

Numeric calculation of antiferromagnetic resonance frequencies for the noncollinear antiferromagnet

V.Glazkov,^{1,2,*} T.Soldatov,^{1,2} and Yu.Krasnikova^{1,2}

¹*P.Kapitza Institute for Physical Problems RAS, Kosygin str.2, 119334 Moscow, Russia*

²*Moscow Institute of Physics and Technology, 141700 Dolgoprudny, Russia*

(Dated: June 29, 2016)

We present an algorithm for the numeric calculation of antiferromagnetic resonance frequencies for the non-collinear antiferromagnets of general type. This algorithm uses general exchange symmetry approach [1] and is applicable for description of low-energy dynamics of an arbitrary noncollinear spin structure in weak fields. Algorithm is implemented as a MatLab and C++ program codes, which are available as an electronic supplementary material. Program codes are tested against some representative analytically solvable cases.

PACS numbers: 75.50.Ee, 76.50.+g

Keywords: antiferromagnetic resonance, non-collinear antiferromagnets

CONTENTS

I	Main matter	2
I.	Introduction	2
II.	Brief basics of exchange symmetry approach and derivation of dynamics equations	2
III.	Solving dynamics equations numerically	5
A.	Search for equilibrium	5
B.	Solving dynamics equation, MatLab implementation details	5
C.	Solving dynamics equation, C++ implementation details	6
D.	Application to the test examples	8
IV.	Conclusions (main matter)	8
II	Supplementary materials: detailed code description, output file formats and detailed tests	8
V.	Source files location	8
VI.	MatLab implementation	8
A.	Editable fragments of the source file	8
B.	Output files format	9
VII.	C++ implementation	9
A.	Source files	9
B.	INI-file format	10
C.	Error handling	11
D.	Output files format	11

* glazkov@kapitza.ras.ru; <http://www.kapitza.ras.ru/rgroups/esrgroup/>

VIII. Detailed test protocols	12
A. Analytically solvable models used as a test cases	12
B. Case of CsNiCl ₃	13
C. Case of Mn ₃ Al ₂ Ge ₃ O ₁₂	17
D. Case of LiCu ₂ O ₂	20
IX. Conclusions (supplementary material)	23
References	23

Part I

Main matter

I. INTRODUCTION

Antiferromagnetic ordering is observed at low temperatures in a vast amount of crystals. In the simplest case magnetic ions can be grouped in two sublattices with antiparallel average spins yielding the collinear antiferromagnetic structure. However, such collinear structures do not cover all possible types of antiferromagnetic order: one can easily imagine magnets with more than two sublattices or a helicoidal structures, which can not be deduced to a finite number of sublattices at all. Numerous examples of such systems are known, e.g.: three-sublattice “triangular” antiferromagnetic order CsNiCl₃ and RbNiCl₃ [2, 3], 12-sublattices ordering in an Mn₃Al₂Ge₃O₁₂ garnet [4, 5], spiral ordering in LiCu₂O₂ [6–8], complicated multi- \vec{k} structure in strongly frustrated pyrochlore magnet Gd₂Ti₂O₇ [9].

Electron spin resonance (antiferromagnetic resonance, AFMR) is a powerful and sensitive tool to study low-energy dynamics of the magnetically ordered system. By exciting uniform oscillations (i.e. $k = 0$ spin waves) of the ordered spin structure one can investigate its properties: orientation of the ordered structure with respect to the crystal, strength of the anisotropic interactions fixing this orientation, various spin-reorientation transitions. Due to high energy resolution of microwave AFMR spectrometry spin waves spectrum details at $\vec{k} = 0$ can be quite routinely determined with accuracy up to 5 mkeV (corresponding to the resolution of 1 GHz), thus strongly complimenting powerful inelastic magnetic neutron scattering techniques at low energies.

However, interpretation of the antiferromagnetic resonance data for complicated magnetic structures is sometimes difficult. Antiferromagnetic resonance frequencies for a collinear structure can be calculated relatively easily in a two-sublattice model [10]. Similar calculations for a noncollinear magnets are much less general: many-sublattice model calculations using a mean-field theory approach are very cumbersome [11], standard (Holstein-Primakoff like) spin-wave theory approach to many-sublattice antiferromagnets is also complicated (e.g., [12, 13]). Moreover, analytical solution of these equations is usually out of the question for general mutual orientation of the magnetic field and sublattices magnetizations. Numeric calculations of spin waves spectra are also known, see e.g. SpinW library by S.Tóth [14], but they rely on strongly model dependent microscopic hamiltonian.

Some of these difficulties can be overcome by using an exchange symmetry approach developed in [1]. This approach allows to build up general hydrodynamic description of low-energy dynamics of an antiferromagnet. It was successfully applied for various magnetic systems [3, 5, 8, 16–18]. However, analytical solution for $f(H)$ dependency (which is the characteristic observable in antiferromagnetic resonance experiment) remains complicated, if possible at all, for arbitrary direction of magnetic field.

In the present manuscript we describe numeric approach to the solution of dynamic equations in the arbitrary case. The developed algorithm is implemented in a program codes, available free of charge on the authors web-page (<http://www.kapitza.ras.ru/rgroups/esrgroup/>) [15].

II. BRIEF BASICS OF EXCHANGE SYMMETRY APPROACH AND DERIVATION OF DYNAMICS EQUATIONS

First, we briefly recall necessary equations of exchange symmetry theory [1] to be used in our calculations. The main limitation of this theory is that distortions of the ordered spin structure are small, which limits its applicability to the low fields $H \ll H_{ex} \simeq J/(g\mu_B)$. In particular, this limitation excludes from consideration various phase transitions with complete restructuring of the order parameter (collinear-noncollinear transitions, various magnetization plateau phases etc.). Under this assumption any noncollinear magnetic structure can be described by three unitary orthogonal

vectors $\vec{l}_{1,2,3}$ (e.g., planar structure with a wavevector \vec{k} can be described as $\vec{S}(\vec{r}) = \vec{l}_1 \cos(\vec{k}\vec{r}) + \vec{l}_2 \sin(\vec{k}\vec{r})$, with $\vec{l}_3 = \vec{l}_1 \times \vec{l}_2$). All static properties and low energy dynamic of this structure can be described by its Lagrangian with Lagrangian density (we use here notations of Refs. [8, 19])

$$\mathcal{L} = \sum_i \frac{I_i}{2} \left(\dot{\vec{l}}_i + \gamma [\vec{l}_i \times \vec{H}] \right)^2 - U_A(\{\vec{l}_i\}) \quad (1)$$

here γ is a free electron gyromagnetic ratio and U_A is the energy of anisotropy. Constants $I_i \geq 0$ are related to susceptibilities as $\vec{M} = \frac{\partial \mathcal{L}}{\partial \vec{H}}$: magnetic susceptibilities for the field applied along i-th vector are $\chi_1 = \gamma^2(I_2 + I_3)$, $\chi_2 = \gamma^2(I_1 + I_3)$, $\chi_3 = \gamma^2(I_1 + I_2)$.

Anisotropy energy should be invariant under crystal symmetry transformation, its exact form depends on the symmetry of the particular crystal and on the exchange symmetry of the ordered phase, relationship between I_i constants is also fixed by symmetry of the susceptibilities tensor for a given spin structure. Some examples for the known analytically solvable cases are given in Sec.VIII A. Note, that I_i constants and exact form of the anisotropy energy are the only parameters of this approach. Once they are deduced only the formal operations remains.

Firstly, static equilibrium position $\vec{l}_i^{(0)}$ have to be found by minimization of potential energy density

$$\Pi = - \sum_i \frac{I_i}{2} \gamma^2 [\vec{l}_i \times \vec{H}]^2 + U_A(\{\vec{l}_i\}) \quad (2)$$

Secondly, frequencies of small oscillations near equilibrium have to be deduced. We suppose here that these oscillations are parameterized by some three non-degenerate variables $\{\phi_\alpha\}$, e.g. Euler angles or other suitable variables. For the sake of simplicity we take that all of $\phi_\alpha = 0$ at equilibrium position. Potential energy has a quadratic minimum at the equilibrium, thus when looking for small oscillations we can replace potential energy by its quadratic expansion. This substitution explicitly excludes possible problems of a numeric algorithm due to the finite accuracy of minimum determination. Lagrangian density is then

$$\mathcal{L} = \sum_i \frac{I_i}{2} \left(\dot{\vec{l}}_i \right)^2 + \gamma \sum_i I_i \left(\dot{\vec{l}}_i \cdot [\vec{l}_i \times \vec{H}] \right) - \frac{1}{2} \sum_{\beta, \delta} \left(\frac{\partial^2 \Pi}{\partial \phi_\beta \partial \phi_\delta} \right)_0 \phi_\beta \phi_\delta \quad (3)$$

here $(\dots)_0$ index means that derivative is calculated at equilibrium position.

To obtain dynamics equations linear in ϕ_α or its time derivatives, \vec{l}_i have to be expanded up to second order in ϕ_α :

$$\begin{aligned} \vec{l}_i &= \vec{l}_i^{(0)} + \sum_\beta \left(\frac{\partial \vec{l}_i}{\partial \phi_\beta} \right)_0 \phi_\beta + \frac{1}{2} \sum_{\beta, \delta} \left(\frac{\partial^2 \vec{l}_i}{\partial \phi_\beta \partial \phi_\delta} \right)_0 \phi_\beta \phi_\delta \\ \dot{\vec{l}}_i &= \sum_\beta \left(\frac{\partial \dot{\vec{l}}_i}{\partial \phi_\beta} \right)_0 \dot{\phi}_\beta + \sum_{\beta, \delta} \left(\frac{\partial^2 \dot{\vec{l}}_i}{\partial \phi_\beta \partial \phi_\delta} \right)_0 \phi_\beta \dot{\phi}_\delta, \end{aligned}$$

then with linear over ϕ_α accuracy

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial \phi_\alpha} &= \gamma \sum_{i, \beta \neq \alpha} \left(\left(\frac{\partial \vec{l}_i}{\partial \phi_\beta} \right)_0 \cdot \left[\left(\frac{\partial \vec{l}_i}{\partial \phi_\alpha} \right)_0 \times \vec{H} \right] \right) \dot{\phi}_\beta + \\ &+ \gamma \sum_{i, \beta} I_i \dot{\phi}_\beta \left(\frac{\partial^2 \vec{l}_i}{\partial \phi_\alpha \partial \phi_\beta} \right)_0 \cdot [\vec{l}_i^{(0)} \times \vec{H}] - \sum_\beta \left(\frac{\partial^2 \Pi}{\partial \phi_\alpha \partial \phi_\beta} \right)_0 \phi_\beta \end{aligned}$$

and so forth.

Variation of the action results in three Euler-Lagrange equations

$$\frac{d}{dt} \frac{\partial \mathcal{L}}{\partial \dot{\phi}_\alpha} - \frac{\partial \mathcal{L}}{\partial \phi_\alpha} = 0 \quad (4)$$

By summing up all terms and by substituting uniform harmonic oscillations $\phi_\beta = \phi_\beta^{(0)} e^{i\omega t}$ we obtain equations on oscillations amplitudes $\phi_\beta^{(0)}$. Required degeneracy of these equations results in the condition $\det \mathcal{M} = 0$ where matrix \mathcal{M} of the linear equations is defined as

$$\begin{aligned} \mathcal{M}_{\alpha\beta} = & -\omega^2 \sum_i I_i \left(\left(\frac{\partial \vec{l}_i}{\partial \phi_\alpha} \right)_0 \cdot \left(\frac{\partial \vec{l}_i}{\partial \phi_\beta} \right)_0 \right) + \\ & + 2i\omega\gamma \sum_i I_i \left(\left(\frac{\partial \vec{l}_i}{\partial \phi_\alpha} \right)_0 \cdot \left[\left(\frac{\partial \vec{l}_i}{\partial \phi_\beta} \right)_0 \times \vec{H} \right] \right) + \left(\frac{\partial^2 \Pi}{\partial \phi_\alpha \partial \phi_\beta} \right)_0 \end{aligned} \quad (5)$$

The equation $\det \mathcal{M} = 0$ results in real cubic equation for ω^2 , all complex coefficients will sum to zero. Solution of this equation yields eigenfrequencies of small oscillations we sought for.

Experimental observation of these small oscillations in standard magnetic resonance experiment is, in fact, observation of the absorption of microwave radiation of certain polarization. Thus, information about oscillation of magnetization $\vec{m}(t) = \vec{m} e^{i\omega t}$ is important as well. It can be calculated straightforwardly as $\vec{M} = \frac{\partial \mathcal{L}}{\partial \vec{H}} = \vec{M}_0 + \vec{m} e^{i\omega t}$, oscillating magnetization vector is

$$\begin{aligned} \vec{m} = & i\gamma\omega \sum_{i,\beta} I_i \left[\left(\frac{\partial \vec{l}_i}{\partial \phi_\beta} \right)_0 \times \vec{l}_i^{(0)} \right] \phi_\beta^{(0)} - \\ & - \gamma^2 \sum_{i,\beta} I_i \left(\left(\left(\frac{\partial \vec{l}_i}{\partial \phi_\beta} \right)_0 \cdot \vec{H} \right) \vec{l}_i^{(0)} + \left(\vec{l}_i^{(0)} \cdot \vec{H} \right) \left(\frac{\partial \vec{l}_i}{\partial \phi_\beta} \right)_0 \right) \phi_\beta^{(0)} \end{aligned} \quad (6)$$

complex form of \vec{m} describes circular or elliptical precession of magnetization: $\vec{m}(t) = (\vec{u} + i\vec{v}) e^{i\omega t}$ means that real magnetization is $\vec{u} \cos \omega t - \vec{v} \sin \omega t$. Average square of longitudinal and transverse components of the oscillating magnetization can be used as a simple indicator of excitation conditions

$$\langle \vec{m}^2 \rangle = \frac{1}{2} (\vec{u}^2 + \vec{v}^2) \quad (7)$$

$$\langle m_{\parallel}^2 \rangle = \frac{1}{2} \left((\vec{u} \cdot \vec{n})^2 + (\vec{v} \cdot \vec{n})^2 \right) \quad (8)$$

$$\langle m_{\perp}^2 \rangle = \langle \vec{m}^2 \rangle - \langle m_{\parallel}^2 \rangle \quad (9)$$

here \vec{n} is a unitary vector in the applied field direction. Being interested in the polarization of oscillating magnetization only we will norm its square averaged (if non zero) to unity: $\langle \vec{m}^2 \rangle = 1$.

Determination of the initial guesses for the model parameters is case-dependent. We will note here, that equation $\det \mathcal{M} = 0$ allows to scale all parameters of \mathcal{M} arbitrary. This means, that (unless one is particularly interested to reproduce both static and dynamical properties without scaling coefficients) one of the coefficients (one of I_i constants or one of the coefficients in anisotropy energy expansion) can be set to unity for convenience. Secondly, the \mathcal{M} matrix simplifies for zero-field problem (its complex part vanishes) which could help to find zero-field gaps in AFMR spectrum. Another possible simplification is softening of the AFMR modes, which commonly appears at spin reorientation transition. In this case $\omega = 0$ and $\det \mathcal{M} = 0$ reduces to $\det \left(\frac{\partial^2 \Pi}{\partial \phi_\alpha \partial \phi_\beta} \right)_0 = 0$. Finally, at high fields one of the AFMR modes is field independent and its frequency can be calculated [20], while field-dependent modes linear asymptotes are (we assume that $\chi_3 = \gamma^2(I_1 + I_2)$ is the largest susceptibility)

$$\begin{aligned} \omega_2 &= \gamma H \\ \omega_3 &= \sqrt{1 - 2 \frac{I_3(I_1 + I_2)}{(I_1 + I_3)(I_2 + I_3)}} \gamma H = \sqrt{1 - \frac{(\chi_1 + \chi_2 - \chi_3)\chi_3}{\chi_1\chi_2}} \gamma H \end{aligned}$$

In the limiting case of $I_1 = I_2$ ($\chi_1 = \chi_2 < \chi_3$) $\omega_3 = \frac{I_1 - I_3}{I_1 + I_3} \gamma H = \frac{\chi_3 - \chi_1}{\chi_1} \gamma H$. Alternatively, I_i constants can be deduced from the susceptibility measurements.

III. SOLVING DYNAMICS EQUATIONS NUMERICALLY

A. Search for equilibrium

We define orientation of $\{\vec{l}_i\}$ vectors by Euler angles θ , ϕ and ψ . Minimization can be performed with any suitable standard numeric minimization procedure. However please note that numeric procedures always look for local minimum. Thus to find a global minimum one have to perform preliminary search for a starting approximation with minimal potential energy Π over some grid in the Euler angles space. On the other hand, it could be of interest to follow a particular local minimum evolution with field, which allows to model response from different magnetic domains. MatLab implementation uses global minimum search only, C++ implementation allows to follow local minimum on user choice.

From this point on we assume that desired equilibrium position $\{\vec{l}_i^{(0)}\}$ is found. Dynamics equations are obtained by varying action $S = \int \mathcal{L} dV dt$ and they can be written down in any suitable variables. Euler angles are, generally, not the best choice for dynamics equation as they suffer from “gimbal lock” problem: one of the degrees of freedom will be lost if at some moment $\vec{l}_3 \parallel Z$. To avoid this problem we used two approaches for calculation of eigenfrequencies: (i) to recalculate our problem to the frame of reference which is definitely free from the “gimbal lock”, or (ii) to use other set of variables for dynamics equations. First approach was implemented in MatLab code, second approach was implemented in C++ code.

B. Solving dynamics equation, MatLab implementation details

First approach was applied in MatLab environment using the Symbolic Math Toolbox, as it provides functions for manipulating symbolic math equations and lets analytically perform differentiation, simplification and transforms. All these opportunities allow to consider general form of the anisotropy energy U_A without any simplifications. GlobalSearch class is used as well for obtaining global minimum point of potential energy Π and finding equilibrium position $\{\vec{l}_i^{(0)}\}$.

Firstly, we rotate laboratory reference frame in such a way that $\theta = \phi = \psi = \pi/6$ for equilibrium position of $\{\vec{l}_i\}$ vectors. The choice of angle equal to $\pi/6$ is fairly arbitrary, it is chosen simply to exclude “gimbal lock” problem. Herewith recalculation of vector components of the external magnetic field and transformation of the anisotropy energy to new coordinates is needed. If $A = \{a_{\alpha\beta}\}$ is the matrix of this rotation, $B = A^{-1} = \{b_{\alpha\beta}\}$ is the inverse matrix, then in new frame of references

$$H'_\alpha = \sum_{\beta} a_{\alpha\beta} H_\beta \quad (10)$$

$$\tilde{U}_A(\{l_i^\alpha\}) = U_A\left(\left\{\sum_{\beta} b_{\alpha\beta} l_i^\beta\right\}\right) \quad (11)$$

Here $\vec{H} = \{H_\alpha\}$ and $\vec{H}' = \{H'_\alpha\}$ are vectors of the external magnetic field in the basic and transformed frames of references correspondingly, $\tilde{U}_A(\{l_i^\alpha\})$ is the anisotropy energy written in new frame of references.

Secondly, we use parametrization of Euler angles for description of small oscillations near the equilibrium position in transformed frame of references, because in such case “gimbal lock” problem is avoided. As magnetic vectors components $\{l_i^\alpha\}$ are known functions of θ , ϕ , ψ parameters, there are no any problems to obtain the values of first derivatives of $\{l_i^\alpha\}$ vectors and the values of first and second derivatives of potential energy Π at $\{\vec{l}_i^{(0)}\}$ position. These values are used for calculations of oscillations eigenfrequencies from the equation $\det \mathcal{M} = 0$ according to Eqn.(5).

Complete algorithm is divided into few steps:

1. We start from specified start field $H = H_{start}$ applied in the specified direction.
2. We look for global minimum of potential energy Π and find a new equilibrium position at field H . Information on equilibrium position (Euler angles, potential energy at equilibrium, projections of $\{\vec{l}_i^{(0)}\}$ vectors on the field direction, longitudinal and transverse susceptibilities) is saved.

3. Components of vector \vec{H}' (Eqn.10) and anisotropy energy \tilde{U}_A (Eqn.11) in transformed frame of references are obtained.
4. Matrix \mathcal{M} (Eqn.5) is calculated and $\det\mathcal{M} = 0$ equation is solved for eigenfrequencies. Results are saved.
5. Eigenvectors and average values of projections of oscillating magnetization vector along and transverse to external magnetic field for all oscillation modes are found and saved.
6. Field is increased by specified increment H_{step} . If the field does not reach its goal value H_{stop} we continue with Step 2.

All input parameters including anisotropy energy function U_A in general case, χ_i and γ coefficients, magnetic field direction, variation boundaries, increment of the value of magnetic field are specified in MatLab script, available at [15]. Calculation results are saved in three files correspondingly with static properties (equilibrium position, energy at equilibrium, projections of $\{\vec{l}_i^{(0)}\}$ vectors on the field direction, longitudinal and transverse susceptibilities), oscillation eigenfrequencies and eigenvectors together with average projections of oscillating magnetization vector along and transverse to the magnetic field. Format of these files is described in details in supplementary materials (see below).

C. Solving dynamics equation, C++ implementation details

MatLab environment allows big flexibility and allow to avoid lot of routine operations. However, it requires commercial software and, being an interpreter, is somewhat slower then a properly compiled program. Thus we propose an alternative implementation in C++ language along with flexible executable program. C++ codes and compiled Win32 executable are available at [15].

For the sake of flexibility we will consider only quadratic terms in anisotropy energy

$$U_A = \sum'_{i,j,\alpha,\beta} a_{ij}^{\alpha,\beta} l_i^\alpha l_j^\beta \quad (12)$$

here \sum' sign means that each $l_i^\alpha l_j^\beta$ combination is counted only once during summation. Higher orders of anisotropy can be included in the program code in a straightforward way, if necessary. This restriction allows to read all $a_{ij}^{\alpha,\beta}$ coefficients from easily editable plain text ini-file (ini-file format is described in supplementary materials below) and to simplify all derivatives calculations for minimum search routine and for dynamics equation derivation. E.g.,

$$\frac{\partial U_A}{\partial x} = \sum'_{i,j,\alpha,\beta} a_{ij}^{\alpha,\beta} \left(\frac{\partial l_i^\alpha}{\partial x} l_j^\beta + l_i^\alpha \frac{\partial l_j^\beta}{\partial x} \right) \quad (13)$$

here x is some variable of choice.

Numerical Recipes [21] FRPRMN routine is used to find an equilibrium position. We continue calculations in the same frame of references attached to the crystal, but small oscillations near the equilibrium are described as a small rotations of $\{\vec{l}_i\}$ vectors parameterized by a vector of small rotations $\vec{\phi} = (\phi_x, \phi_y, \phi_z)$. Length of this vector is rotation angle and its direction defines rotation axis, at equilibrium position $\phi = 0$. Up to quadratic terms in ϕ transformation of $\{\vec{l}_i\}$ can be described as:

$$\vec{l}_i = \vec{l}_i^{(0)} + [\vec{\phi} \times \vec{l}_i^{(0)}] + \frac{1}{2} [\vec{\phi} \times [\vec{\phi} \times \vec{l}_i^{(0)}]] + \underline{Q}(\phi^3) \quad (14)$$

This parametrization is free from ‘‘gimbal lock’’. Note that there are nonzero second order derivatives $\frac{\partial^2 \vec{l}_i}{\partial \phi_\alpha \partial \phi_\beta}$ which have to be taken into account when calculating Hessian matrix $\frac{\partial^2 \Pi}{\partial \phi_\alpha \partial \phi_\beta}$. This allows to complete calculations of oscillations eigenfrequencies.

Once eigenfrequencies are known, complex oscillation vectors $\vec{\phi}$ are found as zero-eigenvalue eigenvectors of \mathcal{M} matrix using standard JACOBI procedure from Numerical Recipes [21]. This allows to compute complex oscillating magnetization vector \vec{m} (see Eqn.(6)) and its average projections on the field direction and on the direction transverse to the field.

Complete algorithm looks as follows:

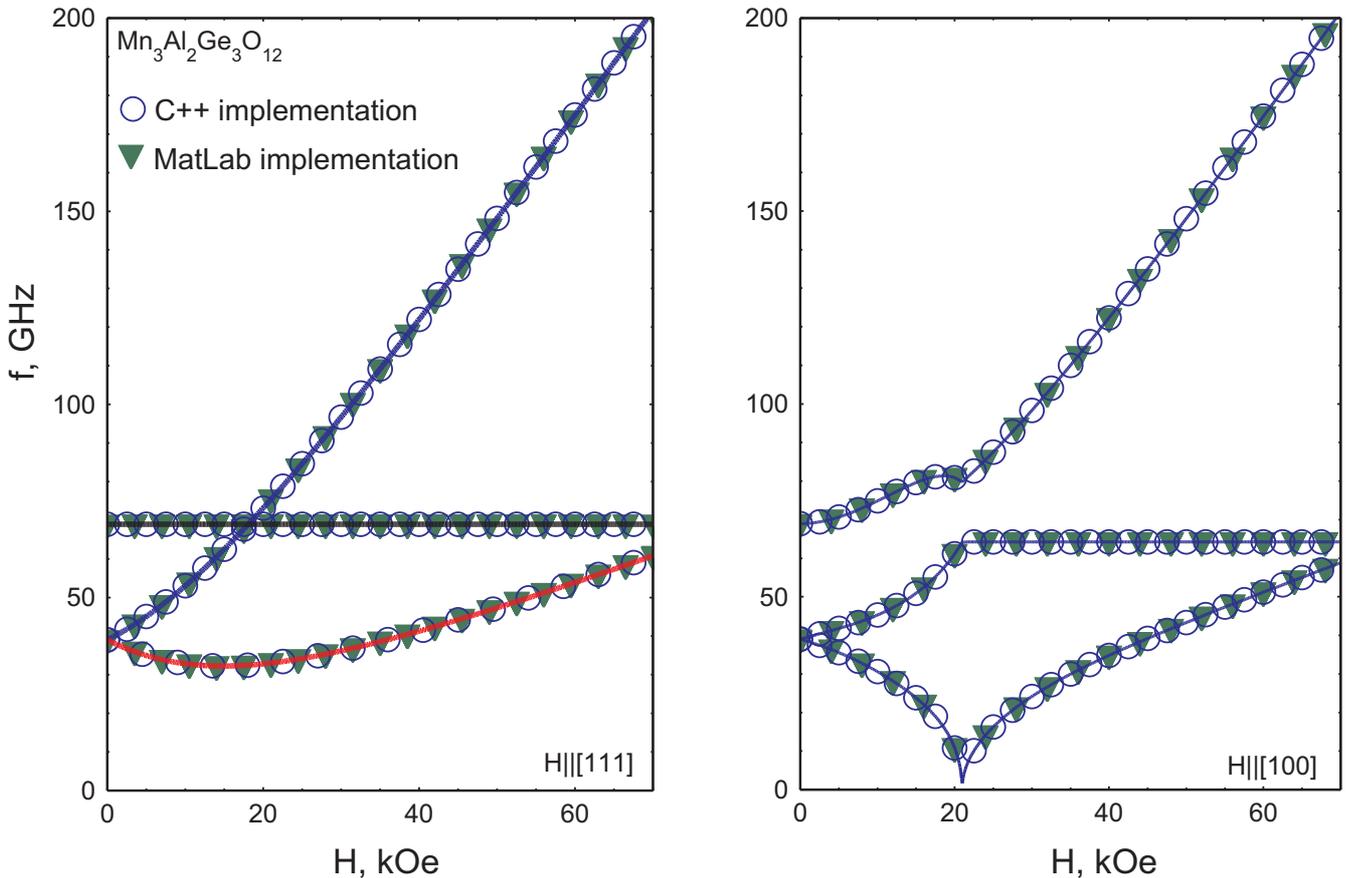


FIG. 1. Application of the numeric algorithms to the test example of $\text{Mn}_3\text{Al}_2\text{Ge}_3\text{O}_{12}$. Model parameters reproduce 1.2K experimental data of Ref.[5] and are listed in the Appendix. On all panels closed symbols are the results of MatLab implemented algorithm, open symbols are the results of C++ implemented algorithm. (Left panel) $\vec{H}||[111]$, bold solid lines are analytical solution; (Right panel) $\vec{H}||[100]$, curves are guide to the eye.

1. We start from specified start field $H = H_{start}$ applied in the specified direction.
2. We look for a new equilibrium position at field H . According to user choice we either look for global minimum or for a local minimum close to some initial approximation (specified initial approximation at first point or previous equilibrium position). Information on equilibrium position (Euler angles, projections of \vec{l}_i on the field direction and longitudinal susceptibility) is saved.
3. Matrix \mathcal{M} (Eqn.5) is calculated and $\det\mathcal{M} = 0$ equation is solved for eigenfrequencies. Results are saved.
4. Oscillating complex magnetization components and average longitudinal and transverse components of the oscillating magnetization for all oscillation modes are found and saved.
5. Field is increased by specified increment H_{step} . If the field does not reach its goal value H_{stop} we continue with Step 2.

All input parameters including anisotropy energy coefficients (Eqn.12), I_i and γ coefficients, magnetic field direction and limiting boundaries are specified in a text ini-file. Calculation results are saved in three files with static properties (equilibrium position, energy at equilibrium, longitudinal and transverse susceptibilities), oscillation eigenfrequencies and eigenvectors correspondingly. Formats of the ini-file and of the output files are described in details in supplementary materials below.

D. Application to the test examples

We tested our algorithms against test cases described in Sec.VIII A. Example of the numerically computed AFMR $f(H)$ dependence is shown at the Figure 1, detailed tests protocols are included in supplementary material below.

Test routine included: application to the test cases with known analytical results for $f(H)$, computation at the equivalent field orientations for cubic crystal, computation of the $f(H)$ curve at canted field orientation. We have found that numeric results coincides with known analytical solutions, both implementations of the algorithm yield the same results, no “gimbal lock” cases occurs.

Some minor instabilities of the numeric procedures were noted in highly degenerate cases (coincidence of resonance frequencies for different modes or presence of a zero-frequency mode), but they affect only less important output data. We found that sometimes determination of the frequency for $\omega = 0$ mode, which is not experimentally observable, is faulty or excitation condition determination is sometimes uncertain for the degenerate modes. Determination of the static properties and $f(H)$ curves for $f \neq 0$ was not affected by these issues.

IV. CONCLUSIONS (MAIN MATTER)

We present the algorithm for numerical solution of antiferromagnetic resonance frequencies for a noncollinear antiferromagnet of a general type within framework of the exchange symmetry theory [1]. Algorithm is implemented in the available MatLab and C++ codes (including ready-to-use compiled win32 executable) [15], implementations are tested against known analytically solvable models.

Authors thank Prof.A.I.Smirnov and Dr. L.E.Svistov (Kapitza Institute) for useful discussions. Work was supported by Russian Foundation for Basic Research grant No.16-02-00688.

Part II

Supplementary materials: detailed code description, output file formats and detailed tests

V. SOURCE FILES LOCATION

Source files of the MatLab and C++ implementations of the numeric algorithm for description of the AFMR modes of noncollinear antiferromagnets are available through the Authors web-site www.kapitza.ras.ru/rgroups/esrgroup/ (“NuMA: Numeric Methods for Antiferromagnets” section) [15].

VI. MATLAB IMPLEMENTATION

A. Editable fragments of the source file

MatLab source file `Noncolaf.m` starts from declarations of variables used and their initialization. We list here location of the editable lines in this file:

lines 1-2: These lines determine MatLab version used and creation date of the `Noncolaf.m` file. Following description is valid for the file dated as 21.04.2016.

lines 44-47: Set field direction (azimuthal and polar angles `hphi` and `htheta` in the crystallographic frame). Set field scan parameters: low field limit `Hlow`, high field limit `Hhigh`, field increment `delta`.

lines 49-62: Set parameters and form of anisotropy energy for the test case examples. Parameters and anisotropy energy for other cases should be specified in the same way. Please comment all unused parameters sets with a `%` symbol in the beginning of the line.

Selection of units is the user choice. In the case of the test examples magnetic field units are `kOe`, gyromagnetic ratio γ units are $\frac{10^9 \text{ rad}\cdot\text{s}^{-1}}{\text{kOe}}$ (which allows to obtain final frequencies as GHz), I_i units are $\frac{\text{kOe}^2}{(10^9 \text{ rad}\cdot\text{s}^{-1})^2}$. We set one of the coefficients of anisotropy energy to unity, units of potential energy density are then kOe^2 .

B. Output files format

Oscillations eigenfrequencies are saved in the default file “Oscillation Eigenfrequencies.txt”. This is 4-column txt-file:

col.1: Magnetic field value;

col.2-4: Oscillation frequencies.

In the default set of units field is measured in kOe and frequencies are measured in GHz.

Static properties are saved in the default file “Static Properties.txt”. This is a 10-columns txt-file:

col.1: Magnetic field value;

col.2-4: Euler angles α , β and γ (radians) describing equilibrium position of $\{\vec{l}_i^{(0)}\}$ vectors;

col.5: Potential energy value at equilibrium;

col.6-8: Projections of $\{\vec{l}_i^{(0)}\}$ vectors on the field direction;

col.9-10: Longitudinal and transverse magnetic susceptibilities at equilibrium ($\chi_{\parallel} = M_{\parallel}/H$, $\chi_{\perp} = M_{\perp}/H$).

Information on oscillations eigenvectors and average values of longitudinal and transverse oscillating magnetization is saved in the default file “Eigenvectors and Oscillating Magnetization Projections.txt”. This is 25-columns txt-file:

col.1: Magnetic field value;

col.2-7: Components of the complex eigenvector for the first oscillation mode. These components are calculated in the Euler angles space in the rotated frame of reference. The order of component is as follows: $Re(\alpha)$, $Im(\alpha)$, $Re(\beta)$, $Im(\beta)$, $Re(\gamma)$, $Im(\gamma)$.

col.8-19: Components of complex eigenvectors for second and third oscillation modes.

col.20-25: Average squared of longitudinal and transverse oscillating magnetization ($\sqrt{\langle \vec{m}_{\parallel}^2 \rangle}$ and $\sqrt{\langle \vec{m}_{\perp}^2 \rangle}$) for all three oscillation modes. The order is as follows: $\sqrt{\langle \vec{m}_{1\parallel}^2 \rangle}$, $\sqrt{\langle \vec{m}_{1\perp}^2 \rangle}$, $\sqrt{\langle \vec{m}_{2\parallel}^2 \rangle}$, $\sqrt{\langle \vec{m}_{2\perp}^2 \rangle}$, $\sqrt{\langle \vec{m}_{3\parallel}^2 \rangle}$, $\sqrt{\langle \vec{m}_{3\perp}^2 \rangle}$.

VII. C++ IMPLEMENTATION

A. Source files

This description corresponds to the v.1.00 dated as June 28,2016 of the program code. One can get program version information by running program with -v command line key (e.g., for the case of compiled Win32 application provided in the package, type `noncolaf-win32.exe -v` in a command line).

Source files includes some files from Numerical Recipes package [21]: `brent.c`, `f1dim.c`, `frprmn.c`, `jacobi.c`, `linmin.c`, `mnbrac.c`, `nrutil.c`, `nr.h`, `nrutil.h`. These files contains minimization routine `frprmn.c` and eigenvector search routine `jacobi.c`. These routines perform calculation with standard float precision, which is found to suffice for our goals.

Main program file is `noncolaf.cpp`, it uses additional functions defined in the set of header files:

cubic.h: Solution of the real cubic equation

vector.h: Defines 3D-vector algebra and reload standard operators to simplify vector operations

ini.h: Reading INI-file

magvect.h: Defines set of $\{\vec{l}_i\}$ vectors and their derivatives

energy.h: Calculates potential energy and its derivatives

minsearch.h: Defines functions to be called by minimization routine and preliminary search of the local minimum over the grid in the Euler angles space

saving.h: Includes calculations of the output parameters and their saving to the appropriate files

Full list of the source files includes 17 files. Source files were compiled using a DevC++ compiler v5.11 into a Win32 console application `noncolaf-win32.exe` which is also available for download.

Default output file names are specified in the lines 34-37 of the main `noncolaf.cpp` file, program version is defined in line 8 of the `noncolaf.cpp` file, default INI-file name is specified in the line 94 of `ini.h` file.

Presently C++ implementation consider only quadratic invariants in the anisotropy energy

$$U_A = \sum'_{i,j,\alpha,\beta} a_{ij}^{\alpha,\beta} l_i^\alpha l_j^\beta$$

here \sum' sign means that each $l_i^\alpha l_j^\beta$ combination is counted only once during summation. If one intend to include higher order terms corresponding modifications have to be done in `energy.h` file (potential energy and its derivatives calculations).

B. INI-file format

All parameters used for calculations are read from plain text INI-file. Default INI-file name is `noncolaf.ini`, examples of this file for the test cases are included in the package. INI-file format includes headers in square brackets (e.g., `[gamma]`) followed by the line with numeric value of appropriate parameter. Order of the parameters specification in the INI file is arbitrary, however numerical value of the parameter declared by its header have to be provided prior to other header declaration.

Dummy INI-file template with all necessary headers present will be created if no ini-file will be found. Examples of the INI-files for the test examples described in Sec.VIII A are available for download [15]. Consistency control of INI-file parameters for common errors (e.g., negative I_i) is performed before starting modeling, program terminates with appropriate warning if such an error is found.

List of the required INI-file headers is given below.

Lagrangian parameters:

[gamma]: Gyromagnetic ratio γ . Default units are $\frac{10^9 \text{rad}\cdot\text{s}^{-1}}{kOe}$ (which allows to obtain final frequencies as GHz).

[I1:], [I2:], [I3:]: Set of I_i parameters. Default units are $\frac{kOe^2}{(10^9 \text{rad}\cdot\text{s}^{-1})^2}$.

[AnisotropyStart:]: This header marks start of the anisotropy energy definition. It have to be followed by **[AnisotropyEnd:]** later (it is the only header that have to be terminated by other header). Anisotropy energy is assumed in the form $U_A = \sum'_{i,j,\alpha,\beta} a_{ij}^{\alpha,\beta} l_i^\alpha l_j^\beta$, all $a_{ij}^{\alpha,\beta} l_i^\alpha l_j^\beta$ terms have to be included between **[AnisotropyStart:]** and **[AnisotropyEnd:]** as separate lines. Format is straightforward and can be illustrated by the example of $\text{Mn}_3\text{Al}_2\text{Ge}_3\text{O}_{12}$: in this case $U_A = \lambda \left[I_{2z}^2 - I_{1z}^2 + \frac{2}{\sqrt{3}} (l_{1x}l_{2x} - l_{1y}l_{2y}) \right]$ and corresponding INI-file fragment looks as

```
[AnisotropyStart:]
l2z|2z;1
l1z|1z;-1
l1x|2x;1.15470054
l1y|2y;-1.15470054
[AnisotropyEnd:]
```

I.e. each line is a description of $l_i^\alpha l_j^\beta$ term followed by numerical value of coefficient, separated by semicolon. Each $l_i^\alpha l_j^\beta$ combination should appear only once, otherwise error message appears and program terminates.

Field scan parameters:

[Hdir:]: Specifies field direction as semicolon separated 3D-vector, e.g.:

```
[Hdir:] (Semicolon separated vector)
0;0;1
```

The length of this vector is arbitrary, program will norm it to unity during operation.

[Hstart:], [Hstop:], [Hstep:]: Starting and final field values and field increment. Field scan can be modeled both on the increasing or on the decreasing field ([Hstep:] should be negative in the later case). Default field units are kOe.

Algorithm versatility parameters:

[minsearch flag:]: Should be 1 or 0. If set to 1 then global minimum search over Euler angles space is performed when looking for $\{\vec{l}_i\}$ equilibrium orientation. If set to 0 then local minimum is followed: specified initial approximation is used on the first step, equilibrium position found is used as an initial approximation on the next step and so on.

[grid size:]: Should be integer N . It determines grid size in the Euler angles space that is used for rough equilibrium search if “global search” option is selected ([minsearch flag] header followed by 1). Rough search includes estimation of the potential energy in the N^3 points regularly spaced in Euler angles space. Recommended value is from 10 to 30.

[starting approximation:]: Determines Euler angles of the approximate equilibrium position used at the first point if “local search” option is selected ([minsearch flag] header followed by 0). Euler angles are defined as semicolon separated line: $\Theta; \phi; \psi$.

C. Error handling

Program checks INI-file for consistency and checks validity of some parameters (positiveness of γ and I_i , consistency of field scan parameters, unique definitions of anisotropy energy terms).

During calculations numeric uncertainties can result in incorrect results especially in strongly degenerated cases (usually if zero oscillation frequency is present). Sometimes this results in small negative or even complex ω^2 roots of $\det \mathcal{M} = 0$ equation. We arbitrary set a small cutoff limit (line 11 of `saving.h`) with default value $-1 \cdot 10^{-4}$, negative ω^2 above $(2\pi)^2$ times cutoff value is set to zero allowing for numeric uncertainty. In other cases (complex roots or larger negative ω^2 corresponding output frequency is set to -1 and oscillation eigenvectors are set to zero.

D. Output files format

Program creates 4 output files, default names are `noncolaf.dsk`, `noncolaf.st`, `noncolaf.frq` and `noncolaf.mag`. All of output data files includes header describing briefly its contents.

File `noncolaf.dsk` contains information about modeling parameters. It essentially duplicates INI file, presenting the same information in a more friendly formatted way.

File `noncolaf.st` contains information about static properties at equilibrium position. It is a 10 column text file:

col.1: Magnetic field

col.2: Potential energy value at equilibrium;

col.3-4: Longitudinal and transverse magnetic susceptibilities at equilibrium ($\chi_{\parallel} = M_{\parallel}/H$, $\chi_{\perp} = M_{\perp}/H$).

col.5-7: Euler angles Θ , ϕ and ψ (radians) describing equilibrium position of $\{\vec{l}_i^{(0)}\}$ vectors;

col.8-10: Directing cosines of magnetic field with respect to $\{\vec{l}_i^{(0)}\}$ vectors (i.e., $(\vec{l}_i^{(0)} \cdot \vec{n})$, here \vec{n} is a unitary vector in the field direction).

File `noncolaf.frq` contains oscillation frequencies. It is a 4 column txt-file:

col.1: Magnetic field value;

col.2-4: Oscillation frequencies.

File `noncolaf.mag` contains information on oscillation eigenvectors and average values of longitudinal and transverse components of oscillating magnetization. It is a 10-column txt-file:

col.1: Magnetic field value;

col.2-7: Average squared of longitudinal and transverse oscillating magnetization ($\sqrt{\langle \vec{m}_{\parallel}^2 \rangle}$ and $\sqrt{\langle \vec{m}_{\perp}^2 \rangle}$) for all three oscillation modes. The order is as follows: $\sqrt{\langle \vec{m}_{1\perp}^2 \rangle}$, $\sqrt{\langle \vec{m}_{1\parallel}^2 \rangle}$, $\sqrt{\langle \vec{m}_{2\perp}^2 \rangle}$, $\sqrt{\langle \vec{m}_{2\parallel}^2 \rangle}$, $\sqrt{\langle \vec{m}_{3\perp}^2 \rangle}$, $\sqrt{\langle \vec{m}_{3\parallel}^2 \rangle}$;

col.8-10: Oscillation eigenvectors for all modes. Each vector is a real space complex vector, each vector components are semicolon separated as follows: $[Re(m_x) + i \cdot Im(m_x)]$; $[Re(m_y) + i \cdot Im(m_y)]$; $[Re(m_z) + i \cdot Im(m_z)]$

VIII. DETAILED TEST PROTOCOLS

A. Analytically solvable models used as a test cases

We recall here some of the known examples of application of exchange symmetry theory to low-energy dynamics of noncollinear antiferromagnets. These analytical solutions were used as a test cases to ascertain correctness of numeric algorithms.

First test example is an antiferromagnet on a triangular lattice CsNiCl₃ [3]. In the ordered phase of this magnet spins form a planar 120° structure. High symmetry of triangular lattice leaves single invariant in the anisotropy energy $U_A = \beta (l_3^z)^2$, here z axis is normal to hexagonal plane and vector \vec{l}_3 is the normal to the plane of the planar spin structure, $\beta > 0$ as at zero field spin plane is orthogonal to the hexagonal crystallographic plane. Magnetic susceptibility normal to the spin plane dominates: $\chi_3 > \chi_2 = \chi_1$ (i.e. $I_3 < I_1 = I_2$). Two of the zero-field frequencies are zero, nonzero zero-field frequency is $\omega_0 = \gamma \sqrt{\frac{I_1 - I_3}{I_1 + I_3}} \beta = \gamma \sqrt{\frac{\chi_3 - \chi_1}{\chi_1}} \beta$. As the field is applied along z axis spin plane reorients at the field $H_0 = \sqrt{\frac{\beta}{\gamma^2(I_1 - I_3)}} = \sqrt{\frac{\beta}{\chi_3 - \chi_1}}$. Magnetic resonance frequencies at $\vec{H} \parallel z$ are given by equations:

$$\begin{aligned} H < H_0 : \omega_1^2 &= \omega_0^2 + (\gamma H)^2 \\ \omega_2 &= \omega_3 = 0 \\ H > H_0 : \omega_{1,2} &= \sqrt{\left(\frac{I_1}{I_1 + I_3} \gamma H\right)^2 - \omega_0^2} \pm \frac{I_3}{I_1 + I_3} \gamma H = \\ &= \sqrt{\left(\frac{\chi_3}{2\chi_1} \gamma H\right)^2 - \omega_0^2} \pm \frac{2\chi_1 - \chi_3}{2\chi_1} \gamma H \\ \omega_3 &= 0 \end{aligned}$$

Because of simplicity of anisotropy energy this problem can be solved analytically at arbitrary field orientation, see Ref.[3] for details.

To reproduce experimental results of Ref.[3] we take for our modeling $\beta = 1 \text{ kOe}^2$, $\gamma = 18.8 \frac{10^9 \text{ rad} \cdot \text{s}^{-1}}{\text{kOe}}$ (3.0 GHz/kOe in frequency units), $I_1 = I_2 = 8.77 \cdot 10^{-6} \frac{\text{kOe}^2}{(10^9 \text{ rad} \cdot \text{s}^{-1})^2}$ and $I_3 = 9.75 \cdot 10^{-7} \frac{\text{kOe}^2}{(10^9 \text{ rad} \cdot \text{s}^{-1})^2}$.

Secondly, we consider twelve-sublattices antiferromagnet Mn₃Al₂Ge₃O₁₂ [5]. Here $I_1 = I_2$ because of the cubic symmetry, anisotropy energy $U_A = \lambda \left[l_{2z}^2 - l_{1z}^2 + \frac{2}{\sqrt{3}} (l_{1x}l_{2x} - l_{1y}l_{2y}) \right]$ ($\lambda > 0$) (we use notations of Ref.[22]). At zero field plane of the spiral structure is orthogonal to one of the $\langle 111 \rangle$ directions. Oscillation eigenfrequencies can be found at $\vec{H} \parallel [111]$:

$$\begin{aligned} \omega_{1,2} &= \sqrt{\left(\frac{I_1}{I_1 + I_3} \gamma H\right)^2 + \frac{4}{3} \frac{\lambda}{(I_1 + I_3)} \pm \frac{I_3}{I_1 + I_3} \gamma H} = \\ &= \sqrt{\left(\frac{\chi_3}{2\chi_1} \gamma H\right)^2 + \frac{4}{3} \frac{\lambda}{\chi_1} \gamma^2 \pm \frac{2\chi_1 - \chi_3}{2\chi_1} \gamma H} \\ \omega_3 &= \sqrt{\frac{8}{3} \frac{\lambda}{I_1}} = \gamma \sqrt{\frac{8}{3} \frac{2\lambda}{\chi_3}} \end{aligned}$$

To reproduce experimental results of Ref. [5] we take for our modeling $\lambda = 1 \text{ kOe}^2$, $\gamma = 17.6 \frac{10^9 \text{ rad} \cdot \text{s}^{-1}}{\text{kOe}}$ (2.80 GHz/kOe), $I_1 = I_2 = 1.42 \cdot 10^{-5} \frac{\text{kOe}^2}{(10^9 \text{ rad} \cdot \text{s}^{-1})^2}$, $I_3 = 7.99 \cdot 10^{-6} \frac{\text{kOe}^2}{(10^9 \text{ rad} \cdot \text{s}^{-1})^2}$. Results of the modeling for this case are shown at the Figure 1.

Finally, it is a spiral magnet LiCu_2O_2 [8]. Despite of the orthorhombic symmetry $I_1 = I_2$ as there is no anisotropy in the plane of the spiral structure, $U_A = \frac{A}{2}I_{3z}^2 + \frac{B}{2}I_{3y}^2$ ($A \leq B \leq 0$). It turns out that in the case of LiCu_2O_2 A and B constants in anisotropy energy are close within 1%. Thus, normal to the spin plane \vec{l}_3 rotates almost freely in the (yz) plane. One of the oscillation frequencies corresponds to the rotation in the plane of spiral structure and is always zero since phase of the helix can be changed at no energy cost. Two other modes have non-zero zero-field frequencies

$$\begin{aligned}\omega_{10}^2 &= -\frac{A}{I_1 + I_3} = -\gamma^2 \frac{A}{\chi_1} \\ \omega_{20}^2 &= \frac{B - A}{I_1 + I_3} = \gamma^2 \frac{B - A}{\chi_1} < \omega_{10}^2\end{aligned}$$

For LiCu_2O_2 $\chi_3 > \chi_1$, in this case at $\vec{H}||z$ vector \vec{l}_3 always remains aligned along z and non-zero oscillation frequencies are

$$\begin{aligned}\omega_{1,2}^2 &= \frac{\omega_{10}^2 + \omega_{20}^2}{2} + \gamma^2 H^2 \frac{I_3^2 + I_1^2}{(I_3 + I_1)^2} \pm \\ &\pm \sqrt{\left(\frac{\omega_{10}^2 - \omega_{20}^2}{2}\right)^2 + 4 \frac{\gamma^4 H^4 I_1^2 I_3^2}{(I_1 + I_3)^4} + 2 \frac{\gamma^2 H^2 (\omega_{10}^2 + \omega_{20}^2) I_3^2}{(I_1 + I_3)^2}}\end{aligned}$$

At $\vec{H}||x$ spin plane rotates orthogonally to the magnetic field at some critical field. Critical field $H_{cx} = \frac{\omega_{10}}{\gamma} \sqrt{\frac{I_1 + I_3}{I_1 - I_3}} = \frac{\omega_{10}}{\gamma} \sqrt{\frac{\chi_1}{\chi_3 - \chi_1}}$ and oscillation frequencies are

$$\begin{aligned}H < H_{cx} : \\ \omega_1^2 &= \omega_{10}^2 + \gamma^2 H^2 \\ \omega_2^2 &= \omega_{20}^2 \\ H > H_{cx} : \\ \omega_{1,2}^2 &= \frac{\omega_{20}^2 - 2\omega_{10}^2}{2} + \frac{I_1^2 + I_3^2}{(I_1 + I_3)^2} \gamma^2 H^2 \pm \\ &\pm \sqrt{\frac{\omega_{20}^4}{4} + 2\gamma^2 H^2 \frac{(\omega_{20}^2 - 2\omega_{10}^2) I_3^2}{(I_1 + I_3)^2} + 4 \frac{\gamma^4 H^4 I_1^2 I_3^2}{(I_1 + I_3)^4}}\end{aligned}$$

To reproduce experimental results of Ref. [8] we take for our modeling $\gamma = 17.59 \frac{10^9 \text{rad}\cdot\text{s}^{-1}}{\text{kOe}}$ (corresponds to 2.80 GHz/kOe), $A = -1\text{kOe}^2$, $B = -0.99 \text{kOe}^2$, $I_1 = I_2 = 1.85 \cdot 10^{-7} \frac{\text{kOe}^2}{(10^9 \text{rad}\cdot\text{s}^{-1})^2}$, $I_3 = 6.18 \cdot 10^{-8} \frac{\text{kOe}^2}{(10^9 \text{rad}\cdot\text{s}^{-1})^2}$

B. Case of CsNiCl_3

Both implementations of numeric procedure reproduce analytical $f(H)$ curves well (Figs.2 and 3). Analytical $f(H)$ dependences can be calculated for CsNiCl_3 in arbitrary field orientation [3], we performed our modeling in certain representative cases: $\vec{H}||z$ ($\phi = 0^\circ$), slightly canted field ($\phi = 10^\circ$) and $\vec{H} \perp z$ ($\phi = 90^\circ$).

C++ implementation demonstrated numeric instability at $\vec{H}||z$ for $H < H_c$. At these fields zero frequency mode is two-fold degenerated and numeric uncertainties of calculation lead to imaginary roots for ω^2 in $\det\mathcal{M} = 0$ equation (two fold degeneracy of cubic equation root means that cubical parabola is tangent to $y = 0$ at some point, condition extremely sensitive to coefficients definition). As described above, this error was handled by setting output frequency to dummy value of -1 . Since degeneracy of oscillation modes is a rare event, we believe that this do not cause big discomfort. MatLab implementation was free from this problem.

Static properties and excitation conditions modeled by both implementations coincide (Fig.4). CsNiCl_3 demonstrate spin-reorientation transition at $H_c \approx 19\text{kOe}$, at this field normal to the plane of the spin structure rotates along the field direction. This is reproduced by both approaches. As the field is canted from the symmetry axis ($\phi = 10^\circ$ case) spin-reorientation became smeared over certain field range, as expected. Note that when looking for the global minimum numeric procedure randomly switches between equivalent orientations $\vec{l}_3||\vec{H}$ and $\vec{l}_3||-\vec{H}$ above H_c . As

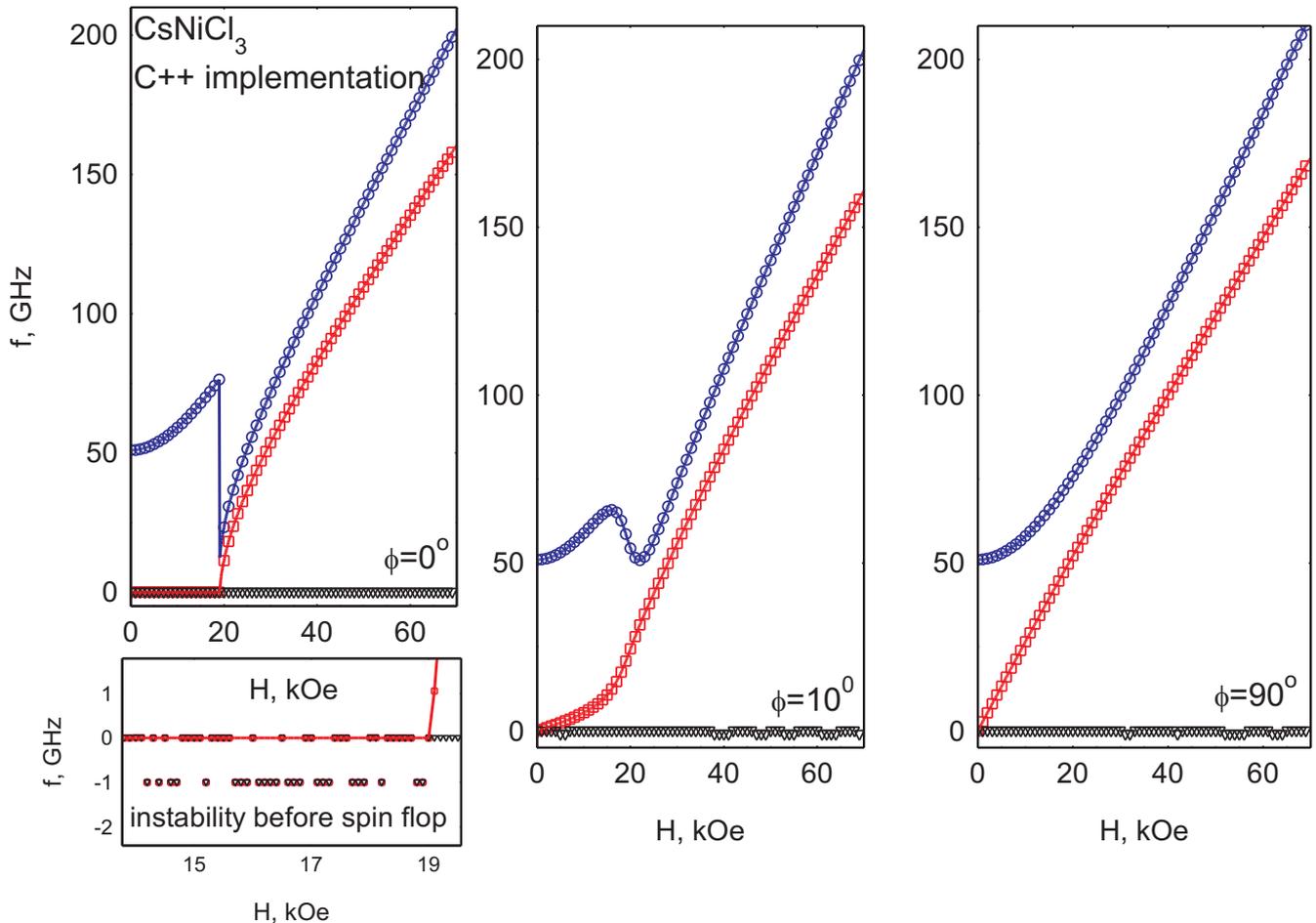


FIG. 2. Frequency-field dependences modeled for CsNiCl_3 using C++ implementation of the numeric procedure. Curves are analytical results, symbols — numeric modeling. Main panels show $f(H)$ curves at different field orientation with respect to the anisotropy axis z ($\phi = 0^\circ$ means $\vec{H} \parallel z$). Small panel illustrates instability of the numeric procedure due to the strong degeneration of the dynamics equations in this case.

these orientations are equivalent, this does not cause any problem when calculating physically observed quantities (oscillation frequencies, susceptibilities etc.). This issue can be evaded by setting on local minimum search option in C++ implementation, which inherits starting approximation for energy minimum search from previous field point. Excitation conditions are shown for one of the modes for $\phi = 10^\circ$ case. Calculation coincide for both implementations, note that average longitudinal magnetization magnitude is not negligible for this mode.

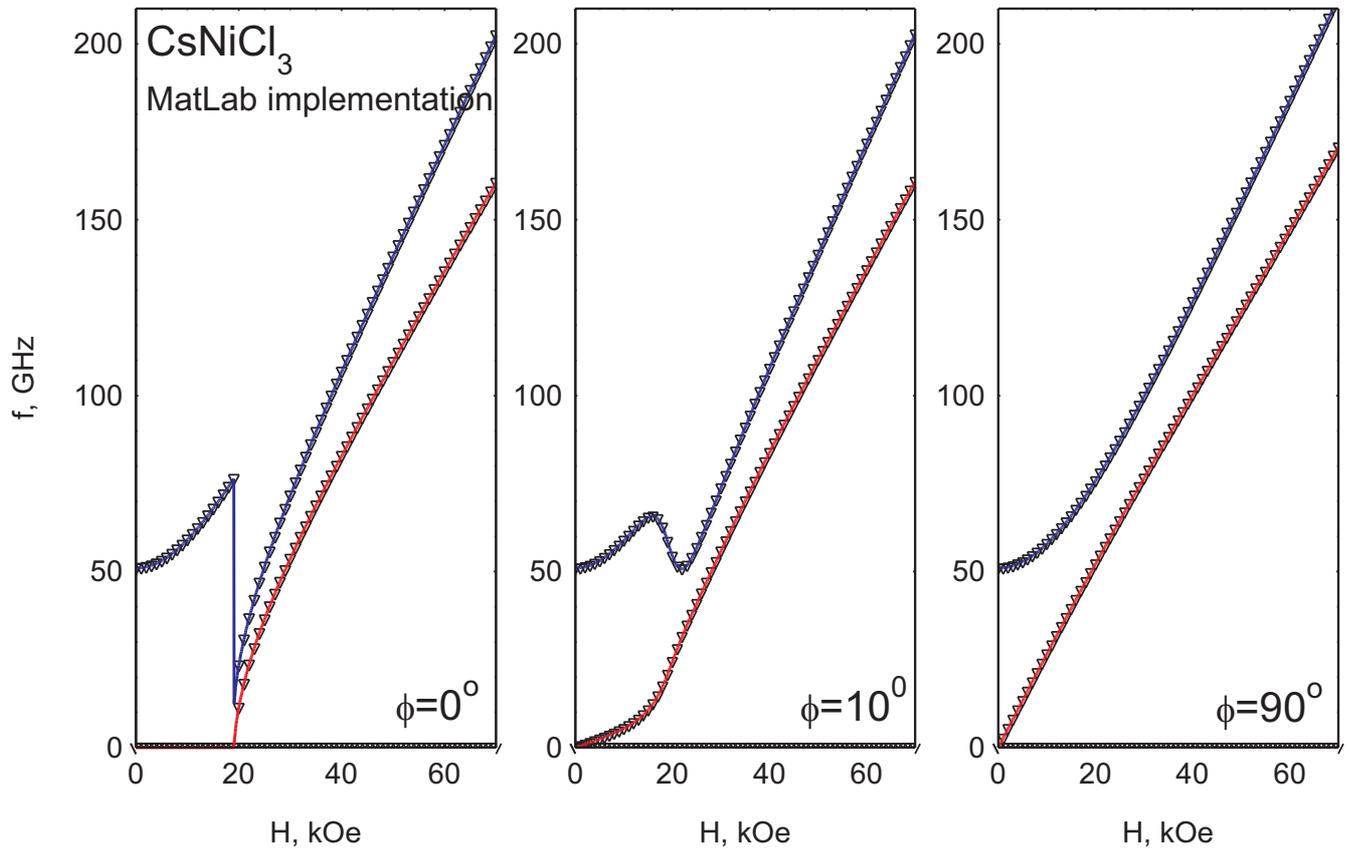


FIG. 3. Frequency-field dependences modeled for CsNiCl_3 using MatLab implementation of the numeric procedure. Curves are analytical results, symbols — numeric modeling. Panels show $f(H)$ curves at different field orientation with respect to the anisotropy axis z ($\phi = 0^\circ$ means $\vec{H} \parallel z$).

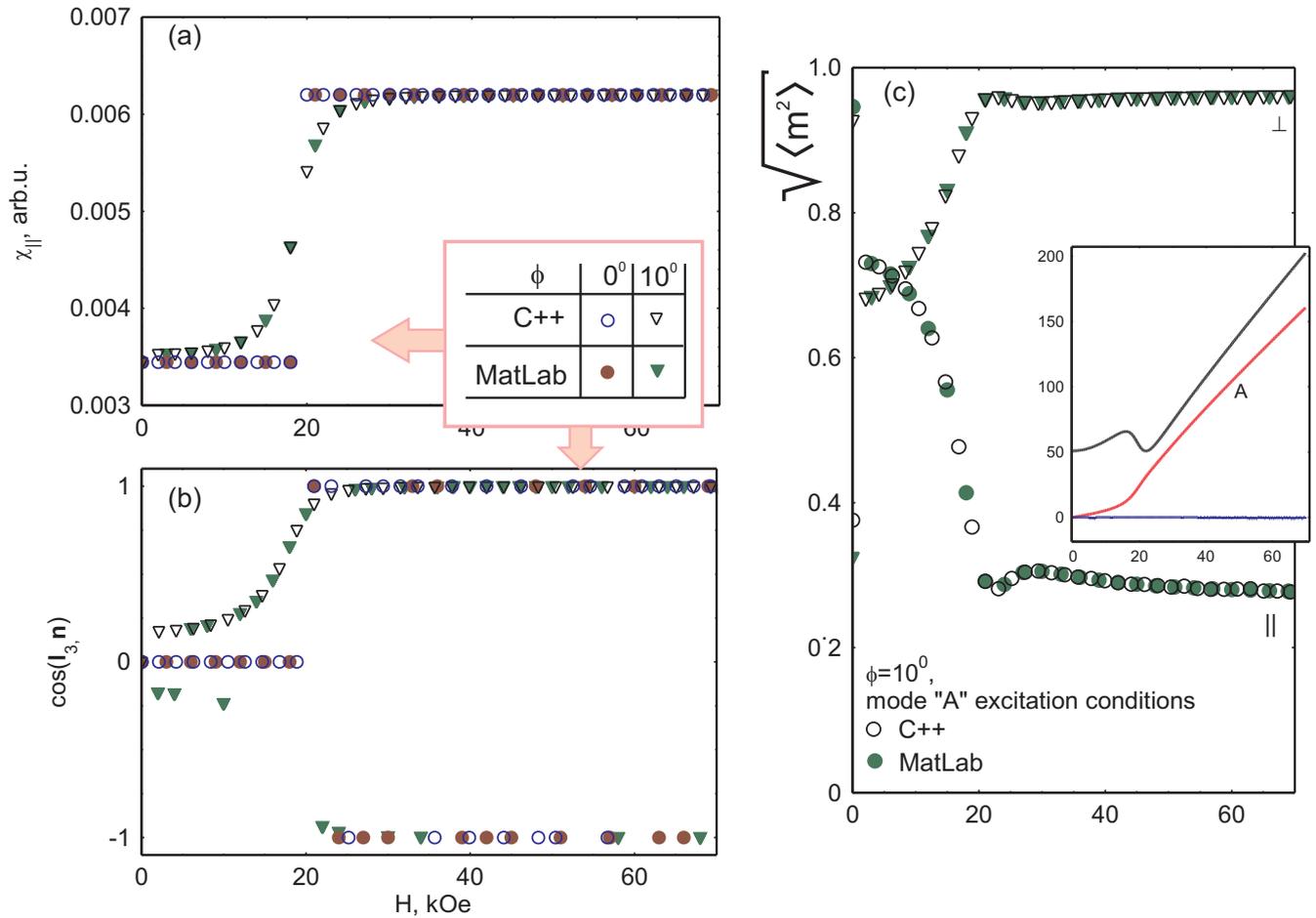


FIG. 4. Comparison of modeled static properties (panels (a) and (b)) and excitation conditions for one of the AFMR mode (panel(c)) calculated for the test case of CsNiCl₃ by different implementations of numeric procedure. (a) — longitudinal static susceptibility; (b) — orientation of spin structure with respect to the magnetic field; (c) — excitation conditions for the mode denoted as "A" on the insert (triangles correspond to conventional transverse pumping, circles — to the longitudinal pumping).

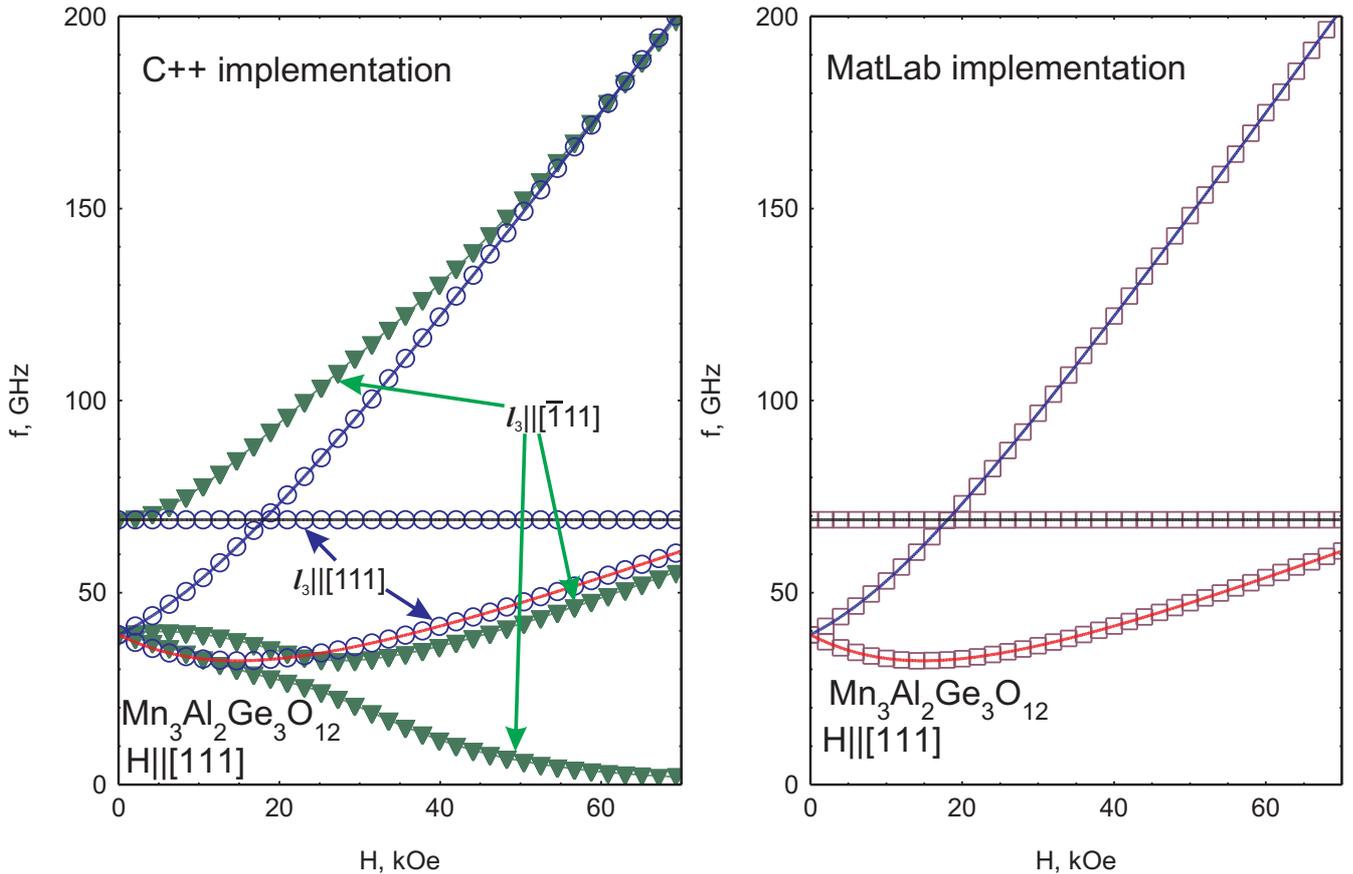


FIG. 5. Modeled $f(H)$ curves for $\text{Mn}_3\text{Al}_2\text{Ge}_3\text{O}_{12}$ at $\vec{H}||[111]$. Left panel — C++ implementation, right panel — MatLab implementation. Thick solid lines on both panels are analytical calculations. On the left panel (C++ implementation) $f(H)$ curves for stable (circles) and metastable (triangles) domains are calculated.

C. Case of $\text{Mn}_3\text{Al}_2\text{Ge}_3\text{O}_{12}$

In the case of $\text{Mn}_3\text{Al}_2\text{Ge}_3\text{O}_{12}$ at zero field four equivalent magnetic domains are possible at zero field, planes of the spin structure in these domains lie orthogonal to different $\langle 111 \rangle$ axes of the cubic crystal. For the field applied along $[111]$ axis one of these domains is stable, while other remains metastable. Dynamics equations can be obtained analytically for the stable domain, we have found that both implementation yields the same numeric results (Fig.5). Additionally, one can obtain $f(H)$ curve for metastable domain making use of local minimum search option in C++ implementation.

At $\vec{H}||[001]$ all domains are equivalent. In this orientation spin reorientation takes place: planes of the spin structure begin to rotate as the field is applied and complete reorientation by setting plane of the spin structure orthogonal to the applied field at the critical field $H_c \approx 21\text{kOe}$. No analytical solution of dynamics equation is possible in this orientation, numeric methods easily solve this problem (Fig.6). Both implementation results coincide. Again, see panel (b) of Fig.6, numeric minimum energy search procedure sometimes switches between equivalent domains, but this does not affect observable quantities ($f(H)$ or $\chi(H)$).

As it is well known, spin reorientation is very sensitive to the exact orientation of the magnetic field with respect to the crystallographic axis. Numeric methods allow to model this situation as well (Fig.7), which eases analysis of experimental data, allowing to estimate canting of the sample, for example.

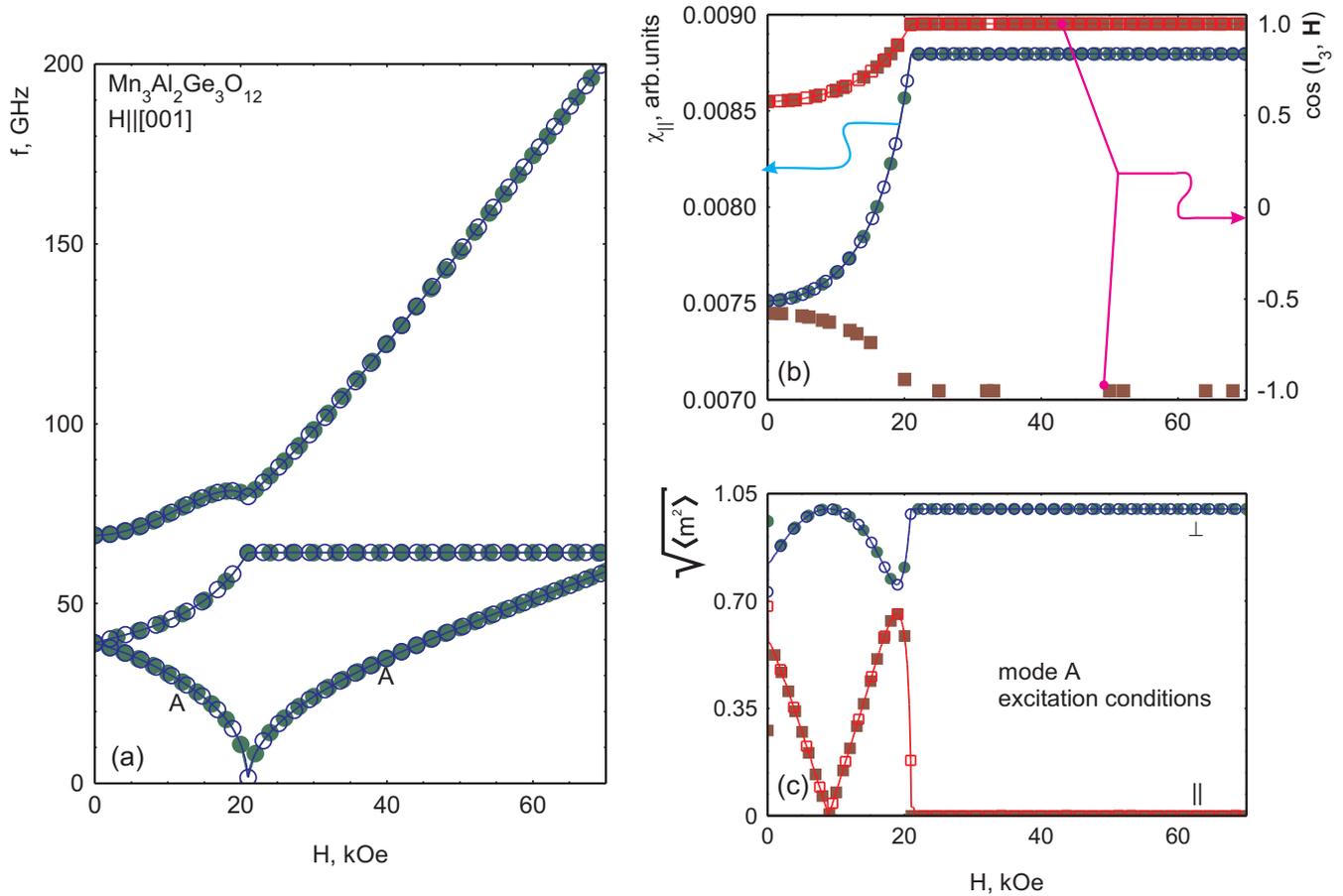


FIG. 6. Modeled $f(H)$ curves, static properties and excitation conditions for $\text{Mn}_3\text{Al}_2\text{Ge}_3\text{O}_{12}$ at $\vec{H} \parallel [001]$. Open symbols — C++ implementation, closed symbols — MatLab implementation, curves — guide to the eye. (a) $f(H)$ dependence, (b) longitudinal susceptibility (circles, left Y-axis) and orientation of the normal to the plane of the spin structure with respect to the field (squares, right Y-axis), (c) average oscillating transverse (circles) and longitudinal (squares) magnetization for mode A (see panel (a)).

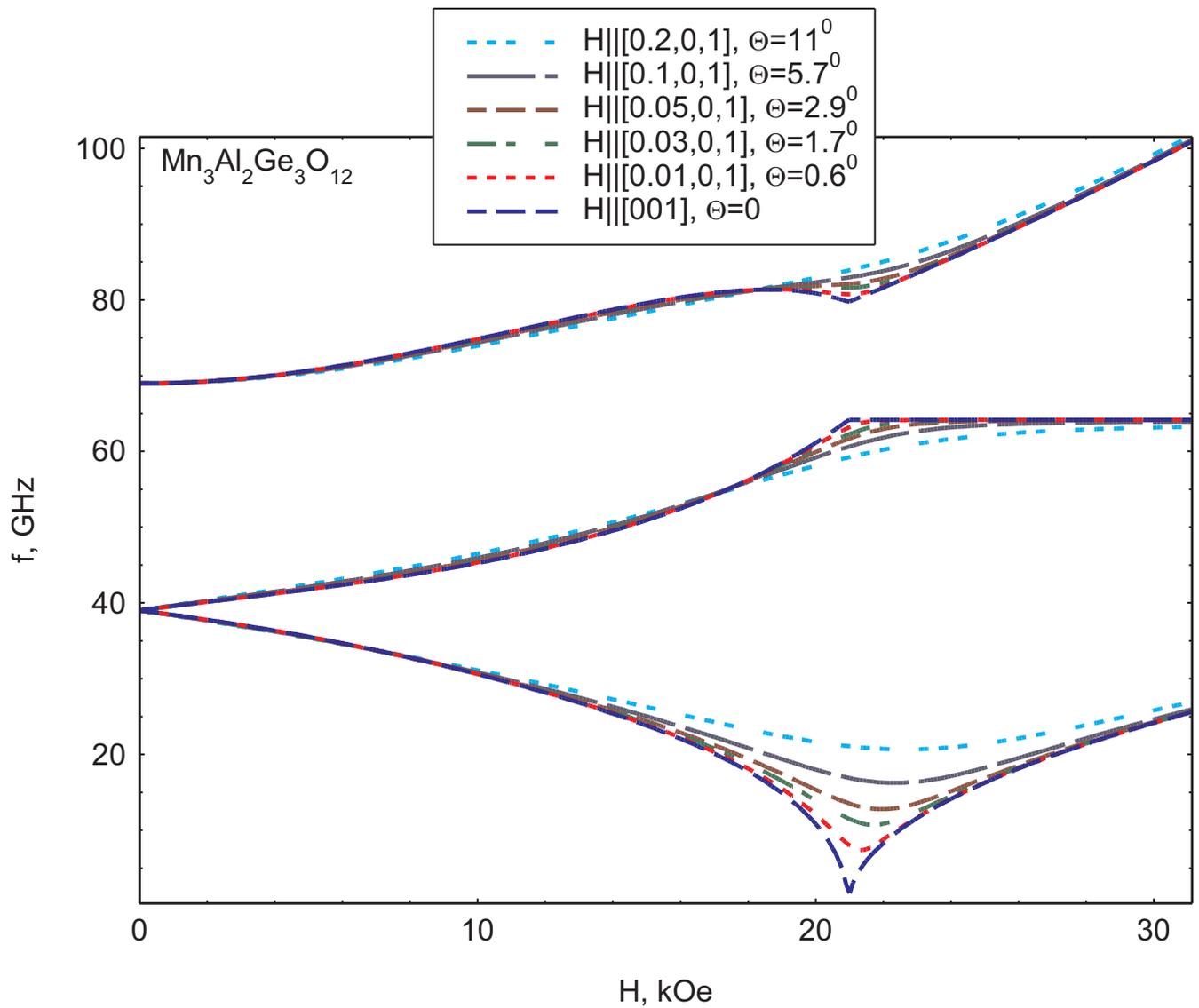


FIG. 7. Modeled $f(H)$ curves for $\text{Mn}_3\text{Al}_2\text{Ge}_3\text{O}_{12}$ at the field $\vec{H} \parallel [\delta 01]$. C++ implementation used only.

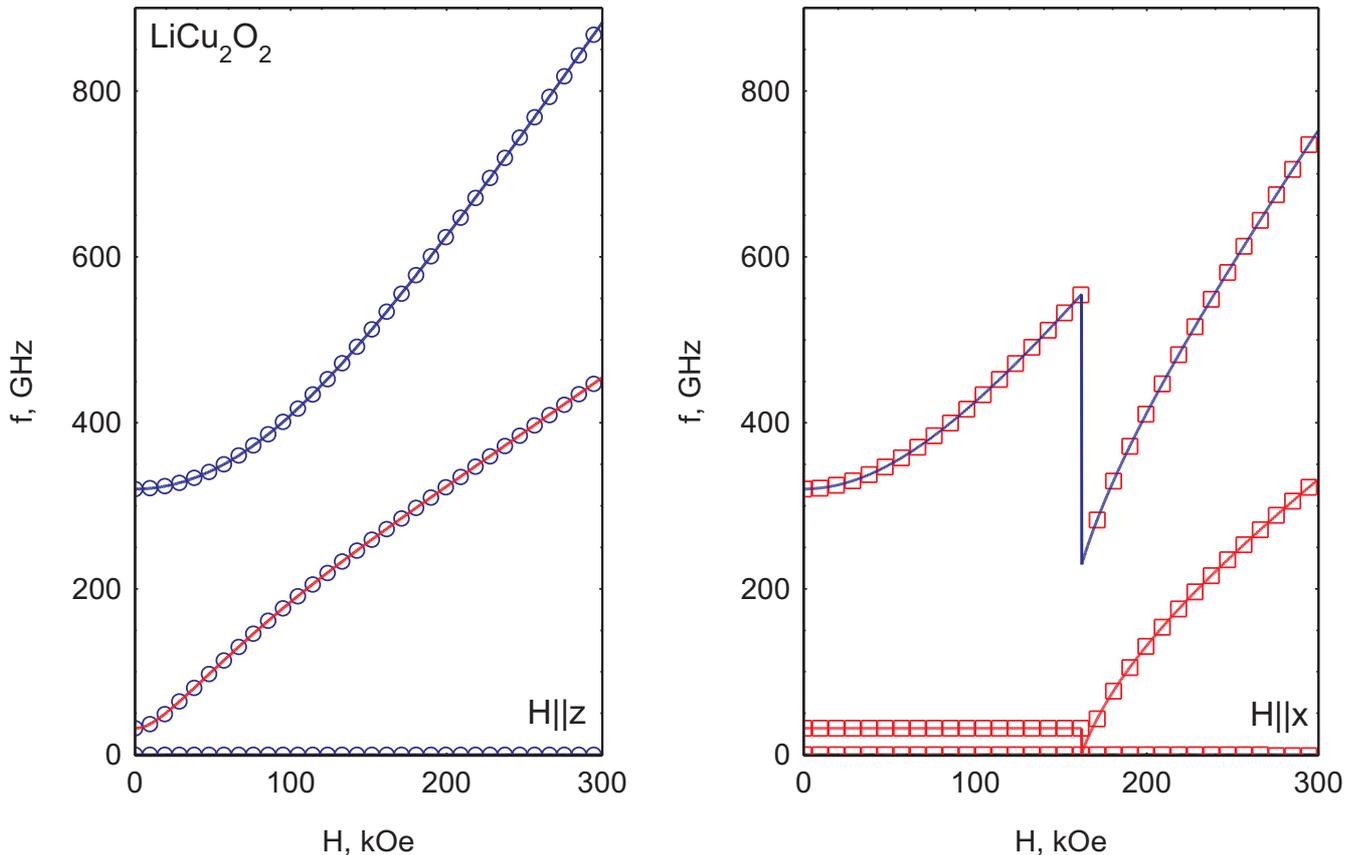


FIG. 8. Modeled $f(H)$ curves for LiCu_2O_2 , C++ implementation of the numeric procedure. Left panel $\vec{H}||z$, right panel $\vec{H}||x$. Thick solid lines on both panels are analytical calculations.

D. Case of LiCu_2O_2

AFMR modes for LiCu_2O_2 can be found analytically in main orientations of applied field. Results of numeric procedure (Figs.8 and 9) fits analytical curves well. As anisotropy constants for LiCu_2O_2 are very close (they differ by 1%) accuracy of the equilibrium position determination affects strongly one of the AFMR modes at $\vec{H}||x$: the field independent $f \approx 32\text{GHz}$ mode corresponds to the oscillation of the spin structure in the (yz) plane and its frequency is determined by difference of anisotropy constants. We have found, that MatLab implementation is more sensitive to this issue (see inset at the Fig.9) with uncertainties up to 1 GHz (3% accuracy) for the set of parameters specified in this text (section VIII A). We have found that stability of the numeric output can be improved by scaling model parameters (anisotropy constants and I_i constants) by the factor of 1000 (as it is described in the main paper, scaling factor can be chosen arbitrary). It seems that this issue is due to some built-in rounding restrictions in MatLab, C++ implementation was free of this issue.

As for other examples, we model static properties and excitation conditions for one of AFMR modes (Fig.10). Both implementations results coincide. At $\vec{H}||x$ a sudden spin-reorientation is expected for LiCu_2O_2 , at this field spin plane rotates normally to the applied field. Note that excitation conditions for the AFMR mode, which is field-independent below H_c also suddenly change at spin-reorientation: the oscillating magnetization is parallel to the applied field below H_c and orthogonal to the applied field above H_c .

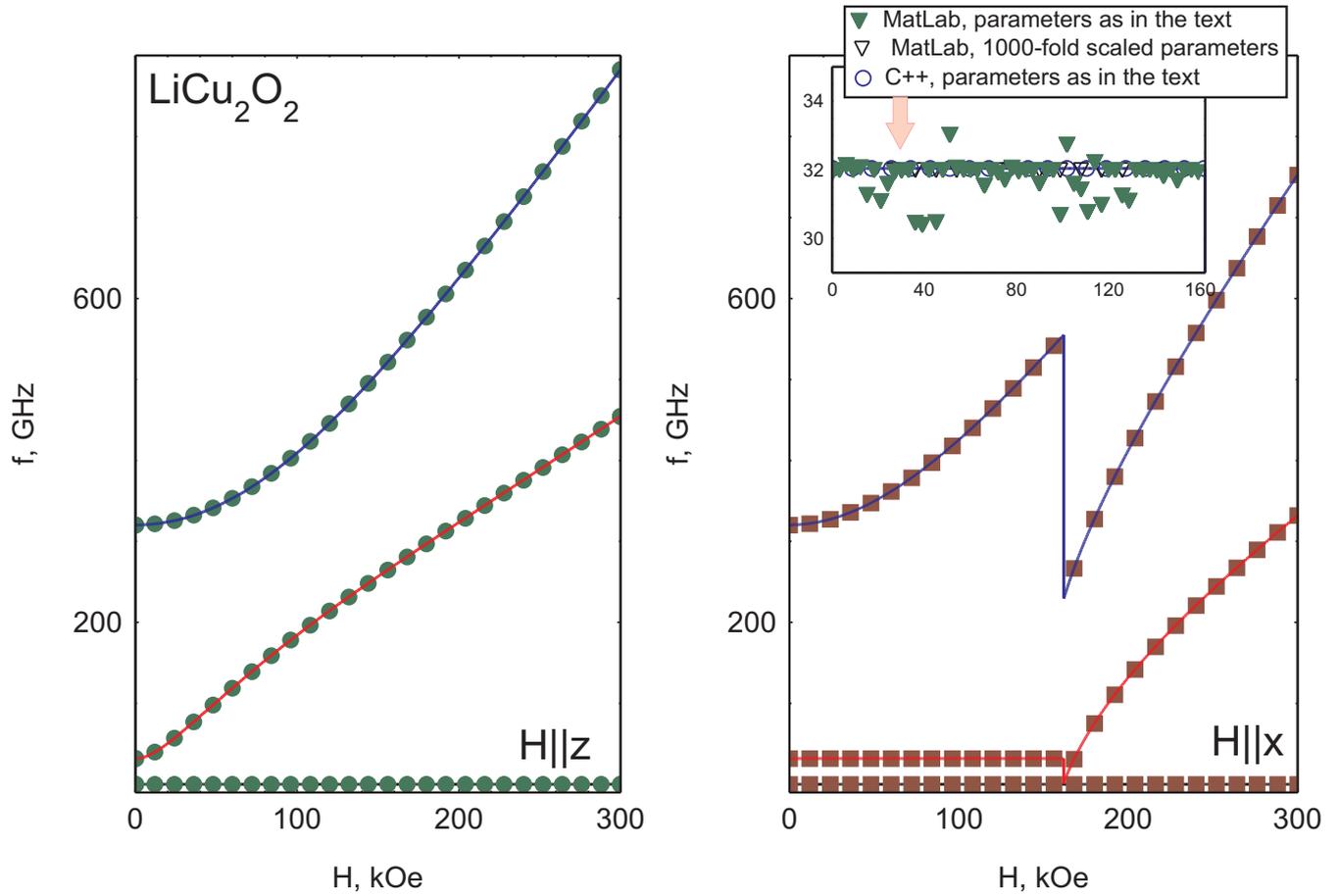


FIG. 9. Modeled $f(H)$ curves for LiCu_2O_2 , MatLab implementation of the numeric procedure. Left panel $\vec{H}||z$, right panel $\vec{H}||x$. Thick solid lines on both panels are analytical calculations. Insert on the left panel illustrates numeric uncertainties of the 32 GHz AFMR mode calculation below H_c .

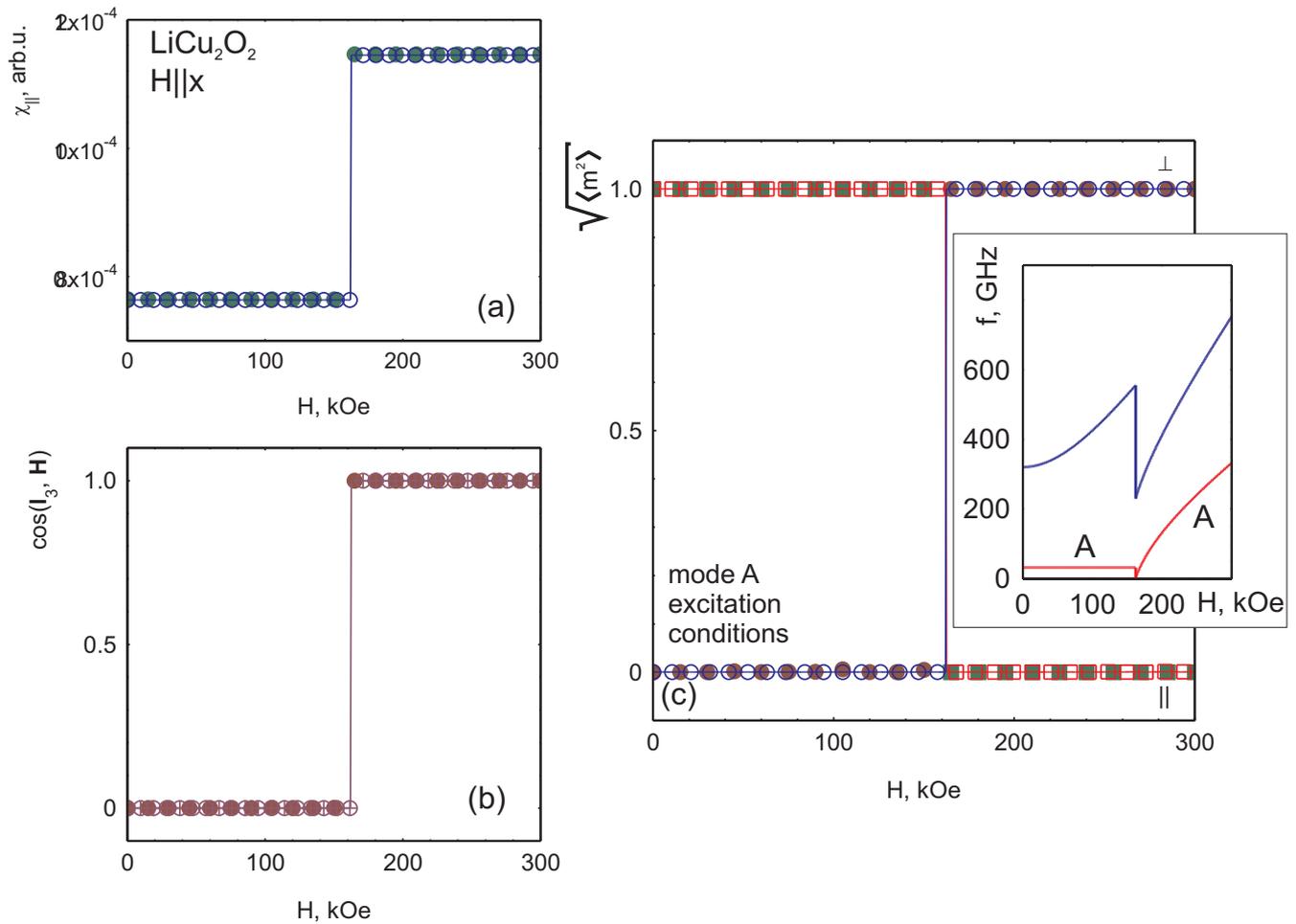


FIG. 10. Modeled static properties and excitation conditions for one of the AFMR modes for LiCu_2O_2 , $\vec{H} \parallel x$. Open symbols — C++ implementation, closed symbols — MatLab implementation, curves are guides to the eye. (a) longitudinal susceptibility, (b) orientation of the spin plane with respect to the magnetic field, (c) average amplitude of longitudinal (squares) and transverse (circles) components of oscillating magnetization in AFMR mode “A” (see inset).

IX. CONCLUSIONS (SUPPLEMENTARY MATERIAL)

We have checked C++ and MatLab implementations of numerical algorithm for description of AFMR $f(H)$ dependences in noncollinear antiferromagnets against some known analytically solvable cases. We have found that modeled results fits to analytical results well, both implementation results coincide. We have not found serious instabilities in the algorithm implementations, several minor issues related to strong degeneracy of the particular cases were observed and discussed.

-
- [1] A.F. Andreev, V.I. Marchenko, *Sov. Phys. Usp.* **130**, 39 (1980)
 - [2] V.J. Minkiewicz, D.E. Cox, G. Shirane, *Solid State Communications* **8**, 1001 (1970)
 - [3] I.A. Zaliznyak, V.I. Marchenko, S.V. Petrov, L.A. Prozorova, A.V. Chubukov, *JETP Letters* **47**, 211 (1988)
 - [4] W. Prandl, *Physica Status Solidi (b)* **55**, K159 (1973)
 - [5] L.A. Prozorova, V.I. Marchenko, Yu.V. Krasnyak, *JETP Letters*, **41**, 637 (1985)
 - [6] A.A. Gippius, E.N. Morozova, A.S. Moskvina, A.V. Zalessky, A.A. Bush, M. Baenitz, H. Rosner, and S.L. Drechsler *Phys. Rev. B* **70**, 020406(R) (2004)
 - [7] T. Masuda, A. Zheludev, A. Bush, M. Markina, and A. Vasiliev, *Phys. Rev. Lett.* **92**, 177201 (2004)
 - [8] L.E. Svistov, L.A. Prozorova, A.M. Farutin, A.A. Gippius, K.S. Okhotnikov, A.A. Bush, K.E. Kamentsev, E. A. Tishchenko, *JETP* **108**, 1000 (2009)
 - [9] J.R. Stewart, G. Ehlers, A.S. Wills, S.T. Bramwell and J.S. Gardner *Journal of Physics: Condensed Matter*, **16**, L321 (2004)
 - [10] T. Nagamiya, K. Yosida and R. Kubo, *Advances in Physics*, **4**, 1 (1955)
 - [11] H. Tanaka, S. Teraoka, E. Kakehashi, K. Iio, and K. Nagata, *J. Phys. Soc. Jpn.* **57**, 3979 (1988)
 - [12] A.V.Chubukov and D.I.Golosov, *J.Phys.:Condens.Matter*, **3**, 69 (1991)
 - [13] S.S.Sosin, L.A.Prozorova, P. Bonville, M.E.Zhitomirsky, *Phys. Rev. B*, **79**, 014419 (2009)
 - [14] SpinW Homepage by S.Tóth, <https://www.psi.ch/spinw/spinw>
 - [15] Authors Web-page, <http://www.kapitza.ras.ru/rgroups/esrgroup/>, see “NuMA: Numeric Methods for Antiferromagnets” section of the web-page
 - [16] S.S.Sosin, A.I.Smirnov, L.A.Prozorova, G.Balakrishnan, M.E.Zhitomirsky, *Phys.Rev B*, **73**, 212402 (2006)
 - [17] V.I.Marchenko and A.M.Tikhonov, *JETP Letters* **69**, 44 (1999)
 - [18] A.N.Vasilev, V.I.Marchenko, A.I.Smirnov, S.S.Sosin, H.Yamada and Y.Ueda, *Phys. Rev. B* **64**, 174403 (2001)
 - [19] V.N.Glazkov, A.M.Farutin, V.Tsurkan, H-A. Krug von Nidda and A.Loidl, *Phys. Rev. B* **79**, 024431 (2009)
 - [20] A.M.Farutin, V.I.Marchenko, *JETP Lett.* **83**, 238 (2006)
 - [21] W.H.Press, S.A.Teukolsky, W.T.Vetterling, B.P.Flannery, *Numerical Recipes: The Art of Scientific Computing*, Cambridge University Press (2007), <http://numerical.recipes>
 - [22] O.G.Udalov, *JETP* **113**, 490 (2011)