Title

Simulating nonlinear neutrino flavor evolution

Journal

Computational Science & Discovery, 1(1)

ISSN

1749-4680

Authors

Duan, H

Fuller, GM

Carlson, J

Publication Date

2008-12-01

DOI

10.1088/1749-4699/1/1/015007

Peer reviewed

Simulating nonlinear neutrino flavor evolution

COMPUTATIONAL SCIENCE&DISCOVERY

Simulating nonlinear neutrino flavor evolution

H Duan[1], **G M Fuller**[2] **and J Carlson**[3]

[1] Institute for Nuclear Theory, University of Washington, Seattle, WA 98195, USA
[2] Department of Physics, University of California, San Diego, La Jolla, CA 92093, USA
[3] Theoretical Division, Los Alamos National Laboratory, Los Alamos, NM 87545, USA
E-mail: hduan@phys.washington.edu, gfuller@ucsd.edu and carlson@lanl.gov

Received 26 March 2008, in final form 1 June 2008
Published 20 November 2008
Computational Science & Discovery **1** (2008) 015007 (22pp)
doi:10.1088/1749-4699/1/1/015007

Abstract. We discuss a new kind of astrophysical transport problem: the coherent evolution of neutrino flavor in core collapse supernovae. Solution of this problem requires a numerical approach which can simulate accurately the quantum mechanical coupling of intersecting neutrino trajectories and the associated nonlinearity which characterizes neutrino flavor conversion. We describe here the two codes developed to attack this problem. We also describe the surprising phenomena revealed by these numerical calculations. Chief among these is that the nonlinearities in the problem can engineer neutrino flavor transformation which is dramatically different to that in standard Mikheyev–Smirnov–Wolfenstein treatments. This happens even though the neutrino mass-squared differences are measured to be small, and even when neutrino self-coupling is sub-dominant. Our numerical work has revealed potential signatures which, if detected in the neutrino burst from a Galactic core collapse event, could reveal heretofore unmeasurable properties of the neutrinos, such as the mass hierarchy and vacuum mixing angle θ_{13}.

Contents

1. Introduction

The ghost-like neutrinos (ν_e, $\bar{\nu}_e$, ν_μ, $\bar{\nu}_\mu$, ν_τ and $\bar{\nu}_\tau$) are chargeless, spin-$\frac{1}{2}$ particles, notorious for their weak interactions. The revelation by recent observations and experiments of nonzero neutrino rest masses and vacuum neutrino flavor mixing has forced the astrophysics community to confront a vexing problem: the nonlinear evolution of neutrino flavor in astrophysical environments where there are appreciable neutrino fluxes. Solution of this problem could be important because many environments associated with compact objects and the very early Universe are energetically and in other ways dominated by neutrinos and their interactions. Perhaps the most impressive example of this is in core collapse supernovae, where the collapse of a highly evolved core supported by relativistically degenerate electrons to a cold neutron star releases $\sim$10% of the core's *rest mass* in neutrinos of all kinds (see, e.g., [1] for an introductory review and [2] for a discussion of general supernova neutrino physics issue).

Neutrinos in this environment are thought to play a pivotal role in nearly every aspect of supernova physics, from the explosion mechanism itself, at fairly early times after the collapse of the core, to setting the conditions for the synthesis of heavy elements at later times. By virtue of their weak interactions and tenuous coupling to ordinary matter, neutrinos can transport energy, entropy, and lepton number through very dense matter that other particles might not be able to penetrate. And neutrinos can more than make up for their feeble individual interactions with huge numbers.

The astrophysics community has long recognized the important role that neutrinos play in core collapse supernova explosions. From the earliest work of Colgate, Wilson, and others [3–6] to the most recent and most sophisticated numerical simulations (e.g., [7–13]), the key problems have been to characterize the transport of neutrinos in and above the neutron star core and to calculate the energy spectra and fluxes of each neutrino species. Solving these problems has required not only front-line coupled multi-dimensional hydrodynamics and Boltzmann neutrino transport computations but also understanding physics which is not readily accessible in the laboratory, such as the equation of state of hot, neutron-rich nuclear matter.

However, the objective changes somewhat in light of the *experimental fact* of neutrino flavor transformation. It has been shown that ν_e and $\bar{\nu}_e$, neutrinos and antineutrinos with electron flavor (i.e. neutrinos associated with electrons and positrons in weak interactions), can be transformed into neutrinos and antineutrinos with other flavors, ν_μ, ν_τ, $\bar{\nu}_\mu$ and $\bar{\nu}_\tau$ (see, e.g., [14] for a review). Now both goals of calculating neutrino transport and finding the emergent neutrino energy spectra and fluxes may not be attainable without an adequate treatment of medium-affected neutrino flavor transformation. As we shall see below, in many circumstances adding on a flavor evolution solver would greatly complicate neutrino transport calculations. We can identify two broad regimes in this problem: (i) the high density partially coherent or non-coherent regime, where the neutrino transport mean free paths are comparable to or smaller than neutrino flavor oscillation lengths or Mikkheyev–Smirnov–Wolfenstein (MSW) [15–17] resonance widths; (ii) the coherent regime, where neutrinos are essentially free streaming on scales relevant for neutrino flavor transformation. The first regime can be very difficult to treat and requires a quantum kinetic approach (e.g. [18–24]). However, it is generally thought that active–active neutrino flavor transformation is negligible in this regime. This prejudice is motivated by the fact that the relevant active–active channel neutrino mass-squared differences are small, the matter density is high in this regime [18, 25], and collective neutrino flavor transformation is suppressed in very dense neutrino gases [26].

Our numerical work focuses only on the second (coherent) regime. The coherent neutrino flavor evolution problem differs in two essential aspects from conventional transport problems. First, of course, is quantum mechanical coherence itself. Second is neutrino self-coupling.

Coherence dictates that we follow the development in time and/or space of the complex amplitudes which describe the flavor states of neutrinos. In the coherent limit, the full quantum kinetic equations reduce to a Schrödinger-like equation. On a given neutrino trajectory, or world line, with affine parameter t, this Schrödinger-like equation is

$$\mathrm{i}\frac{\mathrm{d}}{\mathrm{d}t}|\psi(t)\rangle = \hat{H}|\psi(t)\rangle. \tag{1}$$

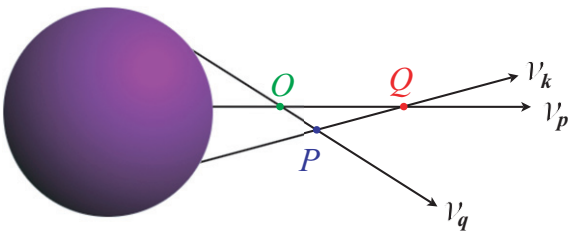

Figure 1. Illustration of the quantum-coupling problem for neutrinos on different, but intersecting trajectories. The world lines for three neutrino beams, ν_k, ν_p, and ν_q, intersect as shown at points O, P, and Q. Neutrino–neutrino forward-scattering at these intersection points will quantum mechanically couple the flavor evolution histories of these neutrinos.

Here, the operator $\hat{H}$ is a Hermitian Hamiltonian operator that contains the vacuum and in-medium generators of neutrino flavor evolution, whereas $|\psi(t)\rangle$ is the ket that describes the flavor state of a neutrino at point t on this given trajectory. Neutrinos in supernovae for the most part will be ultra-relativistic, so it may seem odd that a 'non-relativistic Schrödinger equation' can give an adequate account of flavor evolution. It works for two reasons [27]: (i) we consider only neutrino forward-scattering interactions and neglect inelastic scattering; and (ii) in the ultra-relativistic limit, the information in the four-component Dirac spinor describing the neutrino's flavor state is mostly contained in two components, the two 'large' components.

A serious complication in following neutrino flavor transformation in supernovae stems from neutrino–neutrino forward-scattering contributions to $\hat{H}$. These render the problem nonlinear, since the interactions which dictate flavor transformation amplitudes are themselves dependent on the neutrino flavor states. This seemingly innocuous statement masks a kind of geometric coupling which, to our knowledge, is unique in astrophysical transport problems. Neutrino–neutrino scattering could quantum mechanically couple the flavor evolution on all intersecting trajectories. This is illustrated in figure 1. Following the flavor evolution of neutrino ν_p would entail knowing the flavor states of all neutrinos on which it forward-scattered. That might include, for example, neutrinos ν_k and ν_q propagating on world lines which intersect ν_p's trajectory at points Q and O, respectively. Note, however, that ν_k's and ν_q's flavor histories are not independent, since they may have undergone a forward-scattering event at point P. Obviously, with a large number of neutrinos this quickly can become a daunting problem!

To simplify the problem, we adopt the spherically symmetric 'neutrino bulb' model (hereafter the 'bulb model') [28]. In this model, neutrinos are emitted half-isotropically from a spherical 'neutrino sphere' (essentially the hot neutron star surface) with energy spectra that are initially of black body form. We adopt a smooth matter density profile ρ which depends only on radius r, the distance from the center of the neutron star. At late times, close to the neutron star where matter tends to be isothermal, this matter profile falls exponentially. Further out, where the envelope may be closer to constant entropy in character, the density profile is $\rho \sim r^{-3}$. However, in our numerical calculations for earlier supernova epochs, we have also used more complicated density profiles taken from published numerical supernova models [29].

Because of the assumed spherical symmetry, neutrino trajectories with the same emission angle ϑ_0 become equivalent, and neutrino modes emitted in the same flavor and energy state and with the same emission angle have identical flavor evolution history [28]. Here, the emission angle ϑ_0 is the angle between the neutrino propagation direction and the vector normal to the neutrino sphere at the emission point.

In the bulb model, all neutrino evolution above the neutrino sphere is assumed to be coherent and essentially free-streaming in character. This will be a good approximation for the conditions expected to obtain a few seconds after core bounce, and may be useful for earlier times in some cases. We bin neutrino modes emitted in each flavor by both the emission angle and the energy of the neutrino mode. Because the flavor histories of all neutrino bins are coupled, we solve self-consistently for the flavor evolution of all neutrino modes simultaneously. The flavor evolution of each neutrino mode is described by an equation in the form of (1). Each neutrino trajectory has an individual affine parameter which can be expressed in terms of radius r.

There are a variety of physical processes which can affect neutrino flavor evolution, either by complicating the coherent calculation procedure outlined above, or by introducing decoherent effects. As an example of the

former, the density profile in the core collapse supernova environment can be quite dependent on time, with shocks and multi-dimensional effects of paramount importance in some epochs (e.g. [30, 31]). In the latter case, matter density fluctuations associated with shocks, sound waves and turbulence can lead to neutrino flavor depolarization and decoherence [32–35].

In our numerical work, we have chosen to concentrate on the simplest environments in order to gain insight into the effects of coupled quantum-mechanical flavor evolution. Our rationale is that this physics will always be present in the environments we treat. The nonlinear collective phenomena revealed by our calculations can be modified in real supernovae by the kinds of effects discussed above. However, our calculations lead us to conclude that most of the *qualitative* and even some of the quantitative results for our simple models are likely to survive. Even our simplified supernova models present novel numerical challenges which will have to be faced in any comprehensive treatment of the subject.

2. Solution method

Even with the simplified model discussed in section 1, obtaining a convincing solution to the full, coupled-trajectory supernova neutrino flavor evolution problem, which we term the 'multi-angle' problem, is still a challenging task. So difficult, in fact, that there had been no attempt to solve it prior to our effort. This meant that previous work provided no guidance on numerical strategies for solving multi-angle neutrino flavor evolution. The only clues to general coherent neutrino flavor phenomena in supernovae came from the so-called 'single-angle' calculations, where trajectories were coupled, but the flavor evolution on every trajectory was taken to be the same as that on a radially directed trajectory. In section 2.1, we outline the plan we used to attack this problem. Partly because of the lack of previous work, we have developed two numerical codes more or less independently at UCSD and LANL. As an aid to debugging and as a device for teasing apart physics issues from numerical issues, the results from the two codes were compared for key test problems. This procedure proved efficacious, in part because sometimes numerical issues and problems tended to be different in each code. In sections 2.2–2.4, we highlight the key implementations of *FLAT*, the computer code developed at UCSD. In section 2.5, we discuss some features of *BULB*, the code developed at LANL.

2.1. Project plan

Our ultimate goal is to solve for the flavor histories of neutrinos and antineutrinos (i.e. ν_e, ν_μ, ν_τ, $\bar{\nu}_e$, $\bar{\nu}_\mu$, and $\bar{\nu}_\tau$) emitted in the bulb model with realistic energy spectra and traveling on all physically realizable trajectories. This is what we term the 'three-flavor multi-angle' problem.

In the literature the neutrino oscillation problem with three flavors is usually approximated as two two-flavor oscillation problems that occur at different time/distance scales. The rationale for this is that the neutrino mass-squared differences associated with these two-flavor oscillation problems are different by more than an order of magnitude. Previous work on medium-affected neutrino oscillations exclusively adopted this two-flavor approximation [18, 36–42].

These studies also adopted what we termed above the 'single-angle' approximation. In this approximation, flavor evolution of neutrinos with energy E propagating along different neutrino trajectories is assumed to be the same as that of neutrinos with the same energy E propagating along a representative trajectory, usually taken to be the radial direction. The validity of both the two-flavor and single-angle approximations for supernova neutrino flavor evolution, however, remains to be shown.

Because of the big gap between the ultimate goal we would like to achieve and the state of art of neutrino oscillation studies, it is clearly very risky to attempt a solution to the full problem in one single step. Not only will there be little theoretical guidance in such a monolithic approach, but also finding physical interpretations of any novel numerical results could prove to be problematic. Instead, we have sought to achieve our goal in four stages.

In the first stage, we solve for neutrino flavor transformation using the two-flavor, single-angle approximation. This stage overlaps with studies which already exist in the literature. This procedure should offer a good check on our numerical codes as well as on existing theories.

In the second stage, we will still assume two neutrino flavors, but will solve for neutrino and antineutrino flavor evolution using a full multi-angle implementation of the bulb model. The difference between this stage and the first stage should give us insights on how multi-angle effects may enter into neutrino flavor transformation and how good the single-angle approximation might be.

In the third stage, we will implement full three-neutrino flavor evolution, but with the single-angle approximation. Comparison of the results of this stage and those of the first stage should give us important clues on whether the two-flavor approximation is valid in collective neutrino oscillations.

In the last stage, we will solve the full problem, i.e. neutrino oscillations with three neutrino flavors and a full multi-angle implementation. Presumably, the results of the computations in this stage could be predicted at least crudely from the theories which employ the two-flavor, single-angle model and the results of the second and third stages calculations. Comparison between the numerical results in the full three-neutrino calculations and the theoretical estimates again can be used to check both the numerical codes and the theories.

2.2. FLAT's highly modular design

In this section, we will outline the overall structure of one of our numerical codes designed to solve the neutrino flavor transformation problem defined by equation (1). In the following two sections, we will flesh out this outline with a detailed discussion of the physics in equation (1) and the algorithms employed here to simulate this physics.

As explained above, we plan to achieve our ultimate goal progressively, essentially by solving three intermediate problems before attacking the full problem. These problems differ only in the physical approximations that they employ. Written in C++, *FLAT* takes advantage of the object-oriented programming paradigm and provides a unique and uniform solution to all these problems by offering a hierarchical set of problem-specific modules. Changing a particular physical setting or a numerical algorithm can be done by swapping out appropriate module(s). The structure of *FLAT* is illustrated in figure 2.

At the lowest level of the hierarchy are the `NeuBin` modules. Each of the `NeuBin` modules implements two C++ classes, `NeuPot` and `NeuBin`, which correspond to the Hamiltonian $\hat{H}$ and the flavor state $|\psi\rangle$ in equation (1), respectively. We note that the Schrödinger equation (1), as well as the numerical algorithm that *FLAT* uses to solve this equation, does not depend on the specifics of the representations of $\hat{H}$ or $|\psi\rangle$. Therefore, the higher-level modules in *FLAT* are independent of the internal implementation in the `NeuBin` modules and can operate through a uniform interface on any `NeuBin` module. The `NeuBin` modules are represented as pentagonal blocks in figure 2. The `NeuBin` module interfaces are represented by the common-color layers on top of the pentagonal blocks.

Two examples of the interface functions of a `NeuBin` class are `NeuBin::evolve()` and `NeuBin::diff()`. Function `NeuBin::evolve()` evolves the corresponding flavor state $|\psi\rangle$ residing at affine parameter value t on a given trajectory one step further for given Hamiltonian $\hat{H}$ and step size Δt (see section 2.3). Function `NeuBin::diff()` computes the 'difference' between two flavor states. This difference is used to estimate numerical errors.

There are two `NeuBin` modules that have been used in production runs: `NeuBin_F2C` and `NeuBin_F3C`. These modules define the flavor state and Hamiltonian for two-flavor and three-flavor problems, respectively. Once the program has been proven to work well using the two-flavor approximation, we can use the same program to solve three-flavor problems by swapping out the `NeuBin_F2C` module with `NeuBin_F3C`.

Above the `NeuBin` modules are the `NeuBeam` modules, each of which defines a `NeuBeam` class. A `NeuBeam` object contains an array of `NeuBin` objects. Each element of this array corresponds to a flavor state $|\psi(E)\rangle$ for a particular energy bin $[E - \frac{1}{2}\Delta E, E + \frac{1}{2}\Delta E]$. All `NeuBeam` modules share the same interface through which higher-level modules can operate `NeuBeam` objects.

`NeuBeam::sum()` is one of the important interface functions for the `NeuBeam` modules. This function does a numerical integration over all energy bins. This summation represents an important step in producing the neutrino self-coupling Hamiltonian $\hat{H}_{\nu\nu}$ (see section 2.5). The algorithm used in this numerical integration is specific to each module and, again, is hidden from higher-level modules.

The `NeuBeam` module used in production runs is called `NeuBeam_SS`. In this module, `NeuBeam::sum()` employs a simple midpoint rule for numerical integration. If a more sophisticated algorithm is required, a

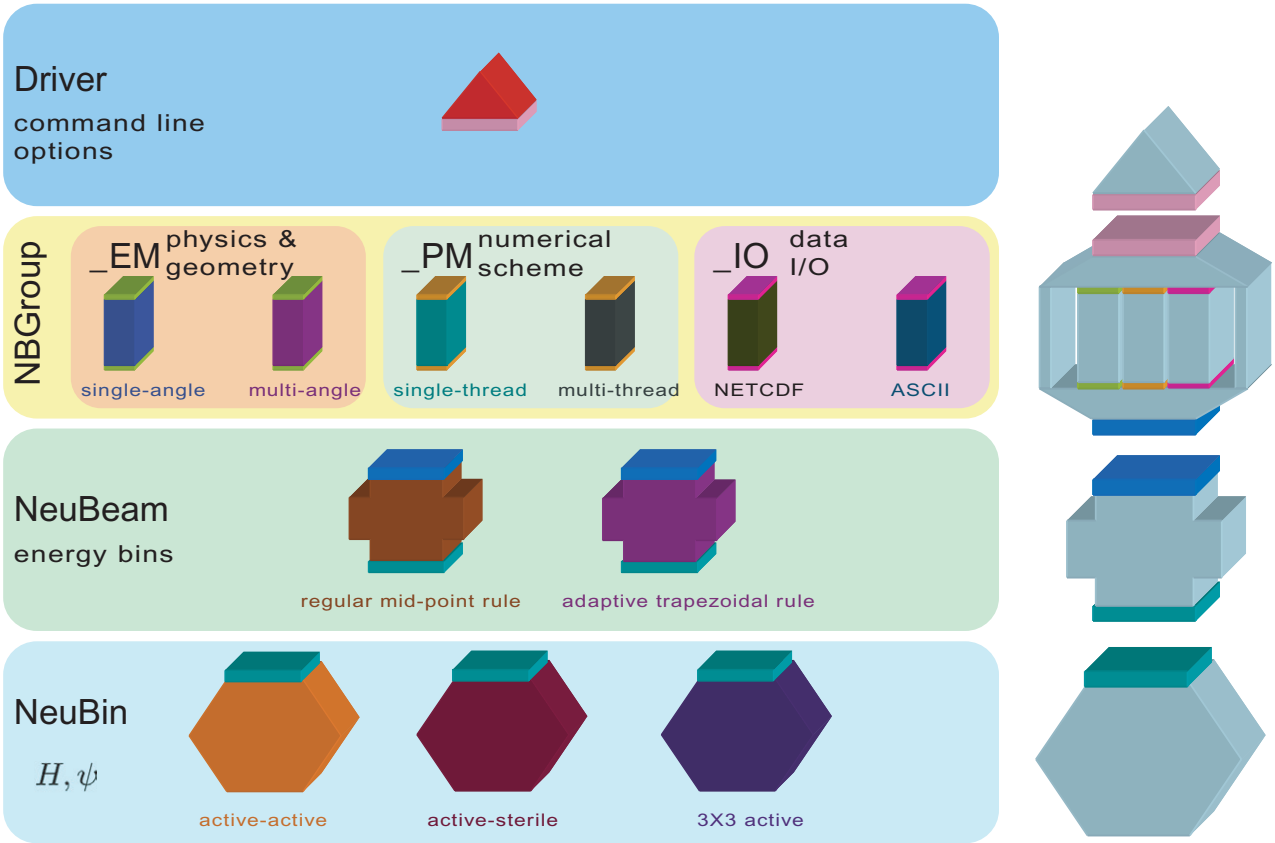

Figure 2. Illustration showing the structure of *FLAT*. This code is a collection of modules. These modules are represented here as various building blocks in the left part of the figure. An executable binary can be compiled by choosing appropriate modules for a particular problem. This is illustrated in the right part of the figure. Building blocks with the same geometric shape and the same top/bottom layer(s), but with different colors, represent modules that have the same interface functions but perhaps employ different physical approximations or numerical algorithms. Changing a particular physical setting or a numerical algorithm can be achieved by swapping out the appropriate module(s) in the code with other module(s) with the same interface functions.

new `NeuBeam` module can be created. The `NeuBeam_SS` module can be swapped out with this new module without modification to the remaining components of *FLAT*. The corresponding upgrade of the code is done automatically.

On top of the `NeuBeam` modules are the `NBGroup` modules. There are three kinds of `NBGroup` modules: `NBGroup_EM`, `NBGroup_IO`, and `NBGroup_PM`. These modules specify the physical environment, the input/output (I/O) methods, and the numerical algorithm to solve differential equations, respectively. These three types of modules share the same core data, i.e. the current flavor states of all neutrinos, and, at the same time, perform nearly independent tasks. This special relation among `NBGroup` modules is realized through the inheritance of classes and virtual functions. An `NBGroup` class is defined first. This class contains all the data that are used by at least two kinds of `NBGroup` modules. This class also declares a variety of virtual functions such as `NBGroup::set_bg()`, `NBGroup::check()`, and `NBGroup::evolve()`. Each of these virtual functions is defined in a child class of `NBGroup`, but may be called by other child classes.

Each of the `NBGroup_EM` modules defines an `NBGroup_EM` class as a child class of `NBGroup`. A key interface function of these modules is `NBGroup_EM::set_bg()` which, by definition of a virtual function, is the 'real function' to be executed when virtual function `NBGroup::set_bg()` is called. `NBGroup_EM::set_bg()` calculates $\hat{H}_{\rm matt}$, the ordinary matter background contribution to the total Hamiltonian $\hat{H}$. `NBGroup_EM::set_bg()` also calculates $\hat{H}_{\nu\nu}$, the background neutrino medium contribution to $\hat{H}$.

There are two `NBGroup_EM` modules used in production runs: `NBGroup_PNS1D` and `NBGroup_PNS2D`. In `NBGroup_PNS1D` the single-angle approximation is implemented, while in `NBGroup_PNS2D` the multi-angle implementation is employed.

Each of the `NBGroup_IO` modules defines an `NBGroup_IO` class as a child class of `NBGroup`. An `NBGroup_IO` class defines `NBGroup_IO::check()`, which is the real function corresponding to `NBGroup::check()`. `NBGroup_IO::check()` performs a variety of critical checks. For example, if necessary, it can save a snapshot of the current flavor states of all the neutrino modes, e.g. at specified time intervals. Also, `NBGroup_IO::check()` can set a stop flag if the specified time allocation is used up.

The `NBGroup_IO` modules used in production runs are `NBGroup_NC` and `NBGroup_NC_SMP`. Module `NBGroup_NC` reads and writes data in the NetCDF format [43], a cross-platform binary data format. Module `NBGroup_NC_SMP` is similar to `NBGroup_NC`, except that it handles I/O in a different POSIX thread (Pthread) [44]. As a result, `NBGroup_NC_SMP` is able to complete `NBGroup_IO::check()` function calls promptly. This helps to augment the overall code performance, because the CPU can continue to do computation during the time in which it would otherwise be idle waiting for slow I/O data transfers to complete.

Each of the `NBGroup_PM` modules defines an `NBGroup_PM` class as a child class of `NBGroup`. An `NBGroup_PM` class defines a member function `NBGroup_PM::evolve()`, the real function corresponding to `NBGroup::evolve()`. Once called, `NBGroup_PM::evolve()` can continuously evolve neutrino flavor states over radial distance until stop flags are encountered.

The `NBGroup_PM` modules used in production runs are `NBGroup_MM` and `NBGroup_MM_SMP`. In the `NBGroup_MM` module, `NBGroup_PM::evolve()` employs an adaptive step-size algorithm to solve equation (1) for each neutrino trajectory (see section 2.4). During each sub-step of the algorithm's execution, `NBGroup_PM::evolve()` calls `NBGroup::set_bg()` to set $\hat{H}_{\rm matt} + \hat{H}_{\nu\nu}$. At the end of each whole step in the algorithm's execution, `NBGroup_PM::evolve()` calls `NBGroup::check()` to check and see if any I/O needs to be done and if the program needs to be terminated. `NBGroup_MM_SMP` implements a multi-threaded version of the algorithm used in `NBGroup_MM`. *FLAT* automatically uses the multi-CPU/core feature of a modern workstation when the `NBGroup_MM` module is replaced by `NBGroup_MM_SMP`.

At the highest level, a `Driver` module defines a class designated as `Neutrinos`. This class inherits the `NBGroup_EM`, `NBGroup_PM` and `NBGroup_IO` classes. In the `Driver` module, `Neutrinos::evolve()` (which is actually `NBGroup_PM::evolve()`) is called to solve the neutrino flavor transformation problem. `Driver` also handles command line options. These might include, for example, setting the time allocation and/or the final radius.

FLAT's highly modular framework endows it with great extensibility. This aids program development as well as helps with the exploration of neutrino flavor conversion physics. With this modular architecture it is straightforward to pick the simplest numerical algorithm to attack a given problem. Additional or different modules can be added easily if more sophisticated algorithms are necessary, or if the intention is to treat different problems. When following the project plan outlined in section 2.1, the modular framework of *FLAT* enables us to minimize redundancy in code writing and maximize usage of existing code. It also maintains consistency when we progress from one stage to the next and, therefore, isolates the differences in various physics approximations when we compare the results obtained in different stages.

2.3. `NeuBin` modules in FLAT

Currently *FLAT* can utilize two different `NeuBin` modules: `NeuBin_F2C` and `NeuBin_F3C`. These implement the Hamiltonian $\hat{H}$ and the neutrino flavor state $|\psi\rangle$ for the two- and three-flavor mixing schemes, respectively.

The `NeuBin_F2C` module defines two classes, `NeuBin` and `NeuPot`. A `NeuBin` object contains two complex variables. These variables correspond to the two complex components of the flavor wavefunction

$$\psi \equiv \begin{pmatrix} a_{\nu_e} \\ a_{\nu_x} \end{pmatrix}, \tag{2}$$

where $a_{\nu_\alpha} \equiv \langle \nu_\alpha | \psi \rangle$ is the amplitude for a neutrino in state $|\psi\rangle$ to be in flavor state $|\nu_\alpha\rangle$, and where the flavor index $\alpha = e, x$, with x representing either the μ or τ neutrino or a linear combination of these. Class `NeuPot`,

however, does not render the corresponding flavor-basis representation of the Hamiltonian $\hat{H}$ as the familiar Hermitian matrix in the Schrödinger equation. In the two-flavor mixing case, for example, the Hamiltonian would be a 2×2 complex array

$$H = \begin{pmatrix} \langle \nu_e|\hat{H}|\nu_e\rangle & \langle \nu_e|\hat{H}|\nu_x\rangle \\ \langle \nu_x|\hat{H}|\nu_e\rangle & \langle \nu_x|\hat{H}|\nu_x\rangle \end{pmatrix}. \tag{3}$$

Instead, `NeuPot` stores the Hamiltonian in a three-component real vector $\boldsymbol{h}$ which is defined by

$$H = h_0 + \boldsymbol{h}\cdot\boldsymbol{\sigma} = h_0 + \sum_i h_i\sigma_i. \tag{4}$$

Here the three Cartesian components of the vector $\boldsymbol{\sigma}$ corresponds to the Pauli matrices:

$$\sigma_1 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad \sigma_2 = \begin{pmatrix} 0 & -\mathrm{i} \\ \mathrm{i} & 0 \end{pmatrix}, \quad \text{and} \quad \sigma_3 = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}. \tag{5}$$

The trace of matrix H (h_0 in equation (4)) is of no physical significance and is ignorable. This is because h_0 generates only an overall common phase in the neutrino flavor amplitudes and, therefore, is not relevant for neutrino oscillations or neutrino flavor transformation.

Equation (1) has an approximate solution for a small parameter step Δt,

$$\psi(t+\Delta t) \simeq \exp(-\mathrm{i}H\Delta t)\psi(t) \tag{6}$$

$$= \begin{pmatrix} \cos(h\Delta t) - \mathrm{i}\tilde{h}_3\sin(h\Delta t) & -(\mathrm{i}\tilde{h}_1 + \tilde{h}_2)\sin(h\Delta t) \\ -(\mathrm{i}\tilde{h}_1 - \tilde{h}_2)\sin(h\Delta t) & \cos(h\Delta t) + \mathrm{i}\tilde{h}_3\sin(h\Delta t) \end{pmatrix}\psi(t), \tag{7}$$

where

$$h \equiv |\boldsymbol{h}| = \sqrt{h_1^2 + h_2^2 + h_3^2} \tag{8}$$

and

$$\tilde{h}_i \equiv \frac{h_i}{h}. \tag{9}$$

With a specified Hamiltonian $\hat{H}$ (a `NeuPot` object), equation (7) is employed in module `NeuBin_F2C` to evolve a `NeuBin` object by a small step Δt. The procedure implied by equation (6) preserves the unitarity of ψ and becomes exact when H is time independent.

For a particular momentum mode λ, the neutrino (antineutrino) density operator $\hat{\rho}_\lambda$ ($\hat{\bar{\rho}}_\lambda$) provides the necessary information to construct a neutrino–neutrino forward-scattering background potential $\hat{H}_{\nu\nu}$ for a given test neutrino. (Note that a momentum mode is equivalent to an energy/angle bin in our numerical calculation.) The density operator for neutrino mode λ is, for example, $\hat{\rho}_\lambda = |\psi_\lambda\rangle\langle\psi_\lambda|$. Potential $\hat{H}_{\nu\nu}$ is essentially a summation of densities operators over all neutrino and antineutrino modes, weighted by the directional dependence of the weak current–current Hamiltonian (see, e.g. [28]). The contents of the density matrices are stored in `NeuPot` objects. In the `NeuBin_F2C` module, this is done by, for example, writing density matrix ρ as

$$\rho = \tfrac{1}{2} + \boldsymbol{s}\cdot\boldsymbol{\sigma}, \tag{10}$$

where

$$\boldsymbol{s} = \begin{pmatrix} \mathrm{Re}(a_{\nu_e}^* a_{\nu_x}) \\ \mathrm{Im}(a_{\nu_e}^* a_{\nu_x}) \\ \frac{1}{2}(|a_{\nu_e}|^2 - |a_{\nu_x}|^2) \end{pmatrix}. \tag{11}$$

The approach adopted in *FLAT* for handling the full 3×3 neutrino flavor evolution problem is essentially similar to the method outlined above for the 2×2 case. It is, of course, necessarily more complicated.

Like the `NeuBin_F2C` module, the `NeuBin_F3C` module defines a `NeuBin` class implementing neutrino flavor wavefunction

$$\psi \equiv \begin{pmatrix} a_{\nu_e} \\ a_{\nu_\mu} \\ a_{\nu_\tau} \end{pmatrix}, \tag{12}$$

in obvious analogy to equation (2). The `NeuPot` class in the `NeuBin_F3C` module defines nine real variables: u_{11}, u_{22}, u_{33}, u_{12}, u_{23} u_{31}, w_{12}, w_{23}, and w_{31}. Variables u_{ij} and w_{ij} are defined in

$$H_{ij} = u_{ij} + \mathrm{i}w_{ij}, \tag{13}$$

where H_{ij} are the elements of the 3×3 flavor-basis Hamiltonian matrix

$$H = \begin{pmatrix} \langle \nu_e|\hat{H}|\nu_e\rangle & \langle \nu_e|\hat{H}|\nu_\mu\rangle & \langle \nu_e|\hat{H}|\nu_\tau\rangle \\ \langle \nu_\mu|\hat{H}|\nu_e\rangle & \langle \nu_\mu|\hat{H}|\nu_\mu\rangle & \langle \nu_\mu|\hat{H}|\nu_\tau\rangle \\ \langle \nu_\tau|\hat{H}|\nu_e\rangle & \langle \nu_\tau|\hat{H}|\nu_\mu\rangle & \langle \nu_\tau|\hat{H}|\nu_\tau\rangle \end{pmatrix}. \tag{14}$$

The 3×3 module `NeuBin_F3C` also uses equation (6) to evolve a neutrino flavor wavefunction over a short step Δt. However, it is difficult to express $\exp(-\mathrm{i}H\Delta t)$ for the 3×3 case in an explicit form like equation (7). Instead, the module `NeuBin_F3C` solves the eigenvalue equations

$$\sum_j H_{ij} V_{jk} = \zeta_k V_{ik}, \qquad i, j, k = 1, 2, 3 \tag{15}$$

and uses

$$\exp(-\mathrm{i}H\Delta t) = V \begin{pmatrix} \mathrm{e}^{-\mathrm{i}\zeta_1 \Delta t} & & \\ & \mathrm{e}^{-\mathrm{i}\zeta_2 \Delta t} & \\ & & \mathrm{e}^{-\mathrm{i}\zeta_3 \Delta t} \end{pmatrix} V^\dagger. \tag{16}$$

Here, V_{ik} represents the ith component of the kth eigenvector of the Hamiltonian H, with corresponding eigenvalue ζ_k.

Module `NeuBin_F3C` employs an analytical method to solve equation (15). This method is based on the algorithm developed in [45] and takes into account the fact that the trace term in H is not relevant for neutrino flavor evolution. Assuming that

$$u_{11} + u_{22} + u_{33} = 0 \tag{17}$$

and, if necessary, shifting u_{ii} to make this sum 0, we define

$$p = -3\left[\left(u_{22}u_{33} - u_{11}^2\right) - \left(u_{12}^2 + w_{12}^2\right) - \left(u_{23}^2 + w_{23}^2\right) - \left(u_{31}^2 + w_{31}^2\right)\right], \tag{18}$$

$$\begin{aligned} q = & -\tfrac{27}{2}\left[u_{11}\left(u_{23}^2 + w_{23}^2\right) + u_{22}\left(u_{31}^2 + w_{31}^2\right) + u_{33}\left(u_{12}^2 + w_{12}^2\right) - u_{11}u_{22}u_{33}\right. \\ & \left. +2\left(-u_{12}u_{23}u_{31} + u_{12}w_{23}w_{31} + w_{12}u_{23}w_{31} + w_{12}w_{23}u_{31}\right)\right], \end{aligned} \tag{19}$$

$$\phi = \arccos(q). \tag{20}$$

The eigenvalues can be written as

$$\zeta_1 = \frac{2\sqrt{p}}{3}\cos\phi, \tag{21a}$$

$$\zeta_2 = \frac{2\sqrt{p}}{3}\cos\left(\phi + \frac{2\pi}{3}\right), \tag{21b}$$

$$\zeta_3 = -\zeta_1 - \zeta_2. \tag{21c}$$

From equation (15), it can be seen that the eigenvector $\boldsymbol{V}_i$ (with corresponding eigenvalue ζ_i) is orthogonal to the complex conjugates of the row vectors in matrix $H - \zeta_i I$. Assuming that $\boldsymbol{X}^{(1)}$ and $\boldsymbol{X}^{(2)}$ are two linearly independent row vectors in matrix $H - \zeta_i I$, we can write

$$\boldsymbol{V}_i = \frac{\boldsymbol{X}^{(1)} \times \boldsymbol{X}^{(2)}}{|\boldsymbol{X}^{(1)} \times \boldsymbol{X}^{(2)}|}. \tag{22}$$

Because the eigenvectors of H are orthogonal to each other, it is necessary to solve for only two eigenvectors, say $\boldsymbol{V}_1$ and $\boldsymbol{V}_2$. The third eigenvector, $\boldsymbol{V}_3$, can be determined from

$$\boldsymbol{V}_3 = \boldsymbol{V}_1^* \times \boldsymbol{V}_2^*. \tag{23}$$

2.4. `NBGroup_PM` modules and the parallelization of FLAT

Module `NBGroup_MM` is a single-thread version of an `NBGroup_PM` module. This single thread implementation uses an algorithm, similar to the modified midpoint method [46], to determine $|\psi_\lambda(r + \Delta r)\rangle$ from $|\psi_\lambda^{(0)}(r)\rangle$ according to equation (1). (Note that the relation between the radial coordinate r and the affine parameter value t for each mode λ depends on the trajectory direction of this neutrino mode.) The algorithm consists of three parts. In the first part, an approximation to $|\psi_\lambda(r + \Delta r)\rangle$ is obtained using radial coordinate step size Δr:

Step 1. Call `NBGroup::set_bg()` to set the background Hamiltonian $\hat{H}_{\rm bg}^{(0)}(\vartheta_0) \equiv \hat{H}_{\rm matt}(n_e(r)) + \hat{H}_{\nu\nu}(\psi^{(0)}, r, \vartheta_0)$ for each neutrino trajectory ϑ_0. This function utilizes flavor states $|\psi_{\lambda'}^{(0)}(r)\rangle$ for all neutrino modes λ' in the problem.

Step 2. Find $|\psi_\lambda^{(1)}(r+\Delta r)\rangle := \hat{U}_\lambda(\hat{H}_{\rm bg}^{(0)}, r, \Delta r)|\psi_\lambda^{(0)}(r)\rangle$ for each neutrino mode λ. Here $\hat{U}_\lambda(\hat{H}_{\rm bg}^{(0)}, r, \Delta r)$ is the operator evolving $|\psi_\lambda\rangle$ through a small step. This 'parameter evolution operator' is implemented as `NeuBin::evolve()`. The step in mode λ's affine parameter $t_{\vartheta_0(\lambda)}$ corresponding to Δr is $\Delta t_{\vartheta_0}(r, \Delta r) = t_{\vartheta_0}(r + \Delta r) - t_{\vartheta_0}(r)$.

Step 3. Compute $\hat{H}_{\rm bg}^{(1)}(\vartheta_0) \equiv \hat{H}_{\rm matt}(n_e(r + \Delta r)) + \hat{H}_{\nu\nu}(\psi^{(1)}, r + \Delta r, \vartheta_0)$ for each neutrino trajectory ϑ_0.

Step 4. Find $|\psi_\lambda^{(2)}(r + \Delta r)\rangle := \hat{U}_\lambda(\hat{H}_{\rm bg}^{(1)}, r, \Delta r)|\psi_\lambda^{(0)}(r)\rangle$ for each neutrino mode λ.

Step 5. Find $|\psi_\lambda^{(3)}(r + \Delta r)\rangle$ as an average of $|\psi_\lambda^{(1)}(r + \Delta r)\rangle$ and $|\psi_\lambda^{(2)}(r + \Delta r)\rangle$ for each neutrino mode λ.

The second part of the algorithm essentially repeats the first part, but with the step size reduced to $\frac{1}{2}\Delta r$:

Step 6. Find $|\psi_\lambda^{(4)}(r + \frac{1}{2}\Delta r)\rangle := \hat{U}(\hat{H}_{\rm bg}^{(0)}, r, \frac{1}{2}\Delta r)|\psi_\lambda^{(0)}(r)\rangle$.

Step 7. Compute $\hat{H}_{\rm bg}^{(4)}(\vartheta_0) \equiv \hat{H}_{\rm matt}(n_e(r + \frac{1}{2}\Delta r)) + \hat{H}_{\nu\nu}(\psi^{(4)}, r + \frac{1}{2}\Delta r, \vartheta_0)$ for each neutrino trajectory ϑ_0.

Step 8. Find $|\psi_\lambda^{(5)}(r + \Delta r)\rangle := \hat{U}(\hat{H}_{\rm bg}^{(4)}, r, r + \Delta r)|\psi_\lambda^{(0)}(r)\rangle$.

Step 9. Compute $\hat{H}_{\rm bg}^{(5)}(\vartheta_0) \equiv \hat{H}_{\rm matt}(n_e(r + \Delta r)) + \hat{H}_{\nu\nu}(\psi^{(5)}, r + \Delta r, \vartheta_0)$ for each neutrino trajectory ϑ_0.

Step 10. Find $|\psi_\lambda^{(6)}(r + \Delta r)\rangle := \hat{U}(\hat{H}_{\rm bg}^{(5)}, r + \frac{1}{2}\Delta r, \frac{1}{2}\Delta r)|\psi_\lambda^{(4)}(r + \frac{1}{2}\Delta r)\rangle$.

Step 11. Find $|\psi_\lambda^{(7)}(r + \Delta r)\rangle$ as an average of $|\psi_\lambda^{(5)}(r + \Delta r)\rangle$ and $|\psi_\lambda^{(6)}(r + \Delta r)\rangle$ for each neutrino mode λ.

The last part of the algorithm estimates the numerical error ϵ and determines the course of subsequent computation:

Step 12. Estimate the numerical error ϵ to be $\frac{1}{3}$ of the maximum of the differences between $|\psi_\lambda^{(3)}(r+\Delta r)\rangle$ and $|\psi_\lambda^{(7)}(r + \Delta r)\rangle$ in all neutrino modes.

Step 13. Set the new step size to be $\Delta r := \xi\sqrt{\epsilon_0/\epsilon}$, where ξ is an empirical constant, and ϵ_0 is the prescribed error tolerance.

Step 14. Set $|\psi_\lambda(r + \Delta r)\rangle := |\psi_\lambda^{(7)}(r + \Delta r)\rangle$ if $\epsilon \leqslant \epsilon_0$. Return to step 1 and repeat the algorithm as necessary.

Module `NBGroup_MM_SMP` is a parallelized version of `NBGroup_MM` using Pthreads. In `NBGroup_MM_SMP`, the algorithm described above is rearranged into a multi-stage pipeline through which all neutrino states must flow. Each working thread first enters a thread-mutual-exclusive routine `NBGroup_PM::new_task()`. In this routine, the thread is assigned a subset of neutrino states $|\psi_\lambda\rangle$ at a particular stage. After finishing the operations on $|\psi_\lambda\rangle$ at this stage, the thread re-enters `NBGroup_PM::new_task()`. Subsequently, the thread will be reassigned with another subset consisting of any remaining neutrino states at the same stage. If there are no remaining states, and if the dependencies are clear, `NBGroup_PM::new_task()` will assign to the working thread a subset of neutrino states at the next stage in the calculation. The stages are arranged so that there is no dependence between neighboring stages except the last two. This is done as follows:

Stage 1. Step 6 in the above algorithm. The last working thread at this stage completes step 7.

Stage 2. Step 2. The last thread also completes step 3.

Stage 3. Step 8. The last thread also completes step 9.

Stage 4. Steps 4 and 5.

Stage 5. Steps 10 and 11.

Stage 6. Steps 12, 13, 14 and 1.

FLAT has been designed to take advantage of computing nodes which are capable of running multiple threads simultaneously, e.g. those nodes with multi-cores, multiple-CPUs and/or simultaneous multithreading (SMT) technology. Multiple threads can be exploited by *FLAT* by swapping out the `NBGroup_MM` module and replacing it with `NBGroup_MM_SMP`. Because Pthreads share a common memory space and do not need to pass messages to each other, the parallelization of *FLAT* using `NBGroup_MM_SMP` is quite efficient. On an IBM 32-way Power4 computing node, multi-angle code with 32 working threads can run as fast as $\sim 31.2\times$ the speed of the single-threaded version. On a single-CPU Pentium4 desktop computer with hyper-threading (HT) technology enabled, threaded single-angle calculations can run $\gtrsim$ 30% faster than non-threaded calculations.

In order to utilize multiple computing nodes, *FLAT* is also parallelized using message passing interface (MPI)[4] in a manner similar to the implementation in *BULB* (see section 2.5). This has been done by rewriting portions of the `NBGroup_EM` and `NBGroup_IO` modules. For example, function `NBGroup_EM::set_bg()` of the `NBGroup_PNS2D` module can be modified to compute the neutrino self-coupling $\hat{H}_{\nu\nu}$ based on not only the neutrino states on the residing node but also the states on other nodes. Because `NBGroup_PM` modules are untouched in this scheme, an MPI/Pthread hybrid parallel model can be realized by choosing the `NBGroup_MM_SMP` module together with the MPI version of the `NBGroup_PNS2D` and `NBGroup_NETCDF` modules.

2.5. The structure of BULB

We have developed a second code called *BULB* to independently verify the results of the first code. *BULB*, developed at LANL, is written in Fortran90 and uses the MPI library to communicate between processes. As such it is suitable for use on many modern large-scale homogeneous computers.

BULB is parallelized by splitting neutrino trajectories into different groups of emission angles. In this implementation, each node independently handles all neutrino energy variables for each angle group. This is acceptable for machines of up to 500–1000 nodes, as the largest calculations can be split into one angle per processor. For larger machines it would also be desirable to split different energy bins among different

[4] *MPI: A Message-Passing Interface Standard* available online at http://www.mpi-forum.org/

processors. This may be needed in the future for more complex, non-spherical geometries where more than roughly 500 energy bins and 500 angle bins may be required.

The numerical integration algorithm on each node is a fairly simple treatment of nonlinear coupled differential equations. For each step, we choose an initial step size Δr, evaluating the matter background potential $H_{\rm matt}$ as an average over the initial and final points. In analogy to the procedure described above for *FLAT*, this increment in radius can be translated into the increment in the affine parameter Δt along each individual trajectory.

Initially, we assume that the background neutrino potential is constant over the step. The overall Hamiltonian H can then be diagonalized efficiently for 2×2 or 3×3 neutrino cases, and the propagated wavefunction at the end of the step is $\psi_{\rm fin} = \exp(-{\rm i}H\Delta t)\psi_{\rm ini}$ (see section 2.3). When the neutrino–neutrino scattering contribution is small, this formulation can allow for significant propagation distances. The step sizes in this case are bounded by the scale height of the matter profile or the length scales related to the fluctuations in the background electron potential. This formulation also preserves the unitarity of the neutrino density matrix.

Upon completion of this initial propagation, the final wavefunction is used to compute a revised background neutrino potential $H_{\nu\nu}$ at the final point. This new potential is then averaged with the initial potential to produce a neutrino potential averaged over the propagation step, similar to the treatment of the electron background potential. An improved final wavefunction is obtained by propagating with this averaged interaction. In principle this averaging scheme can be iterated, though in practice it is nearly always converged after the first iteration.

As in *FLAT*, in *BULB* we check for convergence with respect to the step size by comparing the final density matrix obtained in a single step δr with that obtained by taking two steps with half the step size. If the density matrix does not match within a predefined tolerance, the step size is reduced and the process is repeated. Periodically, we attempt to increase the step size in order to efficiently treat the large radius regions where the neutrino background potential and the electron density are fairly small.

A clear limitation in all these problems is that for each step one must calculate the full neutrino density matrix and the angular dependence of the neutrino potential for each energy and angle. This involves a sum over all energies, which in the present version of *BULB* is performed independently on each node. Calculating neutrino potentials also involves a sum over angles, as the neutrino interaction depends upon $\cos\vartheta_{\boldsymbol{pq}}$, where $\cos\vartheta_{\boldsymbol{pq}}$ is the angle between two neutrinos with momenta $\boldsymbol{p}$ and $\boldsymbol{q}$, respectively.

For spherical symmetry the angular dependence can be retained by evaluating the following sums on each node:

$$\bar{\rho}_n = \sum_{j=1}^{N_{\rm th}} \sum_{i=1}^{N_{\rm e}} \rho(i, j), \tag{24a}$$

$$\bar{\rho}'_n = \sum_{j=1}^{N_{\rm th}} \cos\vartheta_j \sum_{i=1}^{N_{\rm e}} \rho(i, j), \tag{24b}$$

where index n labels the node, ϑ_j is the angle between the jth neutrino trajectory and the radial direction at the current radius, and the sum i and j run over the number of energy grid points $N_{\rm e}$ and the number of angles on the local node $N_{\rm th}$, respectively. The full neutrino potential for all angles can then be evaluated by using MPI global sum calls to sum both the angle-independent terms $\bar{\rho}$ and the angle-dependent terms $\bar{\rho}'$ and by using the formula

$$\cos\vartheta_{ij} = \cos\vartheta_i \cos\vartheta_j + \sin\vartheta_i \sin\vartheta_j \cos(\phi_i - \phi_j), \tag{25}$$

where ϕ_i and ϕ_j are the azimuthal angles of the two beams. In spherical symmetry, the second term vanishes because of the averaging over $(\phi_i - \phi_j)$ [28].

Convergence of the overall calculation is carefully checked by comparing results with different error tolerances and their associated step sizes, and by comparing results with different numbers of energy and angle bins. Although qualitative results can often be obtained with a modest number of angular bins, a rather

large number of energy and angle bins are required to get stable solutions. This can typically involve several hundred thousand coupled nonlinear differential equations.

A 'typical' calculation such as those described in the results section below involve $\sim$ (800 angle bins) $\times$ (400 energy bins). Near the surface of the proto-neutron star, the wavelengths are very small and integration step sizes can be as small as 1 mm.[5] By the time one reaches 20 km, or about twice the radius of the star, the relevant wavelengths have increased and the step size is of the order of 1 cm. The total computational time required for a single calculation is approximately 8000 CPU hours on a typical cluster with 2 GHz AMD Opteron processors. The computational time is roughly equally divided between computing the evolution on each node and gathering the neutrino densities across all nodes. Storage of flavor states for 1000 neutrino modes at various radii for later analysis requires approximately 4 GB of storage space.

For understanding the propagation of the neutrinos from small to large distances, *BULB* can also dump out the neutrino potentials or the full wavefunction at user-defined spatial intervals. Storing the full wavefunction is rather slow, so this is done less often than sums over the wavefunction or density matrix. The latter are sufficient to determine the rough survival probabilities that could be detected in a terrestrial observation.

3. Results

The first simulations carried out with both codes focused on the late time, hot bubble epoch of the core collapse supernova phenomenon. In this post-explosion environment, the matter density surrounding the proto-neutron star is high, but still lower than in the earlier shock re-heating epoch. In the hot bubble, the neutrino self-coupling Hamiltonian $\hat{H}_{\nu\nu}$ can be dominant in some regions and sub-dominant relative to the matter potential in others.

In our calculations we can adopt a variety of matter density profiles, neutrino and antineutrino luminosities, and neutrino and antineutrino energy spectra. For the specific example problem of late-time supernova neutrino flavor evolution, we adopt a simple analytical profile for electron number density:

$$n_{\rm e} \simeq (1.6 \times 10^{36}\,{\rm cm}^{-3}) Y_{\rm e} \exp\left(\frac{R_\nu - r}{0.18\,{\rm km}}\right) + (6.0 \times 10^{30}\,{\rm cm}^{-3}) Y_{\rm e} \left(\frac{10\,{\rm km}}{r}\right)^3, \tag{26}$$

where electron fraction is $Y_{\rm e} = 0.4$ and the radius of the neutrino sphere is $R_\nu = 11$ km. We assume that initial energy spectra for neutrinos are of two-parameter black body form

$$f_\nu(E_\nu) = \frac{1}{F_2(\eta_\nu)} \frac{1}{T_\nu^3} \frac{E_\nu^2}{\exp(E_\nu/T_\nu - \eta_\nu) + 1}, \tag{27}$$

where η_ν is the degeneracy parameter, T_ν is the neutrino temperature, and

$$F_k(\eta) \equiv \int_0^\infty \frac{x^k\,{\rm d}x}{\exp(x - \eta) + 1}. \tag{28}$$

For the late-time supernova environment we take $\langle E_{\nu_e}\rangle = 11$ MeV, $\langle E_{\bar{\nu}_{\rm e}}\rangle = 16$ MeV, $\langle E_{\nu_\tau}\rangle = \langle E_{\bar{\nu}_\tau}\rangle = 25$ MeV, and $\eta_{\nu_{\rm e}} = \eta_{\bar{\nu}_{\rm e}} = \eta_{\nu_\tau} = \eta_{\bar{\nu}_\tau} = 3$. Here by ν_τ and $\bar{\nu}_\tau$ we actually mean linear combinations of neutrinos and antineutrinos of the real μ and τ flavors, respectively. With these choices we have $T_{\nu_e} \simeq 2.76$ MeV, $T_{\bar{\nu}_{\rm e}} \simeq 4.01$ MeV, and $T_{\nu_\tau} = T_{\bar{\nu}_\tau} \simeq 6.26$ MeV.

For our example 2×2 mixing treatment presented below, we have taken the effective mixing angle to be $\theta = 0.1$, and mass-squared difference to be $\Delta m^2 = \pm 3 \times 10^{-3}\,{\rm eV}^2$. Here the plus sign defines a so-called 'normal neutrino mass hierarchy', and the minus sign defines an 'inverted neutrino mass hierarchy'. There are six parameters in the full 3×3 vacuum mixing problem. These parameters consist of two mass-squared differences (Δm^2_{12} and Δm^2_{13}), three mixing angles (θ_{12}, θ_{13} and θ_{23}) and the *CP*-violating phase δ. Mass-squared differences Δm^2_{12} and $|\Delta m^2_{13}|$ and mixing angles θ_{12} and θ_{23} have been measured outright, while δ

[5] We note that the numbers given here can be contingent on the employed numerical algorithms and precision requirements. More efficient algorithms may allow fewer energy and/or angle bins and/or larger step sizes.

and the neutrino mass hierarchy are as yet unmeasured. In our 2×2 mixing treatment $\Delta m^2 \simeq \Delta m^2_{13}$ and $\theta \simeq \theta_{13}$. Although θ_{13} also remains unmeasured, current laboratory results suggest that $\sin^2(2\theta_{13}) \lesssim 0.1$.

Our preliminary single-angle two-flavor simulations gave very puzzling results. Contrary to the studies based on the MSW effect (e.g. [47–50]), these calculations showed that the flavor evolution histories of neutrinos with different energies could be coupled together and that all neutrinos and antineutrinos could experience collective flavor transformation. This could be true in our calculations even when the neutrino–neutrino forward-scattering potential was sub-dominant.

Analysis showed that the collective flavor transformation observed in our simulations was not the well-known synchronized flavor transformation seen in earlier single-angle calculations (e.g., [40, 41]). In synchronized flavor transformation, neutrinos and antineutrinos of all energies can undergo a common MSW-like flavor transformation in the region where a neutrino with a representative energy would experience a conventional MSW resonance. In addition, for neutrino–antineutrino gases that are dominated by neutrinos, synchronized flavor transformation can occur only in the normal neutrino mass hierarchy case. In our single-angle calculations, however, collective flavor transformation occurred for both the normal and inverted neutrino mass hierarchies and exhibited oscillatory behaviors over a relatively wide radius range (see, e.g., figure 8 in [28]).

Based on these preliminary results, we re-investigated collective flavor transformation in dense neutrino–antineutrino gases in the presence of ordinary matter [26]. We found that the oscillatory collective flavor transformation could be explained as 'bipolar oscillations'. This type of collective flavor transformation was first studied in the context of the early Universe [37, 51, 52].

Contrary to earlier widely accepted conclusions, our calculations and analyses showed that collective flavor transformation of the bipolar type can appear even in the presence of a large and dominant matter density. (By 'large and dominant matter density' we mean only that the neutrino–electron forward-scattering potential is larger than the neutrino–neutrino forward-scattering potential, not that inelastic neutrino-matter scattering is important.) This is a surprising and paradigm-changing result. In the past, the astrophysics community had assumed that neutrino flavor transformation in the high matter density environment near the neutron star would be suppressed (e.g. [40]). However, our results now compelled us to conclude that large-scale conversion of neutrino flavors could take place in such an environment. This conclusion was particularly surprising given the fact that the experimentally inferred neutrino mass-squared differences are small.

The neutrino and antineutrino trajectories in our calculations are labeled by emission angle ϑ_0. This is the angle between the trajectory and the normal to the neutrino sphere surface at the neutrino's or antineutrino's point of origin on this surface. For example, $\vartheta_0 = 0$ corresponds to a radially directed neutrino trajectory, while $\vartheta_0 = \pi/2$ corresponds to a neutrino moving tangentially to the neutrino sphere.

In the single-angle approximation, neutrinos and antineutrinos are assumed to have the same flavor evolution histories as those with the same energies but traveling along radial trajectories. To test the validity of this approximation, we carried out multi-angle calculations under the same conditions used in the single-angle cases, except of course for the obvious geometric issues associated with binning the trajectory directions [28, 53].

We first consider the artificial case, where we ignore neutrino self-coupling and consider only matter-driven neutrino and antineutrino flavor evolution. This is achieved by setting neutrino luminosities $L_{\nu_e} = L_{\bar{\nu}_e} = L_{\nu_\tau} = L_{\bar{\nu}_\tau} = 0\,\mathrm{erg\,s^{-1}}$. This case corresponds to the 'MSW' case which had been the paradigm for many if not most studies of neutrino flavor transformation effects for the supernova neutrino signal (e.g. [47–50]) and for nucleosynthesis (e.g. [54–56]). As we will see, this case fails to capture the essential physics of the problem.

The matter density is assumed here to be spherically symmetric. As a result, we expect the pure matter-driven MSW case to exhibit little dependence on neutrino or antineutrino emission angle (trajectory direction). Figure 3 shows a snapshot ($r \simeq 100\,\mathrm{km}$) of one of our simulations for this case. A movie presenting the full simulation is available at stacks.iop.org/CSD/1/015007/mmedia. There are several prominent features of neutrino and antineutrino flavor transformation in this scenario. We see no transformation of antineutrinos in this case, and neutrinos are transformed only at relatively low energy E_ν. This is classic MSW behavior: because the neutrino mass-squared difference is small, we must go to large radius before neutrinos with average energies are affected, and antineutrinos never are. And, indeed, there is little emission angle dependence in this case.

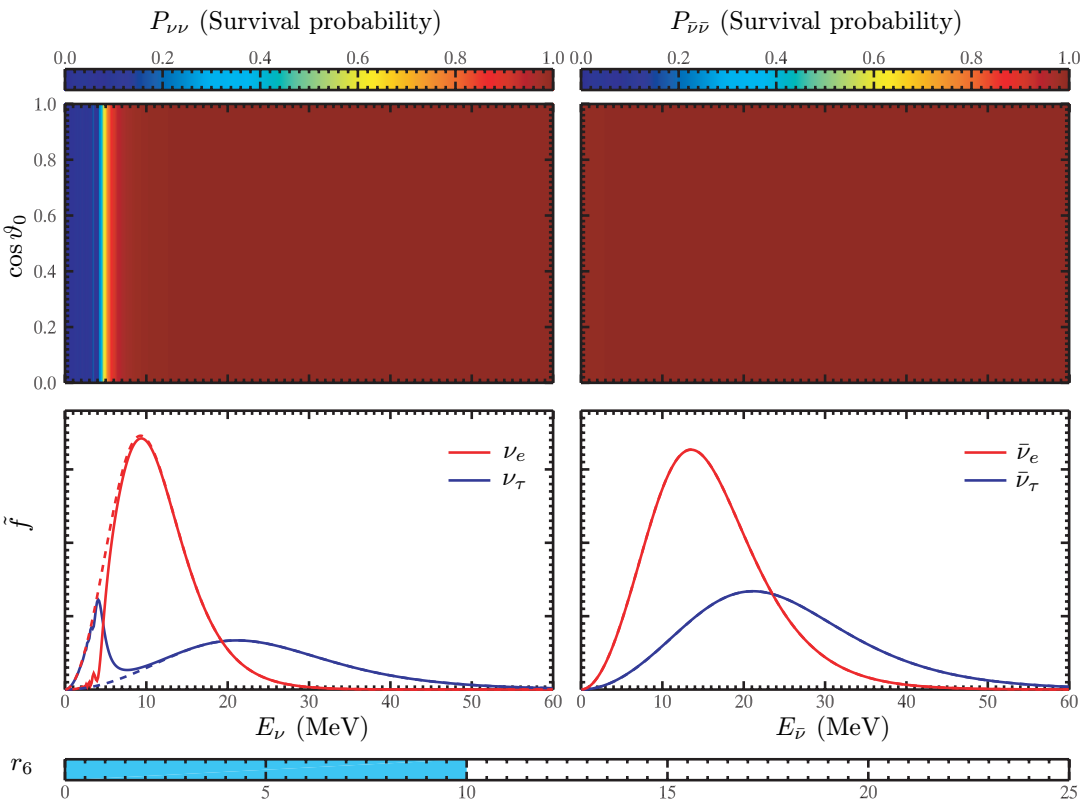

Figure 3. For the pure matter-driven MSW case with the normal neutrino mass hierarchy, we show neutrino survival probability $P_{\nu\nu}$ (upper left panel) and antineutrino survival probability $P_{\bar{\nu}\bar{\nu}}$ (upper right panel) as functions of neutrino energy E_ν or antineutrino energy $E_{\bar{\nu}}$ and cosine of the emission angle, $\cos\vartheta_0$, for values of r_6 (radius in units of 10^6 cm). The progress bar at the bottom shows r_6. Survival probability is given by the color code at the top of the panels; red corresponding to little or no flavor transformation, and blue corresponding to complete flavor conversion. Note that $\cos\vartheta_0$ is unity for radially propagating particles, and zero for tangentially propagating ones. In the bottom panels, the corresponding angle-averaged energy distribution functions $\tilde{f}$ (arbitrary normalization) are shown for neutrinos (lower left panel) or antineutrinos (lower right panel) for flavor e (τ) by the red (blue) curves. The dashed and solid curves in the bottom panels are for the neutrino energy spectra at the neutrino sphere and at the radius indicated by the progress bar, respectively. This figure corresponds to the snapshot at $r_6 \simeq 10$ in the full simulation. A movie presenting the full simulation is available at stacks.iop.org/CSD/1/015007/mmedia.

Figure 4 shows a snapshot ($r \simeq 100$ km) of one of our simulations for the pure matter-driven MSW case, but now for the inverted neutrino mass hierarchy. A movie presenting the full simulation is available at stacks.iop.org/CSD/1/015007/mmedia. These results are in some sense similar to those for the normal neutrino mass hierarchy case, except that here it is the low energy *antineutrino* flavors that are transformed. There is no neutrino flavor conversion in the inverted neutrino mass hierarchy case.

We can repeat these calculations for the same supernova model, but now with the full neutrino interaction Hamiltonian, including both the matter potential and the neutrino self-coupling potential $\hat{H}_{\nu\nu}$. The resulting neutrino and antineutrino flavor evolution is dramatically different from that in the MSW cases.

Figure 5 shows a snapshot ($r \simeq 90$ km) of one of our simulations for the normal neutrino mass hierarchy case but now with the full neutrino self-coupling Hamiltonian. A movie presenting the full simulation is available at stacks.iop.org/CSD/1/015007/mmedia. In this simulation, we have taken all neutrino species to have equal luminosities and $L_{\nu_e} = L_{\bar{\nu}_e} = L_{\nu_\tau} = L_{\bar{\nu}_\tau} = 10^{51}$ erg s^{-1}. This simulation shows that at values of radius, $r \sim 90$ km, both neutrinos and antineutrinos over broad energy ranges experience large-scale simultaneous flavor transformation. Although earlier analytical studies [26, 42] suggested that this behavior

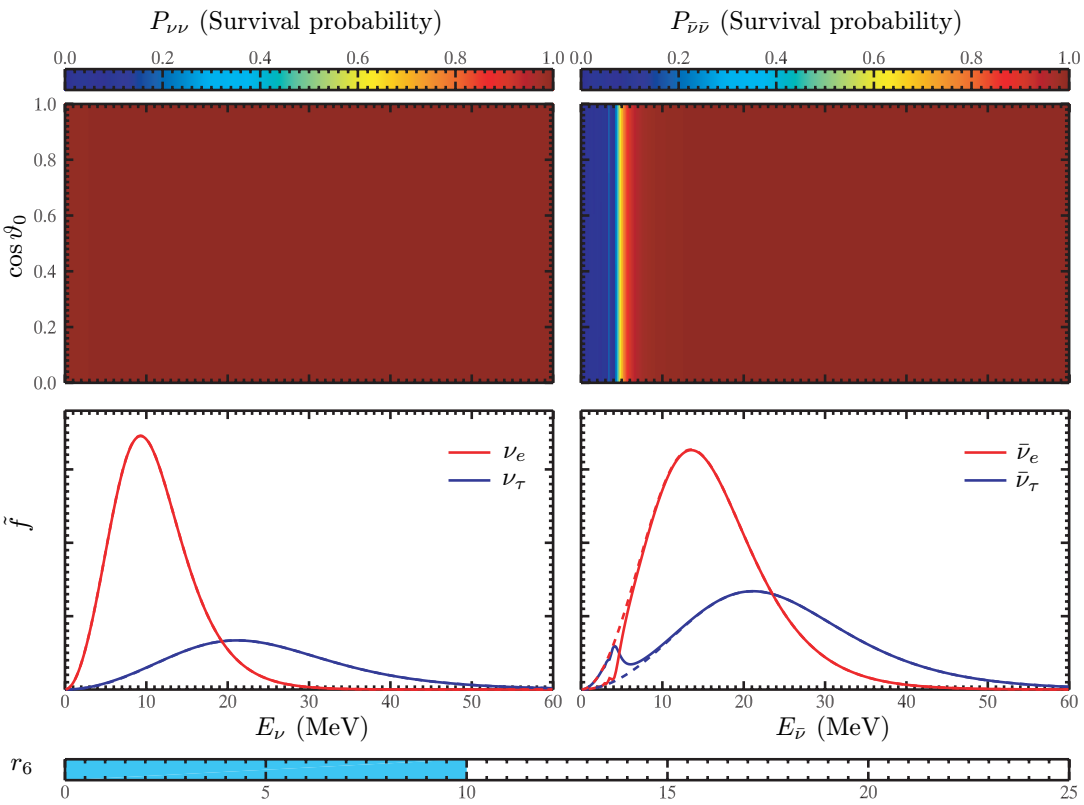

Figure 4. This figure is the same as figure 3, except now we show neutrino and antineutrino flavor transformation results for the pure matter-driven MSW case with the inverted neutrino mass hierarchy. This figure corresponds to the snapshot at $r_6 \simeq 10$ in the full simulation. A movie presenting the full simulation is available at stacks.iop.org/CSD/1/015007/mmedia.

was possible, this is nevertheless a surprising result as previous numerical simulations did not find this behavior in comparable or earlier epochs [40, 41]. In fact, collective flavor transformation in the supernova environment previously was thought to be of the synchronized, collective form [40]. However, Duan *et al* [57] show that the collective flavor transformation observed in this simulation is not synchronization, but rather a neutrino-background-enhanced MSW-like flavor transformation. This phenomenon occurs, where a neutrino with an energy representative of the neutrino–antineutrino gas would encounter a conventional MSW resonance. Because the representative energy of a dense neutrino–antineutrino gas is usually smaller than the average energy of each neutrino species, this type of collective flavor oscillation can occur deeper in the supernova envelope than one would expect for pure matter-driven MSW.

In our simulations, we find that neutrinos and antineutrinos emitted in different directions behave differently. Of course, this is in direct contradiction to the single-angle approximation. In our normal neutrino mass hierarchy simulations, large-scale flavor transformation is initiated by neutrinos and antineutrinos in the most tangentially propagating angle bins (i.e. those with low values of $\cos\vartheta_0$). As the radius increases, we can see from the full simulation that flavor conversion spreads out to other angle bins, eventually causing significant flavor transformation of both neutrinos and antineutrinos for nearly all energies and almost all trajectory directions.

The spreading of significant flavor transformation from tangential to radial directions has its origin in the structure of the weak current. More tangentially directed neutrinos or antineutrinos which forward-scatter on the more radially directed background neutrinos and antineutrinos bring larger flavor diagonal and off-diagonal potentials to the full self-coupling Hamiltonian $\hat{H}_{\nu\nu}$. This is a consequence of the intersection angle dependence of the forward neutrino–neutrino scattering amplitude [25, 28, 42, 53].

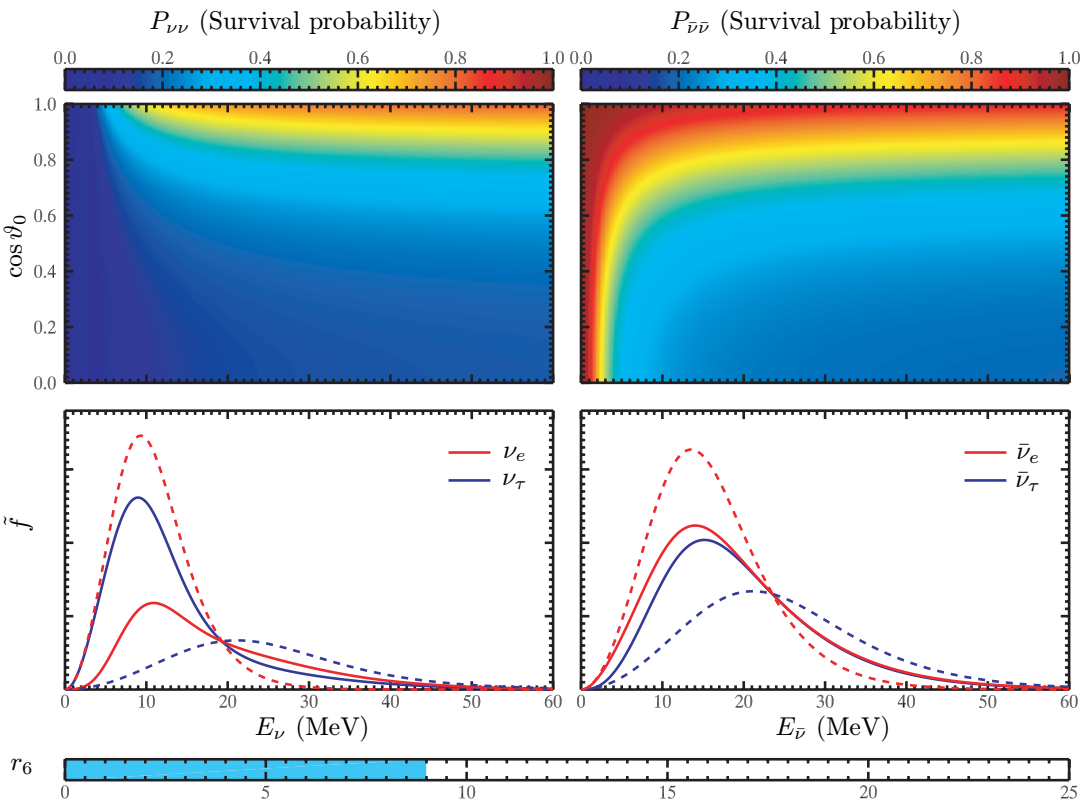

Figure 5. Neutrino and antineutrino flavor transformation results for the normal neutrino mass hierarchy case but now with full neutrino self-coupling and trajectory coupling. Set-ups and definitions are the same as in figure 3. Here, we take the luminosities of all neutrino species to be 10^{51} erg s^{-1}. This figure corresponds to a snapshot at $r_6 \simeq 9.0$ in the full simulation. A movie presenting the full simulation is available at stacks.iop.org/CSD/1/015007/mmedia.

As we move out to even larger radius in these simulations, neutrinos and antineutrinos experience collective bipolar oscillations. This behavior is evident in the full simulation. At large enough radius, the neutrino background becomes ineffective in influencing flavor transformation. The subsequent neutrino and antineutrino flavor evolution is akin to MSW or vacuum neutrino oscillations. However, as will be discussed below, the breakdown of collective flavor transformation can leave an important 'fossil' imprint of the nonlinear neutrino background.

In the inverted mass hierarchy, neutrino and antineutrino flavor evolution with self-coupling is different from the behavior in both the matter-driven MSW case and the normal neutrino mass hierarchy self-coupled case. Figure 6 shows a snapshot ($r \simeq 75$ km) of one of our simulations for the inverted neutrino mass hierarchy case with the full neutrino self-coupling Hamiltonian ($L_{\nu_e} = L_{\bar{\nu}_e} = L_{\nu_\tau} = L_{\bar{\nu}_\tau} = 10^{51}$ erg s^{-1}). A movie presenting the full simulation is available at stacks.iop.org/CSD/1/015007/mmedia. Like the normal mass hierarchy case, both neutrinos and antineutrinos can experience large-scale, and simultaneous flavor conversion over broad ranges of energies and for nearly all emission angles. However, in contrast to the normal mass hierarchy behavior, neutrinos and antineutrinos in the inverted mass hierarchy case do not experience MSW-like flavor transformation, but rather enter the bipolar oscillation mode directly at low radius. Subsequently, as the radius increases, more and more horizontal fringes appear in the survival probability $P_{\nu\nu}(\cos\vartheta_0, E_\nu)$ and $P_{\bar{\nu}\bar{\nu}}(\cos\vartheta_0, E_\nu)$. These horizontal fringes also appear, although to a less 'violent' extent, in the normal mass hierarchy case. This phenomenon is associated with multi-angle effects. The one-by-one excitation of multipoles in $\cos\vartheta_0$ so evident in our full simulation is explained by the 'kinematic decoherence' of bipolar oscillations [58].

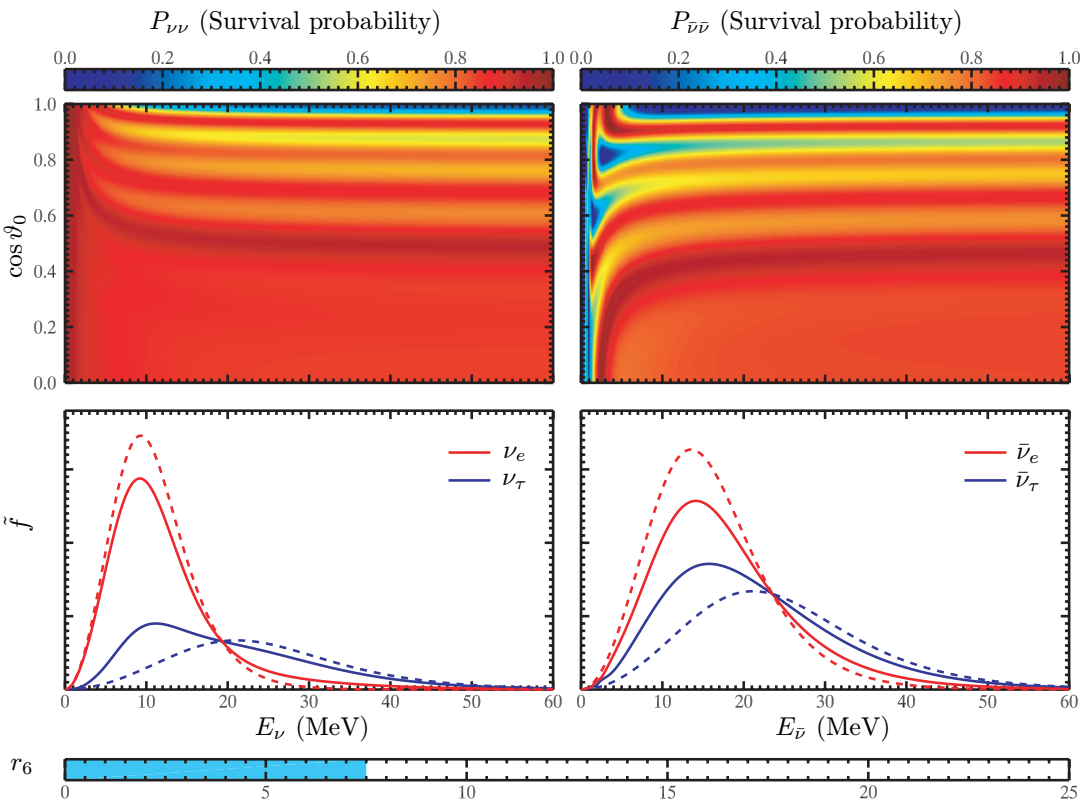

Figure 6. Neutrino and antineutrino flavor transformation results for the inverted neutrino mass hierarchy case but now with full neutrino self-coupling and trajectory coupling. Set-ups and definitions are the same as in figure 3. Here, we take the luminosities of all neutrino species to be $L_\nu = 10^{51}\,\mathrm{erg\,s^{-1}}$. This plot corresponds to the snapshot at $r_6 \simeq 7.5$ in the full simulation. A movie presenting the full simulation is available at stacks.iop.org/CSD/1/015007/mmedia.

Neutrino and antineutrino flavor transformation in the inverted neutrino mass hierarchy case sets in at a radius r_X, where the total neutrino flux drops below a critical value. This can occur closer to the neutron star than MSW-like transformation in the normal neutrino mass hierarchy case. Because r_X is solely determined by neutrino fluxes and energy spectra in this case [26], neutrinos and antineutrinos can experience large-scale flavor transformation even in earlier epochs, where the matter density is much higher. Indeed, a simple classical pendulum analogy shows that the inverted mass hierarchy gives rise to an inherently unstable neutrino flavor field, like a pencil standing on its tip [59, 60].

Although the single-angle approximation may not be accurate in general, the calculations based on this assumption do capture some if not most of the *qualitative* features present in multi-angle calculations. This has been shown in our simulations [28, 53] as well as in the computations carried out by other groups [61, 62].

One of the most interesting behaviors seen in both the single-angle and multi-angle calculations is stepwise spectral swapping (also known as the spectral split) [28, 53]. This is where, e.g. ν_e with energies larger or smaller than some transition energy E^s swap energy spectra and fluxes with neutrinos of another flavor, say ν_τ. The swapping feature is strikingly obvious in, e.g., the later part (larger radius part) of the simulation that corresponds to figures 5. Note the vertical (i.e. for all ϑ_0) line of demarcation between near-zero survival probability for neutrinos with $E_\nu < E^s \simeq 10\,\mathrm{MeV}$ and significant survival probability for more energetic neutrinos. 'Swapping' is an apt description of this phenomenon in the sense that there is nearly complete conversion of neutrino flavors across a broad range of neutrino energy E_ν. Whether the swapping of spectra occurs for values of E_ν above or below E^s depends on the neutrino mass hierarchy. In the normal (inverted) neutrino mass hierarchy, the swap occurs for neutrinos with energies satisfying $E_\nu < E^s$ ($E_\nu > E^s$). An explanation of this phenomenon in the single-angle approximation context has been given by [28, 57, 60, 63, 64].

In our simulations with the normal neutrino mass hierarchy, the swap energy E^{s} decreases as the effective mixing angle θ ($\simeq \theta_{13}$) is decreased. The situation in the inverted neutrino mass hierarchy is different. As alluded to above, there is 'instability' in flavor transformation in the inverted mass scheme. Indeed, our simulations suggest that a swap can be obtained in this scheme even for extremely small values of θ [65].

The direct detection of stepwise swapping in the energy spectra of supernova neutrinos could provide a unique way to determine the neutrino mass hierarchy, even if θ_{13} is too small to be measured in conventional terrestrial neutrino oscillation experiments [65]. This is an important result that could change future supernova neutrino detection strategies. The swap energy is relatively small, $E^{\mathrm{s}} \sim 10$ MeV, of the order of solar neutrino energies. These facts may suggest that supernova neutrino detection schemes should focus on low energy, low flux ν_es at a few seconds post-core-bounce.

We have also run three-neutrino (i.e. 3×3) flavor mixing simulations, where all three neutrino flavors are followed simultaneously and self-consistently. Given the wealth of unexpected phenomena revealed by the nonlinear 2×2 mixing calculations, it would be prudent to investigate whether going to a full 3×3 mixing scheme might change things yet again. Sometimes the full three-neutrino mixing problem should be reducible to two separate two-flavor problems [48, 66–68]. However, this is not always possible.

We have run both single-angle and multi-angle tests which employ the full three-flavor mixing scheme for one particular case: the neutronization burst from an O–Ne–Mg core-collapse supernova [29, 69]. The neutronization neutrino burst occurs when the supernova bounce shock comes through the neutrino sphere at $\sim$10 ms post-core-bounce. O–Ne–Mg core collapse supernovae have as progenitors stars with masses in the range 8–12 $M_{\odot}$. These are very interesting from a neutrino physics standpoint because there is little matter overburden for the shock to transit and, as a result, the neutrino background can become important very early on even in the normal neutrino mass hierarchy case. For this case, in the normal neutrino mass hierarchy, our 3×3 numerical calculation shows two stepwise spectral swaps for neutrinos in the vacuum mass basis, one on the top of the other. This result seems to suggest that, at least for this particular case, the full three-flavor mixing problem is indeed reducible to two separate two-flavor mixing schemes. The result for the inverted mass hierarchy case is similar, except that one of the spectral swaps is forbidden because of the conservation of a 'lepton number' [59, 63].

4. Frontiers

Since the publication of our first numerical results [28, 53], there has been dramatic recent progress in understanding the evolution of the neutrino flavor field in the supernova environment. Numerical simulations have been carried out by several groups using various matter density profiles [61, 62, 69–71]. All of these calculations have produced qualitatively similar results. This is as expected because, as we have shown in [26], collective neutrino oscillations are generally not sensitive to matter densities except for the MSW-like collective conversion in the normal neutrino mass hierarchy case.

Most importantly, the physical mechanism behind collective neutrino oscillations, especially the formation of neutrino energy spectrum swapping, are now well understood [57, 59, 60, 63, 64, 71–73]. In order to obtain an analytical understanding of the problem, these studies usually have assumed the single-angle approximation (however, see [58]). In the single-angle approximation, the angular dependence of neutrino–neutrino coupling is assumed to be the same as that in an isotropic environment (see, e.g., [28]). The neutrino bulb model adopted in our calculations, although a fairly idealized supernova model, offers a way to self-consistently probe collective neutrino oscillations in an anisotropic environment. The agreement between the multi-angle and single-angle calculations suggest that the existence of collective neutrino oscillations can be configuration independent. Nevertheless, our calculations can be improved in several ways.

As recent multi-dimensional hydrodynamics simulations have shown (e.g. [10, 13]), the actual supernova matter density could be quite inhomogeneous. In addition, it is possible that the neutrino flux and angular distributions could deviate significantly from the idealized spherically symmetric models employed in our numerical simulations. Nevertheless we believe that our spherically symmetric coherent regime calculations capture the qualitative features of the collective flavor oscillation modes discussed above. But a road map for

future work should include a plan for checking this assertion. As a first step, it may be useful to introduce inhomogeneities in the neutrino flux and/or the background electron densities in one or two dimensions.

One of the most significant limitations in current multi-angle simulations is the neutrino bulb model employed in these calculations. In the neutrino bulb model, neutrinos are assumed to be emitted isotropically from an infinitely thin layer (i.e. the neutrino sphere) on the surface of a spherically symmetric black-body (the neutron star). In addition, in current neutrino flavor transformation simulations the matter profile outside the neutron star is taken to depend only on the radius.

This model roughly corresponds to the 1D supernova simulations, but is nowhere near the sophistication of those calculations. For example, it is well known that neutrinos with different flavors and with different energies experience different opacities and, therefore, decouple from the matter fields at different radii. Even in the context of 1D supernova models, the initial neutrino energy spectra will show deviations from black-body form. Moreover, the neutrino sphere has a finite thickness. This will alter the relationship between length parameters and emission angles on neutrino trajectories.

Recent sophisticated supernova simulations suggest that 3D features might play significant roles in how massive stars eventually collapse. Needless to say, neutrino emission would not be isotropic in these simulations, and the matter profile obviously would not be spherically symmetric either. In order for *FLAT* to treat the 3D supernova features in neutrino flavor evolution, a new `NBGroup_EM` module would be necessary. This module would have to model correctly the physics and geometry of the problem. *BULB* would have to be modified in corresponding fashion. Of course, actually implementing such modifications in either code would significantly enlarge the dimensionality of the problem space and, once achieved, could increase the computation time by several orders of magnitude.

Flavor evolution of neutrinos emitted from the accretion disk around a massive black hole is an interesting example of non-sphericity. Even the simplest accretion disk model has a more complicated geometry than the neutrino bulb. In principle, this problem could be approached in a manner similar to the 1D to 3D upgrade, i.e. by building a new `NBGroup_EM` module to treat the special geometric structure of the accretion disk.

Yet another interesting problem is the flavor evolution of the neutrinos and antineutrinos in the early Universe. This was the environment, where the first full nonlinear numerical simulations of neutrino flavor evolution were performed [36, 37, 51, 52, 74–76]. These early explorations revealed several interesting features of flavor evolution in dense neutrino gases. These features included, for example, synchronized and bipolar collective neutrino oscillations. This work has had important implications for models of lepton-degenerate cosmologies (see, e.g. [77–79]). Because the early Universe and supernovae have, in some sense, similar physical environments, the interesting phenomena recently discovered in the supernova case might also occur in the early Universe. It could be worthwhile to re-investigate neutrino flavor evolution in the early Universe too, for example, to see if bipolar neutrino oscillations might have consequences for models for lepton number affected nucleosynthesis.

Following neutrino and antineutrino flavor evolution in astrophysical environments is a difficult problem. However, astrophysicists are compelled to solve this problem because of the potential leverage neutrino flavor transformation has in supernova and early Universe physics, and because neutrino flavor mixing is an experimental reality. We believe that the solution of the astrophysical neutrino flavor transformation problem could usher in a new era where there is a synergistic coupling of terrestrial laboratory-based neutrino physics and astrophysical observations.

Acknowledgments

This work was supported in part by NSF grant PHY-04-00359, a LANL/CARE grant, and TSI collaboration's DOE SciDAC grant at UCSD, DOE grant DE-FG02-00ER41132 at INT, and by the DOE Office of Nuclear Physics, the LDRD Program and Open Supercomputing at LANL, and an IGPP/LANL mini-grant at UCSD. This work was also supported in part by NERSC through TSI collaboration using Bassi, SDSC through AAP using DataStar, and LCF at NCCS using Jaguar.

References

[1] Woosley S and Janka T 2005 *Nat. Phys.* **1** 147 (*Preprint* astro-ph/0601261)
[2] Cardall C Y 2007 *Preprint* astro-ph/0701831
[3] Colgate S A and White R H 1966 *Astrophys. J.* **143** 626
[4] Bowers R and Wilson J R 1982 *Astrophys. J.* **263** 366
[5] Bruenn S W 1985 *Astrophys. J. Suppl.* **58** 771
[6] Myra E S, Bludman S A, Hoffman Y, Lichenstadt I, Sack N and van Riper K A 1987 *Astrophys. J.* **318** 744
[7] Mezzacappa A *et al* 2001 *Phys. Rev. Lett.* **86** 1935 (*Preprint* astro-ph/0005366)
[8] Fryer C L and Warren M S 2004 *Astrophys. J.* **601** 391 (*Preprint* astro-ph/0309539)
[9] Walder R, Burrows A, Ott C D, Livne E and Jarrah M 2005 *Astrophys. J.* **626** 317 (*Preprint* astro-ph/0412187)
[10] Kifonidis K, Plewa T, Scheck L, Janka H T and Mueller E 2006 *Astron. Astrophys.* **453** 661 (*Preprint* astro-ph/0511369)
[11] Buras R, Rampp M, Janka H T and Kifonidis K 2006 *Astron. Astrophys.* **447** 1049–92 (*Preprint* astro-ph/0507135)
[12] Scheck L, Kifonidis K, Janka H T and Mueller E 2006 *Astron. Astrophys.* **457** 963 (*Preprint* astro-ph/0601302)
[13] Blondin J M and Mezzacappa A 2006 *Nature* **445** 58 (*Preprint* astro-ph/0611680)
[14] Yao W M *et al* 2006 *J. Phys. G: Nucl. Part. Phys.* **33** 1
[15] Wolfenstein L 1978 *Phys. Rev.* D **17** 2369
[16] Wolfenstein L 1979 *Phys. Rev.* D **20** 2634
[17] Mikheyev S P and Smirnov A Y 1985 *Yad. Fiz.* **42** 1441
Mikheyev S P and Smirnov A Y 1985 *Sov. J. Nucl. Phys.* **42** 913 (Engl. Transl.)
[18] Sigl G and Raffelt G 1993 *Nucl. Phys.* B **406** 423
[19] Yamada S 2000 *Phys. Rev.* D **62** 093026 (*Preprint* astro-ph/0002502)
[20] Abazajian K, Fuller G M and Patel M 2001 *Phys. Rev.* D **64** 023501 (*Preprint* astro-ph/0101524)
[21] Strack P and Burrows A 2005 *Phys. Rev.* D **71** 093004 (*Preprint* hep-ph/0504035)
[22] Hidaka J and Fuller G M 2006 *Phys. Rev.* D **74** 125015 (*Preprint* astro-ph/0609425)
[23] Hidaka J and Fuller G M 2007 *Phys. Rev.* D **76** 083516 (*Preprint* 0706.3886 [astro-ph])
[24] Cardall C Y 2007 *Preprint* 0712.1188 [astro-ph]
[25] Fuller G M, Mayle R W, Wilson J R and Schramm D N 1987 *Astrophys. J.* **322** 795
[26] Duan H, Fuller G M and Qian Y Z 2006 *Phys. Rev.* D **74** 123004 (*Preprint* astro-ph/0511275)
[27] Halprin A 1986 *Phys. Rev.* D **34** 3462
[28] Duan H, Fuller G M, Carlson J and Qian Y Z 2006 *Phys. Rev.* D **74** 105014 (*Preprint* astro-ph/0606616)
[29] Duan H, Fuller G M, Carlson J and Qian Y Z 2008 *Phys. Rev. Lett.* **100** 021101 (*Preprint* 0710.1271 [astro-ph])
[30] Schirato R C and Fuller G M 2002 *Preprint* astro-ph/0205390
[31] Tomas R *et al* 2004 *J. Cosmol. Astropart. Phys.* JCAP09(2004)015 (*Preprint* astro-ph/040732)
[32] Sawyer R F 1990 *Phys. Rev.* D **42** 3908
[33] Loreti F N, Qian Y Z, Fuller G M and Balantekin A B 1995 *Phys. Rev.* D **52** 6664 (*Preprint* astro-ph/9508106)
[34] Friedland A and Gruzinov A 2006 *Preprint* astro-ph/0607244
[35] Kneller J P, McLaughlin G C and Brockman J 2008 *Phys. Rev.* D **77** 045023 (*Preprint* 0705.3835 [astro-ph])
[36] Samuel S 1993 *Phys. Rev.* D **48** 1462
[37] Kostelecky V A and Samuel S 1995 *Phys. Rev.* D **52** 621 (*Preprint* hep-ph/9506262)
[38] Qian Y Z and Fuller G M 1995 *Phys. Rev.* D **51** 1479 (*Preprint* astro-ph/9406073)
[39] Pastor S, Raffelt G G and Semikoz D V 2002 *Phys. Rev.* D **65** 053011 (*Preprint* hep-ph/0109035)
[40] Pastor S and Raffelt G 2002 *Phys. Rev. Lett.* **89** 191101 (*Preprint* astro-ph/0207281)
[41] Balantekin A B and Yüksel H 2005 *New J. Phys.* **7** 51 (*Preprint* astro-ph/0411159)
[42] Fuller G M and Qian Y Z 2006 *Phys. Rev.* D **73** 023004 (*Preprint* astro-ph/0505240)
[43] Rew R K and Davis G P 1990 *IEEE Comput. Graph. Appl.* **10** 76 (available online at http://www.unidata.ucar.edu/software/netcdf/)
[44] 1003.1, 2004 Edition IEEE *Standard for Information Technology—Portable Operating System Interface (POSIX): Base Definitions* (The Open Group Technical Standard Base Specifications, Issue 6)
[45] Kopp J 2006 *Preprint* physics/0610206
[46] Press W H, Teukolsky S A, Vetterling W T and Flannery B P 2002 *Numerical Recipes in* C++*: The Art of Scientific Computing* 2nd edn (Cambridge: Cambridge University Press)
[47] Kuo T K and Pantaleone J T 1988 *Phys. Rev.* D **37** 298
[48] Dighe A S and Smirnov A Y 2000 *Phys. Rev.* D **62** 033007 (*Preprint* hep-ph/9907423)
[49] Lunardini C and Smirnov A Y 2003 *J. Cosmol. Astropart. Phys.* JCAP06(2003)009 (*Preprint* hep-ph/0302033)

[50] Kneller J P and McLaughlin G C 2006 *Phys. Rev.* D **73** 056003 (*Preprint* hep-ph/0509356)
[51] Kostelecky V A and Samuel S 1993 *Phys. Lett.* B **318** 127
[52] Samuel S 1996 *Phys. Rev.* D **53** 5382 (*Preprint* hep-ph/9604341)
[53] Duan H, Fuller G M, Carlson J and Qian Y Z 2006 *Phys. Rev. Lett.* **97** 241101 (*Preprint* astro-ph/0608050)
[54] Heger A *et al* 2005 *Phys. Lett.* B **606** 258 (*Preprint* astro-ph/0307546)
[55] Yoshida T *et al* 2006 *Astrophys. J.* **649** 319 (*Preprint* astro-ph/0606042)
[56] Yoshida T *et al* 2006 *Phys. Rev. Lett.* **96** 091101 (*Preprint* astro-ph/0602195)
[57] Duan H, Fuller G M and Qian Y Z 2007 *Phys. Rev.* D **76** 085013 (*Preprint* 0706.4293 [astro-ph])
[58] Raffelt G G and Sigl G G R 2007 *Phys. Rev.* D **75** 083002 (*Preprint* hep-ph/0701182)
[59] Hannestad S, Raffelt G G, Sigl G and Wong Y Y Y 2006 *Phys. Rev.* D **74** 105010 (*Preprint* astro-ph/0608695)
[60] Duan H, Fuller G M, Carlson J and Qian Y Z 2007 *Phys. Rev.* D **75** 125005 (*Preprint* astro-ph/0703776)
[61] Esteban-Pretel A, Pastor S, Tomas R, Raffelt G G and Sigl G 2007 *Phys. Rev.* D **76** 125018 (*Preprint* 0706.2498 [astro-ph])
[62] Fogli G L, Lisi E, Marrone A and Mirizzi A 2007 *J. Cosomol. Astropart. Phys.* JCAP12(2007)010 (*Preprint* 0707.1998) [hep-ph]
[63] Raffelt G G and Smirnov A Y 2007 *Phys. Rev.* D **76** 081301 (*Preprint* 0706.3886 [hep-ph])
[64] Raffelt G G and Smirnov A Yu 2007 *Phys. Rev.* D **76** 125008 (*Preprint* 0709.4641 [hep-ph])
[65] Duan H, Fuller G M, Carlson J and Qian Y Z 2007 *Phys. Rev. Lett.* **99** 241802 (*Preprint* 0707.0290 [astro-ph])
[66] Kuo T K and Pantaleone J T 1986 *Phys. Rev. Lett.* **57** 1805
[67] Balantekin A B and Fuller G M 1999 *Phys. Lett.* B **471** 195 (*Preprint* hep-ph/9908465)
[68] Caldwell D O, Fuller G M and Qian Y Z 2000 *Phys. Rev.* D **61** 123005 (*Preprint* astro-ph/9910175)
[69] Duan H, Fuller G M, Carlson M and Qian Y Z 2008 in preparation
[70] Esteban-Pretel A, Pastor S, Tomas R, Raffelt G G and Sigl G 2008 *Phys. Rev.* D **77** 065024 (*Preprint* 0712.1137 [astro-ph])
[71] Duan H, Fuller G M and Qian Y Z 2008 *Phys. Rev.* D **77** 085016 (*Preprint* 0801.1363 [astro-ph])
[72] Dasgupta B and Dighe A 2007 *Preprint* 0712.3798 [hep-ph]
[73] Dasgupta B, Dighe A, Mirizzi A and Raffelt G G 2008 *Phys. Rev.* D **77** 113007 (*Preprint* 0801.1660 [hep-ph])
[74] Kostelecky V A and Samuel S 1994 *Phys. Rev.* D **49** 1740
[75] Kostelecky V A, Pantaleone J T and Samuel S 1993 *Phys. Lett.* B **315** 46
[76] Kostelecky V A and Samuel S 1996 *Phys. Lett.* B **385** 159 (*Preprint* hep-ph/9610399)
[77] Dolgov A D *et al* 2002 *Nucl. Phys.* B **632** 363 (*Preprint* hep-ph/0201287)
[78] Wong Y Y Y 2002 *Phys. Rev.* D **66** 025015 (*Preprint* hep-ph/0203180)
[79] Abazajian K N, Beacom J F and Bell N F 2002 *Phys. Rev.* D **66** 013008 (*Preprint* astro-ph/0203442)