Moessfit
a mössbauer fitting program

Sirko Kamusella

September 19, 2014

Moessfit is a fitting programm written for complex fitting tasks including

- global fit parameters
- maximum entropy distributions
- user defined parameter distributions
- usage runspecific parameter function as temperature dependent magnetic order parameter or debye models for Isomershift and spectral area (debye waller factor)

The aim of Moessfit is the simultanous fitting of multiple moessbauer spectra (runs) with a common model (theory). For this purpose a data input from a mbs-textfile similiar to musrfit[SW12].

A typical Moessfit fitting project consists of a project folder consisting of subfolders which contain folded moessbauer spectra (*.fld). Please note that only the newest fld-files of each subfolder are considered by Moessfit. The project folder contains the Moessfit input file (*.mbs), which is a textfile with the following blocks: FITPARAMETER, THEORY, RUN, FUNCTIONS, COMMANDS, PLOT, FITDATA, MEMDATA and FITSTATISTICS. The first three are necessary for a valid Moessfit input. The blocks are explained in the sections below. Although the user can edit parameters via the main panel it is strongly recommended to get used to an advanced text editor to modify the mbs-file directly and reload it subsequently.

To run the programm you have to run the MoessFit.exe. The programm needs access to the following dll located in the same folder like the executable:

- Windows XP: libgcc_s_dw2-1.dll, mingwm10.dll , QtCored4.dll and QtGuid4.dll, libgsl-0.dll, libgslcblas-0.dll.
- Windows7/8: Qt5Widgets.dll, icudt51.dll, icuin51.dll, icuuc51.dll, libEGL.dll, libGLESv2.dll, Qt5Core.dll, Qt5Gui.dll

To perform a fit set up the *.mbs, select it under "'File→open file"' or use command line arguments passing. Moessfit supports all file types which can be opened by "'File→open"' as command line arguments, e.g.

- *.mbs: Moessfit fit model
- *.fld: loads folded data and generates a mbs-file in the same folder to treat the data with an static hamiltonian approach. Multiple fld-files can be opened to be handed to an single mbs-file.
- *.dat/*.ws5: raw data can be folded with Moessfit, a fld-file will be generated in the same folder

Keep in mind the following keys to operate Moessfit:

- "f" ... fit data
- "v" ... view data
- "e" ... error calculation (has to be done manually, as it has N^2 runtime, thus leading to long calculation time for global fits with numerous runs included)
- "u" ... reload model from mbs-file

Keep in mind the following hints

- Report Debugs and personal wishes concerning MoessFit directly to sirko.kammusella@tu-dresden.de
- Defocus any interface panels (by pressing a checkbox for example) to use shortcuts.
- Delete all the appended fitting section in your mbs-file and reload, if you want to start new.
- If a fit model succesfully loads but does show no or wrong plots, varify the fld specified in the RUN block. At least one of them might contain weird data.
- Moessfit supports multi threating, so take advantage of a multi core CPU if available to improve perfomance by the count of cores.
- Restrict the range of a fit parameter p to $[a, b]$ by introducing the function $a + (b - a) \cdot e^{-p^2}$
- "-1" error values indicate, that the fitting minimum is yet not reached.
- Extract spectras by plotting them and then using Menu Plot$\longrightarrow$ ...to clipboard
- A variable global parameter instantly leads to a global fit with the following disadvantages: (a) The Nelder-Mead-simplex algorithm is replaced by a primitve steepest gradient algorithm (slower), (b) multithreating can not be applied as efficient as in a seriell fit.
- Respect the feasibility of error calculations: Consider a global fit of a 7 Parameter model (3 globals) for 20 runs. This leads to a Hesse matrix with 389 different nonzero matrix entries. Every entry refering a global parameter effects the recalculation of 20 moessbauer spectra. Even if there is only one such calculation necessary to determine a second derivative, altogether at least 5240 moessbauer spectra have to be calculated. If these moessbauer spectra contain a MEM, the error calculation can easily exceed 1 hour of calculation time.
- MoessFit takes fld-, mbs-, dat- and ws5-files as command line arguments, so autolink them to the exe.
- Have a look on the example folder representing typical subjects of MoessFit.
- Change single fitting parameter values in the appended fitting section and reload to influence the data directly. More easily take advantage of the play pannel
- Moessbauer deals rather with physical numbers than "mm/s", this concerns easpecially the field gradient und magnetic hyperfine field. For conversation you should use the following values being applied in Moessfit:
  - $\frac{ceQ}{2E_\gamma} = 0.0167 \frac{mm/s}{V/\mathring{A}^2}$ in the quadrupolar interaction $\Delta v_{qp} = \frac{ceQ}{2E_\gamma} \cdot V_{zz} \sqrt{1 + \frac{\eta^2}{3}}$
  - $g_{1/2} = 0.18088$, $g_{3/2} = -0.10327$ and $\frac{c\mu_n}{E_\gamma} = 0.65572 \frac{mm/s}{T}$ for the zeeman splitting $\Delta v_{zeeman} = g_I \frac{c\mu_n}{E_\gamma} B$
- If the number of MEM-Iteration meats the specified maximum, you have to increase the maximum or increase the smoothness of the distribution by reducing λ' (see COMMANDS-MEM)

1 FITPARAMETER

The FITPARAMETER block list all parameters used in THEORY block and describes and classifies them into runspeciffic fitting, global fitting and constant. Only fitting parameters will be effected by the fitting routine. Each line stand for a fitparameter and consists of name, initial value and stepsize. A stepsize of zero marks a parameter to be a constant. If a name begins with "global_", then this fitparameter will be treated as global, else as runspecific. A typical FITPARAMETER block is shown in the next section.

Keep in mind that the decimal place of stepsize is related to the decimal place of calculated errors in the FITDATA output.

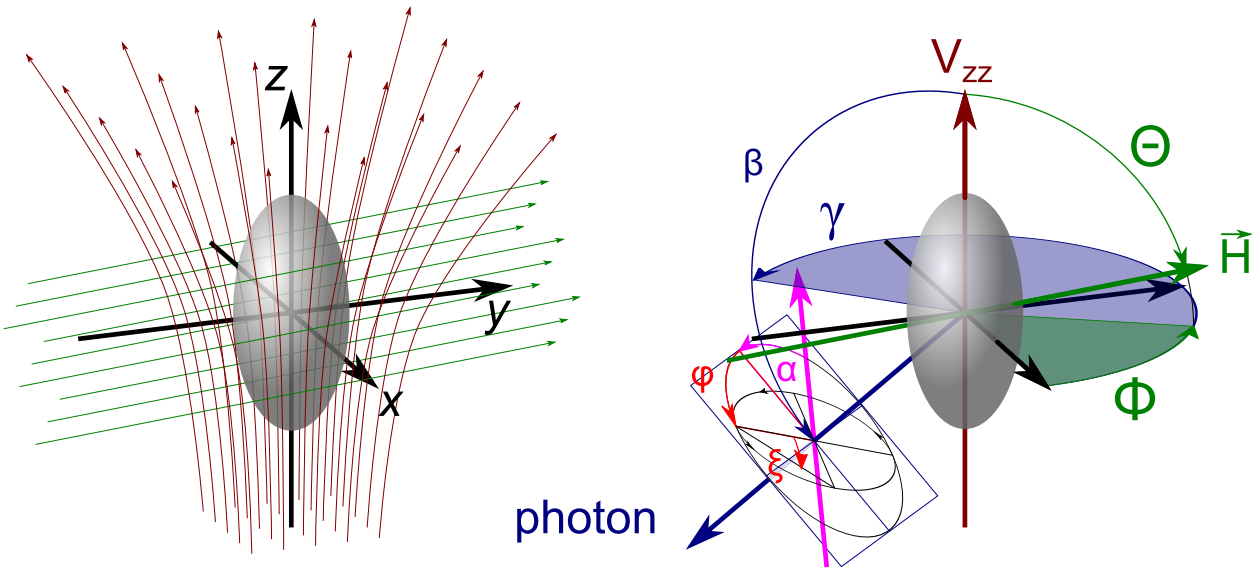

Figure 1: Conventional labeling for the static hamiltonian constructors

2 THEORY

The THEORY block sums up several spectra costruction commands which are seperated by line breaks. These predefined constructors consist of an identifier and a parameter list. So far, the following predefined constructors exist:

- **BL 1**
  Baseline, sets a costant channel content for all channels
  1: I_0 [], channel content, counts

- **SHc 1 2 3 4 5 6 7 8 9 10**
  Static Hamiltonian, crystal
  1: B [T], magnetic hyperfine field
  2: V_{zz} [V/Å^2], principal component of field gradient
  3: η, asymetry parameter
  4: Θ [°], polar angle (B, V_{zz})
  5: Φ [°], azimutal angle (B, V_{zz})
  6: CS [mm/s], center shift
  7: ω [mm/s], line width
  8: A [mm/s], spectral Area
  9: β [°], polar texture angle
  10: γ [°], azimutal texture angle

- **mSSH 1 2 3 4 5 6 7 8 9 10 11 12 13 14**
  magnetized source and Static Hamiltonian sample
  1: B_{spl} [T], magnetic hyperfine field in sample
  2: V_{zz} [V/Å^2], principal component of field gradient
  3: η, asymetry parameter
  4: Θ [°], polar angle (B, V_{zz})
  5: Φ [°], azimutal angle (B, V_{zz})
  6: CS [mm/s], center shift
  7: ω [mm/s], line width
  8: A [mm/s], spectral Area
  9: β [°], polar texture angle
  10: γ [°], azimutal texture angle

11: B_{src} [T], magnetic hyperfine field of source
12: β_{src} [°], polar orientation of source magnetization with respect to the gamma beam, 0°...longitudinal polarization, 90° transverse polarization
13: α [°], azimutal tilting (around gamma beam) of the magnetization with respect to principal axis of the sample system
14: $\epsilon \in [0, 1]$, polarisation, for an source in field typically =1

- **SHp 1 2 3 4 5 6 7 8**
  Static Hamiltonian, powder
  1: B [T], magnetic hyperfine field
  2: V_{zz} [V/Å^2], principal component of field gradient
  3: η, asymetry parameter
  4: Θ [°], polar angle (B, V_{zz})
  5: Φ [°], azimutal angle (B, V_{zz})
  6: CS [mm/s], center shift
  7: ω [mm/s], line width
  8: A [mm/s], spectral Area

- **afmBex 1 2 3 4 5 6 7 8 9 10 11**
  antiferromagnet (uniaxial) in external fields, transverse geometry, powder
  1: B_{afm} [T], internal field of the moments
  2: B_A [T], anisotropy field
  3: B_J [T], exchange field
  4: B_{ex} [T], exchange field
  5: V_{zz} [V/Å^2], principal component of field gradient
  6: Θ [°], polar (B_{afm}, V_{zz})
  7: CS [mm/s], center shift
  8: ω [mm/s], line width
  9: A [mm/s], spectral Area
  10: N, direction number to simulate powder
  11: $N_{V_{zz}}$, V_{zz} sample number, rotated around easy axis

- **fmBex 1 2 3 4 5 6 7 8 9 10 11**
  ferromagnet (uniaxial) in external fields, transverse geometry, powder
  1: B_{afm} [T], internal field of the moments
  2: B_A [T], anisotropy field
  3: B_{ex} [T], exchange field
  4: V_{zz} [V/Å^2], principal component of field gradient
  5: Θ [°], polar (B_{afm}, V_{zz})
  6: CS [mm/s], center shift
  7: ω [mm/s], line width
  8: A [mm/s], spectral Area
  9: N, direction number to simulate powder
  10: $N_{V_{zz}}$, V_{zz} sample number, rotated around easy axis

- **FeCal 1 2 3 4 5 6 7 8**
  iron foil calibration, assuming harmonic drive input, subfolder names should represent monitor voltage in mV
  1: ω_1 [mm/s], outer line width
  2: ω_2 [mm/s], mid line width
  3: ω_3 [mm/s], inner line width
  4: A [], outer peak Area
  5: A [], mid peak Area
  6: A [], inner peak Area
  7: α [(mm/s)/mV], calibration factor of monitor signal

8: CS [mm/s], center shift

- **FeCaltriang 1 2 3 4 5 6 7 8**
  iron foil calibration, assuming triangular drive input, subfolder names should represent monitor voltage in mV
  1: ω_1 [mm/s], outer line width
  2: ω_2 [mm/s], mid line width
  3: ω_3 [mm/s], inner line width
  4: A [], outer peak Area
  5: A [], mid peak Area
  6: A [], inner peak Area
  7: α [(mm/s)/mV], calibration factor of monitor signal
  8: CS [mm/s], center shift

- **CoSrcField 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19**
  Co-source in external field, static hamiltonian approach with line position and intensity output
  1: B [T], magnetic hyperfine field
  2: Θ_{eff}, effectiv field angle with respect to the gamma beam
  3: CS [mm/s], center shift of source
  4: Δv_{11} [mm/s], line postition output
  5: Δv_{12} [mm/s], line postition output
  6: Δv_{13} [mm/s], line postition output
  7: Δv_{14} [mm/s], line postition output
  8: Δv_{21} [mm/s], line postition output
  9: Δv_{22} [mm/s], line postition output
  10: Δv_{23} [mm/s], line postition output
  11: Δv_{24} [mm/s], line postition output
  12: I_{11} [], line intensity output
  13: I_{12} [], line intensity output
  14: I_{13} [], line intensity output
  15: I_{14} [], line intensity output
  16: I_{21} [], line intensity output
  17: I_{22} [], line intensity output
  18: I_{23} [], line intensity output
  19: I_{24} [], line intensity output

- **PseudoVoigt 1 2 3 4**
  Pseudo-Voigt profile, $I(v) = \eta \cdot \frac{1}{\pi\omega} \frac{1}{1+\left(\frac{v-CS}{w}\right)^2} + (1-\eta) \cdot \frac{\sqrt{\ln 2}}{\sqrt{\pi}\omega} e^{-\ln 2\left(\frac{v-CS}{\omega}\right)^2}$
  1: CS [mm/s], center shift
  2: ω [mm/s], line width
  3: η, lorentzian weight
  4: A [mm/s], spectral Area

Every parameter is of one of the following types:

- # ... inserts the pure object value
- MEM[#,#,#,#] ... MEM[storage,min,max,steps] arranges Maximum Entropy fitted distribution onto this parameter. For construction of MEM-subspectra the MEM-parameter is temporary stored in storage. This MEM can be folded with MEM of other arguments
- DISTR[#,#,#,#,#] ... DISTR[storage,min,max,steps,weightfunction] stores all values between min and max one after the other into storage and sets them for the parameter simultanously, weights the resulting spectra with the weightfunction. This can be folded with DISTR of other arguments.

- DISTR[#,#,#,#,#,#] ... DISTR[storage,min,max,steps,weightfunction,parameterfunction] works the same as DISTR, but additionally calculates the parameter value by parameterfunction instead of inheriting directly from storage. The storage variable in that case acts as a parametrisation.

2.0.1 Arguments

In this notation # stands for one of the following objects:

- a fitparameter name defined in FITPARAMETER block
- a function name defined in FUNCTIONS block
- map, i.e. you state map0 or map1 or ... to refere to the runspecific map defined in RUN Block
- a constant number

Function objects are updated just before calculation, after all parameters are deployed.

2.0.2 Minimum example

A simple iron sextett could be describt the following way

##########
FITPARAMETER
Bhyp 33.3 0
omega 0.12 0.01
CS 0.2 0.1
I0 10000 100
A 2000 100
##########
THEORY
BL I0
SHp Bhyp 0 0 0 0 CS omega A
##########

3 RUN

The RUN block lists all subfolder to be taken into account for the fitting and should contain all runs stated in the PLOT block. Every line of the RUN block consists of a float, representing the run specific parameter "rsp" (e.g. temperature/field/voltage) being the name of the subfolder. Runspecific maps, if there are some, follow space seperate (type the name of the referring fit parameter/function/map).

##########
RUN
 # # #...
 # # #...
"C:\fullpath\file.fld" # # # ...
"C:\fullpath\file.fld" # [#,#] ... [#,#] # # ...
...
##########

"C:\fullpath\file.fld": optional path if no subfolder exists
#: rsp
[#,#] ... [#,#]: optional fit range(s)
 # # ...: optional maps

Alternatively to the subfolder-refering first number (rsp), it is possible to state a full path in quotation marks followed by an userdefined number. This number will serve as the run specific parameter for the plotting block or different spectrum constructors. Add optional fitting ranges in brackets (unit: mm/s) after the runspecific number to make the fit operating only onto userdefined parts.

4 FUNCTIONS

The FUNCTIONS block has to list all user defined functions being used in the THEORY block. The block is of the following form:

##########
FUNCTIONS
functionname1 = ...
functionname2 = ...
...
##########
Define a function on the "..."-postion. The function parser is taken from http://warp.povusers.org/FunctionParser/, so please refer to the page concerning syntax questions. A function can contain objects of every kind (fitparameters, other function from FUNCTIONS block, map paremeters ("map#")) and the run specific parameter (i.e. temperature/field/voltage) ("rsp"), which is stated within the RUN block.

The following functions are implemented additionally (simply type the left hand name and the arguments as stated above):

1. $AbsDeb(T,\Theta_D) = e^{-\frac{\hbar^2 k_\gamma^2}{2M}\frac{6T^2}{k_B\Theta_D^3}\int_0^{\Theta_D/T}\frac{xdx}{e^x-1}}$ ... Debye Waller factor of the phonon Debye model, $[T,\Theta_D]$=K,K
2. $QuadDop(T,\Theta_D,M_{eff}) = -\frac{9R}{2N_A M_{eff} c}\frac{T^4}{\Theta_D^3}\int_0^{\Theta_D/T}\frac{x^3dx}{e^x-1}$ ... quadratic Doppler effect in debye approximation of heat capacity; $[T,\Theta_D$,M]=K,K,u
3. $SwaveSFD(T,T_c,\lambda_0^{-2},\Delta_0) = \lambda_0^{-2}\left(1+\int_\Delta^\infty \underbrace{-\frac{e^{E/k_bT}}{k_bT(e^{E/k_bT}+1)}}_{\frac{\partial f}{\partial E}}\cdot\frac{E}{\sqrt{E^2-\Delta^2}}\right)$ ... super fluid density in the s-wave model for μSR fitting purpose with $\Delta = \Delta_0\cdot\tanh(1.82\cdot(1.018\cdot(\frac{T_c}{T}-1)^{0.51}))$; $[T,T_c,\lambda_0^{-2},\Delta_0]$=K,K,$\mu$m^{-2},eV
4. $erf(x) = \frac{2}{\sqrt{\pi}}\int_0^x e^{-t^2}dt$ ... gaussian error function
5. $Hc2WHH(T,T_C,dH_{c2}/dt\mid_{t=1},\alpha,\lambda_{SO})$ ... Temperature dependence of the 2nd critical field of a superconductor within the WHH model [Col86]. $[T,T_C,dH_{c2}/dT\mid_{t=1},\alpha,\lambda_{SO})]$=K,K,G,1. The model has the following restrictions
   - transition is of 2nd order
   - spin orbit scattering frequency λ_{SO} is much less than normal scattering frequency
   - strong coupling is neglected
   - Fermi surface anisotropy is neglected
   - there is normal state spin (Pauli) paramagnetism
   - there is mixed state spin orbit scattering

$$\alpha ... \ Maki\ paraeter \tag{1}$$

$$\lambda_{SO} = \frac{2\hbar}{3\pi k_B T_C \tau_{SO}} ... \ Spin\ orbit\ scattering\ frequency \tag{2}$$

$$t = \frac{T}{T_C} \tag{3}$$

$$\bar{h} = \frac{4}{\pi^2}\cdot h^* \tag{4}$$

$$h^* = \frac{H_{C2}}{-(H_{C2}/dt)_{t=1}} \tag{5}$$

$$\ln\left(\frac{1}{t}\right) = \sum_{n=-\infty}^{n=+\infty}\left(\frac{1}{|2n+1|} - \frac{1}{|2n+1|+\bar{h}/t+\frac{(\alpha\bar{h}/t)^2}{|2n+1|+(\bar{h}+\lambda_{SO})/t}}\right) ...implicite\ WHH-equation \tag{6}$$

6. $NGLsigmaquad(B_{ex}, B_{C2}, \lambda)$ ... Numeric Ginzburg-Landau theory [Bra03] to describe the superconducting damping rate σ_{SC} in an quadratic flux line lattice with criticical field B_{C2} [T], applied field B_{ex} and penetration depth λ. $[B_{ex}, B_{C2}, \lambda]$=T,T,$\mu$m

7. $NGLsigmatri(B_{ex}, B_{C2}, \lambda)$ ... Numeric Ginzburg-Landau theory [Bra03] to describe the superconducting damping rate σ_{SC} in an triangular flux line lattice with criticical field B_{C2} [T], applied field B_{ex} and penetration depth λ. $[B_{ex}, B_{C2}, \lambda]$=T,T,$\mu$m

5 COMMANDS

- MaxIter # ... maximum Iterations # for one fitting procedure. (by default set to: 100)
- I0 # ... parameter # is to be overwritten with the automically determined runspecific baseline
- A # ... parameter # is to be overwritten with the automically determined runspecific spectral area
- ConvergenceCriterium # ... sets minimum improvement per step for fitting algorithms. Fitting stops if improvement is insufficient. (by default set to 0.01)
- MEM # # # # #... maximum steps, stepsize, tollerance, maxlambda' λ_2' and the baseline I_{02} concerning this lambda; by default set to 100, 0.012, 0.1 and 1. The run depedendend λ' is calculated by

$$\lambda' = \frac{\sqrt{\frac{I_{02}}{I_0}}\lambda_2'}{1 + \left(\sqrt{\frac{I_{02}}{I_0}} - 1\right)\lambda_2'} \tag{7}$$

  with regard of the rundepend baseline I_0 to achieve run independend smoothness of the MEM-distributions. λ_2' and I_{02} respectivly should be taken from the run with the highes statistics. That means first MEM-fit these runs with $\lambda_2' = 1$ to gain the maximum λ the algorithm may achieve. Spectra with lower channel content should achieve an equal smoothness more easily.

- TransmissionIntegral # # # #... arguments: effectiv thickness, source linewidth [mm/s], absorber linewidth [mm/s], resonant fraction; calculates the full transition integral for the specified theory, useful to fit thick samples, in this mode you have the set the pure sample linewidths as linewidth, i.e. substract the source line width from the sample line width. The total spectral area argument should be set constant, as the actual area is varied by the background fraction and effective thickness only. Keep care that the spectrum has exact symmetric velocity spacing (7 velocity digits at least), otherwise a slow naiv convolution will be applied.

6 PLOT

The PLOT block defines the runs to be plotted and the range. For several runs, the channel contents will be normalized to baseline. Plotting is called by pressing "p". The typical PLOT block looks like the following:

##########
PLOT
runs 4,2 103,5 290,2
range -2.5 3.5 0.85 1.04
##########

7 Fitting anything

For general fitting purpose moessbauer may serve as proper tool. Data to be fit is a *.fld file with three tab seperated columns: xdata, ydata, yerror. To fit this data you have to type "fit # #" as the constructor in the THEORY block. First argument is used as the x buffer, i.e. xdata will be loaded into this parameter and a fitfunction might work upon this parameter. This fitfunction is stated as second argument. The x buffer should be declared as a constant. The fitfunction is as usual declared in FUNCTIONS block.

To fit parables with equal curvature a and runspecific offset (2.3, 4.7, -4) to three data sets e.g., the concerning *.mbs file could look like the following:

FITPARAMETER
global_a 0.5 0.1
b 3 0.1
x 0 0
##########
THEORY
fit x fitfunc
##########
RUN
"C:\path\to\file1\data1.fld" 2.3
"C:\path\to\file2\data2.fld" 4.7
"C:\path\to\file3\data3.fld" -4
##########
FUNCTIONS
*fitfunc = global_a*x*x+b*x + rsp*

References

[Bra03] Brandt, Ernst H.: Properties of the ideal Ginzburg-Landau vortex lattice. 68 (2003), Aug, 054506. http://dx.doi.org/10.1103/PhysRevB.68.054506. – DOI 10.1103/PhysRevB.68.054506

[Col86] Collings, EW: *Applied Superconductivity, Metallurgy, and Physics of Titanium Alloys: Volume 1: Fundamentals*. Bd. 1. Springer, 1986

[SW12] Suter, A. ; Wojek, B.M.: Musrfit: A Free Platform-Independent Framework for Î$\frac{1}{4}$SR Data Analysis. In: *Physics Procedia* 30 (2012), Nr. 0, 69 - 73. http://dx.doi.org/http://dx.doi.org/10.1016/j.phpro.2012.04.042. – DOI http://dx.doi.org/10.1016/j.phpro.2012.04.042. – ISSN 1875–3892. – 12th International Conference on Muon Spin Rotation, Relaxation and Resonance (Î$\frac{1}{4}$SR2011)