IceCube
South Pole Neutrino Observatory

Happy Pride!

Neutrino Oscillations With nuSQUIDS

IceDUNE
June 17, 2021

Carlos Argüelles

HARVARD UNIVERSITY

Special thanks to Alex Trettin!

nuSQuIDS

Objectives of nuSQUIDS

- Provide a package that computes neutrino oscillation probabilities under various settings: Earth, Sun, …
- Provide a framework that enables the user to extend the physics easily. Important to incorporate new physics scenarios.
- Propagate neutrinos with unitary and non-unitary evolution.
- Propagate high-energy neutrinos in dense environments (neutrino interaction length < baseline).

- Interface with C++ or Python-based analyses codes.
- Provide an efficient way to reweight large size Monte Carlo simulations.

nuSQuIDS Formalism

SQuIDS
Simple Quantum Integro Differential Solver

Neutrino oscillations can be described by solving:

$$i\frac{d\vec{\nu}}{dx} = H\,\vec{\nu} \qquad\qquad H = H_{\mathrm{vac}} + H_{\mathrm{mat}}$$

where Hvac is constant and Hmat depends on x.

However this representation does not allow us to consider non-unitary evolution. Important for:

- Visible neutrino decay
- Decoherence
- Extended neutrino sources
- Neutrino opaque media
- …

nuSQUIDS solution: solve the problem using density matrices!

VE RI TAS

nuSQuIDS Formalism

SQuIDS

Simple Quantum Integro Differential Solver

Neutrino oscillations can also be described by solving:

$$\frac{\partial \rho(E,x)}{\partial x} = -i[H(E,x), \rho(E,x)]$$

where the initial conditions are given by the neutrino flux.

Now we can include non-unitary evolution by:

$$\frac{\partial \rho(E,x)}{\partial x} = -i[H_1(E,x), \rho(E,x)] - \{\Gamma(E,x), \rho(E,x)\} + F\left[\rho, \bar{\rho}; E, x\right]$$

Oscillations

Decay
Absorption

nu-sources/sinks
Cascading down
nu-nu interactions
…

nuSQuIDS Solution

SQuIDS
Simple Quantum Integro Differential Solver

We do the following in order to solve the problem:

- Formulate the problem in the interaction (Dirac) picture:

$$H(E,x) = H_0(E) + H_1(E,x)$$

- Evolve all operators with H0 (exact):

$$O_I(x) = \exp(-iH_0 x) O(x) \exp(iH_0 x)$$

- Numerically evolve the state densities with H1(t):

$$\frac{\partial \rho(E,x)}{\partial x} = -i[H_1(E,x), \rho(E,x)] - \{\Gamma(E,x), \rho(E,x)\} + F[\rho, \bar{\rho}; E, x]$$

- Get your neutrino flux/probability:

$$p_\alpha(t) = \mathrm{Tr}[\rho(t)\Pi_\alpha(t)]$$

nuSQuIDS Solution

SQuIDS
Simple Quantum Integro Differential Solver

We do the following in order to solve the problem:

- Formulate the problem in the interaction (Dirac) picture:

$$H(E,x) = H_0(E) + H_1(E,x)$$

- Evolve all operators with H0 (exact):

$$O_I(x) = \exp(-iH_0 x) O(x) \exp(iH_0 x)$$

- Numerically evolve the state densities with H1(t):

$$\frac{\partial \rho(E,x)}{\partial x} = -i[H_1(E,x), \rho(E,x)] - \{\Gamma(E,x), \rho(E,x)\} + F[\rho, \bar{\rho}; E, x]$$

- Get your neutrino flux/probability:

$$p_\alpha(t) = \mathrm{Tr}[\rho(t)\Pi_\alpha(t)]$$

Problem: lots of evaluations of commutators, anticommutators, rotations, and traces!

nuSQuIDS Solution

SQuIDS
Simple Quantum Integro Differential Solver

Problem: lots of evaluations of commutators, anticommutators, rotations, and traces!

Solution: rewrite all matrices in terms of SU(N) generators + identity

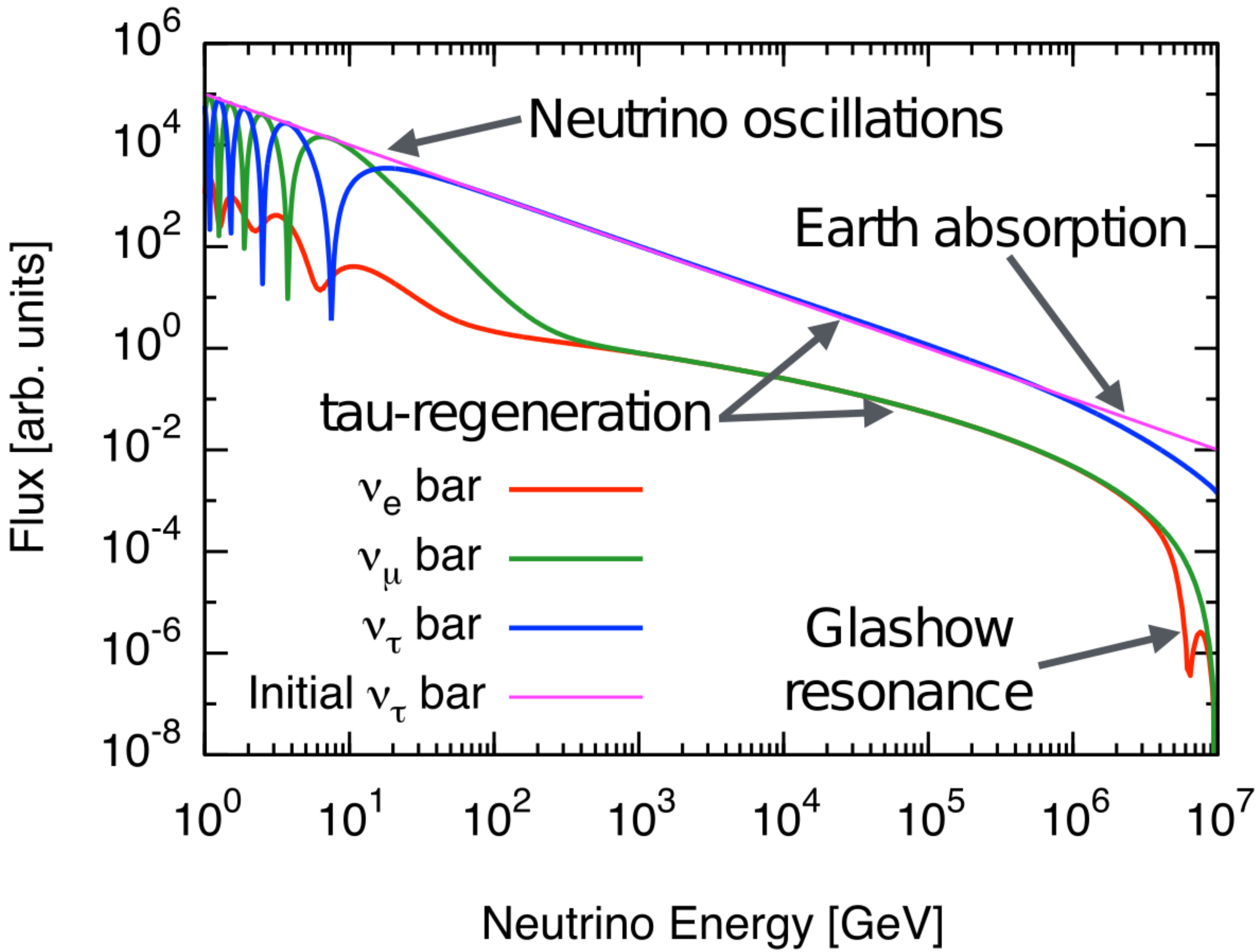
$$F_\nu(E,x) = \sum_i f_i(E,x)\lambda_i$$

$$[\lambda_i, \lambda_j] = \sum_k i f_{ijk}\lambda_k, \quad \{\lambda_i, \lambda_j\} = \sum_k d_{ijk}\lambda_k, \quad \mathrm{Tr}(\lambda_i\lambda_j) = 2\delta_{ij}$$

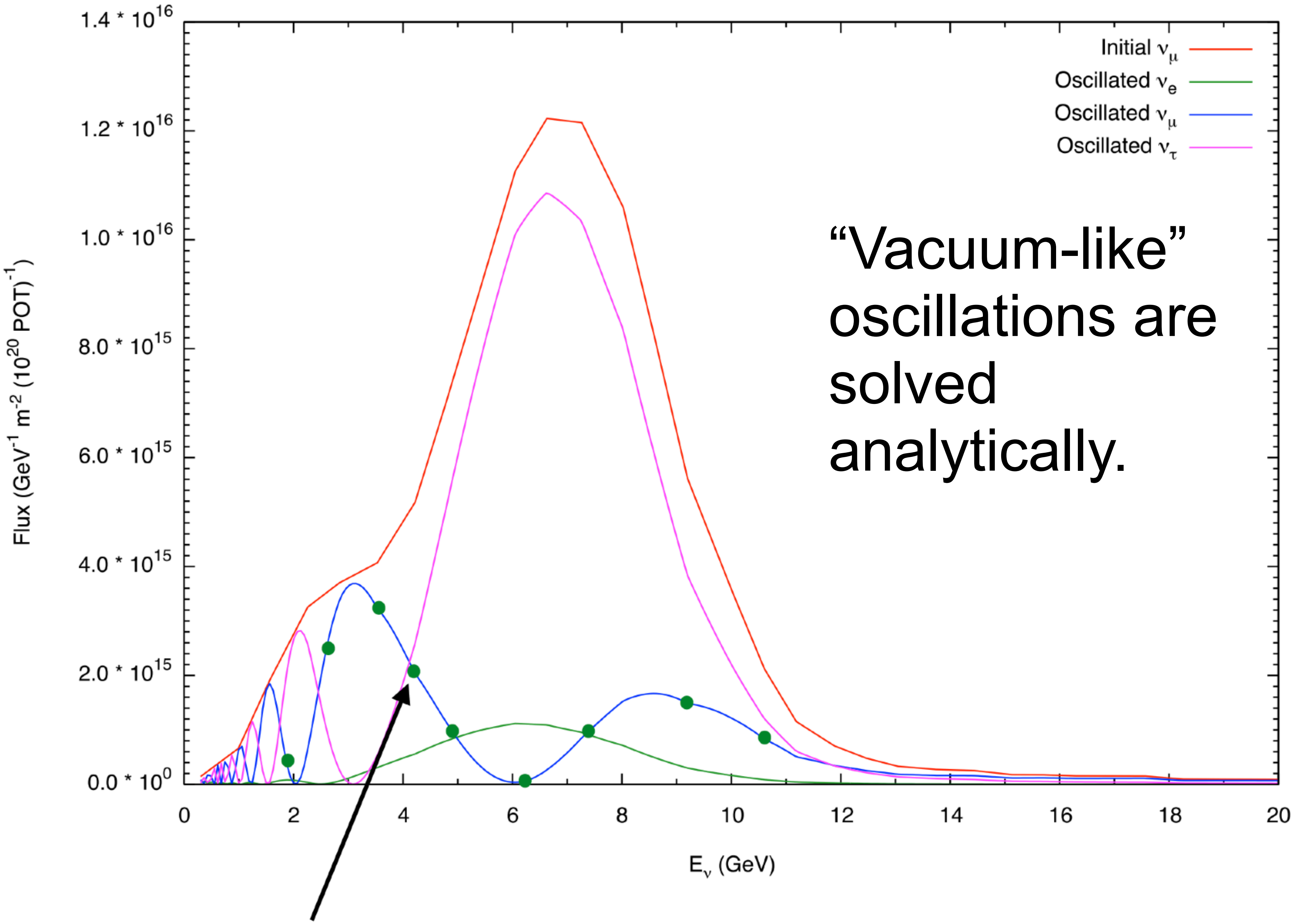
$$U^\dagger(\theta)\lambda_i U(\theta) = \sum_j r_j(\theta)\lambda_j \quad \exp(iH_0 t)\lambda_i \exp(-iH_0 t) = \sum_j h_j(t)\lambda_j$$

Every right-hand-side operation is now simple vector algebra.

nuSQUIDS evolution example



nuSQUIDS oscillations and nodes



"Vacuum-like" oscillations are solved analytically.

Nodes where the calculation is performed

nuSQuIDS in one slide

> formulate problem in interaction (Dirac) picture

$$H_s(t) = H_0 + H_1(t)$$

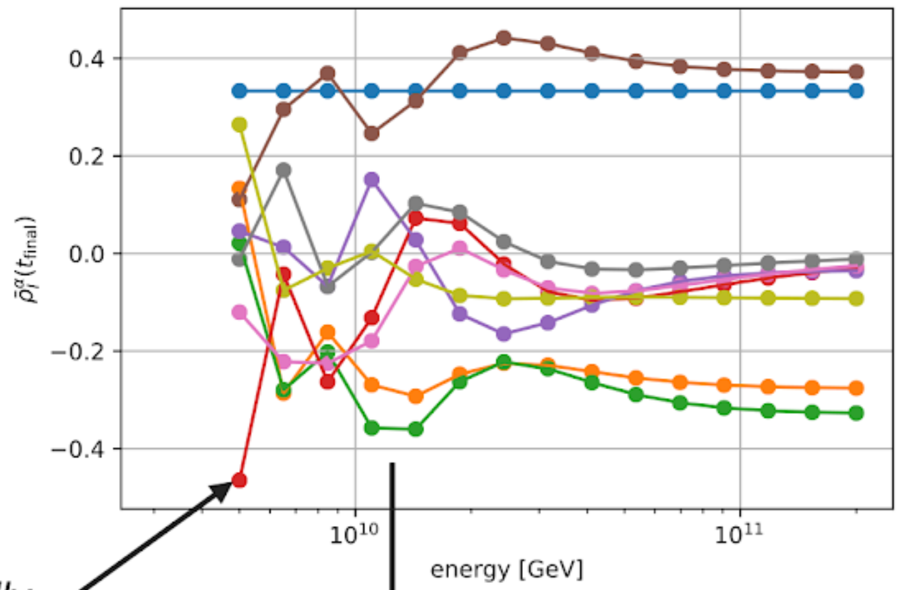
> operators evolve with H_0 (exactly solvable part)*

$$\bar{O}_I(t) = e^{iH_0 t} O_S e^{-iH_0 t}$$

> state densities evolve with $H_1(t)$

$$\partial_t \bar{\rho}_I(t) = -i[\bar{H}_{1,I}(t), \bar{\rho}_I(t)]$$

solve numerically at each node

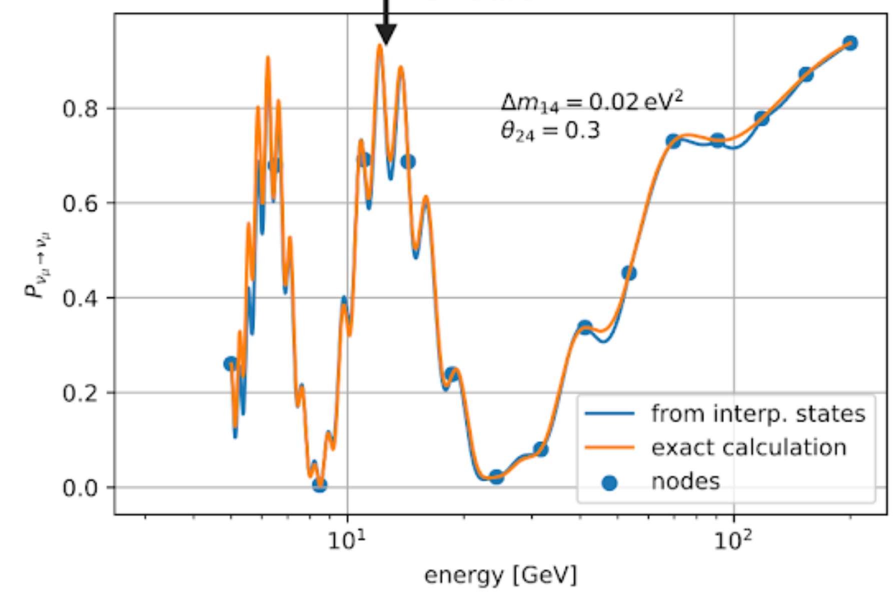


calculate $p_i(t)$ from interpolated $\bar{\rho}_I(t_{\mathrm{final}})$ between nodes

> probability to arrive in flavor state i :

$$p_i(t) = \mathrm{Tr}(\underbrace{\bar{\Pi}^{(i)}(t)}\,\bar{\rho}_I(t))$$

projection operator on flavor state i evolved with H_0



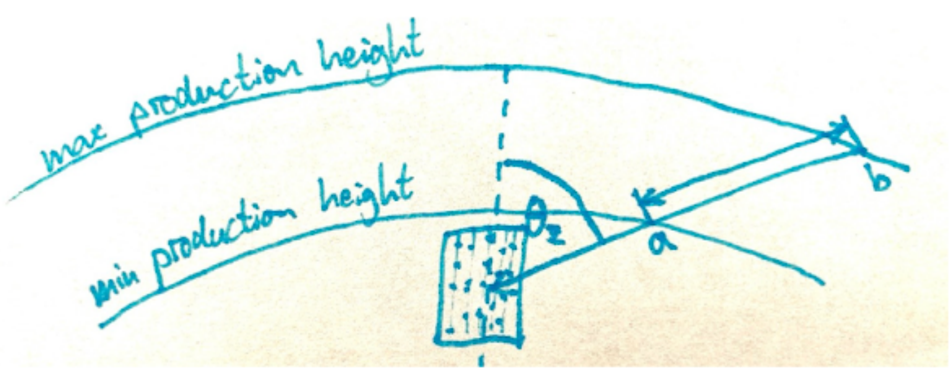
Distance averaging

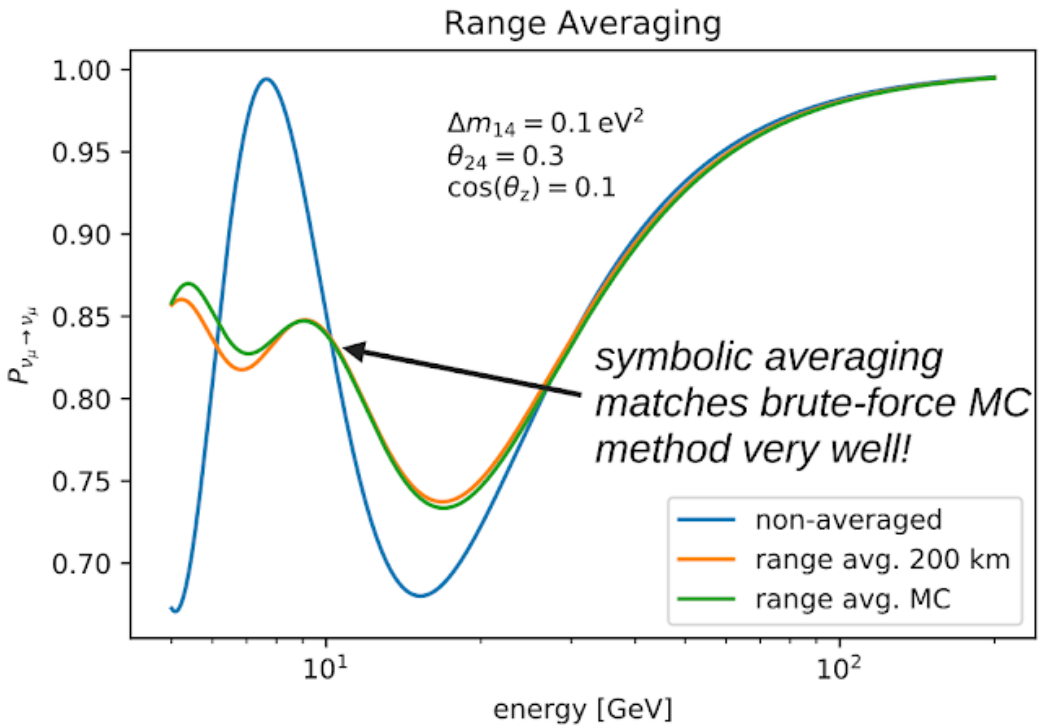
> Replace sines and cosines with their average!

> assumption: uniform distribution in interval $[a, b]$

$$\int_a^b \mathrm{d}t \frac{1}{b-a} \sin(\alpha t) = \frac{[\cos(\alpha a) - \cos(\alpha b)]}{\alpha(b-a)}$$

> calculate interval from minimum to maximum production height

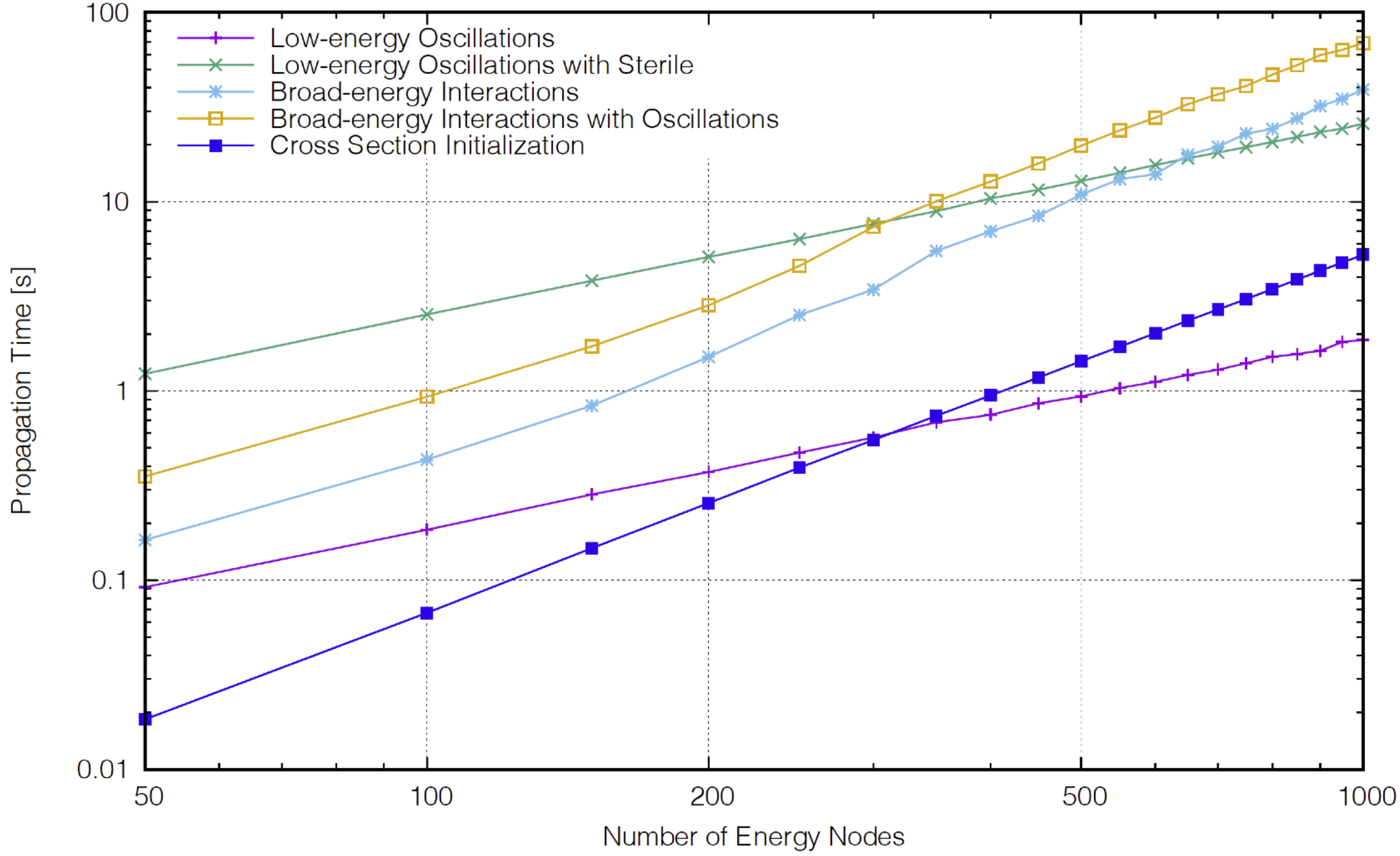




Oscillation probabilities in the presence of a sterile neutrino with averaging over a range of 200 km.

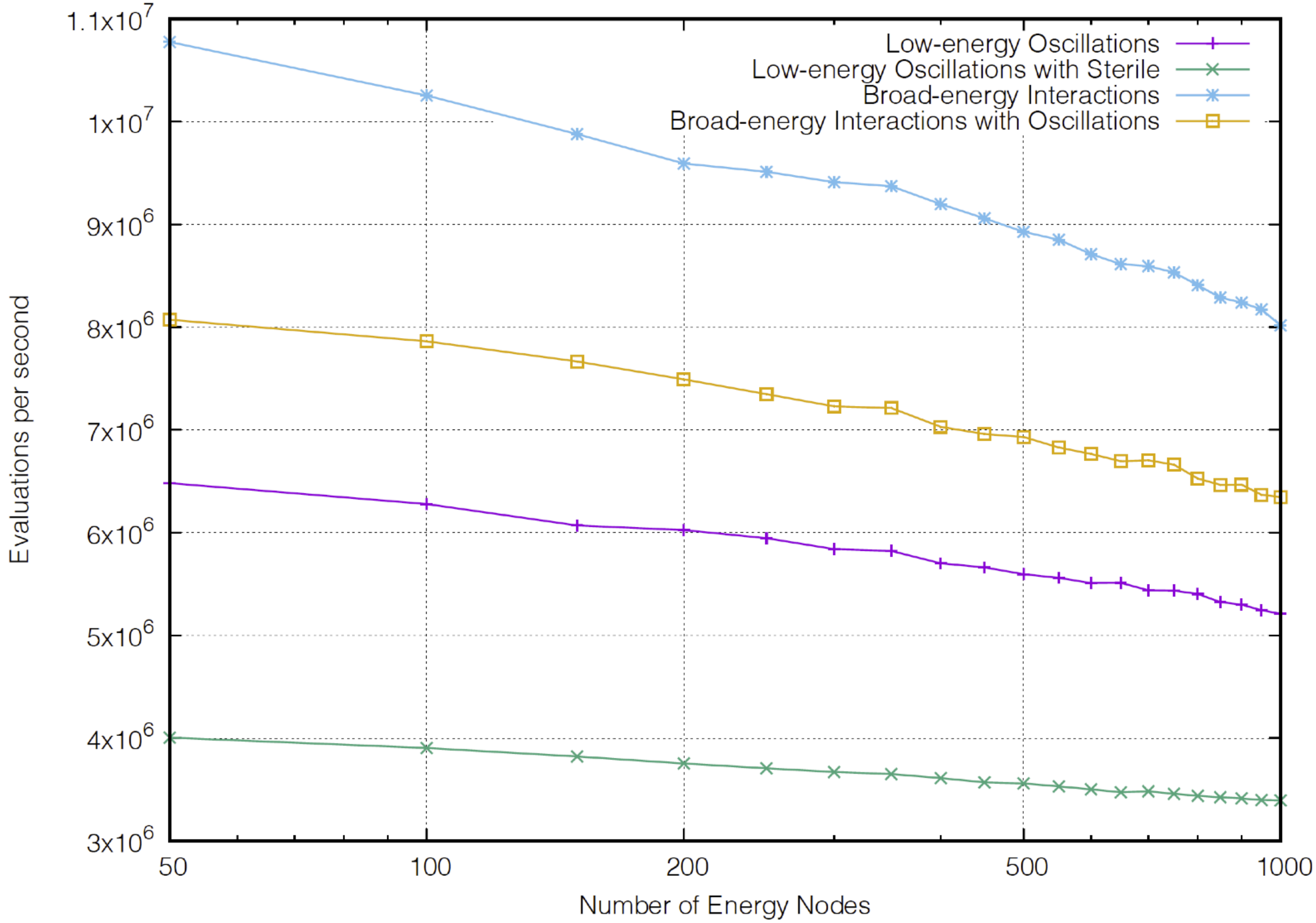
Thanks Alex Trettin for implementing this feature!

nuSQUIDS Performance



Low-energy oscillations are computed from 1 to 10 GeV on a flat spectra.
Broad-energy are computed on power-law spectra from 10 to 10^7 GeV.
All propagations are through the Earth diameter.

nuSQUIDS Performance

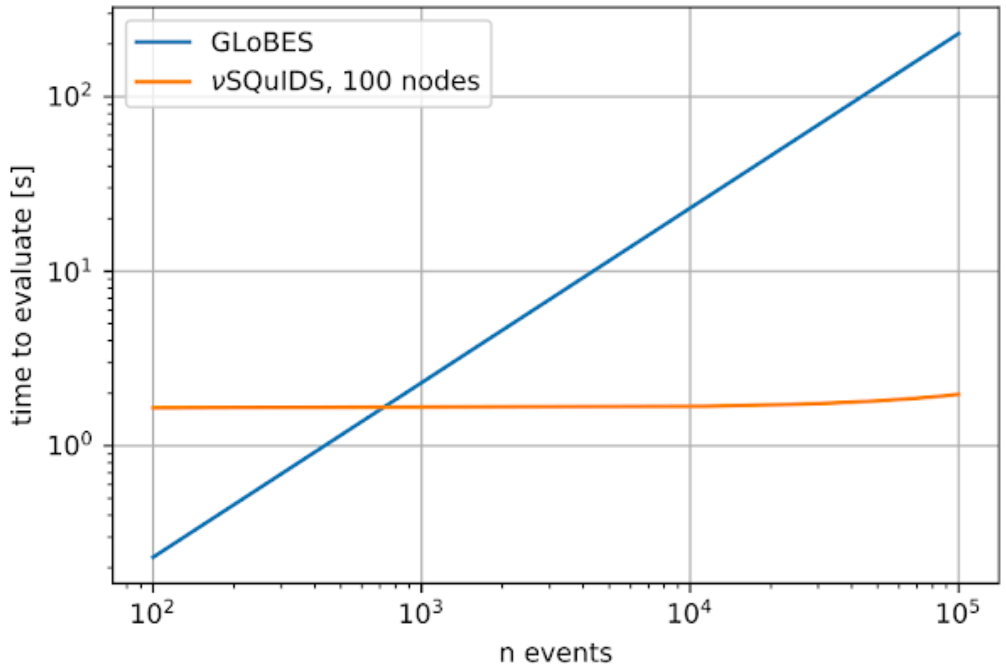


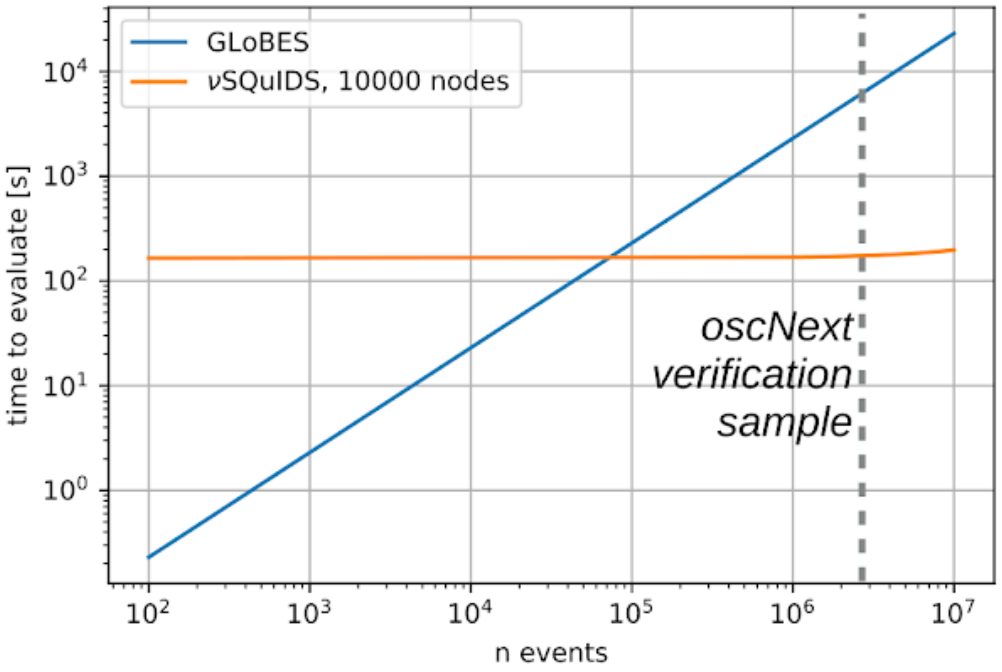
Low-energy oscillations are computed from 1 to 10 GeV on a flat spectra.
Broad-energy are computed on power-law spectra from 10 to 10^7 GeV.
All propagations are through the Earth diameter.

Timing experiment for sterile oscillations through Earth

> GLoBES: 2.3 s for 1000 evaluations

> νSQuIDS: 1.65 s for 100 nodes, 3.5 ms for 1000 interpolated evaluations





> can confirm statement from https://arxiv.org/pdf/1902.00517.pdf that νSQuIDS starts being efficient at 1000 events when interpolating in 1D, need more nodes for 2D grid

nuSQUIDS Feature Summary

- ✓Standard Oscillations
- ✓New physics extensions implemented: NSI, LV, Steriles, Decoherence, Visible and Invisible Decay
- ✓Oscillation-informed interpolation for fast Monte Carlo evaluation
- ✓Analytical averaging over production regions
- ✓Extended emission regions of neutrinos
- ✓Non-coherent neutrino interactions and collective behaviors can be coded
- ✓Atmospheric mode to simplify bookkeeping
- ✓Serializable output and input: can stop the calculation and restart it.
- ✓Python and C++ interfaces.

Get SQuIDS here: https://github.com/jsalvado/SQuIDS

Get nuSQuIDS here: https://github.com/arguelles/nuSQuIDS

Bonus Slides

OS	Compiler	PRNG	Sine	Sine & Cosine	Math Library
CentOS 7	gcc 4.8.5	6.0 ns	40.5 ns	63.5 ns	GNU C Library 2.17
CentOS 7	gcc 7.3.1	2.7 ns	37.1 ns	63.3 ns	GNU C Library 2.17
CentOS 7	clang 5.0.1	4.8 ns	96.4 ns	69.0 ns	GNU C Library 2.17
CentOS 7	icc 18.0.3	4.0 ns	20.9 ns	22.2 ns	libimf 2018.3.222
CentOS 8	gcc 8.3.1	2.7 ns	26.8 ns	31.3 ns	GNU C Library 2.28
Darwin 16.7.0	clang 5.0.0	2.9 ns	19.1 ns	20.1 ns	libsystem_m 3121.6.0
FreeBSD 11.2	clang 6.0.1	2.9 ns	22.3 ns	30.8 ns	BSD libc