GLoBES — General Long Baseline Experiment Simulator

Joachim Kopp

Max Planck-Institut für Kernphysik, Heidelberg, Germany

GDR Neutrino Meeting
14 – 15 October 2008
Marseille

in collaboration with P. Huber, M. Lindner, M. Rolinec and W. Winter
P. Huber, M. Lindner, W. Winter, hep-ph/0407333
P. Huber, JK, M. Lindner, M. Rolinec, W. Winter, hep-ph/0701187

www.mpi-hd.mpg.de/∼globes

globes@mpi-hd.mpg.de

MAX-PLANCK-GESELLSCHAFT

Outline

Outline

Motivation

Neutrino oscillation physics is entering the stage of precision measurements.

We face many new and old challenges:

- Determining the optimal experimental strategy (which combination of experiments gives the best sensitivity?)
- Disentangling correlated parameters
- Understanding the effects of “new physics” on neutrino oscillations
- Systematical errors become more and more important as statistics improve
- …

$\Rightarrow$ We need a “physics” simulation tool, complementary to the detector simulations.

GLoBES history

2004	Development started, based on fragments of code developed earlier (P. Huber, M. Lindner, W. Winter)
Aug 2004	First release (v2.0.0) P. Huber, M. Lindner, W. Winter, hep-ph/0407333
Mar 2005	Major bugfix release (v2.0.11)
Jul 2005	M. Rolinec and JK join the team
Jan 2007	New major release (v3.0), introducing new features and internal improvements P. Huber, JK, M. Lindner, M. Rolinec, W. Winter, hep-ph/0701187

Design goals of GLoBES

Flexibility:	Suitable for many different types of experiments Suitable for many different physiscs scenarios $\rightarrow$ Modular approach
Reliability:	As bug-free as possible ($\rightarrow$ open source, extensively tested, feedback from users) Reproducible result ($\rightarrow$ non-MC simulation)
Efficiency:	The faster a code is, the more complicated problems can be solved
Documentation:	Required for obvious reasons

Outline

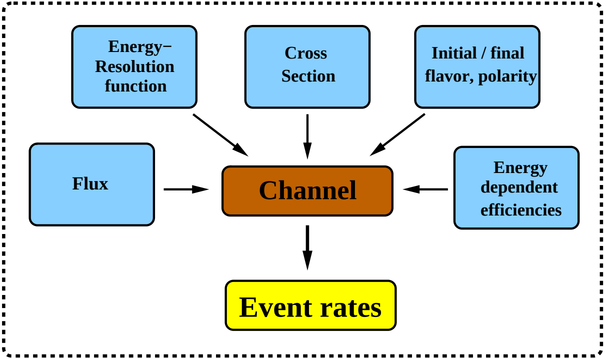
Experiment definition in GLoBES

In GLoBES, experiments are defined using AEDL, the Abstract Experiment Definition Language. AEDL files specify

- Source types and spectra
- Matter density profiles
- Cross sections
- Detector properties: Efficiencies, resolutions, backgrounds, . . .
- Systematical uncertainties

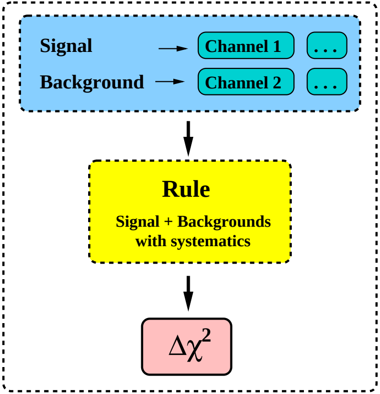
Structure of an AEDL file: Channels

A channel corresponds to oscillations of neutrinos from one flavour into another.



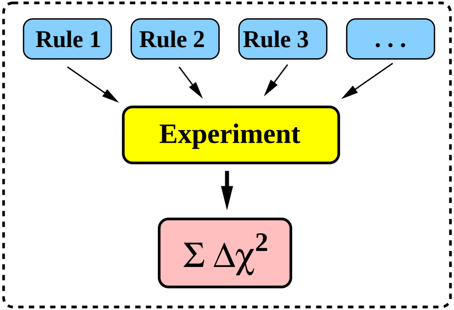
Structure of an AEDL file: Rules

A rule is a combination of signal and background channels, which form an experimental data sample.
Example: ν_e appearance from $\nu_\mu \rightarrow \nu_e$ oscillations in a superbeam, with contamination from $\nu_e \rightarrow \nu_e$.
A rule definition in AEDL also specified associated systematical errors.



Structure of an AEDL file: Experiments

Several rules form an experiment. Several experiments can be handled simultaneously.



Simulation of oscillation physics in GLoBES

- Full three-flavour treatment
- Arbitrary (non-adiabatic) matter profiles
  - PREM profile hard-coded
  - User can choose approximations to PREM, or define completely new profiles
- High numerical efficiency
  Specifically designed numerical algorithms JK, physics/0610206
- Extensibility
  User can modify or replace the GLoBES oscillation engine
  $\rightarrow$ inclusion of arbitrary "new physics"

χ^2 analysis

- Cuts and projections of the multi-dimensional χ^2 manifold ("marginalization")
- Inclusion of systematical uncertainties (fully customizable)
- Inclusion of correlations and degeneracies
- Inclusion of external priors (fully customizable)
- Supports setups with Multiple sources and multiple detectors
- Excellent numerical efficiency

Typical χ^2 function

$$\chi^2(\vec{\lambda}, \vec{a}) = 2 \sum_{\text{exp's}} \sum_{\text{rules}} \sum_{\text{bins}} \left[N^{\text{th}}(\vec{\lambda}, \vec{a}) - N^{\text{obs}} + N^{\text{obs}} \log \frac{N^{\text{obs}}}{N^{\text{th}}(\vec{\lambda}, \vec{a})} \right]$$

$$+ \chi^2_{\text{prior}}(\vec{\lambda}) + \chi^2_{\text{pull}}(\vec{a}),$$

N^{obs}	Observed (simulated) event rates
N^{th}	Theoretically predicted event rates
$\vec{\lambda}$	Oscillation parameters
$\vec{a}$	Systematical biases
$\chi^2_{\text{prior}}(\vec{\lambda})$	External input from other experiments
$\chi^2_{\text{pull}}(\vec{a})$	Restricts systematical biases to lie within errors bounds

GLoBES built-in χ^2 functions

χ^2 function	Analysis	Systematical biases/errors	@errordim (deprecated)
chiSpectrumOnly	Spectral	signal normalization	7
chiNoSysSpectrum	Spectral	—	2
chiSpectrumTilt	Spectral	signal/bckgnd. normalization and spectral “tilt”	0
chiSpectrumCalib	Spectral	signal/bckgnd. normalization and energy calibration error	9
chiTotalRatesTilt	Total rates	signal/bckgnd. normalization and spectral “tilt”	4
chiNoSysTotalRates	Total rates	—	8
chiZero	$\equiv 0$	—	—

Outline

User-defined systematics

Define an arbitrary χ^2 function

```
int glbDefineChiFunction(glb_chi_function chi_func,
    int dim, const char *name, void *user_data)
```

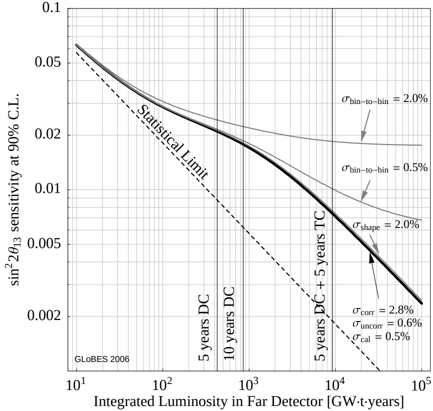
Regsiter it at runtime

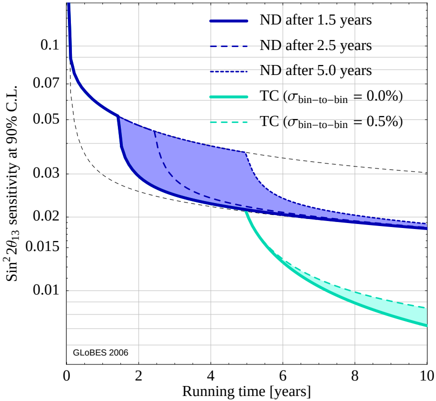
```
double my_chi_function(int exp, int rule, int n_params,
    double *params, double *errors, void *user_data)
```

Refer to it in AEDL by name

```
rule(#rule0)<
  ...
  @sys_off_function = "chiNoSysSpectrum"
  @sys_off_errors   = {}
  @sys_on_function  = "my_chi_function" // χ² for Double Chooz
  @sys_on_errors    = {0.02,0.006,  0.006,  0.005,    0.005    }
>                   //{Flux,Norm FD,Norm ND,Energy FD,Energy ND}
```

θ_{13} sensitivity of next-generation reactor experiments





P. Huber, JK, M. Lindner, M. Rolinec, W. Winter, hep-ph/0601266

Simulation of new physics

Write a user-defined oscillation engine . . .

```
int glb_probability_matrix(double P[3][3], int cp_sign,
    double E, int psteps, const double *length,
    const double *density, double filter_sigma, void *user_data)
```

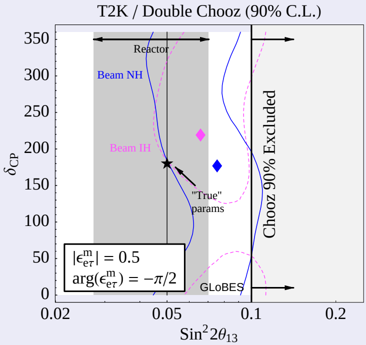
. . . and register it at runtime.

```
int glbRegisterProbabilityEngine(int n_parameters,
    glb_probability_matrix_function prob_func,
    glb_set_oscillation_parameters_function set_params_func,
    glb_get_oscillation_parameters_function get_params_func,
    void *user_data)
```

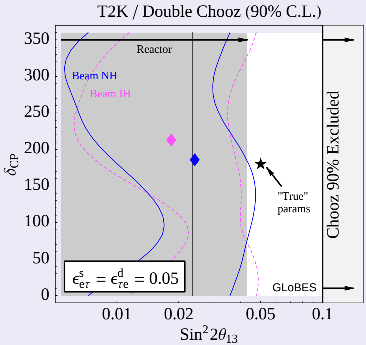
Non-standard interactions at reactors & superbeams

$P_{\bar{\nu}_e^s \to \bar{\nu}_e^d}$ (reactor) and $P_{\nu_\mu^s \to \nu_e^d}$ (superbeam) respond differently to NSI
$\Rightarrow$ Naive standard oscillation fits are affected differently

Mismatch

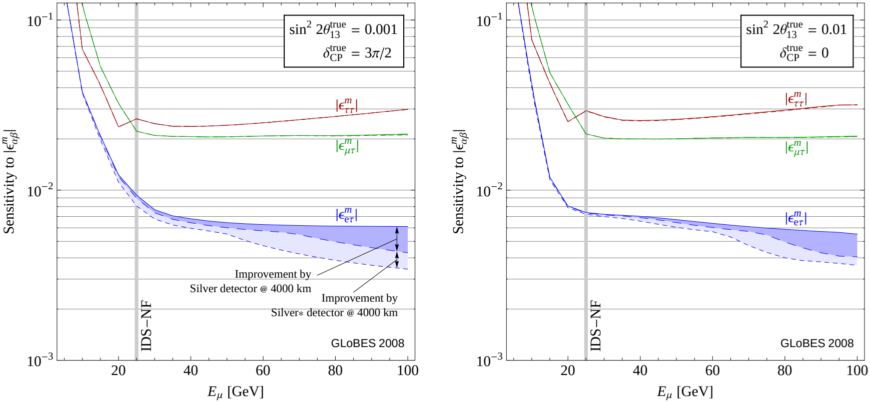


Offset



JK, M. Lindner, T. Ota, J. Sato, arXiv:0708.0152

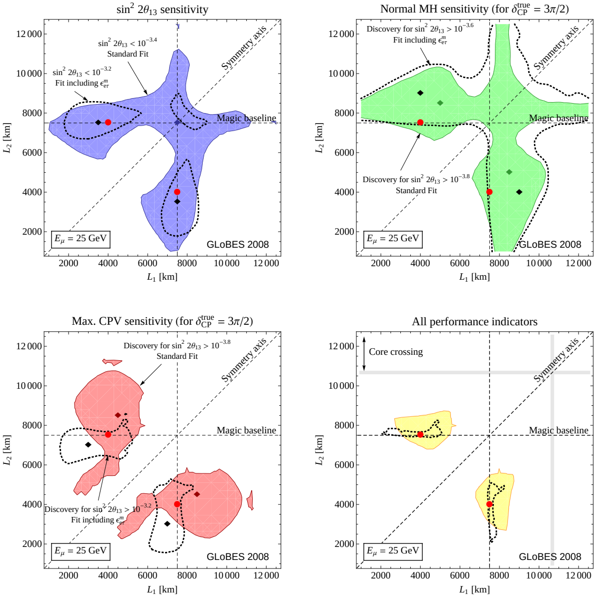
Optimization of ν-fact energy for NSI search



JK Ota Winter arXiv:0804.2261

- $E_\mu = 25$ GeV is optimal.
  ($E_\mu > 25$ GeV $\to$ no significant improvement
  $E_\mu < 25$ GeV $\to$ sensitivity decreases dramatically)
- Silver channel only useful at $E_\mu \gg 25$ GeV.

Optimization of ν-fact baselines for θ_{13}, MH, CPV



JK Ota Winter arXiv:0804.2261

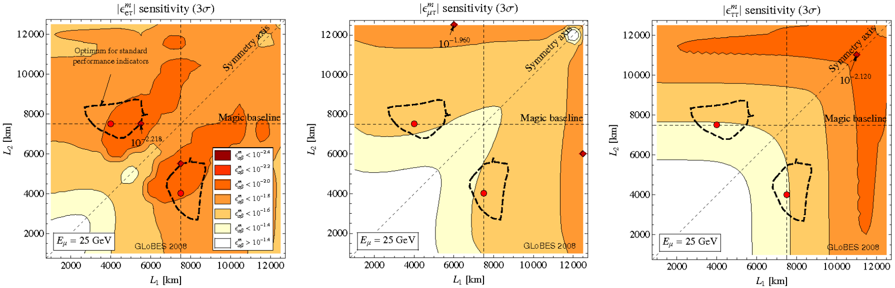
Performance indicators:

- $\sin^2 2\theta_{13}$ sensitivity: What is the new exclusion limit if $\theta_{13}^{true} = 0$?
- Normal MH/max. CPV sensitivity: How large does $\sin^2 2\theta_{13}$ have to be to guarantee detection of NH/max. CPV?

Conclusion:

$L_1 = 4000$ km, $L_2 = 7500$ km is close to optimal even if NSI are included in the fit. (We checked this also for the other $\varepsilon^m_{\alpha\beta}$)

Optimization of ν-fact baselines for NSI search



JK Ota Winter arXiv:0804.2261

- $L_1 = 4000$ km, $L_2 = 7500$ km is OK for $\varepsilon^m_{e\tau}$
- For $\varepsilon^m_{\mu\tau}$ and $\varepsilon^m_{\tau\tau}$, larger baselines are preferred.
- Note: NSI sensitivity could be improved if longer baselines were combined with higher E_μ.

Other new features

- User-defined priors Inclusion of arbitrary external input on oscillation parameters (e.g. complicated χ^2 manifolds from other simulations, or from experiments)
- New AEDL directives
  - Lists
  - Interpolating functions
- Updated AEDL files for many experiments
- Significantly improved performance

Outline

Summary

- GLoBES is a modular open source simulation tool for neutrino oscillation phenomenology.
- Its flexibility has been improved in v3.0 to face modern challenges.
  - User-defined systematics
  - User-defined oscillation engine ($\rightarrow$ simulation of arbitrary new physics)
  - ...
- GLoBES is numerically extremely efficient.
- The software has withstood the test of time, and is at the core of many strategy documents in neutrino physics.

Future plans

- Fitting real data
- Alternative minimization algorithms
- Built-in degeneracy finding algorithms
- Public mailing list archive
- Anonymous access to git repository

Thank you!

Sample AEDL file: A simple neutrino factory

```
$version="3.0.0"
nuflux(#mu_plus)<
   @builtin = 1
   @parent_energy = 50
   @stored_muons = 10.66e+20
   @time = 4
>
$target_mass = 50
$bins = 20
$emin = 4
$emax = 50
$profiletype = 1
$baseline = 3000
energy(#ERES)< /*E res.*/
   @type = 1
   @sigma_e = {0.15, 0, 0}
>
```

```
cross(#CC)< /* Cross sections */
   @cross_file = "XCC.dat"
>
channel(#nu_mu_app)<
   @channel = #mu_plus:+:e:m:#CC:#ERES
>
channel(#nu_mu_bar_disapp)<
   @channel = #mu_plus:-:m:m:#CC:#ERES
>
rule(#Nu_Mu_Appearance)<
   @signal = 0.45@#nu_mu_app
   @signalerror = 0.025 :  0.0001
   @background = 5e-6@#nu_mu_bar_disapp
   @backgrounderror = 0.2 :  0.0001
   @sys_on_function = "chiSpectrumTilt"
   @sys_off_function = "chiNoSysSpectrum"
>
```

Application code snippet: Project χ^2 onto θ_{13} axis

```
/* Define priors for θ12 and Δm²21 */
glbDefineParams(input_errors, theta12*0.1, 0, 0, 0, sdm*0.1, 0);
glbSetDensityParams(input_errors, 0.05, GLB_ALL);
glbSetCentralValues(true_values);
glbSetInputErrors(input_errors);

/* Loop over log(sin² 2θ13) */
double theta13, x;
for (x=-4; x < -2.0+0.001; x+=2.0/50)
{
   theta13 = asin(sqrt(pow(10,x)))/2;

   /* Choose starting value for δCP marginalization */
   glbSetOscParams(test_values, 200.0/2*(x+4)*M_PI/180, GLB_DELTA_CP);

   /* Compute χ² and marginalize over all parameters except θ13 */
   chi2 = glbChiTheta13(test_values, NULL, GLB_ALL);
}
```