UNCECOMP 2015
1st ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete Island, Greece, 25–27 May 2015

IMPROVED GAUSSIAN APPROXIMATION OF THE BUNDLE STRENGTH OF DANIELS' MODEL WITH BRITTLE WEIBULL FIBERS

Václav Sadílek[1] **and Miroslav Vořechovský**[1]

[1]Brno University of Technology, Faculty of Civil Engineering, Institute of Structural Mechanics
Veveří 95, 602 00 Brno, Czech Republic
e-mail: {sadilek.v,vorechovsky.m}@fce.vutbr.cz

Keywords: Fiber bundle model, strength of the bundle, Daniels bundle model, equal load sharing, Weibull distribution, cumulative distribution function.

Abstract. *The paper deals with the classical fiber bundle model with equal load sharing, sometimes referred to as the the Daniels bundle model or the democratic bundle. This model is significant for the strength of quasi-brittle structures and the reliability of many parallel systems.*

In the present paper, the authors exploit their own implementation of the recursive formula for the evaluation of the distribution function for fiber bundle strength. The implementation was carried out in the Python high-level programming language using the NumPy (scientific computing with arrays) and mpmath (library for real and complex floating-point arithmetic with arbitrary precision) packages. This implementation enables the calculation of cumulative distribution function values for large numbers (thousands) of fibers in a bundle, including values deep in the left tail of the distribution (probabilities 10^{-600}*). This computer program has been used to accurately calculate the distribution functions for bundles with Weibull fibers with the unit scale parameter, the varying number of filaments and the varying shape parameter. A database of these distribution functions has been used to calculate the mean values and standard deviations of the peak force to propose an improved Gaussian approximation of bundle strength.*

1 INTRODUCTION

This paper deals with the classical fiber bundle model with equal load sharing, sometimes referred to as Daniels bundle model [1] or the democratic bundle [4]. This model is significant for the strength of fibrous materials and composites, and the generally random strength of quasi-brittle structures. The model is also relevant for the analysis of the reliability of various parallel systems (computer components, infrastructure etc.). Daniels [1] formulated a multidimensional integral and also recursive formula for the evaluation of the strength distribution function in which strength is defined as the peak force per fiber. In the same paper, he showed that the distribution of the strength of the bundle, $G_n(x)$, tends to Gaussian distribution under quite broad conditions and he gave closed formulas for the mean value and standard deviation of the Gaussian distribution. Sornette [4] later confirmed this result using a Kolmogorov theorem. The convergence of a random strength to Gaussian distribution is very slow in terms of the number of fibers and therefore Smith [3] proposed a corrected term for the mean value that improves the original Daniels formula for small bundles. Even though the knowledge of the asymptotic form of G_n for the number of fibers $n \to \infty$ is important, the normality does not hold in the tails of the distribution and it also does not hold when there is a small number of parallel components in the system (fibers). The real cumulative distribution function (CDF), G_n, strongly deviates from the normal distribution for values of x far from the mean strength. Only a little is known about the behavior of the tails. The left tail ($x \to 0$) is of great importance for reliability considerations, however.

In the present paper, the authors describe an analysis of the recursive formula by Daniels and an improved [3] Gaussian approximation of the strength. The advantage of the recursive formula is that it provides exact values of the CDF G_n for arbitrary values of x and for any number of fibers n. The authors describe a computer implementation carried out in the Python high-level programming language [5] using the NumPy [7] (scientific computing with arrays) and the mpmath [6] (library for real and complex floating-point arithmetic with arbitrary precision) packages. This implementation enables the calculation of CDF values for large numbers (thousands) of fibers in a bundle including values deep in the left tail of the distribution (probabilities 10^{-600}). This computer program is used to accurately calculate the distribution functions G_n for bundles with Weibull fibers with the scale parameter $s = 1$, the varying number of fibers n and the varying shape parameter m. The obtained results are stored in a newly created database and compared to the available formulas [1, 2, 3]. We have found that for small bundles (n between 2 and 100) the formulas are not accurate. Therefore, the authors propose an improved closed-form for the standard deviation.

The main motivation for this work is to formulate an improved analytical formula for the distribution function G_n that will be valid deep within the left tail, where the real distribution strongly deviates from the Gaussian approximation. The present paper focuses on the second moment of the bundle strength.

2 BUNDLE STRENGTH

The cumulative distribution function, G_n, of bundle strength formulated by Daniels [1] reads:

$$G_n(x) = \sum_{k=1}^{n} (-1)^{k+1} \binom{n}{k} [F(x)]^k G_{n-k}\left(\frac{n\,x}{n-k}\right) \tag{1}$$

where x is a value of random strength per fiber, X, for which the probability $P(X \leq x) = G_n(x)$ is evaluated, $G_1(x) \equiv F(x)$, $G_0(x) \equiv 1$, $\binom{n}{k}$ is the standard Binomial coefficient and

n is the number of fibers. The most frequently selected distribution function for a single fiber, $F(x)$, is Weibull distribution. With no loss of generality, let us assume the bundles contain independent Weibullian fibers (elements in parallel) so that the CDF of the strength of any single fiber is

$$F(x) = 1 - e^{-(x/s)^m} \tag{2}$$

where s is the scale parameter and m is the shape parameter.

2.1 ALGORITHM AND IMPLEMENTATION

The recursive definition in Eq. (1) can be translated directly into Python using a recursive function. However, this straightforward implementation passes through all levels of the recursive formula and requires a large number of function calls ($2^n - 1$). The most efficient formulation of the algorithm, which is also a computer implementation that was developed, follows; it does not contain any recursive functions. Let us divide Eq. (1) into three parts

$$G_n\left(x\right) = \sum_{k=1}^{n} \underbrace{(-1)^{k+1} \binom{n}{k}}_{B_{i,k}} \underbrace{[F(x)]^k}_{F_i} \underbrace{G_{n-k}\left(x \frac{n}{n-k}\right)}_{S_i}. \tag{3}$$

At the highest level of recursion, formula (3) represents a summation over $k = 1, \ldots, n$. Each of these addends calls for a recursion – an evaluation of the recursion function $G_n(x)$ with a new parameter n renamed here as $n_k = n - k$. By analyzing all arguments of random strength, x, one can see that there are only n different values for which the distribution function of strength is evaluated, namely $x_i = x\frac{n}{i}$, $i = 1, \ldots, n$. The values of x are stored in an array named x_arr.

For each element of vector $\boldsymbol{x}$, one must compute the distribution function of the strength of one fiber, $F(x)$. Let us now define a vector, $\boldsymbol{F}$, that contains the values of the basic CDF evaluated at points x_i:

$$\boldsymbol{F}: \quad F_i = F\left(x_i\right) = F\left(x\frac{n}{i}\right), \quad i = 1, \ldots, n. \tag{4}$$

This vector is precalculated and cached in computer memory as an array named cdf_arr at the beginning of computation. In later stages of computation, the elements of this vector are raised to integer powers $k = 1, \ldots, n$.

The next ingredient is a lower triangular matrix $\boldsymbol{B}$, with n rows and n columns, pre-filled with binomial coefficients multiplied by the alternating sign. Each element of the triangular matrix initially reads

$$\boldsymbol{B}: \quad B_{i,k} = (-1)^{k+1}\binom{n}{k}, \quad i = 1, \ldots, n \text{ and } k = 1, \ldots, i. \tag{5}$$

This matrix is stored in an array named gn_arr.

Once these two ingredients are calculated, the algorithm continues with the following two loops (the algorithm uses in-place operations and updates the values of the $\boldsymbol{B}$ matrix):

1. Loop over n columns of the $\boldsymbol{B}$ matrix – Starting with the first column $k = 1$, each column $k = 1, \ldots, n$ is multiplied by the kth power of elements F_i: $B_{i,k} = B_{i,k}F_i^k$ for $k = 1, \ldots, n$ and $i = k, .., n$.

$$
\begin{array}{l}
S_1 = \sum \\
S_2 = \sum \\
S_3 = \sum \\
G_4 = \sum
\end{array}
\left[
\begin{array}{llll}
B_{1,1}F_1^1 & & & \\
B_{2,1}F_2^1 S_1 & B_{2,2}F_2^2 & & \\
B_{3,1}F_3^1 S_2 & B_{3,2}F_3^2 S_1 & B_{3,3}F_3^3 & \\
B_{4,1}F_4^1 S_3 & B_{4,2}F_4^2 S_2 & B_{4,3}F_4^3 S_1 & B_{4,4}F_4^4
\end{array}
\right]
$$

Figure 1: Diagram of the cached array for the number of fibers $n = 4$. The sum of the last line returns $G_4(x)$. (red: row indexes i, blue: column indexes k, sums of rows S_i as diagonal members)

2. Loop over $n - 1$ rows of the $\boldsymbol{B}$ matrix – Starting with the row $i = 2$, run a cycle over rows $i = 2 \ldots, n$ that (i) sums the elements in the preceding row: $S_{i-1} = \sum_{k=1}^{i-1} B_{i-1,k}$ and (ii) use this value to update (multiply) all elements of sub-diagonal number i within an inner cycle over columns $j = 1, \ldots, n - i + 1$: $B_{i+j-1,j} = S_{i-1}B_{i+j-1,j}$.

After these two cycles are finished, the sum of n elements in the last row is the desired value of $G_n(x)$:

$$G_n(x) = S_n = \sum_{k=1}^{n} B_{n,k} \tag{6}$$

Fig. 1 shows a diagram of the triangular array for the number of fibers $n = 4$; factors featured in Eq. 3 are highlighted.

We note that the sums S_i of any row correspond to the strength distribution function of fiber bundle with i fibers evaluated at load $x\frac{n}{i}$:

$$S_i = G_i(x\frac{n}{i}) = \sum_{k=1}^{i} B_{i,k} \quad i = 1, \ldots, n. \tag{7}$$

The described algorithm was used to create a database for various numbers of filaments n and shape parameters m. Figure 2 shows the strengths of a bundle for shape parameter $m = 6$ and various numbers of fibers n. To plot the CDF of G_n we used a Weibull plot (x-axis is $\ln(x)$, y-axis is $\ln(-\ln(1 - G_n(x)))$) for better insight regarding the left tail of the distribution. Weibull distribution is displayed in this plot as a straight line (see the CDF of $n = 1$ in Fig. 2). The database serves as data support to formulate an improved approximation of G_n. In the following section it was used to formulate an improved approximation of the standard deviation of bundle strength.

To implement the described algorithm we have to import several required packages and set the number precision:

```
import numpy as np
import mpmath as mp

# set number of decimal places for multiprecision numbers
mp.mp.dps = 1000

# pre-conversion of frequently used values
MPF_ZERO = mp.mpf('0')
MPF_ONE = mp.mpf('1')
MPF_TWO = mp.mpf('2')
MPF_THREE = mp.mpf('3')
```

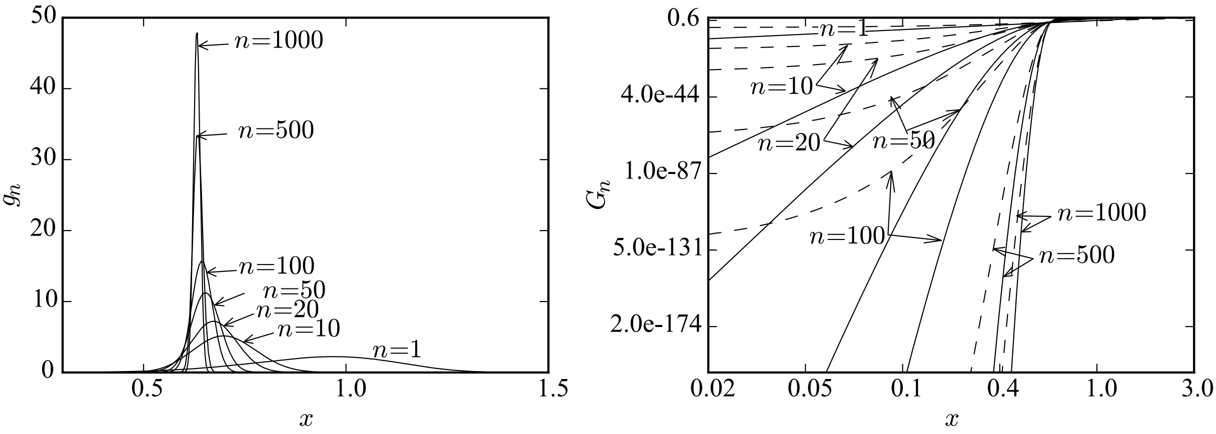

Figure 2: Left: probability density function g_n calculated as a derivative of G_n (Weibull shape parameter $m = 6$). Right: Weibull plot of cumulative distribution function G_n (solid line), and the Gaussian approximation by Smith [3] (dashed line).

MPF_ZERO, MPF_ONE,... are pre-converted values because repeated type conversions from floats, strings and integers are expensive. The precision used to evaluate $G_n(x)$ up to $n = 1500$ was set to 1000 decimal places (3325 bits). This value was found to be sufficient while considering demands on the execution time and requested accuracy. No tests of optimal precision have been performed yet.

The Binomial coefficients $\binom{n}{k}$ are calculated including the sign $(-1)^{k+1}$ and stored in the lower triangle of a 2D array named binom_tab.

```
def get_binom_tab(n):
    binom_tab = np.zeros((n, n), dtype=object)
    for i in range(1, n + 1):
        for j in range(1, i + 1):
            binom_tab[i - 1, j - 1] = (mp.binomial(i, j) * (-1) ** (j + 1))
    return binom_tab
```

This array can be precalculated for greater n and stored in a file on a hard-disk. It is useful in the case that repeated calculations are necessary.

The following source code implements the described algorithm using the numpy and mpmath packages.

```
def gn_mp(x, scale, shape, n, binom_tab):
    ''' Return value of CDF of a bundle strength considering Weibullian fibers.

    ## Parameters ##
    x, scale, shape, n : mp.mpf
        Bundle strength, scale and shape of Weibull distrib., number of filament
    binom_tab: array of mp.mpf
        numpy array of Binomial coefficients

    ## Returns ##
    out : mp.mpf
        CDF value for the strength x

    ## Examples ##
    >>> mp.mp.pretty = True
    >>> mp.mp.dps = 30
```

```
>>> x = mp.exp(mp.mpf('-1'))
>>> shape = mp.mpf('6.'); scale = mp.mpf('1.'), n_fil = mp.mpf('10')
>>> binom_tab = get_binom_tab(n_fil)
>>> gn_mp(x, scale, shape, n_fil , binom_tab)
0.000000329859130502740500994574682994
'''
ni = int(n) # retype n from mp.mpf to int
cdf_arr = np.zeros(ni, dtype=object)
x_arr = np.zeros(ni, dtype=object)
# precalculate x and F(x) vectors
for i in range(0, ni):
    x_i = mp.fraction(n, n - i) * x
    x_arr[ni - i - 1] = x_i
    # pre-calculate Weibull distr. - Eq. (2)
    cdf_arr[ni - i - 1] = MPF_ONE - mp.exp(-(x_i / scale) ** shape)
# prepare B matrix
gn_arr = binom_tab[:ni, :ni].copy()
for i in range(ni): # loop 1
    gn_arr[i:ni, i] *= cdf_arr[i:ni] ** (i + 1)

for k in xrange(1, ni): # loop 2
    idx1 = np.arange(0, ni - k)
    idx2 = np.arange(k, ni)
    idx3 = np.arange(0, k)
    gn_arr[idx2, idx1] *= np.sum(gn_arr[k - 1, idx3])
gn_m = np.sum(gn_arr[-1, :])

return gn_m
```

We note that the above-described algorithm enables fast computation of the exact CDF of bundle strength for arbitrary probabilities. The importance of result is not only for bundle models of statistical strength of structures but also for reliability considerations of many engineering systems. Moreover, many systems and also strength of heterogeneous structures and composites can be modeled by the chain-of-bundles model. The strength if such systems is dictated by the weakest bundle in the chain and therefore, a correct representation of the left tail of the bundle CDF is extremely important.

3 IMPROVED APPROXIMATION OF THE STANDARD DEVIATION

The response of a single fiber with known breaking strength ξ can be written as

$$eH(\xi - e)$$

where e is the strain of the fiber and he symbol $H(x)$ represents the Heaviside function yielding zero for $x < 0$ and one for $x \geq 0$. We now consider that the number of parallel fibers in a bundle is very large and therefore we can calculate the average response of a given bundle as a function of the strain e as an average value of the above linear-brittle response. The breaking strain ξ is considered to be a random variable described by its probability density function (PDF) $f_\xi(e)$ (and CDF denoted as $F_\xi(e)$). By definition of the mean value we can write that the average response of a fiber represents the mean response of a bundle with $n \to \infty$ [8]:

$$\mu(e) = \int_0^\infty eH(\xi - e)\, \mathrm{d}F_\xi(\xi) = e \int_{\xi - e}^\infty f_\xi(\xi)\, \mathrm{d}\xi = e[1 - F_\xi(e)] \qquad (8)$$

The asymptotic strength is the peak value of this asymptotic response. The strength is a random variable. Its asymptotic mean value is obtained at a stationary point $e^\star$ maximizing the function in Eq. (8), i.e. $E[X] = e^\star[1 - F_\xi(^\star)]$. The asymptotic standard deviation of the strength can be calculated from the variance: $D[X] = n \cdot (e^\star)^2 \cdot F_\xi(e^\star) \cdot [1 - F_\xi(e^\star)]$. The asymptotic mean strength does not depend on n, while the standard deviation decreases in inverse proportion to $\sqrt{n}$. Daniels derived this result in [1] and Smith later [3] refined the expression for the mean value in such a manner that the mean value depends on n. Figure 3 presents the derived mean values and standard deviations of bundle strength and compares them to our highly accurate solution obtained by using the calculated solutions for the bundle strength cumulative density G_n.

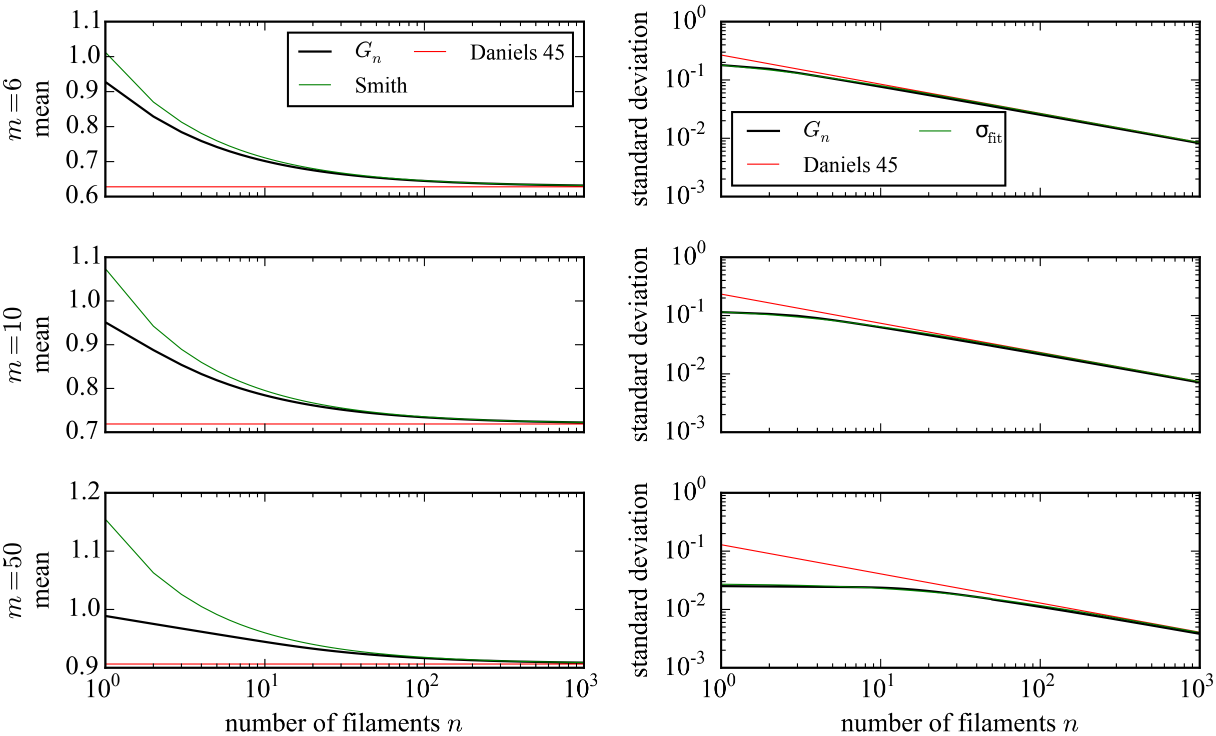

Figure 3: Plot of the mean value (left) and standard deviation (right) of G_n as a function of the number of fibers n for various shape parameters m containing approximations proposed by Smith [3] and Daniels [1].

We will now focus on the standard deviation of random strength X. Daniels result [1] for the standard deviation of strength with Weibull fibers reads (consider the stationary point $e^\star = s \cdot m^{-1/m}$):

$$\sigma_{\mathrm{D}}(n) = \sqrt{D[X]} = \frac{m^{-\frac{1}{m}} s \sqrt{e^{-\frac{1}{m}}(1 - e^{-\frac{1}{m}})}}{\sqrt{n}} \tag{9}$$

This formula is a power law, the graph of which, in logarithmic coordinates, is a straight line of slope −½. Our numerical results, however, suggest that for a low number of fibers, n, the standard deviation is lower then $\sigma_{\mathrm{D}}(n)$. This is especially true for high values of the shape parameters, m, for which the standard deviation for $n \to 0$ tends to a horizontal asymptote. Therefore, we suggest a modified formula for the standard deviation that has horizontal asymptote for $n \to 0$ and tends to Daniels $\sigma_{\mathrm{D}}(n)$ when $n \to \infty$. A possible form complying with these two requirements is proposed as:

$$\sigma_{\text{fit}} = k\left(1+\frac{n}{n_0}\right)^{-\frac{1}{2}} \tag{10}$$

This formula has two fitting parameters: k and n_0. The first parameter, k, is the left horizontal asymptote. The second parameter, n_0 has a meaning of the crossover size n, i.e. it is the horizontal coordinate of the intersection of the left horizontal asymptote and the large-size asymptote of slope −½. We, however, require that the two large-size asymptotes match:

$$\lim_{n\to\infty}\sigma_{\text{fit}}(n) = \lim_{n\to\infty}\sigma_{\text{D}}(n). \tag{11}$$

Therefore, the pair of parameters k and n_0 is interconnected. The requirement is that the intersection of the horizontal asymptote with the large-size asymptote occurs at points k and n_0:

$$k = \sigma_{\text{D}}(n_0) \tag{12}$$

We now use this solution in the proposed formula for σ_{fit} to obtain an improved approximation for the standard deviation:

$$\sigma_{\text{fit}} = \frac{m^{-\frac{1}{m}}s\sqrt{e^{-\frac{1}{m}}(1-e^{-\frac{1}{m}})}}{\sqrt{n_0}}\left(1+\frac{n}{n_0}\right)^{-\frac{1}{2}} \tag{13}$$

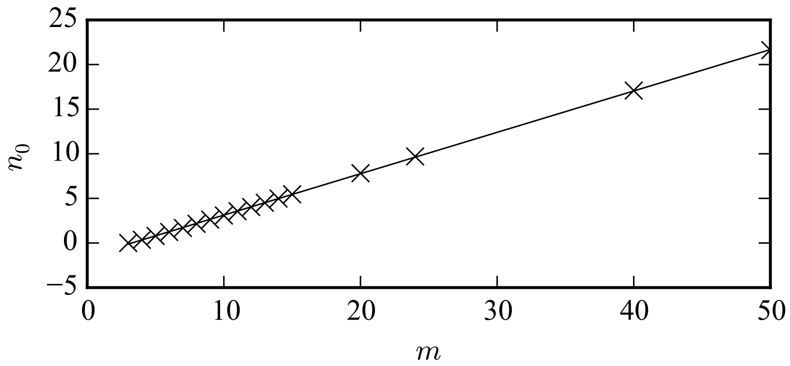

Figure 4: Dependence of the fitting parameter n_0 on shape parameter m.

In this formula, there is one fitting parameter n_0. This parameter depends on the shape parameter m and was obtained by fitting the numerical results using the recursive formula. The m parameter was varied from 3 to 50 and the range for fiber bundle sizes n was considered as being from 1 to 1000. Using the least square method it was found that the dependence of n_0 on m is approximately linear (see Fig. 4):

$$n_0 = 0.461m - 1.464. \tag{14}$$

Fig. 3 compares the exact solution for the standard deviation obtained from G_n with the approximation proposed in Eq. (13). It is clear that the proposed approximation has excellent performance and, to a large extent, fixes the errors achieved by using the original Daniels solution, which is valid only for $n \to \infty$.

4 CONCLUSIONS

The paper describes an algorithm and the implementation of a recursive formula for the evaluation of the cumulative distribution function of random bundle strength, $G_n(x)$. This implementation enables the calculation of $G_n(x)$ for thousands of fibers and small probabilities. An arbitrary probabilistic distribution of the random strength of fibers can be assumed.

The presented algorithm is being used to create a database of cumulative densities of bundle strengths for various Weibull strength distribution parameters for a single fiber m and various bundle sizes n. The database will serve as data support for newly formulated analytical approximate formulas for the CDF of Daniels bundle strength.

In this paper, the database was used to formulate the improved approximation of the standard deviation of bundle strength for small numbers of fibers.

ACKNOWLEDGMENT

The work was supported by the Ministry of Education, Youth and Sports of the Czech Republic under project number FAST-S-14-2443, and by Brno University of Technology, project number CZ.1.07/2.3.00/30.0005. The work of the second author was supported by the Czech Science Foundation under project number GACR GC13-19416J.

REFERENCES

[1] H.E. Daniels, The Statistical Theory of the Strength of Bundles of Threads, *Proc. Royal Soc. (London)*, **183A**, 405–435, 1945.

[2] H.E. Daniels, The Maximum of a Gaussian Process Whose Mean Path Has a Maximum, with an Application to the Strength of Bundles of Fibres, *Advances in Applied Probability*, **21** (2), 315–333, 1989.

[3] R.L. Smith, S.L. Phoenix, Asymptotic Distributions for the Failure of Fibrous Materials Under Series-Parallel Structure and Equal Load-Sharing, *Journal of Applied Mechanics*, **48**, 75–82, 1981.

[4] D. Sornette, Elasticity and failure of a set of elements loaded in parallel, *J. Phys. A: Math. Gen.*, **22**, L243–L250, 1989. (Letter to the Editor).

[5] G. Rossum, et al., The Python Programming Language, `http://python.org`, 1991.

[6] F. Johansson, et al., Mpmath: a Python library for arbitrary-precision floating-point arithmetic (version 0.18), `http://mpmath.org`, 2014.

[7] T. Oliphant, et al., NumPy: Open source scientific package for scientific computing with Python, `http://numpy.scipy.org`, 2006.

[8] M. Vořechovský and R. Chudoba, Stochastic modeling of multi-filament yarns: II. Random properties over the length and size effect, *International Journal of Solids and Structures*, **43** (3-4), 435–458, ISSN 0020-7683, doi:10.1016/j.ijsolstr.2005.06.062, 2006.