Forschungszentrum Karlsruhe
Technik und Umwelt

Wissenschaftliche Berichte
FZKA 6155

WEISTRABA
A Code for the Numerical Analysis of Weibull Stress Parameters from ABAQUS Finite Element Stress Analysis

Procedural Background and Code Description

H. Riesch-Oppermann, A. Brückner-Foit

Institut für Materialforschung
Projekt Kernfusion

August 1998

Forschungszentrum Karlsruhe

Technik und Umwelt

Wissenschaftliche Berichte

FZKA 6155

WEISTRABA
A code for the numerical analysis of Weibull stress parameters from ABAQUS finite element stress analysis

-Procedural background and code description-

H. Riesch-Oppermann, A. Brückner-Foit*

Institut für Materialforschung
Projekt Kernfusion

*Universität Karlsruhe,
Institut für Zuverlässigkeit und Schadenskunde im Maschinenbau

Forschungszentrum Karlsruhe GmbH, Karlsruhe
1998

Als Manuskript gedruckt
Für diesen Bericht behalten wir uns alle Rechte vor

Forschungszentrum Karlsruhe GmbH
Postfach 3640, 76021 Karlsruhe

Mitglied der Hermann von Helmholtz-Gemeinschaft
Deutscher Forschungszentren (HGF)

ISSN 0947-8620

Abstract

Numerical analyses are used within the framework of the local approach to determine the critical stress at cleavage fracture. A set of ABAQUS post-processing modules serving this purpose is described in this report. The modules are intended to perform several steps that are necessary to obtain the parameters of the Weibull distribution of the critical Weibull stress at cleavage fracture. The main steps are determination of the first principal stress envelope at the experimentally obtained load levels at fracture, calculation of the Weibull stresses at fracture and an iterative maximum likelihood procedure for the distribution parameters of the Weibull stress. Some remarks on limits/modifications of the model in case of other mechanisms are also included in the report.

Zusammenfassung

WEISTRABA – Ein Programm zur numerischen Bestimmung der Parameter für die Weibullspannung aus ABAQUS Spannungsanalysen

– Verfahren und Programmbeschreibung –

Im Rahmen der Methodik des *Local Approach* wird eine kritische Spaltbruchspannung aus numerischen Analysen ermittelt. Der vorliegende Bericht enthält die Beschreibung für eine Reihe von ABAQUS postprocessing Modulen für diesen Zweck, wobei die Bestimmung der Parameter der Weibullverteilung der kritischen Weibullspannung bei Einsetzen des Spaltbruchs in mehreren Schritten erfolgt. Die wichtigsten Schritte sind zunächst die Ermittlung der Einhüllenden der maximalen Hauptspannung für die experimentell beim Bruch ermittelten Belastungsniveaus, die Berechnung der Weibullspannungen beim Bruch, sowie die iterative Ermittlung der Verteilungsparameter der Weibullspannung. Einige Anmerkungen zum Gültigkeitsbereich, bzw. zu notwendigen Modifizierungen im Falle daß andere Mechanismen vorliegen, geben einen Ausblick auf mögliche zukünftige Erweiterungen.

Contents

1

Introduction

The present analysis is part of the **European Fusion Technology Programme - EBP Structural Material** for 1995 to 1998. It is related to Work Package SM 5 **Rules for mechanical design, fabrication, and inspection**, the Task 5.2 **Fracture mechanics concept** and the subtask 5.2.1 **Fracture mechanics studies.**

Within this framework, it is intended to develop a fracture mechanics concept for the description of the ductile-to-brittle-transition behaviour of ferritic-martensitic steels. Due to the need of transferability, a concept based on the mechanisms of ductile or brittle behaviour is indispensable for the assessment of size and geometry effects, irradiation effects, and effects due to complex mechanical as well as thermal loading conditions.

In contrast to the global approaches, where geometrical limits on validity of test results are imposed to ensure transferability of test data to component design, a local approach relies on the combination of local (i.e. microstructurally based) fracture criteria and stress field analyses of selected geometries to ensure the transferability of material data. That is, within a local approach transferability is inherently guaranteed *as long as the local fracture mechanism remains unchanged*, which has to be verified by suitable investigations of the fractured specimens.

A key issue of the local fracture description is the determination of the fracture parameters, which requires considerable (numerical and experimental) efforts. Fracture parameters are obtained by numerical (FE) elasto-plastic deformation analyses of fracture tests. In the case of brittle fracture, a statistical approach is necessary because of the inherent scatter. If the metallographic investigations of fractured specimens indicate that ductile damage precedes final cleavage fracture, changes in the stress field have to be accounted for by appropriate damage models. This is outside the scope of the present investigation.

The following report is intended to give a brief description of theoretical background together with the relevant background information on the programming philosophy that is used for the calculation of the parameters of brittle fracture from experimental results and from the corresponding stress analyses.

Furthermore several modules that are necessary to perform the calculation of Weibull stress parameters from a stress analysis with the ABAQUS finite element code [1] (current version 5.6) shall be described. The first module, `fil_ou`, is used to determine the maximum principal stress together with the basic data of the finite element model from the ABAQUS `*.fil` binary results file. In the second module, `gau_wei`, a numerical integration is performed to calculate the Weibull stress for a given load step. The third module, `wei_ml`, contains the statistical analysis and the determination of the distribution parameters by the maximum likelihood method. The module `fil_ou` is a stand-alone one because its purpose is data processing only. The second and third module, `gau_wei` and `wei_ml`, are linked with each other because the statistical analysis has to be performed iteratively with repeated calculations of the Weibull stresses.

Special attention is paid to the application to components of nuclear fusion reactors and ferritic-martensitic reduced activation alloys. In this case, neutron irradiation leads to a pronounced shift in the ductile to brittle transition temperature and combined thermal and mechanical loading lead to specific design requirements that are still under development.

Some remarks on modifications that might be necessary to handle these different fields of application are made. This includes constraint effects, irradiation effects, and effects of large stress gradients.

In the Appendix, a documentation of the relevant subroutines of the three modules developed up to now is given.

Additionally, a reference example shall be presented for the Weibull stress calculation taken from experiments on F82Hmod and calculations that are performed within the European Blanket Programme.

2

Basic relations for cleavage fracture analysis

In this chapter, a short summary of the theoretical relations which are the basis of the developed computer programs is given. The Weibull stress is defined and the necessary relations for numerical integration of the finite element stress results are given. Also, the maximum likelihood procedure is summarized. Special attention is given to the iterative procedure which is necessary in order to obtain the Weibull stress distribution parameters correctly. Finally, some remarks are made on limitations of the use of the Weibull stress concerning effects of changes in the stress field (constraint effects, steep stress gradients) and fracture mechanism (ductile damage).

2.1 The Weibull stress as cleavage fracture parameter

The Weibull stress at cleavage fracture is a random variable that characterizes the fracture resistance of the material against cleavage (brittle) fracture. The Weibull stress σ_W is defined by

$$\sigma_W^m = \frac{1}{V_0} \int_{V_{pl}} \sigma_1^m \, \mathrm{d}V \tag{2.1}$$

where m is the so-called Weibull slope, V_0 is a reference volume, V_{pl} is the volume of the plastic zone, and σ_1 is the first principal stress.

The statistical distribution of its critical value, e.g. the value at cleavage fracture is given by

$$F_{\sigma_W}(\sigma_W) = 1 - \exp\left(-\left(\frac{\sigma_W}{\sigma_u}\right)^m\right) \tag{2.2}$$

The distribution parameters σ_u and m of the Weibull stress σ_W at fracture are material parameters (i.e. independent of the stress state in the material), but may depend on temperature.

The Weibull slope m characterizes the scatter of the Weibull stress. The coefficient of variation (C.O.V) of σ_W is a function of m alone and given by

$$COV_{\sigma_W} = \frac{\sqrt{\Gamma(1+\frac{2}{m}) - \left(\Gamma(1+\frac{1}{m})\right)^2}}{\Gamma(1+\frac{1}{m})} \tag{2.3}$$

where $\Gamma(.)$ denotes the Euler's gamma function.

The parameter σ_u gives the $1 - 1/e$ (=63.2%)-quantile of σ_W.

The reference volume V_0 which appears in eq. (2.1) is introduced for dimensional purposes only and set to 1mm^3 unless stated otherwise [1].

For the analysis, the Weibull stress at fracture has to be determined from suitably chosen experimental loading parameters, such as e.g. the diameter reduction for notched tensile specimens at fracture or the value of the J-integral for cracked specimens.

2.2 Numerical integration of finite element stress results

For numerical reasons, the integration of the Weibull stress according to eq. (2.1) is performed after normalizing σ_1 by a suitably chosen reference stress, e.g. the flow stress. This is done to avoid numerical difficulties resulting from large values of the Weibull exponent m which is typically in the range of 10-30. The correction is removed after the numerical integration is complete. Eq. (2.1) then reads:

$$\left(\frac{\sigma_W}{\sigma_{\text{ref}}}\right)^m = \frac{1}{V_0}\int_{V_{pl}} \left(\frac{\sigma_1}{\sigma_{\text{ref}}}\right)^m \mathrm{d}V \tag{2.4}$$

and final correction is simply made by multiplying the resulting integral value by the value of the reference stress σ_{ref}^m.

The first principal stress values are obtained from the ABAQUS stress analysis with the help of the postprocessing routine `fil_ou` which is described below. Stresses are given at the integration points of the ABAQUS elements. Reduced integration is used, which means that we have 2x2=4 integration points per element in the 2D case and 2x2x2=8 integration points in 3D problems. The Weibull stress is integrated element-by-element. In the general case of a 3D model, we have

$$\begin{aligned} \sigma_W &= \sigma_{\text{ref}} \left[\frac{1}{V_0}\sum_{\text{el}} \sigma_{W_{\text{el}}}\right]^{\frac{1}{m}} \qquad \text{with the auxiliary quantity of} \\ \sigma_{W_{\text{el}}} &= \sum_{i=1}^{k_i} w_i \sum_{j=1}^{k_j} w_j \sum_{k=1}^{k_k} w_k \left(\frac{\sigma_1(r_i, s_j, t_k)}{\sigma_{\text{ref}}}\right)^m (\det J(r_i, s_j, t_k)) \end{aligned} \tag{2.5}$$

with k_i, k_j, k_k the number of integration points in each dimension and w_i, w_j, w_k the respective weights. The contributions from each element are summed up to give the final result. For $k_i = k_j = k_k = 2$, we have $w_i = w_j = w_k = 1$ and r_i, r_j, $r_k = 1/\sqrt{3}$.

A plastic zone indicator flag is used to extend numerical integration only over the plastic zone and not over the entire volume of the specimen. This plasticity flag is set to 1 for each integration point where plasticity occurs (in terms of a von Mises yield criterion or by checking the plastic strains of the FE output) and 0 otherwise. Any averaging procedures are avoided. Only the stress values at the integration points, which are known to be the most exact values within an element [4], are used.

[1] Some authors use V_0 as an additional parameter related to σ_u (see e.g. [2, 3]) chosen to be small enough that stress gradients can be neglected and large enough that the weakest link argument for finding a microcrack of a given size still holds (e.g. 10 grains). If stress gradients are important for the fracture behaviour, this can be directly incorporated into the fracture model leading to eq. (2.2) at the expense of losing the meaning of σ_u and m as material parameters (see below).

For each FE load step, corresponding to a specimen fracture event, the first principal stress values are checked against the values of the previous step and a stress envelope is constructed to take into account locally decreasing stresses due to stress redistribution which may lead to decreasing values of the local risk of rupture.

2.3 Maximum likelihood procedure

The determination of the two parameters m and σ_u has to be performed iteratively as σ_W depends on the (unknown) parameter m.

Step 1: A starting value of e.g. $m = 20$ is used and the Weibull stress σ_W at fracture is calculated for each fractured specimen (i.e. at different load steps according to the experimental loading parameter) as described above.

Step 2: A plotfile is generated containing the results in increasing order of Weibull stress σ_W together with $\ln \ln[\frac{1}{1-\overline{F(x_n)}}]$ as a function of $\ln x_{(n)}$, where $x_{(n)}$ is the Weibull stress of the specimen with rank n and $\overline{F(x_n)} = \frac{n}{N+1}$ is the mean (cumulative) frequency of the n-th observation (using $\frac{n}{N+1}$ as plotting position is generally recommended for statistical reasons – e.g. [5] –, although it plays no role provided that the maximum likelihood method is used for parameter estimation). As the theoretical relation between failure probability and σ_W is given by

$$P_f = 1 - \exp\left[-\left(\frac{\sigma_W}{\sigma_u}\right)^m\right] \quad ,$$

a plot of $\ln \ln[\frac{1}{1-\overline{F(x_n)}}]$ versus $\ln \sigma_{W_{(n)}}$, where $\sigma_{W_{(n)}}$ is the "experimental" Weibull stress for the specimen with rank n, should give an approximately linear relation.
(Step 2 is only for illustration and not necessary for Step 3)

Step 3: The maximum likelihood method is used to determine the parameters m and σ_u of the Weibull distribution of the Weibull stress. The maximum likelihood estimators of m and σ_u are denoted by $\hat{m}$ and $\widehat{\sigma_u}$, respectively. $\hat{m}$ is the solution of the nonlinear equation

$$\frac{N}{\hat{m}} + \sum_{i=1}^{N} \ln \sigma_{W_{(i)}} - N \frac{\sum_{i=1}^{N} \sigma_{W_{(i)}}^{\hat{m}} \ln \sigma_{W_{(i)}}}{\sum_{i=1}^{N} \sigma_{W_{(i)}}^{\hat{m}}} = 0$$

which is obtained by an interval sectioning procedure. Using $\hat{m}$, the maximum likelihood estimator $\widehat{\sigma_u}$ is obtained from the equation

$$\widehat{\sigma_u} = \left(\frac{1}{N} \sum_{i=1}^{N} \sigma_{W_{(i)}}^{\hat{m}}\right)^{\frac{1}{\hat{m}}}$$

The parameter $\hat{m}$ is corrected with the unbiasing factor $b(N)$ obtained by subroutine `BIAS` described below (see also Table B.1). $\hat{m}_{unb} = \hat{m} * b(N)$.

Step 4: If the maximum likelihood estimators $\widehat{\sigma_u}$ and $\hat{m}_{unb}$ agree within a fixed tolerance with those of the previous iteration, their values are considered acceptable. Otherwise, steps 2-4 are repeated. A flow diagram is given in Figure 2.1 to illustrate the iterative procedure.

Confidence intervals for m and σ_u

Confidence intervals for the Weibull parameters m and σ_u determined by the maximum likelihood method are obtained according to the following procedure:

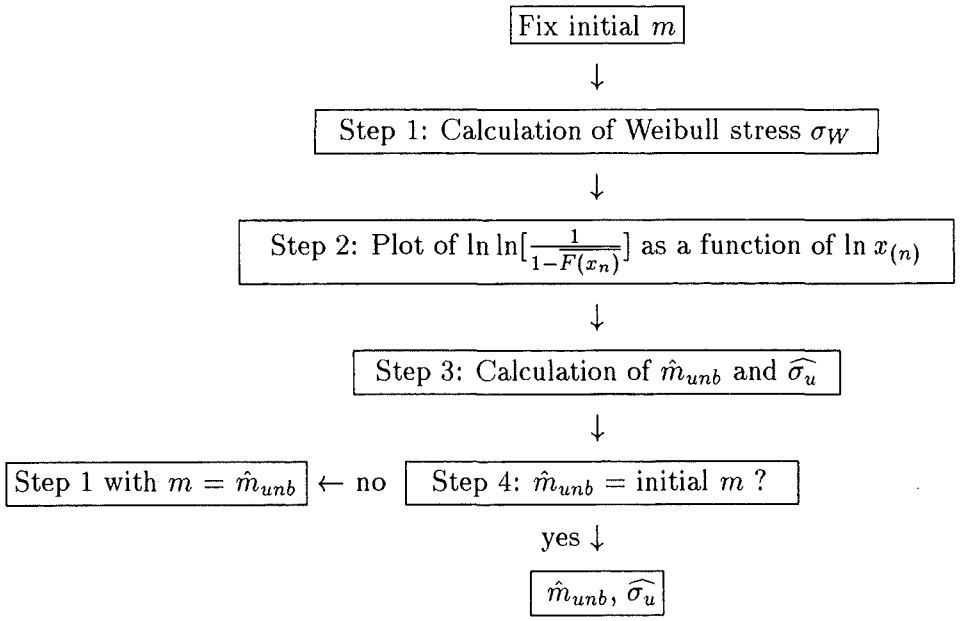

Figure 2.1: Flow diagram for iterative Weibull parameter estimation procedure

1. For a confidence level $1 - \alpha$ (80 %, 90% or 96% is possible, i.e. $\alpha = 0.20$, $\alpha = 0.10$ or $\alpha = 0.04$) $\alpha_1 = \alpha/2$ and $\alpha_2 = 1 - \alpha/2$ are calculated.

2. $t_1(N, \alpha_1)$ and $t_2(N, \alpha_2)$ are taken from Table B.2
   $A = \widehat{\sigma_u} * \exp(-t_2/\hat{m})$ and $B = \widehat{\sigma_u} * \exp(-t_1/\hat{m})$ are calculated.
   $[A, B]$ is reported to be the confidence interval for σ_u for a confidence level of $1 - \alpha$.

3. $l_1(N, \alpha_1)$ and $l_2(N, \alpha_2)$ is taken from Table B.3
   $C = \hat{m}/l_2$ and $D = \hat{m}/l_1$ are calculated.
   $[C, D]$ is reported to be the confidence interval for m for a confidence level of $1 - \alpha$.

These quantities have to be calculated with the maximum likelihood estimate of m without the unbiasing factors.

Note: The confidence intervals for m and σ_u are valid **only, if** $\hat{m}$ and $\widehat{\sigma_u}$ were obtained by the maximum likelihood method. Any other estimation procedure for the Weibull parameters yields different confidence intervals.

The Tables B.1 - B.3 were taken from Ref. [6]. [2]

Upon completion of the analysis, a plotfile is generated. It contains the values of the Weibull stress as well as preprocessed data in a form that allows immediate generation of a Weibull plot via some plotting programs like e.g. `gnuplot`. Figure 2.2 shows an example of a Weibull plot template. The calculated values for $\ln \sigma_W$ are plotted together with the Weibull distribution which is a line with the slope $\hat{m}$ and containing the point $(\widehat{\sigma_u}, 0)$.

2.4 Some remarks on constraint corrections

Constraint correction is inherent in the Weibull stress for fracture mechanics (precracked) specimens (see e.g. [7]).

[2] An EXCEL template for the evaluation of the Weibull parameters is available from the authors.

For a power-law hardening material

$$\frac{\epsilon}{\epsilon_0} = \frac{\sigma}{\sigma_0} + \alpha\left(\frac{\sigma}{\sigma_0}\right)^n \tag{2.6}$$

(σ_0 - reference stress, $\epsilon_0 = \sigma_0/\epsilon_0$, n - hardening exponent, α - strain offset at σ_0), the stress field in the vicinity of a mode I crack tip can be described by a three-term asymptotic expansion [8]

$$\frac{\sigma_{ij}}{\sigma_0} = A_0\bar{r}^s\tilde{\sigma}_{ij}^{(0)}(\theta) - A\bar{r}^t\tilde{\sigma}_{ij}^{(1)}(\theta) + \frac{A^2}{A_0}\bar{r}^{2t-s}\tilde{\sigma}_{ij}^{(2)}(\theta) \tag{2.7}$$

with the dimensionless quantities

$$\begin{aligned} \bar{r} &= \frac{r}{J/\sigma_0}, \text{ where } J \text{ is the J-integral} \\ \tilde{\sigma}_{ij}^{(k)} &- n \text{ - dependent angular stress functions.} \end{aligned} \tag{2.8}$$

The coefficient A_0 is given by

$$A_0 = (\alpha\epsilon_0 I_n)^{-1/(n+1)} \tag{2.9}$$

with I_n according to [9]. The exponent $s = -1/(n+1)$ is theoretically known [9, 10] and for the exponent r an eigenvalue problem has to be solved (e.g. [8]). The amplitude A is determined by curve fitting of eq. (2.7) to FE crack tip stress results. The three-term approximation of the stress field is used for the calculation of the Weibull stress σ_W (see eq. (2.1)). In case of small scale yielding, i.e. if the first term of eq. (2.7) yields a good approximation of the stress field, and for two-dimensional cracks with a constant J along the crack front, it can be shown [2, 3] that σ_W can be re-written as

$$\sigma_W^m = \frac{J_c^2}{\sigma_0^2}\frac{B}{V_0}\int_{U_{pl}} \sigma_1^m \, \mathrm{d}U \tag{2.10}$$

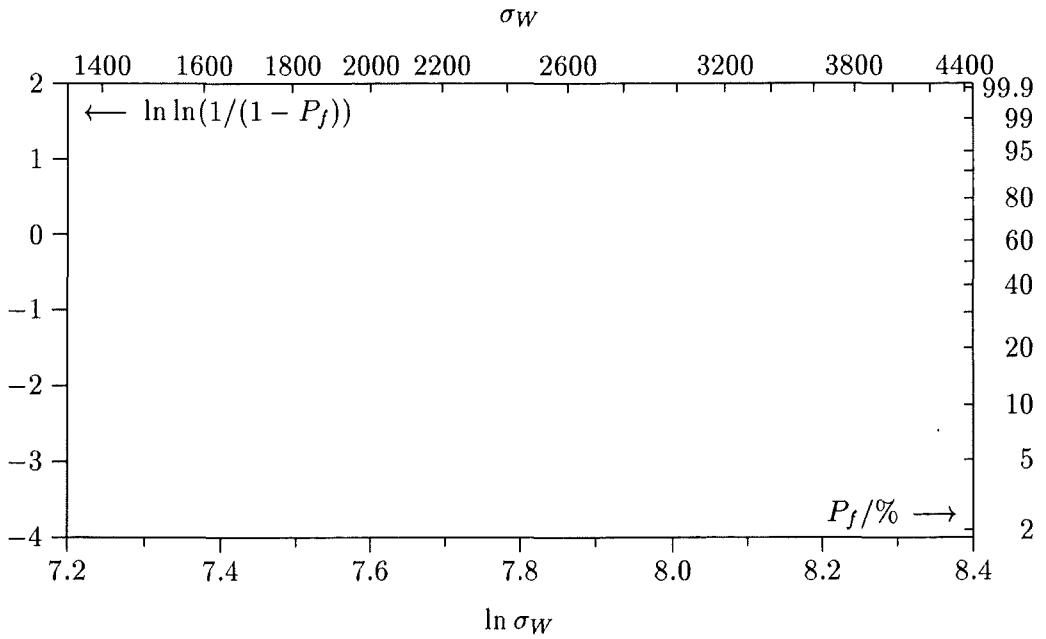

Figure 2.2: Template for Weibull plot of σ_W

where B is the specimen thickness, J_c is the value of the J-integral at the onset of cleavage fracture and U_{pl} is the normalized plastic zone size of a specimen of unit thickness given in terms of $\bar{r} = rJ/\sigma_0$.

Under small-scale yielding conditions, and in case of a constant J along the crack front, σ_W and J_{Ic} can thus be expressed in terms of each other by identifying corresponding values of the cumulative distribution function of both quantities. This leads to the relation (see e.g. [2])

$$1 - \exp\left(-\left(\frac{\sigma_W}{\sigma_u}\right)^m\right) = 1 - \exp\left(-\left(\frac{J_{\mathrm{Ic}}}{b}\right)^2\right) \tag{2.11}$$

where b is a distribution parameter of the J_{Ic} distribution, or, solved for σ_W:

$$\left(\frac{\sigma_W}{\sigma_u}\right)^m = \left(\frac{J_{\mathrm{Ic}}}{b}\right)^2 \qquad \text{or} \qquad J_{\mathrm{Ic}} = b\left(\frac{\sigma_W}{\sigma_u}\right)^{m/2} \tag{2.12}$$

If there is a significant loss of constraint, higher-order terms are needed for a description of the stress field. In this case, the Weibull stress is of the form

$$\left(\frac{\sigma_W}{\sigma_0}\right)^m = \frac{J_c^2}{\sigma_0^2}\frac{B}{V_0}G(A, M) \tag{2.13}$$

where M stands for the material parameters and the dimensionsless function

$$G(A, M) = \int_{U_{pl}} \left(\frac{\sigma_1}{\sigma_0}\right)^m \mathrm{d}U \tag{2.14}$$

depends on the load level only and not explicitly on the crack size or specimen geometry. Thus, it is possible to select a reference solution for A, e.g. the small-scale yielding value, A_{SSY}. A_{SSY} can be obtained by a modified boundary layer solution for small-scale yielding for suitably selected values of the stress intensity factor K and the amplitude T resulting in prescribed elastic displacements at the boundary of the elasto-plastic boundary value problem. For a given value of σ_W, J_c can then be transformed into an equivalent small-scale yielding value, J_{SSY}, by

$$\left(\frac{J_c}{J_{\mathrm{SSY}}}\right)^2 = \frac{G(A_{\mathrm{SSY}}, M)}{G(A, M)} \tag{2.15}$$

which, as $\sigma_W \propto J^{2/m}$ holds for SSY, implies that the failure probability can be written in terms of a Weibull distribution for the transformed values of J_{SSY} with a shape factor of $m = 2$.

For a given amplitude A, is is thus possible to predict J_{Ic} from the σ_W results using the following two-step procedure:

1. Compute σ_W at fracture from the J_c results according to eq. (2.13).

2. Calculate J_{Ic} from experimentally obtained J_c values according to eq. (2.15) and determine the parameter b of the J_{Ic} distribution (which is a Weibull distribution with $m = 2$)

This procedure additionally allows the scatter bands in the data to be determined by using the appropriate relations for the respective quantiles of σ_W and J_{Ic}.

Analysis of literature data in Ref. [7], where J_c/J_{Ic} was predicted for typical fracture mechanics specimens (ECP, CCP, 3PB, CT) showed promising results. Good agreement was found for the following function

$$G(A, M) = \exp\left(a_0(M) + a_1(M)A + a_2(M)A^2\right) \tag{2.16}$$

with the material-dependent parameters a_i.

The essential advantage of this scaling approach is the fact that there is no explicit dependence of crack size or specimen geometry. Thus, the stress field is characterized by the (elastic) boundary conditions (J and T) of an elasto-plastic (modified) boundary layer problem, from which the scaling function $G(A, M)$ is deduced, and FE analysis of the specimen is replaced by use of appropriate stress amplitudes in the MBL approach.

In the field of fusion applications, this scaling approach seems to be especially promising for the processing of data from small (subsized) specimen testing results, provided that the influence of material heterogeneity on this scale still allows the use of a continuum mechanics approach.

2.5 Some remarks on transition and influence of ductile damage

In the transition regime, a competitive process between ductile and cleavage fracture mechanisms takes place. Void nucleation and growth may change the stress and strain field and final cleavage fracture can only occur, if the stresses remain sufficiently high to trigger unstable crack propagation. So, for a volume element $\mathrm{d}V$, the two competitive processes can be stated as follows: Cleavage occurs, if a critical cleavage stress is exceeded in $\mathrm{d}V$, ductile failure by void coalescence occurs, if a critical void volume fraction $f_0^c(\epsilon_{eq}^p)$ (depending on the equivalent plastic strain ϵ_{eq}^p) is exceeded in $\mathrm{d}V$.

The modelling of this competitive process must consider the respective probabilities. In case of ductile and cleavage fracture being independent of each other, the respective survival probabilities multiply and give the overall survival probability for combined fracture. A detailed analysis would exceed the scope of the present report.

2.6 Some remarks on potential application to irradiation hardening

In terms of the local approach, irradiation damage is described mainly by the influence on yield stress. It turned out that the critical cleavage stress is not affected by irradiation effects. The basic framework of the local approach is therefore easily adopted and it is only necessary to identify a suitable description of irradiation hardening [12, 13]. It has to be ensured, however, that for the material under consideration neutron irradiation does not generate additional populations of flaws.

2.7 Some remarks on steep stress gradients

If steep stress gradients exist, which means that the assumption of a constant stress along the existing cleavage origins is violated, the weakest link argument leading to the Weibull distribution

of σ_W still holds. However, the fracture mechanics description of the cleavage origins as micro-cracks with a critical crack size $a_c \propto 1/\sigma^2$ is no longer valid. Instead, a_c depends not only on the magnitude of the local stress field, but additionally on the stress gradient, or, equivalently, on the location of the crack. As a consequence, m loses its significance as material parameter [14]. Steep stress gradients may be relevant at very low temperatures due to very small plastic zone sizes as well as for thermal loading. In these cases, weight function methods are necessary and the evaluation of the stress integral requires the use of location-dependent critical stresses which may be obtained e.g. by neural network approaches [15].

3

Program description

In the following sections, programming considerations for the three modules of `fil_ou`, `gau_wei`, `wei_ml` are given together with an application example from a series of notched round bar tests at the lower shelf.

3.1 Scope of the modules

The analysis is performed with three different modules. The first module is used to extract the stresses and plastic zone size from finite element results. The second and third modules are used to determine the Weibull stress parameters. This allows the stress analysis to be performed independently of the final ML procedure for the Weibull stress. Communication between the modules is via a communication file `*.wst` generated from the FE output (e.g. ABAQUS `*.fil`-file) whose structure will be described below.

3.1.1 General features

Maximum array sizes are given in `PARAMETER` statements (see Table 3.1). The `PARAMETER` statements of Table 3.1 are compiled in a separate file `PARAM` which is included in the respective subroutines via a FORTRAN `INCLUDE` statement. Most of the variables and arrays are transferred to the subroutines via the different COMMON blocks which are given in Table 3.2 together with the maximum sizes of the arrays. The COMMON blocks of Table 3.2 are compiled in a separate file `COMMON` which is included in the respective subroutines via a FORTRAN `INCLUDE` statement.

3.1.2 Stress analysis module

In the stress analysis module named `fil_ou`, an analysis of the ABAQUS result file `*.fil` is performed. The communication file `*.wst` for the ML procedure is generated for this purpose. It contains the data describing the FE model, the maximum principal stresses at the integration points of each element and a plasticity flag for each integration point.

3.1.3 Maximum likelihood module

The iterative procedure for the determination of the Weibull modulus, m, and the parameter σ_u of the Weibull stress σ_W is performed in the maximum likelihood module named `wei_ml`. The

Variable	Description	Default
UN, HA, QU,	auxiliary quantities: 1.D0, .5D0, .25D0	-
ZE, PI	0.D0, 3.1415926...D0	-
MAXDIM	dimension of the FE model	3
MAXEL	maximum number of elements	3000
MAXIPE	maximum number of GAUSS points per element	20
MAXNPE	maximum number of nodes per element	20
MAXITW	maximum number of iterations for ML procedure	100
MAXSTP	maximum number of load steps in FE analysis	100
MAXNO	maximum number of nodes	10000
MAXIP	maximum number of integration points (= MAXIPE*MAXEL)	-
WI1, WI2	weights for GAUSS quadrature	1.

Table 3.1: Variables defined in PARAMETER statements. Most variables are used to adjust array dimensions in COMMON blocks in order to save memory.

COMMON block name	Variable (bounds)
MESH	NELEMS, NNODES, NDIM, NPE, IPE
ELEMTS	NODE(MAXEL,MAXNPE),COOR(MAXNO, MAXDIM)
STRESS	STRSIP(MAXEL, MAXIPE, 8),S1ENV(MAXEL, MAXIPE)
SHAPEF	HI(MAXNPE)
SHAPED	DHIDR(MAXIPE,MAXNPE),DHIDS(MAXIPE,MAXNPE),
	DHIDT(MAXIPE,MAXNPE)
JACMAT	DJACM(MAXDIM,MAXDIM), DJDET
GAUQUA	WI(2),GP(2)
SIGW	WSTR, WM, WMO, SIGU, SIGREF, SIGW(MAXSTP)
PZONE	PZVOL,IPFLAG(MAXEL,MAXIPE)
AUX	IWUN, FNAME, ELTYPE

Table 3.2: Variables declared in COMMON blocks.

necessary input is read from the communication file *.wst and the Weibull stress values for each experimental load step at fracture are calculated using a starting value of m. This gives a sample of Weibull stresses at fracture, which is used for the maximum likelihood estimates for the distribution parameters $\hat{m}$ and $\widehat{\sigma_u}$ and the corresponding unbiased value $\hat{m}_{unb}$. If $\hat{m}_{unb}$ coincides with the starting value, m, the procedure is terminated, otherwise m is set to $\hat{m}_{unb}$ and the iteration is continued until convergence is achieved.

3.2 Program structure

3.2.1 Module fil_ou

The module fil_ou has to organize input data for the Weibull stress analysis; for this purpose, the ABAQUS binary results file *.fil is analysed record by record and a communication file *.wst is generated for the maximum likelihood (ML) Weibull stress analysis. The *.wst communication

Name	Description
NELEMS	number of elements in FE model
NNODES	number of nodes in FE model
NDIM	dimension of FE model
NPE	number of nodes per element in FE model
IPE	number of integration points per element in FE model
NODE(*,*)	contiguity list of nodes for each element
COOR(*,*)	array of initial nodal coordinates
STRSIP(*,*,*)	stress tensor at integration points for each element
S1ENV(*,*)	array of 1st principal stresses at integration points for each element
HI(*)	array of shape function values at integration points
DHID[RST](*,*)	derivatives of shape functions with respect to reference coordinates
DJACM(*,*)	Jacobi matrix
DJDET	determinant of Jacobi matrix
WI(*)	weights for Gaussian quadrature
GP(*)	evaluation points for Gaussian quadrature
WSTR	Weibull stress for current load step
WM	Weibull modulus for current iteration
WMO	Weibull modulus for previous iteration
SIGU	parameter σ_u
SIGREF	reference stress value for plasticity flag
SIGW(*)	array of Weibull stresses for all load steps
PZVOL	volume of plastic zone
IPFLAG(*,*)	plasticity flag
IWUN	FORTRAN unit of communication file *.wst
FNAME	root name of communication file *.wst
ELTYPE	ABAQUS element type

Table 3.3: Description of variables in COMMON blocks.

file contains data of the finite element model (e.g. nodal and element data) and the envelope of the first principal stress during successive loadcases (array S1ENV) as well as a flag (array IPFLAG) that indicates whether the stress evaluation point lies within the plastic zone or not. This facilitates the subsequent numerical integration procedure. The plasticity flag array IPFLAG is determined using either a von Mises yield criterion or equivalent plastic strain values from the ABAQUS results file. Details of the structure of the *.wst-file are contained in Table A.1.

3.2.2 Module gau_wei

In the module gau_wei, the numerical integration of the Weibull stress is performed. First, finite element nodal and coordinate data are initialized (subroutine PSENV, 1st call) by reading them from the *.wst-file and the derivatives of the interpolation functions for the selected finite element type are evaluated at the integration points of the unit reference element (subroutines DHIDRS for 2-D problems or DHDRST for 3-D problems, respectively, depending on the dimension of the FE model). This is only done at the beginning of the calculation procedure in order to reduce the computational effort.

The following steps are then repeated for every load step. The principal stress envelope array S1ENV for load step LST is read together with the corresponding plasticity flag array IPFLAG (subroutine PSENV, 2nd and subsequent calls). Then, numerical integration of the maximum

Name	Purpose
`wstrit`	main program - controls iterative procedure for ML parameters
`bias`	bias correction of maximum likelihood estimate for m
`dhidrs`	derivatives of shape functions for 2D (8 node) elements at integration points
`dhdrst`	derivatives of shape functions for 3D (20 node) elements at integration points
`djac2d`	calculation of Jacobi matrix and determinant (2D case)
`djac3d`	calculation of Jacobi matrix and determinant (3D case)
`int2ws`	calculation of Weibull stress from 1st principal stresses (2D)
`int3ws`	calculation of Weibull stress from 1st principal stresses (3D)
`maxl`	maximum likelihood estimation of a given sample of σ_W
`output`	print some results
`psenv`	read `*.wst`-file; FE data and principal stress envelope

Table 3.4: Subroutine names and their purpose

principal stress envelope is performed by subroutine `INT2WS` for 2D problems or `INT3WS` for 3D problems. Gaussian integration is used with 2 evaluation points per element in each dimension (corresponding to reduced integration element types in ABAQUS). Subroutines `DJA2D/DJAC3D` supply the Jacobi determinant which is necessary for the volume integration to be performed in the standard unit element. The stress envelope is normalized by a suitably chosen reference stress `SIGREF`, e.g. the flow stress, in order to avoid numerical difficulties caused by high values of the Weibull exponents. The normalization is corrected after the numerical integration is completed. A summary is printed upon the completion of all load steps (subroutine `OUTPUT`).

3.2.3 Module `wei_ml`

After completion of all load steps corresponding to a sample of failed specimens, the parameters of the statistical distribution of σ_W are estimated using the maximum likelihood (ML) method. Subroutine `MAXL` serves this purpose. A bias correction for the Weibull modulus m is calculated by the function `BIAS`. Bias correction is available for sample sizes between 5 and 120. Module `gau_wei` is used again until convergence of the parameter m is achieved.

Confidence intervals for m and σ_u are given (subroutine `CNFLIM`) at the end of the iterative procedure.

Subroutine `MLPLOT` generates a plotdata file for the graphical presentation of the ML results.

3.3 Application example

In this section, the calculation of the Weibull stress for a set of round notched bar tensile specimens at a lower shelf temperature of -150°C is presented for a ferritic-martensitic steel designated F82Hmod. Details on the material can be found elsewhere [11], whereas a sketch of the numerical procedure is outlined below with special emphasis being put on the structure of data processing and the processing of experimental results.

3.3.1 Stress analysis and envelope of the maximum principal stress

Experimental results are available in terms of load vs. diameter reduction recordings, from which the respective values at fracture are obtained. An elasto-plastic ABAQUS finite element analysis

```
/v1/home/imf2/riesch/netze/f82hmod/f82h2w
  810.d0   22.d0    33  (sig0 fuer xtra bzw. SIGREF, wm0, iwun fuer wstress)
   2        8       4  (ndim, npe, ipe )
  15                   (lstmax)
```

Table 3.5: Example of auxiliary file.

is performed covering the whole range of observed diameter reduction recordings and resulting in a numerical load vs. diameter reduction curve. Numerical and experimental data are plotted and should coincide, otherwise the numerical model has to be improved, e.g. by taking into account ductile damage. ABAQUS load step control has to be chosen such that the diameter reductions at fracture are met by the displacement boundary conditions of the successive steps. This facilitates the subsequent Weibull stress analysis. The final ABAQUS `*.fil` results file is assumed to contain the values of the maximum principal stress at the integration points of each element. This has to be ensured by appropriate ABAQUS output control statements.

An auxiliary file is used for the determination of the envelope of the maximum principal stress. This file (see example in Table 3.5) is read from stdin and contains the file name to be read (recommended name is base name of ABAQUS `*.fil` file – line 1), a reference stress value (e.g. the flow stress – line 2), a suitably chosen starting value for m (e.g. 22 – line 2), the Fortran unit number for the `*.wst` file to be generated (e.g. 33 – line 2), the dimension, number of nodes and integration points per element used in FE analysis (line 3), and the number of load steps at which the Weibull stress is calculated (i.e the number of specimens – line 4). Upon completion of the procedure, the communication file `*.wst` is written in the directory, where the ABAQUS results file resides and which is to be specified in the auxiliary file.

3.3.2 Weibull stress parameters

With the completion of the `*.wst`-file, all data are available for the iterative Weibull stress parameter calculation. Modules `gau_wei` and `wei_ml` use the same auxiliary file as module `fil_ou`, thus ensuring consistency of the reference stress for both purposes. The results are written on a `*.dat` file that resides in the current directory (i.e. `/weistr/`) and contains a (comprehensive) printout of the iterative procedure, a summary of the calculated Weibull stresses for each iteration step, and the Weibull stress distribution parameters together with a plotdata segment which can be used to generate a Weibull diagram.

3.3.3 Presentation of results - graphics

A Weibull diagram template is provided for use with the `gnuplot` plotting program. The plotdata segment of the output file is ready to be used by a `gnuplot` plot statement. An example of the plotdata file is given in Table 3.6, while the corresponding plot is shown in Figure 3.1.

```
# PLOT: GENERATE PLOTFILE FOR ML RESULTS ( AT THE MOMENT: GNUPLOT-FORMAT)
#  RESULTS FROM .wst-FILE: /v1/home/imf2/riesch/netze/f82hmod/f82h2w
#  SIGU=   2106.668       WMB =   11.77931       WMUB=   10.69561
#  LN(SIGW)    LN LN 1/(1-FI)    SIGW        FI
   7.448905        -2.740493        1717.981        0.6250000E-01
   7.486048        -2.013419        1782.992        0.1250000
   7.493316        -1.571953        1795.998        0.1875000
   7.573672        -1.245899        1946.274        0.2500000
   7.594648       -0.9816471        1987.531        0.3125000
   7.606468       -0.7550149        2011.162        0.3750000
   7.612556       -0.5527521        2023.444        0.4375000
   7.631592       -0.3665129        2062.332        0.5000000
   7.631593       -0.1903393        2062.332        0.5625000
   7.634804       -0.1935689E-01    2068.966        0.6250000
   7.652639        0.1511325        2106.196        0.6875000
   7.654677        0.3266343        2110.494        0.7500000
   7.671654        0.5152019        2146.628        0.8125000
   7.673846        0.7320994        2151.340        0.8750000
   7.799211         1.019781        2438.678        0.9375000
# MLPLOT: PLOTFILE GENERATED...
# COLS ARE: LN(SIGW(I)),LN LN 1/1-FI,SIGW(I),FI
```

Table 3.6: Plotdata file for part of Figure 3.1.

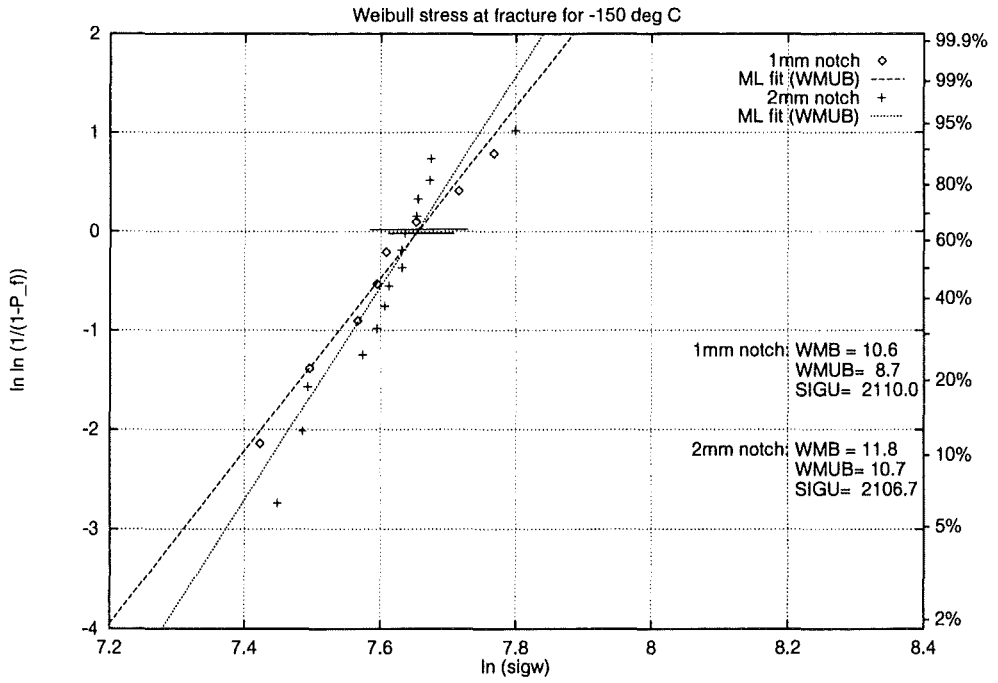

Figure 3.1: Maximum likelihood results of σ_W for two notch geometries at $-150°$ C (2mm notch results correspond to data shown in Table 3.6)

References

[1] ABAQUS/Standard User's Manual (Version 5.4), Hibbit, Karlsson & Sorensen, Inc. 1994.

[2] F. Mudry, A local approach to cleavage fracture, Nucl. Engng. and Des. **105** (1987), 65-76.

[3] F.M. Beremin, A local criterion for cleavage fracture of a nuclear pressure vessel steel, Met. Trans. **14A** (1983), 2277-2287.

[4] K.-J. Bathe, Finite Element Procedures in Engineering Analysis, Prentice-Hall, Englewood Cliffs, NJ 1982.

[5] E.J. Gumbel, Statistics of Extremes, Columbia University Press, 1958.

[6] D.R. Thoman, L.J. Bain, C.E. Antle, Inferences on the Parameters of the Weibull Distribution, Technometrics, **11** (1969), 445-460.

[7] A. Brückner-Foit, G.P. Nikishkov, D. Munz, Prediction of Cleavage Probability Using Higher Order Terms of the Crack Tip Field, Journal de Physique IV **6** (1996), C6-307-314.

[8] S. Yang, Y.J. Chao, M.A. Sutton, Higher order asymptotic crack tip fields in a power-law hardening material, Engng. Fract. Mech. **45** (1993), 1-20.

[9] J.W. Hutchinson, Singular behaviour at the end of a tensile crack in a hardening material, J. Mech. Phys. Solids **16** (1968), 13-31.

[10] J.R. Rice, G.F. Rosengren, Plane strain deformation near a crack tip in a power-law hardening material, J. Mech. Phys. Solids **16** (1968), 1-12.

[11] H. Riesch-Oppermann, E. Diegele, A. Brückner-Foit, Application of local approach to cleavage fracture behaviour of the F82Hmod ferritic martensitic structural steel, (subm. to ECF 12 Sheffield 1998).

[12] D.M. Parks, Interpretation of irradiation effects on the fracture toughness of a pressure vessel steel in terms of crack tip stress analysis, J. Engng. Mat. Technol. **98** (1976), 30-36.

[13] M. Al Mundheri, P. Soulat, A. Pineau, Irradiation embrittlement of a low alloy steel interpreted in terms of a local approach of cleavage fracture, Fatigue Fract. Engng Mater. Struct. **12** (1989), 19-30.

[14] A. Brückner-Foit, T. Fett, H. Riesch-Oppermann, M. Sckuhr, Modification of the WEIBULL theory for strongly inhomogeneous stress fields, ESIS TC1/TC8 meeting, Geesthacht, 15./16.4.1997.

[15] M. Sckuhr, Spannungssingularitäten und deren Bewertung in mechanisch beanspruchten Zweistoffverbunden unter Berücksichtigung der Plastizität, PhD Thesis, University of Karlsruhe, 1997.

Appendix A

Description of the subroutines

A.1 Module `fil_ou`

A.1.1 Program `XTRA`

Description

This is the main program of the module `fil_ou`. Its purpose is to generate the `*.wst` communication file from ABAQUS `*.fil` binary results files. The structure of the `*.wst`-file is given in Table A.1.

`XTRA` reads the base name of the ABAQUS `*.fil` binary results file (which is used as the base name of the `*.wst` communication file) from standard input together with a reference stress value that is used for the calculation of the plasticity flag during the analysis.

`XTRA` then scans each record of the ABAQUS `*.fil` binary results file and writes the ABAQUS record identification key to the standard output. The records are either analysed and kept in suitable variables, or they are skipped and a warning message is issued if the identification key is not contained in the predefined key list. (Details of the scanning procedure depend on the ABAQUS data management schemes which can be looked up in the ABAQUS manuals [1] and are not repeated here.)

The basic data of the FE model are also written into the communication file. This includes the size and type of the model, data for nodes, elements, and meshing. Element-based stress results are analysed in detail. For each load case, an array `S1ENV` is generated, which contains the envelope of the values of the maximum principal stress at all integration points in every element of the model. The plasticity flag array `IPFLAG` is written into the file together with the array `S1ENV`. Array `IPFLAG` indicates whether the plasticity criterion is met at the Gauss point currently analysed and will be subsequently used in the numerical integration of the Weibull stress. At the moment, a stress-based criterion (von Mises) is used, but the analysis of plastic strain values is equally possible.

For every loadcase, one record per element is generated in the `*.wst` file. It contains the first principal stress values at integration points `1, ... IPE`, followed by the plasticity flag values at integration points `1, ... IPE`. The length of the record thus differs for 2D and 3D analyses.

Before writing the record, a check is performed to ensure that the values of the maximum principal stress do not decrease at the current integration point. If so, the value of the previous load step is retained, thus giving a stress envelope which contains non-decreasing values for subsequent load steps.

No. of lines	Content	FORMAT
NEL+NNODE+3	Prologue	
1	no. of elements / nodes / sigref /comments	'I8,I8,2X,G10.4,2A'
1	ABAQUS version / date / time	'A'
NEL	element nodes / element type	(variable)
NNODE	nodal coordinates	'I5,3(2X,G14.7)'
1	heading	'A'
NEL+1	body - repeated for each load step	
1	' START OF NEW INCREMENT'	'A'
NEL	envelope of max. principal stress + plasticity flag	(variable)

Table A.1: Structure of the `*.wst` communication file.

Some technical details

To run XTRA, it is necessary to generate a load module `xtra.x` using the ABAQUS statement: `abaqus make job=xtra`. `xtra.x` can then be invoked by `xtra.x < fname`, where `fname` is the name of the above-mentioned auxiliary file in the current directory.

Parameters In:

`NPRECD` precision flag set by ABAQUS PARAMETER statement (current value: `NPRECD=1`, i.e. single precision)

`MAXEL` maximum number of elements (set by: `PARAMETER(MAXEL=1000)`)

`MAXNOD` maximum number of nodes (set by: `PARAMETER(MAXNOD=8000)`)

`MAXNPE` maximum number of nodes per element (set by: `PARAMETER(MAXNPE=20)`)

`MAXIPE` maximum number of integration points per element (set by: `PARAMETER(MAXIPE=8)`

Parameters Out:

none

External Subroutines:

`POST` `POST(FNAME)` initializes FORTRAN unit number (set by call of ABAQUS subroutine `DBRNU(JUNIT)`) and defines some additional parameters (set by call of ABAQUS subroutine `INITPF(FNAME, NRU, LRUNIT,LOUTPF)`)

`DBFILE` internal ABAQUS routine (for `LOP=0`, `DBFILE(LOP, ARRAY, JRCD)` reads the next record of ABAQUS `*.fil` file into array `ARRAY` and sets EOF file marker `JRCD`)

External Functions:

none

Local Variables:

`KEY` record identifier key

`LOP` required by subroutine `DBFILE`. `LOP=0` is used only.

JRCD EOF file marker

NW number of words per ABAQUS record

JRRAY auxiliary array for ABAQUS record

ARRAY auxiliary array for ABAQUS record (JRRAY(NPRECD,513) and ARRAY(513) are connected via a FORTRAN EQUIVALENCE statement)

FNAME base name of *.wst file

I,J,K,N loop counters

CHAR auxiliary character variable

CHAR1 auxiliary character variable

HEADG heading of ABAQUS results file

TMPCHA ABAQUS version, date, time, ...

CFLAG(0:2) output location identification

SNAME set name (node or element set), blank if unspecified

ELTYPE element type (only for element output)

ACYFLG actively yielding flag (ABAQUS)

FC,SC,TC first, second, and third nodal coordinate

PE1,...,PE6 plastic strain components

PEEQ equivalent plastic strain

PEMAG plastic strain magnitude

S1,S2,S3 1st, 2nd, and 3rd principal stress

SIGEQ von Mises stress

SIGO reference flow stress value

TLEN typical element length of FE mesh (ABAQUS)

STRESS(8) components of stress tensor (incl. stress invariants) (temporary array for output of nodal or averaged stresses to stdio)

STRESI(MAXEL,MAXIPE,6) components of stress tensor at integration points (written for each increment to auxiliary file 'stresses' at current directory)

S1ENV array of envelope of maximum principal stress

IPFLAG plasticity flag array

JEL no. of element currently analysed

JPNT no. of node/integration point currently analysed (0 if centroidal values or nodal averaged values are given)

JSPNT 0 for continuum elements

JLOC location identifier for elemental output

NDI no. of direct stress components

NSHR no. of shear stress components

NDIR not used

NSFC not used

JFLAG location identifier upon start of output request (0 - element based output)

IEL loop counter for elements

NEL counter for actual number of elements in current increment

II loop counter for integration points

IKOMP loop counter for stress components

IINT number of integration points

INODE current node number for nodal coordinates record

KEL current element number

KNOD current node number for node definitions record

I1900, I1901 loop counter

NODE contiguity array of nodes for each element

COORD array of initial nodal coordinates

NELEMS no of elements in the model

NNODES no of nodes in the model

IDIM dimension of the model

IWUN output unit for communication file *.wst

A.1.2 Subroutine POST

Description

Subroutine POST initializes the FORTRAN unit number (set by call of ABAQUS subroutine DBRNU (JUNIT) and defines some additional parameters (set by call of ABAQUS subroutine INITPF(FNAME, NRU, LRUNIT, LOUTPF)).

Parameters In:

FNAME name of ABAQUS *.fil file

Parameters Out:

none

External Subroutines:

INITPF(FNAME,NRU,LRUNIT,LOUTF) (see above)

DBRNU(JUNIT) sets FORTRAN unit number for file FNAME

External Functions:

none

Local Variables:

NRU number of result files to be read (NRU=1 is used)

LRUNIT(2,NRU) FORTRAN unit numbers and flag indicating binary or ASCII format of ABAQUS results file

LOUTF flag for output file format (not used)

JUNIT FORTRAN unit number for file FNAME required by ABAQUS subroutine DBRNU

A.2 Modules gau_wei, wei_ml

A.2.1 Program WSTRIT

Description

This is the main program of module gau_wei. Its purpose is to calculate the Weibull stress from the ABAQUS output (2 or 3D analysis) via the *.wst communication file and perform iterative ML estimation including confidence intervals. This is done in several steps:

1. parameter input
2. calculation of shape function derivatives at integration points
3. reading of the stress envelope results from the *.wst communication file
4. numerical integration of Weibull stress at fracture
5. output of Weibull stresses for complete sample
6. maximum likelihood procedure
7. repeat steps 3 to 6, if convergence not achieved
8. calculate confidence limits for maximum likelihood estimators
9. generate results file including plotdata file

The subroutines which are invoked are described below.

The results file is generated in a format that can be used as a source to generate a Weibull plot. At the moment, gnuplot format is preferred.

Parameters In:

none

Parameters Out:

none

External Subroutines:

PSENV	reads initial values from standard input at first call; read communication file FNAME at subsequent calls
DHDRST	calculates derivative of shape functions at integration points for 3-D problems
DHIDRS	calculates derivative of shape functions at integration points for 2-D problems
INT2WS	calculates Weibull stress for 2-D problems
INT3WS	calculates Weibull stress for 3-D problems
MAXL	performs maximum likelihood estimation of Weibull stress parameters
OUTPUT	generates printout of results for each load step
CNFLIM	calculates confidence intervals for Weibull stress distribution parameters
MLPLOT	generates plotdata file

External Functions:

BIAS calculates unbiasing factor for $\hat{m}$

Local Variables:

ITW counts no. of iterations for ML procedure

WTOL convergence criterion (currently set to 0.1)

LST load step number

A.2.2 Subroutine PSENV

Description

Reads FE data and principal stress envelope array from *.wst-file.

Parameters In:

LST indicator for load step (LST=0 means: read prologue of *.wst-file; LST > 0 means: read data for load step LST)

ITW counter for ML iteration; if ITW=0 and LST=0, also initial control data is read from standard input

FNAME name of *.wst-file to be analysed (read from stdin)

SIGO reference stress for plasticity flag (read from stdin)

WMO initial value for Weibull modulus (read from stdin)

IWUN FORTRAN unit of input file FNAME (read from stdin)

NDIM dimension of FE model in file FNAME (read from stdin)

NPE number of nodes per element of FE model in file FNAME (read from stdin)

IPE number of integration points per element of FE model in file FNAME (read from stdin)

LSTMAX number of load steps in file FNAME (read from stdin)

NELEMS number of elements (read from *.wst-file)

NNODES number of nodes (read from *.wst-file)

SIGREF reference stress (read from *.wst-file and checked against SIGO for consistency)

NODES(*) node list of FE mesh (read from *.wst-file)

COOR(*) nodal coordinate list of FE mesh (read from *.wst-file)

S1ENV(*) array of principal stress envelope for load step LST (read from *.wst-file)

IPFLAG(*) plasticity flag array for load step LST (read from *.wst-file)

Parameters Out:

NODES(*) node list of FE mesh

COOR(*) nodal coordinate list of FE mesh

S1ENV(*) array of principal stress envelope

IPFLAG(*) plasticity flag array

External Subroutines:

none

External Functions:

none

Local Variables:

FORNOD	auxiliary character variable for FORTRAN FORMAT statement
FORS1E	auxiliary character variable for FORTRAN FORMAT statement
FORCHA	auxiliary character variable for FORTRAN FORMAT statement
KEL	dummy variable
ELTYPE	dummy variable
I, J	loop counter
IEL	loop counter
II	loop counter
INOD	loop counter

A.2.3 Subroutine DHDRST

Description

This subroutine calculates derivatives of shape functions with respect to the natural coordinates r, s, t at the integration points of the 3-D quadratic element with 20 nodes and reduced integration. Node numbering is according to ABAQUS convention. The calculation scheme is as follows (taken from Bathe [4], p. 201): First, the derivatives of the quadratic interpolation functions at the midside nodes, i.e. nodes 20, 19, ... to 9 are calculated for each of the three directions r, s, t and at each integration point IP. Then, the derivatives of the linear interpolation functions at the corner nodes i.e. nodes 8, 7, ... to 1 are determined and corrected for the quadratic terms from the adjacent nodes. The complete scheme for direction r is as follows, where r, s, t are to be taken at the coordinates of integration point number i (= IP):

$$
\begin{aligned}
\frac{\partial h_i(20)}{\partial r} &= -\frac{1}{4} * (1+s) * (1-t^2) \\
\frac{\partial h_i(19)}{\partial r} &= \frac{1}{4} * (1+s) * (1-t^2) \\
\frac{\partial h_i(18)}{\partial r} &= \frac{1}{4} * (1-s) * (1-t^2) \\
\frac{\partial h_i(17)}{\partial r} &= -\frac{1}{4} * (1-s) * (1-t^2) \\
\frac{\partial h_i(16)}{\partial r} &= -\frac{1}{4} * (1-s^2) * (1+t) \\
\frac{\partial h_i(15)}{\partial r} &= -\frac{1}{2} * r * (1+s) * (1+t) \\
\frac{\partial h_i(14)}{\partial r} &= \frac{1}{4} * (1-s^2) * (1+t) \\
\frac{\partial h_i(13)}{\partial r} &= -\frac{1}{2} * r * (1-s) * (1+t) \\
\frac{\partial h_i(12)}{\partial r} &= -\frac{1}{4} * (1-s^2) * (1-t)
\end{aligned}
$$

$$
\begin{aligned}
\frac{\partial h_i(11)}{\partial r} &= -\frac{1}{2} * r * (1+s) * (1-t) \\
\frac{\partial h_i(10)}{\partial r} &= \frac{1}{4} * (1-s^2) * (1-t) \\
\frac{\partial h_i(9)}{\partial r} &= -\frac{1}{2} * r * (1+s) * (1-t) \\
\frac{\partial h_i(8)}{\partial r} &= -\frac{1}{2} * \left(-\frac{1}{4} * (1+s) * (1+t) - \frac{\partial h_i(15)}{\partial r} - \frac{\partial h_i(16)}{\partial r} - \frac{\partial h_i(20)}{\partial r}\right) \\
\frac{\partial h_i(7)}{\partial r} &= -\frac{1}{2} * \left(\frac{1}{4} * (1+s) * (1+t) - \frac{\partial h_i(14)}{\partial r} - \frac{\partial h_i(15)}{\partial r} - \frac{\partial h_i(19)}{\partial r}\right) \\
\frac{\partial h_i(6)}{\partial r} &= -\frac{1}{2} * \left(\frac{1}{4} * (1-s) * (1+t) - \frac{\partial h_i(13)}{\partial r} - \frac{\partial h_i(14)}{\partial r} - \frac{\partial h_i(18)}{\partial r}\right) \\
\frac{\partial h_i(5)}{\partial r} &= -\frac{1}{2} * \left(-\frac{1}{4} * (1-s) * (1+t) - \frac{\partial h_i(16)}{\partial r} - \frac{\partial h_i(13)}{\partial r} - \frac{\partial h_i(17)}{\partial r}\right) \\
\frac{\partial h_i(4)}{\partial r} &= -\frac{1}{2} * \left(-\frac{1}{4} * (1+s) * (1-t) - \frac{\partial h_i(11)}{\partial r} - \frac{\partial h_i(12)}{\partial r} - \frac{\partial h_i(20)}{\partial r}\right) \\
\frac{\partial h_i(3)}{\partial r} &= -\frac{1}{2} * \left(\frac{1}{4} * (1+s) * (1-t) - \frac{\partial h_i(10)}{\partial r} - \frac{\partial h_i(11)}{\partial r} - \frac{\partial h_i(19)}{\partial r}\right) \\
\frac{\partial h_i(2)}{\partial r} &= -\frac{1}{2} * \left(\frac{1}{4} * (1-s) * (1-t) - \frac{\partial h_i(9)}{\partial r} - \frac{\partial h_i(10)}{\partial r} - \frac{\partial h_i(18)}{\partial r}\right) \\
\frac{\partial h_i(1)}{\partial r} &= -\frac{1}{2} * \left(-\frac{1}{4} * (1-s) * (1-t) - \frac{\partial h_i(12)}{\partial r} - \frac{\partial h_i(9)}{\partial r} - \frac{\partial h_i(17)}{\partial r}\right)
\end{aligned}
\tag{A.1}
$$

Results for directions s, t are determined accordingly, but omitted here for brevity. These results are used in subroutine `DJACD` which calculates the Jacobi matrix. As the derivatives are given in the reference configuration, it is only necessary to determine them at the beginning of the iterative procedure once and for all.

Parameters In:

none

Parameters Out:

`DHIDR(IP,*)` derivatives of shape function HI with respect to R for integration point No. IP at nodes 1...20

`DHIDS(IP,*)` derivatives of shape function HI with respect to S for integration point No. IP at nodes 1...20

`DHIDT(IP,*)` derivatives of shape function HI with respect to T for integration point No. IP at nodes 1...20

`HI(*)` shape function HI at nodes 1...20 (not used in the sequel)

External Subroutines:

none

External Functions:

none

Local Variables:

IP auxiliary variable (counter for integration points)

L auxiliary variable (loop counter)

R, R2 auxiliary variables (coordinates r, r^2)

S, S2 auxiliary variables (coordinates s, s^2)

T, T2 auxiliary variables (coordinates t, t^2)

UPR, UMR, UMR2 auxiliary variables ($1+r$, $1-r$, $1-r^2$)

UPS, UMS, UMS2 auxiliary variables ($1+s$, $1-s$, $1-s^2$)

UPT, UMT, UMT2 auxiliary variables ($1+t$, $1-t$, $1-t^2$)

WU3 auxiliary variable ($\sqrt{3}$)

SUM auxiliary variable (consistency check)

SUMR auxiliary variable (consistency check)

SUMS auxiliary variable (consistency check)

SUMT auxiliary variable (consistency check)

A.2.4 Subroutine DHIDRS

Description

This subroutine calculates derivatives of shape functions with respect to the natural coordinates r, s at the integration points of 2-D quadratic element with 8 nodes and reduced integration. Node numbering is according to ABAQUS convention. The calculation scheme is as follows (taken from Bathe [4], p. 200): First, the derivatives of the quadratic interpolation functions at the midside nodes, i.e. nodes 8, 7, 6, 5 are calculated for each of the three directions r, s, t and at each integration point IP. Then, the derivatives of the linear interpolation functions at the corner nodes i.e. nodes 4, 3, 2, 1 are determined and corrected for the quadratic terms from the adjacent nodes. The complete scheme for direction r is as follows, where r, s, t are to be taken at the coordinates of integration point number i (= IP):

$$
\begin{aligned}
\frac{\partial h_i(8)}{\partial r} &= -\frac{1}{2} * (1-s^2) \\
\frac{\partial h_i(7)}{\partial r} &= -r * (1+s) \\
\frac{\partial h_i(6}{\partial r} &= \frac{1}{2} * (1-s^2) \\
\frac{\partial h_i(5)}{\partial r} &= -r * (1-s) \\
\frac{\partial h_i(4)}{\partial r} &= \frac{1}{2} * \left(-\frac{1}{2} * (1+s) - \frac{\partial h_i(7)}{\partial r} - \frac{\partial h_i(8)}{\partial r} \right) \\
\frac{\partial h_i(3)}{\partial r} &= \frac{1}{2} * \left(\frac{1}{2} * (1+s) - \frac{\partial h_i(6)}{\partial r} - \frac{\partial h_i(7)}{\partial r} \right) \\
\frac{\partial h_i(2)}{\partial r} &= \frac{1}{2} * \left(\frac{1}{2} * (1-s) - \frac{\partial h_i(5)}{\partial r} - \frac{\partial h_i(6)}{\partial r} \right) \\
\frac{\partial h_i(1)}{\partial r} &= \frac{1}{2} * \left(-\frac{1}{2} * (1-s) - \frac{\partial h_i(8)}{\partial r} - \frac{\partial h_i(5)}{\partial r} \right)
\end{aligned}
\tag{A.2}
$$

Results for direction s are determined accordingly. These results are used in subroutine DJACD, which calculates the Jacobi matrix. As the derivatives are given in the reference configuration, subroutine DHIDRS has to be called only once.

Parameters In:

none

Parameters Out:

`DHIDR(IP,*)` derivatives of shape function HI with respect to R for integration point no. IP at nodes 1...8

`DHIDS(IP,*)` derivatives of shape function HI with respect to S for integration point no. IP at nodes 1...8

External Subroutines:

none

External Functions:

none

Local Variables:

`IP` auxiliary variable (counter for integration points)

`I, J` auxiliary variable (loop counter)

`R, R2` auxiliary variables

`S, S2` auxiliary variables

`UPR, UMR, UMR2` auxiliary variables

`UPS, UMS, UMS2` auxiliary variables

`WU3` auxiliary variable

A.2.5 Subroutine `DJACD`

Description

This subroutine performs calculation of the Jacobi determinant at integration point IP for 2-D 8-node quadratic elements with reduced integration. Node numbering is according to ABAQUS convention.

Parameters In:

`INTEL` element no.

`IP` integration point no.

Parameters Out:

`DET` Jacobi determinant

External Subroutines:

none

External Functions:

none

Local Variables:

IQ	auxiliary variable (loop counter)
NQ	number of nodes (set to 8)
DJ11	auxiliary variable
DJ12	auxiliary variable
DJ21	auxiliary variable
DJ22	auxiliary variable
DETJ	auxiliary variable

A.2.6 Subroutine DJAC3D

Description

This subroutine performs calculation of the Jacobi determinant at integration point IP for 3-D 20-node quadratic elements with reduced integration. Node numbering is according to ABAQUS convention.

Parameters In:

INTEL	element no.
IP	integration point no.

Parameters Out:

DET	Jacobi determinant

External Subroutines:

none

External Functions:

none

Local Variables:

IQ	auxiliary variable (loop counter)
NQ	number of nodes (set to 20)
DJ11	auxiliary variable
DJ12	auxiliary variable
DJ13	auxiliary variable
DJ21	auxiliary variable
DJ22	auxiliary variable
DJ23	auxiliary variable
DJ31	auxiliary variable
DJ32	auxiliary variable
DJ33	auxiliary variable
DETJ	auxiliary variable

A.2.7 Subroutine INT2WS

Description

This subroutine calculates the Weibull stress (numerical integration over plastic zone) for 2-D problems and stores the results as SIGW(LST) in the array SIGW.

Parameters In:

LST load step number

Parameters Out:

none

External Subroutines:

DJACD calculates Jacobi determinant

External Functions:

none

Local Variables:

INTEL	auxiliary variable (loop counter)
I1DIM	auxiliary variable (loop counter)
I2DIM	auxiliary variable (loop counter)
IP	auxiliary variable
PS1	auxiliary variable
WS1	auxiliary variable
WS2	auxiliary variable
WSTREL	auxiliary variable
WSTR	auxiliary variable

A.2.8 Subroutine INT3WS

Description

This subroutine calculates the Weibull stress (numerical integration over plastic zone) for 3-D problems and stores the results as SIGW(LST) in the array SIGW.

Parameters In:

LST load step number

Parameters Out:

none

External Subroutines:

`DJAC3D` calculates Jacobi determinant

External Functions:

none

Local Variables:

`INTEL` auxiliary variable (loop counter)
`I1DIM` auxiliary variable (loop counter)
`I2DIM` auxiliary variable (loop counter)
`I3DIM` auxiliary variable (loop counter)
`IP` auxiliary variable
`PS1` auxiliary variable
`WS1` auxiliary variable
`WS2` auxiliary variable
`WS3` auxiliary variable
`WSTREL` auxiliary variable
`WSTR` auxiliary variable

A.2.9 Function `BIAS`

Description

Bias correction factor $b(N)$ for maximum likelihood estimate $\hat{m}$.

Parameters In:

`N` sample size (must be in the range of 5 to 40. Otherwise, a warning message is issued and `BIAS` is set to unity.)

Parameters Out:

none

External Subroutines:

none

External Functions:

none

Local Variables:

`B(*)` auxiliary array (contains bias correction factors $b(N)$ for $N = 5, \ldots 40$)

A.2.10 Subroutine MAXL

Description

Yields maximum likelihood estimates of Weibull stress parameters.

Parameters In:

LSTMAX	sample size (corresponds to max. no. of load steps), i.e. no. of fractured specimens analysed.

Parameters Out:

none

External Subroutines:

none

External Functions:

BIAS	bias correction
DMLF	maximum likelihood function for m

Local Variables:

I	auxiliary variable (loop counter)
ISTEP	auxiliary variable (loop counter)
DMLM	auxiliary variable
DMLO	auxiliary variable
DMLU	auxiliary variable
SUMW	auxiliary variable
WMLHM	auxiliary variable
WMLHO	auxiliary variable
WMLHU	auxiliary variable
WMTOL	tolerance for accuracy of interval sectioning method

A.2.11 Function DMLF

Description

Maximum likelihood function for m.

Parameters In:

DM	value of m
N	sample size

Parameters Out:

none

External Subroutines:

none

External Functions:

none

Local Variables:

`I`	auxiliary variable (loop counter)
`SUMW`	auxiliary variable ($\sum_1^N \sigma_W^m$)
`SUMLW`	auxiliary variable ($\sum_1^N \ln(\sigma_W)$)
`SUMWLW`	auxiliary variable ($\sum_1^N \ln(\sigma_W) * \sigma_W^m$)

A.2.12 Subroutine `CNFLIM`

Description

80, 90, and 96 percent confidence intervals of ML estimates of parameters m and σ_u are calculated and printed.

Parameters In:

`N`	sample size
`DM`	ML estimate of parameter m
`XO`	ML estimate of parameter σ_u

Parameters Out:

`WMLO`	lower bound of confidence interval for parameter m
`WMUP`	upper bound of confidence interval for parameter m
`SIGULO`	lower bound of confidence interval for parameter σ_u
`SIGUUP`	upper bound of confidence interval for parameter σ_u

External Subroutines:

none

External Functions:

none

Local Variables:

`IROW`	auxiliary variable
`CNFTAB`	auxiliary array
`L1, L2`	auxiliary variables for m intervals
`T1, T2`	auxiliary variables for σ_u intervals

Appendix B

Tables for Weibull parameter evaluation

B.1 Unbiasing factors $b(N)$ for $\hat{m}$

N	B(N)
5	0.669
6	0.752
7	0.792
8	0.820
9	0.842
10	0.859
11	0.872
12	0.883
13	0.893
14	0.901
15	0.908
16	0.914

N	B(N)
17	0.919
18	0.923
19	0.927
20	0.931
21	0.935
22	0.938
23	0.941
24	0.943
25	0.945
26	0.947
27	0.949
28	0.951

N	B(N)
29	0.953
30	0.955
31	0.957
32	0.958
33	0.959
34	0.960
35	0.961
36	0.962
37	0.963
38	0.964
39	0.965
40	0.960

Table B.1: Unbiasing factors $b(N)$

B.2 Percentage points, $t_1(N, \alpha_1)$ and $t_2(N, \alpha_2)$, for the confidence interval for σ_u at confidence level $1 - \alpha$

N	$\alpha_1 = 0.02$	$\alpha_1 = 0.05$	$\alpha_1 = 0.10$	$\alpha_2 = 0.90$	$\alpha_2 = 0.95$	$\alpha_2 = 0.98$
5	-1.631	-1.247	-0.888	0.772	1.107	1.582
6	-1.396	-1.007	-0.740	0.666	0.939	1.291
7	-1.196	-0.874	-0.652	0.598	0.829	1.120
8	-1.056	-0.784	-0.591	0.547	0.751	1.003
9	-0.954	-0.717	-0.544	0.507	0.691	0.917
10	-0.876	-0.665	-0.507	0.475	0.644	0.851
11	-0.813	-0.622	-0.477	0.448	0.605	0.797
12	-0.762	-0.587	-0.451	0.425	0.572	0.752
13	-0.719	-0.557	-0.429	0.406	0.544	0.714
14	-0.683	-0.532	-0.410	0.389	0.520	0.681
15	-0.651	-0.509	-0.393	0.374	0.499	0.653
16	-0.624	-0.489	-0.379	0.360	0.480	0.627
17	-0.599	-0.471	-0.365	0.348	0.463	0.605
18	-0.578	-0.455	-0.353	0.338	0.447	0.584
19	-0.558	-0.441	-0.342	0.328	0.433	0.566
20	-0.540	-0.428	-0.332	0.318	0.421	0.549
22	-0.509	-0.404	-0.314	0.302	0.398	0.519
24	-0.483	-0.384	-0.299	0.288	0.379	0.494
26	-0.460	-0.367	-0.286	0.276	0.362	0.472
28	-0.441	-0.352	-0.274	0.265	0.347	0.453
30	-0.423	-0.338	-0.264	0.256	0.334	0.435
32	-0.408	-0.326	-0.254	0.247	0.323	0.420
34	-0.394	-0.315	-0.246	0.239	0.312	0.406
36	-0.382	-0.305	-0.238	0.232	0.302	0.393
38	-0.370	-0.296	-0.231	0.226	0.293	0.382
40	-0.360	-0.288	-0.224	0.220	0.285	0.371
42	-0.350	-0.280	-0.218	0.214	0.278	0.361
44	-0.341	-0.273	-0.213	0.209	0.271	0.352
46	-0.333	-0.266	-0.208	0.204	0.264	0.344
48	-0.325	-0.260	-0.203	0.199	0.258	0.336
50	-0.318	-0.254	-0.198	0.195	0.253	0.328
52	-0.312	-0.249	-0.194	0.191	0.247	0.321
54	-0.305	-0.244	-0.190	0.187	0.243	0.315
56	-0.299	-0.239	-0.186	0.184	0.238	0.309
58	-0.294	-0.234	-0.183	0.181	0.233	0.303
60	-0.289	-0.230	-0.179	0.177	0.229	0.297
62	-0.284	-0.226	-0.176	0.174	0.225	0.292
64	-0.279	-0.222	-0.173	0.171	0.221	0.287
66	-0.274	-0.218	-0.170	0.169	0.218	0.282
68	-0.270	-0.215	-0.167	0.166	0.214	0.278
70	-0.266	-0.211	-0.165	0.164	0.211	0.274

Table B.2: Auxiliary variables for the confidence interval for σ_u

N	$\alpha_1 = 0.02$	$\alpha_1 = 0.05$	$\alpha_1 = 0.10$	$\alpha_2 = 0.90$	$\alpha_2 = 0.95$	$\alpha_2 = 0.98$
72	-0.262	-0.208	-0.162	0.161	0.208	0.269
74	-0.259	-0.205	-0.160	0.159	0.205	0.266
76	-0.255	-0.202	-0.158	0.157	0.202	0.262
78	-0.252	-0.199	-0.155	0.155	0.199	0.258
80	-0.248	-0.197	-0.153	0.153	0.197	0.255
85	-0.241	-0.190	-0.148	0.148	0.190	0.246
90	-0.234	-0.184	-0.144	0.143	0.185	0.239
95	-0.227	-0.179	-0.139	0.139	0.179	0.232
100	-0.221	-0.174	-0.136	0.136	0.175	0.226
110	-0.211	-0.165	-0.129	0.129	0.166	0.215
120	-0.202	-0.158	-0.123	0.123	0.159	0.205

Table B.2: Auxiliary variables for the confidence interval for σ_u (cont'd.)

B.3 Percentage points, $l_1(N, \alpha_1)$ and $l_2(N, \alpha_2)$, for the confidence interval for m at confidence level $1 - \alpha$

N	$\alpha_1 = 0.02$	$\alpha_1 = 0.05$	$\alpha_1 = 0.10$	$\alpha_2 = 0.90$	$\alpha_2 = 0.95$	$\alpha_2 = 0.98$
5	0.604	0.683	0.766	2.277	2.779	3.518
6	0.623	0.697	0.778	2.030	2.436	3.067
7	0.639	0.709	0.785	1.861	2.183	2.640
8	0.653	0.720	0.792	1.747	2.015	2.377
9	0.665	0.729	0.797	1.665	1.896	2.199
10	0.676	0.738	0.802	1.602	1.807	2.070
11	0.686	0.745	0.807	1.553	1.738	1.972
12	0.695	0.752	0.811	1.513	1.682	1.894
13	0.703	0.759	0.815	1.480	1.636	1.830
14	0.710	0.764	0.819	1.452	1.597	1.777
15	0.716	0.770	0.823	1.427	1.564	1.732
16	0.723	0.775	0.826	1.406	1.535	1.693
17	0.728	0.779	0.829	1.388	1.510	1.660
18	0.734	0.784	0.832	1.371	1.487	1.630
19	0.739	0.788	0.835	1.356	1.467	1.603
20	0.743	0.791	0.838	1.343	1.449	1.579
22	0.752	0.798	0.843	1.320	1.418	1.538
24	0.759	0.805	0.848	1.301	1.392	1.504
26	0.766	0.810	0.852	1.284	1.370	1.475
28	0.772	0.815	0.856	1.269	1.351	1.450
30	0.778	0.820	0.860	1.257	1.334	1.429
32	0.783	0.824	0.863	1.246	1.319	1.409
34	0.788	0.828	0.866	1.236	1.306	1.392
36	0.793	0.832	0.869	1.227	1.294	1.377
38	0.797	0.835	0.872	1.219	1.283	1.363
40	0.801	0.839	0.875	1.211	1.273	1.351

Table B.3: Auxiliary variables for the confidence interval for m

N	$\alpha_1 = 0.02$	$\alpha_1 = 0.05$	$\alpha_1 = 0.10$	$\alpha_2 = 0.90$	$\alpha_2 = 0.95$	$\alpha_2 = 0.98$
42	0.804	0.842	0.877	1.204	1.265	1.339
44	0.808	0.845	0.880	1.198	1.256	1.329
46	0.811	0.847	0.882	1.192	1.249	1.319
48	0.814	0.850	0.884	1.187	1.242	1.310
50	0.817	0.852	0.886	1.182	1.235	1.301
52	0.820	0.854	0.888	1.177	1.229	1.294
54	0.822	0.857	0.890	1.173	1.224	1.286
56	0.825	0.859	0.891	1.169	1.218	1.280
58	0.827	0.861	0.893	1.165	1.213	1.273
60	0.830	0.863	0.894	1.162	1.208	1.267
62	0.832	0.864	0.896	1.158	1.204	1.262
64	0.834	0.866	0.897	1.155	1.200	1.256
66	0.836	0.868	0.899	1.152	1.196	1.251
68	0.838	0.869	0.900	1.149	1.192	1.246
70	0.840	0.871	0.901	1.146	1.188	1.242
72	0.841	0.872	0.903	1.144	1.185	1.237
74	0.843	0.874	0.904	1.141	1.182	1.233
76	0.845	0.875	0.905	1.139	1.179	1.229
78	0.846	0.876	0.906	1.136	1.176	1.225
80	0.848	0.878	0.907	1.134	1.173	1.222
85	0.852	0.881	0.910	1.129	1.166	1.213
90	0.855	0.883	0.912	1.124	1.160	1.206
95	0.858	0.886	0.914	1.120	1.155	1.199
100	0.861	0.888	0.916	1.116	1.150	1.192
110	0.866	0.893	0.920	1.110	1.141	1.181
120	0.871	0.897	0.923	1.104	1.133	1.171

Table B.3: Auxiliary variables for the confidence interval for m (cont'd)