User Guide to the *GA95* Program

v2.1

Online: June 2002

David F. Plusquellic

Optical Technology Division
National Institute of Standards and Technology
Gaithersburg, MD 20899-8441

A. *Overview of Ga95.exe and the Jb95.exe interface*

The Genetic Algorithms (GA) dialog in Jb95.exe is used as a front end to the console based program Ga95.exe. The purpose of the dialog is to allow the user convenient access to the parameters of the GA and to permit the evaluation of its performance in real time. All parameters needed to run Ga95.exe appear in the input file Ga95.gai and therefore, the program may be run directly without the Jb95.exe front end. The source code and GA library are compatible with both UNIX and Windows platforms.

The overall strategy implemented in the GA is similar to that reported by Meerts, et al. [1]. The C++ GA library used is one developed at MIT [2]. The GA uses an overlapping population scheme, where the size of the parent population is specified in the **POP SIZE** field and the number of new individuals (children) generated with each generation is specified by **REPL SIZE**. For the first generation, scores are computed for all individuals in the parent population. In subsequent generations up to **NUM GENS**, scores are computed for the children only except where noted below. The best-fit individuals have the smallest scores and the new parent population on each generation is selected from among the elite (best scoring individuals) of the parents and children.

The two parents for each new child are randomly selected from the population. The parameter values (genes) for the child are generated by a blend cross over scheme which operates in the following manner. For a given gene, P_C, a real number is randomly selected from within a range of values between the two parent values, P_i and P_j, plus and minus half this difference, i.e.,

$$P_C = P_i - (P_j - P_i)/2 \text{ to } P_j + (P_j - P_i)/2 \text{ for } P_i < P_j$$

Note that exceptions to this rule occur since the parameter range specified by the user is strictly enforced.

Both the cross over and mutation probabilities are specified with real values from 0 to 1 inclusive. Note, however, that the current GA library has no mechanism for keeping identical individuals (clones) out of the population. Therefore, the cross over probability is usually set to 1.0. A future release of this software will check for and remove clones. The mutation scheme selects a random value from within the full range of the parameter specified by the user. Therefore, the mutation probability one uses is typically small (<0.05).

B. *Initial setup*

The simplest way to begin using the GA is to first enter parameters into the ROTOR CALCULATION OPTIONS dialog and to display the predicted spectrum together with the experimental data. For most parameters, estimates of the maximum and minimum values of the range may be obtained using the trackbar dialogs. In the GA dialog, three fields are given for each of the parameters in the GA dialog. The first two are used to specify the lower and upper values of the range for each parameter. The third field is the initial value and may be copied from the ROTOR CALCULATION OPTIONS dialog by using the key marked **IAR->GA**. The initial values are only to aid the user in selecting a proper range for each of the parameters (although they also must be valid). All initial scores in the population are computed from parameter values randomly selected from within the parameter ranges. After starting the GA, the population distribution of values of each gene may be graphically displayed as discussed below.

Once the GA parameters have been set, it is usually a good idea to write the **GA INPUT FILE** to disk. Before writing the input file, it is customary to change the name of the **BASE OUT FLNM**. For example, if the **BASE OUT FLNM** is TropGA2 and the input file is written and then read, the **SCORE OUT FLNM** and **POP FLNM** fields will be updated to TropGA2.gao and TropGA2.gap, respectively. The input file should also be written after the GA finishes a run.

C. *Experimental data*

The experimental data file must be an x:y file containing an ordered list of frequency:intensity pairs. Furthermore, the intensities must be defined on even frequency intervals. The experimental data may be made available to the GA in two different ways. If the **DISK** status box is unchecked, the ascii x:y file will be created based on the experimental data that is currently being displayed in the **ACTIVE** channel. If the experimental data is being viewed at a shrunken or condensed level of view, the user will be queried to use the averaged data. If answered NO, the data will be written at the full experimental resolution. If the displayed data is offset and scaled, the user should answer YES to the next question if these factors are to be applied before creating the ascii data file. Before the GA is started, the user may also delimit the region in the spectrum using the **M->** and **<-M** keys from the main window display. Checking the limit status (**STAT**) box directs the program to write only the data within the marked section to the ascii file. If the ascii file exists, the **DISK** status box should be checked otherwise the current file will be overwritten. It should be keep in mind that the experimental offset is an important factor in the scoring function. The experimental offset is currently not defined as a GA parameter but will appear in a future release.

D. *Score functions and Operational modes of the GA*

The score function **XCOR** is the cross correlation function defined by Meerts, et al. [1] and is the only one currently recommended for use.

The GA may be run in one of three different ways depending of the value of the **EXACT UPDATE** parameter.

1. *Exact scoring* (**EXACT UPDATE** = 1)

When the **EXACT UPDATE** parameter is set to 1, the rotor program will diagonalize the Hamiltonian matrix, calculate transition intensities and create a convoluted spectrum from the line set for each set of parameters generated by the GA. This scheme results in exact scores always being computed for all new individuals.

2. *Approximate Scoring* (**EXACT UPDATE** > 1)

A second approach that has been implemented is based on an approximate scoring scheme and in many cases will produce identical results and result in shorter run times. The method is based on the computing approximate scores based on the energy derivatives and line strengths calculated for a prior individual in the population. For example, if this parameter is set to 10, then every 10th individual in the population will have its score computed exactly. Exact scores in the parent population are recorded to prevent reevaluation in subsequent generations and are indicated in the output with an "E" label. The energy derivatives and line strengths will be used to compute scores for the next 9 individuals to follow. A drawback to this approach is that the whole population must be reevaluated (population + replacement size) on each generation (as opposed to just the children in the exact scoring scheme) This is done to continuously improve the scores in the parent population. For the initial generations, the scores are only poorly approximated and serve to maintain some diversity in the population. As the population converges, all scores converge to their exact values. This approach should be used with caution when parameter ranges are large and/or when quantum interference effects associated with internal rotation or axis rotation are present [3].

A known problem occurs when an approximate score is better than its exact value. If the invalid score remains the best score over the generations to follow, the parameters values of the best individual will not be updated properly in the GA dialog (see options discussed below). In this case, it is best to obtain the average parameter values. This condition is easily recognized using the features described below and will be fixed in a future release.

3. *Exact Scoring* (**EXACT UPDATE** =0)

This method should be used when the GA is run with a parameter set that includes one or more of the following variables,

T1, TS, T2 Wt, Lorentzian Width, Gaussian Width, TM (a-bc) and TM (b-c)

In this case, one exact calculation is performed and scores that follow are based on this one set of calculated line strengths. All scores are exact. This type of calculation is typically run after all Hamiltonian parameters have been well determined (through least squares fitting to an assigned line set, for example).

E. *Real-time evaluation of GA performance*

A few key functions are provided to follow the progress of the GA. Since the GA runs as a separate console based program, both ascii and binary files are used for interprocess communication between the Ga95.exe and Jb95.exe. When the GA is started, a generic input file is written called Ga95.gai. This file gets updated after each generation with the parameter values of the best individual in the population. Upon depressing the **UPDATE GA PRMS** key, these parameters are displayed in the **VALUE** fields of the GA dialogs. Depressing **GA->IAR** copies these parameters (except for the line shape factors) over into the ROTOR CALCULATION OPTIONS dialog. A new simulation may be generated based on these new parameters to directly compare with the experimental data. The **GET AVE PRMS** button will compute the average parameter values in the population.

The best, worst and average scores of each generation are written to an ascii file specified in the **SCORE OUT FLNM** dialog box. (The best individual's parameters are also recorded in this file.) These three scores are displayed as a function of generation in a separate window when the **SHOW SCORE** box is checked.

For each generation, the parameter values of the entire population are written out to a binary file specified in the **POP FLNM** dialog box. To examine the statistics of a parameter in the population, select the desired parameter (gene) from among those listed in the scrollable list dialog box. Upon checking the **SHOW POP** box, the distribution of the parameter values will be displayed in a separate window over the full range of the parameter. The range for the parameter will be divided by the number of pixels displayed along the abscissa. The ordinate will show the number of individuals that have values in each of these bins. If the **SELECT DIST** box is unchecked, the parameter values will be displayed on the ordinate as a function of each individual in the population along the abscissa.

Since the last population is always saved to disk, one may restart the GA using the old population by checking the **OLD POP STAT** box before starting the GA. At this time, the same set of parameters must be used on restarts.

F. *Description of GA parameters*

Most of the parameters used in the GA are exactly the same as those appearing in the ROTOR CALCULATION OPTIONS DIALOG. A few of them are treated differently by the GA and deserve some mention.

The transition moment orientation may be specified in one of two ways. The percentage values are the magnitudes of the TM components along each of the principle axes. Alternatively, two Euler angles may be used to specify the TM orientation. The GA, however, uses only the Euler angle method and therefore, the range must be specified in terms of angle. To aid the user in selecting the proper angles, values may be entered in either set of fields which gray differently depending on the set selected.

The inertial defect is an alternative method for specifying one of the rotational constants used in the GA. To enable this feature, the variation status of either A, B or C must remain unchecked. A future release will include the option to vary B+C and B-C.

The two temperature model requires that T_2 remain larger than T_1. To properly encode this into the GA, T_2 is defined as T_S*T_1. Therefore, the lower limit for T_S should be >1.

If nuclear spin weights are used, errors will be incurred when perturbations terms are large enough to switch the normal order of the K_a and K_c quantum levels.

Refinement of the line shape parameters are best performed in final iterations using small cross correlation values. It is also noted that the values of the Lorentzian and Gaussian widths and number of line widths (**# WIDTHS**) define the size of regions added to the beginning and end of the data arrays for the cross correlation function.

REFERENCES:

1. J. A. Hageman, R. Wehrens, R. de Gelder, W. L. Meerts and L. M. C. Buydens, J. Chem. Phys. 113 (2000) 7955.

2. Copyright (c) 1995-1996 Massachusetts Institute of Technology, all rights reserved

Permission to use, copy, modify, and distribute this software and its documentation for any non-commercial purpose and without fee is hereby granted, provided that

- the above copyright notice appear in all copies
- both the copyright notice and this permission notice appear in supporting documentation
- the name of M.I.T. not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

M.I.T. makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

M.I.T. DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL M.I.T. BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.