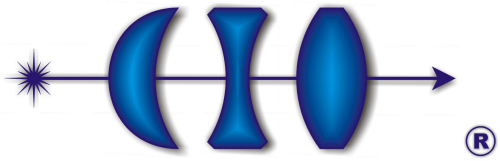

CENTRO DE INVESTIGACIONES
EN OPTICA, A.C.

# TINIBA® Tutorial: Version 2.0

Bernardo S. Mendoza and J. L. Cabellos
INDAUTOR Certificate A.1

División de Fotónica, Centro Investigaciones en Óptica, A.C
Lomas de Bosque # 115, Col. El Campestre, León Gto. México C.P. 37150
"Grupo de Propiedades Ópticas de Nano-Sistemas, Interfases
y Superficies" (*PRONASIS*)

August 19, 2014

# Contents

# Chapter 1

# Introduction

*WARNING:* First of all here we adopt the philosophy **TINIBA® is not a black box** and you might have some idea about what are you doing in order to make sense. and the other thing that you have to know GNULinux at basic level, understand something scripting under bash and perl, and enjoy tiniba.

## 1.1 Why this guide?

In order to improve the calculations faster and easier.

## 1.2 Publications with tiniba

A list of publications that used TINIBA® package:
1.- Phys. Rev. B 80, 201312 (2009)
2.- Phys. Rev. B 80, 245204 (2009)

## 1.3 At the begining

The propouse of this toutorials is teaching the use of TINIBA® on *Medusa*.
Before you follow this tutorials you need to talk Dr. Bernardo. Maybe you can find him in the coffe break, around 12:00 A.M.

## 1.4 Installation on medusa

- Copy the latest version of TINIBA® from `/home/bms/tiniba/tiniba/tiniba.tar.gz`
- Install TINIBA® in `$$TINIBA$$/`
  - `$$TINIBA$$/ > tar -xvf tiniba.tar.gz`
- Make sure that the TINIBA® version you may want to use is set in `utils/version-tiniba.txt`.
- Make sure that the `ABINIT`® version you may want to use is set in `utils/version-abinit.txt`. This information is used in `clustering: all_nodes.sh`, and `runSCF_19_Octubre_2009.sh`, and in `utils/check_abinit.sh`.
- `.bashrc`

  In order to run the shells without having to give all the route include in the file `.bashrc`:

  ```
  export TINIBA=$$TINIBA$$
  export PATH="$$TINIBA$$:$PATH"
  ```

  Remember to read `.bashrc` into the shell, or exit and enter the shell anew.
- a `.bash_profile` is provided at `$TINIBA/utils`. Put this in your `/home/user/` and change at will, but DON'T remove the `abinit` environment variables:
  `export I_MPI_FABRICS=shm:tcp`
  `LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/usr/lib64`

## 1.5 Administration on medusa

Here a list of usefull scripts:

- `./clustering/`run_tiniba.sh
- `./utils/`checkMountQuads.sh
- `./utils/`createRemoteDir.sh
- `./utils/`limpiaSetUpAbinit.pl
- `./utils/`alive.sh
- `./utils/`apagarMedusa_0.sh

- `./utils/`MakeMakefile2010.PL
- `./utils/`compiler.sh

  Here are some useful commands to check the running jobs:
- `cluster_exec -N QUADS ps -u usrname`
- `cluster_exec -N XEON ps -u usrname`
- `cluster_exec -N ITANIUM ps -u usrname`

### 1.5.1 How to : restart InfiniBand communications link

Here you need the superuser password!

1. `$HOME > ssh quad01`
2. `$HOME > su`
3. `$HOME > /etc/init.d/openibd stop`
4. `$HOME > /etc/init.d/openibd restart`
5. To check if infiniband (only quads) is working you can run: infiniband.sh

### 1.5.2 How to : mount homeib

Here you need the superuser password!

For the `quads` proceed as follows:

1. `$HOME > ssh quad01`
2. `$HOME > su`
3. `$HOME > mount master:/home /home`
4. `$HOME > mount quad01ib:/homeib /homeib`
5. `$HOME > exit`
6. `$HOME > exit`

and so on for the other nodes ...

You can check if the mounting is correct with: checkMountQuads.sh

# Chapter 2

# How to run a TINIBA test case

CODE:

"drakgreen" for the TINIBA® commands

"blue" for the input

"orange" for the Linux commands

`verbatim for directories or files`

"gray" for output

`i.e.` is for examples

General Considerations

- The same version of `abinit` is runed in all platforms. If a new version is needed, just compile it, and let everybody know. Include this in `utils/version-abinit.txt` (see 1.4).
- There are three switches, one for the Xeons, one for the Quads and one for the Hexas. Thus is convenient to chose only one platform for the whole run, at least for the SCF cycle. This would take advantage of doing the `mpi` and `rcp` over the `myrinet` or `infiniband` switches.
- `ssh` to any of the `xeon (node01-32)`, `itanium01-04`, `quad01-14` or `hexa1-36` platforms. Then, `cd` to the corresponding `/home`, and work from there.

- The RAID system has the following storage HD's
  `/homea/ /homeb/ /homee/ /homeib/ /homen/ /homeu/`
  where each user must store the work, once finished running in the cluster.
- It is very convenient to call the working directory with a name that relates to the case you are working with.
- Since the SCF uses the parallelized version of `abinit` is better to use only one platform for this cycle.
- If running in the `quads` work in `/homeib/` and for the SCF cylce use at least one `quad01` in `.machines_scf.original`.
- The matrix elements could be calculated in the three plataforms, although for copying the wave function is much faster if it is done through the switches.

The example is a slab of 6 layers mimicking Si(111):As with no spin-orbit coupling.

1. `$PWD > ssh [quad01,hexaN]`
2. If `quad01`
   - `$PWD > cd /homeib/user`
3. If `hexaN`
   - `$PWD > cd /home?/user/` with ?=a,b,e,n,u
4. `$PWD >` creatingTree.sh `si_as_6` (optional)
   if so
   - `$PWD >` cd `si_as_6/abinit-si_as_6/si_as_6/`
     else
   - `$PWD >` cd `si_as_6`

   The `setUpAbinit_si_as_6.in`, `si_as_6.xyz` and `.machines_*.original` files are found in `$TINIBA/examples/surface/nospin`. Copy these files to your working directory and
   - WARNING: The psudopotentials are in
     `$TINIBA/psp`
     Be sure that in that the path to them is correctly set in
     `setUpAbinit_case.in`
   - Check that the input parameters are consistently taken. i.e. with/without spin-orbit coupling, etc.
   - Plot the coordinates to make sure they are correct.

- `.machines_scf.original` for the SCF (only chose one platform!)
- `.machines_pmn.original` for the matrix elements (could chose mixed platforms)
- `.machines_latm.original` for the integrands (only one CPU from any platform)
- `.machines_res.original` for the **k**-integration (chose as many CPU's, from any platform, as tensor components to be calculated)

5. `$PWD >` createRemoteDir.sh (optional)
   - Check that the chosen nodes are working:
     - above removes all the nodes that don't work or exist
     - You may want to remove above nodes from the `*.original` files
6. `$PWD >` cluster_exec -N ALL ps -u username
   - run in `medusa`.
   - So you can check that the nodes chosen are not being used by other user.
7. `$PWD >` abinitCheck.sh 1
8. `$PWD >` abinitCheck.sh 2
   - Surface Calculations: If the slab is centrosymmetric, responses like Surface SHG would be identically zero. To get the correct surface response use
     •`odd_rank.sh`
     This shell takes the original centrosymmetric `case.xyz`, adds a small part to the $z$-coordinate of the bottom surface atom(s), so the structure is non-centrosymmetric. Then, `abinit_check.sh` is run with both options, so the `sym.d` file does not has inversion symmetry. The coordinates are reset to the original centrosymmetric `case.xyz`.
   - Check for the total memory needed and be sure that it fits in the RAM memory:
     This job should need less than ? Mbytes of memory
9. `$PWD >` rklist.sh $N_x$ $N_y$ $N_z$ [abinit or wine2k]
   - $N_i$ must be odd.
   - If in real space $L_x = n \times L_y$ take $N_y = n \times N_x$.
   - For a surface take $N_z = 2$.
   - Do you want to add inversion? (1=yes, 0=no)
     This inversion is the time-reversal symmetry. If your case has it answer 1 otherwise 0
   - `i.e. $PWD >` rklist.sh 19 19 2 abinit
     This generates 64 $N_k$-points in the IBZ.

10. `$PWD >` rlayer.sh (Skip for Bulk a Calculation)
    - Used to generate the layers of the slab
    - Follow instructions
    - Plot to confirm that the layers are ok:
      ```
      gnuplot> p 'case.xyz' u 1:3 w lp
      gnuplot> load 'front.layers.xy'
      gnuplot> load 'back.layers.xy'
      ```
    - `layers.d` is the file with the layers to be calculated.
    - To select any given set of layers
      `$PWD > chose_layers.sh`
      - Follow instructions
      - This is convenient in order to speed up the calculation for slabs with many layers so one chooses only a few important layers.
      - The original layers are kept in `layers.d.original`
11. `$PWD >` run_tiniba.sh -r setkp -k $N_k$ -g $w_I$ -G $w_Q$
    - This distributes $N_k$ points in the nodes given in `.machines_pmn.original`
    - It chooses the nodes that are working and for the `quads` and `hexas` the ones that have `infiniband` as well
    - where $w_{I,Q}$ are the weights for how fast are the Itanium and Quad processors with respect to the Xeon processors. Since it is better to run in only one platform, this is irrelevant, and we may delete it altogether in a future version of TINIBA®
    - `i.e. $PWD >` run_tiniba.sh -r setkp -k 64 -g 2 -G 4
    - 1=Choose another weight or 2=Exit to run the full script
      answer accordingly

12. `$PWD >` run_tiniba.sh -r run -k $N_k$ -N $N_{\text{Layer}}$ -x [s-1 or p-2] options:

```
-w      Wave function(k)
-m      rho(z) for a set of k-points in case.klist_rho
-e      Energies E_{m}(k)
-p      Momentum Matrix Elements p_{mn}(k), includes m=n
-d      Layered ndot Matrix Elements rho_{cc'}(k)
-c      Layered Momentum Matrix Elements calp_{mn}(k)
-l      Layered Diagonal Momentum Matrix Elements calp_{mm}(k)
-s      Spin Matrix Elements S_{cc'}(k)
-n      Layered Spin Matrix Elements calS_{cc'}(k)
-b      bypass WF checkup (Never use on first run)
```

where

- `-k Nk`: is the number of **k**-points or `rho` for the -m option
- `-N N_Layer`: is the number of layers ($\geq 0$)
- `-x [s-1,p-2]` runs the SCF in *s*erial-1 or *p*arallel-2 (recommended)
- Besides `-w` any option *includes* the calculation of the wave function

*WARNING* don't run using options `-c, -l or -n` simultaneoulsy!

13. Tips

- Run in a separate sequence. i.e.: `-w` → `-e -p` → any other one option.
- Use the option `-b` after second run, just to be sure that the `WF` file was correctly copied.
- The file `rvarios.sh` is a shell with all the steps to perform a *whole enchilada* calculation including the response functions. Get one from `$TINIBA/examples/rvarios.sh` and modify it *ad libitum*.
- Other `rvarios-shg.sh` file for SHG to perform a *whole enchilada* calculation is at `$TINIBA/examples/rvarios-shg.sh`. Modify it *ad libitum*.
- run with
  `> nohup rvarios.sh`
  Then kill the terminal so `nohup` works correctly.
- To run *without* coherences for the spin injection go to
  `$TINIBA/latm/SRC_1setinput/integrands.f90`
  comment/uncomment the corresponding lines (look for *without*), compile, rename the response for $\zeta$ and $\xi$ in
  `$TINIBA/utils/responses.txt`
  and run `all_responses.sh`. Then, undo the changes!

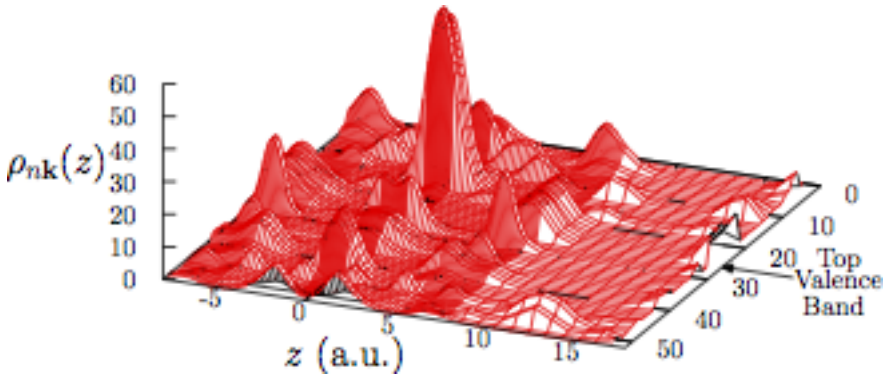


Figure 2.1: $\rho_{n\mathbf{k}}(z)$ for the Si(111)As surface of the example calcuated with spin-orbit interaction.

14. LDA Energy Gap and Scissors correction
    To calculate the LDA gap and the scissors correction run and follow instructions
    - `gap_correction.sh`
15. Monitors:
    - To monitor the status of the run
      - `culster_exec -N ALL 'ps -u user'`
    - To monitor the calculation time and status of the matrix elements use:
      - ontoi.sh
16. If the SCF doesn't run, before you tray again, do
    - `$PWD >` run_tiniba.sh -r erasescf
17. Integrated density $\rho_{n\mathbf{k}}(z) = \int dx \int dy \, |\psi_{n\mathbf{k}}(\mathbf{r})|^2$ (see Fig. 2.1)
    - write or generate `case.klist_rho` with the **k**-points where it needs to be evaluated, typical the $\Gamma$ point and some other few points of maximum symmetry. Modify `.machines_pmn` such that the number of **k**-points is the same as the number of CPU's.
    - run with option `-m` and follow instructions.
18. After you are done with the calculation erase as follows:
    - `$PWD >` run_tiniba.sh -r erase
      to erase the files in all the nodes.

- `$PWD >` run_tiniba.sh -r erasescf
  to erase the SCF calculation.Warning: this erases the wave-function and may be better to keep it for future calculations, perhaps in a back-up HD.

## 2.1 Benchmark Test

Above example is run automatically by using and following the instructions of
- `$PWD >$TINIBA/utils/run-whole-enchilada-surface.sh`

where the output ought to be compared with benchmark results in order to check that TINIBA® is working correctly.

However the calculation of $\chi^{xyz}(2\omega)$ for GaAs is a better benchmark, since it is more sensitive to all the ingredients of the calculation. It is run with
- `$PWD >$TINIBA/utils/run-whole-enchilada-bulk.sh`

# Chapter 3

# Optics

The program to run all the optical responses is
•`$TINIBA/utils/all_responses.sh`
which in turn runs
•`$TINIBA/utils/responses.sh`
To run it just execute
• `$PWD > all_responses.sh`
and follow instructions. In particular one can calculate from all valence bands to any number of conductions bands (`-o 1`), or from a given valence band to a given conduction band (`-o 2`).

- For SHG bulk/surface use response 21/44. Both are coded in the Length Gauge. Response 42 only works for bulk and is coded the Velocity Gauge. Since it has to calculate extra commutators is slower than response 21. Thus use 42 only if you doubt your results and would like to get some reasurance that they are correctly calculated by 21.
- The file for SHG has for columns

| Column → | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| Quantity → | $\hbar\omega$ | Re$[\chi_{ijk}(1\omega)]$ | Im$[\chi_{ijk}(1\omega)]$ | Re$[\chi_{ijk}(2\omega)]$ | Im$[\chi_{ijk}(2\omega)]$ |

Units: pm/V
So far it only works for one component of $\chi_{ijk}$.

- Degree of Spin Polarization (DSP), $\mathcal{D}^a$.
  1. WARNING: the factor of $(\hbar/2)$ is NOT included in $\zeta^{abc}$. However it is included in the expression for $\mathcal{D}^a$, and thus the value of $\zeta^{abc}$ as it comes out of the code must be used to compute $\mathcal{D}^a$. To report the value of $\zeta^{abc}$ one must multiply by $\hbar/2$ and use the appropriate S.I. units $\mathrm{V}^{-2}\mathrm{m}^{-1}\mathrm{J}$.

2. For a surface calculation use a symmetric slab, i.e. same top and bottom surfaces.
3. Use `dsp.sh` to gather the results for the DSP calculation. It does so for bulk (full) or layer-by-layer. The data is written as follows:

| Column → | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| `kk`→ | $\hbar\omega$ | $\xi_{xx}$ | $\xi_{yy}$ | $\xi_{zz}$ | $\zeta_{ijk}$ |

4. `dsp.sh` uses the `kk` response files and not the `sm` files. The onset of the DSP is very sharp and the `sm` files misscalculate it.
5. `gnuplot> p 'dsp-file' u 1:(2*$5/($j+$k+eta)) w l`
where $j$ and $k$ are the corresponding columns (2,3 or 4) according to the $jk$ Cartesian components of $\zeta_{ijk}$. Also, `eta->0`, so *gnuplot* plots the DSP onset properly.
6. If you have calculated `dsp.sh` with the `full` option for a surface, i.e. the whole "bulk-like" structure, then you must get the same result as with the `half-slab` option, since both $\xi^{ab}$ and $\zeta^{abc}$ are twice as much for the whole slab than for the half-slab, but the factor of two cancels out when taking the ratio in the expression for $\mathcal{D}^a$.

## 3.1 Formulas and Units

These are the formulas coded:

- Linear response (See Sipe and Shkrebtii PRB **61**, 5337 (2000) Eq. 34 where $m = c$ and $n = v$ for the resonant condition with $\omega > 0$),

$$\mathrm{Im}[\chi^{\mathrm{ab}}(\omega)] = \frac{\pi e^2}{\hbar}\int \frac{d\mathbf{k}}{8\pi^3}\sum_{vc} r^{\mathrm{a}}_{vc}(\mathbf{k}) r^{\mathrm{b}}_{cv}(\mathbf{k})\delta(\omega_{cv}(\mathbf{k})-\omega).$$

  For the layer response we replace $r^{\mathrm{a}}_{vc}(\mathbf{k}) \to \mathcal{R}^{(\ell)\mathrm{a}}_{vc}(\mathbf{k})$ see Mendoza et al, Phys. Rev. B **74**, 075318 (2006).

- Carrier injection rate, $\dot{n}(\omega) = \xi^{ij}(\omega)E^i(-\omega)E^j(\omega)$, where is better to redefine as $\dot{\tilde{n}} = (\hbar/2)\dot{n} = \tilde{\xi}^{ab}E^a(-\omega)E^b(\omega)$, with $\tilde{\xi}^{ab} = (\hbar/2)\xi^{ab}$.

$$\tilde{\xi}^{\mathrm{ab}}(\omega) = \frac{\pi e^2}{\hbar}\int \frac{d\mathbf{k}}{8\pi^3}\sum_{vc} r^{\mathrm{a}}_{vc}(\mathbf{k}) r^{\mathrm{b}}_{cv}(\mathbf{k})\delta(\omega_{cv}(\mathbf{k})-\omega).$$

  For the layer response:

$$\tilde{\xi}^{ab}(\ell;\omega) = \frac{\pi e^2}{\hbar}\int \frac{d^3k}{8\pi^3}\sum_{vcc'} \frac{1}{2}\mathrm{Re}\Big[\rho_{cc'}(\ell) r^a_{vc} r^b_{c'v} + \rho_{c'c}(\ell) r^a_{vc'} r^b_{cv}\Big]\delta(\omega-\omega_{cv}).$$

We notice that $\tilde{\xi}^{ab}(\omega) = \mathrm{Im}[\chi^{ab}(\omega)]$, with $\epsilon^{ab}(\omega) = 1 + 4\pi\chi^{ab}(\omega)$, however $\tilde{\xi}^{ab}(\ell;\omega) \neq \mathrm{Im}[\chi^{ab}(\ell;\omega)]$!!! Therefore, for a bulk calculation, one must calculate $\mathrm{Im}[\chi^{ab}(\omega)]$ for $\dot{n}(\omega)$.

- Second Harmonic Generation
  - Velocity-Gauge
    The SHG $\chi_{\mathrm{v}}^{\mathrm{abc}}(2\omega)$ is programmed within the velocity-gauge according to Phys. Rev. B **80**, 155205-1-13 (2009).

$$\mathrm{Im}[\chi_{\mathrm{v}}^{abc}(-2\omega;\omega,\omega)] = \frac{\pi|e|^3}{2\hbar^2}\int\frac{d^3k}{8\pi^3}\Big[\sum_{vc}\frac{16}{(\omega_{cv}^S)^3}\Big(\sum_{c'}\frac{\mathrm{Im}[v_{vc}^{\Sigma,a}\{v_{cc'}^{\Sigma,b}v_{c'v}^{\Sigma,c}\}]}{\omega_{cv}^S-2\omega_{c'v}^S}$$

$$-\sum_{v'}\frac{\mathrm{Im}[v_{vc}^{\Sigma,a}\{v_{cv'}^{\Sigma,b}v_{v'v}^{\Sigma,c}\}]}{\omega_{cv}^S-2\omega_{cv'}^S}\Big)\delta(\omega_{cv}^S-2\omega)$$

$$+\sum_{(vc)\neq\ell}\frac{1}{(\omega_{cv}^S)^3}\Big(\frac{\mathrm{Im}[v_{\ell c}^{\Sigma,a}\{v_{cv}^{\Sigma,b}v_{v\ell}^{\Sigma,c}\}]}{\omega_{c\ell}^S-2\omega_{cv}^S}-\frac{\mathrm{Im}[v_{v\ell}^{\Sigma,a}\{v_{\ell c}^{\Sigma,b}v_{cv}^{\Sigma,c}\}]}{\omega_{\ell v}^S-2\omega_{cv}^S}\Big)\delta(\omega_{cv}^S-\omega)$$

$$-\sum_{vc}\frac{1}{(\omega_{cv}^S)^3}\Big(4\mathrm{Re}[\,v_{vc}^{\Sigma,a}\{\mathcal{F}_{cv}^{bc}\}]\delta(\omega_{cv}^S-2\omega)+\mathrm{Re}[\{\mathcal{F}_{vc}^{ab}v_{cv}^{\Sigma,c}\}]\delta(\omega_{cv}^S-\omega)\Big)\Big].$$

    Programs: `shg1v` and `shg2v`
  - Length-Gauge: Layered response by bms-unpublished. See `shg-layer.pdf`

$$\mathrm{Im}[\chi_{e,\mathrm{abc},\omega}^{s(\ell)}] = \frac{\pi|e|^3}{2\hbar^2}\sum_{vc\mathbf{k}}\sum_{l\neq(v,c)}\left[\frac{\omega_{lc}^S\mathrm{Re}[\mathcal{R}_{lc}^{\mathrm{a}(\ell)}\{r_{cv}^{\mathrm{b}}r_{vl}^{\mathrm{c}}\}]}{\omega_{cv}^S(2\omega_{cv}^S-\omega_{cl}^S)}-\frac{\omega_{vl}^S\mathrm{Re}[\mathcal{R}_{vl}^{\mathrm{a}(\ell)}\{r_{lc}^{\mathrm{c}}r_{cv}^{\mathrm{b}}\}]}{\omega_{cv}^S(2\omega_{cv}^S-\omega_{lv}^S)}\right]\delta(\omega_{cv}^S-\omega)$$

$$\mathrm{Im}[\chi_{i,\mathrm{abc},\omega}^{s(\ell)}] = \frac{\pi|e|^3}{2\hbar^2}\sum_{cv\mathbf{k}}\frac{1}{\omega_{cv}^S}\left[\mathrm{Im}[\{r_{cv}^{\mathrm{b}}\left(\mathcal{R}_{vc}^{\mathrm{a}(\ell)}\right)_{;k^{\mathrm{c}}}\}]+\frac{2\mathrm{Im}[\mathcal{R}_{vc}^{\mathrm{a}(\ell)}\{r_{cv}^{\mathrm{b}}\Delta_{cv}^{\mathrm{c}}\}]}{\omega_{cv}^S}\right]\delta(\omega_{cv}^S-\omega)$$

$$\mathrm{Im}[\chi_{e,\mathrm{abc},2\omega}^{s(\ell)}] = \frac{\pi|e|^3}{2\hbar^2}\sum_{vc\mathbf{k}}4\left[\sum_{v'\neq v}\frac{\mathrm{Re}[\mathcal{R}_{vc}^{\mathrm{a}(\ell)}\{r_{cv'}^{\mathrm{b}}r_{v'v}^{\mathrm{c}}\}]}{2\omega_{cv'}^S-\omega_{cv}^S}-\sum_{c'\neq c}\frac{\mathrm{Re}[\mathcal{R}_{vc}^{\mathrm{a}(\ell)}\{r_{cc'}^{\mathrm{c}}r_{c'v}^{\mathrm{b}}\}]}{2\omega_{c'v}^S-\omega_{cv}^S}\right]\delta(\omega_{cv}^S-2\omega)$$

$$\mathrm{Im}[\chi_{i,\mathrm{abc},2\omega}^{s(\ell)}] = \frac{\pi|e|^3}{2\hbar^2}\sum_{vc\mathbf{k}}\frac{4}{\omega_{cv}^S}\left[\mathrm{Im}[\mathcal{R}_{vc}^{\mathrm{a}(\ell)}\{\left(r_{cv}^{\mathrm{b}}\right)_{;k^{\mathrm{c}}}\}]-\frac{2\mathrm{Im}[\mathcal{R}_{vc}^{\mathrm{a}(\ell)}\{r_{cv}^{\mathrm{b}}\Delta_{cv}^{\mathrm{c}}\}]}{\omega_{cv}^S}\right]\delta(\omega_{cv}^S-2\omega)$$

    Programs: `shg1l` and `shg2l` for bulk, i.e. $\mathcal{R}_{vc}^{\mathrm{a}(\ell)} \to r_{vc}^{\mathrm{a}}$.
    `shg1c` and `shg2c` for layered.

- Injection current

$$\eta^{\mathrm{abc}}(\ell|0;\omega,-\omega) = \frac{i\pi e^3}{\hbar^2}\int\frac{d^3k}{8\pi^3}\sum_{vc}\Delta_{cv}^{\mathrm{a}}(\ell;\mathbf{k})\mathrm{Im}\big[r_{cv}^{\mathrm{b}}(\mathbf{k})r_{vc}^{\mathrm{c}}(\mathbf{k})\big]\delta(\omega_{cv}(\mathbf{k})-\omega).$$

  For the bulk response we replace $\Delta_{cv}^{\mathrm{a}}(\ell;\mathbf{k}) \to \Delta_{cv}^{\mathrm{a}}(\mathbf{k})$

- Spin injection

$$\begin{aligned}\zeta^{\rm abc}(\omega) &= \frac{i\pi e^2}{\hbar^2}\int\frac{d^3k}{8\pi^3}\sum_{vcc'}{}' \,{\rm Im}\Big[S^{\rm a}_{c'c}(\mathbf{k})r^{\rm b}_{vc'}(\mathbf{k})r^{\rm c}_{cv}(\mathbf{k}) \\ &+ S^{\rm a}_{cc'}(\mathbf{k})r^{\rm b}_{vc}(\mathbf{k})r^{\rm c}_{c'v}(\mathbf{k})\Big]\delta(\omega_{cv}(\mathbf{k})-\omega).\end{aligned}$$

  Notice that the units of $\zeta^{\rm abc}(\omega)$ have a $\hbar/2$ factor coming from the spin matrix elements $S^{\rm a}_{cc'}$ (recall that $\hat{\mathbf{S}}=(\hbar/2)\hat{\boldsymbol{\sigma}}$), besides the other units. The degree of spin polarization is defined as

$$\mathcal{D}=\frac{2\zeta^{\rm zxy}}{\hbar(\xi^{\rm xx}+\xi^{\rm yy})/2}=\frac{2\zeta^{\rm zxy}}{(\tilde{\xi}^{\rm xx}+\tilde{\xi}^{\rm yy})},$$

  and it is a dimensionless quantity, as it must.

  For the layered response we replace $S^{\rm a}_{cc'}(\mathbf{k})\to\mathcal{S}^{\ell,{\rm a}}_{cc'}(\mathbf{k})$.

- Units.
  - In general,

$$\epsilon_0\chi^j_{\rm S.I.}=\frac{4\pi\epsilon_0}{(3\times10^4)^{j-1}}\chi^j_{\rm c.g.s}\times\frac{m^{j-2}C}{V^j},$$

    with $j$ the order of the response.
  - As an example we work the injection current units:

$$\gamma=\frac{\pi e^3}{\hbar^2}\times\frac{1}{\Omega}\times v\times r\times r\times\frac{1}{\omega},$$

    the first term is the prefactor, the second is the volume of the **k**-integration, the third is the velocity of $\Delta$ the fourth and fifth are the matrix elements of the position operator and the last one comes from de Dirac delta function. Using $r=v/\omega$ unit-wise

$$\begin{aligned}\gamma &= \frac{\pi e^3}{\hbar^2}\times\frac{1}{a_0^3}\times v\times\frac{v^2}{\omega^2}\times\frac{1}{\omega} \\ &= \frac{\pi e^3}{\hbar^2 a_0^3}\times\frac{v^3}{\omega^3} \\ &= \frac{\pi e^3\hbar}{a_0^3}\times\frac{p^3}{m^3(\hbar\omega)^3}=\frac{\pi e^3\hbar}{m^3a_0^3}\times\frac{\hbar^3}{a_0^3}\times\frac{1}{[\rm eV]^3} \\ &= \frac{\pi e^3\hbar^4}{m^3a_0^6}\times\frac{1}{[\rm eV]^3}\times\frac{[27.21\,\rm eV]^3}{H^3} \\ &= \frac{\pi e^3\hbar^4}{m^3a_0^6}\times[27.21]^3\times\frac{a_0^3}{e^6}=\frac{\pi\hbar^4}{e^3(ma_0)^3}\times[27.21]^3 \\ &= \frac{\pi\hbar^4}{e^3(\hbar^2/e^2)^3}\times[27.21]^3=\frac{\pi e^3}{\hbar^2}\times[27.21]^3,\end{aligned}$$

with $a_0 = \hbar^2/me^2$ Bohr's radius, $\hbar\omega$ is measured in eV, $v = p/m$ and $[p] = \hbar/a_0$, the Hartree $H = e^2/a_0 = 27.21$ eV, in c.g.s $e = -4.8066 \times 10^{-10}$ statcoulomb and $\hbar = 1.05457 \times 10^{-27}$ erg·s, then

$$\gamma = -\pi \times [27.21]^3 \times [9.9853 \times 10^{25}].$$

For the injection current we have $j = 2$,

$$\eta_{\text{S.I.}} = \frac{4\pi\epsilon_0}{3 \times 10^4} \eta_{\text{c.g.s}} \times \frac{C}{V^2},$$

and

$$\frac{1}{4\pi\epsilon_0} = 9 \times 10^9,$$

since the units of $\epsilon_0$ are already taken into account, then

$$\eta_{\text{S.I.}} = \frac{1}{27 \times 10^{13}} \eta_{\text{c.g.s}} \times \frac{C^3}{J^2},$$

where $J = VC$. However, $\eta$ gives the injection current, i.e. $dJ/dt = \eta EE$, then there is a second coming from the $J$ and another second coming from the time derivative, which finally give

$$\begin{aligned} \gamma &= -\pi \times [27.21]^3 \times [9.9853 \times 10^{25}] \times \frac{1}{27 \times 10^{13}} \\ &= -\pi \times [27.21]^3 \times [9.9853 \times 10^{25}] \times [3.7037 \times 10^{-15}], \end{aligned}$$

as the prefactor of $\eta$ in $C^3/J^2s^2$. For the surface $\eta_s$ there is a factor of meters since $\eta_s \sim L\eta$, then the units of $\eta_s$ are $mC^3/J^2s^2$. The $L$ factor is set at the `gnuplot` file for the plot.

We remark that for the injection current the bulk prefactor $\gamma_{\text{bulk}} = \gamma/2$ since we use the commutator instead of the imaginary part of $r^{\text{b}}_{cv} r^{\text{c}}_{vc}$ and $i\text{Im}[ab] = [a, b]/2$.

– SHG

$$[\chi] = \frac{\pi}{2} \frac{[e]^3}{[\Omega][\hbar]^2} \frac{[r]^3}{[\omega]^2},$$

– Values: Are given in `cgs` and `mks` units in `~/tiniba/tiniba2010/SRC_response/SRC_set/PhysicalConstantsMod.f90`

# Chapter 4

# Examples

## 4.1 Bulk

For an example copy from `$TINIBA/examples/surface/nospin/gaas`: `setUpAbinit_gaas.in` and `gaas.xyz`. This is a bulk GaAs crystal. See results in `gaas/res` and Fig. 4.1.

## 4.2 Surface

For an example copy from `$TINIBA/examples/surface/nospin/si_as_6`: `setUpAbinit_si_as_6.in` and `si_as_6.xyz`. This is a Si(111)1 $\times$ 1:As surface with 4 layers of Si, one top and one bottom layer of As. See results in `si_as_6/res` and Fig. 4.2.

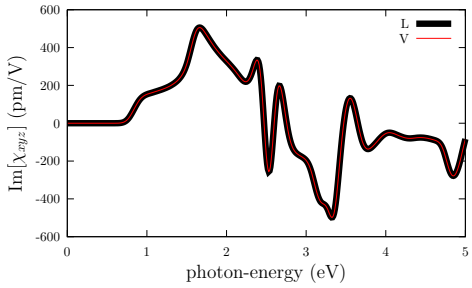


Figure 4.1: $\chi_{xyz}$ for GaAs, $E_{cut} = 20$ Ha and $N_k = 1661$. Both the length ($L$) and velocity ($V$) gauge results are shown. The scissors shift is 1.051 eV.

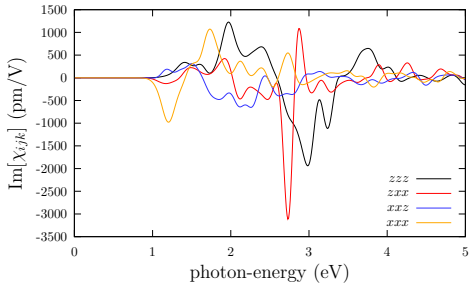


Figure 4.2: Surface $\chi_{ijk}$ for all the allowed components of the si_as_6 example, $E_{cut} = 5$ Ha, $N_k = 64$, and with a scissors shift of 0 eV.

# Chapter 5

# For Developers: Including New Optical Responses

Optical responses are calculated in two steps:

1. the integrand,
2. and the integration over $k$.

Both steps are implemented in FORTRAN code (thanks to Fred Nastos) and both are included with TINIBA. Including new optical subroutines only requires modifying the *integrand*. For more information see [2].

## 5.1 a la Cabellos

This subsection only runs $\chi_{ij}(\omega)$ and $\chi_{ijk}(2\omega)$ for bulk semiconductors. It assumes that the experimental or GW band gap is known. We are planing to extend this version to other responses and the layer-by-layer analysis as well. $\chi_{ijk}(2\omega)$ is calculated in the length and velocity gauges with and without the appropriate scissor correction. See [1] for more information.

- go to:
  `cd $HOME/$TINIBA/SRC_response/SRC_set`
- Make a module. This has "two" names, one is the name of the file itself, i.e. `ishg1la.f90` and the other is the name of the module itself, i.e.

`IntegrandSHG1Mod.f90`, that goes as the first line of the file. The file with the module has two functions with the response prefactor and a "delta function factor" depending if the response is 1-$\omega$ or 2-$\omega$. It also has the subroutine with the integrand that codifies the response to be calculated.
Then you have to link the new module to these next subroutines/modules:

- in `IntegrandsMod.f90`
  - at the top of the file insert the name of the new module, i.e.
    `USE IntegrandSHG1Mod.f90, ONLY :: SHG1`
  - Asign a case nomber, i.e.
    ```
    case (21)
    CALL SGH1(i_spectra)
    ```

  Notice that the case number is arbitrary but is is linked to the `ONLY :: SHG1`

- in `SymmetryOperationsMod.f90`
  - write the *case* number in the appropriate symmetry *transformation*, i.e.
    ```
    CASE(21,22,26,27,28,29,30,60,61,62,63,64,65,80,81)
    CALL transformationSecondOrderResponse(i_spectra)
    ```

  since the example is for SHG which is a second order response given by a third rank tensor.

- in `SpectrumParametersFileMod.f90`
  - chose the tensor rank and its total number of Cartesian components, i.e.
    ```
    CASE(21,22)
     WRITE(6,*) "Second-harmonic generation: Length Gauge"
     dims  = 3
     length  = 27
    ```

  since the rank is 3 and $3^3$=27

- in `FileControlMod.f90` include
  - `USE IntegrandSHG1Mod, ONLY : SHG1_factor, SHG1DeltaFunctionFactor`
  - ```
    CASE(21)
    rtmp2  = rtmp*SHG1_factor()
    iTmp = SHG1DeltaFunctionFactor()
    ```

When you finish, you have to compile in the `master`, then you can make use of the advanced tool MakeMakefile.PL (no one uses it) :
    ```

1. `./utils/`MakeMakefile2010.PL
2. `./utils/`compiler.sh
   and follow instructions... the executable is in the previous directory.

After this a script has to driver this subrutine, there are several of them if you want to use one of this:
You have to include the number of response in those scripts
/home/$USER/$TINIBA/SRC_response/responseSHG.sh
In this script include a line like this

```
NAMERES[1]=chi1  ;SIZERES[1]="2"
```

As you see one of this is the name of the subroutine and other is the lenght of the tensor
`$HOME/$TINIBA/SRC_response/menu.sh` In this script include a line like this

```
printf "\t${GREEN}22${NC} ${BLU}${NAMERES[22]}${NC}
(${GREEN}Length 2 omega${NC})\n"
```

In order to be able to run, try it with small case. if you are not able to run ask to Bernardo en the coffe break in order to solve the problem.

## 5.2 Programs for Bulk and Layer-by-Layer response

Do all the compilation in the `master` node and in both `SRC_1setinput` and `SRC_2latam`.

There are two main steps in the calculation of the optical response:

1. the first one sets up the integrand and the files are in
   `$TINIB/latm/SRC_1setinput`
2. the second one does the **k**-integration and the files are in
   `$TINIBA/latm/SRC_2latam`

- `$TINIBA/latm/SRC_1setinput`

  1. in `integrands.f90`
     include a CASE and the name of the subroutine, i.e

```
CASE(21)
CALL SHG1L
```

then write the subroutine itself. You can follow the structure from any other subroutine that works, like SHG1L.

2. in `inparams.f90` do the following
   - in the `MODULE inparams` add the response prefactor as a parameter, i.e.
     `REAL(DP),PARAMETER::shg1_factor` $= \cdots$
   - in the `MODULE inparams` add the response prefactor in the array `spectrum_factor(number_of_known_spectrum_types)`=

     ```
     ⋮
     !    21           22          23
       shg1_factor, shg2_factor, leo_factor,···
     ⋮
     ```

     where it is **very important** that the number chosen for the `CASE` of the response, i.e. 21, coincides with the position-entree in the array.
   - in the `SUBROUTINE readSpectrumFile` add the `CASE` number in the appropriate response, where `dims` is the rank and `length` is the total number of Cartesian components of the tensor's response, i.e.

     ```
     ⋮
      CASE(21,42,44)
       WRITE(6,*) "Second-harmonic generation"
       dims = 3
       length = 27
     ⋮
     ```

     since the rank is 3 and $3^3$=27
   - in the `SUBROUTINE deltaFactor` add the correct `deltaFunctionFactor`=1 or 2 for 1-$\omega$ and 2-$\omega$ transitions, respectively. The default value is `deltaFunctionFactor`=1 for 1-$\omega$ response terms. Otherwise include the response `CASE` number, i.e.,

     ```
     ⋮
      CASE(22,43,45)!shg 2-omega terms
        deltaFunctionFactor=2
      END SELECT
     ⋮
     ```

     since `shg2` is a 2-$\omega$ response term.
3. in `set_input_ascii.f90` the `'all'` the `'standard'` and the `'layer-by-layer'` position and momentum matrix elements are set

into arrays so they can be used by the `intergands.f90` file. Most of the responses already have the required matrix elements so this file should not be changed unless new matrix elements are needed. Check that the momentum-matrix elements are properly renormalized due to the scissors shift.

4. in `symmetry_operations.f90` include the `CASE` number of the response in the appropriate transformation (linear, secon-order, etc.), i.e.

   ⋮

   ```
   CASE(21,22,42,43,44,45)
    CALL transformationSecondOrderResponse(i_spectra)
   ```

   ⋮

   The symmetries are obtained from the file generated by the LDA program and they obtain the following array

   $$R_{ij\cdots} = \sum_{\mathrm{ab}\cdots} M_{i\mathrm{a}} M_{j\mathrm{b}} \cdots , \tag{5.2.1}$$

   where the *italic* Cartesian subindices of $R_{ij\cdots}$ are chosen by the tensor component to be calculated, `-t` option, and the roman Cartesian subindices of the rotation matrices $M_{i\mathrm{a}}$ are summed over.

   The following four files basically do not need to be changed.

5. in `arrays.f90` the arrays are allocated and de-allocated for things like the momentum matrix elements. So far it has all the arrays needed for many responses, so most likely there is no need to edit this file. The file also reads the LDA energies and scissored energies with the value of the provided scissor correction.
6. `constants.f90` has only constants and there is no need to modify it.
7. `file_control.f90` controls the flow of the calculation, there is no need to modify it.
8. `functions.f90` calculates the standard and layer-by-layer position matrix elements, the generalized derivatives of the position matrix elements, and other goodies not required. There is no apparent need to modify it unless one wants to change the layer-by-layer response.
9. Compilation
   - run `compila_all.sh` to compile in the three plataforms
   - the executable is one directory down, i.e.
     `../set_input_32b` for the `Xeon`
     `../set_input_64b` for the `Itanium`
     `../set_input_quad` for the `quad`

  - you may want to change the compiler and compilation flags in:
    `Makefile32b` for the `Xeon`
    `Makefile64b` for the `Itanium`
    `Makefilequad` for the `quad`

- `$TINIBA/latm/SRC_2latam`

  1. `inparams.f90` is the same as that of
     `$TINIBA/latm/SRC_1setinput`
     it is linked so if you change it in `SRC_1setinput` you must compile in `SRC_2latam`!
  2. `tetra_method.f90`
     - does the integration using the Tetrahedral Method.
     - you may want to play with the `SUBROUTINE Which_Transitions` to chose particular transitions, i.e. look in `tetra_method_vc.f90`, but be sure that is implemented for 2-$\omega$ terms.
  3. `constants.f90` has only constants and there is no need to modify it.
  4. `globals.f90` global statements and there is no need to modify it.
  5. `piksort.f90` there is no need to modify it.
  6. Compilation
     - run `compila_all.sh` to compile in the three plataforms
     - the executable is one directory down, i.e.
       `../tetra_method_32b` for the `Xeon`
       `../tetra_method_64b` for the `Itanium`
       `../tetra_method_quad` for the `quad`
     - you may want to change the compiler and compilation flags in:
       `Makefile32b` for the `Xeon`
       `Makefile64b` for the `Itanium`
       `Makefilequad` for the `quad`

- Name for the Response

  The name of the response (i.e. `rhomm`) must go in the following files complying with the particular context

  - in `$TINIBA/utils`
    1. `all_responses.sh`
    2. `responses.sh`
  - in `$TINIBA/latm/SRC_1setinput`
    1. `arrays.f90`
    2. `inparams.f90`
    3. `set_input_ascii.f90`
  - To include a new response once is coded as explained in Sec. 5.2

  * edit `$TINIBA/utils/responses.txt` and add the number and name of the respnse, perhaps you may want to modify `$TINIBA/utils/print_responses.pl` for fine tunning of the displayed text.
  * NOTE:The names that appear in `responses.txt` are the ones used, `verbatim`, for the name of the calculated responses. Thus for a different name for the same response just modify this file!

## 5.3 Shell for Bulk and Layer-by-Layer response

- To run the optical responses execute and follow the instruction of the bash `shell`:

  `$TINIBA/utils/all_responses.sh`
  which uses
  `$TINIBA/utils/responses_bms.sh`

- To include a new response once is coded as explained in Sec. 5.2

  – edit `$TINIBA/utils/responses.txt` and add the number and name of the respnse, perhaps you may want to modify `$TINIBA/utils/print_responses.pl` for fine tunning of the displayed text.

- The SHG is calculated with `CASE`=21 and does both 1-$\omega$ and 2-$\omega$ terms.

# Chapter 6

# Miscellaneous

## 6.1 Wien2k

In general you can use a all electron (wien2k) or pseudopotential (abinit) codes in order to calculate the first and second order response, you can follow the next seteps to achive the result using wien2k code. Now from here the user is going to be: usrname but it could be anybody, then take care of this.

- /home/usrname/temporal/GAAS/wien2k-GAAS/GAAS ¿cd /temporal/
- /home/usrname/temporal ¿ /$TINIBA/wien2k/creatingTree.sh GAAS
- /home/$USER/$TINIBA/wien2k/creatingTree.sh

```
$HOME/temporal >~/\$TINIBA/wien2k/creatingTree.sh GAAS
    ====================
    ====================
    Making tree to run abinis and wien2k
    $HOME/temporal/GAAS/abinit-GAAS/GAAS
    $HOME/temporal/GAAS/wien2k-GAAS/GAAS
    ====================

    ====================
$HOME/temporal/GAAS
   +---abinit-GAAS
      |   +---GAAS
   +---wien2k-GAAS
      |   +---GAAS
    ====================
```

- $HOME/temporal >cd $HOME/temporal/GAAS/wien2k-GAAS/GAAS
  $HOME/temporal/GAAS/wien2k-GAAS/GAAS >

- /home/$USER/$TINIBA/wien2k/wien2k.sh

```
Usage:
    $HOME/\$TINIBA/wien2k/wien2k.sh  [option-0]  [option-1]
    [option-0] = scf    -run ONLY scf
    [option-0] = klist  -generate rklist
    [option-0] = run    -run ONLY momentum and energy  matrix elements
    [option-0] = help   -example files for si bulk
    [option-0] = clean_lapw -remove unnecessary files
    [option-0] = whatneed   -what files I need
    [option-1] = 0.001      -energy convercence LIMIT (0.0001 Ry)
    ==================
    $HOME/\$TINIBA/wien2k/wien2k.sh scf 0.001
    $HOME/\$TINIBA/wien2k/wien2k.sh klist
    $HOME/\$TINIBA/wien2k/wien2k.sh run
    $HOME/\$TINIBA/wien2k/wien2k.sh clean_lapw
    $HOME/\$TINIBA/wien2k/wien2k.sh help
    Stoping right now...  wien2k.sh
```

## 6.2 Refinement over k-points

If you want to refine the density of k-points over one window of energy you can follow this script

- /home/$USER/$TINIBA/refine.sh

```
Usage:
$HOME/\$TINIBA/refine.sh abinit first
$HOME/\$TINIBA/refine.sh abinit refine noscissors
$HOME/\$TINIBA/refine.sh abinit refine scissors
Stoping right now...
$HOME/\$TINIBA/refine.sh
I need input Args
```

## 6.3 The GW method and Bethe-Salpeter Equation

to be done

## 6.4 Surfaces

*God made solids ...*
*but surfaces were the work of the devil*, W. Pauli

## 6.5 Contributors

This list of contributors is not complete, if there are others let us know

- Bernardo Mendoza
- JL Cabellos
- Tonatiuh Rangel
- Cuahutemoc Salzar
- Norberto Arzate
- Fred Nastos

# Appendix A

# Extras

1. Compiling IBZ: executable `ibz.medusa` or `ibz.quad`

XEON: `$$TINIBA$$/src_ibz> make clean -f Makefile_xeon` `$$TINIBA$$/src_ibz> make -f Makefil`

QUAD: `ssh quad01`
`$$TINIBA$$/src_ibz> make clean -f Makefile_quad` `$$TINIBA$$/src_ibz> make -f Makefil`

2. Compiling matrix elements: executable `rpmns.itanium` or `rpmns.quad` or `rpmns.xeon`

   - `$TINIBA/matrix_elements/ver3.0> compilerRPMNS.sh`
     Follow instructions to compile in the three plataforms.

3. Old way to calculate Second Order Response (SHG)

   Here you must to run responseSHG.sh with option number 21 and 22 for lenght gauge formalism and 64 and 65 for velocity gauge formalism and you must provide the LDA band gap in order to calcuate zero scissors correction and you must provide experimental band gap in order to calculate the SHG with scissors correction. The typical value of semar is 0.15 eV, in the case of GaAs the tensor componet different of zero is xzy. when you finish with the glue.sh script, if everething is ok, you get a file that contains in the first row energy in eV, the second one the real part of 1 $\omega$ contribution of lenght, the third one the imaginary part of 1 $\omega$ contribution of lenght, the 4 one the real part of 2 $\omega$ contribution of lenght, the 5 one the imaginary part of 2 $\omega$ contribution of lenght, and the others four rows (6-9) contains the SHG for velocity formalism. (6=Re,1 $\omega$), (7=Im,1 $\omega$), (8=Re,2 $\omega$) and (9=Im,2 $\omega$)
   If you want to plot the $|\chi_2^{xyz}|$ for lenght gauge formalism in the case of GaAs, the typical line text in gnuplot is

   ```
   p 'shg1l_shg2l_shg1t_shg2t_xxz_sm_0-150_3876_10-
   ```

```
nospin_32_N_0-849_0-0d0_0-0d0L_0-0d0t_SRC_response'
u 1:(sqrt(($2+$4)**2+($3+$5)**2)) smooth csplines title '$\chi^2_{xyz}$'
w l lw 2
```

- `$HOME/$TINIBA/SRC_response/`responseSHG.sh
- `$HOME/$TINIBA/SRC_response/all/`doNuevoSmearingt_juniot_14t_2010.sh
- `$HOME/$TINIBA/1response/`glue.sh

INDAUTOR
Instituto Nacional del Derecho de Autor

**CERTIFICADO**

**Registro Público del Derecho de Autor**

SECRETARÍA DE EDUCACIÓN PÚBLICA
SEP

Para los efectos de los artículos 13, 162, 163 fracción I, 164 fracción I, 168, 169, 209 fracción III y demás relativos de la Ley Federal del Derecho de Autor, se hace constar que la OBRA cuyas especificaciones aparecen a continuación, ha quedado inscrita en el Registro Público del Derecho de Autor, con los siguientes datos:

| | |
|---|---|
| **AUTORES:** | CABELLOS QUIROZ JOSE LUIS<br>MENDOZA SANTOYO BERNARDO<br>RANGEL GORDILLO TONATIUH |
| **TITULO:** | TINIBA |
| **RAMA:** | PROGRAMAS DE COMPUTACION |
| **TITULAR:** | CENTRO DE INVESTIGACIONES EN OPTICA, A.C. (ARTICULO 103 PARRAFO PRIMERO) |

L.F.D.A.- Artículo 168.- Las inscripciones en el registro establecen la presunción de ser ciertos los hechos y actos que en ellas consten, salvo prueba en contrario. Toda inscripción deja a salvo los derechos de terceros. Si surge controversia, los efectos de la inscripción quedarán suspendidos en tanto se pronuncie resolución firme por autoridad competente.

**Número de Registro:** **03-2009-120114033400-01**

**México D.F., a 11 de diciembre de 2009**

**EL SUBDIRECTOR DE REGISTRO DE OBRAS Y CONTRATOS**

**ARTURO NOE CALDERON AGUILAR**

ESTADOS UNIDOS MEXICANOS
SECRETARIA DE EDUCACION PUBLICA
INSTITUTO NACIONAL
DEL DERECHO DE AUTOR
REGISTRO PUBLICO

Puebla 143, Col. Roma, Del. Cuauhtémoc, México, D. F., 06700
t.+52(55) 3601 8210, 3601 8216 ó (01800) 2283 400, www.sep.gob.mx

Figure A.1: INDAUTOR Certificate of TINIBA®

# Bibliography

[1] J. Cabellos, Bernardo Mendoza, M. Escobar, F. Nastos, and J. Sipe. Effects of nonlocality on second-harmonic generation in bulk semiconductors. *Physical Review B*, 80(15), October 2009. 5.1

[2] J. E. Sipe and A. I. Shkrebtii. Second-order optical response in semiconductors. *Physical Review B*, 61(8):5337, 2000. 5