Proc. of the 15th Int. Workshop on Slow Positron Beam Techniques and Applications, Prague, September 2–6, 2019

PLRF Code for Decomposition of Positron Lifetime Spectra

J. Čížek

Faculty of Mathematics and Physics, Charles University, V Holešovičkách 2, 18000, Praha 8, Czech Republic

A new code PLRF (version 19) was developed for decomposition of positron lifetime spectra. The PLRF code is based on the least squares fitting of positron lifetime spectra and utilizes a minimization routine MINUIT from the ROOT package developed at CERN. Multiple positron lifetime spectra can be fitted simultaneously using the PLRF code. Each fitting parameter can be selected to be either common (i.e., the same value for all spectra) or individual (i.e., different values for each spectrum). Positronium (Ps) contribution is considered as a double exponential component consisting of short lived para-positronium (p-Ps) and long lived ortho-positronium (o-Ps) component. Ratio of o-Ps and p-Ps contribution can be constrained. Several models were implemented into the PLRF code: independent exponential components, simple trapping model, and diffusion trapping model. Physically relevant parameters of the model are varied to get best agreement with experimental positron lifetime spectra. As a consequence the user obtains physically meaningful parameters from fitting, e.g., positron trapping rates in the case of the simple trapping model, grain or cell size and dislocation density in the case of the diffusion trapping model, etc.

DOI: 10.12693/APhysPolA.137.177

PACS/topics: 78.70.Bj, 02.70.-c

1. Introduction

Positron lifetime spectrum is a superposition of exponential components convoluted with the resolution function of spectrometer. Each component corresponds to certain positron state in the sample. Hence, decomposition of positron lifetime spectra into individual components is crucial for identification of defects present in the sample. Decomposition of positron lifetime spectra is, however, not an easy task mainly due to smearing of the components by convolution with resolution function of the spectrometer and mutual correlations between the parameters of exponential components and the parameters of resolution function.

Several codes for decomposition of positron lifetime spectra have been developed so far. Positronfit [1–3], LT [4], and PALSfit [5, 6] represent the most frequently used ones. PLRF is a new code for decomposition of positron lifetime spectra developed in the positron annihilation group at the Charles University, Prague. The driving force for the development of the PLRF code was twofold:

(i) Digital positron lifetime spectrometers [7–14] are extensively used nowadays. Pulses from scintillation detectors are sampled by fast digitizers and positron lifetime spectra are constructed by analysis of acquired waveforms. Contrary to the traditional analogue configuration, in the digital setup both detectors can be used as start and stop simultaneously [8]. Hence, even by using two detectors one obtains two independent positron lifetime spectra from each measurement [15]. For multiple detectors the number positron lifetime spectra obtained simultaneously is even higher. The best way of analysis of these spectra is simultaneous fitting using common parameters which are the same for all spectra (e.g., lifetimes and intensities of exponential components) and individual parameters which are different for each spectrum (e.g., parameters of the resolution function). The PLRF code enables simultaneous fitting of multiple positron lifetime spectra.

(ii) Results of decomposition of positron lifetime spectra are interpreted using a trapping model which describes the kinetics of positron trapping in the sample. Positron trapping model enables to calculate physically relevant parameters describing the real structure of the material from lifetimes and intensities of individual components resolved in positron lifetime spectra. The simple trapping model (STM) [16] is used most frequently. However, proper trapping model can be in principle developed for every structure. An example is trapping model describing shallow traps in semiconductors [17, 18] or diffusion trapping model (DTM) developed for description of positron trapping at grain boundaries [19] and in metals with cellular dislocation structure [20]. It is desirable to fit the trapping model directly to positron lifetime spectra, i.e., to calculate model function using the relevant trapping model and vary the physical parameters describing the real structure of material to achieve the best agreement between the model function and the experimental spectrum [21]. This is more accurate than application of trapping model *ex-post* on lifetimes and intensities of exponential components obtained from unconstrained decomposition of positron lifetime spectra. The PLRF code enables not only fitting of positron lifetime spectra by independent exponential components (model 0) but also direct fitting within selected positron trapping model. Two trapping models, namely STM (model 1) and DTM (model 2) have been integrated in the PLRF code (version 19) so far.

corresponding author; e-mail: jakub.cizek@mff.cuni.cz

2. Description of the PLRF code

2.1. Model function

The default model in the PLRF code is represented by independent exponential components. Positron lifetime spectrum is considered as a superposition of N exponential components each characterized by its lifetime τ_i and relative intensity I_i convoluted with the resolution function R of the spectrometer

$$S(t) = N_{\mathrm{tot}} \sum_{i=1}^{N} \frac{I_i}{\tau_i} \mathrm{e}^{-\frac{t-t_0}{\tau_i}} \star R(t-t_0) + b, \tag{1}$$

where N_{tot} is the total statistics accumulated in the spectrum, t_0 is the origin of time scale, b is background from random coincidences and the symbol $\star$ stands for convolution. The intensities of individual components are normalized, so that $\sum_{i=1}^{N} I_i = 1$. The resolution function is parameterized as a sum of N_G Gaussians

$$R(t) = \sum_{j=1}^{N_G} \frac{I_{G_j}}{\sqrt{2\pi}\sigma_j} \exp\left(-\frac{(t-\mu_j)^2}{2\sigma_j^2}\right), \tag{2}$$

where μ_i and σ_i, respectively, describe the expectation value and the standard deviation of the j-th Gaussian. The relative intensities of Gaussians are normalized, i.e.,

$$\sum_{j=1}^{N_G} I_{G_j} = 1.$$

The convolution of i-th exponential component with j-th Gaussian can be solved analytically

$$c_{i,j}(t) = \frac{1}{2} \exp\left(\frac{\sigma_j^2}{2\tau_i^2} - \frac{t-t_0+\mu_j}{\tau_i}\right) \times \left[1 - \mathrm{erf}\left(\frac{-t+t_0-\mu_j+\sigma_j^2/\tau_i}{\sqrt{2}\sigma_j}\right)\right], \tag{3}$$

where $\mathrm{erf}(t)$ is the error function

$$\mathrm{erf}(t) = \frac{2}{\sqrt{\pi}} \int_0^t \mathrm{e}^{-x^2} \mathrm{d}x. \tag{4}$$

The positron lifetime spectrum is described by the expression

$$S(t) = N_{\mathrm{tot}} \sum_{i=1}^{N} \sum_{j=1}^{N_G} \frac{I_i}{\tau_i} c_{i,j} + b. \tag{5}$$

The model function described by (5) is fitted to the experimental data by the least squares method. Estimators of parameters $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_m)$ of the model function are obtained by minimizing the χ^2 functional

$$\chi^2(\boldsymbol{\theta}) = \sum_{k=1}^{N_S} \sum_{i=1}^{N_{\mathrm{ch}}} \frac{(S_k(t_i|\boldsymbol{\theta}) - y_{k,i})^2}{S_k(t_i|\boldsymbol{\theta})}. \tag{6}$$

where N_S denotes the number of spectra fitted simultaneously, N_{ch} is number of channels of each spectrum, and $y_{k,i}$ denotes number of counts in the i-th channel of the k-th spectrum. The parameters that can be estimated are, for example, lifetimes and intensities of exponential components, parameters of resolution function, time origin, background etc.

2.2. Positronium contribution

Positronium (Ps) contributions are described by double exponential components representing decay of ortho-Ps (long-lived component) and para-Ps (short-lived component). Ratio of these components can be constrained at desired value, usually 0.75, corresponding to the formation ratio 3:1 of ortho-Ps and para-Ps.

2.3. Parametrization of the resolution function

The resolution function is considered as a sum of Gaussians. Several parametrizations of the resolution function can be used in the PLRF code.

Figure 1 shows the "parametrization 1" of the resolution function consisting of two constrained Gaussians. The width (FWHM) of Gaussians w_1 and w_2 are fitting parameters. The Gaussians are placed symmetrically around the time origin T_0. The distance between their maxima DT is additional fitting parameter. Both Gaussians have also the same intensity. Hence, the resolution function is described by 3 fitting parameters w_1, w_2, and DT.

In the "parametrization 2" the assumptions of symmetrical position around the time origin is relaxed. The intensities of both Gaussians however are kept the same. Hence, in this parametrization each Gaussian is described by its position μ_i and width (FWHM) w_i. The resolution function is described then by 4 fitting parameters w_1, w_2, μ_1, and μ_2.

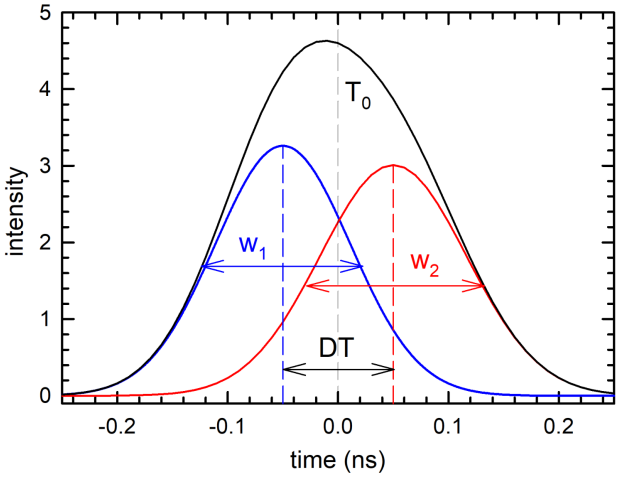

Fig. 1. Parametrization of resolution function consisting of two Gaussians. The symbols w_1, w_2 denote FWHM of the two Gaussians, DT is the separation of Gaussians, and T_0 is the time origin.

In the "parametrization 3" the resolution function is described by two completely unconstrained Gaussians. Each Gaussian is described by its position μ_i, width (FWHM) w_i and relative intensity I_{G_j}. Both Gaussians intensities are normalized, so that $I_{G_1} + I_{G_2} = 100\%$. Therefore the resolution function is described by 5 fitting parameters w_1, w_2, μ_1, μ_2, and I_{G_1}.

It was found that the "parametrization 1" is sufficient to describe well positron lifetime spectra measured by the majority of spectrometers. Releasing of constrain and adding of free fitting parameters in the parametrizations 2 and 3 usually did not led to a statistically significant improvement of the agreement between the model function and the experimental data but only increased uncertainties of fitting parameters.

2.4. Fitting of positron lifetime spectra

Minimization of the χ^2 functional is performed using MINUIT [22] routine which is included in the ROOT [23] package developed at CERN. The MINUIT routine calculates minimum of a function of multiple variables (here χ^2). User can select and combine various minimization algorithms including Monte Carlo minimization [24] (SEEK), Simplex method of Nelder and Mead [25] (SIMPLEX), and variable-metric [26] gradient algorithm (MINIMIZE). Note that MINUIT has been already employed for fitting of positron lifetime spectra as described in [27].

Each parameter of the model function can be selected to be either common or individual. Common parameters (lifetimes and intensities of exponential components, parameters of the source contribution, etc.) have the same values for all spectra fitted. Individual parameters (parameters of resolution function, time origin, random background, etc.) have different, independent values for different spectra. Hence, each individual parameter is actually a vector of k parameters, where k is the number of spectra fitted simultaneously.

2.5. Input file

When executed the PLRF reads input file containing information about fitted spectra, time calibration, selection of model, and definition of all fitting parameters and their initial guesses. The default name of the input file is `plrf19.ini`. Source code I shows an example of the input file for simultaneous fitting of two positron lifetime spectra containing two exponential components (their lifetimes and intensities are fitted as common parameters) and source contribution consisting of single exponential component and a Ps component (this is typical source contribution for ^{22}Na source deposited on a Mylar foil [13, 15]).

One can see in Source code I that each fitting parameter is defined on a separate line. The meaning of various quantities used in definition of fitting parameters is explained in Fig. 2. The first value is the number of the parameter. It increases progressively for every next parameter. It is followed by name of the parameter, initial guess

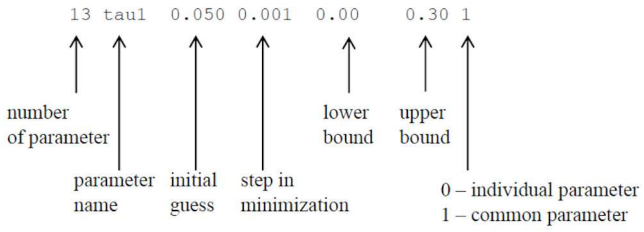

Fig. 2. Definition of a fitting parameter in the input file.

of the parameter value, and step in the parameter value which will be applied in minimizing. Upper and lower bounds defines the interval to which the parameter must fall. This interval should correspond to physically relevant values of corresponding parameter (e.g., 0–100 % for intensity). The code keeps the parameter values inside this region during minimization of the χ^2 functional. The last value defines whether the parameter is considered as common or individual. The value 1 means common parameter with the same value for all spectra fitted simultaneously while the value 0 means individual parameter with different values for different spectra. In the latter case a vector of k parameters, where k is the number of spectra fitted simultaneously, is created for every individual parameter. These parameters are numbered progressively. Hence, for example, when fitting simultaneously two spectra the definition line

```
 1 bcg 20.0 0.01 0.00 100.0
```

means that two parameters with numbers 1 and 2 are created. First one `bcg(1)` describes the background level of the first spectrum, the second one `bcg(2)` describes the background level of the second spectrum. Numbers of subsequent parameters are shifted accordingly. It means in the present example the number of the next parameter (`T0`) will be 3. The definition of all parameters with their assigned numbers is written in the beginning of the output file (default name is `plrf19.res`) which is shown in Source code II.

The last part of the input file contains commands for MINUIT providing instructions for minimization. The command FIX keeps the selected fitting parameters constant during minimization. In the present example the parameters of the source contributions (`Is, tau1s, Int1s, tau2sp, I2rats`) are fixed at the values determined using a reference Fe sample. Note that lifetime of the long-lived o-Ps component of the source contribution (`tau2so`) is fitted as a free parameter because it is well separated from the other components.

The command SCAN is used for one-dimensional minimization of χ^2 with respect a selected parameters while the other parameters are kept constant. In the present example it is used to determine proper initial guess of the time origin of each spectrum. This step improves convergence of fitting procedure since χ^2 strongly depends on the position of time origin and initial guess should be close to the proper value.

Source code I: An example of input file of the PLRF fitting code. Sections corresponding to various parts of the model function are separated by hashtags. Intensities are given in percents (%), time variables (lifetimes, FWHM and separation of Gausians) are in nanoseconds (ns). Comments are written in italics. The present example of input file corresponds to simultaneous fitting of two spectra containing two exponential components and in addition source contribution consisting of single exponential component and a Ps contribution.

```
#spectrum
2,1 ! number of spectra, format of spectra (1- xy format, 2- xyz format, 3-single column format)
f:\data\timesp_Ks1cr1.dat  !path to first spectrum
f:\data\timesp_Ks1ci1.dat  !path to second spectrum
9999  !number of channels
1,9999  !fitted region in the first spectrum (in channels)
1,9999  !fitted region in the second spectrum (in channels)
#calibration(ns)
0.003125  time calibration (ns per channel)
#parameters
#bcg
1  bcg     20.0 0.01 0.00 100.0    0  !background
2  T0    2500.0 0.1  0.00 10000.0 0  !time origin (in channels)
#RF  !parameters of resolution function
1  !parametrization of resolution function, see Fig. 1
3  DT      0.040 0.001 -0.20 0.20 0  !separation of Gaussians
4  w1      0.144 0.001  0.00 0.20 0  !FWHM of first Gaussian
5  w2      0.156 0.001  0.00 0.20 0  !FWHM of second Gaussian
#inverted_spectrum
6  Iinv    0.000 0.000  0.00 0.10 0  !fraction of inverse spectrum
#source  !source contribution (I_s is the total intensity of source contribution, intensities of individual
         components of source contribution are normalized to 100%)
1,1  !number of discrete exponential components, number of Ps components
7  Is      8.8   0.01   0.00 100.00 1  !total intensity of source contribution
8  tau1s   0.368 0.001  0.00   0.50 1  !lifetime of the first exponential component
9  Int1s   86.5  0.01   0.00 100.00 1  !relative intensity of the first component
10 tau2sp 0.125 0.001  0.00   5.50 1  !lifetime of p-Ps component
11 tau2so 1.5   0.001  0.00   5.50 1  !lifetime of o-Ps component
12 I2rats 75.00 0.01   0.00  100.0 1  !relative fraction of o-Ps
#sample
0  !selected model 0 – independent exponential components, 1 – STM, 2 – DTM
2,0  !number of discrete exponential components, number of Ps components
13 tau1   0.050 0.001  0.00   0.30 1  !lifetime of the first exponential component
14 Int1   60.0  0.01   0.00  100.0 1  !intensity of the first exponential component
15 tau2   0.155 0.001  0.00   0.50 1  !lifetime of the second exponential component
!intensities of individual components are normalized to 100%
#commands
FIX,13,14,15,16,18  !Is,tau1s, Int1s,tau2sp, I2rats will be fixed
SCAN,3  !find estimation of time origin for the first spectrum
SCAN,4  !find estimation of time origin for the second spectrum
SEEK,100  !Monte Carlo minimization
SIMPLEX  !simplex minimization
MINIMIZE  !gradient minimization
RETURN  !write results and finish
```

Source code II: An example of output file created by the PLRF code. The output file corresponds to data for a martensitic steel fitted using the input file in Source code I, i.e., simultaneous fitting of two spectra containing two exponential components and in addition a source contribution consisting of single component and a Ps contribution. Comments are written in italics.

```
!output begins with information about fitted spectra
number of spectra: 2
type of spectra: 1 > xy format
PL spectrum 1: f:\data\timesp_Ks1cr1.dat
PL spectrum 2: f:\data\timesp_Ks1ir1.dat
total number of channels: 9999
calibration: 0.00312500 ns per channel

Spectrum f:\data\timesp_Ks1cr1.dat:
fitted range (channels): 1 - 9999
total statistics: 4065463.000000
maximum 38544 in channel 4005
Spectrum f:\data\timesp_Ks1ir1.dat:
fitted range (channels): 1 - 9999
total statistics: 5087249.000000
maximum 48636 in channel 2449

!summary of defined fitting parameters and their initial guesses
Background and T0
 1 bcg(1)  24.4160   0.0010    0.0000  1000.0000
 2 bcg(2)  28.2460   0.0010    0.0000  1000.0000
 3 T0(1) 4005.0000   0.1000    0.0000 10000.0000
 4 T0(2) 2449.0000   0.1000    0.0000 10000.0000
RF model: 1 mirrored double-Gaussian
 5 DT(1)    0.0140   0.0010   -0.2000     0.2000
 6 DT(2)    0.0140   0.0010   -0.2000     0.2000
 7 w1(1)    0.1400   0.0010    0.0000     0.2000
 8 w1(2)    0.1400   0.0010    0.0000     0.2000
 9 w2(1)    0.1500   0.0010    0.0000     0.2000
10 w2(2)    0.1500   0.0010    0.0000     0.2000
 inverted spectrum   !inverse spectrum is not used here
11 Iinv(1)  0.0000   0.0000    0.0000     0.1000
12 Iinv(2)  0.0000   0.0000    0.0000     0.1000
source model: 1 component(s), 1 complex component(s)
13 Is       8.8300   0.0100   0.0000    100.0000
14 tau1s    0.3680   0.0010   0.0000      0.5000
15 Int1s   86.5000   0.0100   0.0000    100.0000
16 tau2s1   0.1250   0.0010   0.0000      5.5000
17 tau2s2   2.5000   0.0010   0.0000      5.5000
18 I2rats  75.0000   0.0100   0.0000    100.0000
sample model: independent 2 exponential component(s)
                          0 complex component(s)
19 tau1     0.0500   0.0010   0.0100      0.3000
20 int1    50.0000   0.0100   0.0000    100.0000
21 tau2     0.2000   0.0010   0.0100      5.3000
```

Source code II continue

```
spectrum 1 initial RF total FWHM: 0.143293 ns
spectrum 2 initial RF total FWHM: 0.143293 ns
number of parameters: 19
common parameters: 13,14,15,16,17,18,19,20,21,
number of common parameters: 9
!estimators of fitting parameters obtained from fitting
*************** plrf19 results ***************
spectra:
f:\data\timesp_Ks1cr1.dat total area: 4.065e+06 counts
f:\data\timesp_Ks1ci1.dat total area: 5.087e+06 counts
number of parameters: 19
number of free parameters : 13
number of fixed parameters: 6
Background and T0
 1 bcg(1)  24.36948 +/- 0.05372
 2 bcg(2)  28.89186 +/- 0.05923
 3 T0(1)  3988.1524 +/- 0.09143
 4 T0(2)  2426.6082 +/- 0.09495
RF model 1 mirrored double-Gaussian
 5 DT(1)    0.01267 +/- 0.00023
 6 DT(2)    0.01263 +/- 0.00109
 7 w1(1)    0.14311 +/- 0.00077
 8 w1(2)    0.14549 +/- 0.00081
 9 w2(1)    0.15849 +/- 0.00103
10 w2(2)    0.15390 +/- 0.00154
inverted spectrum
11 Iinv(1)  0.00000 +/- 0.00000
12 Iinv(2)  0.00000 +/- 0.00000
source model: 1 component(s), 1 complex component(s)
13 Is       8.83000 +/- 0.00000
14 taus1    0.36800 +/- 0.00000
15 Is1     86.50000 +/- 0.00000
16 taus1c1  0.12500 +/- 0.00000
17 taus2c1  1.71379 +/- 0.02729
18 Iratc21 75.00000 +/- 0.00000
sample model: independent 2 exponential component(s)
0 complex component(s)
19 tau1     0.02054 +/- 0.00531
20 I1       6.67647 +/- 0.24412
21 tau2     0.15483 +/- 0.0013
   I2      93.32353 ! I2 is calculated as a complement to 100%
!total FWHM of the resolution function
spectrum 1 RF total FWHM: 0.143995 ns
spectrum 2 RF total FWHM: 0.143781 ns
!χ2 divided by the number of degrees of freedom
spectrum 1 chi2 per degrees of freedom: 1.019897 +/- 0.014152
spectrum 2 chi2 per degrees of freedom: 1.012574 +/- 0.014152
```

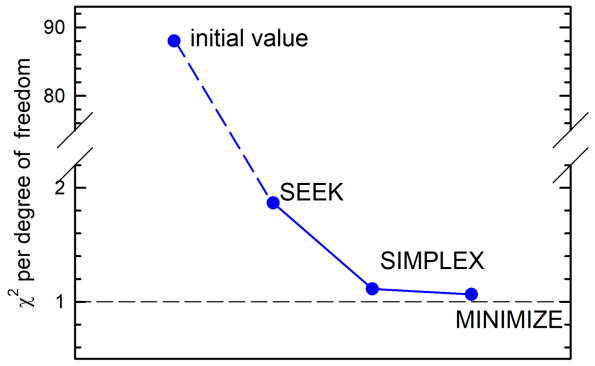

Fig. 3. The development of the χ^2 value divided by the number of degrees of freedom during fitting of positron lifetime spectra using the input file shown in Source code I.

The command SEEK starts Monte Carlo minimizing. Subsequently the command SIMPLEX starts simplex minimization which is very robust and provides rough localization of the minimum of χ^2. Finally the gradient minimization is performed by the command MINIMIZE. The variable-metric gradient minimization is very precise but it is much more sensitive to the initial guess than the simplex algorithm. The development of the χ^2 value divided by the number of degrees of freedom during fitting performed using the input file in Source code I is plotted in Fig. 3.

2.6. Output files

When fitting is finished the PLRF code creates several output files:
`plrf19.res` — file contains results of fitting
`plrf19.p1, plrf19.p2, ...` — files contain experimental data and calculated model function for the first, second, ... spectra fitted. These files can be used for plotting of results, as shown in Fig. 4.
`plrf19.cnt` — file is an input file with initial guesses replaced by the best estimators obtained from fitting. One can use this file as an input file for next iteration, e.g., after releasing some fitted parameter, in order to start with initial guess close to the actual minima of the χ^2 functional.

Source code II shows an example of the output file `plrf19.res` corresponding to the input file defined in Source code I.

2.7. Performance of the PLRF code

The PLRF (version 19) code was written in C++ language and is available on the web page of Positron annihilation group at the Charles University [28]. The code uses ROOT libraries. Hence, ROOT system has to be installed in the computer in order to run the PLRF code. ROOT is a modular scientific software toolkit developed in CERN which is freely available on the CERN webpage [23].

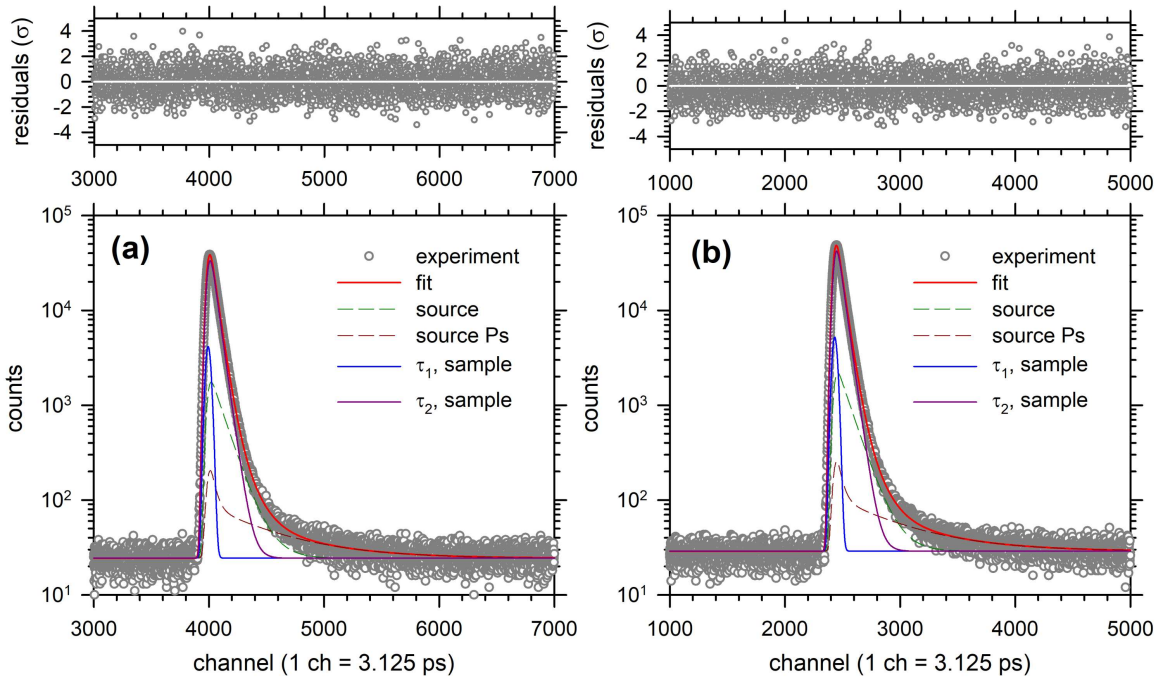

Fig. 4. Results of simultaneous fitting of two positron lifetime spectra for a martensitic steel sample measured by a digital positron lifetime spectrometer [8]. Fitting was performed using the input file in Source code I and output file in Source code II. Experimental data are shown by circles; model functions obtained from fitting are plotted by solid lines. Residuals in units of one standard deviation are shown in the upper panels.

The built PLRF code can be started from the command line using the following syntax
`plrf19 [-i myinput.txt] [-o myoutput.txt][-1].`

The default name of the input file is `plrf19.ini`. The option `-i` enables to specify different name of the input file. Similarly the command line option `-o` allows to specify the name of the output different from the default name `plrf19.res`.

When using the command line option `-1` the PLRF will only plot the model function corresponding to the initial guess of the parameters, and then it will finish. This option is useful to check if the initial guesses of fitting parameters in the input file are reasonable.

During running the PLRF code prints on the screen the current value of the χ^2 functional. The positron lifetime spectra and the model function corresponding to the best estimates of fitting parameters are plotted when minimization of χ^2 is finished. Residuals, i.e.,

$$r_{k,i} = \frac{S_k\left(t_i|\boldsymbol{\theta}\right) - y_{k,i}}{\sqrt{S_k\left(t_i|\boldsymbol{\theta}\right)}}, \tag{7}$$

read as weighted differences between the model function and experimental spectrum are plotted for each spectrum as well. This enables to check visually the quality of fit. An example of such plot corresponding to fitting using the input file listed in Source code I and the output file in Source code II is shown in Fig. 4.

2.8. Inverted spectrum

Accidental pile-ups of detector events cause distortion of the leading edge of positron lifetime spectrum which can be described as a small fraction of the spectrum inverted along the time origin. This is taken into account in the PLRF code and the model function may contain a small fraction of spectrum inverted along the time origin in order to provide better description of the leading edge of experimental positron lifetime spectrum. The fraction of events belonging to the inverted spectrum is a fitting parameter (`Iinv`), see Source code I. The inverted spectrum is not used for description of positron lifetime spectra measured by the digital spectrometer [8] since pile-up events are almost completely suppressed by digital filters. Indeed, one can see in Fig. 4 that positron lifetime spectra measured by the digital spectrometer are well described by model function with zero fraction of inverted spectrum. On the other hand, spectrum measured using an analogue spectrometer [29] contains a small fraction of inverted spectrum as illustrated in Fig. 5.

2.9. STM

Direct fitting within STM [16] can be performed by selecting model 1 in the input file. The meaning of parameters in the section `#sample` of the input file is modified as illustrated in Source code III. The corresponding section of the output file is shown in the Source code III as well. The data correspond to the same martensitic steel sample as that considered in Source code II. The bulk

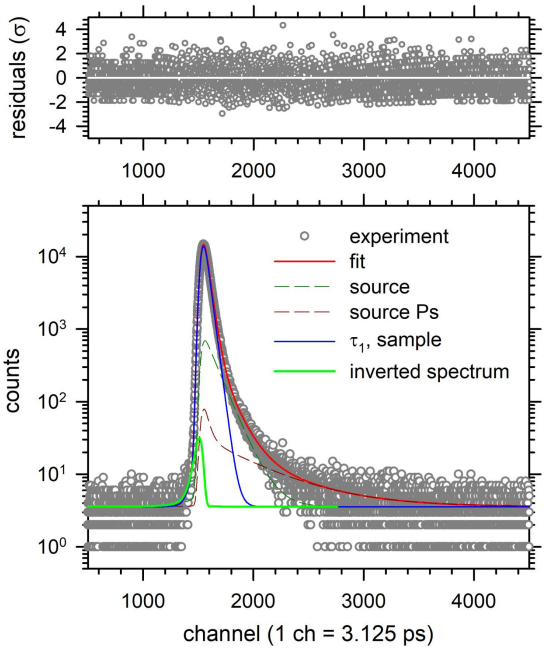

Fig. 5. Results of fitting of positron lifetime spectra for a well annealed Fe measured using an analogue spectrometer [29]. Leading edge of the spectrum contains a small fraction of inverted spectrum caused by contribution of pile-up events.

positron lifetime was fixed at 107 ps measured for well annealed Fe [13]. From comparison of Source code II and Source code III one can conclude that results of direct fitting within STM are consistent with results of fitting by unconstrained exponential components.

Direct fitting within STM reduces the correlation between lifetime τ_1 of the short-lived free positron component and the resolution function and it enables to use knowledge of the bulk positron lifetime which is often known from measurement of reference samples. Moreover the constraint among the exponential components

$$\frac{1}{\tau_B} = \sum_{i=1}^{N} \frac{I_i}{\tau_i}, \tag{8}$$

which holds within STM is inherently used in direct fitting using the model 1. Figure 6 shows histograms of positron lifetimes obtained from fitting of 100 simulated spectra. The spectra were simulated with the same parameters as those of the spectrum for the martensitic steel sample shown in Source code II. Simulated spectra were fitted by the PLRF code using independent components (model 0) and using STM (model 1). In the latter case the lifetime of the free positron component was calculated from results of fitting using the relation

$$\tau_1 = \frac{1}{(1/\tau_B) + K_D}. \tag{9}$$

It is clear from Fig. 6 that the spread of lifetimes obtained from fitting using independent components (model 0) is significantly larger than the spread of lifetimes obtained from direct fitting within STM (model 1).

Source code III: First part of the code is an example of the section `#sample` in the input file to be used for direct fitting of positron lifetime spectra within STM. Second part of Source code III shows corresponding part of the output file with results of fitting for a martensitic steel the spectra of which are shown in Fig. 4. In the present example single kind of positron traps is considered.

```
!Input file
#sample
1   !STM model
2,0 !two exponential components (two kinds of positron traps)
13 tauB  0.107 0.001 0.0  0.30 1 !bulk lifetime
14 tauD1 0.155 0.001 0.0  0.50 1 !lifetime of positrons trapped at defects
15 KD1   1.000 0.001 0.0 50.00 1 !positron trapping rate to defects
!Output file
sample model: STM with 2 positron states
19 tauB  0.10700 +/- 0.00000 !bulk lifetime was fixed at 0.107 ns
20 tauD1 0.15485 +/- 0.00013 !lifetime of positrons trapped at defects
21 KD1  33.57637 +/- 0.34820 !positron trapping rate to defects
lifetimes and intensities calculated within STM:
tau1 = 0.02330 I1 = 7.91960
tau2 = 0.15485 I2 = 92.08040
spectrum 1 RF total FWHM: 0.144476 ns
spectrum 2 RF total FWHM: 0.144297 ns
spectrum 1 chi2 per degrees of freedom: 1.001496 +/- 0.015824
spectrum 2 chi2 per degrees of freedom: 0.991604 +/- 0.015824
```

Source code IV: First part of the code is an example of the section `#sample` in the input file to be used for direct fitting of positron lifetime spectra within DTM. Second part of the code shows corresponding part of the output file with results of fitting for a plastically deformed ferritic steel sample.

```
!Input file
#sample
2    !DTM model
1,0  !single type of defects
13 tauB 0.107 0.001 0.00    0.5  1 !bulk positron lifetime
14 tauD 0.155 0.001 0.00    0.50 1 !lifetime of positrons trapped at dislocations
15 R   20.0   0.1   0.00  200.0  1 !dislocation cell radius (nm)
14 roD  5.0   0.01  0.00 1000.00 1 !dislocation density in dislocation walls (10^14 m^-2)
5.0 !width of dislocation walls (nm)
1.87 !positron diffusion coefficient (cm^2/s)
0.5e-4 !specific positron trapping rate (m^2/s)
!Output file
sample model: DTM
13 tauB 0.107               !bulk positron lifetime was fixed at 0.107 ns
15 tauD 0.1525  +/- 0.0034 !lifetime of positrons trapped at dislocations
13 R   54.6973  +/- 0.0384 !dislocation cell radius (nm)
14 roD  4.2400  +/- 0.5801 !dislocation density in dislocation walls (10^14 m^-2)
volume fraction: 0.230810  !volume fraction of dislocation walls
boundary width:  5.0       !width of dislocation walls (nm)
e+ diffusion coefficient: 1.87 !positron diffusion coefficient (cm^2/s)
specific e+ trapping rate for dislocations: 0.5e-4 !specific e+ trapping rate (m^2/s)
mean dislocation density: 0.975 !mean dislocation density in the sample (10^14 m^-2)
```

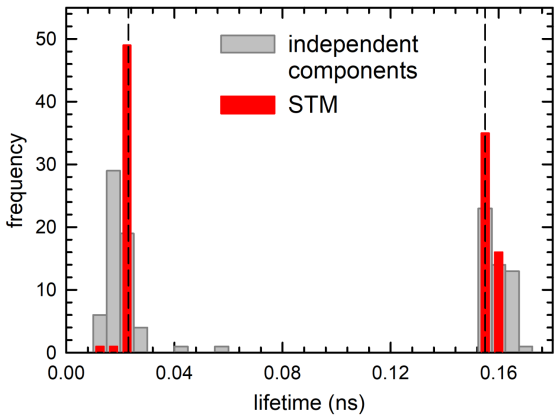

Fig. 6. Results of fitting of 100 positron lifetime spectra simulated with the same parameters as that for the martensitic steel sample shown in Source code II. The gray histogram shows results of fitting using independent components (model 0) while the red histogram shows results of fitting within STM (model 1). Actual values of lifetimes used in simulations are indicated by dashed vertical lines.

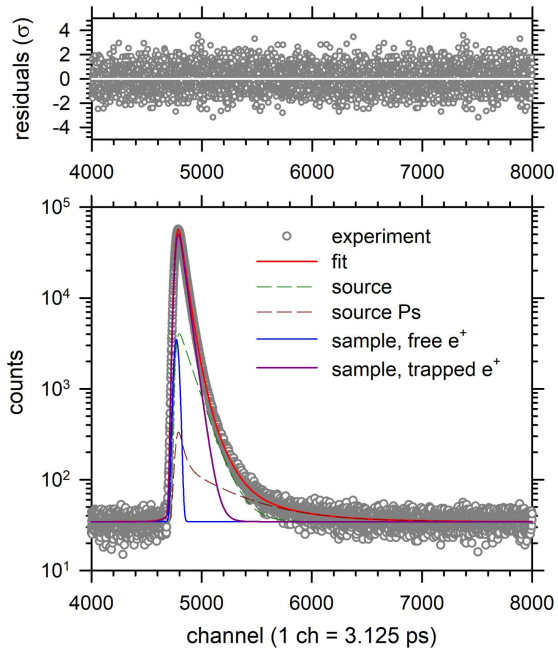

Fig. 7. Results of fitting of positron lifetime spectra for plastically deformed ferritic steel with a cellular dislocation structure using DTM. The free positron contribution consists of a series of exponential components with lifetimes and intensities gradually decreasing; see [19, 20].

Single type of defects was assumed in the example shown in Source code III. Additional kinds of traps can be added simply by increasing the number of components and defining additional lifetimes and trapping rates in the input file.

2.10. DTM

Direct fitting within DTM [19, 20] is made by choosing model 2 in the input file. Source code IV shows an example of the section `#sample` in the input file for direct fitting within DTM. Corresponding part of the output file with results for plastically deformed ferritic steel sample is shown in Source code IV as well. The following fitting parameters are used in DTM: bulk positron lifetime, lifetime of positrons trapped at dislocations, radius of grains or cells, dislocation density in dislocation walls or interfaces. Detailed description of DTM parameters is given in [19, 20].

Figure 7 shows results of fitting of positron lifetime spectrum for plastically deformed ferritic steel using DTM. The results of fitting are listed in the second part of Source code IV. Note that within DTM the free positron contribution is expressed by a series of exponential components with gradually decreasing lifetime and intensity. Sum of the series is plotted in Fig. 7 as a free positron contribution. From inspection of Fig. 7 one can conclude that DTM provides accurate description of positron lifetime spectrum of deformed ferritic steel sample.

2.11. Sequential fitting

The PLRF code enables automatic fitting of a sequence of spectra. This is useful in particular for analysis of spectra measured using a pulsed slow positron beam when one usually obtains series of spectra for various energies of incident positrons. Whenever the name of positron lifetime spectra in the section `#spectrum` of the input file contains ⋆ it is expanded during the PLRF execution by a number of spectrum. For example the following commands in the input file

```
#spectrum
2,1  !number of spectra, format of spectra
c:\bulk\Al\f05mi-*.dat
c:\bulk\Al\f05mr-*.dat
```

mean that the PLRF code will start fitting of couple of spectra

```
c:\bulk\Al\f05mi-1.dat
c:\bulk\Al\f05mr-1.dat
```

subsequently fitting will continue for spectra

```
c:\bulk\Al\f05mi-2.dat
c:\bulk\Al\f05mr-2.dat
```

and so on for the whole sequence of spectra. The number of spectra available is checked automatically. An output file `plrf19.seq` containing results of fitting of the whole sequence in a form of a table is created.

The aforementioned option assumes that names of spectra in the sequence are numbered consequently. There is also another option. The section `#spectrum` of the input may contain name of a file followed by `!` character, for example:

```
#spectrum
1,1 !number of spectra, format of spectra
c:\data\splist.txt!
```

In this case it is assumed that names of spectra are listed in the file `splist.txt`. PLRF will perform sequential fitting and file names of individual spectra will be read from the file `splist.txt`.

Note that positron lifetime spectra measured using a pulsed slow positron beam do not contain the source contribution. Hence, for fitting of such spectra the source contribution can be omitted by setting the number of component in the section `#source` to zero

```
#source
0,0  !number of discrete exponential
     components, number of Ps components.
```

3. Conclusions

PLRF is a new code for decomposition of positron lifetime spectra developed in the positron annihilation group at the Charles University. Fitting of positron lifetime spectra is performed using the least squares method. A routine MINUIT from the ROOT package developed at CERN is employed for minimizing the χ^2 functional. The resolution function of spectrometer is considered in the PLRF code as a sum of Gausians. Multiple positron lifetime spectra can be fitted simultaneously using common parameters with the same values for all spectra and individual parameters with different values for each spectrum. Positronium contribution is described by a double exponential component with constrained intensities corresponding to p-Ps and o-Ps contribution. The PLRF code enables direct fitting of positron lifetime spectra within simple trapping model or diffusion trapping model.

Acknowledgments

Support by the Czech Science Foundation (project 18-09347S) is highly acknowledged.

References

[1] P. Kirkegaardand M. Eldrup, *Comput. Phys. Commun.* **3**, 240 (1972).

[2] P. Kirkegaard, M. Eldrup, O.E. Mogensen, N.J. Pedersen, *Comput. Phys. Commun.* **23**, 307 (1981).

[3] P. Kirkegaard, M. Eldrup, *Comput. Phys. Commun.* **35**, C-255 (1984).

[4] J. Kansy, *Nucl. Instr. Meth. A* **374**, 235 (1996).

[5] J.V. Olsen, P. Kirkegaard, N.J. Pedersen, M. Eldrup, *Phys. Stat. Sol. C* **4**, 4004 (2007).

[6] J.V. Olsen, P. Kirkegaard, M. Eldrup, PALSfit3, 2017.

[7] H. Saito, Y. Nagashima, T. Kurihara, T. Hyodo, *Nucl. Instr. Meth. A* **487**, 612 (2002).

[8] F. Bečvář, J. Čížek, I. Procházka, J. Janotová, *Nucl. Instr. Meth. A* **539**, 372 (2005).

[9] K. Rytsölä, J. Nissilä, J. Kokkonen, A. Laakso, R. Aavikko, K. Saarinen, *Appl. Surf. Sci.* **194**, 260 (2002).

[10] J. Nissilä, K. Rytsölä, R. Aavikko, A. Laakso, K. Saarinen, P. Hautojärvi, *Nucl. Instr. Meth. A* **538**, 778 (2005.

[11] H. Saito, T. Hyodo, *Phys. Rev. Lett.* **90**, 193401 (2003).

[12] Feng-ming Zhu, S.C. Hsieh, W.W. Yen, H.P. Chou, *Nucl. Instr. Meth. A* **652**, 454 (2011).

[13] F. Bečvář, J. Čížek, I. Procházka, *Appl. Surf. Sci.* **255**, 111 (2008).

[14] A.Krille, R. Krause-Rehberg, M. Jungmann, F. Bečvář, G. Brauer, *Appl. Surf. Sci.* **194**, 260 (2002).

[15] F. Bečvář, J. Čížek, I. Procházka, *Acta Physica Polonica A* **113**, 1279 (2008).

[16] R.N. West, *Adv. Phys.* **22**, 263 (1973).

[17] M. Puska, C. Corbel, R.M. Nieminen, *Phys. Rev. B* **41**, 9980 (1990).

[18] C. Corbel, F. Pierre, K. Saarinen, P. Hautojärvi, P. Moser, *Phys. Rev. B* **45**, 3386 (1992).

[19] A. Dupasquier, R. Romero, A. Somoza, *Phys. Rev. B* **48**, 9235 (1993).

[20] J. Čížek, I. Procházka, M. Cieslar, R. Kužel, J. Kuriplach, F. Chmelík, I. Stulíková, F. Bečvář, O. Melikhova, R.K. Islamgaliev, *Phys. Rev. B* **65**, 094106 (2002).

[21] J. Čížek, *J. Mater. Sci. Technol.* **34**, 577 (2018).

[22] F. James, "Function Minimization and Error Analysis Reference Manual", CERN, Geneva, 2004.

[23] CERN, ROOT package.

[24] F. James, in: *Monte Carlo for Particle Physics in: Methods in Subnuclear Physics*, Ed. M. Nikolic, Gordon and Breach Publ., Vol. IV, part 3.

[25] J.A. Nelder, R. Mead. *Comput. J.* **7**, 308 (1965).

[26] R. Fletcher, *Comput. J.* **13**, 317 (1970).

[27] I. Procházka, I. Novotný, F. Bečvář, *Mater. Sci. Forum* **255**, 772 (1997).

[28] physics.mff.cuni.cz/kfnt/pas.

[29] F. Bečvář, J. Čížek, L. Lešták, I. Novotný, I. Procházka, F. Šebesta, *Nucl. Instr. Meth. A* **443**, 557 (2000).