (\mathbb{A}_R + \mathbb{C}_R) x e^{-\mathbb{B}_R(x+W_q-W_R)} + \frac{(\mathbb{A}_R + \mathbb{C}_R)}{\mathbb{B}_R} e^{-\mathbb{B}_R(x+W_q-W_R)} - (\mathbb{A}_R + \mathbb{C}_R)(W_R - W_q) - \frac{(\mathbb{A}_R + \mathbb{C}_R)}{\mathbb{B}_R} \\ &- (\mathbb{A}_{R-1} + \mathbb{C}_R)(W_{R-1} - W_q) - \frac{(\mathbb{A}_{R-1} + \mathbb{C}_R)}{\mathbb{B}_R} + (\mathbb{A}_R + \mathbb{C}_R) x e^{-\mathbb{B}_R(x+W_q-W_R)} + \frac{(\mathbb{A}_R + \mathbb{C}_R)}{\mathbb{B}_R} e^{-\mathbb{B}_R(x+W_q-W_R)} \\ &+ \sum_{s=1}^{R-1} \left[\frac{(\mathbb{A}_s - \mathbb{A}_{s-1})}{\mathbb{B}_s} + (\mathbb{A}_s + \mathbb{C}_s)(W_s - W_q) - (\mathbb{A}_{s-1} + \mathbb{C}_s)(W_{s-1} - W_q) \right] \\ &- 2x(\mathbb{A}_R + \mathbb{C}_R) e^{\mathbb{B}_R(W_R - W_q)} e^{-\mathbb{B}_R x} + 2x\mathbb{C}_R \end{aligned} \right)
\end{aligned}$$

$$= \frac{1}{1-\chi_q} \left(-\frac{\mathbb{A}_q}{\mathbb{B}_q} - \sum_{s=R}^{q-1} \left[\frac{\mathbb{A}_s}{\mathbb{B}_s} - \frac{\mathbb{A}_s}{\mathbb{B}_{s+1}} - (\mathbb{C}_{s+1} - \mathbb{C}_s)(W_s - W_q) \right] \right. \\ \left. + 2 \frac{(\mathbb{A}_R + \mathbb{C}_R)}{\mathbb{B}_R} e^{-\mathbb{B}_R(x+W_q-W_R)} + 2x\mathbb{C}_R \right. \\ \left. - 2 \frac{\mathbb{C}_R}{\mathbb{B}_R} + \sum_{s=1}^{R-1} \left[\frac{\mathbb{A}_s}{\mathbb{B}_s} - \frac{\mathbb{A}_s}{\mathbb{B}_{s+1}} - (\mathbb{C}_{s+1} - \mathbb{C}_s)(W_s - W_q) \right] \right)$$

Therefore

$$[39] \quad \frac{d^2S}{d_{\text{MFP}_{32}}dx} = \frac{1}{1-\chi_q} (-2(\mathbb{A}_R + \mathbb{C}_R)e^{-\mathbb{B}_R(x+W_q-W_R)} + 2\mathbb{C}_R)$$

and again there may be a break in the gradient $dS(x)/dx$ between $x = W_r$ and $x = W_{r-1}$.

We can derive expressions for the change in Br by comparing the change in the total volume of all spacings with widths between W_r and W_{r-1} as we add additional macrofractures from dipset q , calculated by integrating [37] or [39], with the expressions obtained by direct differentiation of [1].

When $q \geq r$, integrating [37] gives

$$[40] \quad \int_{W_r}^{W_{r-1}} -\frac{d^2S}{d_{\text{MFP}_{32}}dx} x dx = \frac{1}{1-\chi_q} \int_{W_r}^{W_{r-1}} \left(-(\mathbb{A}_r + \mathbb{C}_r)\mathbb{B}_r x^2 e^{-\mathbb{B}_r(x-W_r)} \right. \\ \left. + (\mathbb{A}_r + \mathbb{C}_r)\mathbb{B}_r W_q x e^{-\mathbb{B}_r(x-W_r)} \right) dx \\ + \frac{1}{1-\chi_q} \int_{W_r}^{W_{r-1}} (2(\mathbb{A}_R + \mathbb{C}_R)x e^{-\mathbb{B}_R(x+W_q-W_R)} - 2\mathbb{C}_R x) dx$$

Although the first integral in [40] can be solved easily, the second is more difficult since the fracture dipset R , defined such that $W_R \leq x+W_q < W_{R-1}$, may vary between $x = W_r$ and $x = W_{r-1}$. In this case, we must split this integral up into separate terms:

$$[41] \quad \int_{W_r}^{W_{r-1}} (2(\mathbb{A}_R + \mathbb{C}_R)x e^{-\mathbb{B}_R(x+W_q-W_R)} - 2\mathbb{C}_R x) dx \\ = \int_{W_r}^{W_{U-1}-W_q} (2(\mathbb{A}_U + \mathbb{C}_U)x e^{-\mathbb{B}_U(x+W_q-W_U)} - 2\mathbb{C}_U x) dx \\ + \sum_{s=V+1}^{U-1} \int_{W_s-W_q}^{W_{s-1}-W_q} (2(\mathbb{A}_s + \mathbb{C}_s)x e^{-\mathbb{B}_s(x+W_q-W_s)} - 2\mathbb{C}_s x) dx \\ + \int_{W_V-W_q}^{W_{r-1}} (2(\mathbb{A}_V + \mathbb{C}_V)x e^{-\mathbb{B}_V(x+W_q-W_V)} - 2\mathbb{C}_V x) dx \\ = - \left[2 \frac{(\mathbb{A}_U + \mathbb{C}_U)}{\mathbb{B}_U} \left(x + \frac{1}{\mathbb{B}_U} \right) e^{-\mathbb{B}_U(x+W_q-W_U)} + \mathbb{C}_U x^2 \right]_{W_r}^{W_{U-1}-W_q} \\ - \sum_{s=V+1}^{U-1} \left[2 \frac{(\mathbb{A}_s + \mathbb{C}_s)}{\mathbb{B}_s} \left(x + \frac{1}{\mathbb{B}_s} \right) e^{-\mathbb{B}_s(x+W_q-W_s)} + \mathbb{C}_s x^2 \right]_{W_s-W_q}^{W_{s-1}-W_q} \\ - \left[2 \frac{(\mathbb{A}_V + \mathbb{C}_V)}{\mathbb{B}_V} \left(x + \frac{1}{\mathbb{B}_V} \right) e^{-\mathbb{B}_V(x+W_q-W_V)} + \mathbb{C}_V x^2 \right]_{W_V-W_q}^{W_{r-1}} \\ = 2 \frac{(\mathbb{A}_U + \mathbb{C}_U)}{\mathbb{B}_U} \left(W_r + \frac{1}{\mathbb{B}_U} \right) e^{-\mathbb{B}_U(W_r+W_q-W_U)} + \mathbb{C}_U W_r^2 \\ - 2 \frac{(\mathbb{A}_{U-1} + \mathbb{C}_U)}{\mathbb{B}_U} \left(W_{U-1} - W_q + \frac{1}{\mathbb{B}_U} \right) - \mathbb{C}_U (W_{U-1} - W_q)^2 \\ + \sum_{s=V+1}^{U-1} \left[2 \frac{(\mathbb{A}_s + \mathbb{C}_s)}{\mathbb{B}_s} \left(W_s - W_q + \frac{1}{\mathbb{B}_s} \right) + \mathbb{C}_s (W_s - W_q)^2 \right. \\ \left. - 2 \frac{(\mathbb{A}_{s-1} + \mathbb{C}_s)}{\mathbb{B}_s} \left(W_{s-1} - W_q + \frac{1}{\mathbb{B}_s} \right) + \mathbb{C}_s (W_{s-1} - W_q)^2 \right] \\ + 2 \frac{(\mathbb{A}_V + \mathbb{C}_V)}{\mathbb{B}_V} \left(W_V - W_q + \frac{1}{\mathbb{B}_V} \right) + \mathbb{C}_V (W_V - W_q)^2$$

$$\begin{aligned}
& -2 \frac{(\mathbb{A}_V + \mathbb{C}_V)}{\mathbb{B}_V} \left(W_{r-1} + \frac{1}{\mathbb{B}_V} \right) e^{-\mathbb{B}_V(W_{r-1} + W_q - W_V)} - \mathbb{C}_V W_{r-1}^2 \\
& = 2 \frac{(\mathbb{A}_U + \mathbb{C}_U)}{\mathbb{B}_U} \left(W_r + \frac{1}{\mathbb{B}_U} \right) e^{-\mathbb{B}_U(W_r + W_q - W_U)} + \mathbb{C}_U W_r^2 - 2 \frac{\mathbb{C}_U}{\mathbb{B}_U^2} \\
& + \sum_{s=V}^{U-1} \left[2 \frac{\mathbb{A}_s}{\mathbb{B}_s^2} - 2 \frac{\mathbb{A}_s}{\mathbb{B}_{s+1}^2} + 2 \left(\frac{(\mathbb{A}_s + \mathbb{C}_s)}{\mathbb{B}_s} - \frac{(\mathbb{A}_s + \mathbb{C}_{s+1})}{\mathbb{B}_{s+1}} \right) (W_s - W_q) - (\mathbb{C}_{s+1} - \mathbb{C}_s) (W_s - W_q)^2 \right] \\
& + 2 \frac{\mathbb{C}_V}{\mathbb{B}_V^2} - \mathbb{C}_V W_{r-1}^2 - 2 \frac{(\mathbb{A}_V + \mathbb{C}_V)}{\mathbb{B}_V} \left(W_{r-1} + \frac{1}{\mathbb{B}_V} \right) e^{-\mathbb{B}_V(W_{r-1} + W_q - W_V)}
\end{aligned}$$

where fracture dipsets U and V are defined such that $W_U \leq W_r + W_q < W_{U-1}$, and $W_V \leq W_{r-1} + W_q < W_{V-1}$. Note that if $U = V$, [41] reduces to

$$\begin{aligned}
[42] \quad & \int_{W_r}^{W_{r-1}} \left(2(\mathbb{A}_R + \mathbb{C}_R) x e^{-\mathbb{B}_R(x + W_q - W_R)} - 2\mathbb{C}_R x \right) dx \\
& = 2 \frac{(\mathbb{A}_U + \mathbb{C}_U)}{\mathbb{B}_U} \left(\left(W_r + \frac{1}{\mathbb{B}_U} \right) e^{-\mathbb{B}_U(W_r + W_q - W_U)} - \left(W_{r-1} + \frac{1}{\mathbb{B}_U} \right) e^{-\mathbb{B}_U(W_{r-1} + W_q - W_U)} \right) \\
& + \mathbb{C}_U (W_{r-1}^2 - W_r^2)
\end{aligned}$$

Inserting [41] into [40] gives

$$\begin{aligned}
[43] \quad & \int_{W_r}^{W_{r-1}} - \frac{dS^2}{d_{\text{MFP}_{32}} dx} x dx = \\
& \frac{1}{1 - \chi_q} \left(\begin{aligned} & (\mathbb{A}_{r-1} + \mathbb{C}_r) \left(W_{r-1}^2 + \frac{2W_{r-1}}{\mathbb{B}_r} + \frac{2}{\mathbb{B}_r^2} \right) - (\mathbb{A}_r + \mathbb{C}_r) \left(W_r^2 + \frac{2W_r}{\mathbb{B}_r} + \frac{2}{\mathbb{B}_r^2} \right) \\ & - (\mathbb{A}_{r-1} + \mathbb{C}_r) W_q \left(W_{r-1} + \frac{1}{\mathbb{B}_r} \right) + (\mathbb{A}_r + \mathbb{C}_r) W_q \left(W_r + \frac{1}{\mathbb{B}_r} \right) \\ & + 2 \frac{(\mathbb{A}_U + \mathbb{C}_U)}{\mathbb{B}_U} \left(W_r + \frac{1}{\mathbb{B}_U} \right) e^{-\mathbb{B}_U(W_r + W_q - W_U)} + \mathbb{C}_U W_r^2 - 2 \frac{\mathbb{C}_U}{\mathbb{B}_U^2} \\ & + \sum_{s=V}^{U-1} \left[2 \frac{\mathbb{A}_s}{\mathbb{B}_s^2} - 2 \frac{\mathbb{A}_s}{\mathbb{B}_{s+1}^2} + 2 \left(\frac{(\mathbb{A}_s + \mathbb{C}_s)}{\mathbb{B}_s} - \frac{(\mathbb{A}_s + \mathbb{C}_{s+1})}{\mathbb{B}_{s+1}} \right) (W_s - W_q) \right. \\ & \quad \left. - (\mathbb{C}_{s+1} - \mathbb{C}_s) (W_s - W_q)^2 \right] \\ & + 2 \frac{\mathbb{C}_V}{\mathbb{B}_V^2} - \mathbb{C}_V W_{r-1}^2 - 2 \frac{(\mathbb{A}_V + \mathbb{C}_V)}{\mathbb{B}_V} \left(W_{r-1} + \frac{1}{\mathbb{B}_V} \right) e^{-\mathbb{B}_V(W_{r-1} + W_q - W_V)} \end{aligned} \right)
\end{aligned}$$

Similarly, when $q > r$ integration of [39] gives

$$\begin{aligned}
[44] \quad & \int_{W_r}^{W_{r-1}} - \frac{d^2 S}{d_{\text{MFP}_{32}} dx} x dx = \frac{1}{1 - \chi_q} \int_{W_r}^{W_{r-1}} \left(2(\mathbb{A}_R + \mathbb{C}_R) x e^{-\mathbb{B}_R(x + W_q - W_R)} - 2\mathbb{C}_R x \right) dx \\
& = \frac{1}{1 - \chi_q} \left(\begin{aligned} & 2 \frac{(\mathbb{A}_U + \mathbb{C}_U)}{\mathbb{B}_U} \left(W_r + \frac{1}{\mathbb{B}_U} \right) e^{-\mathbb{B}_U(W_r + W_q - W_U)} + \mathbb{C}_U W_r^2 - 2 \frac{\mathbb{C}_U}{\mathbb{B}_U^2} \\ & + \sum_{s=V}^{U-1} \left[2 \frac{\mathbb{A}_s}{\mathbb{B}_s^2} - 2 \frac{\mathbb{A}_s}{\mathbb{B}_{s+1}^2} + 2 \left(\frac{(\mathbb{A}_s + \mathbb{C}_s)}{\mathbb{B}_s} - \frac{(\mathbb{A}_s + \mathbb{C}_{s+1})}{\mathbb{B}_{s+1}} \right) (W_s - W_q) \right. \\ & \quad \left. - (\mathbb{C}_{s+1} - \mathbb{C}_s) (W_s - W_q)^2 \right] \\ & + 2 \frac{\mathbb{C}_V}{\mathbb{B}_V^2} - \mathbb{C}_V W_{r-1}^2 - 2 \frac{(\mathbb{A}_V + \mathbb{C}_V)}{\mathbb{B}_V} \left(W_{r-1} + \frac{1}{\mathbb{B}_V} \right) e^{-\mathbb{B}_V(W_{r-1} + W_q - W_V)} \end{aligned} \right)
\end{aligned}$$

Direct differentiation of [1] gives

$$\begin{aligned}
[45] \quad & \int_{W_r}^{W_{r-1}} - \frac{d^2 S}{d_{\text{MFP}_{32}} dx} x dx = \frac{d}{d_{\text{MFP}_{32}}} \int_{W_r}^{W_{r-1}} - \frac{d}{dx} \left[(\mathbb{A}_r + \mathbb{C}_r) e^{-\mathbb{B}_r(x - W_r)} - \mathbb{C}_r \right] x dx \\
& = \frac{d}{d_{\text{MFP}_{32}}} \int_{W_r}^{W_{r-1}} \left((\mathbb{A}_r + \mathbb{C}_r) \mathbb{B}_r x e^{-\mathbb{B}_r(x - W_r)} \right) dx
\end{aligned}$$

$$\begin{aligned}
&= \frac{d}{d_{\text{MFP}_{32}}^{q_P}} \left[-(\mathbb{A}_r + \mathbb{C}_r) \left(x + \frac{1}{\mathbb{B}_r} \right) e^{-\mathbb{B}_r(x-W_r)} \right]_{W_r}^{W_{r-1}} \\
&= \frac{d}{d_{\text{MFP}_{32}}^{q_P}} \left[\frac{\mathbb{A}_r - \mathbb{A}_{r-1}}{\mathbb{B}_r} + W_r(\mathbb{A}_r + \mathbb{C}_r) - W_{r-1}(\mathbb{A}_{r-1} + \mathbb{C}_r) \right] \\
&= \frac{1}{\mathbb{B}_r} \left(\frac{d\mathbb{A}_r}{d_{\text{MFP}_{32}}^{q_P}} - \frac{d\mathbb{A}_{r-1}}{d_{\text{MFP}_{32}}^{q_P}} \right) - \frac{d\mathbb{B}_r}{d_{\text{MFP}_{32}}^{q_P}} \frac{\mathbb{A}_r - \mathbb{A}_{r-1}}{\mathbb{B}_r^2} + W_r \frac{d\mathbb{A}_r}{d_{\text{MFP}_{32}}^{q_P}} - W_{r-1} \frac{d\mathbb{A}_{r-1}}{d_{\text{MFP}_{32}}^{q_P}} - (W_{r-1} - W_r) \frac{d\mathbb{C}_r}{d_{\text{MFP}_{32}}^{q_P}}
\end{aligned}$$

Rearranging [2], we can see that

$$[46] \quad \mathbb{C}_r = \frac{\mathbb{A}_r - \mathbb{A}_{r-1}}{1 - e^{-\mathbb{B}_r(W_{r-1} - W_r)}} - \mathbb{A}_r$$

and therefore

$$\begin{aligned}
[47] \quad \frac{d\mathbb{C}_r}{d_{\text{MFP}_{32}}^{q_P}} &= \frac{\frac{d\mathbb{A}_r}{d_{\text{MFP}_{32}}^{q_P}} - \frac{d\mathbb{A}_{r-1}}{d_{\text{MFP}_{32}}^{q_P}}}{1 - e^{-\mathbb{B}_r(W_{r-1} - W_r)}} - \frac{(\mathbb{A}_r - \mathbb{A}_{r-1})(W_{r-1} - W_r)e^{-\mathbb{B}_r(W_{r-1} - W_r)} \frac{d\mathbb{B}_r}{d_{\text{MFP}_{32}}^{q_P}}}{(1 - e^{-\mathbb{B}_r(W_{r-1} - W_r)})^2} - \frac{d\mathbb{A}_r}{d_{\text{MFP}_{32}}^{q_P}} \\
&= \frac{\frac{d\mathbb{A}_r}{d_{\text{MFP}_{32}}^{q_P}} e^{-\mathbb{B}_r(W_{r-1} - W_r)} - \frac{d\mathbb{A}_{r-1}}{d_{\text{MFP}_{32}}^{q_P}}}{1 - e^{-\mathbb{B}_r(W_{r-1} - W_r)}} - \frac{d\mathbb{B}_r}{d_{\text{MFP}_{32}}^{q_P}} \frac{(\mathbb{A}_r - \mathbb{A}_{r-1})(W_{r-1} - W_r)e^{-\mathbb{B}_r(W_{r-1} - W_r)}}{(1 - e^{-\mathbb{B}_r(W_{r-1} - W_r)})^2}
\end{aligned}$$

We can combine [45] and [53] to obtain an expression $d\mathbb{B}_r/d_{\text{MFP}_{32}}^{q_P}$:

$$\begin{aligned}
[48] \quad \int_{W_r}^{W_{r-1}} - \frac{d^2 S}{d_{\text{MFP}_{32}}^{q_P} dx} x dx - \frac{1}{\mathbb{B}_r} \left(\frac{d\mathbb{A}_r}{d_{\text{MFP}_{32}}^{q_P}} - \frac{d\mathbb{A}_{r-1}}{d_{\text{MFP}_{32}}^{q_P}} \right) - W_r \frac{d\mathbb{A}_r}{d_{\text{MFP}_{32}}^{q_P}} + W_{r-1} \frac{d\mathbb{A}_{r-1}}{d_{\text{MFP}_{32}}^{q_P}} + \frac{d\mathbb{B}_r}{d_{\text{MFP}_{32}}^{q_P}} \frac{\mathbb{A}_r - \mathbb{A}_{r-1}}{\mathbb{B}_r^2} \\
= -(W_{r-1} - W_r) \left(\frac{\frac{d\mathbb{A}_r}{d_{\text{MFP}_{32}}^{q_P}} e^{-\mathbb{B}_r(W_{r-1} - W_r)} - \frac{d\mathbb{A}_{r-1}}{d_{\text{MFP}_{32}}^{q_P}}}{(1 - e^{-\mathbb{B}_r(W_{r-1} - W_r)})} - \frac{d\mathbb{B}_r}{d_{\text{MFP}_{32}}^{q_P}} \frac{(\mathbb{A}_r - \mathbb{A}_{r-1})(W_{r-1} - W_r)e^{-\mathbb{B}_r(W_{r-1} - W_r)}}{(1 - e^{-\mathbb{B}_r(W_{r-1} - W_r)})^2} \right) \\
\Rightarrow \int_{W_r}^{W_{r-1}} - \frac{d^2 S}{d_{\text{MFP}_{32}}^{q_P} dx} x dx - \frac{1}{\mathbb{B}_r} \left(\frac{d\mathbb{A}_r}{d_{\text{MFP}_{32}}^{q_P}} - \frac{d\mathbb{A}_{r-1}}{d_{\text{MFP}_{32}}^{q_P}} \right) - W_r \frac{d\mathbb{A}_r}{d_{\text{MFP}_{32}}^{q_P}} + W_{r-1} \frac{d\mathbb{A}_{r-1}}{d_{\text{MFP}_{32}}^{q_P}} \\
+ (W_{r-1} - W_r) \frac{\frac{d\mathbb{A}_r}{d_{\text{MFP}_{32}}^{q_P}} e^{-\mathbb{B}_r(W_{r-1} - W_r)} - \frac{d\mathbb{A}_{r-1}}{d_{\text{MFP}_{32}}^{q_P}}}{(1 - e^{-\mathbb{B}_r(W_{r-1} - W_r)})} \\
= \frac{d\mathbb{B}_r}{d_{\text{MFP}_{32}}^{q_P}} \left(\frac{(\mathbb{A}_r - \mathbb{A}_{r-1})(W_{r-1} - W_r)^2 e^{-\mathbb{B}_r(W_{r-1} - W_r)}}{(1 - e^{-\mathbb{B}_r(W_{r-1} - W_r)})^2} - \frac{\mathbb{A}_r - \mathbb{A}_{r-1}}{\mathbb{B}_r^2} \right) \\
\Rightarrow \frac{d\mathbb{B}_r}{d_{\text{MFP}_{32}}^{q_P}} = \frac{\left(\int_{W_r}^{W_{r-1}} - \frac{d^2 S}{d_{\text{MFP}_{32}}^{q_P} dx} x dx - \frac{1}{\mathbb{B}_r} \left(\frac{d\mathbb{A}_r}{d_{\text{MFP}_{32}}^{q_P}} - \frac{d\mathbb{A}_{r-1}}{d_{\text{MFP}_{32}}^{q_P}} \right) - W_r \frac{d\mathbb{A}_r}{d_{\text{MFP}_{32}}^{q_P}} + W_{r-1} \frac{d\mathbb{A}_{r-1}}{d_{\text{MFP}_{32}}^{q_P}} \right. \\
\left. + (W_{r-1} - W_r) \frac{\frac{d\mathbb{A}_r}{d_{\text{MFP}_{32}}^{q_P}} e^{-\mathbb{B}_r(W_{r-1} - W_r)} - \frac{d\mathbb{A}_{r-1}}{d_{\text{MFP}_{32}}^{q_P}}}{(1 - e^{-\mathbb{B}_r(W_{r-1} - W_r)})} \right)}{\left(\frac{(\mathbb{A}_r - \mathbb{A}_{r-1})(W_{r-1} - W_r)^2 e^{-\mathbb{B}_r(W_{r-1} - W_r)}}{(1 - e^{-\mathbb{B}_r(W_{r-1} - W_r)})^2} - \frac{\mathbb{A}_r - \mathbb{A}_{r-1}}{\mathbb{B}_r^2} \right)} \\
= \frac{\left(\int_{W_r}^{W_{r-1}} - \frac{d^2 S}{d_{\text{MFP}_{32}}^{q_P} dx} x dx - \frac{d\mathbb{A}_r}{d_{\text{MFP}_{32}}^{q_P}} \left(\frac{1}{\mathbb{B}_r} + W_r - (W_{r-1} - W_r) \frac{e^{-\mathbb{B}_r(W_{r-1} - W_r)}}{(1 - e^{-\mathbb{B}_r(W_{r-1} - W_r)})} \right) \right. \\
\left. + \frac{d\mathbb{A}_{r-1}}{d_{\text{MFP}_{32}}^{q_P}} \left(\frac{1}{\mathbb{B}_r} + W_{r-1} - (W_{r-1} - W_r) \frac{1}{(1 - e^{-\mathbb{B}_r(W_{r-1} - W_r)})} \right) \right)}{\left(\frac{(\mathbb{A}_r - \mathbb{A}_{r-1})(W_{r-1} - W_r)^2 e^{-\mathbb{B}_r(W_{r-1} - W_r)}}{(1 - e^{-\mathbb{B}_r(W_{r-1} - W_r)})^2} - \frac{\mathbb{A}_r - \mathbb{A}_{r-1}}{\mathbb{B}_r^2} \right)}
\end{aligned}$$

We can now substitute [43] or [44] into [54] to calculate $d\mathbb{B}_r/d^q_{\text{MF}P_{32}}$. Once we have calculated the new value for \mathbb{B}_r , we can calculate \mathbb{C}_r using [46].

Note that when $\mathbb{B}_r \rightarrow 0$ and $\mathbb{C}_r \rightarrow \infty$, we can use the limit approximations

$$[49] \quad \left. \begin{aligned} \frac{W_{r-1}-W_r}{(1-e^{-\mathbb{B}_r(W_{r-1}-W_r)})} - \frac{1}{\mathbb{B}_r} &\rightarrow \frac{W_{r-1}-W_r}{2} \\ \frac{(W_{r-1}-W_r)^2 e^{-\mathbb{B}_r(W_{r-1}-W_r)}}{(1-e^{-\mathbb{B}_r(W_{r-1}-W_r)})^2} - \frac{1}{\mathbb{B}_r^2} &\rightarrow \frac{W_{r-1}-W_r}{\mathbb{B}_r} \end{aligned} \right\} \text{as } \mathbb{B}_r \rightarrow 0$$

to show that

$$[50] \quad \frac{d\mathbb{B}_r}{d^q_{\text{MF}P_{32}}} = \frac{\left(\int_{W_r}^{W_{r-1}} -\frac{d^2 S}{d^q_{\text{MF}P_{32}} dx} x dx - \frac{d\mathbb{A}_r}{d^q_{\text{MF}P_{32}}} \left(\frac{1}{\mathbb{B}_r} - \frac{W_{r-1}-W_r}{(1-e^{-\mathbb{B}_r(W_{r-1}-W_r)})} + (W_{r-1}-W_r) + W_r \right) \right.}{(\mathbb{A}_r - \mathbb{A}_{r-1}) \left(\frac{(W_{r-1}-W_r)^2 e^{-\mathbb{B}_r(W_{r-1}-W_r)}}{(1-e^{-\mathbb{B}_r(W_{r-1}-W_r)})^2} - \frac{1}{\mathbb{B}_r^2} \right)} \\ \left. + \frac{d\mathbb{A}_{r-1}}{d^q_{\text{MF}P_{32}}} \left(\frac{1}{\mathbb{B}_r} - \frac{W_{r-1}-W_r}{(1-e^{-\mathbb{B}_r(W_{r-1}-W_r)})} + W_{r-1} \right) \right)$$

$$\rightarrow \frac{\int_{W_r}^{W_{r-1}} -\frac{d^2 S}{d^q_{\text{MF}P_{32}} dx} x dx + \left(\frac{d\mathbb{A}_{r-1}}{d^q_{\text{MF}P_{32}}} - \frac{d\mathbb{A}_r}{d^q_{\text{MF}P_{32}}} \right) \frac{W_{r-1}+W_r}{2}}{\frac{(\mathbb{A}_r - \mathbb{A}_{r-1})(W_{r-1}-W_r)}{\mathbb{B}_r}}$$

and

$$[51] \quad (\mathbb{A}_{r-1} + \mathbb{C}_r) \left(W_{r-1}^2 + \frac{2W_{r-1}}{\mathbb{B}_r} + \frac{2}{\mathbb{B}_r^2} \right) - (\mathbb{A}_r + \mathbb{C}_r) \left(W_r^2 + \frac{2W_r}{\mathbb{B}_r} + \frac{2}{\mathbb{B}_r^2} \right) \\ - (\mathbb{A}_{r-1} + \mathbb{C}_r) W_q \left(W_{r-1} + \frac{1}{\mathbb{B}_r} \right) + (\mathbb{A}_r + \mathbb{C}_r) W_q \left(W_r + \frac{1}{\mathbb{B}_r} \right) \\ = (\mathbb{A}_{r-1} W_{r-1}^2 - \mathbb{A}_r W_r^2) + \frac{2}{\mathbb{B}_r} (\mathbb{A}_{r-1} W_{r-1} - \mathbb{A}_r W_r) + \frac{2}{\mathbb{B}_r^2} (\mathbb{A}_{r-1} - \mathbb{A}_r) \\ + \frac{\mathbb{A}_r - \mathbb{A}_{r-1}}{(1-e^{-\mathbb{B}_r(W_{r-1}-W_r)})} (W_{r-1}^2 - W_r^2) + \frac{2}{\mathbb{B}_r} \frac{(\mathbb{A}_r - \mathbb{A}_{r-1})(W_{r-1}-W_r)}{(1-e^{-\mathbb{B}_r(W_{r-1}-W_r)})} \\ - \mathbb{A}_r \left(W_{r-1}^2 - W_r^2 + \frac{2(W_{r-1}-W_r)}{\mathbb{B}_r} \right) \\ - W_q \frac{\mathbb{A}_{r-1} - \mathbb{A}_r}{\mathbb{B}_r} - W_q (\mathbb{A}_{r-1} W_{r-1} - \mathbb{A}_r W_r) \\ - W_q (W_{r-1} - W_r) \frac{\mathbb{A}_r - \mathbb{A}_{r-1}}{(1-e^{-\mathbb{B}_r(W_{r-1}-W_r)})} + W_q (W_{r-1} - W_r) \mathbb{A}_r \\ = -(\mathbb{A}_r - \mathbb{A}_{r-1}) W_{r-1}^2 - \frac{2}{\mathbb{B}_r} (\mathbb{A}_r - \mathbb{A}_{r-1}) W_{r-1} + \frac{2(\mathbb{A}_r - \mathbb{A}_{r-1})}{\mathbb{B}_r} \left(\frac{(W_{r-1}-W_r)}{(1-e^{-\mathbb{B}_r(W_{r-1}-W_r)})} - \frac{1}{\mathbb{B}_r} \right) \\ + \frac{\mathbb{A}_r - \mathbb{A}_{r-1}}{(1-e^{-\mathbb{B}_r(W_{r-1}-W_r)})} (W_{r-1}^2 - W_r^2) \\ + W_q (\mathbb{A}_r - \mathbb{A}_{r-1}) \left(\frac{1}{\mathbb{B}_r} - \frac{(W_{r-1}-W_r)}{(1-e^{-\mathbb{B}_r(W_{r-1}-W_r)})} \right) + W_q W_{r-1} (\mathbb{A}_r - \mathbb{A}_{r-1}) \\ \rightarrow -(\mathbb{A}_r - \mathbb{A}_{r-1}) W_{r-1}^2 - \frac{2}{\mathbb{B}_r} (\mathbb{A}_r - \mathbb{A}_{r-1}) W_{r-1} + \frac{(\mathbb{A}_r - \mathbb{A}_{r-1})(W_{r-1}-W_r)}{\mathbb{B}_r} \\ + \frac{\mathbb{A}_r - \mathbb{A}_{r-1}}{(1-e^{-\mathbb{B}_r(W_{r-1}-W_r)})} (W_{r-1}^2 - W_r^2) \\ - \frac{1}{2} W_q (\mathbb{A}_r - \mathbb{A}_{r-1}) (W_{r-1} - W_r) + W_q W_{r-1} (\mathbb{A}_r - \mathbb{A}_{r-1}) \\ = (\mathbb{A}_r - \mathbb{A}_{r-1}) (W_{r-1} + W_r) \left(\frac{(W_{r-1}-W_r)}{(1-e^{-\mathbb{B}_r(W_{r-1}-W_r)})} - \frac{1}{\mathbb{B}_r} \right) - (\mathbb{A}_r - \mathbb{A}_{r-1}) W_{r-1}^2$$

$$\begin{aligned}
& + \frac{1}{2} W_q (A_r - A_{r-1}) (W_{r-1} + W_r) \\
& \rightarrow -\frac{1}{2} (A_r - A_{r-1}) (W_{r-1}^2 + W_r^2) + \frac{1}{2} W_q (A_r - A_{r-1}) (W_{r-1} + W_r)
\end{aligned}$$

Therefore, as $B_r \rightarrow 0$, the denominator term of [48] becomes infinite while the numerator terms all remain finite, and hence $dB_r/d^q_{MF}P_{32}$ will be zero except when $A_r=0$, in which case we can use the approximation $(A_r - A_{r-1})/B_r \rightarrow 1$, so that

$$[52] \quad \frac{dB_r}{d^q_{MF}P_{32}} \rightarrow \begin{cases} 0 & \text{when } A_r > 0, B_r \rightarrow 0 \\ 1 & \text{when } A_r \rightarrow 0, B_r \rightarrow 0 \end{cases}$$

Alternatively, we can derive an expression for $dC_r/d^q_{MF}P_{32}$ by substituting for $dB_r/d^q_{MF}P_{32}$ in [45]. From [2], we can see that

$$[53] \quad \frac{dB_r}{d^q_{MF}P_{32}} = \frac{\frac{dA_r}{d^q_{MF}P_{32}} + \frac{dC_r}{d^q_{MF}P_{32}}}{(W_{r-1} - W_r)(A_r + C_r)} - \frac{\frac{dA_{r-1}}{d^q_{MF}P_{32}} + \frac{dC_r}{d^q_{MF}P_{32}}}{(W_{r-1} - W_r)(A_{r-1} + C_r)}$$

We can combine [45] and [53] to obtain an expression $dC_r/d^q_{MF}P_{32}$:

$$\begin{aligned}
[54] \quad & \int_{W_r}^{W_{r-1}} -\frac{d^2 S}{d^q_{MF}P_{32} dx} x dx - \frac{1}{B_r} \left(\frac{dA_r}{d^q_{MF}P_{32}} - \frac{dA_{r-1}}{d^q_{MF}P_{32}} \right) - W_r \frac{dA_r}{d^q_{MF}P_{32}} + W_{r-1} \frac{dA_{r-1}}{d^q_{MF}P_{32}} \\
& + \frac{A_r - A_{r-1}}{(W_{r-1} - W_r) B_r^2} \left(\frac{dA_r}{d^q_{MF}P_{32}} - \frac{dA_{r-1}}{d^q_{MF}P_{32}} \right) \\
& = \frac{dC_r}{d^q_{MF}P_{32}} \left(\left(\frac{A_r - A_{r-1}}{(W_{r-1} - W_r) B_r^2} \right) \left(\frac{1}{(A_{r-1} + C_r)} - \frac{1}{(A_r + C_r)} \right) - (W_{r-1} - W_r) \right) \\
& \Rightarrow \frac{dC_r}{d^q_{MF}P_{32}} = \frac{\left(\int_{W_r}^{W_{r-1}} -\frac{d^2 S}{d^q_{MF}P_{32} dx} x dx - \frac{1}{B_r} \left(\frac{dA_r}{d^q_{MF}P_{32}} - \frac{dA_{r-1}}{d^q_{MF}P_{32}} \right) - W_r \frac{dA_r}{d^q_{MF}P_{32}} + W_{r-1} \frac{dA_{r-1}}{d^q_{MF}P_{32}} \right. \\
& \quad \left. + \frac{A_r - A_{r-1}}{(W_{r-1} - W_r) B_r^2} \left(\frac{dA_r}{d^q_{MF}P_{32}} - \frac{dA_{r-1}}{d^q_{MF}P_{32}} \right) \right)}{\left(\left(\frac{A_r - A_{r-1}}{(W_{r-1} - W_r) B_r^2} \right) \left(\frac{1}{(A_{r-1} + C_r)} - \frac{1}{(A_r + C_r)} \right) - (W_{r-1} - W_r) \right)} \\
& = \frac{\left(\int_{W_r}^{W_{r-1}} -\frac{d^2 S}{d^q_{MF}P_{32} dx} x dx - \frac{dA_r}{d^q_{MF}P_{32}} \left(\frac{1}{B_r} + W_r - \frac{A_r - A_{r-1}}{(A_r + C_r)(W_{r-1} - W_r) B_r^2} \right) \right. \\
& \quad \left. + \frac{dA_{r-1}}{d^q_{MF}P_{32}} \left(\frac{1}{B_r} + W_{r-1} - \frac{A_r - A_{r-1}}{(A_{r-1} + C_r)(W_{r-1} - W_r) B_r^2} \right) \right)}{\left(\left(\frac{A_r - A_{r-1}}{(W_{r-1} - W_r) B_r^2} \right) \left(\frac{1}{(A_{r-1} + C_r)} - \frac{1}{(A_r + C_r)} \right) - (W_{r-1} - W_r) \right)}
\end{aligned}$$

We can now substitute [43] or [44] into [54] to calculate $dC_r/d^q_{MF}P_{32}$. Once we have calculated the new value for C_r , we can use [2] to calculate B_r .

In practice, however, the $dB_r/d^q_{MF}P_{32}$ and $dC_r/d^q_{MF}P_{32}$ terms in [45] are often very small compared with the $dA_r/d^q_{MF}P_{32}$ terms. As a result, small errors introduced by the approximation of $S(x)$ with a single exponential curve, especially in the early timesteps, can lead to large errors in the calculated values of B_r and C_r . It is therefore usually preferable to calculate values for B_r and C_r iteratively, as described in Section 2.1.3.

2.2 Probability of microfracture deactivation

In Section 5.1.2 of Welch et al. (2020), we show that microfractures will be deactivated if they come to lie within the stress shadow of a nearby parallel macrofracture. Since this is the only mechanism for microfracture deactivation, the number of microfractures in any fracture set that are deactivated during any time period will be directly proportional to the increase in stress shadow volume of macrofractures of the same set during that period.

2.2.1 Cumulative microfracture activation probability

As we have already seen, all the microfractures are initially active, and they become inactive when their centres come to lie within the stress shadow of a parallel macrofracture. The microfracture centres do not move, so the probability that a macrofracture of set i will be inactive due to macrofracture stress shadow growth at any time t is simply given by the total stress shadow volume of parallel macrofractures of set i , expressed as a proportion of the total rock volume, and denoted as $\psi_{MF}^i(t)$. This can be expressed as:

$$[55] \quad 1 - \theta_{i,I}(t) = \psi_{MF}^i(t)$$

If we have multiple fracture dipsets in the propagating fracture set i , then the total macrofracture stress shadow volume ψ_{MF}^i is given by:

$$[56] \quad \psi_{MF}^i(t) = \sum_{Dipset\ m} [{}^{lm}_{MF}\bar{W} {}^{lm}_{MF}P_{32}(0,t)] = \sum_{Dipset\ m} [2Mm_{hh} {}^{lm}_{MF}P_{33}(0,t)]$$

Note that

- The lm superscript in ${}^{lm}_{MF}P_{32}$, ${}^{lm}_{MF}P_{33}$, ${}^{lm}_{MF}\bar{W}$ and ψ_{MF}^{lm} denotes that we are referring to the total linear density, volumetric ratio, mean stress shadow width and stress shadow volume of only those macrofractures of dipset m in set i .
- Since the total macrofracture stress shadow volume is identical to the total half-macrofracture stress shadow volume, we can substitute ${}^{lm}_{MF}P_{32}(0,t)$ and ${}^{lm}_{MF}P_{33}(0,t)$ for ${}^{lm}_{MF}P_{32}(0,t)$ and ${}^{lm}_{MF}P_{33}(0,t)$ in [56].
- Mm_{hh} is the horizontal mode factor for dipset m , that will depend on whether it contains Mode 1 or Mode 2 fractures.

2.2.2 Effective cumulative microfracture activation probability

We can use [56] to calculate the proportion of microfractures that remain active at the start of any timestep. However if we want to use the cumulative microfracture activation probability to calculate the population of macrofractures that will nucleate in a given timestep, the situation is more complicated.

This is because [56] assumes that the microfractures themselves have no stress shadow, and will therefore cease propagating when the centre of the microfracture falls within the stress shadow of a macrofracture. This will occur when the centre of the microfracture lies within a distance of half the macrofracture stress shadow width (${}^{lm}_{MF}\bar{W}/2$) from a macrofracture of dipset m (since ${}^{lm}_{MF}\bar{W}$ represents the width of the stress shadow on both sides of the macrofracture, ${}^{lm}_{MF}\bar{W}/2$ is the width of the stress shadow on each side). However as soon as the microfractures become macrofractures they will acquire their own stress shadow, and fractures with a stress shadow will stop propagating when their stress shadow intersects the stress shadow of another fracture. This will occur instantaneously if a new macrofracture of dipset n lies within a distance of ${}^{lm}_{MF}\bar{W}/2 + {}^{ln}_{MF}\bar{W}/2$ from a macrofracture of dipset m , existing macrofracture.

If all macrofractures in set I have the same stress shadow width, we can define an “exclusion zone” around each of the existing macrofractures, with a width equal to twice the macrofracture stress shadow width (i.e. a width of $l_{MF}\bar{W}$ on each side of the fracture, giving a total width of $2l_{MF}\bar{W}$, since the macrofracture stress shadow has a width equal to $l_{MF}\bar{W}/2$ on each side of the fracture). No microfractures lying within this exclusion zone will be able to form macrofractures, even though the zone may contain some active microfractures. Calculating the total volume of the macrofracture exclusion zone, χ_{MF}^I , is more complicated than calculating the total stress shadow volume, since the exclusion zones can partially overlap. The exact value of χ_{MF}^I is controlled by the spatial distribution of the macrofractures, characterised by the cumulative macrofracture spacing distribution function $S(x)$, which follows an exponential form.

The situation is even more complicated if the fracture set contain multiple dipsets. Unlike the total stress shadow volume, the volume of the exclusion zone “seen” by a fracture is dependent on the dipset of that fracture; i.e. the exclusion zone volume for a Mode 1 fracture nucleating in a given fracture network will be different to the exclusion zone volume for a Mode 2 fracture nucleating in the same fracture network. This is because the total volume in which a macrofracture will experience stress shadow interaction is dependent on the width of its own stress shadow as well as those of the other fractures: a fracture with a wide stress shadow will experience stress shadow interaction at a greater distance than a fracture with a narrow stress shadow, so it will “see” a larger exclusion zone.

We can use the methodology described in Section 2.1 to derive an expression for the total exclusion zone volume χ_{MF}^{Im} applicable to set I macrofractures of dipset m. To do this, we must first rank the fracture dipsets in order of stress shadow width, from largest to smallest. Hence if the Mode 1 stress shadows are wider (i.e. $W_1 > W_2$), then we can specify $m=1$ for Mode 1 fractures and $m=2$ for Mode 2 fractures. However if the Mode 2 stress shadows are wider (i.e. $W_2 > W_1$), then we can specify $m=1$ for Mode 2 fractures and $m=2$ for Mode 1 fractures. Note that, since the Mode 2 stress shadow width is dependent on the mechanical properties of the fractures, this may vary in different scenarios.

The effective cumulative microfracture activation probability for a set I macrofracture of dipset m, $\theta'_{Im,N}$ is then given by:

$$[57] \quad \theta'_{Im,N} = \left(1 - \chi_{MF}^{Im}(t_N)\right) = \begin{cases} \frac{A_1}{B_1} & \text{for } m = 1 \\ \frac{A_1}{B_1} + \frac{A_2}{B_2} - \frac{A_1}{B_2} & \text{for } m = 2 \\ \frac{A_1}{B_1} + \sum_{s=2}^m \left[\frac{A_s}{B_s} - \frac{A_{s-1}}{B_s} \right] & \text{for } m > 2 \end{cases}$$

The cumulative macrofracture spacing distribution parameters A_1 , A_2 , B_1 , B_2 , etc. must be determined numerically by recalculating their values at the end of each timestep, based on their previous values and on the increase in total mean linear density of each macrofracture dipset during the timestep.

2.3 Probability of macrofracture deactivation due to stress shadow interaction

Macrofractures may also cease propagating due to stress shadow interaction. As described in Section 5.1.3.1 of Welch et al. (2020), this will occur when the tip of an active macrofracture propagates into the stress shadow surrounding the tip of another macrofracture (which may be active or static), as illustrated schematically in Figure 3.

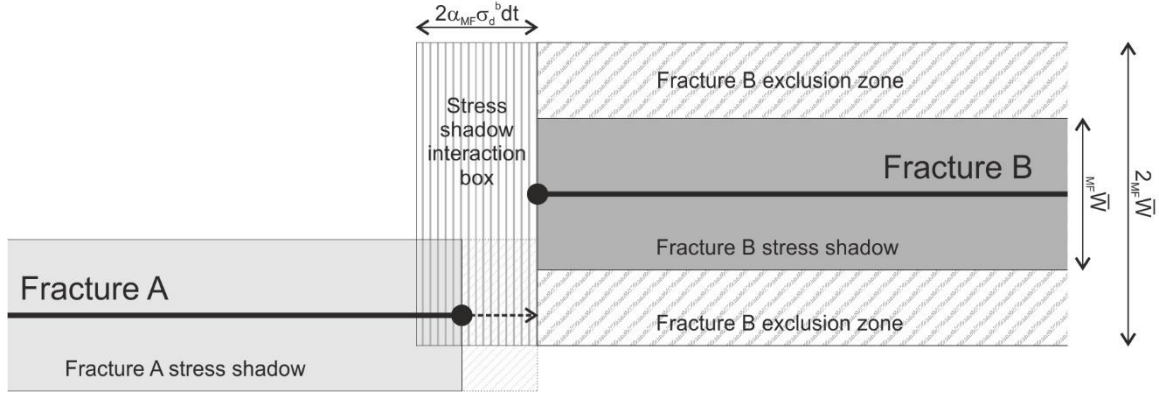


Figure 3: Schematic illustration of stress shadow interaction between two layer-bound macrofractures, viewed from above. Stress shadow interaction between fractures A and B will occur within the short time interval dt if the propagating tip of fracture A lies within the stress shadow interaction box shown around the tip of fracture B.

If we consider a very short time interval dt , so that the increase in the total macrofracture exclusion zone volume $d\chi_{MF}^I$ during dt is small, then the tip of any active half-macrofracture in set I (e.g. fracture A in Figure 3) will interact with the stress shadow around another active half-macrofracture tip, also in set I but propagating in the opposite direction (e.g. fracture B in Figure 3), if tip A lies within a box projected ahead of tip B, labelled the “stress shadow interaction box” on Figure 3. An expression for the volume of the stress shadow interaction box, and hence for the probability of macrofracture stress shadow interaction, is derived in Section 5.1.3.1 of Welch et al. (2020). However this assumes that all set I macrofractures have the same stress shadow width.

If the propagating fracture set I contains multiple dipsets, we must modify the equations given by Welch et al. (2020). We start by deriving a version of the ξ function for a dipset m macrofracture tip propagating towards a dipset n macrofracture:

$$[58] \quad \xi_{Im,n}(t) = \frac{1}{2}\alpha_{MF}^{Im}\sigma_d^b \left(\left(1 + \frac{I_m\sigma_d^b}{I_m\sigma_d^b} \right) \frac{aInP_{30}}{MF}(0,t) + {}^{Im,In}\zeta_{MF}^{sInI}P_{30}(0,t) + {}^{Im,J}\zeta_{MF}^{sInJ}P_{30}(0,t) \right)$$

$aIn_{MF}P_{30}$, $sInI_{MF}P_{30}$ and $sInJ_{MF}P_{30}$ represent the volumetric density of half-macrofracture tips of dipset n that are currently active, deactivated due to stress shadow interaction, and deactivated due to intersection respectively. ${}^{Im,In}\zeta$ and ${}^{Im,J}\zeta$ are the modifiers to take account of obstruction effects at the static half-macrofracture tips; again we can use approximate values of ${}^{Im,In}\zeta \approx 0.25$ and ${}^{Im,J}\zeta \approx 0$.

We can now derive a version of the ${}^I_{IB}P_{32}$ function for a propagating dipset m fracture tip:

$$[59] \quad {}^I_{IB}P_{32}(t + dt) = \sum_{Dipset\ n} \left[\left({}^I_{MF}P_{32}(t) + h\xi_{Im,n}(t)dt \right) \right]$$

$$\Rightarrow \frac{d^I_{IB}P_{32}}{dt} = \sum_{Dipset\ n} [h\xi_{Im,n}(t)]$$

As we noted in Section 2.2.2, the volume of the exclusion zone “seen” by a propagating fracture tip is dependent on the fracture dipset. We can derive approximate expressions for the total clear zone volume seen by a dipset m fracture, $1 - \chi_{MF}^{Im,N-1}$, and the rate of change of the exclusion zone volume seen by dipset m fractures when we add dipset n fractures, $d\chi_{MF}^{Im}/d^I_{IB}P_{32,N-1}$, as described in Section 2.1.3:

$$[60] \quad 1 - \chi_{MF}^{Im}(t) \approx \frac{A_{1,N-1}}{B_{1,N-1}} + \sum_{s=2}^q \left[\frac{\frac{A_{s,N-1}}{B_{s,N-1}} - \frac{A_{s-1,N-1} + (C_{s,N-1} - C_{s-1,N-1})}{B_{s,N-1}}}{-(C_{s,N-1} - C_{s-1,N-1})(W_{s-1} - W_q)} \right]$$

and

$$[61] \quad \frac{d\chi_{MF}^{Im}}{dI_{BP32}^{Im}} = \frac{2 \frac{A_{1,N-1}}{B_{1,N-1}} (1 - e^{-B_{1,N-1} W_n}) + A_{1,N-1} (W_1 - W_n)}{\left(A_{1,N-1} + B_{1,N-1} \sum_{s=2}^n \left[\frac{A_{s,N-1}}{B_{s,N-1}} - \frac{A_{s-1,N-1} + (C_{s,N-1} - C_{s-1,N-1})}{B_{s,N-1}} \right] - (C_{s,N-1} - C_{s-1,N-1})(W_{s-1} - W_q) \right)} \\ + \sum_{s=2}^m \left[\left(\frac{A_{s,N-1} - (A_{s-1,N-1} + (C_{s,N-1} - C_{s-1,N-1}))}{B_{s,N-1}^2} \right) \frac{d B_{s,N-1}}{d I_{BP32}^{Im}} - \frac{1}{B_{s,N-1}} \left(\frac{d A_{s,N-1}}{d I_{BP32}^{Im}} - \frac{d A_{s-1,N-1}}{d I_{BP32}^{Im}} \right) \right]$$

Therefore

$$[62] \quad F_{Im,I,N} dt = \sum_{Dipset\ n} \left[\frac{\frac{d\chi_{MF}^{Im}}{d I_{BP32}^{Im}} \frac{d I_{BP32}^{Im}}{dt} dt}{1 - \chi_{MF,N-1}^{Im}} \right]$$

and thus

$$[63] \quad F_{Im,I,N} = \sum_{Dipset\ n} \left[\frac{\frac{d\chi_{MF}^{Im}}{d I_{BP32}^{Im}}}{1 - \chi_{MF,N-1}^{Im}} h \xi_{Im,n,N-1} \right]$$

Inserting [59], [60] and [61] into [63] gives:

$$[64] \quad F_{Im,I,N} = \sum_{Dipset\ n} \left[\left(\frac{2 \frac{A_{1,N-1}}{B_{1,N-1}} (1 - e^{-B_{1,N-1} W_n}) + A_{1,N-1} (W_1 - W_n)}{B_{1,N-1} \left(\frac{A_{1,N-1}}{B_{1,N-1}} + \sum_{s=2}^n \left[\frac{A_{s,N-1}}{B_{s,N-1}} - \frac{A_{s-1,N-1} + (C_{s,N-1} - C_{s-1,N-1})}{B_{s,N-1}} \right] - (C_{s,N-1} - C_{s-1,N-1})(W_{s-1} - W_q) \right)} \right) \left(\frac{A_{1,N-1}}{B_{1,N-1}} + \sum_{s=2}^n \left[\frac{A_{s,N-1}}{B_{s,N-1}} - \frac{A_{s-1,N-1} + (C_{s,N-1} - C_{s-1,N-1})}{B_{s,N-1}} \right] - (C_{s,N-1} - C_{s-1,N-1})(W_{s-1} - W_q) \right) \right) \\ + \sum_{s=2}^m \left[\left(\frac{A_{s,N-1} - (A_{s-1,N-1} + (C_{s,N-1} - C_{s-1,N-1}))}{B_{s,N-1}^2} \right) \frac{d B_{s,N-1}}{d I_{BP32}^{Im}} - \frac{1}{B_{s,N-1}} \left(\frac{d A_{s,N-1}}{d I_{BP32}^{Im}} - \frac{d A_{s-1,N-1}}{d I_{BP32}^{Im}} \right) \right] \right] h \xi_{Im,n,N-1}$$

where $\psi_{MF,I,N}$ represents the total stress shadow volume of all half-macrofractures of set I during timestep **N**, which is assumed to be constant throughout **N**. The calculation can be simplified using the approximation

$$[65] \quad \frac{d\chi_{MF}^{Im}}{d I_{BP32}^{Im}} \approx \frac{W_m + W_n}{2 W_m} \frac{d\chi_{MF}^{Im}}{d I_{BP32}^{Im}}$$

so that [63] can be written as

$$[66] \quad F_{Im,I,N} = \frac{\frac{d\chi_{MF}^{Im}}{d I_{BP32}^{Im}}}{1 - \chi_{MF,N-1}^{Im}} \sum_{Dipset\ n} \left[\frac{W_m + W_n}{2 W_m} h \xi_{Im,n,N-1} \right]$$

[66] is easier to evaluate than [63] as we only need to calculate one differential $d\chi_{MF}/d I_{BP32}$ for each fracture dipset.

Again we can calculate expressions for the cumulative macrofracture activation probability and the mean macrofracture deactivation probability for stress shadow interaction during a timestep **N** from [64]:

$$[67] \quad \Phi_{Im,I,N} = e^{-F_{Im,I,N} \Delta t_N}$$

and

$$[68] \quad \overline{F_{Im,I,N}} = \frac{1}{\Delta t_N} (1 - e^{-F_{Im,I,N} \Delta t_N})$$

2.4 Probability of macrofracture deactivation due to intersection

Macrofractures will also cease propagating if they intersect a perpendicular- or oblique-striking macrofracture from another fracture set. Although this process is not directly related to the fracture stress shadows, the probability of intersecting a macrofracture from another set is dependent on the spatial distribution of those macrofractures, which in turn is controlled by the width of their stress shadows. This is described in Section 5.1.3.2 of Welch et al. (2020), and illustrated in *Figure 4*.

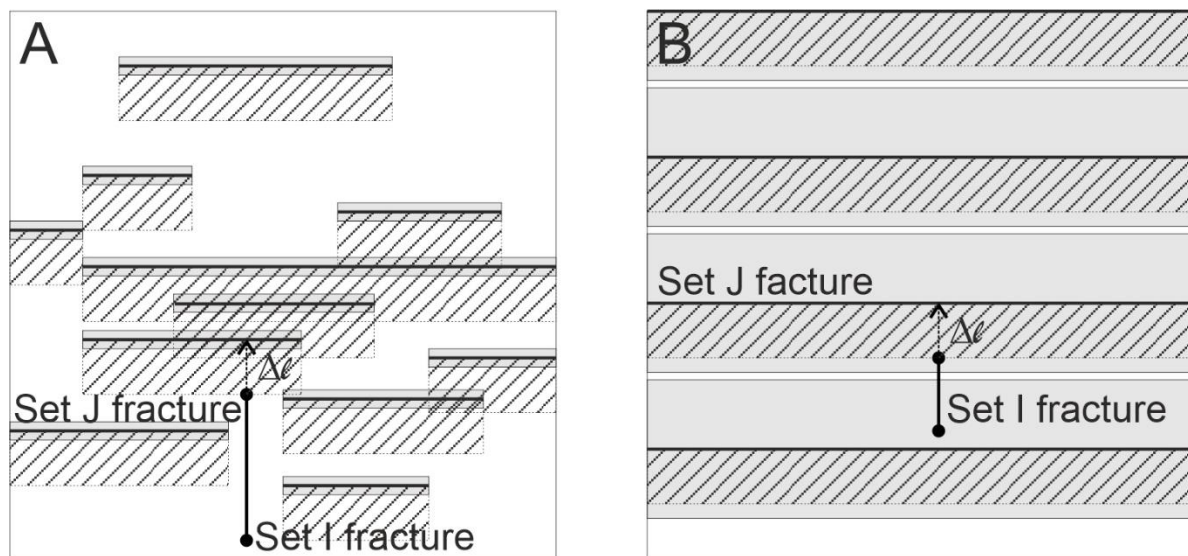


Figure 4: Schematic illustration of a N-propagating fracture tip (Fracture I) intersecting a set of E-W striking fractures (set J) surrounded by narrow stress shadows (A) or by wide stress shadows (B), shown by grey boxes. The striped boxes on the S side of the E-W striking fractures show the intersection boxes around each fracture; the N-propagating fracture tip will intersect an E-W striking fracture within a time period Δt if it lies within an intersection box. Note that with wide stress shadows (B), the intersection boxes lie entirely within the stress shadows and cannot overlap, but with narrow stress shadows (A) the intersection boxes extend beyond the stress shadows and can partially overlap.

If we take a short, but not infinitesimal, time period Δt , the propagating fracture tip (fracture I in *Figure 4*) will intersect a fracture from set J during Δt if that fracture lies within the distance that the set I half-macrofracture tip can propagate in the time period Δt (i.e. $\Delta \ell = \alpha_{MF}^I \sigma_d^b \Delta t$). We can therefore define a set of “intersection boxes”, extending towards the propagating fracture from each of the set J fractures, with width $\Delta \ell$. An expression for the total volume of all intersection boxes as a proportion of the total rock volume, and hence for the probability that the propagating fracture tip A will intersect a fracture from set J during the time period Δt , is derived in Section 5.1.3.1 of Welch et al. (2020). However this assumes that all set J macrofractures have the same stress shadow width.

If fracture set J contains multiple dipsets, whose stress shadows may have different widths but still cannot overlap each other, then we can get an approximation of the total intersection box volume using the one-sided proximity zone formula for varying stress shadow widths, as described in Section 2.1.4. A general version of this “one-sided proximity zone” formula, which gives the total

intersection box volume $\chi'(\Delta\ell)$ in all circumstances, is given by [35]. From this we can derive the cumulative macrofracture activation probability for fracture intersection $\Phi_{Im,J,N}$, which represents the probability that an active set I macrofracture tip of dipset m will not be deactivated due to intersection with a set J macrofracture during timestep **N**:

$$[69] \quad \Phi_{Im,J,N} \approx \begin{cases} 1 - \frac{J}{MF} P_{32}(t_{N-1}) \Delta\ell_{m,N} & \text{if } \Delta\ell_{m,N} \leq \frac{\bar{J}}{MF} \bar{W} \\ \left(\frac{(\mathbb{A}_{r,N-1}^J + \mathbb{C}_{r,N-1}^J)}{\mathbb{B}_{r,N-1}^J} e^{-\mathbb{B}_{r,N-1}^J \left((\Delta\ell_N - \frac{\bar{J}}{MF} \bar{W}) - \frac{J_r}{MF} \bar{W} \right)} - \frac{(\mathbb{A}_{r-1,N-1}^J + \mathbb{C}_{r,N-1}^J)}{\mathbb{B}_{r,N-1}^J} \right) & \text{if } \Delta\ell_{m,N} > \frac{\bar{J}}{MF} \bar{W} \\ -\mathbb{C}_{r,N-1}^J \left(\frac{J_{r-1}}{MF} \bar{W} - (\Delta\ell_N - \frac{\bar{J}}{MF} \bar{W}) \right) + \sum_{s=1}^{r-1} \left[\frac{\mathbb{A}_{s,N-1}^J - \mathbb{A}_{s-1,N-1}^J}{\mathbb{B}_{s,N-1}^J} \right] & \end{cases}$$

$$[70] \quad \Delta\ell_{m,N} = \alpha_{MF}^{Im} \sigma_{dN}^b \Delta t_N$$

$$[71] \quad \frac{\bar{J}}{MF} \bar{W} = \frac{\sum_{Dipset\ n} [\psi_{MF,N-1}^J]}{\sum_{Dipset\ n} [\frac{J_n}{MF} P_{32}(t_{N-1})]} = \frac{\psi_{MF,N-1}^J}{\frac{J}{MF} P_{32}(t_{N-1})}$$

$$[72] \quad \frac{J}{MF} P_{32}(N) = \sum_{Dipset\ n} \left[\frac{J_n}{MF} P_{32}(t_{N-1}) \right]$$

and r is the fracture dipset for which $J_r \bar{W} \leq \Delta\ell_{m,N} - \frac{\bar{J}}{MF} \bar{W} < J_{r-1} \bar{W}$

Note that [69] is only an approximation as it does not fully account for the potential intersection box overlap when fractures of different dipsets lie adjacent to each other.

The mean probability of macrofracture deactivation due to intersection during timestep **N** can now be calculated using [69]. This gives

$$[73] \quad \overline{F_{Im,J,N}} \approx \begin{cases} \frac{1}{\Delta t_N} - \frac{J}{MF} P_{32}(t_{N-1}) \alpha_{MF}^{Im} \sigma_d^b & \text{if } \alpha_{MF}^{Im} \sigma_{dN}^b \Delta t_N \leq \frac{\bar{J}}{MF} \bar{W} \\ \frac{1}{\Delta t_N} \left(\frac{(\mathbb{A}_{r,N-1}^J + \mathbb{C}_{r,N-1}^J)}{\mathbb{B}_{r,N-1}^J} e^{-\mathbb{B}_{r,N-1}^J \left((\Delta\ell_N - \frac{\bar{J}}{MF} \bar{W}) - \frac{J_r}{MF} \bar{W} \right)} - \frac{(\mathbb{A}_{r-1,N-1}^J + \mathbb{C}_{r,N-1}^J)}{\mathbb{B}_{r,N-1}^J} \right) & \text{if } \alpha_{MF}^{Im} \sigma_{dN}^b \Delta t_N > \frac{\bar{J}}{MF} \bar{W} \\ -\mathbb{C}_{r,N-1}^J \left(\frac{J_{r-1}}{MF} \bar{W} - (\Delta\ell_N - \frac{\bar{J}}{MF} \bar{W}) \right) + \sum_{s=1}^{r-1} \left[\frac{\mathbb{A}_{s,N-1}^J - \mathbb{A}_{s-1,N-1}^J}{\mathbb{B}_{s,N-1}^J} \right] & \end{cases}$$

We can also derive an expression for the instantaneous probability of macrofracture deactivation due to intersection from [69]. To do this, we must first calculate $d\Phi_{Im,J,N}/dt$. For a fracture set with multiple dipsets, $d\Phi_{Im,J,N}/dt$ can be calculated by combining the differentials of [69] and [70]. For $\alpha_{MF}^{Im} \sigma_d^b t_n \leq \frac{J}{MF} \bar{W}$, this is quite simple and gives

$$[74] \quad \frac{d\Phi_{Im,J,N}}{dt} = -\frac{J}{MF} P_{32}(t_{N-1}) \alpha_{MF}^{Im} \sigma_{dN}^b$$

For $\alpha_{MF}^{Im} \sigma_d^b t_n \leq \frac{J}{MF} \bar{W}$, if we assume that the values of A and B are constant throughout timestep **N**, this gives

$$[75] \quad \frac{d\Phi_{Im,J,N}}{dt} = -\alpha_{MF}^{Im} \sigma_{dN}^b (\mathbb{A}_{r,N-1} + \mathbb{C}_{r,N-1}) e^{-\mathbb{B}_{r,N-1} \left((\alpha_{MF}^{Im} \sigma_{dN}^b t_n - \frac{\bar{J}}{MF} \bar{W}) - \frac{J_r}{MF} \bar{W} \right)} + \alpha_{MF}^{Im} \sigma_{dN}^b \mathbb{C}_{r,N-1}$$

Therefore

$$[76] \quad F_{Im,J}(t_n) = \frac{-\frac{d\Phi_{Im,J}}{dt}}{\Phi_{Im,J}(t)}$$

$$\approx \begin{cases} \frac{\frac{M_{32}^{IP}(N)\alpha_{MF}^{Im}\sigma_{dN}^b}{1 - \frac{M_{32}^{IP}(N)\alpha_{MF}^{Im}\sigma_{dN}^b}{t_n}}}{\alpha_{MF}^{Im}\sigma_{dN}^b (\mathbb{A}_{r,N-1} + \mathbb{C}_{r,N-1}) e^{-\mathbb{B}_{r,N-1} \left((\alpha_{MF}^{Im}\sigma_{dN}^b t_n - \frac{J^r \bar{W}}{M_F}) - \frac{J^r \bar{W}}{M_F} \right)} - \alpha_{MF}^{Im}\sigma_{dN}^b \mathbb{C}_{r,N-1}} & \text{if } t_n \leq \frac{\frac{J^r \bar{W}}{M_F}}{\alpha_{MF}^{Im}\sigma_{dN}^b} \\ \frac{\left(\frac{(\mathbb{A}_{r,N-1}^J + \mathbb{C}_{r,N-1}^J)}{\mathbb{B}_{r,N-1}^J} e^{-\mathbb{B}_{r,N-1}^J \left((\Delta \ell_N - \frac{J^r \bar{W}}{M_F}) - \frac{J^r \bar{W}}{M_F} \right)} - \frac{(\mathbb{A}_{r,N-1}^J + \mathbb{C}_{r,N-1}^J)}{\mathbb{B}_{r,N-1}^J} \right)}{\left(-\mathbb{C}_{r,N-1}^J \left(\frac{J^{r-1} \bar{W}}{M_F} - (\Delta \ell_N - \frac{J^r \bar{W}}{M_F}) \right) + \sum_{s=1}^{r-1} \left[\frac{\mathbb{A}_{s,N-1}^J - \mathbb{A}_{s-1,N-1}^J}{\mathbb{B}_{s,N-1}^J} \right] \right)} & \text{if } t_n > \frac{\frac{J^r \bar{W}}{M_F}}{\alpha_{MF}^{Im}\sigma_{dN}^b} \end{cases}$$

Since the instantaneous probability of macrofracture intersection F_{IJ} is dependent on half-macrofracture length, strictly speaking [69] is valid only for half-macrofractures with zero length at the start of the timestep, and must be modified for pre-existing half-macrofractures with a finite initial length.

We can derive an expression for the cumulative macrofracture activation probability for fracture intersection of a half-macrofracture with an initial length ℓ_0 , ${}^{\ell_0}\Phi_{Im,J}$, by considering the case of a zero length half-macrofracture propagating in two stages, the first from length 0 to length ℓ_0 , and the second from length ℓ_0 to length $\ell_0 + \Delta \ell$. In this case, ignoring deactivation due to other factors, there are three possible outcomes:

1. The half-macrofracture could be deactivated due to intersection during the first stage, before reaching length ℓ_0 . The probability of this is given by the total intersection box volume $\chi'(\ell_0)$, which can be calculated using the appropriate “one-sided proximity zone” formula from Section 2.1.4.
2. The half-macrofracture could be deactivated due to intersection during the second stage, after reaching length ℓ_0 but before reaching length $\ell_0 + \Delta \ell$. This represents the inverse of the cumulative macrofracture activation probability for fracture intersection of a half-macrofracture with an initial length ℓ_0 , $1 - {}^{\ell_0}\Phi_{Im,J}$.
3. The half-macrofracture could reach length $\ell_0 + \Delta \ell$ without being deactivated. The probability of this is given by the inverse of the total intersection box volume, $1 - \chi'(\ell_0 + \Delta \ell)$, which can also be calculated using the appropriate “one-sided proximity zone” formula from Section 2.1.4.

Since these are the only possible outcomes, we can say that

$$\begin{aligned} [77] \quad \chi'(\ell_0) + (1 - {}^{\ell_0}\Phi_{Im,J}) + (1 - \chi'(\ell_0 + \Delta \ell)) &= 1 \\ \Rightarrow 1 - {}^{\ell_0}\Phi_{Im,J} &= \chi'(\ell_0 + \Delta \ell) - \chi'(\ell_0) \end{aligned}$$

However if we now consider a half-macrofracture that has already reached length ℓ_0 , then the first outcome is ruled out, and the probability of the second outcome is now given by the original probability of the second outcome, divided by the inverse of the original probability of the first outcome. Therefore

$$\begin{aligned} [78] \quad 1 - {}^{\ell_0}\Phi_{Im,J} &= \frac{\chi'(\ell_0 + \Delta \ell) - \chi'(\ell_0)}{1 - \chi'(\ell_0)} = 1 - \frac{1 - \chi'(\ell_0 + \Delta \ell)}{1 - \chi'(\ell_0)} \\ \Rightarrow {}^{\ell_0}\Phi_{Im,J} &= \frac{1 - \chi'(\ell_0 + \Delta \ell)}{1 - \chi'(\ell_0)} \end{aligned}$$

We can derive a reasonable approximation for the cumulative macrofracture activation probability for fracture intersection of a set of half-macrofractures during a timestep \mathbf{N} , by substituting the mean length of active half-macrofractures in the set, $\bar{\ell}$, for ℓ_0 in [78]. Thus

$$[79] \quad \Phi_{I,J,N} = \frac{1-\chi'(\overline{\ell} + \Delta \ell_N)}{1-\chi'(\overline{\ell})}$$

where $1-\chi'(\overline{\ell})$ and $1-\chi'(\overline{\ell} + \Delta \ell_N)$ are calculated using the “one-sided proximity zone” formula from Section 2.1.4, and $\overline{\ell}$ can be calculated from the volumetric and mean linear densities of half-macrofractures at the end of the previous timestep:

$$[80] \quad \overline{\ell} = \frac{\underline{\text{MFP}}_{32}^{aP}(0, t_{N-1})}{\underline{\text{MFP}}_{30}^{aP}(0, t_{N-1})h}$$

3 Calculating the mean length of residual half-macrofractures

In Section 5.4.3 of Welch et al. (2020) we derive expressions for the mean length that a residual half-macrofracture will reach before being deactivated, $\bar{\ell}$, by integrating the the cumulative macrofracture activation probability with respect to macrofracture length ℓ :

$$[81] \quad \bar{\ell} = \int_0^\infty -\frac{d\Phi}{d\ell} \ell d\ell$$

Integrating by parts, [81] can be rearranged to give

$$[82] \quad \bar{\ell} = \int_0^\infty \Phi d\ell - [\Phi(\ell)\ell]_0^\infty$$

A similar procedure is used to calculate the mean linear density of residual active half-macrofractures of length ℓ_N or greater:

$$[83] \quad \bar{\ell}(\ell_N)\Phi(\ell_N) = \int_{\ell_N}^\infty \Phi d\ell - [\Phi(\ell)\ell]_{\ell_N}^\infty$$

The resulting expressions, given by equations 5.99 and 5.101 of Welch et al. (2020), are essential for calculating the increase in the mean linear density of residual macrofractures during any given timestep. However, as explained in the text, equations 5.99 and 5.101 are approximations as they assume that the instantaneous frate of macrofracture deactivation is constant, i.e. they do not take into account the effect of the spatial distribution of perpendicular macrofractures on the probability of macrofracture intersection.

We can calculate more precise values for both the true mean residual half-macrofracture length and the mean linear residual active half-macrofracture density by splitting Φ into two components $\Phi_{||}$ and Φ_{\perp} , and inserting the semi-regular version of $\Phi_{||}$ into [81].

3.1 Calculating $\bar{\ell}$ with uniform stress shadow width

If all fractures in set J have the same stress shadow width,

$$[84] \quad \Phi_N(\Delta\ell_N) = \Phi_{I,J,N}(\Delta\ell_N)\Phi_{I,I,N}(\Delta\ell_N)$$

$$= \begin{cases} \left(1 - \frac{J}{MF} P_{32} \Delta\ell_N\right) e^{-\frac{F_{I,I,N}\Delta\ell_N}{\alpha_{MF}\sigma_{dN}^b}} & \text{if } \Delta\ell_N \leq \frac{J}{MF} \bar{W} \\ \left(\frac{\frac{J}{MF} P_{32} + \frac{C_{2,N-1}^J}{\mathbb{B}_{2,N-1}^J} e^{-\frac{F_{I,I,N}\Delta\ell_N}{\alpha_{MF}\sigma_{dN}^b}} - \frac{C_{2,N-1}^J}{\mathbb{B}_{2,N-1}^J} - \frac{C_{2,N-1}^J}{\mathbb{B}_{2,N-1}^J} (2\frac{J}{MF} \bar{W} - \Delta\ell_N) + \left(\frac{\frac{J}{MF} - 1}{\mathbb{B}_{N-1}^J} - \frac{\frac{J}{MF} - 1}{\mathbb{B}_{2,N-1}^J}\right) e^{-\frac{F_{I,I,N}\Delta\ell_N}{\alpha_{MF}\sigma_{dN}^b}}\right) e^{-\frac{F_{I,I,N}\Delta\ell_N}{\alpha_{MF}\sigma_{dN}^b}} & \text{if } \frac{J}{MF} \bar{W} < \Delta\ell_N \leq 2\frac{J}{MF} \bar{W} \\ \frac{\frac{J}{MF} - 1}{\mathbb{B}_{N-1}^J} e^{-\frac{F_{I,I,N}\Delta\ell_N}{\alpha_{MF}\sigma_{dN}^b}} e^{-\frac{F_{I,I,N}\Delta\ell_N}{\alpha_{MF}\sigma_{dN}^b}} & \text{if } \Delta\ell_N > 2\frac{J}{MF} \bar{W} \end{cases}$$

and

$$[85] \quad \int \Phi d\ell - [\Phi(\ell)\ell] = [\Phi_{I,J,N} \int \Phi_{I,I,N} d\ell] - \int \left(\frac{d\Phi_{I,J,N}}{d\ell} \int \Phi_{I,I,N} d\ell\right) d\ell - [\Phi_{I,J,N} \Phi_{I,I,N} \ell]$$

where

$$[86] \quad \int \Phi_{I,I,N} d\ell = -\left[\frac{\alpha_{MF}\sigma_{dN}^b}{F_{I,I,N}} e^{-\frac{F_{I,I,N}\ell}{\alpha_{MF}\sigma_{dN}^b}}\right]$$

Therefore when $\ell \leq \frac{J}{MF} \bar{W}$,

$$[87] \quad \int \Phi d\ell - [\Phi(\ell)\ell] = -\left[\left(1 - \frac{J}{MF} P_{32} \ell\right) \frac{\alpha_{MF}\sigma_{dN}^b}{F_{I,I,N}} e^{-\frac{F_{I,I,N}\ell}{\alpha_{MF}\sigma_{dN}^b}}\right] - \int \frac{J}{MF} P_{32} \frac{\alpha_{MF}\sigma_{dN}^b}{F_{I,I,N}} e^{-\frac{F_{I,I,N}\ell}{\alpha_{MF}\sigma_{dN}^b}} d\ell$$

$$\begin{aligned}
& - \left[\left(\ell - \frac{J}{\underline{M}_F} P_{32} \ell^2 \right) e^{-\frac{F_{I,I,N} \ell}{\alpha_{MF} \sigma_{dN}^b}} \right] \\
& = - \left[\left(1 - \frac{J}{\underline{M}_F} P_{32} \ell \right) \frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} e^{-\frac{F_{I,I,N} \ell}{\alpha_{MF} \sigma_{dN}^b}} \right] + \left[\frac{J}{\underline{M}_F} P_{32} \left(\frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} \right)^2 e^{-\frac{F_{I,I,N} \ell}{\alpha_{MF} \sigma_{dN}^b}} \right] \\
& \quad - \left[\left(\ell - \frac{J}{\underline{M}_F} P_{32} \ell^2 \right) e^{-\frac{F_{I,I,N} \ell}{\alpha_{MF} \sigma_{dN}^b}} \right] \\
& = \left[\left(\frac{J}{\underline{M}_F} P_{32} \ell^2 + \left(\frac{\frac{J}{\underline{M}_F} P_{32} \alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} - 1 \right) \ell + \left(\frac{J}{\underline{M}_F} P_{32} \left(\frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} \right)^2 - \frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} \right) \right) e^{-\frac{F_{I,I,N} \ell}{\alpha_{MF} \sigma_{dN}^b}} \right]
\end{aligned}$$

and hence

$$\begin{aligned}
[88] \quad \Rightarrow \int_0^{J_{MF} \bar{W}} -\frac{d\Phi}{d\ell} \ell d\ell & = \left(\frac{J}{\underline{M}_F} P_{32} \frac{J}{\underline{M}_F} \bar{W}^2 + \left(\frac{\frac{J}{\underline{M}_F} P_{32} \frac{J}{\underline{M}_F} \bar{W} \alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} - \frac{J}{\underline{M}_F} \bar{W} \right) \right) e^{-\frac{F_{I,I,N} J_{MF} \bar{W}}{\alpha_{MF} \sigma_{dN}^b}} \\
& \quad + \left(\frac{J}{\underline{M}_F} P_{32} \left(\frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} \right)^2 - \frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} \right) e^{-\frac{F_{I,I,N} J_{MF} \bar{W}}{\alpha_{MF} \sigma_{dN}^b}} \\
& \quad - \left(\frac{J}{\underline{M}_F} P_{32} \left(\frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} \right)^2 - \frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} \right) \\
& = \left(\psi_{MF MF}^J \bar{W} + \left(\frac{\psi_{MF}^J \alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} - \frac{J}{\underline{M}_F} \bar{W} \right) \right) e^{-\frac{F_{I,I,N} J_{MF} \bar{W}}{\alpha_{MF} \sigma_{dN}^b}} - \left(\frac{J}{\underline{M}_F} P_{32} \left(\frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} \right)^2 - \frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} \right)
\end{aligned}$$

except when $F_{I,I,N} \rightarrow 0$, in which case

$$\begin{aligned}
[89] \quad \Phi_N(\Delta \ell_N) & \rightarrow \Phi_{I,J,N}(\Delta \ell_N) = 1 - \frac{J}{\underline{M}_F} P_{32} \Delta \ell_N \\
\Rightarrow \int_0^{J_{MF} \bar{W}} -\frac{d\Phi}{d\ell} \ell d\ell & \rightarrow \int_0^{J_{MF} \bar{W}} \frac{J}{\underline{M}_F} P_{32} \ell d\ell = \left[\frac{J}{\underline{M}_F} P_{32} \frac{\ell^2}{2} \right]_0^{J_{MF} \bar{W}} = \frac{\frac{J}{\underline{M}_F} P_{32} \frac{J_{MF} \bar{W}^2}{2}}{2} = \frac{1}{2} \psi_{MF MF}^J \bar{W}
\end{aligned}$$

When $J_{MF} \bar{W} < \ell \leq 2 J_{MF} \bar{W}$,

$$\begin{aligned}
[90] \quad \frac{d\Phi_{I,J,N}}{d\ell} & = - \left(\frac{J}{\underline{M}_F} P_{32} + \mathbb{C}_{2,N-1}^J \right) e^{-\mathbb{B}_{2,N-1}^J (\ell - J_{MF} \bar{W})} + \mathbb{C}_{2,N-1}^J \\
& \Rightarrow \int \Phi d\ell - [\Phi(\ell) \ell] = \\
& - \left[\left(\frac{\frac{J}{\underline{M}_F} P_{32} + \mathbb{C}_{2,N-1}^J}{\mathbb{B}_{2,N-1}^J} e^{-\mathbb{B}_{2,N-1}^J (\ell - J_{MF} \bar{W})} - \frac{\mathbb{C}_{2,N-1}^J}{\mathbb{B}_{2,N-1}^J} - \mathbb{C}_{2,N-1}^J (2 J_{MF} \bar{W} - \ell) \right) \left(\frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} e^{-\frac{F_{I,I,N} \ell}{\alpha_{MF} \sigma_{dN}^b}} + \ell e^{-\frac{F_{I,I,N} \ell}{\alpha_{MF} \sigma_{dN}^b}} \right) \right. \\
& \quad \left. + \left(\frac{\mathbb{A}_{N-1}^J}{\mathbb{B}_{N-1}^J} - \frac{\mathbb{A}_{N-1}^J}{\mathbb{B}_{2,N-1}^J} \right) \right] \\
& \quad + \int \left(- \left(\frac{J}{\underline{M}_F} P_{32} + \mathbb{C}_{2,N-1}^J \right) e^{-\mathbb{B}_{2,N-1}^J (\ell - J_{MF} \bar{W})} + \mathbb{C}_{2,N-1}^J \right) \frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} e^{-\frac{F_{I,I,N} \ell}{\alpha_{MF} \sigma_{dN}^b}} d\ell \\
& = - \left[\left(\frac{\frac{J}{\underline{M}_F} P_{32} + \mathbb{C}_{2,N-1}^J}{\mathbb{B}_{2,N-1}^J} e^{-\mathbb{B}_{2,N-1}^J (\ell - J_{MF} \bar{W})} - \frac{\mathbb{C}_{2,N-1}^J}{\mathbb{B}_{2,N-1}^J} \right) \left(\left(\frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} + \ell \right) e^{-\frac{F_{I,I,N} \ell}{\alpha_{MF} \sigma_{dN}^b}} \right) \right. \\
& \quad \left. - \mathbb{C}_{2,N-1}^J (2 J_{MF} \bar{W} - \ell) + \left(\frac{\mathbb{A}_{N-1}^J}{\mathbb{B}_{N-1}^J} - \frac{\mathbb{A}_{N-1}^J}{\mathbb{B}_{2,N-1}^J} \right) \right] \\
& \quad - \int \left(\frac{J}{\underline{M}_F} P_{32} + \mathbb{C}_{2,N-1}^J \right) e^{-\mathbb{B}_{2,N-1}^J \ell} \frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} e^{-\left(\mathbb{B}_{2,N-1}^J + \frac{F_{I,I,N}}{\alpha_{MF} \sigma_{dN}^b} \right) \ell} d\ell
\end{aligned}$$

$$\begin{aligned}
& + \int \mathbb{C}_{2,N-1}^J \frac{\alpha_{MF\sigma dN}^b}{F_{I,I,N}} e^{-\frac{F_{I,I,N}\ell}{\alpha_{MF\sigma dN}^b}} d\ell \\
& = - \left[\left(\frac{\frac{M_F^{JP_{32} + C_{2,N-1}^J}}{\mathbb{B}_{2,N-1}^J} e^{-\mathbb{B}_{2,N-1}^J(\ell - M_F^J \bar{W})} - \frac{C_{2,N-1}^J}{\mathbb{B}_{2,N-1}^J}}{\mathbb{B}_{2,N-1}^J} \right) \left(\left(\frac{\alpha_{MF\sigma dN}^b}{F_{I,I,N}} + \ell \right) e^{-\frac{F_{I,I,N}\ell}{\alpha_{MF\sigma dN}^b}} \right) \right. \\
& \quad + \left[\frac{\left(\frac{M_F^{JP_{32} + C_{2,N-1}^J}}{\mathbb{B}_{2,N-1}^J} \right) \frac{\alpha_{MF\sigma dN}^b}{F_{I,I,N}}}{\left(\mathbb{B}_{2,N-1}^J + \frac{F_{I,I,N}}{\alpha_{MF\sigma dN}^b} \right)} e^{\mathbb{B}_{2,N-1}^J M_F^J \bar{W}} e^{-\left(\mathbb{B}_{2,N-1}^J + \frac{F_{I,I,N}}{\alpha_{MF\sigma dN}^b} \right) \ell} \right] \\
& \quad \left. - \left[\mathbb{C}_{2,N-1}^J \left(\frac{\alpha_{MF\sigma dN}^b}{F_{I,I,N}} \right)^2 e^{-\frac{F_{I,I,N}\ell}{\alpha_{MF\sigma dN}^b}} \right] \right] \\
& = - \left[\frac{\frac{M_F^{JP_{32} + C_{2,N-1}^J}}{\mathbb{B}_{2,N-1}^J} \left(\ell + \frac{1}{\mathbb{B}_{2,N-1}^J + \frac{F_{I,I,N}}{\alpha_{MF\sigma dN}^b}} \right) e^{\mathbb{B}_{2,N-1}^J M_F^J \bar{W}} e^{-\left(\mathbb{B}_{2,N-1}^J + \frac{F_{I,I,N}}{\alpha_{MF\sigma dN}^b} \right) \ell}}{\mathbb{B}_{2,N-1}^J} + \mathbb{C}_{2,N-1}^J \left(\frac{\alpha_{MF\sigma dN}^b}{F_{I,I,N}} \right)^2 e^{-\frac{F_{I,I,N}\ell}{\alpha_{MF\sigma dN}^b}} \right. \\
& \quad \left. + \left(\left(\frac{A_{N-1}^J}{\mathbb{B}_{N-1}^J} - \frac{A_{N-1}^J}{\mathbb{B}_{2,N-1}^J} \right) - \frac{C_{2,N-1}^J}{\mathbb{B}_{2,N-1}^J} - \mathbb{C}_{2,N-1}^J (2 M_F^J \bar{W} - \ell) \right) \left(\ell + \frac{\alpha_{MF\sigma dN}^b}{F_{I,I,N}} \right) \right]
\end{aligned}$$

and hence

$$\begin{aligned}
[91] \quad \int_{M_F^J \bar{W}}^{2 M_F^J \bar{W}} - \frac{d\Phi}{d\ell} \ell d\ell & = \left(\frac{\frac{M_F^{JP_{32} + C_{2,N-1}^J}}{\mathbb{B}_{2,N-1}^J} \left(M_F^J \bar{W} + \frac{1}{\mathbb{B}_{2,N-1}^J + \frac{F_{I,I,N}}{\alpha_{MF\sigma dN}^b}} \right) + \mathbb{C}_{2,N-1}^J \left(\frac{\alpha_{MF\sigma dN}^b}{F_{I,I,N}} \right)^2}{\mathbb{B}_{2,N-1}^J} \right) e^{-\frac{F_{I,I,N} M_F^J \bar{W}}{\alpha_{MF\sigma dN}^b}} \\
& \quad + \left(\left(\frac{A_{N-1}^J}{\mathbb{B}_{N-1}^J} - \frac{A_{N-1}^J}{\mathbb{B}_{2,N-1}^J} \right) - \frac{C_{2,N-1}^J}{\mathbb{B}_{2,N-1}^J} - \mathbb{C}_{2,N-1}^J M_F^J \bar{W} \right) \left(M_F^J \bar{W} + \frac{\alpha_{MF\sigma dN}^b}{F_{I,I,N}} \right) \\
& \quad - \left(\frac{\frac{M_F^{JP_{32} + C_{2,N-1}^J}}{\mathbb{B}_{2,N-1}^J} \left(2 M_F^J \bar{W} + \frac{1}{\mathbb{B}_{2,N-1}^J + \frac{F_{I,I,N}}{\alpha_{MF\sigma dN}^b}} \right) e^{-\mathbb{B}_{2,N-1}^J M_F^J \bar{W}} + \mathbb{C}_{2,N-1}^J \left(\frac{\alpha_{MF\sigma dN}^b}{F_{I,I,N}} \right)^2}{\mathbb{B}_{2,N-1}^J} \right) e^{-\frac{2 F_{I,I,N} M_F^J \bar{W}}{\alpha_{MF\sigma dN}^b}} \\
& \quad + \left(\left(\frac{A_{N-1}^J}{\mathbb{B}_{N-1}^J} - \frac{A_{N-1}^J}{\mathbb{B}_{2,N-1}^J} \right) - \frac{C_{2,N-1}^J}{\mathbb{B}_{2,N-1}^J} \right) \left(2 M_F^J \bar{W} + \frac{\alpha_{MF\sigma dN}^b}{F_{I,I,N}} \right) \\
& \Rightarrow \int_{M_F^J \bar{W}}^{2 M_F^J \bar{W}} - \frac{d\Phi}{d\ell} \ell d\ell = \left(\frac{\frac{M_F^{JP_{32} + C_{2,N-1}^J}}{\mathbb{B}_{2,N-1}^J} \left(M_F^J \bar{W} + \frac{1}{\mathbb{B}_{2,N-1}^J + \frac{F_{I,I,N}}{\alpha_{MF\sigma dN}^b}} \right) + \mathbb{C}_{2,N-1}^J \left(\frac{\alpha_{MF\sigma dN}^b}{F_{I,I,N}} \right)^2}{\mathbb{B}_{2,N-1}^J} \right) e^{-\frac{F_{I,I,N} M_F^J \bar{W}}{\alpha_{MF\sigma dN}^b}} \\
& \quad + \left(\left(\frac{A_{N-1}^J}{\mathbb{B}_{N-1}^J} - \frac{A_{N-1}^J + C_{2,N-1}^J}{\mathbb{B}_{2,N-1}^J} \right) - \mathbb{C}_{2,N-1}^J M_F^J \bar{W} \right) \left(M_F^J \bar{W} + \frac{\alpha_{MF\sigma dN}^b}{F_{I,I,N}} \right) \\
& \quad - \left(\frac{\frac{A_{N-1}^J + C_{2,N-1}^J}{\mathbb{B}_{2,N-1}^J} \left(2 M_F^J \bar{W} + \frac{1}{\mathbb{B}_{2,N-1}^J + \frac{F_{I,I,N}}{\alpha_{MF\sigma dN}^b}} \right) + \mathbb{C}_{2,N-1}^J \left(\frac{\alpha_{MF\sigma dN}^b}{F_{I,I,N}} \right)^2}{\mathbb{B}_{2,N-1}^J} \right) e^{-\frac{2 F_{I,I,N} M_F^J \bar{W}}{\alpha_{MF\sigma dN}^b}} \\
& \quad + \left(\frac{A_{N-1}^J}{\mathbb{B}_{N-1}^J} - \frac{A_{N-1}^J + C_{2,N-1}^J}{\mathbb{B}_{2,N-1}^J} \right) \left(2 M_F^J \bar{W} + \frac{\alpha_{MF\sigma dN}^b}{F_{I,I,N}} \right)
\end{aligned}$$

As $F_{I,I,N} \rightarrow 0$

$$\begin{aligned}
[92] \quad \frac{d\Phi}{d\ell} & \rightarrow \frac{d\Phi_{I,I,N}}{d\ell} = - \left(\frac{M_F^{JP_{32}}}{\mathbb{B}_{2,N-1}^J} + \mathbb{C}_{2,N-1}^J \right) e^{-\mathbb{B}_{2,N-1}^J(\ell - M_F^J \bar{W})} + \mathbb{C}_{2,N-1}^J \\
& \Rightarrow \int_0^{M_F^J \bar{W}} - \frac{d\Phi}{d\ell} \ell d\ell \rightarrow - \left[\frac{\left(\frac{M_F^{JP_{32} + C_{2,N-1}^J}}{\mathbb{B}_{2,N-1}^J} \right) \left(\ell + \frac{1}{\mathbb{B}_{2,N-1}^J} \right) e^{-\mathbb{B}_{2,N-1}^J(\ell - M_F^J \bar{W})}}{\mathbb{B}_{2,N-1}^J} + \mathbb{C}_{2,N-1}^J \frac{\ell^2}{2} \right]_{M_F^J \bar{W}}^{M_F^J \bar{W}} \\
& = \frac{M_F^{JP_{32}}}{\mathbb{B}_{2,N-1}^J} \left(M_F^J \bar{W} + \frac{1}{\mathbb{B}_{2,N-1}^J} \right) - \frac{M_F^{JP_{32}}}{\mathbb{B}_{2,N-1}^J} \left(2 M_F^J \bar{W} + \frac{1}{\mathbb{B}_{2,N-1}^J} \right) e^{-\mathbb{B}_{2,N-1}^J M_F^J \bar{W}} - \frac{3}{2} \mathbb{C}_{2,N-1}^J M_F^J \bar{W}^2 \\
& = \frac{M_F^{JP_{32}}}{\mathbb{B}_{2,N-1}^J} \left(M_F^J \bar{W} + \frac{1}{\mathbb{B}_{2,N-1}^J} \right) - \frac{A_{N-1}^J}{\mathbb{B}_{2,N-1}^J} \left(2 M_F^J \bar{W} + \frac{1}{\mathbb{B}_{2,N-1}^J} \right) - \frac{3}{2} \mathbb{C}_{2,N-1}^J M_F^J \bar{W}^2
\end{aligned}$$

Finally, when $\ell > 2_{MF}^J \bar{W}$,

$$\begin{aligned}
[93] \quad \frac{d\Phi_{I,J,N}}{d\ell} &= -\mathbb{A}_{N-1}^J e^{-\mathbb{B}_{N-1}^J (\ell - 2_{MF}^J \bar{W})} \\
&\Rightarrow \int \Phi d\ell - [\Phi(\ell)] = - \left[\left(\frac{\mathbb{A}_{N-1}^J}{\mathbb{B}_{N-1}^J} e^{-\mathbb{B}_{N-1}^J (\ell - 2_{MF}^J \bar{W})} \right) \left(\frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} e^{-\frac{F_{I,I,N} \ell}{\alpha_{MF} \sigma_{dN}^b}} + \ell e^{-\frac{F_{I,I,N} \ell}{\alpha_{MF} \sigma_{dN}^b}} \right) \right] \\
&\quad - \int \mathbb{A}_{N-1}^J e^{-\mathbb{B}_{N-1}^J (\ell - 2_{MF}^J \bar{W})} \frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} e^{-\frac{F_{I,I,N} \ell}{\alpha_{MF} \sigma_{dN}^b}} d\ell \\
&= - \left[\left(\frac{\mathbb{A}_{N-1}^J}{\mathbb{B}_{N-1}^J} e^{-\mathbb{B}_{N-1}^J (\ell - 2_{MF}^J \bar{W})} \right) \left(\frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} e^{-\frac{F_{I,I,N} \ell}{\alpha_{MF} \sigma_{dN}^b}} + \ell e^{-\frac{F_{I,I,N} \ell}{\alpha_{MF} \sigma_{dN}^b}} \right) \right] \\
&\quad - \int \mathbb{A}_{N-1}^J \frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} e^{2\mathbb{B}_{N-1}^J \bar{W}} e^{-\left(\mathbb{B}_{N-1}^J + \frac{F_{I,I,N}}{\alpha_{MF} \sigma_{dN}^b}\right) \ell} d\ell \\
&= - \left[\left(\frac{\mathbb{A}_{N-1}^J}{\mathbb{B}_{N-1}^J} e^{-\mathbb{B}_{N-1}^J (\ell - 2_{MF}^J \bar{W})} \right) \left(\frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} e^{-\frac{F_{I,I,N} \ell}{\alpha_{MF} \sigma_{dN}^b}} + \ell e^{-\frac{F_{I,I,N} \ell}{\alpha_{MF} \sigma_{dN}^b}} \right) \right] \\
&\quad + \left[\frac{\mathbb{A}_{N-1}^J \frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}}}{\mathbb{B}_{N-1}^J + \frac{F_{I,I,N}}{\alpha_{MF} \sigma_{dN}^b}} e^{2\mathbb{B}_{N-1}^J \bar{W}} e^{-\left(\mathbb{B}_{N-1}^J + \frac{F_{I,I,N}}{\alpha_{MF} \sigma_{dN}^b}\right) \ell} \right] \\
&= - \left[\frac{\mathbb{A}_{N-1}^J}{\mathbb{B}_{N-1}^J} \left(\ell + \frac{1}{\mathbb{B}_{N-1}^J + \frac{F_{I,I,N}}{\alpha_{MF} \sigma_{dN}^b}} \right) e^{2\mathbb{B}_{N-1}^J \bar{W}} e^{-\left(\mathbb{B}_{N-1}^J + \frac{F_{I,I,N}}{\alpha_{MF} \sigma_{dN}^b}\right) \ell} \right]
\end{aligned}$$

and hence

$$[94] \quad \int_{2_{MF}^J \bar{W}}^{\infty} \frac{d\Phi}{d\ell} \ell d\ell = \frac{\mathbb{A}_{N-1}^J}{\mathbb{B}_{N-1}^J} \left(2_{MF}^J \bar{W} + \frac{1}{\mathbb{B}_{N-1}^J + \frac{F_{I,I,N}}{\alpha_{MF} \sigma_{dN}^b}} \right) e^{-\frac{2F_{I,I,N} \mathbb{B}_{N-1}^J \bar{W}}{\alpha_{MF} \sigma_{dN}^b}}$$

As $F_{I,I,N} \rightarrow 0$

$$\begin{aligned}
[95] \quad \frac{d\Phi}{d\ell} &\rightarrow \frac{d\Phi_{I,J,N}}{d\ell} = -\mathbb{A}_{N-1}^J e^{-\mathbb{B}_{N-1}^J (\ell - 2_{MF}^J \bar{W})} \\
&\Rightarrow \int_{2_{MF}^J \bar{W}}^{\infty} \frac{d\Phi}{d\ell} \ell d\ell = - \left[\frac{\mathbb{A}_{N-1}^J}{\mathbb{B}_{N-1}^J} \left(\ell + \frac{1}{\mathbb{B}_{N-1}^J} \right) e^{-\mathbb{B}_{N-1}^J (\ell - 2_{MF}^J \bar{W})} \right]_{2_{MF}^J \bar{W}}^{\infty} \\
&= \frac{\mathbb{A}_{N-1}^J}{\mathbb{B}_{N-1}^J} \left(2_{MF}^J \bar{W} + \frac{1}{\mathbb{B}_{N-1}^J} \right)
\end{aligned}$$

Therefore the mean residual half-macrofracture length, $\bar{\ell}$, will be given by:

$$\begin{aligned}
[96] \quad \bar{\ell} &= \int_0^{2_{MF}^J \bar{W}} \frac{d\Phi}{d\ell} \ell d\ell + \int_{2_{MF}^J \bar{W}}^{\infty} \frac{d\Phi}{d\ell} \ell d\ell + \int_{2_{MF}^J \bar{W}}^{\infty} \frac{d\Phi}{d\ell} \ell d\ell \\
&= \left(\psi_{MF}^J \bar{W} + \left(\frac{\psi_{MF}^J \alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} - \bar{W} \right) \right) e^{-\frac{F_{I,I,N} \bar{W}}{\alpha_{MF} \sigma_{dN}^b}} - \left(\mathbb{P}_{32}^J \left(\frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} \right)^2 - \frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} \right) \\
&\quad + \left(\frac{\mathbb{P}_{32}^J + \mathbb{C}_{2,N-1}^J}{\mathbb{B}_{2,N-1}^J} \left(\bar{W} + \frac{1}{\mathbb{B}_{2,N-1}^J + \frac{F_{I,I,N}}{\alpha_{MF} \sigma_{dN}^b}} \right) + \mathbb{C}_{2,N-1}^J \left(\frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} \right)^2 \right) e^{-\frac{F_{I,I,N} \bar{W}}{\alpha_{MF} \sigma_{dN}^b}} \\
&\quad + \left(\left(\frac{\mathbb{A}_{N-1}^J}{\mathbb{B}_{N-1}^J} - \frac{\mathbb{A}_{N-1}^J + \mathbb{C}_{2,N-1}^J}{\mathbb{B}_{2,N-1}^J} \right) - \mathbb{C}_{2,N-1}^J \bar{W} \right) \left(\bar{W} + \frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} \right) e^{-\frac{F_{I,I,N} \bar{W}}{\alpha_{MF} \sigma_{dN}^b}}
\end{aligned}$$

$$\begin{aligned}
& - \left(\frac{\mathbb{A}_{2,N-1}^J + \mathbb{C}_{2,N-1}^J}{\mathbb{B}_{2,N-1}^J} \left(2_{MF}^J \bar{W} + \frac{1}{\mathbb{B}_{2,N-1}^J + \frac{F_{I,I,N}}{\alpha_{MF} \sigma_{dN}^b}} \right) + \mathbb{C}_{2,N-1}^J \left(\frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} \right)^2 \right) e^{-\frac{2F_{I,I,N} \bar{W}}{\alpha_{MF} \sigma_{dN}^b}} \\
& + \left(\frac{\mathbb{A}_{N-1}^J}{\mathbb{B}_{N-1}^J} - \frac{\mathbb{A}_{N-1}^J + \mathbb{C}_{2,N-1}^J}{\mathbb{B}_{2,N-1}^J} \right) \left(2_{MF}^J \bar{W} + \frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} \right) \\
& + \frac{\mathbb{A}_{N-1}^J}{\mathbb{B}_{N-1}^J} \left(2_{MF}^J \bar{W} + \frac{1}{\mathbb{B}_{N-1}^J + \frac{F_{I,I,N}}{\alpha_{MF} \sigma_{dN}^b}} \right) e^{-\frac{2F_{I,I,N} \bar{W}}{\alpha_{MF} \sigma_{dN}^b}}
\end{aligned}$$

If there is no stress shadow interaction (i.e. $F_{I,I,N} \rightarrow 0$), this reduces to

$$\begin{aligned}
[97] \quad \bar{\ell} &= \int_0^{J_{MF} \bar{W}} -\frac{d\Phi}{d\ell} \ell d\ell + \int_{J_{MF} \bar{W}}^{2_{MF}^J \bar{W}} -\frac{d\Phi}{d\ell} \ell d\ell + \int_{2_{MF}^J \bar{W}}^{\infty} -\frac{d\Phi}{d\ell} \ell d\ell \\
&= \frac{1}{2} \psi_{MF}^J \bar{W} + \frac{J_{MF} P_{32}}{\mathbb{B}_{2,N-1}^J} \left(\bar{W} + \frac{1}{\mathbb{B}_{2,N-1}^J} \right) - \frac{\mathbb{A}_{N-1}^J}{\mathbb{B}_{2,N-1}^J} \left(2_{MF}^J \bar{W} + \frac{1}{\mathbb{B}_{2,N-1}^J} \right) - \frac{3}{2} \mathbb{C}_{2,N-1}^J \bar{W}^2 + \frac{\mathbb{A}_{N-1}^J}{\mathbb{B}_{N-1}^J} \left(2_{MF}^J \bar{W} + \frac{1}{\mathbb{B}_{N-1}^J} \right)
\end{aligned}$$

3.2 Calculating $\bar{\ell}$ with multiple dipsets

If the fracture set J contains multiple dipsets with varying stress shadow width, then [84] becomes

$$\begin{aligned}
[98] \quad \Phi_N(\Delta \ell_N) &= \begin{cases} \left(1 - \frac{J_{MF} P_{32} \Delta \ell_N}{\mathbb{B}_{2,N-1}^J} \right) e^{-\frac{F_{I,I,N} \Delta \ell_N}{\alpha_{MF} \sigma_{dN}^b}} & \text{if } \Delta \ell_N \leq \bar{W}_{MF} \\ \left(\frac{(\mathbb{A}_{r,N-1}^J + \mathbb{C}_{r,N-1}^J)}{\mathbb{B}_{r,N-1}^J} e^{-\frac{J_{r,N-1}^J}{\mathbb{B}_{r,N-1}^J} (\Delta \ell_N - \bar{W}_{MF})} - \frac{(\mathbb{A}_{r-1,N-1}^J + \mathbb{C}_{r,N-1}^J)}{\mathbb{B}_{r,N-1}^J} \right) \\ - \mathbb{C}_{r,N-1}^J \left(\bar{W}_{MF}^{r-1} - (\Delta \ell_N - \bar{W}_{MF}) \right) + \sum_{s=1}^{r-1} \left[\frac{\mathbb{A}_{s,N-1}^J - \mathbb{A}_{s-1,N-1}^J}{\mathbb{B}_{s,N-1}^J} \right] \left[-\mathbb{C}_{s,N-1}^J \left(\bar{W}_{MF}^{s-1} - \bar{W}_{MF} \right) \right] \right) e^{-\frac{F_{I,I,N} \Delta \ell_N}{\alpha_{MF} \sigma_{dN}^b}} & \text{if } \Delta \ell_N > \bar{W}_{MF} \end{cases}
\end{aligned}$$

where r is the fracture dipset for which $\bar{W}_r < \Delta \ell_N \leq \bar{W}_{r-1}$, and \bar{W}_{MF} and \mathbb{B}_{n+1}^J are given by [71] and [2] respectively.

When $\ell \leq \bar{W}_{MF}$, therefore, [87], [88], and [89] remain valid.

When $\ell > \bar{W}_{MF}$,

$$[99] \quad \frac{d\Phi_{I,I,N}}{d\ell} = -(\mathbb{A}_{r,N-1}^J + \mathbb{C}_{r,N-1}^J) e^{-\frac{J_{r,N-1}^J}{\mathbb{B}_{r,N-1}^J} (\ell - \bar{W}_{MF})} + \mathbb{C}_{r,N-1}^J$$

Therefore

$$\begin{aligned}
[100] \quad \int \Phi d\ell - [\Phi(\ell)\ell] &= [\Phi_{I,I,N} \int \Phi_{I,I,N} d\ell] - \int \left(\frac{d\Phi_{I,I,N}}{d\ell} \int \Phi_{I,I,N} d\ell \right) d\ell - [\Phi_{I,I,N} \Phi_{I,I,N} \ell] \\
&= - \left[\left(\frac{(\mathbb{A}_{r,N-1}^J + \mathbb{C}_{r,N-1}^J)}{\mathbb{B}_{r,N-1}^J} e^{-\frac{J_{r,N-1}^J}{\mathbb{B}_{r,N-1}^J} (\ell - \bar{W}_{MF})} - \frac{(\mathbb{A}_{r-1,N-1}^J + \mathbb{C}_{r,N-1}^J)}{\mathbb{B}_{r,N-1}^J} \right) \left(\frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} e^{-\frac{F_{I,I,N} \ell}{\alpha_{MF} \sigma_{dN}^b}} + \right. \right. \\
&\quad \left. \left. - \mathbb{C}_{r,N-1}^J \left(\bar{W}_{MF}^{r-1} - (\ell - \bar{W}_{MF}) \right) + \sum_{s=1}^{r-1} \left[\frac{\mathbb{A}_{s,N-1}^J - \mathbb{A}_{s-1,N-1}^J}{\mathbb{B}_{s,N-1}^J} \right] \left[-\mathbb{C}_{s,N-1}^J \left(\bar{W}_{MF}^{s-1} - \bar{W}_{MF} \right) \right] \right) \right] \\
&\quad \left[\ell e^{-\frac{F_{I,I,N} \ell}{\alpha_{MF} \sigma_{dN}^b}} \right] \\
&\quad - \int \left(\left((\mathbb{A}_{r,N-1}^J + \mathbb{C}_{r,N-1}^J) e^{-\frac{J_{r,N-1}^J}{\mathbb{B}_{r,N-1}^J} (\ell - \bar{W}_{MF})} - \mathbb{C}_{r,N-1}^J \right) \frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} e^{-\frac{F_{I,I,N} \ell}{\alpha_{MF} \sigma_{dN}^b}} \right) d\ell
\end{aligned}$$

$$\begin{aligned}
&= - \left[\left(\frac{(\mathbb{A}_{r,N-1}^J + \mathbb{C}_{r,N-1}^J)}{\mathbb{B}_{r,N-1}^J} e^{-\mathbb{B}_{r,N-1}^J \left((\ell - \bar{M}_F \bar{W}) - \bar{M}_F \bar{W} \right)} - \frac{(\mathbb{A}_{r-1,N-1}^J + \mathbb{C}_{r,N-1}^J)}{\mathbb{B}_{r,N-1}^J} \right) \left(\left(\frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} + \ell \right) e^{-\frac{F_{I,I,N} \ell}{\alpha_{MF} \sigma_{dN}^b}} \right) \right. \\
&\quad \left. - \mathbb{C}_{r,N-1}^J \left(\bar{M}_F^{r-1} \bar{W} - \left(\ell - \bar{M}_F \bar{W} \right) \right) + \sum_{s=1}^{r-1} \left[\frac{\mathbb{A}_{s,N-1}^J - \mathbb{A}_{s-1,N-1}^J}{\mathbb{B}_{s,N-1}^J} \right] \left[-\mathbb{C}_{s,N-1}^J \left(\bar{M}_F^{s-1} \bar{W} - \bar{M}_F \bar{W} \right) \right] \right] \left(\left(\frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} + \ell \right) e^{-\frac{F_{I,I,N} \ell}{\alpha_{MF} \sigma_{dN}^b}} \right) \\
&\quad - \int (\mathbb{A}_{r,N-1}^J + \mathbb{C}_{r,N-1}^J) e^{\mathbb{B}_{r,N-1}^J (\bar{M}_F \bar{W} + \bar{M}_F \bar{W})} \frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} e^{-\left(\mathbb{B}_{r,N-1}^J + \frac{F_{I,I,N}}{\alpha_{MF} \sigma_{dN}^b} \right) \ell} d\ell \\
&\quad + \int \mathbb{C}_{r,N-1}^J \frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} e^{-\frac{F_{I,I,N} \ell}{\alpha_{MF} \sigma_{dN}^b}} d\ell \\
&= - \left[\left(\frac{(\mathbb{A}_{r,N-1}^J + \mathbb{C}_{r,N-1}^J)}{\mathbb{B}_{r,N-1}^J} e^{-\mathbb{B}_{r,N-1}^J \left((\ell - \bar{M}_F \bar{W}) - \bar{M}_F \bar{W} \right)} - \frac{(\mathbb{A}_{r-1,N-1}^J + \mathbb{C}_{r,N-1}^J)}{\mathbb{B}_{r,N-1}^J} \right) \left(\left(\frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} + \ell \right) e^{-\frac{F_{I,I,N} \ell}{\alpha_{MF} \sigma_{dN}^b}} \right) \right. \\
&\quad \left. - \mathbb{C}_{r,N-1}^J \left(\bar{M}_F^{r-1} \bar{W} - \left(\ell - \bar{M}_F \bar{W} \right) \right) + \sum_{s=1}^{r-1} \left[\frac{\mathbb{A}_{s,N-1}^J - \mathbb{A}_{s-1,N-1}^J}{\mathbb{B}_{s,N-1}^J} \right] \left[-\mathbb{C}_{s,N-1}^J \left(\bar{M}_F^{s-1} \bar{W} - \bar{M}_F \bar{W} \right) \right] \right] \\
&\quad + \left[\frac{(\mathbb{A}_{r,N-1}^J + \mathbb{C}_{r,N-1}^J) \frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}}}{\left(\mathbb{B}_{r,N-1}^J + \frac{F_{I,I,N}}{\alpha_{MF} \sigma_{dN}^b} \right)} e^{\mathbb{B}_{r,N-1}^J (\bar{M}_F \bar{W} + \bar{M}_F \bar{W})} e^{-\left(\mathbb{B}_{r,N-1}^J + \frac{F_{I,I,N}}{\alpha_{MF} \sigma_{dN}^b} \right) \ell} \right] \\
&\quad - \left[\mathbb{C}_{r,N-1}^J \left(\frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} \right)^2 e^{-\frac{F_{I,I,N} \ell}{\alpha_{MF} \sigma_{dN}^b}} \right] \\
&= \\
&\quad - \left[\frac{(\mathbb{A}_{r,N-1}^J + \mathbb{C}_{r,N-1}^J)}{\mathbb{B}_{r,N-1}^J} \left(\ell + \frac{1}{\mathbb{B}_{r,N-1}^J + \frac{F_{I,I,N}}{\alpha_{MF} \sigma_{dN}^b}} \right) e^{\mathbb{B}_{r,N-1}^J (\bar{M}_F \bar{W} + \bar{M}_F \bar{W})} e^{-\left(\mathbb{B}_{r,N-1}^J + \frac{F_{I,I,N}}{\alpha_{MF} \sigma_{dN}^b} \right) \ell} + \mathbb{C}_{r,N-1}^J \left(\frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} \right)^2 e^{-\frac{F_{I,I,N} \ell}{\alpha_{MF} \sigma_{dN}^b}} \right. \\
&\quad \left. + \left(\sum_{s=1}^{r-1} \left[\frac{\mathbb{A}_{s,N-1}^J - \mathbb{A}_{s-1,N-1}^J}{\mathbb{B}_{s,N-1}^J} \right] \left[-\mathbb{C}_{s,N-1}^J \left(\bar{M}_F^{s-1} \bar{W} - \bar{M}_F \bar{W} \right) \right] - \frac{(\mathbb{A}_{r-1,N-1}^J + \mathbb{C}_{r,N-1}^J)}{\mathbb{B}_{r,N-1}^J} - \mathbb{C}_{r,N-1}^J \left(\bar{M}_F^{r-1} \bar{W} - \left(\ell - \bar{M}_F \bar{W} \right) \right) \right) \left(\ell + \frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} \right) e^{-\frac{F_{I,I,N} \ell}{\alpha_{MF} \sigma_{dN}^b}} \right]
\end{aligned}$$

Hence when $\bar{M}_F^{r-1} \bar{W} \leq \ell - \bar{M}_F \bar{W} < \bar{M}_F^{r-1} \bar{W}$,

$$\begin{aligned}
[101] \quad &\int_{\bar{M}_F \bar{W} + \bar{M}_F \bar{W}}^{\bar{M}_F^{r-1} \bar{W} + \bar{M}_F \bar{W}} \frac{d\Phi}{d\ell} \ell d\ell = \\
&\left(\frac{(\mathbb{A}_{r,N-1}^J + \mathbb{C}_{r,N-1}^J)}{\mathbb{B}_{r,N-1}^J} \left(\left(\bar{M}_F \bar{W} + \bar{M}_F \bar{W} \right) + \frac{1}{\mathbb{B}_{r,N-1}^J + \frac{F_{I,I,N}}{\alpha_{MF} \sigma_{dN}^b}} \right) + \mathbb{C}_{r,N-1}^J \left(\frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} \right)^2 \right. \\
&\quad \left. + \left(\sum_{s=1}^{r-1} \left[\frac{\mathbb{A}_{s,N-1}^J - \mathbb{A}_{s-1,N-1}^J}{\mathbb{B}_{s,N-1}^J} \right] \left[-\mathbb{C}_{s,N-1}^J \left(\bar{M}_F^{s-1} \bar{W} - \bar{M}_F \bar{W} \right) \right] - \frac{(\mathbb{A}_{r-1,N-1}^J + \mathbb{C}_{r,N-1}^J)}{\mathbb{B}_{r,N-1}^J} - \mathbb{C}_{r,N-1}^J \left(\bar{M}_F^{r-1} \bar{W} - \bar{M}_F \bar{W} \right) \right) \left(\left(\bar{M}_F \bar{W} + \bar{M}_F \bar{W} \right) + \frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} \right) \right) e^{-\frac{F_{I,I,N} \left(\bar{M}_F \bar{W} + \bar{M}_F \bar{W} \right)}{\alpha_{MF} \sigma_{dN}^b}} \\
&\quad - \left(\frac{(\mathbb{A}_{r-1,N-1}^J + \mathbb{C}_{r,N-1}^J)}{\mathbb{B}_{r,N-1}^J} \left(\left(\bar{M}_F^{r-1} \bar{W} + \bar{M}_F \bar{W} \right) + \frac{1}{\mathbb{B}_{r,N-1}^J + \frac{F_{I,I,N}}{\alpha_{MF} \sigma_{dN}^b}} \right) + \mathbb{C}_{r,N-1}^J \left(\frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} \right)^2 \right. \\
&\quad \left. + \left(\sum_{s=1}^{r-1} \left[\frac{\mathbb{A}_{s,N-1}^J - \mathbb{A}_{s-1,N-1}^J}{\mathbb{B}_{s,N-1}^J} \right] \left[-\mathbb{C}_{s,N-1}^J \left(\bar{M}_F^{s-1} \bar{W} - \bar{M}_F \bar{W} \right) \right] - \frac{(\mathbb{A}_{r-1,N-1}^J + \mathbb{C}_{r,N-1}^J)}{\mathbb{B}_{r,N-1}^J} \right) \left(\left(\bar{M}_F^{r-1} \bar{W} + \bar{M}_F \bar{W} \right) + \frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} \right) \right) e^{-\frac{F_{I,I,N} \left(\bar{M}_F^{r-1} \bar{W} + \bar{M}_F \bar{W} \right)}{\alpha_{MF} \sigma_{dN}^b}}
\end{aligned}$$

and when $0 < \ell - \bar{M}_F \bar{W} \leq \bar{M}_F^{r-1} \bar{W}$, $r = n+1$ and

$$\begin{aligned}
[102] \quad &\int_{\bar{M}_F \bar{W}}^{\bar{M}_F^{r-1} \bar{W} + \bar{M}_F \bar{W}} \frac{d\Phi}{d\ell} \ell d\ell = \\
&\left(\frac{\bar{M}_F^{r-1} \bar{W} + \bar{M}_F \bar{W}}{\mathbb{B}_{n+1,N-1}^J} \left(\bar{M}_F \bar{W} + \frac{1}{\mathbb{B}_{n+1,N-1}^J + \frac{F_{I,I,N}}{\alpha_{MF} \sigma_{dN}^b}} \right) + \mathbb{C}_{n+1,N-1}^J \left(\frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} \right)^2 \right. \\
&\quad \left. + \left(\sum_{s=1}^n \left[\frac{\mathbb{A}_{s,N-1}^J - \mathbb{A}_{s-1,N-1}^J}{\mathbb{B}_{s,N-1}^J} \right] \left[-\mathbb{C}_{s,N-1}^J \left(\bar{M}_F^{s-1} \bar{W} - \bar{M}_F \bar{W} \right) \right] - \frac{\mathbb{A}_{n,N-1}^J + \mathbb{C}_{n+1,N-1}^J}{\mathbb{B}_{n+1,N-1}^J} - \mathbb{C}_{n+1,N-1}^J \bar{M}_F \bar{W} \right) \left(\bar{M}_F \bar{W} + \frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} \right) \right) e^{-\frac{F_{I,I,N} \bar{M}_F \bar{W}}{\alpha_{MF} \sigma_{dN}^b}}
\end{aligned}$$

$$\begin{aligned}
& - \left(\frac{\frac{A_{n,N-1}^J + C_{n+1,N-1}^J}{\mathbb{B}_{n+1,N-1}^J} \left(\left(\frac{J_{MF}^n \bar{W}}{MF} + \frac{\bar{J} \bar{W}}{MF} \right) + \frac{1}{\mathbb{B}_{n+1,N-1}^J + \frac{F_{I,I,N}}{\alpha_{MF} \sigma_{dN}^b}} \right) + \mathbb{C}_{n+1,N-1}^J \left(\frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} \right)^2}{\left(\sum_{s=1}^n \left[\frac{\frac{A_{s,N-1}^J - A_{s-1,N-1}^J}{\mathbb{B}_{s,N-1}^J} \right] - \frac{(A_{n,N-1}^J + C_{n+1,N-1}^J)}{\mathbb{B}_{n+1,N-1}^J} \right) \left(\left(\frac{J_{MF}^n \bar{W}}{MF} + \frac{\bar{J} \bar{W}}{MF} \right) + \frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} \right)} \right) e^{\frac{F_{I,I,N} \left(\frac{J_{MF}^n \bar{W}}{MF} + \frac{\bar{J} \bar{W}}{MF} \right)}{\alpha_{MF} \sigma_{dN}^b}} \\
& + \sum_{r=1}^n \left[\left(\frac{\frac{A_{r,N-1}^J + C_{r,N-1}^J}{\mathbb{B}_{r,N-1}^J} \left(\left(\frac{J_{MF}^{r-1} \bar{W}}{MF} + \frac{\bar{J} \bar{W}}{MF} \right) + \frac{1}{\mathbb{B}_{r,N-1}^J + \frac{F_{I,I,N}}{\alpha_{MF} \sigma_{dN}^b}} \right) + \mathbb{C}_{r,N-1}^J \left(\frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} \right)^2}{\left(\sum_{s=1}^{r-1} \left[\frac{\frac{A_{s,N-1}^J - A_{s-1,N-1}^J}{\mathbb{B}_{s,N-1}^J} \right] - \frac{(A_{r-1,N-1}^J + C_{r,N-1}^J)}{\mathbb{B}_{r,N-1}^J} - \mathbb{C}_{r,N-1}^J \left(\frac{J_{MF}^{r-1} \bar{W}}{MF} - \frac{J_{MF}^r \bar{W}}{MF} \right) \right) \left(\left(\frac{J_{MF}^{r-1} \bar{W}}{MF} + \frac{\bar{J} \bar{W}}{MF} \right) + \frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} \right)} \right) e^{\frac{F_{I,I,N} \left(\frac{J_{MF}^{r-1} \bar{W}}{MF} + \frac{\bar{J} \bar{W}}{MF} \right)}{\alpha_{MF} \sigma_{dN}^b}} \right. \\
& \quad \left. - \left(\frac{\frac{A_{r-1,N-1}^J + C_{r,N-1}^J}{\mathbb{B}_{r,N-1}^J} \left(\left(\frac{J_{MF}^{r-1} \bar{W}}{MF} + \frac{\bar{J} \bar{W}}{MF} \right) + \frac{1}{\mathbb{B}_{r,N-1}^J + \frac{F_{I,I,N}}{\alpha_{MF} \sigma_{dN}^b}} \right) + \mathbb{C}_{r,N-1}^J \left(\frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} \right)^2}{\left(\sum_{s=1}^{r-1} \left[\frac{\frac{A_{s,N-1}^J - A_{s-1,N-1}^J}{\mathbb{B}_{s,N-1}^J} \right] - \frac{(A_{r-1,N-1}^J + C_{r,N-1}^J)}{\mathbb{B}_{r,N-1}^J} \right) \left(\left(\frac{J_{MF}^{r-1} \bar{W}}{MF} + \frac{\bar{J} \bar{W}}{MF} \right) + \frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} \right)} \right) e^{\frac{F_{I,I,N} \left(\frac{J_{MF}^{r-1} \bar{W}}{MF} + \frac{\bar{J} \bar{W}}{MF} \right)}{\alpha_{MF} \sigma_{dN}^b}} \right]
\end{aligned}$$

and when $\Delta \ell_N \geq \bar{J}_{MF} \bar{W}$, this will be given by

$$\begin{aligned}
[105] \quad \bar{\ell}(\Delta \ell_N) \Phi(\Delta \ell_N) &= \int_{\Delta \ell_N}^{J_{MF}^{q-1} \bar{W} + \bar{J}_{MF} \bar{W}} -\frac{d\Phi}{d\ell} \ell d\ell + \sum_{r=1}^{q-1} \left[\int_{J_{MF}^r \bar{W} + \bar{J}_{MF} \bar{W}}^{J_{MF}^{r-1} \bar{W} + \bar{J}_{MF} \bar{W}} -\frac{d\Phi}{d\ell} \ell d\ell \right] \\
&= \left(\frac{\frac{A_{q,N-1}^J + C_{q,N-1}^J}{\mathbb{B}_{q,N-1}^J} \left(\Delta \ell_N + \frac{1}{\mathbb{B}_{q,N-1}^J + \frac{F_{I,I,N}}{\alpha_{MF} \sigma_{dN}^b}} \right) e^{-\mathbb{B}_{q,N-1}^J (\Delta \ell_N - \bar{J}_{MF} \bar{W} - J_{MF}^q \bar{W})} + \mathbb{C}_{q,N-1}^J \left(\frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} \right)^2}{\left(\sum_{s=1}^{q-1} \left[\frac{\frac{A_{s,N-1}^J - A_{s-1,N-1}^J}{\mathbb{B}_{s,N-1}^J} \right] - \frac{(A_{q-1,N-1}^J + C_{q,N-1}^J)}{\mathbb{B}_{q,N-1}^J} - \mathbb{C}_{q,N-1}^J \left(\frac{J_{MF}^{q-1} \bar{W}}{MF} - (\Delta \ell_N - \bar{J}_{MF} \bar{W}) \right) \right) \left(\Delta \ell_N + \frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} \right)} \right) e^{\frac{F_{I,I,N} \Delta \ell_N}{\alpha_{MF} \sigma_{dN}^b}} \\
&\quad - \left(\frac{\frac{A_{q-1,N-1}^J + C_{q,N-1}^J}{\mathbb{B}_{q,N-1}^J} \left((J_{MF}^{q-1} \bar{W} + \bar{J}_{MF} \bar{W}) + \frac{1}{\mathbb{B}_{q,N-1}^J + \frac{F_{I,I,N}}{\alpha_{MF} \sigma_{dN}^b}} \right) + \mathbb{C}_{q,N-1}^J \left(\frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} \right)^2}{\left(\sum_{s=1}^{q-1} \left[\frac{\frac{A_{s,N-1}^J - A_{s-1,N-1}^J}{\mathbb{B}_{s,N-1}^J} \right] - \frac{(A_{q-1,N-1}^J + C_{q,N-1}^J)}{\mathbb{B}_{q,N-1}^J} \right) \left((J_{MF}^{q-1} \bar{W} + \bar{J}_{MF} \bar{W}) + \frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} \right)} \right) e^{\frac{F_{I,I,N} \left(\frac{J_{MF}^{q-1} \bar{W}}{MF} + \frac{\bar{J} \bar{W}}{MF} \right)}{\alpha_{MF} \sigma_{dN}^b}} \\
&\quad + \sum_{r=1}^{q-1} \left[\left(\frac{\frac{A_{r,N-1}^J + C_{r,N-1}^J}{\mathbb{B}_{r,N-1}^J} \left(\left(\frac{J_{MF}^r \bar{W}}{MF} + \frac{\bar{J} \bar{W}}{MF} \right) + \frac{1}{\mathbb{B}_{r,N-1}^J + \frac{F_{I,I,N}}{\alpha_{MF} \sigma_{dN}^b}} \right) + \mathbb{C}_{r,N-1}^J \left(\frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} \right)^2}{\left(\sum_{s=1}^{r-1} \left[\frac{\frac{A_{s,N-1}^J - A_{s-1,N-1}^J}{\mathbb{B}_{s,N-1}^J} \right] - \frac{(A_{r-1,N-1}^J + C_{r,N-1}^J)}{\mathbb{B}_{r,N-1}^J} - \mathbb{C}_{r,N-1}^J \left(\frac{J_{MF}^{r-1} \bar{W}}{MF} - \frac{J_{MF}^r \bar{W}}{MF} \right) \right) \left(\left(\frac{J_{MF}^r \bar{W}}{MF} + \frac{\bar{J} \bar{W}}{MF} \right) + \frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} \right)} \right) e^{\frac{F_{I,I,N} \left(\frac{J_{MF}^r \bar{W}}{MF} + \frac{\bar{J} \bar{W}}{MF} \right)}{\alpha_{MF} \sigma_{dN}^b}} \right. \\
&\quad \left. - \left(\frac{\frac{A_{r-1,N-1}^J + C_{r,N-1}^J}{\mathbb{B}_{r,N-1}^J} \left(\left(\frac{J_{MF}^{r-1} \bar{W}}{MF} + \frac{\bar{J} \bar{W}}{MF} \right) + \frac{1}{\mathbb{B}_{r,N-1}^J + \frac{F_{I,I,N}}{\alpha_{MF} \sigma_{dN}^b}} \right) + \mathbb{C}_{r,N-1}^J \left(\frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} \right)^2}{\left(\sum_{s=1}^{r-1} \left[\frac{\frac{A_{s,N-1}^J - A_{s-1,N-1}^J}{\mathbb{B}_{s,N-1}^J} \right] - \frac{(A_{r-1,N-1}^J + C_{r,N-1}^J)}{\mathbb{B}_{r,N-1}^J} \right) \left(\left(\frac{J_{MF}^{r-1} \bar{W}}{MF} + \frac{\bar{J} \bar{W}}{MF} \right) + \frac{\alpha_{MF} \sigma_{dN}^b}{F_{I,I,N}} \right)} \right) e^{\frac{F_{I,I,N} \left(\frac{J_{MF}^{r-1} \bar{W}}{MF} + \frac{\bar{J} \bar{W}}{MF} \right)}{\alpha_{MF} \sigma_{dN}^b}} \right]
\end{aligned}$$

where q is the fracture dipset for which $J_{MF}^q \bar{W} \leq \Delta \ell_N - \bar{J}_{MF} \bar{W} < J_{MF}^{q-1} \bar{W}$.

In the later stages of fracture growth, the population distribution for the smaller dipsets can tend towards a linear function, where the \mathbb{B} terms tend to 0 and the \mathbb{C} terms tend to infinity. In this case

$$[106] \quad \frac{d\Phi_{I,I,N}}{d\ell} \rightarrow -\frac{A_{r,N-1}^J - A_{r-1,N-1}^J}{J_{MF}^{r-1} \bar{W} - \frac{J_{MF}^r \bar{W}}{MF}} \left(J_{MF}^{r-1} \bar{W} - \left(\ell - \frac{\bar{J} \bar{W}}{MF} \right) \right) + A_{R-1}$$

and hence [100] can be rewritten as

$$\begin{aligned}
[109] \quad \int_{\frac{J^n \bar{W}}{M_F \bar{W}} + \frac{\bar{J} \bar{W}}{M_F \bar{W}}} - \frac{d\Phi}{d\ell} \ell d\ell = & \left(\left(\mathbb{A}_{n,N-1}^J - \frac{M_F^{J P_{32} - A_{n,N-1}^J}}{J_{n,N-1}^J} \left(\frac{J^n \bar{W}}{M_F \bar{W}} + \frac{\bar{J} \bar{W}}{M_F \bar{W}} \right) \right) \left(\frac{\alpha_{MF\sigma dN}^b}{F_{I,IN}} \right)^2 \right. \\
& \left. + \left(\frac{M_F^{J P_{32} - A_{n,N-1}^J}}{J_{n,N-1}^J} \left(\frac{\alpha_{MF\sigma dN}^b}{F_{I,IN}} \right)^2 + \sum_{s=1}^n \left\{ \left[\frac{1}{2} (\mathbb{A}_{s,N-1}^J + \mathbb{A}_{s-1,N-1}^J) \left(\frac{J^{s-1} \bar{W}}{M_F \bar{W}} - \frac{J^s \bar{W}}{M_F \bar{W}} \right) \right] \right\} \right) \left(\frac{\bar{J} \bar{W}}{M_F \bar{W}} + \frac{\alpha_{MF\sigma dN}^b}{F_{I,IN}} \right) \right) e^{\frac{F_{I,IN} \bar{J} \bar{W}}{\alpha_{MF\sigma dN}^b}} \\
& - \left(\left(\mathbb{A}_{n,N-1}^J - \frac{M_F^{J P_{32} - A_{n,N-1}^J}}{J_{n,N-1}^J} \left(\frac{J^n \bar{W}}{M_F \bar{W}} + \frac{\bar{J} \bar{W}}{M_F \bar{W}} \right) \right) \left(\frac{\alpha_{MF\sigma dN}^b}{F_{I,IN}} \right)^2 \right. \\
& \left. + \left(\frac{M_F^{J P_{32} - A_{n,N-1}^J}}{J_{n,N-1}^J} \left(\frac{\alpha_{MF\sigma dN}^b}{F_{I,IN}} \right)^2 + \sum_{s=1}^n \left\{ \left[\frac{1}{2} (\mathbb{A}_{s,N-1}^J + \mathbb{A}_{s-1,N-1}^J) \left(\frac{J^{s-1} \bar{W}}{M_F \bar{W}} - \frac{J^s \bar{W}}{M_F \bar{W}} \right) \right] \right\} \right) \left(\left(\frac{J^n \bar{W}}{M_F \bar{W}} + \frac{\bar{J} \bar{W}}{M_F \bar{W}} \right) + \frac{\alpha_{MF\sigma dN}^b}{F_{I,IN}} \right) \right) e^{\frac{F_{I,IN} \left(\frac{J^n \bar{W}}{M_F \bar{W}} + \frac{\bar{J} \bar{W}}{M_F \bar{W}} \right)}{\alpha_{MF\sigma dN}^b}}
\end{aligned}$$

respectively. [103], [104] and [105] can hence be updated accordingly.

If there is no stress shadow interaction (i.e. $F_{I,IN} \rightarrow 0$), $d\Phi/d\ell$ reduces to

$$[110] \quad \frac{d\Phi}{d\ell} \rightarrow \frac{d\Phi_{I,IN}}{d\ell} = -(\mathbb{A}_{r,N-1}^J + \mathbb{C}_{r,N-1}^J) e^{-\mathbb{B}_{r,N-1}^J \left(\left(\ell - \frac{J^n \bar{W}}{M_F \bar{W}} \right) - \frac{J^r \bar{W}}{M_F \bar{W}} \right)} + \mathbb{C}_{r,N-1}^J$$

Therefore

$$\begin{aligned}
[111] \quad \int_{\frac{J^n \bar{W}}{M_F \bar{W}} + \frac{\bar{J} \bar{W}}{M_F \bar{W}}} - \frac{d\Phi}{d\ell} \ell d\ell = & - \left[\frac{M_F^{J P_{32} + C_{n+1,N-1}^J}}{\mathbb{B}_{n+1,N-1}^J} \left(\ell + \frac{1}{\mathbb{B}_{n+1,N-1}^J} \right) e^{-\mathbb{B}_{n+1,N-1}^J \left(\ell - \frac{J^n \bar{W}}{M_F \bar{W}} \right)} + \frac{C_{n+1,N-1}^J \ell^2}{2} \right] \frac{J^n \bar{W}}{M_F \bar{W}} + \frac{\bar{J} \bar{W}}{M_F \bar{W}} \\
& = \frac{M_F^{J P_{32} + C_{n+1,N-1}^J}}{\mathbb{B}_{n+1,N-1}^J} \left(\frac{\bar{J} \bar{W}}{M_F \bar{W}} + \frac{1}{\mathbb{B}_{n+1,N-1}^J} \right) - \frac{M_F^{J P_{32} + C_{n+1,N-1}^J}}{\mathbb{B}_{n+1,N-1}^J} \left(\left(\frac{J^n \bar{W}}{M_F \bar{W}} + \frac{\bar{J} \bar{W}}{M_F \bar{W}} \right) + \frac{1}{\mathbb{B}_{r,N-1}^J} \right) e^{-\mathbb{B}_{n+1,N-1}^J \frac{J^n \bar{W}}{M_F \bar{W}}} \\
& \quad + \frac{1}{2} \mathbb{C}_{n+1,N-1}^J \left(\frac{\bar{J} \bar{W}^2}{M_F \bar{W}^2} - \left(\frac{J^n \bar{W}}{M_F \bar{W}} + \frac{\bar{J} \bar{W}}{M_F \bar{W}} \right)^2 \right) \\
& = \frac{M_F^{J P_{32}}}{\mathbb{B}_{n+1,N-1}^J} \left(\frac{\bar{J} \bar{W}}{M_F \bar{W}} + \frac{1}{\mathbb{B}_{n+1,N-1}^J} \right) - \frac{A_{n,N-1}^J}{\mathbb{B}_{n+1,N-1}^J} \left(\left(\frac{J^n \bar{W}}{M_F \bar{W}} + \frac{\bar{J} \bar{W}}{M_F \bar{W}} \right) + \frac{1}{\mathbb{B}_{n+1,N-1}^J} \right) \\
& \quad - \frac{C_{n+1,N-1}^J}{\mathbb{B}_{n+1,N-1}^J} \frac{J^n \bar{W}}{M_F \bar{W}} - \frac{1}{2} \mathbb{C}_{n+1,N-1}^J \left(\left(\frac{J^n \bar{W}}{M_F \bar{W}} + \frac{\bar{J} \bar{W}}{M_F \bar{W}} \right)^2 - \frac{\bar{J} \bar{W}^2}{M_F \bar{W}^2} \right)
\end{aligned}$$

and

$$\begin{aligned}
[112] \quad \int_{\frac{J^{r-1} \bar{W}}{M_F \bar{W}} + \frac{\bar{J} \bar{W}}{M_F \bar{W}}} - \frac{d\Phi}{d\ell} \ell d\ell = & - \left[\frac{(\mathbb{A}_{r,N-1}^J + \mathbb{C}_{r,N-1}^J)}{\mathbb{B}_{r,N-1}^J} \left(\ell + \frac{1}{\mathbb{B}_{r,N-1}^J} \right) e^{-\mathbb{B}_{r,N-1}^J \left(\left(\ell - \frac{J^{r-1} \bar{W}}{M_F \bar{W}} \right) - \frac{J^r \bar{W}}{M_F \bar{W}} \right)} + \frac{C_{r,N-1}^J \ell^2}{2} \right] \frac{J^{r-1} \bar{W}}{M_F \bar{W}} + \frac{\bar{J} \bar{W}}{M_F \bar{W}} \\
& = \frac{A_{r,N-1}^J}{\mathbb{B}_{r,N-1}^J} \left(\left(\frac{J^r \bar{W}}{M_F \bar{W}} + \frac{\bar{J} \bar{W}}{M_F \bar{W}} \right) + \frac{1}{\mathbb{B}_{r,N-1}^J} \right) - \frac{A_{r-1,N-1}^J}{\mathbb{B}_{r,N-1}^J} \left(\left(\frac{J^{r-1} \bar{W}}{M_F \bar{W}} + \frac{\bar{J} \bar{W}}{M_F \bar{W}} \right) + \frac{1}{\mathbb{B}_{r,N-1}^J} \right) \\
& \quad - \frac{C_{r,N-1}^J}{\mathbb{B}_{r,N-1}^J} \left(\frac{J^{r-1} \bar{W}}{M_F \bar{W}} - \frac{J^r \bar{W}}{M_F \bar{W}} \right) - \frac{1}{2} \mathbb{C}_{r,N-1}^J \left(\left(\frac{J^{r-1} \bar{W}}{M_F \bar{W}} + \frac{\bar{J} \bar{W}}{M_F \bar{W}} \right)^2 - \left(\frac{J^r \bar{W}}{M_F \bar{W}} + \frac{\bar{J} \bar{W}}{M_F \bar{W}} \right)^2 \right)
\end{aligned}$$

Hence the mean residual half-macrofracture length, $\bar{\ell}_N$, is given by:

$$[113] \quad \bar{\ell} = \int_0^{\frac{J^n \bar{W}}{M_F \bar{W}} + \frac{\bar{J} \bar{W}}{M_F \bar{W}}} - \frac{d\Phi}{d\ell} \ell d\ell + \int_{\frac{J^n \bar{W}}{M_F \bar{W}} + \frac{\bar{J} \bar{W}}{M_F \bar{W}}} - \frac{d\Phi}{d\ell} \ell d\ell + \sum_{r=1}^n \left[\int_{\frac{J^{r-1} \bar{W}}{M_F \bar{W}} + \frac{\bar{J} \bar{W}}{M_F \bar{W}}} - \frac{d\Phi}{d\ell} \ell d\ell \right]$$

$$\begin{aligned}
&= \frac{1}{2} \psi_{MF}^J \bar{J} \bar{W} + \frac{M_{32}^J}{\mathbb{B}_{n+1,N-1}^J} \left(\bar{J} \bar{W} + \frac{1}{\mathbb{B}_{n+1,N-1}^J} \right) - \frac{A_{n,N-1}^J}{\mathbb{B}_{n+1,N-1}^J} \left(\left(\bar{J} \bar{W} + \frac{1}{\mathbb{B}_{n+1,N-1}^J} \right) \right. \\
&\quad \left. - \frac{C_{n+1,N-1}^J}{\mathbb{B}_{n+1,N-1}^J} \bar{J} \bar{W} - \frac{1}{2} \mathbb{C}_{n+1,N-1}^J \left(\left(\bar{J} \bar{W} + \frac{1}{\mathbb{B}_{n+1,N-1}^J} \right)^2 - \bar{J} \bar{W}^2 \right) \right. \\
&\quad \left. + \sum_{r=1}^n \left[\frac{A_{r,N-1}^J}{\mathbb{B}_{r,N-1}^J} \left(\left(\bar{J} \bar{W} + \frac{1}{\mathbb{B}_{r,N-1}^J} \right) \right) - \frac{A_{r-1,N-1}^J}{\mathbb{B}_{r,N-1}^J} \left(\left(\bar{J} \bar{W} + \frac{1}{\mathbb{B}_{r,N-1}^J} \right) \right) \right. \right. \\
&\quad \left. \left. - \frac{C_{r,N-1}^J}{\mathbb{B}_{r,N-1}^J} \left(\bar{J} \bar{W} - \frac{1}{\mathbb{B}_{r,N-1}^J} \right) - \frac{1}{2} \mathbb{C}_{r,N-1}^J \left(\left(\bar{J} \bar{W} + \frac{1}{\mathbb{B}_{r,N-1}^J} \right)^2 - \left(\bar{J} \bar{W} + \frac{1}{\mathbb{B}_{r,N-1}^J} \right)^2 \right) \right] \right]
\end{aligned}$$

and

$$\begin{aligned}
[114] \quad \bar{\ell}(\Delta \ell_N) \Phi(\Delta \ell_N) &= \frac{1}{2} \left(\psi_{MF}^J \bar{J} \bar{W} - \frac{M_{32}^J}{\mathbb{B}_{n+1,N-1}^J} \left(\bar{J} \bar{W} + \frac{1}{\mathbb{B}_{n+1,N-1}^J} \right) - \frac{A_{n,N-1}^J}{\mathbb{B}_{n+1,N-1}^J} \left(\left(\bar{J} \bar{W} + \frac{1}{\mathbb{B}_{n+1,N-1}^J} \right) \right. \right. \\
&\quad \left. \left. - \frac{C_{n+1,N-1}^J}{\mathbb{B}_{n+1,N-1}^J} \bar{J} \bar{W} - \frac{1}{2} \mathbb{C}_{n+1,N-1}^J \left(\left(\bar{J} \bar{W} + \frac{1}{\mathbb{B}_{n+1,N-1}^J} \right)^2 - \bar{J} \bar{W}^2 \right) \right. \right. \\
&\quad \left. \left. + \sum_{r=1}^n \left[\frac{A_{r,N-1}^J}{\mathbb{B}_{r,N-1}^J} \left(\left(\bar{J} \bar{W} + \frac{1}{\mathbb{B}_{r,N-1}^J} \right) \right) - \frac{A_{r-1,N-1}^J}{\mathbb{B}_{r,N-1}^J} \left(\left(\bar{J} \bar{W} + \frac{1}{\mathbb{B}_{r,N-1}^J} \right) \right) \right. \right. \right. \\
&\quad \left. \left. \left. - \frac{C_{r,N-1}^J}{\mathbb{B}_{r,N-1}^J} \left(\bar{J} \bar{W} - \frac{1}{\mathbb{B}_{r,N-1}^J} \right) - \frac{1}{2} \mathbb{C}_{r,N-1}^J \left(\left(\bar{J} \bar{W} + \frac{1}{\mathbb{B}_{r,N-1}^J} \right)^2 - \left(\bar{J} \bar{W} + \frac{1}{\mathbb{B}_{r,N-1}^J} \right)^2 \right) \right] \right] \right)
\end{aligned}$$

when $\Delta \ell_N < \bar{J} \bar{W}$, or

$$\begin{aligned}
[115] \quad \bar{\ell}(\Delta \ell_N) \Phi(\Delta \ell_N) &= \frac{A_{q,N-1}^J}{\mathbb{B}_{q,N-1}^J} \left(\Delta \ell_N + \frac{1}{\mathbb{B}_{q,N-1}^J} \right) e^{-\mathbb{B}_{q,N-1}^J \left(\Delta \ell_N - \left(\bar{J} \bar{W} + \frac{1}{\mathbb{B}_{q,N-1}^J} \right) \right)} - \frac{A_{q-1,N-1}^J}{\mathbb{B}_{q,N-1}^J} \left(\left(\bar{J} \bar{W} + \frac{1}{\mathbb{B}_{q,N-1}^J} \right) \right. \\
&\quad \left. - \frac{C_{q,N-1}^J}{\mathbb{B}_{q,N-1}^J} \left(\left(\bar{J} \bar{W} + \frac{1}{\mathbb{B}_{q,N-1}^J} \right) - \Delta \ell_N \right) - \frac{1}{2} \mathbb{C}_{q,N-1}^J \left(\left(\bar{J} \bar{W} + \frac{1}{\mathbb{B}_{q,N-1}^J} \right)^2 - \Delta \ell_N^2 \right) \right. \\
&\quad \left. + \sum_{r=1}^{q-1} \left[\frac{A_{r,N-1}^J}{\mathbb{B}_{r,N-1}^J} \left(\left(\bar{J} \bar{W} + \frac{1}{\mathbb{B}_{r,N-1}^J} \right) \right) - \frac{A_{r-1,N-1}^J}{\mathbb{B}_{r,N-1}^J} \left(\left(\bar{J} \bar{W} + \frac{1}{\mathbb{B}_{r,N-1}^J} \right) \right) \right. \right. \\
&\quad \left. \left. - \frac{C_{r,N-1}^J}{\mathbb{B}_{r,N-1}^J} \left(\bar{J} \bar{W} - \frac{1}{\mathbb{B}_{r,N-1}^J} \right) - \frac{1}{2} \mathbb{C}_{r,N-1}^J \left(\left(\bar{J} \bar{W} + \frac{1}{\mathbb{B}_{r,N-1}^J} \right)^2 - \left(\bar{J} \bar{W} + \frac{1}{\mathbb{B}_{r,N-1}^J} \right)^2 \right) \right] \right)
\end{aligned}$$

when $\Delta \ell_N \geq \bar{J} \bar{W}$.

When \mathbb{B}_s tends to 0 and \mathbb{C}_s tends to infinity, the appropriate terms of the sums in [113], [114] and [115] can be replaced with the linear form:

$$[116] \quad \left[\frac{A_{r,N-1}^J - A_{r-1,N-1}^J}{\mathbb{B}_{r,N-1}^J - \mathbb{B}_{r-1,N-1}^J} \frac{1}{6} \left(\left(\bar{J} \bar{W} + \frac{1}{\mathbb{B}_{r,N-1}^J} \right)^3 - 3 \left(\bar{J} \bar{W} + \frac{1}{\mathbb{B}_{r,N-1}^J} \right) \left(\bar{J} \bar{W} + \frac{1}{\mathbb{B}_{r,N-1}^J} \right)^2 + 2 \left(\bar{J} \bar{W} + \frac{1}{\mathbb{B}_{r,N-1}^J} \right)^3 \right) \right. \\
\left. - \frac{1}{2} A_{r-1,N-1}^J \left(\left(\bar{J} \bar{W} + \frac{1}{\mathbb{B}_{r,N-1}^J} \right)^2 - \left(\bar{J} \bar{W} + \frac{1}{\mathbb{B}_{r,N-1}^J} \right)^2 \right) \right]$$

Where required, the n+1 terms in [113] and [114] can also be replaced with the linear forms

$$[117] \quad \frac{M_{32}^J - A_{n,N-1}^J}{\mathbb{B}_{n,N-1}^J} \frac{1}{6} \left(\left(\bar{J} \bar{W} + \frac{1}{\mathbb{B}_{n,N-1}^J} \right)^3 - 3 \left(\bar{J} \bar{W} + \frac{1}{\mathbb{B}_{n,N-1}^J} \right) \bar{J} \bar{W}^2 + 2 \bar{J} \bar{W}^3 \right) - \frac{1}{2} A_{n,N-1}^J \left(\left(\bar{J} \bar{W} + \frac{1}{\mathbb{B}_{n,N-1}^J} \right)^2 - \bar{J} \bar{W}^2 \right)$$

and the terms for dipset q in [115] can be replaced with

$$[118] \quad \left[\frac{A_{q,N-1}^J - A_{q-1,N-1}^J}{\mathbb{B}_{q,N-1}^J - \mathbb{B}_{q-1,N-1}^J} \frac{1}{6} \left(\left(\bar{J} \bar{W} + \frac{1}{\mathbb{B}_{q,N-1}^J} \right)^3 - 3 \left(\bar{J} \bar{W} + \frac{1}{\mathbb{B}_{q,N-1}^J} \right) \Delta \ell_N^2 + 2 \Delta \ell_N^3 \right) \right. \\
\left. - \frac{1}{2} A_{q-1,N-1}^J \left(\left(\bar{J} \bar{W} + \frac{1}{\mathbb{B}_{q,N-1}^J} \right)^2 - \Delta \ell_N^2 \right) \right]$$

Note that in the above equations we are calculating $\bar{\ell}(\ell_N)\Phi(\ell_N)$, which is equivalent to the combined length of half-macrofractures with length $\ell > \ell_N$, divided by the total number of half-macrofractures of any length. If we want to calculate the mean length of only those half-macrofractures that are longer than ℓ_N , we must multiply the results obtained in [104], [105], [114] and [115] by $\Phi(\ell_N)$.

Similarly, we can calculate equivalent functions for the dipset of half-macrofractures that have nucleated a minimum distance E from the nearest set J macrofracture. This may be useful if, for example, there are stress shadow effects between the set I and set J fractures that prevent set I fractures from nucleating close to set J fractures. In this case, we define $\bar{\ell}(\ell_N: \ell > E)\Phi'(\ell_N: \ell > E)$ as the combined length of half-macrofractures within this dipset with length $\ell > \ell_N$, divided by the total number of half-macrofractures in the dipset. $\Phi'(\ell_N: \ell > E)$ represents the proportion of the half-macrofractures in the dipset that propagate a distance of at least ℓ_N before being deactivated, and is given by

$$[119] \quad \Phi'(\ell_N: \ell > E) = \begin{cases} \Phi_{II}(\ell_N) & \text{if } \ell_N < E \\ \frac{\Phi_{II}(E)}{\Phi(E)} \Phi(\ell_N) = \frac{\Phi(\ell_N)}{\Phi_{IJ}(E)} & \text{if } \ell_N \geq E \end{cases}$$

Therefore

$$[120] \quad \bar{\ell}'(\ell_N: \ell > E)\Phi'(\ell_N: \ell > E) = \int_{\ell_N}^{\infty} -\frac{d\Phi'}{d\ell} \ell d\ell$$

$$= \begin{cases} \int_{\ell_N}^E -\frac{d\Phi_{II}}{d\ell} \ell d\ell + \frac{(\bar{\ell}(E)\Phi(E))}{\Phi_{IJ}(E)} & \text{if } \ell_N < E \\ \frac{(\bar{\ell}(\ell_N)\Phi(\ell_N))}{\Phi_{IJ}(E)} & \text{if } \ell_N \geq E \end{cases}$$

Since $\Phi_{II}(\ell)$ is an exponential function, [120] can be expanded to

$$[121] \quad \bar{\ell}'(\ell_N: \ell > E)\Phi'(\ell_N: \ell > E) = \int_{\ell_N}^{\infty} -\frac{d\Phi'}{d\ell} \ell d\ell$$

$$= \begin{cases} \left(\frac{\alpha_{MF}\sigma dN^b}{F_{I,I,N}} + \ell_N \right) e^{-\frac{F_{I,I,N}\ell_N}{\alpha_{MF}\sigma dN^b}} - \left(\frac{\alpha_{MF}\sigma dN^b}{F_{I,I,N}} + E \right) e^{-\frac{F_{I,I,N}E}{\alpha_{MF}\sigma dN^b}} + \frac{(\bar{\ell}(E)\Phi(E))}{\Phi_{IJ}(E)} & \text{if } \ell_N < E \\ \frac{(\bar{\ell}(\ell_N)\Phi(\ell_N))}{\Phi_{IJ}(E)} & \text{if } \ell_N \geq E \end{cases}$$

As $F_{I,I,N} \rightarrow 0$, [121] reduces to

$$[122] \quad \bar{\ell}'(\ell_N: \ell > E)\Phi'(\ell_N: \ell > E) = \begin{cases} \frac{(\bar{\ell}(E)\Phi(E))}{\Phi_{IJ}(E)} & \text{if } \ell_N < E \\ \frac{(\bar{\ell}(\ell_N)\Phi(\ell_N))}{\Phi_{IJ}(E)} & \text{if } \ell_N \geq E \end{cases}$$

4 Workflow to generate implicit and explicit fracture models

In order to apply this method to fractured layers in real geological formations, we must create an implementation of the method as either a stand-alone computer program or a plug-in for a more general geological modelling software package, that can read geological input data, e.g. geometry, mechanical properties, deformation history etc., and output fracture population data. To do this we must describe a workflow that includes all the elements of the calculation in an appropriate order. However before we do this, we must first consider the form of the input data and output data.

The simplest models, for example those used in sensitivity studies, can assume that the layer thickness and depth, the mechanical properties, and the in situ stress and strain are homogeneous and extend laterally for an infinite distance.

However to simulate fracture growth in real geological layers, we need to recreate the complex geometry of these layers (including features such as dip, folding, thickness changes, and offset due to faulting), as well as lateral variations in the mechanical properties and in situ stress state. Static geomodels typically deal with these problems by discretising the layer into polyhedral gridblocks with defined cornerpoints and internally homogeneous properties. We can thus calculate the fracture population data independently for each gridblock in the geomodel, using the properties defined for that gridblock, and taking an average value of thickness and depth if these are not uniform.

The output will mostly take the form of cumulative functions of volumetric and mean linear fracture density as a function of fracture size for each of the fracture sets ($\mu_F P_{30}(r)$, $\mu_F P_{32}(r)$, $\underline{M}_F P_{30}(\ell)$ and $\underline{M}_F P_{32}(\ell)$), but may also include properties such as fracture porosity, stress shadow volume, mean length, and connectivity (the proportion of fracture tips connecting to another fracture); indeed it may include any parameter calculated during the simulation. The cumulative functions are output in the form of piecewise functions – i.e. an array of values for arbitrary, user-defined indices (e.g. $\underline{M}_F P_{30}(\ell)$ for $\ell = 0m, 1m, 2m, 5m, 10m, 20m$). The final output data comprises the values of the specified output parameters at the end of the final timestep, but it is also possible to generate intermediate output data at the end of earlier timesteps. This represents an implicit fracture model, as the fractures are not represented directly but by gridblock properties.

We can however also generate an explicit representation of the resulting fracture network, in the form of a Discrete Fracture Network (DFN) model. A DFN is a means of representing a network of fractures in the subsurface explicitly, as a set of planar objects in 3D space. Since it is impossible to image or map individual fractures in the subsurface, the DFN cannot be an exact representation of the actual fracture network; instead we try to create an artificial fracture network with statistical properties (e.g. density, spatial and size distribution, geometry, orientation and connectivity) that match as closely as possible the statistical properties of the actual fracture network.

4.1 Workflow to generate the implicit fracture model

Once the fractured layer has been discretised into gridblocks, and these have been assigned properties and upscaled as required, we can run the calculations for each gridblock independently. This can be done sequentially or in parallel. A workflow to do this is illustrated in *Figure 5*.

4.1.1 Workflow steps

The workflow starts by calculating the initial stress state of each gridblock prior to deformation (i.e. prior to applying external horizontal strain). The initial stress depends on the depth of burial and fluid pressure, and includes compaction-related strain. The workflow then loops through the timesteps three times:

- The first loop calculates the evolution of the total active and static volumetric density $P_{30}(0)$, mean linear density $P_{32}(0)$ and volumetric ratio $P_{33}(0)$ for microfractures and half-macrofractures in each fracture set. The duration and number of timesteps are determined dynamically during this loop, and can not be predicted in advance. However this information, along with the mean fracture driving stress, the half-macrofracture propagation rate, the fracture deactivation probabilities and some other key parameters, is cached for use in the later loops. The total active and static microfracture and half-macrofracture density data for each fracture set are also cached at the end of each timestep.
- The second loop calculates values for the full array of indices for the microfractures (i.e. $\mu_F P_{30}(r)$, $\mu_F P_{32}(r)$ and $\mu_F P_{33}(r)$ for a specified list of r values). These are not cached but the final values are output at the end of the timestep; however it is necessary to loop through each timestep as the static microfracture density data is calculated incrementally. During this loop, the timestep durations, as well as the mean fracture driving stress and fracture deactivation probabilities, are taken from the data cached during the first loop.
- The third loop calculates values for the full array of indices for the half-macrofractures (i.e. $\mu_F P_{30}(\ell)$, $\mu_F P_{32}(\ell)$ and $\mu_F P_{33}(\ell)$ for a specified list of ℓ values). These are not cached but the final values are output at the end of the timestep; the timestep durations, mean fracture driving stress and fracture deactivation probabilities are taken from the data cached during the first loop. Calculating the piecewise cumulative function data during separate loops allows us to set the indices automatically, to reflect the theoretical maximum half-macrofracture length given the model duration and half-macrofracture propagation rate.

The specified output data is written to the output file or database at the end of the final timesteps of each loop.

Within the first loop, the approximate calculation sequence for each timestep is as follows:

1. Calculate the in situ stress state at the start of the timestep (described in Chapter 6 of Welch et al. 2020).
2. Calculate the initial driving stress for each fracture set, and the rate of increase in fracture driving stress for each fracture set if the driving stress is variable (Chapter 6 of Welch et al. 2020).
3. Calculate the optimal timestep duration for each active fracture set, and set the current timestep duration to the shortest of these (Appendix C of Welch et al. 2020).
4. Calculate the weighted mean driving stress for the timestep (Chapter 6 of Welch et al. 2020), the fracture growth factor parameters γ and Γ (Section 3.2 of Welch et al. 2020), and the macrofracture propagation distance $\alpha_{MF}\Lambda$, (Section 4.2 of Welch et al. 2020).

5. Calculate the probabilities of half-macrofracture deactivation during the timestep, based on the total macrofracture populations at the end of the previous timestep (Section 5.1 of Welch et al. 2020; see also Section 2 of this document).
6. Calculate the total active half-macrofracture populations $a_{MF}P_{30}(0)$, $a_{MF}P_{32}(0)$ and $a_{MF}P_{33}(0)$ at the end of the timestep, and the increments in static half-macrofracture populations $\Delta^s_{MF}P_{30}(0)$, $\Delta^s_{MF}P_{32}(0)$ and $\Delta^s_{MF}P_{33}(0)$ during the timestep (Sections 5.3 and 5.4 of Welch et al. 2020). Also calculate the total stress shadow volume ψ at the end of the timestep.
7. Calculate the probabilities of microfracture deactivation during the timestep (Section 5.1 of Welch et al. 2020; see also Section 2 of this document). Note that this is a function of the total stress shadow volume at the end of the timestep, so cannot be calculated before the half-macrofracture population data is updated.
8. Calculate the total active microfracture populations $a_{\mu F}P_{30}(0)$, $a_{\mu F}P_{32}(0)$ and $a_{\mu F}P_{33}(0)$ at the end of the timestep, and the increments in static half-macrofracture populations $\Delta^s_{\mu F}P_{30}(0)$, $\Delta^s_{\mu F}P_{32}(0)$ and $\Delta^s_{\mu F}P_{33}(0)$ during the timestep (Section 5.2 of Welch et al. 2020). Note that this is dependent on the probability of half-macrofracture deactivation during the timestep, so cannot be calculated before the half-macrofracture population data is updated.
9. Check the updated fracture and timestep data to determine whether the calculation termination criteria are met (see Section 4.1.2). If not, continue to the next timestep.

Within the second and third loops, most of these steps can be skipped by reusing data calculated in the first loop. If the timestep duration, the weighted mean driving stress, the fracture growth factor parameters γ and Γ , the macrofracture propagation distance $\alpha_{MF}\Lambda$, and the fracture deactivation probabilities q , θ , F and Φ are cached during the first loop, then only step 8 need be calculated in the second loop, and only step 6 need be calculated in the third loop.

4.1.2 Terminating the calculation

Any implementation of the workflow must include termination criteria that determine when to stop the calculation. As noted previously, it is important to determine this correctly. Often the geometry of a fracture network may change significantly during the late stages of the evolution, for example the growth of large numbers of short residual fractures, or the development of a late secondary fracture set connecting the previously isolated primary fractures. In some cases, there may be a long “dormant” period between the growth of the primary fracture set and the growth of the secondary fracture set (see Welch et al. 2019).

We can combine different types of termination criteria in the same algorithm. If we do so, we must determine which combination of criteria are required to terminate the calculation. Some criteria apply to the whole model while others apply to individual fracture sets, so we must also determine how many fracture sets need to meet the termination criteria in order to terminate the calculation. We will start by examining individual termination criteria in Section 4.1.2.1, and then investigate how best to combine them in Section 4.1.2.2.

4.1.2.1 Termination criteria

Termination criteria can be classified into two types: static criteria, which are determined in advance, and dynamic criteria, which depend on the values of parameters calculated during the simulation. Static criteria typically apply to the entire model, while dynamic criteria often apply to individual fracture sets.

The main static termination criteria are:

- **Duration of then model (in geological time):** If we know the duration of the deformation episode during which the fractures developed, we can set the simulation to terminate when this time has elapsed (i.e. when the sum of all timestep durations exceeds the preset maximum). This criterion may be a limiting factor for some slow-growing fracture sets, but initial studies suggest that many fracture networks will develop over much shorter timescales than tectonic deformation episodes (thousands to hundreds of thousands of years, rather than millions of years; see Welch et al. 2019); model duration will not be an important criterion in these cases. If the duration of the deformation episode is uncertain or unknown, a better approach may be to output the fracture data at regular intervals during the fracture evolution, and use these multiple realisations for calibration or uncertainty analysis.
- **Number of timesteps:** The purpose of this termination criterion is to prevent excessive calculation times, especially in cases where none of the other termination criteria are met, or where the timestep duration is set so low that the fracture network cannot grow. The calculation will be terminated immediately when the preset number of timesteps is exceeded. Care must be taken when applying this criterion, as it can cause the calculation to stop at an arbitrary time which has no geological or physical basis. Generally, triggering this termination criterion should be taken as an indication that there is a problem with the model that must be resolved before the results can be used.

The dynamic criteria are typically used to determine when individual fracture sets have stopped growing. They are defined in terms of the fracture set parameters. There are many permutations that can be used to do this, but some common ones are listed below:

- **Proportion of active fractures:** A fracture set can be considered to have stopped growing when the ratio of active fractures to total fracture (i.e. $\frac{a_{MF}P_{30}}{MF P_{30}}$) falls below a specified value (e.g. 0.01). However care should be taken when applying this criterion, as it may exclude the development of residual fractures: in some cases (e.g. secondary fracture sets), a significant proportion of the final fracture network may grow during the residual fracture stage, which is typically characterised by very low ratios of active to total fractures.
- **Ratio of current to peak historic active fractures:** Typically we see an initial rapid rise in the population of active macrofractures in a given fracture set as new macrofractures nucleate; however as the macrofracture deactivation rate starts to rise, the population of active macrofractures reaches a peak before declining to negligible levels (see *Figure 6*). We can therefore consider that the fracture set has stopped growing when the population of active fractures (typically expressed as a mean linear density $a_{MF}P_{32}$ or volumetric ratio $a_{MF}P_{33}$) drops below a specified proportion of its peak value (e.g. 0.01). However care should be taken when applying this criterion, as it may exclude the development of residual fractures: in some cases (e.g. secondary fracture sets), a significant proportion of the final fracture network may grow during the residual fracture stage, which is typically characterised by very low numbers of active fractures.
- **Total clear zone volume:** If the total clear zone volume reaches zero, then the entire rockmass lies within a stress shadow exclusion zone and there is no space for any fractures to nucleate or grow. In practice we often define a minimum clear zone volume slightly greater than zero (e.g. 0.01) since some of the equations (e.g. those used to calculate the probability of macrofracture stress shadow interaction or residual fracture populations) can give unstable results when the clear zone volume is very low. Using this termination criteria to determine when a fracture set is inactive has the advantage that it will not exclude the development of residual fractures. However there are two drawbacks: it cannot be used in the evenly distributed stress scenario

(where there are no stress shadows); and it can be complicated to apply when there are multiple dipsets in the same fracture set, since the clear zone volume is dependent on the dipset of the propagating fracture (see Section 2.2.2), so some fracture dipsets may appear to be deactivated while others are not. We can avoid this latter problem by also checking the total stress shadow volume for the fracture set ψ . This is independent of the dipset of the propagating fracture, and when it reaches 1 we can consider the entire fracture set to be deactivated.

- **Microfracture population singularity:** If the subcritical fracture propagation index $b < 2$, any microfracture can be extrapolated back to an initial time t_i when it had zero radius. If this time t_i is later than the start of the model, it is impossible to accurately calculate the cumulative density of microfractures of that size or smaller; in theory the model would contain an infinite number of microfractures of this size. Therefore, if we reach a point at which this is the case for microfractures large enough to be important (e.g. the upper boundary of the smallest bin used in the numerical integration) then we must abort the calculation. We can calculate t_i for as a function of microfracture radius r using equation A.2 from Appendix A of Welch et al. 2020. However this criteria will not apply in most cases as the subcritical fracture propagation index will be > 2 in most geologically realistic scenarios.

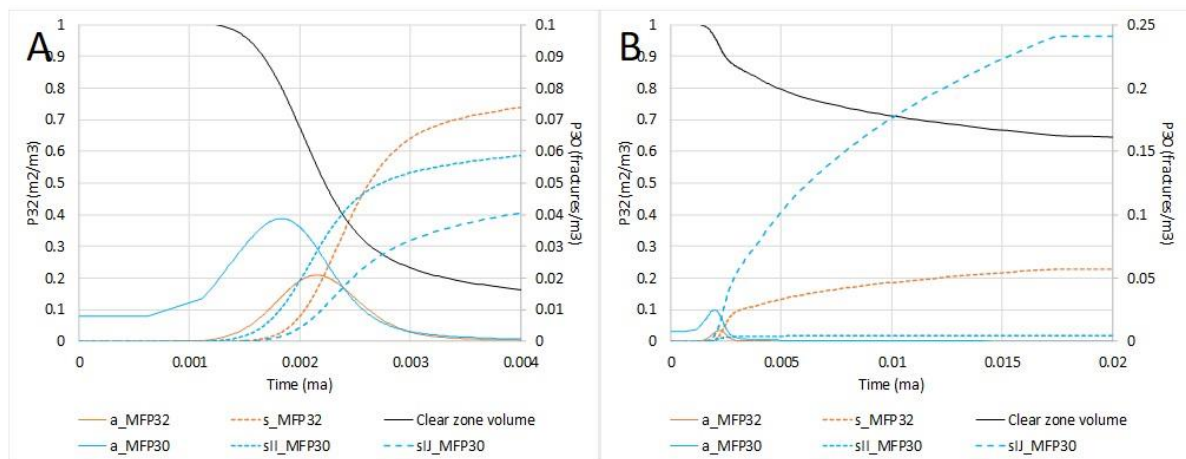


Figure 6: Typical evolution of macrofracture populations through time: (A) shows the primary fracture set and (B) shows the secondary fracture set. In the primary fracture set, we see an initial rapid rise in the number of active macrofractures as new macrofractures nucleate; however as macrofracture deactivation rate starts to rise, the population of active macrofractures reaches a peak before declining to negligible levels, while the population of static macrofractures continues to rise. However for the secondary set, macrofracture deactivation rates are much higher from the outset because of the high probability of intersecting primary macrofractures. Therefore the active secondary macrofracture population never reaches the level of the active primary macrofracture population, and many of the secondary macrofractures develop during the residual fracture stage.

4.1.2.2 Combining termination criteria

The termination criteria described in Section 4.1.2.1 can be applied in any combination. However a typical setup would be as follows:

- Calculation is terminated immediately if either of the static termination criteria are met, regardless of the state of the fracture sets. Model duration is set to the duration of the deformation episode, if known, otherwise to a maximum value (e.g. the age of the layer). Maximum number of timesteps is set high (e.g. 1000); if this termination criterion is triggered the model (including other termination criteria) should be modified and rerun.

- Fracture sets are considered to be deactivated if any of the dynamic termination criteria are met. The minimum proportion of active fractures and minimum ratio of current to peak historic active fractures should however be set low or not applied at all, to allow residual fractures to grow.
- If there are multiple dipsets within a fracture set, the minimum clear zone volume criterion should be tested for each dipset, and the fracture set should only be considered deactivated if the criterion is met for all *active* dipsets. However *inactive* dipsets, i.e. dipsets for which no fractures have nucleated, should not be tested.
- The calculation is terminated if all fracture sets are deactivated. The calculation should not be terminated if some sets are deactivated but other sets are not yet active (i.e. no fractures have yet nucleated). However to speed up the calculation, the deactivated sets should no longer be considered when determining the optimal timestep duration; this should allow the calculation to advance directly to the point at which the next fracture set becomes active.

4.2 Workflow to generate the explicit DFN

The conventional way of creating a Discrete Fracture Network (DFN) is to place fractures in random locations, and assign them a size and orientation using stochastic distribution functions based on observations from well data, seismic attribute analysis or outcrop analogues. However this method has two main drawbacks:

- The statistical properties of the actual fracture network are often poorly constrained: especially size, geometry and connectivity, but also density and orientation away from the wellbores.
- Even where the statistical properties of the actual fracture network are constrained, the stochastic method takes no account of the physical mechanisms of fracture nucleation and growth, so can result in a geologically unrealistic fracture network (for example fractures cutting across a thin ductile horizon that in fact acts as a barrier to fracture propagation, or a failure to match variation in fracture orientation around a larger structure such as a fold or salt dome).

We can generate more accurate and realistic DFNs based on the results of the implicit fracture modelling described in the Section 4.1. To do this, we nucleate seed fractures at random locations, at a rate described by the fracture nucleation rate equations (see Sections 3.3 and 4.2 of Welch et al. 2020), and then simulate the growth of those fractures through time, the equations for fracture propagation rate derived in Sections 3.2 and 4.1 of Welch et al. 2020. As the fractures grow, we must also check for fracture interaction and deactivate them where appropriate.

This technique can be applied to model the both microfractures and macrofractures explicitly. Since the algorithms used to determine fracture nucleation, propagation and interaction rates are consistent with the physical mechanisms of fracture nucleation and growth, the result will be a DFN that honours the geological and geomechanical data (for example fracture termination at layer boundaries or variation in fracture size and orientation in areas with different stress histories), even where there is no direct data to condition the DFN.

To build a DFN of an real geological layer, we should therefore first build an implicit fracture model of the layer, as described in Section 4.1. We will then use the same gridblocks and timesteps to generate the explicit DFN. It is helpful, however, to use several different spatial and temporal coordinate systems for the gridblocks and timesteps, as this will simplify the calculations required. We will examine these in Section 4.2.1. We will then present a complete workflow for generating a DFN, and explain how to use the various coordinate systems to perform the required calculations, in Section 4.2.2.

4.2.1 Coordinate systems

If we are generating an explicit DFN model comprising planar objects representing the fractures, we need a 3 dimensional coordinate system to position them in space, as well as a temporal coordinate to define the timing of various events (e.g. fracture nucleation and interaction). There are various ways we can define these, and different coordinate systems have advantages at different stages of the calculation. It is therefore convenient to use multiple coordinate systems, and to derive methods of switching between systems when necessary.

4.2.1.1 Spatial coordinate systems

As we have already stated, we will use the gridblocks that we defined when generating the implicit fracture model as the framework for the generating the explicit DFN model. This means that we will simulate the nucleation, propagation and interaction of fractures within each gridblock, and only at the end of the workflow will we combine these to create a global fracture network extending across the entire fractured layer. We can therefore choose either to define our coordinate system relative to specific gridblocks (a local coordinate system), or to define the location of the gridblocks relative to the coordinate system (a global coordinate system).

It is also often convenient to use a coordinate system defined relative to the fractures themselves. However the orientations of the fracture sets may vary between gridblocks, if the in situ stress orientation varies, so any such relative coordinate system must be a local coordinate system. On the other hand, fractures may also propagate across gridblock boundaries into adjacent gridblocks, so we must also have a global coordinate system capable of defining trajectories consistently across gridblock boundaries. We will therefore define three spatial coordinate systems: a global system (XYZ), a local system for each gridblock (uvw), and a local system relative to each fracture set (IJK).

Note that to generate an DFN using the workflow described here, it is an essential requirement that the gridblocks should be hexahedral (i.e. 6-sided polyhedra), and also that they should be contiguous (i.e. there must be no gaps between them); this is necessary to allow fractures to propagate across gridblock boundaries.

For most of the calculations in this section, we will also modify the gridblocks slightly by projecting the gridblock pillars to vertical. The pillars of a hexahedral gridblock are the four edges connecting the top and bottom surfaces, and in most geomodels they can take any orientation (and indeed they must be non-vertical to represent features such as inclined faults accurately). However it is easiest to simulate the lateral propagation of macrofractures on a single surface, corresponding to the centre of the fractured layer. We will therefore create “vertically-aligned” gridblocks by projecting the pillars to vertical lines, with their centrepoinsts as invariants (see *Figure 7*). In this way we can simulate the growth of the macrofractures as planes projected up and down from the central surface. At the end of the simulation, we can then project the corners of the fractures back onto the original gridblock edges. Note that the top, centre and bottom surfaces of the gridblock need not be horizontal, or even parallel.

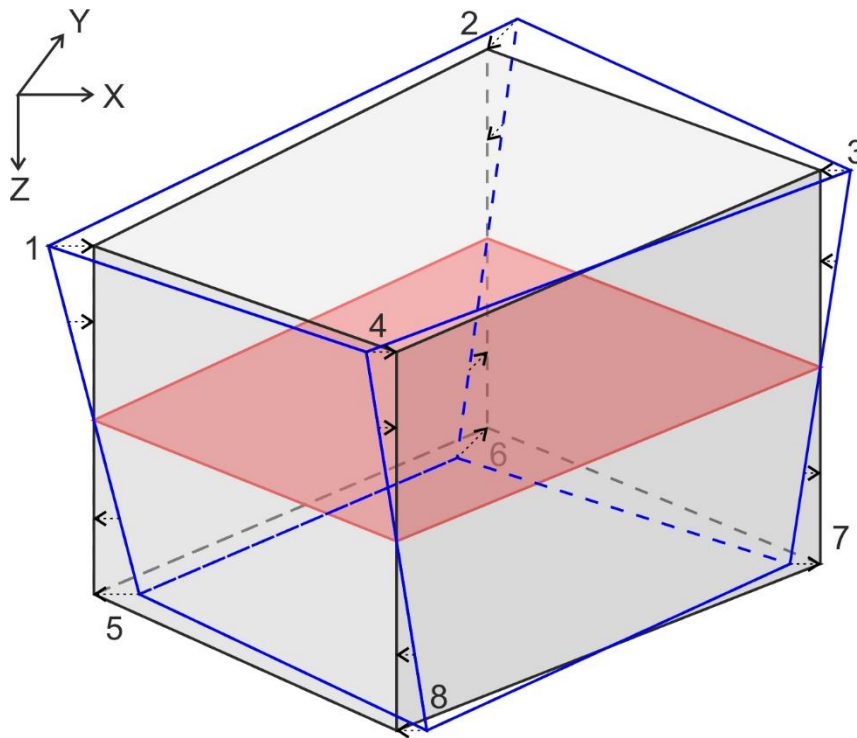


Figure 7: The original hexahedral gridblocks (blue outline) can be converted to “vertically aligned” gridblocks (black outline with grey shading) by projecting the gridblock pillars to vertical around their centrepoints. We can then simulate the propagation of macrofractures along a single surface representing the centre of the layer (red outline and shading). This surface is invariant in the vertical alignment transformation.

4.2.1.2 Global coordinates (XYZ)

The global coordinate system is defined to be consistent across the entire static geomodel, so that it can be used to define the location of any point in the model. It can therefore be used to define the geometry of the entire fractured layer (as top, centre and bottom surfaces), and the final global fracture network (i.e. the output DFN). It can also be used to define the location and size of the gridblocks themselves, by defining the position of the 8 cornerpoints for each gridblock. Note that, if the gridblocks are contiguous as stated above, then any cornerpoint of any gridblock should be identical to the corresponding cornerpoints of the adjacent gridblocks, when expressed in XYZ coordinates.

4.2.1.2.1 Defining a global coordinate system

It is common in static geomodelling to align the axes of the global coordinate system with the compass directions, so that the X coordinate represents distance to the east (negative X values represent distance to the west) and the Y coordinate represents distance to the north (negative Y values represent distance to the south). However there is sometimes an advantage in using a different alignment for the global coordinate axes: for example to align them with a linear geological structure (e.g. a fold or a basin axis), a preferred flow direction (e.g. a well alignment or waterflood direction), or even a data acquisition orientation (e.g. seismic inlines and crosslines). The workflow described here is valid for any alignment of the global coordinate axes, as long as the X and Y axes are horizontal and the Y axis is oriented 90° anticlockwise from the X axis.

The Z axis is always vertical in static geomodels, but in some models it is positive downwards (so the Z coordinate represents depth) whereas in others it is positive upwards (so the Z coordinate represents height, and depths are represented by negative Z values). Care should therefore be taken

to check the convention used in any particular geomodel. The equations given here assume that Z coordinates are positive downwards, so depth values are positive; however it is easy to adapt the equations to the opposite convention if required, by reversing the polarity of the Z coordinate terms.

The origin of the global coordinate system for static geomodels is often chosen to be at the nearest left hand corner of the model, relative to the X and Y axes. This ensures that the X and Y coordinates for all points in the model are positive, and have a large range (i.e. the length and width of the model is of the same order of magnitude as the absolute values of the coordinates). However it is also common to choose an origin to be consistent with some universal standard coordinate system (e.g. Universal Transverse Mercator or UTM coordinates). In this case the X and Y coordinates of points in the model may be negative, and more importantly they may have a narrow range – i.e. the length and width of the model may be much smaller than the absolute value of the X and Y coordinates of points within the model. In this situation, there is a risk that errors may be introduced by rounding of the coordinate values during calculation, and it may be necessary to “rebase” the coordinates to a local origin before running the workflow. The Z coordinate origin is generally taken as either mean sea level (MSL) or ground level (GL) for onshore geomodels.

4.2.1.2.2 Calculating gridblock volume

If we define the cornerpoints of the gridblock in XYZ coordinates, we can use them to calculate the gridblock volume V_{gr} . In a vertically aligned gridblock, this will be given by the horizontal gridblock area A_{gr} , times the mean gridblock height \bar{h} .

The horizontal gridblock area is calculated by projecting the centre surface of the gridblock vertically onto a horizontal plane, as shown in *Figure 8*. The horizontal projection of the centre surface of a hexahedral gridblock will be a quadrilateral, and if the gridblock is vertically aligned, it will be identical to the horizontal projection of the top and bottom gridblock surfaces. The area of this quadrilateral can best be calculated by splitting it into two triangles. We then need to calculate the lengths of the four sides of the quadrilateral $|12|$, $|14|$, $|23|$ and $|43|$, and two opposite internal angles $\angle 1$ and $\angle 3$:

$$\begin{aligned}
 [123] \quad |\vec{12}| &= \sqrt{(X_2 - X_1)^2 + (Y_2 - Y_1)^2} \\
 |\vec{14}| &= \sqrt{(X_4 - X_1)^2 + (Y_4 - Y_1)^2} \\
 |\vec{23}| &= \sqrt{(X_3 - X_2)^2 + (Y_3 - Y_2)^2} \\
 |\vec{43}| &= \sqrt{(X_3 - X_4)^2 + (Y_3 - Y_4)^2} \\
 \angle 1 &= \cos^{-1} \left(\frac{(X_2 - X_1)(X_4 - X_1) + (Y_2 - Y_1)(Y_4 - Y_1)}{\sqrt{(X_2 - X_1)^2 + (Y_2 - Y_1)^2} \sqrt{(X_4 - X_1)^2 + (Y_4 - Y_1)^2}} \right) \\
 \angle 3 &= \cos^{-1} \left(\frac{(X_3 - X_2)(X_3 - X_4) + (Y_3 - Y_2)(Y_3 - Y_4)}{\sqrt{(X_3 - X_2)^2 + (Y_3 - Y_2)^2} \sqrt{(X_3 - X_4)^2 + (Y_3 - Y_4)^2}} \right)
 \end{aligned}$$

The total area of the projected surface A_{gr} is thus given by

$$[124] \quad A_{gr} = (|\vec{12}| \times |\vec{14}| \times \sin \angle 1) + (|\vec{23}| \times |\vec{43}| \times \sin \angle 3)$$

The mean height of the gridblock \bar{h} is given by

$$[125] \quad Z_p \bar{h} = \frac{1}{4}(h_1 + h_2 + h_3 + h_4) = \frac{1}{4}((Z_5 - Z_1) + (Z_6 - Z_2) + (Z_3 - Z_7) + (Z_8 - Z_4))$$

so the total gridblock volume V_{gr} is

$$[126] \quad \mathbb{V}_{gr} = \frac{1}{4} \left((|\vec{12}| \times |\vec{14}| \times \sin \angle 1) + (|\vec{23}| \times |\vec{43}| \times \sin \angle 3) \right) ((Z_5 - Z_1) + (Z_6 - Z_2) + (Z_3 - Z_7) + (Z_8 - Z_4))$$

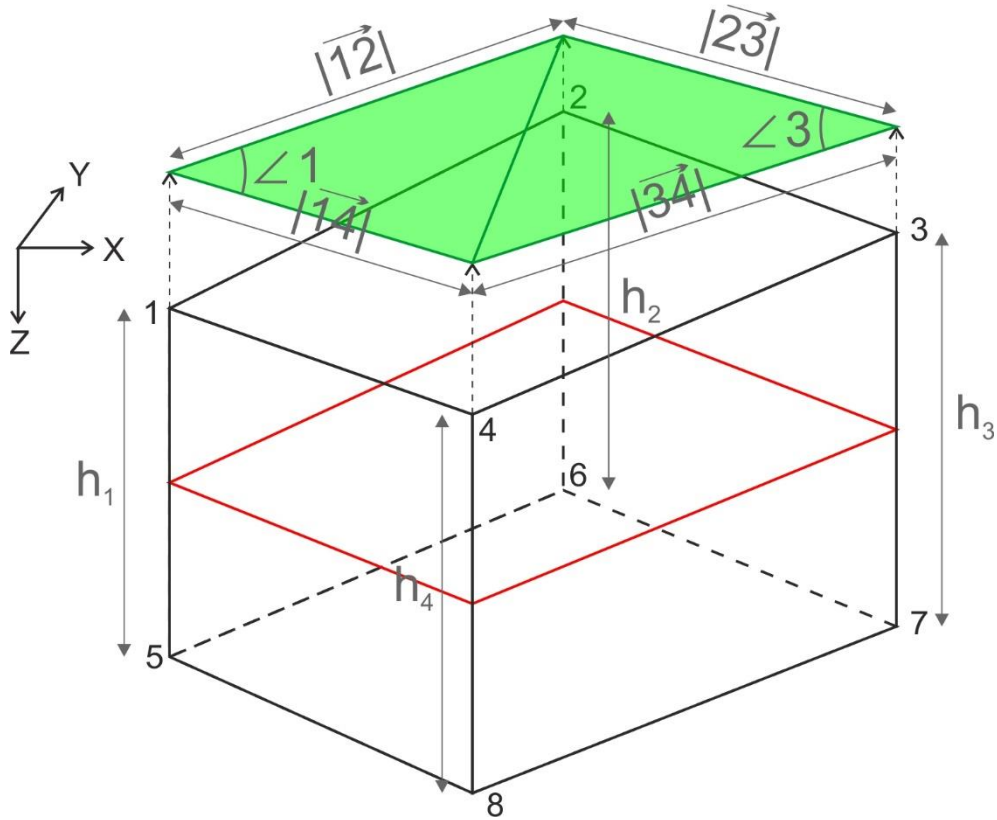


Figure 8: Schematic illustration showing how to calculate the volume of a vertically aligned gridblock. The green surface represents the centre surface of the gridblock (shown in red) projected onto a horizontal plane.

4.2.1.3 Local coordinates relative to the gridblock (uvw)

It is also possible to define a local coordinate system relative to the boundaries of a specific gridblock, as shown in Figure 9. We call this the uvw coordinate system. For a vertically aligned gridblock (i.e. with vertical pillars):

- u represents the relative distance from the left hand gridblock boundary to the right hand gridblock boundary. $u=0$ for a point lying on the left hand boundary, $u=1$ for a point lying on the right hand boundary, and $u=0.5$ for a point halfway between the two boundaries.
- v represents the relative distance from the near gridblock boundary to the far gridblock boundary. $v=0$ for a point lying on the near boundary, $v=1$ for a point lying on the far boundary, and $v=0.5$ for a point halfway between the two boundaries.
- w represents the relative distance from the top gridblock boundary to the bottom gridblock boundary. $w=0$ for a point lying on the top boundary, $w=1$ for a point lying on the bottom boundary, and $w=0.5$ for a point halfway between the two boundaries. Note that this assumes the global Z axis is positive downwards, and must be swapped if the Z axis is positive upwards.

Note that the u and v axes of the uvw coordinate system will not be orthogonal, and their orientation will vary spatially, unless the sides of the gridblock are all rectangular (i.e. the gridblock is a rectangular prism).

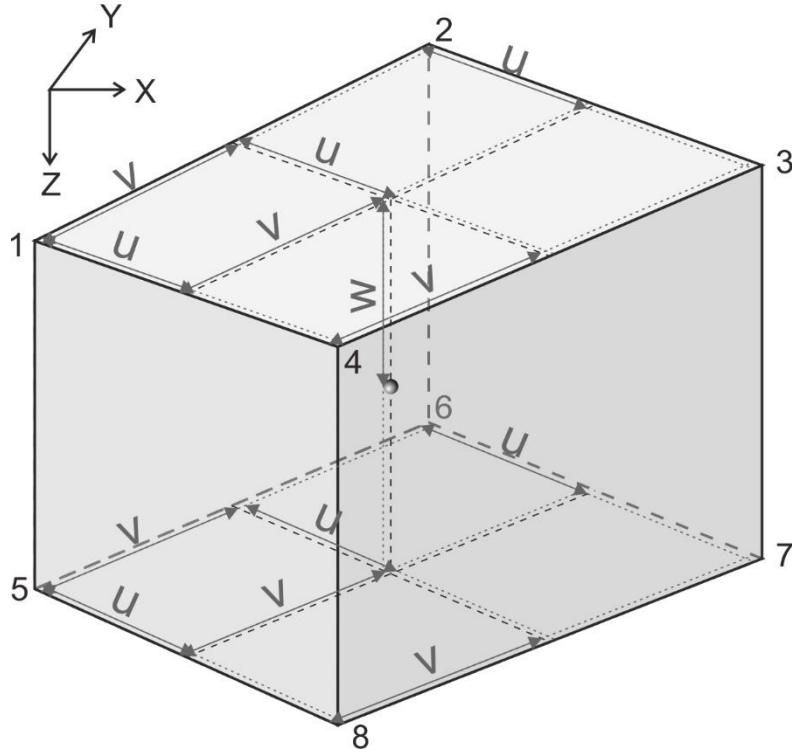


Figure 9: Schematic illustration showing the local uvw coordinate system relative to the gridblock, and to the global XYZ coordinate system, for a vertically aligned gridblock.

4.2.1.3.1 Converting from uvw to XYZ coordinates

If the coordinates of the eight gridblock cornerpoints and four pillar centrepoints illustrated in Figure 9 are labelled $X_1, Y_1, Z_1, X_2, Y_2, Z_2$, etc., then by definition

$$\begin{aligned}
 [127] \quad X_9, Y_9, Z_9 &= \frac{X_1 + X_5}{2}, \frac{Y_1 + Y_5}{2}, \frac{Z_1 + Z_5}{2} \\
 X_{10}, Y_{10}, Z_{10} &= \frac{X_2 + X_6}{2}, \frac{Y_2 + Y_6}{2}, \frac{Z_2 + Z_6}{2} \\
 X_{11}, Y_{11}, Z_{11} &= \frac{X_3 + X_7}{2}, \frac{Y_3 + Y_7}{2}, \frac{Z_3 + Z_7}{2} \\
 X_{12}, Y_{12}, Z_{12} &= \frac{X_4 + X_8}{2}, \frac{Y_4 + Y_8}{2}, \frac{Z_4 + Z_8}{2}
 \end{aligned}$$

If the pillars are vertical

$$\begin{aligned}
 [128] \quad X_1 &= X_5 = X_9 \\
 Y_1 &= Y_5 = Y_9 \\
 X_2 &= X_6 = X_{10} \\
 Y_2 &= Y_6 = Y_{10} \\
 X_3 &= X_7 = X_{11} \\
 Y_3 &= Y_7 = Y_{11} \\
 X_4 &= X_8 = X_{12} \\
 Y_4 &= Y_8 = Y_{12}
 \end{aligned}$$

and if they are not already vertical, they can be projected onto the vertical by setting $X_1, X_2, X_3, X_4, X_5, X_6, X_7, X_8, Y_1, Y_2, Y_3, Y_4, Y_5, Y_6, Y_7$ and Y_8 to the values given in [128]; thus we can convert any hexahedral gridblock to a vertically aligned gridblock.

The X and Y coordinates of any point p, X_p and Y_p , are simply a function of the u and v coordinates u_p and v_p :

$$[129] \quad X_p = (1 - u_p)(1 - v_p)X_1 + (1 - u_p)v_pX_2 + u_pv_pX_3 + u_p(1 - v_p)X_4$$

$$Y_p = (1 - u_p)(1 - v_p)Y_1 + (1 - u_p)v_pY_2 + u_pv_pY_3 + u_p(1 - v_p)Y_4$$

The Z coordinates of the top and bottom surfaces of the gridblock directly above and below point p, $Z_{top,p}$ and $Z_{bottom,p}$ will be given by

$$[130] \quad Z_{top,p} = (1 - u_p)(1 - v_p)Z_1 + (1 - u_p)v_pZ_2 + u_pv_pZ_3 + u_p(1 - v_p)Z_4$$

$$Z_{bottom,p} = (1 - u_p)(1 - v_p)Z_5 + (1 - u_p)v_pZ_6 + u_pv_pZ_7 + u_p(1 - v_p)Z_8$$

and the Z coordinate of p, Z_p , can be calculated from $Z_{top,p}$, $Z_{bottom,p}$, and the w coordinate of p, w_p :

$$[131] \quad Z_p = Z_{top,p} + w_p(Z_{bottom,p} - Z_{top,p})$$

$$= (1 - u_p)(1 - v_p)((1 - w_p)Z_1 + w_pZ_5) + (1 - u_p)v_p((1 - w_p)Z_2 + w_pZ_6)$$

$$+ u_pv_p((1 - w_p)Z_3 + w_pZ_7) + u_p(1 - v_p)((1 - w_p)Z_4 + w_pZ_8)$$

4.2.1.3.2 Converting from XYZ to uvw coordinates

Converting from XYZ to uvw coordinates is slightly more complicated. From [129] we can see that

$$[132] \quad X_1 - u_pX_1 - v_pX_1 + u_pv_pX_1 + v_pX_2 - u_pv_pX_2 + u_pv_pX_3 + u_pX_4 - u_pv_pX_4 - X_p = 0$$

$$\Rightarrow (X_1 - X_p) + (X_4 - X_1)u_p + (X_2 - X_1)v_p + (X_1 - X_2 + X_3 - X_4)u_pv_p = 0$$

and

$$[133] \quad Y_1 - u_pY_1 - v_pY_1 + u_pv_pY_1 + v_pY_2 - u_pv_pY_2 + u_pv_pY_3 + u_pY_4 - u_pv_pY_4 - Y_p = 0$$

$$\Rightarrow (Y_1 - Y_p) + (Y_4 - Y_1)u_p + (Y_2 - Y_1)v_p + (Y_1 - Y_2 + Y_3 - Y_4)u_pv_p = 0$$

Therefore

$$[134] \quad u_p = \frac{(X_p - X_1) - (X_2 - X_1)v_p}{(X_4 - X_1) + (X_1 - X_2 + X_3 - X_4)v_p} = \frac{(Y_p - Y_1) - (Y_2 - Y_1)v_p}{(Y_4 - Y_1) + (Y_1 - Y_2 + Y_3 - Y_4)v_p}$$

$$[135] \quad v_p = \frac{(X_p - X_1) - (X_4 - X_1)u_p}{(X_2 - X_1) + (X_1 - X_2 + X_3 - X_4)u_p} = \frac{(Y_p - Y_1) - (Y_4 - Y_1)u_p}{(Y_2 - Y_1) + (Y_1 - Y_2 + Y_3 - Y_4)u_p}$$

If we define

$$[136] \quad X_a = (X_p - X_1) \quad Y_a = (Y_p - Y_1)$$

$$X_b = (X_2 - X_1) \quad Y_b = (Y_2 - Y_1)$$

$$X_c = (X_4 - X_1) \quad Y_c = (Y_4 - Y_1)$$

$$X_d = (X_1 - X_2 + X_3 - X_4) \quad Y_d = (Y_1 - Y_2 + Y_3 - Y_4)$$

then we can write [135] as

$$\begin{aligned}
[137] \quad v_p &= \frac{X_a - X_c u_p}{X_b + X_d u_p} = \frac{Y_a - Y_c u_p}{Y_b + Y_d u_p} \\
&\Rightarrow (X_a - X_c u_p)(Y_b + Y_d u_p) = (Y_a - Y_c u_p)(X_b + X_d u_p) \\
&\Rightarrow X_a Y_b - X_c Y_b u_p + X_a Y_d u_p - X_c Y_d u_p^2 - Y_a X_b + Y_c X_b u_p - Y_a X_d u_p + Y_c X_d u_p^2 = 0 \\
&\Rightarrow (X_d Y_c - X_c Y_d) u_p^2 + (X_a Y_d - X_d Y_a + X_b Y_c - X_c Y_b) u_p + (X_a Y_b - Y_a X_b) = 0
\end{aligned}$$

[137] can be solved using the quadratic formula:

$$[138] \quad u_p = \frac{-B_u \pm \sqrt{B_u^2 - 4A_u C_u}}{2A_u}$$

where

$$\begin{aligned}
[139] \quad A_u &= (X_d Y_c - X_c Y_d) \\
B_u &= (X_a Y_d - X_d Y_a + X_b Y_c - X_c Y_b) \\
C_u &= (X_a Y_b - X_b Y_a)
\end{aligned}$$

Similarly we can write [134] as

$$[140] \quad u_p = \frac{X_a - X_b v_p}{X_c + X_d v_p} = \frac{Y_a - Y_b v_p}{Y_c + Y_d v_p}$$

and, since we know u_p , we can calculate v_p directly from [140]. Finally, w_p can be calculated from $Z_{top,p}$, $Z_{bottom,p}$, and the Z coordinate of p, Z_p :

$$[141] \quad w_p = \frac{Z_p - Z_{top,p}}{Z_{bottom,p} - Z_{top,p}} = \frac{Z_p - (1-u_p)(1-v_p)Z_1 - (1-u_p)v_p Z_2 - u_p v_p Z_3 - u_p(1-v_p)Z_4}{(1-u_p)(1-v_p)(Z_5 - Z_1) + (1-u_p)v_p(Z_6 - Z_2) + u_p v_p(Z_7 - Z_3) + u_p(1-v_p)(Z_8 - Z_4)}$$

There are three important points to note:

- If the sides of the gridblock are rectangular (i.e. the gridblock is a rectangular prism) then $X_d=Y_d=0$ and hence $A_u=0$. In this case, u_p can be calculated using a linear formula:

$$[142] \quad u_p = -\frac{C_u}{B_u} = -\frac{(X_a Y_b - Y_a X_b)}{(X_b Y_c - X_c Y_b)} = \frac{(X_2 - X_1)(Y_p - Y_1) - (X_p - X_1)(Y_2 - Y_1)}{(X_2 - X_1)(Y_4 - Y_1) - (X_4 - X_1)(Y_2 - Y_1)}$$

- If the sides of the gridblock are not rectangular, there will in most cases be two solutions to the quadratic formula, and hence two potential sets of u and v coordinates. Although both sets of uv coordinates will convert back to the same XYZ coordinates through [129], only one set represents the position of the point p on lines linking the gridblock surfaces, while the other represents the position of the point p on lines projected back from outside the gridblock. We must therefore use the set of uv coordinates with values between 0 and 1 (if the point lies within the gridblock), or closest to 0 and 1 (if the point lies outside the gridblock).
- There will always be at least one real solution to [138] for points lying within the gridblock. However for points lying outside the gridblock, there may be no real solutions to [138]. Therefore, while it is possible to define uvw coordinates for some points outside the gridblock, it is not possible to define uvw coordinates for all points outside the gridblock.

4.2.1.3.3 Uses of the uvw coordinate system

One common use of the uvw coordinate system is to interpolate the height of the gridblock at any point p within the gridblock, h_p . This will be given by

$$\begin{aligned}
[143] \quad h_p &= (Z_{bottom,p} - Z_{top,p}) \\
&= (1 - u_p)(1 - v_p)(Z_5 - Z_1) + (1 - u_p)v_p(Z_6 - Z_2) \\
&\quad + u_pv_p(Z_7 - Z_3) + u_p(1 - v_p)(Z_8 - Z_4)
\end{aligned}$$

value of spatially variable properties at any point within a gridblock, if we know its values at the gridblock cornerpoints. If we take a property Q, with values at the eight gridblock cornerpoints given by Q_1, Q_2 , etc., then the value of Q at any point p within the gridblock, Q_p , will be given by

$$\begin{aligned}
[144] \quad Q_p &= (1 - u_p)(1 - v_p)((1 - w_p)Q_1 + w_pQ_5) + (1 - u_p)v_p((1 - w_p)Q_2 + w_pQ_6) \\
&\quad + u_pv_p((1 - w_p)Q_3 + w_pQ_7) + u_p(1 - v_p)((1 - w_p)Q_4 + w_pQ_8)
\end{aligned}$$

assuming Q varies linearly within the gridblock.

Another use of the uvw coordinate system is to determine whether a point specified in XYZ coordinates lies within the gridblock or not. As we have seen, it is possible to refer to some (but not all) points outside the gridblock using this coordinate system, but only points inside the gridblock will have uvw coordinates within the range 0 to 1. It is therefore very easy to test whether any point p in the geomodel lies within the gridblock or not, by converting its XYZ coordinates to uvw coordinates as described in Section 4.2.1.3.2, and testing whether $0 \leq u_p \leq 1$, $0 \leq v_p \leq 1$ and $0 \leq w_p \leq 1$. If it is not possible to convert the XYZ coordinates of p to uvw coordinates, it must lie outside the gridblock.

A third use of the uvw coordinate system is to generate a point at a random location within a gridblock. There are two methods of doing this – a quick but potentially biased method, and a slower but exact method.

For the quick method, we simply define three random numbers between 0 and 1, which we can label R_1, R_2 and R_3 , and use these to determine the position of the random point p along the u, v and w axes:

$$[145] \quad \begin{pmatrix} u_p \\ v_p \\ w_p \end{pmatrix} = \begin{pmatrix} R_1 \\ R_2 \\ R_3 \end{pmatrix}$$

Since u_p, v_p and w_p must lie between 0 and 1, the random point must lie within the gridblock, and we can convert the uvw coordinates into XYZ coordinates as described in Section 4.2.1.3.1. However if the gridblock is very irregular in shape, the location of this point will not be truly random but may be biased towards certain corners of the gridblock. This is illustrated in *Figure 10* – a random point generated using the quick method will have an equal probability of lying in the yellow quadrant and the pink quadrant, even though the pink quadrant covers a much larger volume than the yellow quadrant.

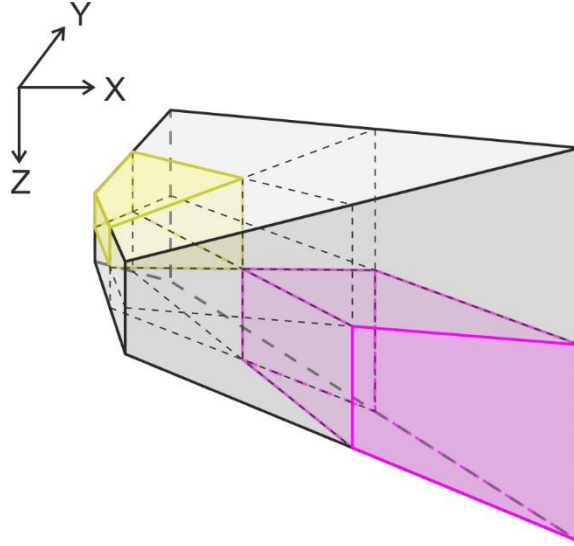


Figure 10: Relative size of two quadrants in a very irregular gridblock. If a random point is generated using the quick method, it will have an equal probability of lying in the yellow quadrant and the pink quadrant, even though the pink quadrant covers a much larger volume than the yellow quadrant.

This bias occurs because the uvw axes are not orthogonal, and can vary in orientation spatially. We can therefore avoid this bias if we generate a random point in XYZ coordinates instead. Again we use three random numbers between 0 and 1, R_1 , R_2 and R_3 , to determine the location of the random point p along the three axes, but this time we must scale them to cover the total range of X, Y and Z values within the gridblock:

$$[146] \quad \begin{pmatrix} X_p \\ Y_p \\ Z_p \end{pmatrix} = \begin{pmatrix} (1 - R_1)\min\{X_1, X_2, X_5, X_6\} + R_1 \max\{X_3, X_4, X_7, X_8\} \\ (1 - R_2)\min\{Y_1, Y_4, Y_5, Y_8\} + R_2 \max\{Y_2, Y_3, Y_6, Y_7\} \\ (1 - R_3)\min\{Z_1, Z_2, Z_3, Z_4\} + R_3 \max\{Z_5, Z_6, Z_7, Z_8\} \end{pmatrix}$$

However not all such points will lie within the gridblock. We must therefore test whether the randomly selected point p does indeed lie within the gridblock by converting to uvw coordinates, as described in Section 4.2.1.3.2, and testing whether $0 \leq u_p \leq 1$, $0 \leq v_p \leq 1$ and $0 \leq w_p \leq 1$. If point p does not lie within the gridblock, we must discard the randomly calculated coordinates and repeat the procedure until we have generated a point that does lie within the gridblock. If the gridblock is very irregular in shape, this may take many iterations, but the resulting point will be truly randomly located within the gridblock.

4.2.1.4 Local coordinates relative to the fractures (IJK)

For some calculations involving specific fractures, it is useful to use a coordinate system defined relative to the orientation of the fractures themselves. Therefore we can define an IJK coordinate system, relative to each fracture set within each gridblock (see Figure 11). In this system the I axis is parallel to the fracture strike direction, and the J axis is 90° clockwise from the I axis (which will correspond either to the azimuth or the opposite to the azimuth of any individual fracture). Note that the I axis can therefore be chosen in one of two opposite directions. The choice is arbitrary, but it is also advisable, although not essential, to ensure consistency between adjacent gridblocks – i.e. if the I axis of the fracture set perpendicular to h_{min} is oriented approximately north in one gridblock, it should be chosen to be approximately north rather than approximately south in the adjacent

gridblocks, assuming that the variation in principal stress orientation between the adjacent gridblocks is not too large.

The K axis is again assumed to be vertical and positive downwards; however the datum is taken as the centre surface of the gridblock. In most gridblocks the centre surface will not be horizontal, so the IJK coordinate system has the effect of shearing the gridblock vertically to bring the centre surface to the horizontal. Lengths and angles measured in the IJK coordinate system will therefore not correspond exactly with lengths and angles measured in the XYZ coordinate system, although if the dip of the layer is relatively low (<c.20°) the discrepancies should be small.

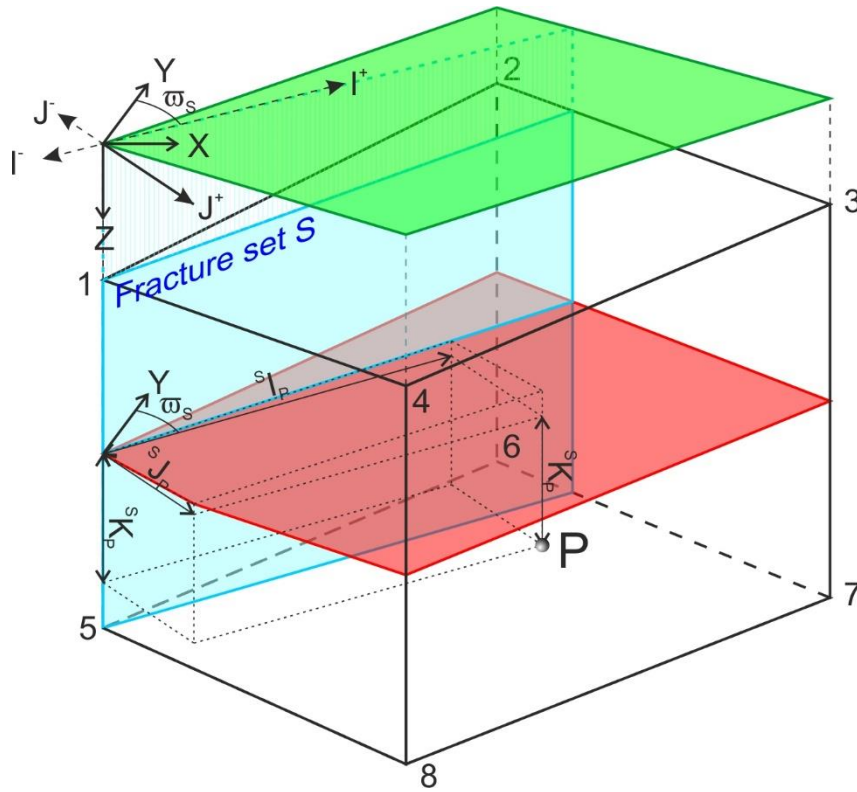


Figure 11: Schematic illustration showing the local IJK coordinate system relative to the fracture set, the gridblock, and to the global XYZ coordinate system. Note that, while the I and J coordinates are measured on a horizontal plane (shown in green), the K coordinate (depth) is measured from the centre surface of the gridblock (shown in red), which is not horizontal but dips gently towards the near corner (cornerpoints 4 and 8).

4.2.1.4.1 Converting from IJK to XYZ coordinates

If fracture set S has strike ϖ_s , then a point p, with I and J coordinates relative to fracture set S given by sI_p and sJ_p , will have X and Y coordinates X_p and Y_p given by

$$[147] \quad X_p = ^sI_p \sin \varpi_s + ^sJ_p \cos \varpi_s$$

$$Y_p = ^sI_p \cos \varpi_s - ^sJ_p \sin \varpi_s$$

The Z coordinate Z_p will be given by

$$[148] \quad Z_p = \frac{Z_{bottom,p} + Z_{top,p}}{2} + ^sK_p$$

Note that this transformation is valid regardless of the location of the origin; however if the origin is a long way from the gridblock, then the range of IJK coordinates of points in the gridblock may be very different to the range of XYZ coordinates of points in the gridblock.

4.2.1.4.2 Converting from XYZ to IJK coordinates

Similarly, if fracture set S has strike ϖ_s , then a point p, with global X and Y coordinates given by X_p and Y_p , will have I and J coordinates relative to fracture set S, SI_p and SJ_p , given by

$$\begin{aligned} [149] \quad ^SI_p &= X_p \sin \varpi_s + Y_p \cos \varpi_s \\ ^SJ_p &= X_p \cos \varpi_s - Y_p \sin \varpi_s \end{aligned}$$

The K coordinate SK_p will be given by

$$[150] \quad ^SK_p = Z_p - \frac{Z_{bottom,p} + Z_{top,p}}{2}$$

4.2.1.4.3 Converting between IJK coordinates of different fracture sets

We can also convert directly between the IJK coordinates of different fracture sets. If we have two fracture sets S1 and S2, where fracture set S1 has strike ϖ_{S1} and fracture set S2 has strike ϖ_{S2} , then a point p, with I and J coordinates relative to fracture set S1 given by $^{S1}I_p$ and $^{S1}J_p$, will have I and J coordinates relative to fracture set S2, $^{S2}I_p$ and $^{S2}J_p$, given by

$$\begin{aligned} [151] \quad ^{S2}I_p &= X_p \sin \varpi_{S2} + Y_p \cos \varpi_{S2} = (^{S1}I_p \sin \varpi_{S1} + ^{S1}J_p \cos \varpi_{S1}) \sin \varpi_{S2} + (^{S1}I_p \cos \varpi_{S1} - ^{S1}J_p \sin \varpi_{S1}) \cos \varpi_{S2} \\ &= ^{S1}I_p (\sin \varpi_{S1} \sin \varpi_{S2} + \cos \varpi_{S1} \cos \varpi_{S2}) + ^{S1}J_p (\cos \varpi_{S1} \sin \varpi_{S2} - \sin \varpi_{S1} \cos \varpi_{S2}) \\ &= ^{S1}I_p \cos(\varpi_{S2} - \varpi_{S1}) + ^{S1}J_p \sin(\varpi_{S2} - \varpi_{S1}) \\ ^{S2}J_p &= X_p \cos \varpi_{S2} - Y_p \sin \varpi_{S2} = (^{S1}I_p \sin \varpi_{S1} + ^{S1}J_p \cos \varpi_{S1}) \cos \varpi_{S2} - (^{S1}I_p \cos \varpi_{S1} - ^{S1}J_p \sin \varpi_{S1}) \sin \varpi_{S2} \\ &= ^{S1}I_p (\sin \varpi_{S1} \cos \varpi_{S2} - \cos \varpi_{S1} \sin \varpi_{S2}) + ^{S1}J_p (\sin \varpi_{S1} \sin \varpi_{S2} + \cos \varpi_{S1} \cos \varpi_{S2}) \\ &= -^{S1}I_p \sin(\varpi_{S2} - \varpi_{S1}) + ^{S1}J_p \cos(\varpi_{S2} - \varpi_{S1}) \end{aligned}$$

The K coordinate will be referenced to the same datum point so will not change

$$[152] \quad ^{S2}K_p = ^{S1}K_p$$

Note that if we have two fracture hmin and hmax sets striking perpendicular to σ_{hmin} and σ_{hmax} respectively such that $\varpi_{hmax} = \varpi_{hmin} + 90^\circ$, then

$$\begin{aligned} [153] \quad ^{hmax}I_p &= ^{hmin}I_p \cos(\varpi_{hmax} - \varpi_{hmin}) + ^{hmin}J_p \sin(\varpi_{hmax} - \varpi_{hmin}) = ^{hmin}J_p \\ ^{hmax}J_p &= -^{hmin}I_p \sin(\varpi_{hmax} - \varpi_{hmin}) + ^{hmin}J_p \cos(\varpi_{hmax} - \varpi_{hmin}) = -^{hmin}I_p \end{aligned}$$

and similarly

$$\begin{aligned} [154] \quad ^{hmin}I_p &= ^{hmax}I_p \cos(\varpi_{hmin} - \varpi_{hmax}) + ^{hmax}J_p \sin(\varpi_{hmin} - \varpi_{hmax}) = -^{hmax}J_p \\ ^{hmin}J_p &= -^{hmax}I_p \sin(\varpi_{hmin} - \varpi_{hmax}) + ^{hmax}J_p \cos(\varpi_{hmin} - \varpi_{hmax}) = ^{hmax}I_p \end{aligned}$$

Thus it is very simple to use IJK coordinates to calculate intersection points between orthogonal fracture sets.

4.2.1.4.4 Defining individual fractures using IJK coordinates

Because they are defined relative to the orientation of the fracture sets, we can use the IJK coordinates to define the location and geometry of individual fractures in the DFN much more efficiently than we can using XYZ coordinates. All fractures in a given dipset m will have the same dip ω_m (typically 90° for Mode 1 dilatant fractures and between 45° and 75° for Mode 2 shear fractures, depending on the friction coefficient μ_{fr}). The dip azimuth of fractures will always be parallel to the J axis, but may be in the positive J direction or in the negative J direction with equal probability.

Since microfractures are circular and planar, the location and geometry of any specific microfracture in set S , ${}^S\mu F_n$, can be fully described by specifying the location of the centre, in IJK coordinates, the fracture radius r , the fracture dipset m (and hence dip ω_m), and the dip direction j (either J^+ or J^-):

$$[155] \quad {}^S\mu F_n = ({}^SI, {}^SJ, {}^SK, r, m, j)$$

Note that, as discussed in Welch et al. (2020), we assume that dipping microfractures have a circular profile when projected onto a vertical plane; their actual shape will be slightly elliptical.

Individual macrofractures may extend across several gridblocks, and their orientation may vary between gridblocks. Furthermore, since the macrofracture tips propagate (and terminate) independently, we must simulate the propagation of macrofracture as two half-macrofractures, each extending from the macrofracture nucleation point to one of the tips (see Chapter 5 of Welch et al. 2020). We can thus define any specific macrofracture in set S , SMF_n , as a series of contiguous planar segments ${}^SMF_{np}$, with segment boundaries at the nucleation point, the gridblock boundaries and the macrofracture tips.

Since, by definition, the layer-bound macrofractures extend from the top surface to the bottom surface, we can define each segment in terms of just two nodes located on the centre surface of the gridblock. Furthermore, since (again by definition) the segment must strike parallel to the I axis, the J coordinates of each node will be the same, so we need specify only three coordinates to define the location and geometry of the segment: a minimum I coordinate (I^-), a maximum I coordinate (I^+) and a J coordinate. We must of course specify the fracture dipset m (and hence dip ω_m), and the dip direction j (either J^+ or J^-), but we must also specify whether the segment is still actively propagating, and if so in which direction. We can do this by specifying an active node a , which may be either the I^+ node, the I^- node, or neither ($a=0$). The macrofracture SMF_n can thus be fully described by:

$$[156] \quad {}^SMF_n = \{ {}^SMF_{n_1}, {}^SMF_{n_2}, \dots, {}^SMF_{n_p}, \dots \}$$

$${}^SMF_{np} = (gr_p, {}^SI^-, {}^SI^+, {}^SJ, m, j, a)$$

where gr_p denotes the specific gridblock that the segment p lies within. The length of each segment L_{np} will be given by

$$[157] \quad L_{np} = {}^SI^+ - {}^SI^-$$

4.2.1.5 Temporal coordinate systems

In the workflow used to generate the implicit fracture model, the timestep durations are determined dynamically, in response to the growth of the fracture network within the gridblock. The timesteps within different gridblocks will therefore not be aligned. Since fractures can propagate across gridblock boundaries, we must therefore have a universal temporal coordinate (real time) which is the same for all gridblocks.

However the fracture driving stress, and hence the macrofracture propagation rate, may not be constant within each timestep. We resolve this problem by calculating a “weighted” average fracture driving stress that can be used as a constant in the fracture propagation equations for the entire timestep. If we want to simulate specific events that occur during the timestep (e.g. the nucleation or interaction of specific fractures), it is also useful to define a “weighted” time coordinate \mathcal{T} , such that a macrofracture will always propagate the same distance during the same duration of “weighted” time, regardless of whether this is near the start of the timestep or near the end.

4.2.1.6 Real time t

The real time simply represents physical time elapsed since the start of the model. It is therefore consistent across all gridblocks, timesteps and fracture sets and can be used to ascertain the relative order of events occurring in different gridblocks, including the start and end times of each timestep. It is advisable, although not essential, to set the origin ($t=0$) at the start of the deformation episode rather than at the present time, so that the real time coordinates of events in the model are all positive.

One problem that arises using real time coordinates is the extreme range of timescales between different processes when simulating fracture propagation. Critical fracture propagation can occur at such rapid rates (c.2000m/s) that a fracture may propagate across an entire gridblock in less than a second, whereas geological processes such as strain accumulation may occur over millions of years. Care must therefore be taken that rounding errors do not lead to inaccurate results when calculating times and rates. It is advisable to use SI units (s) for time, but it may be necessary to rebase the time coordinates (i.e. reset the origin time to the start of the event in question) for specific calculations.

4.2.1.7 Weighted time \mathcal{T}

As noted above, the weighted time \mathcal{T} is defined such that a macrofracture will always propagate the same distance during the same duration of “weighted” time. It is therefore dependent on the value of the fracture driving stress as well as the real time, and thus is specific to each fracture dipset within each gridblock.

If the propagation rate of a fracture of set S and dipset m during timestep N is given by $d\ell/dt_{Sm,N}$, then we can convert between the real time t_n of any instant within timestep N , and the weighted time relative to fracture set S dipset m fractures, \mathcal{T}_{Sm} , using

$$[158] \quad \mathcal{T}_{Sm} = \frac{d\ell}{dt_{Sm,N}} (t_n - t_{N-1}) = \alpha_{MF} \sigma_{dSm,N}^b (t_n - t_{N-1}) = A \left(\frac{\sqrt{2h} \sigma_{dSm,N}}{\sqrt{\pi} K_C} \right)^b (t_n - t_{N-1})$$

and

$$[159] \quad t_n = t_{N-1} + \frac{\mathcal{T}_{Sm}}{\frac{d\ell}{dt_{Sm,N}}} = t_{N-1} + \frac{\mathcal{T}_{Sm}}{\alpha_{MF} \sigma_{dSm,N}^b} = t_{N-1} + \frac{\mathcal{T}_{Sm}}{A} \left(\frac{\sqrt{\pi} K_C}{\sqrt{2h} \sigma_{dSm,N}} \right)^b$$

Note that we set the base of the weighted time coordinate (the instant when $\mathcal{T}=0$) to be at the start of the timestep, t_{N-1} .

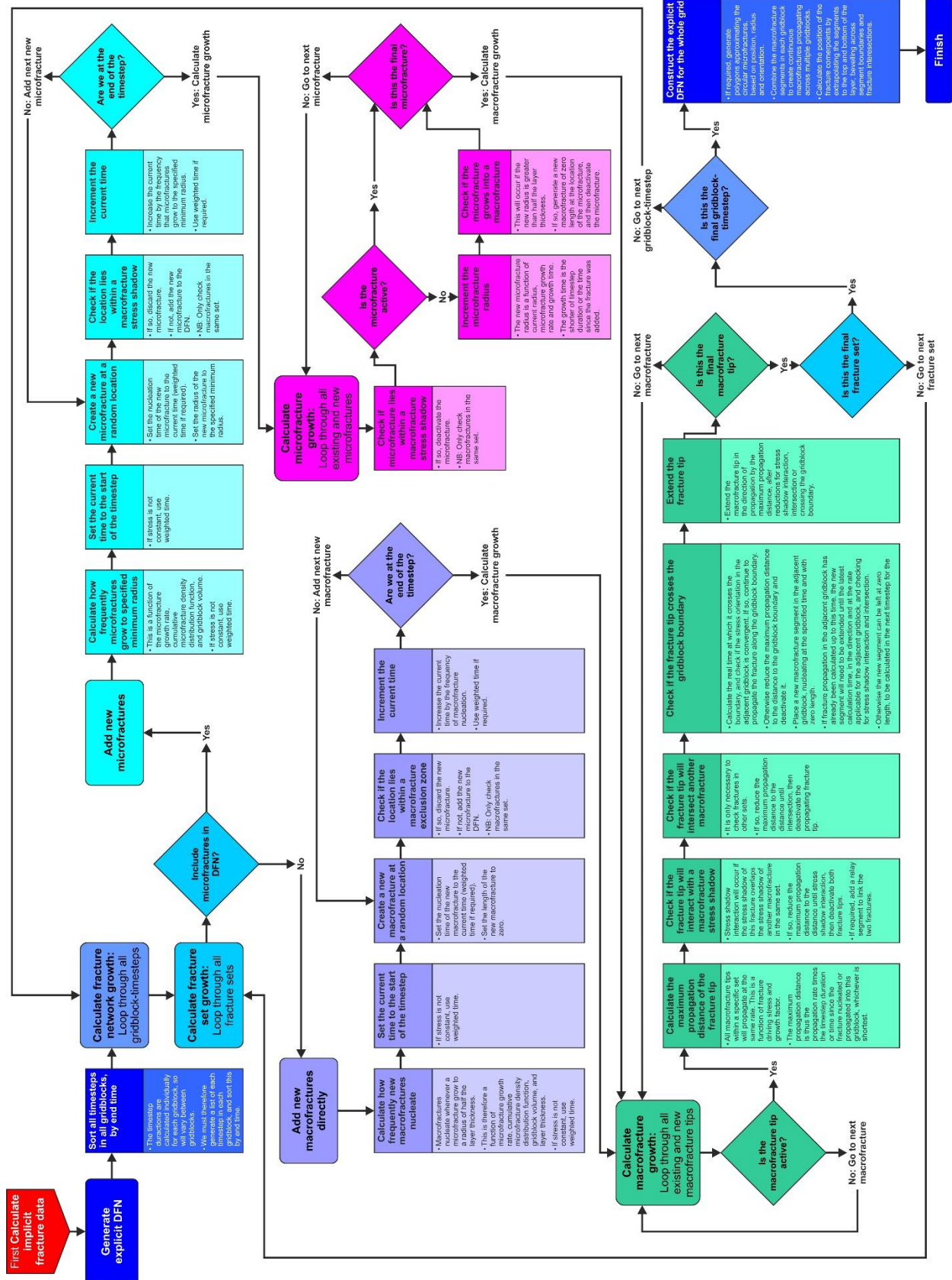


Figure 12: Flow diagram for a workflow to generate an explicit DFN model for a fractured layer discretised into gridblocks.

4.2.2 Workflow steps

Figure 12 presents a workflow for generating an explicit DFN model for an entire fractured layer, based on an implicit fracture model generated as described in Section 4.1 and on Figure 5. Note that it is possible to generate a DFN that comprises only macrofractures, or a DFN that comprises both microfractures (above a specified minimum radius) and macrofractures.

In the workflow shown in Figure 12, the growth of the DFN is calculated independently for each timestep in each gridblock. This allows us to use the local coordinate systems described in Section 4.2.1 to simplify the calculations of fracture nucleation and propagation. These calculations are described in Sections 4.2.2.1, 4.2.2.2, 4.2.2.3 and 4.2.2.4.

Although these calculations do allow for fractures to propagate between adjacent gridblocks (see Section 4.2.2.4.3), the end result is a series of local DFNs, each covering a single gridblock. However the ultimate objective is to generate a DFN that covers the entire fractured layer, so we must then combine each of these local DFNs into a single global DFN referenced in global coordinates. This is described in Section 4.2.2.5.

4.2.2.1 Adding new microfractures

If the microfractures follow a power law distribution, then the theoretical total microfracture density tends to infinity as the microfracture radius tends to zero. Therefore, if the DFN is to include microfractures, then we must specify a minimum radius limit for the microfractures to be included. We can then determine how many microfractures will reach their minimum radius in any specific gridblock and timestep. Finally, we can determine when each of these microfractures will reach the minimum radius, and where it will be located.

If there are no macrofracture stress shadows (e.g. in the evenly distributed stress scenario), we can use the expression for $\mu_F P_{30}(r,t)$ derived in Section 3.3 of Welch et al. (2020) (equation 3.14) to determine the rate at which microfractures reach the minimum radius. If we assume that microfractures start with a perfect power law size distribution, then they will reach the minimum radius limit r_{min} at a regular rate, and we can use a discretised form of Welch et al. (2020) equation 3.14 to determine when individual microfractures will reach the minimum radius. If $S_{m,i}^m(t)$ is the total number of set S and dipset m microfractures with radius greater than the minimum limit in the gridblock at time t_n during timestep N, then the time at which the i th microfracture will reach the minimum radius, $t_{rmin}(i)$, will be given by

$$\begin{aligned}
 [160] \quad \mu_F P_{30}(r_{min}, t_n) &= B \left(r_{min}^{\frac{1}{\beta}} + \Gamma_{Sm,N-1} + \gamma_{Sm,N}^{\frac{1}{\beta}} (t_n - t_{N-1}) \right)^{-\beta c} \\
 \Rightarrow i(S_{m,i}^m) &= \left\lfloor \mathbb{V}_{gr} B \left(r_{min}^{\frac{1}{\beta}} + \Gamma_{Sm,N-1} + \gamma_{Sm,N}^{\frac{1}{\beta}} (t_n - t_{N-1}) \right)^{-\beta c} \right\rfloor \\
 \Rightarrow S_{m,i}^m &= \mathbb{V}_{gr} B \left(r_{min}^{\frac{1}{\beta}} + \Gamma_{Sm,N-1} + \gamma_{Sm,N}^{\frac{1}{\beta}} (t_{rmin}(S_{m,i}^m) - t_{N-1}) \right)^{-\beta c} \\
 \Rightarrow \left(\frac{S_{m,i}^m}{\mathbb{V}_{gr} B} \right)^{-\frac{1}{\beta c}} &= r_{min}^{\frac{1}{\beta}} + \Gamma_{Sm,N-1} + \gamma_{Sm,N}^{\frac{1}{\beta}} t_{rmin}(S_{m,i}^m) - \gamma_{Sm,N}^{\frac{1}{\beta}} t_{N-1} \\
 \Rightarrow t_{rmin}(S_{m,i}^m) &= \gamma_{Sm,N}^{-\frac{1}{\beta}} \left(\left(\frac{S_{m,i}^m}{\mathbb{V}_{gr} B} \right)^{-\frac{1}{\beta c}} - r_{min}^{\frac{1}{\beta}} - \Gamma_{Sm,N-1} + \gamma_{Sm,N}^{\frac{1}{\beta}} t_{N-1} \right)
 \end{aligned}$$

where N is the current timestep, V_{gr} is the total gridblock volume (see Section 4.2.1.2.2). In practice there will be some random scatter in the initial microfracture radii and therefore the time at which individual microfractures will reach the minimum radius cannot be predicted; however [160] should give a good approximation of the average rate at which microfracture reach the minimum radius throughout the timestep, as long as the gridblock is large enough and the timestep long enough.

However if the gridblock is so small, or the timestep so short, that the average rate at which microfractures reach the minimum radius is near to or less than the timestep duration (so the average number of microfractures expected to reach the minimum radius during the timestep is one or less) then a discretisation error will be introduced into the calculation, as it is only possible to add a whole number of microfractures to the DFN per gridblock per timestep. There are two ways around this problem:

- To calculate the times at which microfractures will reach the minimum radius over all timesteps, so that microfractures will be added to the DFN in some timesteps but not others. In this way, the average rate at which microfractures are added to the DFN over a series of timesteps should approximate to the calculated rate at which microfractures reach the minimum radius.
- To add a probabilistic element to determining whether microfractures will be added to the DFN or not – for example, if the time until the next microfracture is calculated to reach the minimum radius is twice the length of the current timestep, use a random probability of 0.5 to determine whether to add a microfracture at the end of the timestep

Note that it will usually be necessary to add a batch of new microfractures at the start of the first timestep, to represent the initial microfracture population.

The new microfractures should be placed at random locations in the gridblock, determined using the methods described in Section 4.2.1.3.3. The microfracture dip direction j should also be assigned randomly. Thus, if the random centrepoint of microfracture i is $(^S I_i, ^S J_i, ^S K_i)$, converted into IJK coordinates, and the randomly determined dip direction is j_i , the new microfracture will be described by

$$[161] \quad ^S \mu F_i = (^S I_i, ^S J_i, ^S K_i, r_{min}, m, j_i)$$

If the new microfracture lies within the stress shadow of a macrofracture of the same set then it will not grow. The actual rate at which microfractures reach the minimum radius limit will therefore be lower than that given by [160]. However we can include this effect in the DFN very easily, by adding microfractures at random locations at the rate specified by [160], and then checking whether the new microfractures lie within the stress shadow of a macrofracture of the same set. If they do, they should be discarded and no new microfracture should be added at that time; in this way, the number of new microfractures added will be reduced in proportion to the total macrofracture stress shadow volume, which is consistent with the implicit microfracture populations calculated as described in Section 5.2.1 of Welch et al. (2020).

We can easily determine whether a microfracture of set S lies within the stress shadow of macrofracture segment of set S and dipset n , using the IJK coordinates for set S . If the centre of the microfracture has coordinates $^S I_{\mu F}, ^S J_{\mu F}, ^S K_{\mu F}$, the l^- macrofracture segment node has coordinates $^S I_{MF-}, ^S J_{MF-}, 0$, and the l^+ macrofracture segment node has coordinates $^S I_{MF+}, ^S J_{MF+}, 0$, then the microfracture will lie within the macrofracture segment stress shadow if

$$[162] \quad ^S I_{MF-} \leq ^S I_{\mu F} \leq ^S I_{MF+}$$

$$S_{J_{MF}} - \frac{n_{MF}\bar{W}}{2} \leq S_{J_{\mu F}} \leq S_{J_{MF}} + \frac{n_{MF}\bar{W}}{2}$$

where $n_{MF}\bar{W}$ is the mean width of the stress shadow of a dipset n macrofracture – note that the macrofracture dipset n may be different from the microfracture dipset m . Of course we must check this for all macrofracture segments of set S in the gridblock. Note also that microfractures located near the gridblock boundaries will not be deactivated if they lie within the stress shadow of a macrofracture segment in the adjacent gridblock; this will introduce a discrepancy into the calculation, but it will be small as long as the gridblock size is significantly larger than the macrofracture stress shadow width.

4.2.2.2 Microfracture growth

We can simulate the growth of microfractures during any timestep using equation 3.8, as derived in Section 3.2 of Welch et al. (2020). However before doing so, we must recheck each microfracture for deactivation, since the growth of the macrofractures during the previous timestep may have brought some microfractures into their stress shadow. We can check whether existing microfractures lie within a macrofracture stress shadow using the same algorithm that we used to check new microfractures, described in Section 4.2.2.1.

If a microfracture has radius r_{N-1} at the start of a timestep N , then the radius at the end of timestep N r_N will be given by

$$[163] \quad r_N = \left(r_{N-1}^{\frac{1}{\beta}} - \gamma_{Sm,N}^{\frac{1}{\beta}} (t_N - t_{N-1}) \right)^{\beta}$$

For microfractures that have nucleated during timestep N , we must of course adjust [163] to take account of the time of nucleation. However this is easy to do if the time of nucleation is expressed in weighted time, \mathcal{T}_{rmin} , as we can use the relationship

$$[164] \quad \mathcal{T}_{rmin} = A \left(\frac{\sqrt{2h}\sigma_{dSm,N}}{\sqrt{\pi}K_C} \right)^b (t_{rmin} - t_{N-1}) = \left(\frac{h}{2} \right)^{\frac{b}{2}} \gamma_{Sm,N} (t_{rmin} - t_{N-1})$$

Thus [163] can be rewritten as

$$[165] \quad r_N = \left(r_{min}^{\frac{1}{\beta}} - \gamma_{Sm,N}^{\frac{1}{\beta}} (t_N - t_{N-1}) + \mathcal{T}_{rmin} |\beta| \left(\frac{h}{2} \right)^{-\frac{b}{2}} \right)^{\beta}$$

After calculating the growth of the microfracture, we must check whether it is still contained within the layer; if the new microfracture radius r_N is greater than half the layer thickness $h/2$, the microfracture will evolve into a layer-bound macrofracture. In this case we must deactivate the microfracture and add a new macrofracture seed at the microfracture location, as described in Section 4.2.2.3. The exact time at which the new macrofracture nucleates in weighted time, \mathcal{T}_{MFO} , can be calculated by substituting [164] into [163], and inverting:

$$[166] \quad \begin{aligned} \frac{h}{2} &= \left(r_{N-1}^{\frac{1}{\beta}} - \mathcal{T}_{MFO} |\beta| \left(\frac{h}{2} \right)^{-\frac{b}{2}} \right)^{\beta} \\ \Rightarrow \left(\frac{h}{2} \right)^{\frac{1}{\beta}} &= r_{N-1}^{\frac{1}{\beta}} - \mathcal{T}_{MFO} |\beta| \left(\frac{h}{2} \right)^{-\frac{b}{2}} \\ \Rightarrow \mathcal{T}_{MFO} &= |\beta| \left(\frac{h}{2} \right)^{\frac{b}{2}} \left(r_{N-1}^{\frac{1}{\beta}} - \left(\frac{h}{2} \right)^{\frac{1}{\beta}} \right) \end{aligned}$$

If the microfracture in question nucleated during the timestep, [166] must be modified to

$$[167] \Rightarrow \mathcal{T}_{MF0} = |\beta| \left(\frac{h}{2} \right)^{\frac{b}{2}} \left(r_{min}^{\frac{1}{\beta}} + \mathcal{T}_{rmin} |\beta| \left(\frac{h}{2} \right)^{-\frac{b}{2}} - \left(\frac{h}{2} \right)^{\frac{1}{\beta}} \right)$$

Note however that, since the microfracture centrepoints need not lie at the centre of the layer, it is possible for one side of the microfractures top intersect the top or bottom of the layer before the microfracture radius r reaches $h/2$. In reality, this will lead to microfractures with distorted shapes or varying upward and downward growth rates; however a simpler fix for this problem is to adjust the microfracture centrepoint up or down slightly so that the entire microfracture remains within the layer.

4.2.2.3 Macrofracture nucleation

If we are including microfractures in the DFN, then new macrofractures will nucleate whenever a microfracture grows to radius $h/2$, as described in Section 4.2.2.2. The new macrofracture will be a member of the same fracture set and dipset $m_{\mu F}$ as the original microfracture, and will have the same dip direction $j_{\mu F}$. It will comprise two segments ($s1$ and $s2$), representing the two half-macrofractures that comprise the new macrofracture. Each segment will have zero length, so that the macrofracture will be described by

$$[168] \quad {}^S MF_n = \{ {}^S MFn_{s1}, {}^S MFn_{s2} \}$$

$${}^S MFn_{s1} = (gr_p, {}^S I_{\mu F}, {}^S I_{\mu F}, {}^S J_{\mu F}, m_{\mu F}, j_{\mu F}, I^-)$$

$${}^S MFn_{s2} = (gr_p, {}^S I_{\mu F}, {}^S I_{\mu F}, {}^S J_{\mu F}, m_{\mu F}, j_{\mu F}, I^+)$$

where ${}^S I_{\mu F}$ and ${}^S J_{\mu F}$ are the coordinates of the centre of the microfracture. The time of nucleation of the macrofracture \mathcal{T}_{MF0} , in weighted time, is given by [166] or [167] respectively.

If the DFN does not include microfractures, we can nucleate zero-length macrofractures directly, using a modified version of [160] to determine when the i th macrofracture of dipset m in set S will nucleate. We can simply replace r_{min} with $h/2$ to get:

$$[169] \Rightarrow t_{rmin}({}^S m_i) = \gamma_{Sm,N}^{-\frac{1}{\beta}} \left(\left(\frac{{}^S m_i}{\mathbb{V}_{grB}} \right)^{-\frac{1}{\beta c}} - \left(\frac{h}{2} \right)^{\frac{1}{\beta}} - \Gamma_{Sm,N-1} + \gamma_{Sm,N}^{\frac{1}{\beta}} t_{N-1} \right)$$

The new macrofractures should be placed at random locations on the surface representing the centre of the gridblock. These can be determined by generating a random point in XYZ coordinates using the methods described in Section 4.2.1.3.3, then converting this to IJK coordinates and setting $K=0$. The dip direction j should also be assigned randomly. Thus, if the random centrepoint of a new set S and dipset m macrofracture i is $({}^S I_i, {}^S J_i, 0)$, in IJK coordinates, and the randomly determined dip direction is j_i , the new macrofracture will be described by

$$[170] \quad {}^S MF_i = \{ {}^S MFi_{s1}, {}^S MFi_{s2} \}$$

$${}^S MFi_{s1} = (gr_p, {}^S I_i, {}^S I_i, {}^S J_i, m, j, I^-)$$

$${}^S MFi_{s2} = (gr_p, {}^S I_i, {}^S I_i, {}^S J_i, m, j, I^+)$$

If the gridblock is so small, or the timestep so short, that the average rate at which macrofractures nucleate is near to or less than the timestep duration then a discretisation error will be introduced into the calculation, as it is only possible to add a whole number of macrofractures to the DFN per

gridblock per timestep. As for microfractures, this can be solved either by calculating the times at which macrofractures nucleate over all timesteps, so that macrofractures will be added to the DFN in some timesteps but not others, or by adding a probabilistic element to determining whether macrofractures will be added to the DFN or not.

Before adding any new macrofracture, we must check whether it lies within the exclusion zone of an existing macrofracture of the same set. Note that we must do this for macrofractures that nucleated from microfractures, as well as for new macrofractures added directly, since the microfractures are only deactivated if they lie in the stress shadow of a macrofracture, rather than the wider exclusion zone. We can determine whether the new macrofracture lies within the exclusion zone of another macrofracture segment of the same set S , using a similar method to that used to check stress shadows in Section 4.2.2.1. If the nucleation point of the new macrofracture has coordinates s_{I_i} and s_{J_i} , the I^- node of the existing macrofracture segment has coordinates $s_{I_{MF-}}$, $s_{J_{MF}}$, 0, and the I^+ node of the existing macrofracture segment has coordinates $s_{I_{MF+}}$, $s_{J_{MF}}$, 0, then the new macrofracture will lie within the exclusion zone of the existing macrofracture segment if

$$[171] \quad s_{I_{MF-}} \leq s_{I_i} \leq s_{I_{MF+}} \\ s_{J_{MF} - \frac{m_{MF}\bar{W} + n_{MF}\bar{W}}{2}} \leq s_{J_i} \leq s_{J_{MF} + \frac{m_{MF}\bar{W} + n_{MF}\bar{W}}{2}}$$

where m is the dipset of the new macrofracture, n is the dipset of the existing macrofracture segment (which may be different to the dipset of the new macrofracture), and ${}^m_{MF}\bar{W}$ and ${}^n_{MF}\bar{W}$ are the mean widths of the stress shadow of a dipset m and a dipset n macrofracture respectively. Again we must check this for all macrofracture segments of set S in the gridblock. Note also that new macrofractures located near the gridblock boundaries will not be deactivated if they lie within the exclusion zone of a macrofracture segment in the adjacent gridblock; this will introduce a discrepancy into the calculation, but it will be small as long as the gridblock size is significantly larger than the macrofracture stress shadow width.

4.2.2.4 Macrofracture propagation and interaction

As we have seen, within a specific gridblock all macrofracture tips of a given set S and dipset m will propagate at the same rate. We can therefore define a maximum propagation distance for a macrofracture tip during a timestep \mathbf{N} , $\Delta^{sm}\ell_N$. For a macrofracture that was already present at the start of the timestep \mathbf{N} , this can be calculated by modifying equation 4.2 in Section 4.1 of Welch et al. (2020) to get

$$[172] \quad \Delta^{sm}\ell_N = A \left(\frac{\sqrt{2h}\sigma_{dSm,N}}{\sqrt{\pi}K_C} \right)^b \Delta t_N = \alpha_{MF}\sigma_{dSm,N}^b \Delta t_N$$

where Δt_N is the duration of timestep \mathbf{N} . For a macrofracture that nucleated during the timestep \mathbf{N} , we must subtract the weighted nucleation time \mathcal{T}_{MF0} :

$$[173] \quad \Delta^{sm}\ell_N = A \left(\frac{\sqrt{2h}\sigma_{dSm,N}}{\sqrt{\pi}K_C} \right)^b (\Delta t_N - \mathcal{T}_{MF0}) = \alpha_{MF}\sigma_{dSm,N}^b (\Delta t_N - \mathcal{T}_{MF0})$$

The propagation distances given by [172] and [173] represent maximum distances; the actual propagation distance of a specific macrofracture tip will be less if it is deactivated due to stress shadow interaction or intersection, or if it crosses the gridblock boundary, during the timestep \mathbf{N} . In the following sections we will show how to check these events, and reduce the fracture propagation distance accordingly. Once we have determined the actual propagation distance for a specific

fracture tip $\Delta^{sm}\ell_{MF,N}$, we can easily calculate the new position of the propagating tip in IJK coordinates by adding or subtracting this to the I coordinate; the J coordinate will not change:

$$[174] \quad {}^SI_{MF,N} = {}^SI_{MF,N-1} + \Delta^{sm}\ell_{MF,N}$$

$${}^SJ_{MF,N} = {}^SJ_{MF,N-1}$$

for a macrofracture tip propagating in the I^+ direction, and

$$[175] \quad {}^SI_{MF,N} = {}^SI_{MF,N-1} - \Delta^{sm}\ell_{MF,N}$$

$${}^SJ_{MF,N} = {}^SJ_{MF,N-1}$$

for a macrofracture tip propagating in the I^- direction, where ${}^SI_{MF,N-1}, {}^SJ_{MF,N-1}, 0$ are the IJK coordinates of the original tip and ${}^SI_{MF,N}, {}^SJ_{MF,N}, 0$ are the IJK coordinates of the new tip.

4.2.2.4.1 Macrofracture tip deactivation due to stress shadow interaction

We have already seen in Section 4.2.2.2 that checking for microfracture stress shadow interaction is made much simpler by using IJK coordinates. The same is true for macrofracture stress shadow interaction.

A macrofracture tip of set S and dipset m, propagating in the I^+ direction, will be deactivated due to stress shadow interaction if there is an I^- macrofracture tip of set S and dipset n lying within the space

$$[176] \quad {}^SI_{MF2-} - \Delta^{sm}\ell_N < {}^SI_{MF1+} \leq {}^SI_{MF2-}$$

$${}^SJ_{MF2-} - \frac{{}^m\bar{W} + {}^n\bar{W}}{2} \leq {}^SJ_{MF1} \leq {}^SJ_{MF2-} + \frac{{}^m\bar{W} + {}^n\bar{W}}{2}$$

where ${}^SI_{MF1+}, {}^SJ_{MF1}, 0$ are the IJK coordinates of the macrofracture tip propagating in the I^+ direction and ${}^SI_{MF2-}, {}^SJ_{MF2}, 0$ are the IJK coordinates of the macrofracture tip propagating in the I^- direction. Note that, while the two macrofractures must belong to the same set S, they need not have the same dipset (dipset n may be different from dipset m). If the two macrofracture stress shadows do interact, we can reduce the maximum propagation distance of the macrofracture tip propagating in the I^+ direction $\Delta^{sm}\ell_{MF1+,N}$ to

$$[177] \quad \Delta^{sm}\ell_{MF1+,N} = {}^SI_{MF2-} - {}^SI_{MF1+}$$

Note however that we must still continue to check for stress shadow interaction with all remaining I^- macrofracture tips of set S, and also check for intersection (see Section 4.2.2.4.2), as it is possible that the I^+ macrofracture tip may interact with another macrofracture stress shadow, or intersect a macrofracture from another set, before reaching the stress shadow of the macrofracture we have just checked.

The equivalent equations to [176] and [177] for a macrofracture tip propagating in the I^- direction are given by

$$[178] \quad {}^SI_{MF2+} \leq {}^SI_{MF1-} < {}^SI_{MF2+} + \Delta^{sm}\ell_N$$

$${}^SJ_{MF2+} - \frac{{}^m\bar{W} + {}^n\bar{W}}{2} \leq {}^SJ_{MF1-} \leq {}^SJ_{MF2+} + \frac{{}^m\bar{W} + {}^n\bar{W}}{2}$$

and

$$[179] \quad \Delta^{sm}\ell_{MF1-,N} = {}^SI_{MF1-} - {}^SI_{MF2+}$$

If macrofracture stress shadow interaction does occur, we must of course deactivate both macrofracture tips. Note that the procedure described here will introduce an element of bias into the resulting fracture network, since it assumes that the macrofracture tip that is checked first will propagate all the way to the second macrofracture tip, which will not propagate at all. In reality, of course, if both tips are active then they will both propagate and intersect at the mid-point. However if the timestep duration is relatively short (so that the maximum macrofracture tip propagation distance $\Delta^{sm}\ell_N$ is small) this will not cause a large discrepancy.

4.2.2.4.2 Macrofracture tip deactivation due to intersection with another macrofracture

Checking for macrofracture interaction is also made simpler if we use IJK coordinates, especially if the macrofracture sets have perpendicular strike.

In the general case, to check whether a macrofracture tip of set S1 and dipset m, propagating in the I^+ direction, will intersect a macrofracture segment of set S2, we must first convert the coordinates nodes of the two tips of the S2 segment into IJK coordinates relative to set S1, $^{S1}I_{MF2-}, ^{S1}J_{MF2-}, 0$ and $^{S1}I_{MF2+}, ^{S1}J_{MF2+}, 0$ respectively. A propagating S1 macrofracture tip with IJK coordinates $^{S1}I_{MF1+}, ^{S1}J_{MF1}, 0$ will intersect the S2 segment if

$$[180] \quad ^{S1}I_{MF12} - \Delta^{sm}\ell_N < ^{S1}I_{MF1+} \leq ^{S1}I_{MF12}$$

$$\min \left\{ \begin{matrix} ^{S1}J_{MF2-} \\ ^{S1}J_{MF2+} \end{matrix} \right\} \leq ^{S1}J_{MF1} \leq \max \left\{ \begin{matrix} ^{S1}J_{MF2-} \\ ^{S1}J_{MF2+} \end{matrix} \right\}$$

and we can reduce the maximum propagation distance of the S1 macrofracture tip to

$$[181] \quad \Delta^{sm}\ell_{MF1+,N} = ^{S1}I_{MF12} - ^{S1}I_{MF1+}$$

where

$$[182] \quad ^{S1}I_{MF12} = \frac{^{S1}J_{MF2+} - ^{S1}J_{MF1}}{^{S1}J_{MF2+} - ^{S1}J_{MF2-}} ^{S1}I_{MF2-} + \frac{^{S1}J_{MF1} - ^{S1}J_{MF2-}}{^{S1}J_{MF2+} - ^{S1}J_{MF2-}} ^{S1}I_{MF2+}$$

For a macrofracture tip of set S1 and dipset m propagating in the I^- direction, the equivalent equations to [180], [181] and [182] are given by

$$[183] \quad ^{S1}I_{MF12} \leq ^{S1}I_{MF1-} < ^{S1}I_{MF12} + \Delta^{sm}\ell_N$$

$$\min \left\{ \begin{matrix} ^{S1}J_{MF2-} \\ ^{S1}J_{MF2+} \end{matrix} \right\} \leq ^{S1}J_{MF1-} \leq \max \left\{ \begin{matrix} ^{S1}J_{MF2-} \\ ^{S1}J_{MF2+} \end{matrix} \right\}$$

and

$$[184] \quad \Delta^{sm}\ell_{MF1-,N} = ^{S1}I_{MF1-} - ^{S1}I_{MF12}$$

where

$$[185] \quad ^{S1}I_{MF12} = \frac{^{S1}J_{MF2+} - ^{S1}J_{MF1}}{^{S1}J_{MF2+} - ^{S1}J_{MF2-}} ^{S1}I_{MF2-} + \frac{^{S1}J_{MF1} - ^{S1}J_{MF2-}}{^{S1}J_{MF2+} - ^{S1}J_{MF2-}} ^{S1}I_{MF2+}$$

However if the two fracture sets S1 and S2 have perpendicular strike we can take advantage of the simple conversion between S1 and S2 IJK coordinates, given by [153] and [154], to avoid explicitly converting the IJK coordinates of the set S2 segment to S1 IJK coordinates. In this case, the propagating tip of a macrofracture striking perpendicular to σ_{hmin} will intersect a fracture segment striking perpendicular to σ_{hmax} if

$$\begin{aligned}
[186] \quad & \left. \begin{aligned} -h_{\max} J_{MF2} - \Delta^{h_{\min,m}} \ell_N < h_{\min} I_{MF1+} \leq -h_{\max} J_{MF2} \\ h_{\max} I_{MF2-} \leq h_{\min} J_{MF1} \leq h_{\max} I_{MF2+} \end{aligned} \right\} \text{for propagation in } I^+ \text{ direction} \\
& \left. \begin{aligned} -h_{\max} J_{MF2} \leq h_{\min} I_{MF1-} < -h_{\max} J_{MF2} + \Delta^{h_{\min,m}} \ell_N \\ h_{\max} I_{MF2-} \leq h_{\min} J_{MF1} \leq h_{\max} I_{MF2+} \end{aligned} \right\} \text{for propagation in } I^- \text{ direction}
\end{aligned}$$

and the reduced maximum propagation distance will be

$$\begin{aligned}
[187] \quad & \Delta^{h_{\min,m}} \ell_{MF1+,N} = -h_{\max} J_{MF2} - h_{\min} I_{MF1+} \quad \text{for propagation in } I^+ \text{ direction} \\
& \Delta^{h_{\min,m}} \ell_{MF1-,N} = h_{\min} I_{MF1-} + h_{\max} J_{MF2} \quad \text{for propagation in } I^- \text{ direction}
\end{aligned}$$

Similarly, the propagating tip of a macrofracture striking perpendicular to $\sigma_{h_{\max}}$ will intersect a fracture segment striking perpendicular to $\sigma_{h_{\min}}$ if

$$\begin{aligned}
[188] \quad & \left. \begin{aligned} h_{\min} J_{MF2} - \Delta^{h_{\max,m}} \ell_N < h_{\max} I_{MF1+} \leq h_{\min} J_{MF2} \\ -h_{\min} I_{MF2+} \leq h_{\max} J_{MF1} \leq -h_{\min} I_{MF2-} \end{aligned} \right\} \text{for propagation in } I^+ \text{ direction} \\
& \left. \begin{aligned} h_{\min} J_{MF2} \leq h_{\max} I_{MF1-} < h_{\min} J_{MF2} + \Delta^{h_{\max,m}} \ell_N \\ -h_{\min} I_{MF2+} \leq h_{\max} J_{MF1} \leq -h_{\min} I_{MF2-} \end{aligned} \right\} \text{for propagation in } I^- \text{ direction}
\end{aligned}$$

and the reduced maximum propagation distance will be

$$\begin{aligned}
[189] \quad & \Delta^{h_{\max,m}} \ell_{MF1+,N} = h_{\min} J_{MF2} - h_{\max} I_{MF1+} \quad \text{for propagation in } I^+ \text{ direction} \\
& \Delta^{h_{\max,m}} \ell_{MF1-,N} = h_{\max} I_{MF1-} - h_{\min} J_{MF2} \quad \text{for propagation in } I^- \text{ direction}
\end{aligned}$$

Again, even if we find that the propagating macrofracture tip of set S1 will intersect the macrofracture segment we are checking, we must still continue to check for intersection with all remaining macrofracture segments of all sets other than S1, and also check for stress shadow interaction with macrofracture tips from set S1 (see Section 4.2.2.4.1), as it is possible that the propagating macrofracture tip may intersect another macrofracture segment, or interact with the stress shadow of another macrofracture, before reaching the macrofracture segment we have just checked.

4.2.2.4.3 Crossing gridblock boundaries

If a propagating macrofracture tip crosses the boundary of the gridblock it lies within, it will not be deactivated, but further propagation will be controlled by the stress orientation and magnitude in the adjacent gridblock. We must therefore deactivate the currently propagating macrofracture segment and create a new zero-length macrofracture segment in the adjacent gridblock, located on the gridblock boundary. We will then add this new segment to the list of segments defining the macrofracture, as specified in [156].

The easiest way to check if a propagating macrofracture tip of set S will cross a gridblock boundary is to convert the locations of the cornerpoints of the gridblock to IJK coordinates. For a vertically aligned gridblock, each boundary segment can be defined by two cornerpoints located on the surface corresponding to the centre of the layer, with IJK coordinates $^S I_{B-}, ^S J_{B-}, 0$ and $^S I_{B+}, ^S J_{B+}, 0$ respectively. A macrofracture tip propagating in the I^+ direction with IJK coordinates $^S I_{MF+}, ^S J_{MF}, 0$ will cross the boundary segment if

$$[190] \quad ^S I_{BMF} - \Delta^{S_m} \ell_N < ^S I_{MF+} \leq ^S I_{BMF}$$

$$\min \left\{ \frac{s_{J_{B-}}}{s_{J_{B+}}} \right\} \leq s_{J_{MF}} \leq \max \left\{ \frac{s_{J_{B-}}}{s_{J_{B+}}} \right\}$$

and the propagation distance up to the boundary $\Delta^{sm} \ell_{MF+,N}$ will be given by

$$[191] \quad \Delta^{sm} \ell_{MF+,N} = s_{I_{BMF}} - s_{I_{MF+}}$$

where

$$[192] \quad s_{I_{BMF}} = \frac{s_{J_{B+}} - s_{J_{MF}}}{s_{J_{B+}} - s_{J_{B-}}} s_{I_{B-}} + \frac{s_{J_{MF}} - s_{J_{B-}}}{s_{J_{B+}} - s_{J_{B-}}} s_{I_{B+}}$$

Similarly, a macrofracture tip propagating in the I^- direction with IJK coordinates $s_{I_{MF-}}, s_{J_{MF}}, 0$ will cross the boundary segment if

$$[193] \quad s_{I_{BMF}} \leq s_{I_{MF-}} < s_{I_{BMF}} + \Delta^{sm} \ell_N$$

$$\min \left\{ \frac{s_{J_{B-}}}{s_{J_{B+}}} \right\} \leq s_{J_{MF}} \leq \max \left\{ \frac{s_{J_{B-}}}{s_{J_{B+}}} \right\}$$

and the propagation distance up to the boundary $\Delta^{sm} \ell_{MF-,N}$ will be given by

$$[194] \quad \Delta^{sm} \ell_{MF-,N} = s_{I_{MF-}} - s_{I_{BMF}}$$

where

$$[195] \quad s_{I_{BMF}} = \frac{s_{J_{B+}} - s_{J_{MF}}}{s_{J_{B+}} - s_{J_{B-}}} s_{I_{B-}} + \frac{s_{J_{MF}} - s_{J_{B-}}}{s_{J_{B+}} - s_{J_{B-}}} s_{I_{B+}}$$

We should always check if a propagating macrofracture tip will be deactivated due to stress shadow interaction or intersection before checking if it crosses a gridblock boundary, since these events will always happen before the macrofracture tip reaches the gridblock boundary.

If a macrofracture tip does cross a gridblock boundary, it will continue propagating into the adjacent gridblock. However we cannot simply “carry over” the excess propagation distance into the adjacent gridblock, since both the macrofracture propagation rates and the timestep durations in the adjacent gridblock will be different to those in the current gridblock. Instead we must calculate the real time t_b at which the macrofracture reaches the gridblock boundary, and nucleate the new fracture segment in the adjacent gridblock at that same time. The weighted time \mathcal{T}_B at which the macrofracture reaches the gridblock boundary will be given by

$$[196] \quad \mathcal{T}_B = \begin{cases} \Delta^{sm} \ell_{MF+,N} = s_{I_{BMF}} - s_{I_{MF+}} & \text{for propagation in } I^+ \text{ direction} \\ \Delta^{sm} \ell_{MF-,N} = s_{I_{MF-}} - s_{I_{BMF}} & \text{for propagation in } I^- \text{ direction} \end{cases}$$

for a macrofracture that was already present at the start of timestep \mathbf{N} , and

$$[197] \quad \mathcal{T}_B = \begin{cases} \Delta^{sm} \ell_{MF+,N} - \mathcal{T}_{MF0} = s_{I_{BMF}} - s_{I_{MF+}} - \mathcal{T}_{MF0} & \text{for propagation in } I^+ \text{ direction} \\ \Delta^{sm} \ell_{MF-,N} - \mathcal{T}_{MF0} = s_{I_{MF-}} - s_{I_{BMF}} - \mathcal{T}_{MF0} & \text{for propagation in } I^- \text{ direction} \end{cases}$$

for a macrofracture that nucleated during timestep \mathbf{N} . We can then convert the weighted time \mathcal{T}_B to real time using [159]. Note that, since the timesteps in different gridblocks are not aligned, the real time at which the macrofracture crosses the gridblock boundary t_b may lie within a timestep that has already been calculated in the adjacent gridblock, in which case we will need to extend the new segment by an appropriate distance before we calculate the next timestep in that gridblock.

Normally, the new macrofracture segment in the adjacent gridblock will belong to the same set and have the same propagation direction as the previous segment. However if there is a significant misalignment in the stress orientation across the gridblock boundary, it may be more realistic to assign the new macrofracture segment to a different fracture set that is more closely aligned with the original segment. Another situation that may cause problems is where the fracture propagation directions are convergent across the gridblock boundary, so that macrofractures will propagate towards the gridblock boundary from both sides, as shown in *Figure 13*. In this situation, following the algorithm described here will cause an infinite loop as the macrofracture continues to propagate back and forth across the gridblock boundary. This can be resolved by allowing the macrofracture to propagate along the gridblock boundary, as a boundary tracking fracture (see *Figure 13*). Simulating the propagation of boundary tracking fractures is more complicated, since the simple relationships between fracture propagation and fracture tip location in IJK coordinates that were described previously may no longer apply. In addition, it is possible for boundary tracking fractures to intersect other macrofractures from the same set, so this must also be checked for.

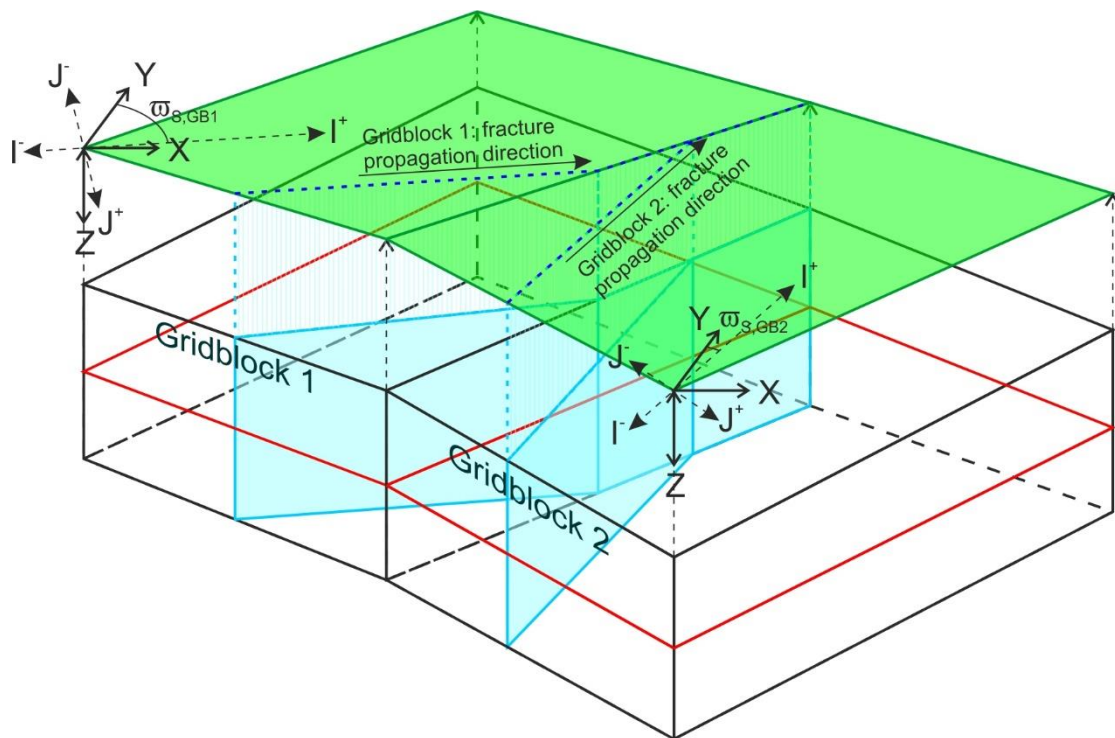


Figure 13: Schematic illustration showing convergent macrofracture propagation directions in adjacent gridblocks. This will cause the macrofracture to propagate along the gridblock boundary.

4.2.2.5 Generating and calibrating the global fracture network

Once we have generated the local DFNs for all gridblocks, we must combine them into a single global DFN that extends across the entire fractured layer. The fractures in this global DFN must of course be referenced in global coordinates. In this section we will show how to populate the global DFN with microfractures and macrofractures, and then examine how we can test and calibrate it.

4.2.2.5.1 Populating the global DFN with microfractures

Since each microfracture is confined to a single gridblock, we can populate the global DFN with microfractures by simply copying across all the microfractures from each local DFN.

However the microfractures in the local DFNs are described in a minimal form using the local IJK coordinate system, as described in Section 4.2.1.4.4 (see [155]). In the global DFN, the

microfractures must be defined in a more complete form using the global XYZ coordinate system. This can be done in one of two ways:

- Each microfracture can be defined as a circle. In this case we must define a centrepoint, radius, dip and dip azimuth. The centrepoint can be defined by converting the IJK coordinates of the centrepoint in the local form (sI , sJ and sK in [155]) to XYZ coordinates, as described in Section 4.2.1.4.1. The radius is given by r in [155]. The dip azimuth is determined by the strike of the fracture set in the appropriate gridblock, and can be calculated by adding $\pm 90^\circ$ to the I axis bearing ϖ_s , depending on the dip direction j . The fracture dip ω_m is determined by the fracture dipset m .
- Each microfracture can be defined as a planar polygon with n cornerpoints. In this case we must first determine the centrepoint (in XYZ coordinates), radius, dip and dip azimuth as described above. We then use these to calculate the positions in XYZ coordinates of a series of n equally spaced points around the circumference of the fracture, starting at the top. Although this is a less precise and more cumbersome form for defining the geometry of a microfracture than the circular form, it may provide better compatibility with some geological modelling software packages that can only handle polygonal fractures. Note that it is not necessary to define the orientation or radius of the microfracture in this format, as this information is implicit.

4.2.2.5.2 Populating the global DFN with macrofractures

Populating the global DFN with macrofractures is more complicated for two reasons:

- Macrofractures will typically comprise multiple segments which may not be coplanar, as described in Section 4.2.1.4.4 (see [156]).
- Individual macrofractures may span multiple gridblocks (see Section 4.2.2.4.3), although each segment will be confined to a single gridblock.

We must therefore combine the individual macrofracture segments from all gridblocks into complete macrofractures, that may span multiple gridblocks. This is best done while simulating the growth of the DFN in individual gridblocks: we create a global macrofracture object, comprising a list of macrofracture segments, whenever a new macrofracture nucleates (as described in [156]); and then add new segments to this list whenever a macrofracture propagates across a gridblock boundary (as described in Section 4.2.2.4.3). We can use the gr_p flag in the description of the individual macrofracture segments to link each segment to the appropriate gridblock. Finally, we can populate the global DFN with these macrofractures.

As with the microfractures, the macrofracture segments in the local DFNs are described in a minimal form using the local IJK coordinate system, which we must convert to a more complete form using the global XYZ coordinate system. The geometry of the macrofractures in the global DFN can be described in one of two ways:

- We can describe the macrofracture as a series of planar quadrilateral polygons, each defined by four cornerpoints specified in XYZ coordinates. In this case the cornerpoints of adjacent segments will be coincident so there will be redundant information in the description, but it may provide better compatibility with some geological modelling software packages that can only handle planar polygons. Note that it is not necessary to define the orientation of the segments, as this information is implicit.
- We can describe the macrofracture as a single series of cornerpoints listed in sequence around the fracture boundary around the macrofracture boundary, starting at the top nucleation point.

This format minimises the data stored but it is more complicated to reconstruct and visualise the fracture, since it may not be planar.

In either case the procedure for calculating the positions of the cornerpoints in XYZ coordinates is the same: we first extend each of the segments vertically to the top and bottom surfaces of the appropriate gridblock, and then we calculate the intersection points between the top and bottom edges of adjacent segments (see *Figure 14*).

For a specific segment p of macrofracture n , ${}^s\text{MFn}_p$, the top and bottom corners TI^-_p , TI^+_p , BI^-_p and BI^+_p of the extended segment p in IJK coordinates will be given by

$$\begin{aligned}
 [198] \quad TI^-_p &= \begin{pmatrix} {}^sI^- \\ \left\{ \begin{array}{ll} {}^sJ - h({}^sI^-, {}^sJ) \tan \omega_m & \text{for } j = J^+ \\ {}^sJ + h({}^sI^-, {}^sJ) \tan \omega_m & \text{for } j = J^- \end{array} \right\} \\ -h({}^sI^-, {}^sJ) \end{pmatrix} \\
 TI^+_p &= \begin{pmatrix} {}^sI^+ \\ \left\{ \begin{array}{ll} {}^sJ - h({}^sI^+, {}^sJ) \tan \omega_m & \text{for } j = J^+ \\ {}^sJ + h({}^sI^+, {}^sJ) \tan \omega_m & \text{for } j = J^- \end{array} \right\} \\ -h({}^sI^+, {}^sJ) \end{pmatrix} \\
 BI^-_p &= \begin{pmatrix} {}^sI^- \\ \left\{ \begin{array}{ll} {}^sJ + h({}^sI^-, {}^sJ) \tan \omega_m & \text{for } j = J^+ \\ {}^sJ - h({}^sI^-, {}^sJ) \tan \omega_m & \text{for } j = J^- \end{array} \right\} \\ h({}^sI^-, {}^sJ) \end{pmatrix} \\
 BI^+_p &= \begin{pmatrix} {}^sI^+ \\ \left\{ \begin{array}{ll} {}^sJ + h({}^sI^+, {}^sJ) \tan \omega_m & \text{for } j = J^+ \\ {}^sJ - h({}^sI^+, {}^sJ) \tan \omega_m & \text{for } j = J^- \end{array} \right\} \\ h({}^sI^+, {}^sJ) \end{pmatrix}
 \end{aligned}$$

where fracture dip ω_m is the fracture dip, determined by the fracture dipset m , and $h({}^sI, {}^sJ)$ is the thickness of the gridblock at point $({}^sI, {}^sJ)$.

We can then convert the cornerpoint locations into XYZ coordinates as described in Section 4.2.1.4.1. However if two adjacent segments do not have a parallel strike (e.g. if they are in different gridblocks where the I-axis directions are not parallel), the top and bottom cornerpoints of the extended segments will not be coincident even though the segment nodes are (see *Figure 14*). In this case we must calculate the intersection points between the top and bottom edges of the two segments, and use these as the respective cornerpoints for both segments.

The intersection point $T_{p,q}$ between the top edges of two adjacent fracture segments p and q , in XYZ coordinates, will be given by

$$[199] \quad T_{p,q} = \begin{pmatrix} XTI^-_p + \Delta XT_p \frac{\Delta XT_q(YTI^-_q - YTI^-_p) - (XTI^-_q - XTI^-_p)\Delta YT_q}{\Delta XT_q\Delta YT_p - \Delta XT_p\Delta YT_q} \\ YTI^-_p + \Delta YT_p \frac{\Delta XT_q(YTI^-_q - YTI^-_p) - (XTI^-_q - XTI^-_p)\Delta YT_q}{\Delta XT_q\Delta YT_p - \Delta XT_p\Delta YT_q} \\ ZTI^-_p + \Delta ZT_p \frac{\Delta XT_q(YTI^-_q - YTI^-_p) - (XTI^-_q - XTI^-_p)\Delta YT_q}{\Delta XT_q\Delta YT_p - \Delta XT_p\Delta YT_q} \end{pmatrix}$$

where

$$[200] \quad \Delta XT_p = XTI^+_p - XTI^-_p$$

$$\Delta YT_p = YTI^+_p - YTI^-_p$$

$$\Delta ZT_p = ZTI^+_p - ZTI^-_p$$

$$\Delta XT_q = XTI^+_q - XTI^-_q$$

$$\Delta YT_q = YTI^+_q - YTI^-_q$$

$$\Delta ZT_q = ZTI^+_q - ZTI^-_q$$

and $(XTI^-_p, YTI^-_p, ZTI^-_p)$, $(XTI^+_p, YTI^+_p, ZTI^+_p)$, $(XTI^-_q, YTI^-_q, ZTI^-_q)$ and $(XTI^+_q, YTI^+_q, ZTI^+_q)$ represent the four top cornerpoints of the extended segments p and q in XYZ coordinates. A similar formula can be used to calculate the intersection point between the bottom edges of segments p and q.

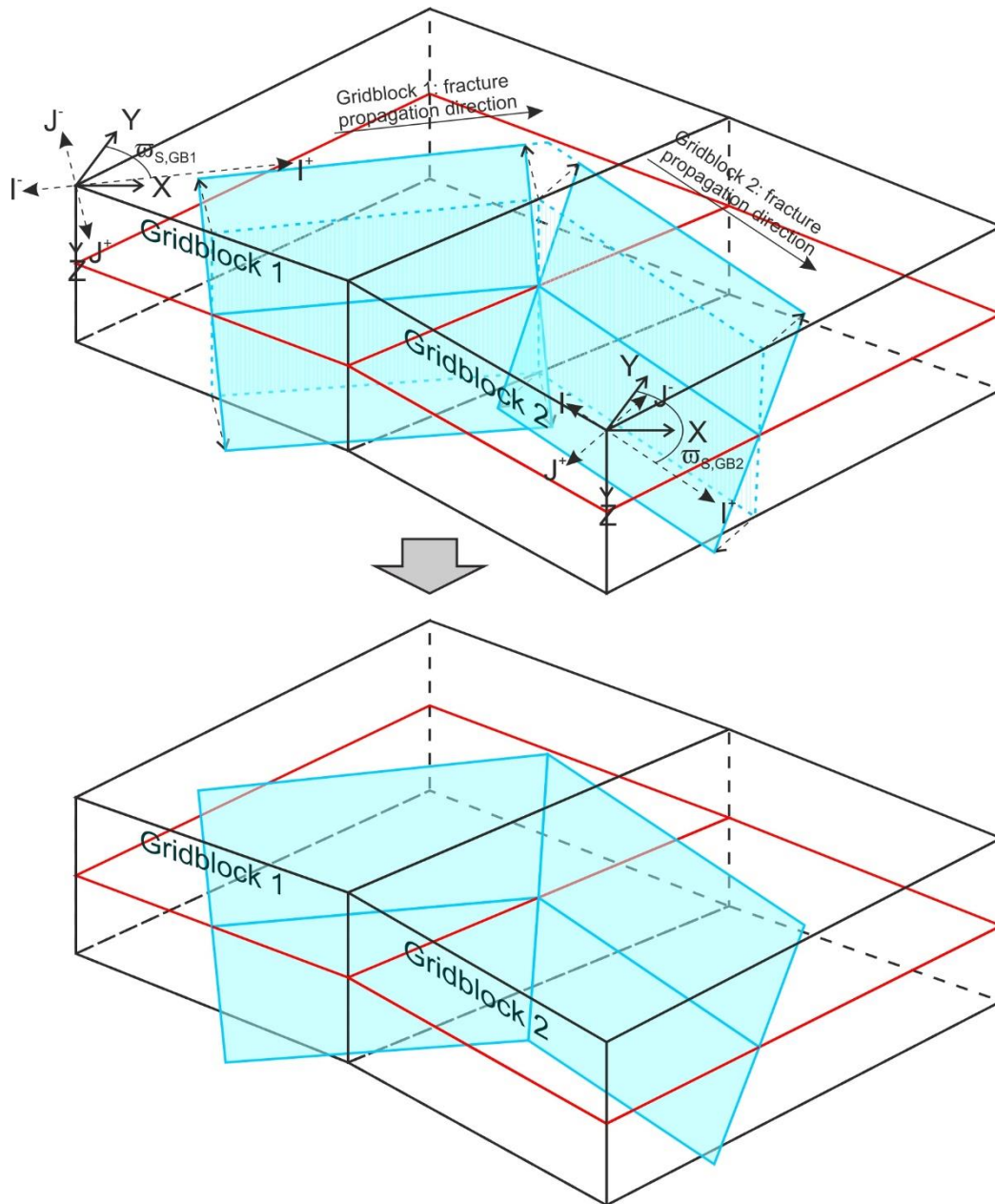


Figure 14: Schematic illustration showing bevelling of the cornerpoints between two non-parallel segments of a layer-bound macrofracture in adjacent gridblocks.

Some studies (e.g. Huggins et al. 1995, Ferrill and Morris 2001, Walsh et al. 2003) have suggested that most large fractures and faults observed in outcrop and subsurface data have formed by the linkage of initially independent but parallel segments that have propagated laterally towards each other, eventually connecting via the development of relay segments. We can include this phenomenon in our explicit DFN by combining separate macrofractures to form a single macrofracture object, whenever they terminate against each other due to stress shadow interaction. To do this we simply combine the individual macrofracture segments for both fractures into a single segment list, representing a single macrofracture, and add an additional relay segment connecting the two interacting tips. Note that the relay segment will strike perpendicular to the two segments it is connecting.

Two other points to bear in mind when linking macrofractures in this way are:

- The two macrofractures may dip in opposite directions. This will generate a combined macrofracture that switches polarity along the strike. However this is consistent with observations of linked fault segments in outcrop and subsurface data (Nicol et al. 1995).
- Since the calculation of the implicit fracture population functions does not take account of macrofracture linkage, this will introduce a discrepancy between the explicit and implicit cumulative fracture populations: specifically, the implicit fracture population functions will underestimate the density and area of large macrofractures and overestimate the density of small macrofractures. This is examined further in Section 8.3.5 of Welch et al. (2020). However the total half-macrofracture area $\underline{MF}P_{32}(0)$ will be unaffected by macrofracture linkage so should still be consistent for both explicit and implicit fracture models.

Despite these problems, linking macrofractures in this way may allow more accurate calculation of fluid flow in the fractured layer, since it reproduces the long flow pathways that such connected fractures and faults can provide.

5 Structure of the DFN Generator code

The workflows described in Section 4 have been implemented in object-oriented code (mostly C Sharp), packaged in a Microsoft Visual Studio solution. This is available for download from Github (<https://github.com/JointFlow/DFN-Generator>) as open source software under the Apache license, and can be freely modified or enhanced under the terms of the license.

We envisage two potential objectives for modifying or enhancing the DFN Generator code:

- To provide a new interface or to integrate the code with another software package (e.g. as a plug-in to a geomodelling package).
- To provide additional functionality or modify existing functionality (including bug fixing).

The Visual Studio solution has been structured so that the interface and calculation components of the code are kept in separate projects within the solution. It should therefore be possible to modify or add interfaces without modifying the calculation code, and similarly it should be possible to modify or add functionality with minimal changes to the interface code.

5.1 Interface and calculation code

As we discussed in Section 4, while the simplest fracture models can be described by limited input data, models designed to simulate fracture growth in real geological layers will generally have complex geometry and include lateral variations in the mechanical properties and in situ stress state. These are often described in the form of static geomodels, which discretise the geological layers into polyhedral gridblocks with defined cornerpoints and internally homogeneous properties.

There are already many static geomodelling packages available, either as commercial or open source software. It is not our intention to replicate these, but rather to enable the DFN Generator to interact easily with them. In this way, the existing static geomodelling packages can be used to define the model geometry and input properties (including in situ stress and strain), as well as to display the output, while the DFN Generator code is used to run the algorithms and generate the fracture models. The integration of DFN Generator with the static geomodelling packages may be a soft integration (i.e. as two stand-alone software packages transferring data by reading and writing files with a standardised format), or a hard integration (e.g. using interface code to create a plug-in or sub-module for the static geomodelling package).

As noted above, the DFN Generator code is split into two separate components:

- The **interface code** reads in the input data, handles the model building and property population, launches the calculation and then displays or outputs the results in an appropriate format.
- The **calculation code** runs the calculations to implement the workflows described in Section 4, based on the algorithms described in Welch et al. (2020).

This is illustrated schematically in *Figure 15*.

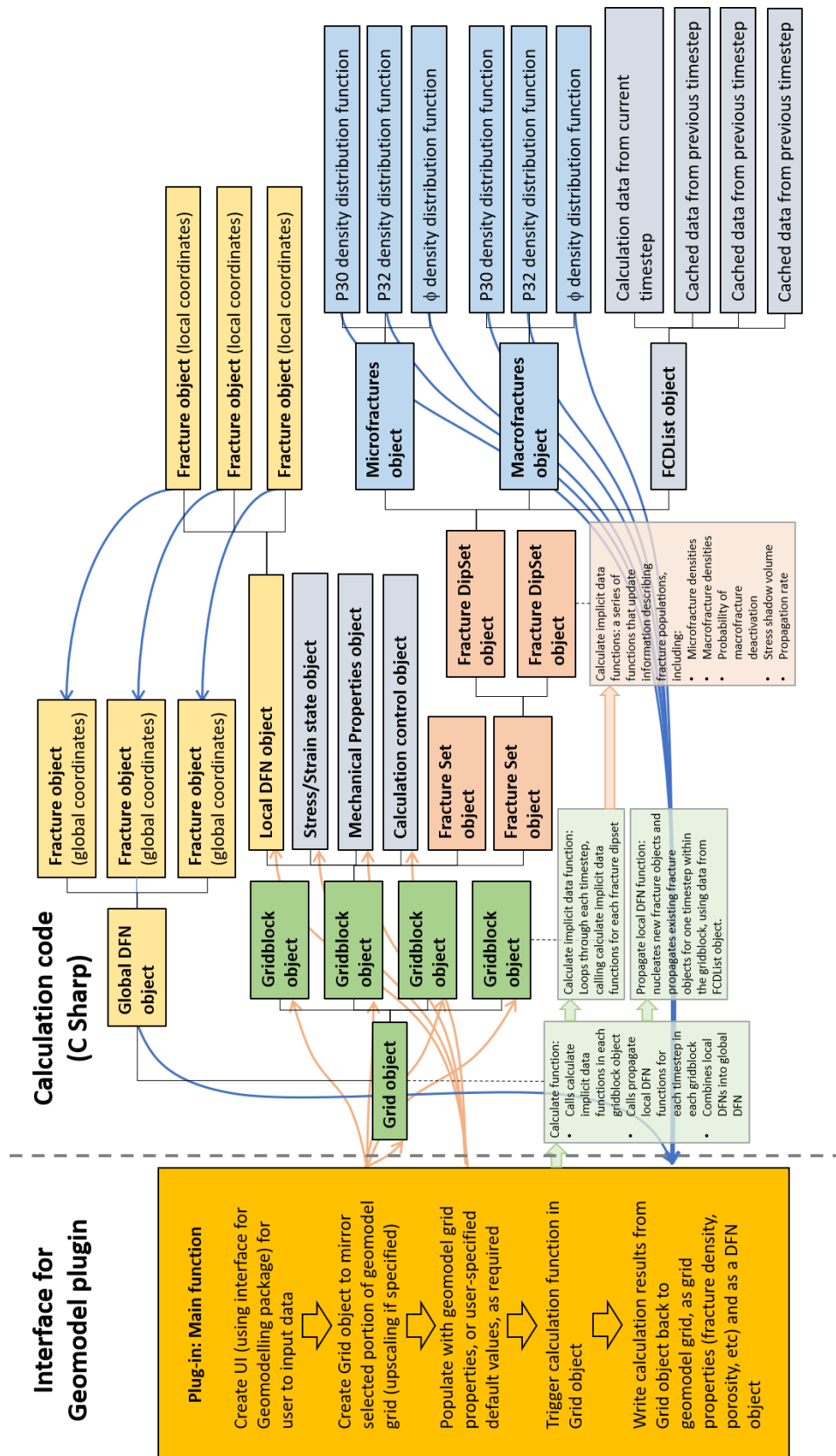


Figure 15: Schematic diagram showing the overall structure of the DFN Generator code, including the separation of interface code and calculation code, and some of the key objects and methods in the calculation code.

The DFN Generator code can have multiple interfaces, for example to handle different data formats or to interact with different geomodelling packages. Each interface is contained within its own project in the Visual Studio solution, which can be compiled separately. At present two such interfaces are available:

- A standalone interface which reads input data and writes output data to text files, written in C Sharp.
- An plug-in interface for Schlumberger's Petrel geomodelling package, written using the Ocean programming interface. Note that this will require an Ocean licence to compile and a Petrel license to run, which are available from Schlumberger. Also included in the Visual Studio solution is a separate project to generate Petrel installers for the compiled Ocean code.

The calculation code is all written in C Sharp and is all included within a shared project in the Visual Studio solution, called "DFNGenerator_SharedCode". This project will not compile by itself, but the classes defined within the project will be included whenever an interface is compiled.

We ask that anyone making changes to the code respects the separation of interface and calculation code, and ensures that all modifications and enhancements are made in the appropriate projects. In particular no interface code should be added to the DFNGenerator_SharedCode project, which should only contain standard C Sharp code. This will help to ensure that any enhancements and modifications to the calculation code can be accessed by all interfaces and thus maintain consistency between the different interfaces, and will prevent changes in one interface creating compile problems for other interfaces.

5.2 Adding a new interface

New interfaces may be added to the DFN Generator code for a variety of reasons, for example to run the DFN Generator code in a new environment, to allow data input and output in a new format, or to create a DFN Generator plug-in for a new geomodelling package. A new interface can be added by creating a new project within the DFN Generator Visual Studio solution, and adding a reference to the DFNGenerator_SharedCode project.

The exact form of the interface will depend on the required runtime environment, the type of interface or the format of the programming interface for the geomodelling package in question, but it will typically contain the following elements (see *Figure 15*):

- Code to specify a mechanism for reading the input data – for example to open a file or to establish a connection with a database, plus code to specify the exact location and format of the data describing the model geometry and each property.
- Code to specify default values where the geometry and property data cannot be read from the specified source. These may be user-defined or hardcoded (or both).
- Code to build a representation of the required model using the objects specified in the calculation code, and populate them with the required properties. This will typically be done as follows:
 - Create a FractureGrid object with the required number of rows and columns.
 - Loop through the rows and columns:
 - For each cell in the grid, create a GridblockConfiguration object.
 - Assign coordinates to the corners of the GridblockConfiguration object using geometry data read from the specified source. NB If the gridblocks are

contiguous, it will not be necessary to assign coordinates to all cornerpoints in every gridblock, as internal cornerpoints need only be specified once.

- Assign mechanical properties to the GridblockConfiguration object using data read from the specified source, or default data if this is not available.
- Assign initial stress and strain properties to the GridblockConfiguration object using data read from the specified source, or default data if this is not available.
- Assign the gridblock-specific calculation control properties to the GridblockConfiguration object using data read from the specified source, or default data if this is not available.
- Create the appropriate number of fracture sets and dipsets and assign their orientation, based on user-specified or default settings.
- Insert the new GridblockConfiguration object into the grid at the required location, and link it to neighbouring GridblockConfiguration objects where required (NB in some cases, such as faulted contacts, it may be desirable to not link neighbouring Gridblock objects).
 - Assign the global calculation control properties to the FractureGrid object using data read from the specified source, or default data if this is not available.
- Code to define a Progress Reporter class that implements the IProgressReporterWrapper interface defined in the DFNGenerator_SharedCode project. This will be used to provide updates on the progress of the calculation in an appropriate manner (e.g. by writing text to a console, through a progress bar or a graphical interface), and may also (optionally) allow the user to abort the calculation. When building a plug-in for another software package, this will typically be a wrapper for an object native to the package in question. If there is no appropriate method for updating progress, then it is possible to use the DefaultProgressReporter class specified in the DFNGenerator_SharedCode project; this implements the required interface but does not do anything.
- Code to launch the calculation(s), using the appropriate public methods defined in the FractureGrid object. The exact methods and arguments used will depend on the calculation required, which may be specified by the user or hardcoded into the interface (e.g. it may be required to generate an implicit fracture model, but not an explicit DFN). The Progress Reporter object will also be passed into the methods called, allowing them to provide updates on the progress of the calculation in an appropriate manner.
- Once the calculation has run, code to extract the required data from the objects in the FractureGrid and process or output it as necessary. The required output data may include the cumulative density distribution functions for the various fracture dipsets in each GridblockConfiguration object, the total density or porosity for all fractures in each GridblockConfiguration object, other properties extracted from the GridblockConfiguration objects (e.g. fracture anisotropy, bulk rock compliance tensor, final stress state), or the geometry and properties of the explicit fractures specified in the CurrentDFN object. Furthermore the output data may be from the final fracture network, or may be cached data from intermediate stages of fracture growth. This data may be output to file, written to a database, displayed graphically, or used to generate objects native to the software package which the interface connects to. Note that the calculation code itself may write some output data to file; this can be used for debugging or to archive the calculation data.

The interface may also contain code to provide additional feedback to the user, or to generate a GUI (e.g. a dialog box) to allow the user to specify input data and other settings. For an example of a simple interface, which builds and populates model based on text file input and then uses the file

output capabilities inbuilt into the calculation code to write output data to file, see the DFNGenerator_Standalone project (NB this interface also includes several Excel spreadsheets which can be used to analyse and display the output data from simple models). For an example of a more complex interface which generates a plug-in for a third-party geomodelling package, which reads input data from the package and creates native objects with the output, and which includes a dialog box GUI, see the DFNGenerator_Ocean project (NB this requires a license for the Ocean interface from Schlumberger to compile).

Note that it is possible to generate a new interface without making any changes to the calculation code in the DFNGenerator_SharedCode project. Interface code for third-party packages and other non-standard C Sharp code should never be included in the DFNGenerator_SharedCode project.

5.3 Structure of the calculation code

The calculation code is designed as object oriented code. It uses classes on different levels to build up a compound representation of the geology of the fractured layer, and contains in-built methods to simulate the growth of the fracture network and other important parameters (e.g. in situ stress and bulk rock compliance). The broad structure of this representation is illustrated in *Figure 15*.

In this section we describe some of the key classes and methods used. These are presented in “bootom up” order, starting with basic geometric classes and building up to the FractureGrid class that contains the entire representation of the geology of the fractured layer. However more details on individual methods and the algorithms they use are given in comments within the code itself.

Basic geometric classes: Define basic geometric elements such as points, vectors and tensors, used to describe explicit fractures as well as properties such as stress, strain and displacement in 3D.

- **PointIJK:** Point with coordinates specified in the local reference frame of the fracture set (i.e. IJK coordinates; see Section 4.2.1.4). Used to specify the nodes of microfractures and macrofracture segments.
- **PointXYZ:** Point with coordinates specified in the global reference frame of the grid (i.e. XYZ coordinates; see Section 4.2.1.2). Used to specify the cornerpoints of microfractures and macrofractures.
- **VectorXYZ:** Vector specified a global 3D Cartesian reference frame (i.e. X, Y, Z) (see Section 4.2.1.2). Includes standard vector operators such as addition, and scalar, vector and outer products. Used to specify displacement, normal, azimuth and strike of fractures, and stress and strain resolved onto a plane.
- **Tensor2S:** Second order symmetric tensor with 6 components (XX, YY, ZZ, XY, YZ, ZX). Includes standard tensor operators such as addition, multiplication with vectors and other tensors, inversion, extraction of invariants (including trace and determinant) and extraction of principal values and directions. Used to define stress and strain.
- **Tensor4_2Sx2S:** Fourth order tensor with 36 components relating two symmetric second order tensors. Includes standard tensor operators such as addition, multiplication with second and other fourth order tensors, inversion and extraction of invariants (including trace and determinant). Used to define compliance and stiffness.

Building blocks for the explicit DFN: Represent individual fracture segments, whole fractures and the fracture network.

- **MicrofractureIJK:** Discrete circular microfracture lying within the layer, referenced in local (IJK) coordinates. It is defined by a centrepoint and radius.

- **MicrofractureXYZ:** Skeleton of a discrete circular microfracture lying within the layer, referenced in global (XYZ) coordinates. It can be defined either by a centrepoint and radius, or by a series of points demarcating a polygon. It is linked to a MicrofractureIJK object representing the same microfracture in local (IJK) coordinates.
- **MacrofractureSegmentIJK:** Skeleton of a quadrilateral planar segment of a layer-bound macrofracture, referenced in local (IJK) coordinates. It is defined by two nodes demarcating the centreline and a dip azimuth and magnitude.
- **MacrofractureXYZ:** Discrete layer-bound macrofracture, referenced in global (XYZ) coordinates. It comprises one or more planar quadrilateral segments, which need not be coplanar. It is defined by a series of points marking the corners of the segments. It is linked to a series of MacrofractureSegmentIJK objects representing the individual segments in local (IJK) coordinates.
 - *PopulateData():* A function called to reconstruct the geometry of the layer-bound fracture, calculate the position of the segment cornerpoints and set the dynamic fracture data, based on the location of the segment nodes in local (IJK) coordinates, their dip and magnitude, and the top and bottom surfaces of the layer.
- **GlobalDFN:** Holder for discrete MicrofractureXYZ and MacrofractureXYZ objects representing the entire fracture network. It spans the entire grid and is referenced in global (XYZ) coordinates.

Building blocks for the implicit fracture model: Represent entire fracture populations statistically.

- **FractureCalculationData:** Cache for data from a previous timestep (e.g. fracture nucleation or growth rate) that is needed for later calculation or data output.
- **FCD_List:** Series of FractureCalculationData objects containing cached data from all previous timesteps.
- **FractureDipSet:** Representation of a single fracture dipset in a single gridblock. Contains fracture dip, current volumetric (P30) and mean linear (P32) densities for microfractures and macrofractures, cumulative fracture density distribution functions (density vs. radius for microfractures, density vs length for half-macrofractures), and various functions used to calculate fracture growth, including:
 - *getOptimalDuration(Tensor2S Sigma_Const, Tensor2S Sigma_Var, double d_MFP33):* Calculate the constant and variable components of the driving stress for the upcoming timestep, and estimate the optimal timestep duration based on a specified maximum increase in fracture density.
 - *calculateTotalMacrofracturePopulation():* Calculate the increment in total active and static half-macrofracture density (a_MFP30, sII_MFP30, sIJ_MFP30, a_MFP32 and s_MFP32) during the current timestep, in response to the applied driving stress and taking into account fracture deactivation.
 - *calculateTotalMicrofracturePopulation(int no_r_bins):* Calculate the increment in total active and static microfracture density (a_uFP30, s_uFP30, a_uFP32, s_uFP32, a_uFP33 and s_uFP33) during the current timestep, in response to the applied driving stress and taking into account fracture deactivation. Calculating uFP32 requires a numerical integration, carried out using the specified number of bins.
 - *calculateCumulativeMacrofracturePopulationArrays():* Calculate the cumulative active and static half-macrofracture density distribution functions a_MFP30(ℓ), sII_MFP30(ℓ), sIJ_MFP30(ℓ), a_MFP32(ℓ) and s_MFP32(ℓ) at the current time, for a specified list of lengths ℓ . Uses data from all previous timesteps cached in the FCD_list.
 - *calculateCumulativeMicrofracturePopulationArrays():* Calculate the cumulative active and static microfracture density distribution functions a_uFP30(r), s_uFP30(r),

$a_{uFP32}(r)$, $s_{uFP32}(r)$, $a_{uFP33}(r)$ and $s_{uFP33}(r)$ at the current time, for a specified list of radii r . Uses data from all previous timesteps cached in the `FCD_list`.

- **Gridblock_FractureSet:** Representation of a fracture set in a single gridblock. Contains fracture strike, one or more `FractureDipSet` objects representing dipsets, as well as a list of `MicrofractureIJK` and `MacrofractureSegmentIJK` objects representing individual fracture segments in the explicit DFN. Also contains various functions used to calculate data relating to the fracture set as a whole, including:
 - *calculateMacrofractureSpacingDistributionData()*: Recalculate the macrofracture spacing distribution function parameters A , B , and C and exclusion zone volume χ , for each dipset, in response to fracture growth in the previous timestep (see Section 2.1).
 - *Calculate_Phill_ByTime(FractureDipSet mp_fds, PropagationDirection mp_PropDir, double time)*: Calculate the probability that an active half-macrofracture from a specified dipset will not be deactivated due to stress shadow interaction with any other fractures from this set, within a specified time period (see Section 2.3).
 - *Calculate_Phill_ByDistance(FractureDipSet mp_fds, double propagationDistance)*: Calculate the probability that an active half-macrofracture from a specified dipset will not be deactivated due to intersection with fractures from another set, as it propagates over a specified distance (see Section 2.4).
 - *checkInMFStressShadow(PointIJK point)*: Check whether a specified point lies within the stress shadows of any of the macrofracture segments in the explicit DFN associated with this fracture set.
 - *checkInMFExclusionZone(PointIJK point, double inputfracStressShadowWidth)*: Check whether a specified point lying on a fracture in this set (in local IJK coordinates) lies within the exclusion zone of any of the macrofracture segments in the explicit DFN associated with this fracture set.
 - *checkInMFProximityZone(PointIJK point, List<double> proximityZoneHalfWidths)*: Check whether a specified point (in local IJK coordinates) lies within a proximity zone of arbitrary width around any of the macrofracture segments in the explicit DFN associated with this fracture set; the proximity zone width can vary for different fracture dip sets.
 - *checkStressShadowInteraction(MacrofractureSegmentIJK propagatingSegment, Gridblock_FractureSet interacting_fs, ref double propagationLength, bool terminateIfInteracts)*: Check whether a propagating macrofracture segment from this fracture set will terminate due to stress shadow interaction with any of the other macrofracture segments in the explicit DFN associated with this fracture set, as it propagates a specified distance.
 - *checkFractureIntersection(MacrofractureSegmentIJK propagatingSegment, Gridblock_FractureSet intersecting_fs, ref double propagationLength, bool terminateIfIntersects)*: Check whether a propagating macrofracture segment from this fracture set will intersect a non-boundary tracking macrofracture segment from another fracture set in this gridblock.
 - *checkBoundaryIntersection(MacrofractureSegmentIJK propagatingSegment, ref double propagationLength, out GridDirection boundaryCrossed, bool terminateIfIntersects, bool terminateIfNoNeighbour)*: Check whether a propagating macrofracture segment from this fracture set will intersect a gridblock boundary.

Classes defining the geological model: Classes containing the geometric and geomechanical data that define the overall geological model.

- **MechanicalProperties:** Contains data representing the mechanical properties of a single gridblock.
- **StressStrainState:** Contains data representing the stress and strain state of a single gridblock, including tensors for the in situ stress, elastic and inelastic strain.
- **GridblockConfiguration:** Class representing an entire gridblock. Contains a Mechanical Properties object representing the mechanical properties, a StressStrainState object representing the current stress and strain state, and a collection of Gridblock_FractureSet objects representing the fracture sets. Also contains various functions used to calculate the implicit and explicit fracture models, including:
 - *CalculateFractureData():* Calculate the implicit fracture model based on the parameters specified in the appropriate PropagationControl object.
 - *PropagateDFN(GlobalDFN global_DFN, DFNGenerationControl DFNControl):* Grow the explicit DFN for the next timestep, based on data from the implicit model calculation, cached in the appropriate FCD_list objects.
 - *ExtendFracture(bool use_MF_min_length_cutoff, bool checkStressShadow, bool TerminateAtGridBoundary, int fsIndex, Gridblock_FractureSet fs, MacrofractureSegmentIJK MFSegment, int dipsetIndex, ref double maxPropLength):* Extend an explicit macrofracture segment by a specified maximum amount, checking for intersection or stress shadow interactions with other fracture segments and whether it crosses the gridblock boundary.
 - *PropagateMFIntoGridblock(MacrofractureSegmentIJK initiatorSegment, int segmentFSIndex, GridDirection FromBoundary, PointXYZ insertionPoint, double newSegmentNucleationTime, bool use_MF_min_length_cutoff, bool checkStressShadow, bool TerminateAtGridBoundary):* Propagate an explicit macrofracture into this gridblock from a neighbouring gridblock.
- **PropagationControl:** Contains parameters used to control calculation of the implicit fracture model in a specific gridblock, including applied strain rate tensor.
- **DFNGenerationControl:** Contains parameters used to control calculation of the explicit DFN across the entire grid.
- **FractureGrid:** Class representing the entire fractured layer, containing a 2D array of gridblocks. Also contains various functions used to calculate the implicit and explicit fracture models, including:
 - *CalculateAllFractureData(IProgressReporterWrapper progressReporter):* Calculate fracture data for each cell in the gridblock based on the appropriate PropagationControl objects.
 - *GenerateDFN(IProgressReporterWrapper progressReporter):* Generate a global DFN based on based on the appropriate DFNGenerationControl object.

6 References

Ferrill, D.A., Morris, A.P. 2001. Displacement gradient and deformation in normal fault systems. *Journal of Structural Geology* 23, 619-638.

Huggins, P., Watterson, J., Walsh, J.J., Childs, C. 1995. Relay zone geometry and displacement transfer between normal faults recorded in coal-mine plans. *Journal of Structural Geology* 17, 1741-1755.

Nicol, A., Walsh, J.J., Watterson, J., Bretan, P.G. 1995. Three-dimensional geometry and growth of normal faults. *Journal of Structural Geology* 17, 847-862.

Walsh, J.J., Bailey, W.R., Childs, C., Nicol, A., Bonson, C.G. 2003. Formation of segmented normal faults: a 3-D perspective. *Journal of Structural Geology* 25, 1251-1262.

Welch, M.J., Lüthje, M., Glad, A.C. 2019. Influence of fracture nucleation and propagation rates on fracture geometry: Insights from geomechanical modelling. *Petroleum Geoscience* 25, 470-489. <https://doi.org/10.1144/petgeo2018-161>.

Welch, M.J., Lüthje, M., Oldfield, S.J. 2020. *Modelling the Evolution of Natural Fracture Networks*. Springer Nature Switzerland AG. <https://doi.org/10.1007/978-3-030-52414-2>.