Neutrino oscillation experiments will be entering the precision era in the next decade with the advent of high statistics experiments like DUNE, HK, and JUNO. Correctly estimating the confidence intervals from data for the oscillation parameters requires very large Monte Carlo data sets involving calculating the oscillation probabilities in matter many, many times. In this paper, we leverage past work to present a new, fast, precise technique for calculating neutrino oscillation probabilities in matter optimized for long-baseline neutrino oscillations in the Earth's crust including both accelerator and reactor experiments. For ease of use by theorists and experimentalists, we provide fast c++ and fortran codes.

NuFast: Fast and Accurate Algorithm for Calculating Long-Baseline Neutrino Oscillation Probabilities with Matter Effects

Peter B. Denton

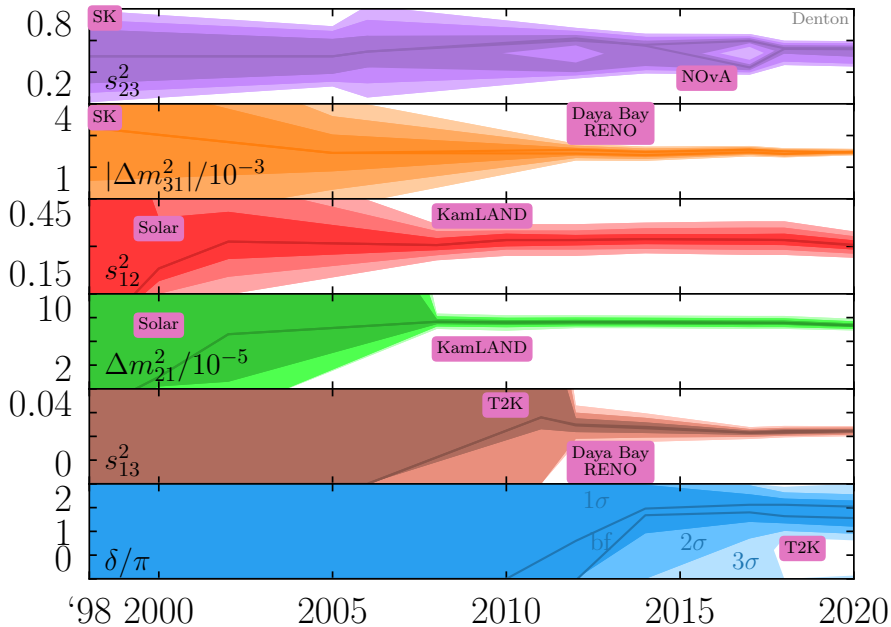DUNE Collaboration Meeting @ FNAL

May 22, 2024

2405.02400 with S. Parke

github.com/PeterDenton/NuFast
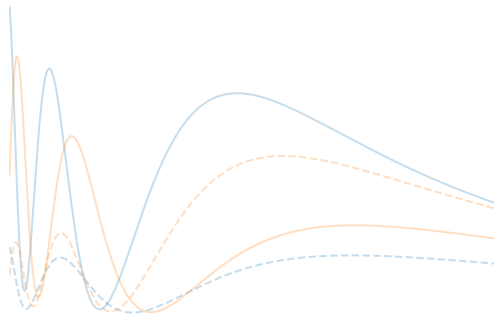
Brookhaven National Laboratory

Brookhaven NDI
Neutrino Discovery Initiative

The problem

$$\Delta m^2_{21}, \Delta m^2_{31}$$
$$s^2_{23}, s^2_{13}, s^2_{12}$$
$$\delta$$

$\Rightarrow$
in matter

Many approaches

<div align="center">

Solve the Schrödinger equation

$$i\frac{d}{dt}|\nu\rangle = H(t)|\nu\rangle$$

</div>

If $H(t) = H$ (constant density)

$$\mathcal{A}(\nu_\alpha \to \nu_\beta) = \left[e^{-iHL}\right]_{\alpha\beta} \qquad\qquad P = |\mathcal{A}|^2$$

Exponential requires computing eigenvalues and eigenvectors of H

Many approaches

Modify vacuum probabilities

► Get the eigenvalues by solving the cubic

<div align="right">

Cardano 1545
V. Barger, et al. PRD 22 (1980) 2718

</div>

► Get the eigenvectors

<div align="right">

H. Zaglauer, K. Schwarzer Z.Phys. C40 (1988) 273
K. Kimura, A. Takamura, H. Yokomakura hep-ph/0205295
PBD, S. Parke, X. Zhang 1907.02534
A. Abdulahi, S. Parke 2212.12565

</div>

Other approaches?
Are approximations useful?
Optimal hybrids?
What is the goal?

Fermilab computing experts bolster NOvA evidence, 1 million cores consumed

July 3, 2018 | Marcia Teckenbrock

Array

How do you arrive at the physical laws of the universe when you're given experimental data on a renegade particle that interacts so rarely with matter, it can cruise through light-years of lead? You call on the power of advanced computing.

The NOvA neutrino experiment, in collaboration with the Department of Energy's Scientific Discovery through Advanced Computing (SciDAC-4) program and the HEPCloud program at DOE's Fermi National Accelerator Laboratory, was able to perform the largest-scale analysis ever to support the recent evidence of antineutrino oscillation, a phenomenon that may hold clues to how our universe evolved.

Using Cori, the newest supercomputer at the National Energy Research Scientific Computing Center (NERSC), located at Lawrence Berkeley National Laboratory, NOvA used over 1 million computing cores, or CPUs, between May 14 and 15 and over a short timeframe one week later. This is the largest number of CPUs ever used concurrently over this duration — about 54 hours — for a single high-energy physics experiment. This unprecedented amount of computing enabled scientists to carry out some of the most complicated techniques used in neutrino physics, allowing them to dig deeper into the seldom seen interactions of neutrinos. This Cori allocation was more than 400 times the amount of Fermilab computing allocated to the NOvA experiment and 50 times the total computing capacity of Fermilab allocated for all of its rare-physics experiments. A continuation of the analysis was performed on NERSC's Cori and Edison supercomputers one week later. In total, nearly 35 million core-hours were consumed by NOvA in the 54-hour period. Executing the same analysis on a single desktop computer would take 4,000 years.

FNAL Newsroom
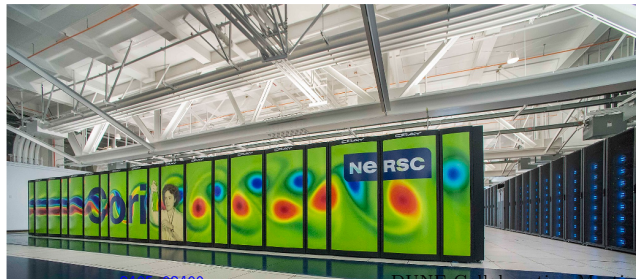
- **Completed productions:**
  - LBL FD production, reconstruction step (24M events)
  - Atmospheric FD-HD reconstruction step (15M events)
  - PD-HD
  - PD-SP 2 GeV
  - ND: 2x2+Minerva samples produced

From Kenneth Herner's talk on Monday

Monte-Carlo estimates of statistical significances
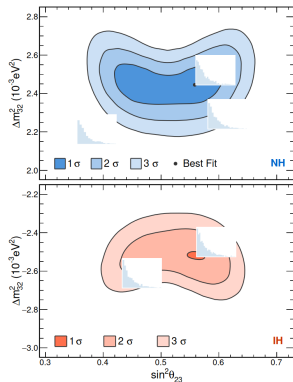
Wilks' theorem is often wrong

At each point in parameter space, simulate the experiment many times

"many" means $\gg 1/p$ for a desired p-value

This is sometimes called Feldman-Cousins

G. Feldman, R. Cousins `physics/9711021`
This isn't actually what was novel in the FC paper

Study found most of
the time was spent
computing probabilities

NOvA is a $\sim 3\sigma$ experiment,
but DUNE will be a $\gtrsim 5\sigma$ experiment!

Start at the end

What is needed for experiments?

1. All 9 channels ($\nu_\alpha \to \nu_\beta$)
   - ▶ DUNE will certainly do ν_τ appearance

     See e.g. P. Machado, H. Schulz, J. Turner 2007.00015
   - ▶ $\nu_\tau \to \nu_\beta$ channels are not needed, but come from free from unitarity
2. Different energies, baselines, and densities
3. ν and $\bar\nu$
4. NO and IO
5. Oscillation parameters are mostly known
   - ▶ Don't need to consider e.g. $\Delta m_{21}^2 > |\Delta m_{31}^2|$ or $\theta_{23} \sim 10°$

How to achieve speed

1. Avoid costly operations
   - ▶ `sin`, `cos` (and inverse functions) are very slow
   - ▶ `sqrt` is quite slow, but not as bad as trigs
   - ▶ Division is slower than multiplication ($0.2x$ may be faster than $x/5$)
2. Reduce repeated calculations
   - ▶ Compute $\frac{L}{4E}$ in the correct units once
   - ▶ Compute each of the three $\sin \frac{\Delta m_{ij}^2 L}{4E}$ once

Optimal structure of the probability

1. Amplitude requires four trig functions of kinematic variables $(\Delta m_{ij}^2 L/4E)$ ✗
2. Writing the probabilities out requires three trig functions ✓
3. Disappearance structure is straightforward:

$$P_{\alpha\alpha} = 1 - 4 \sum_{i>j} |V_{\alpha i}|^2 |V_{\alpha j}|^2 \sin^2 \frac{\Delta\lambda_{ij} L}{4E}$$

H in matter has eigenvalues λ_i and eigenvectors $V_{\alpha i}$

Optimal structure of the probability

4. Appearance structure:

$$P_{\mu e}^{CPC} = 2 \sum_{i>j} (|V_{\tau k}|^2 - |V_{\mu i}|^2 |V_{ej}|^2 - |V_{\mu j}|^2 |V_{ei}|^2) \sin^2 \frac{\Delta\lambda_{ij}L}{4E}$$

> Fun fact:
> $$2\Re(V_{\alpha i} V_{\beta j}^* V_{\alpha j}^* V_{\beta i})$$
> $$= |V_{\alpha k}|^2 |V_{\beta k}|^2 - |V_{\alpha i}|^2 |V_{\beta i}|^2 - |V_{\alpha j}|^2 |V_{\beta j}|^2$$
> $$= |V_{\gamma k}|^2 - |V_{\alpha i}|^2 |V_{\beta j}|^2 - |V_{\alpha j}|^2 |V_{\beta i}|^2$$

$$P_{\mu e}^{CPV} = -8J \frac{\Delta m_{21}^2 \Delta m_{31}^2 \Delta m_{32}^2}{\Delta\lambda_{21} \Delta\lambda_{31} \Delta\lambda_{32}} \sin \frac{\Delta\lambda_{21}L}{4E} \sin \frac{\Delta\lambda_{31}L}{4E} \sin \frac{\Delta\lambda_{32}L}{4E}$$

Leverages NHS identity:
V. Naumov IJMP 1992
P. Harrison, W. Scott hep-ph/9912435

Account for matter

1. Need the eigenvalues λ_i

2. For eigenvectors, naively need $\Re(V_{\alpha i} V_{\beta j}^* V_{\alpha j}^* V_{\beta i})$

3. Given our form, need only the $|V_{\alpha i}|^2$ and J
   - ▶ Don't need any phase information of the eigenvectors!

   Leverages PBD, S. Parke, X. Zhang 1907.02534

4. Can compute the $|V_{\alpha i}|^2$ from the λ_i and submatrix eigenvalues (requires only a square root) using Eigenvector-Eigenvalue Identity

$$|V_{\alpha i}|^2 = \frac{\prod_{k=1}^{n-1}(\lambda_i - \xi_k^\alpha)}{\prod_{k=1; k \neq i}^{n}(\lambda_i - \lambda_k)}$$

See e.g. PBD, S. Parke, T. Tao, X. Zhang 1908.03795
Can actually avoid the $\sqrt{}$ in practice

Eigenvalues are hard

The eigenvalues in matter λ_i depend on S:

$$S = \cos\left\{ \frac{1}{3} \cos^{-1} \left[\frac{2A^3 - 9AB + 27C}{2\left(A^2 - 3B\right)^{3/2}} \right] \right\}$$

where

$$A = \sum \lambda_i = \Delta m_{21}^2 + \Delta m_{31}^2 + a$$

$$B = \sum_{i>j} \lambda_i \lambda_j = \Delta m_{21}^2 \Delta m_{31}^2 + a[\Delta m_{21}^2(1 - |U_{e2}|^2) + \Delta m_{31}^2(1 - |U_{e3}|^2)]$$

$$C = \prod \lambda_i = a \Delta m_{21}^2 \Delta m_{31}^2 |U_{e1}|^2$$

Approximate eigenvalues

1. Instead, approximate one eigenvalue
   - ▶ λ_3 is best because is never parametrically small and easy to approximate
2. From DMP:

$$\lambda_3 \approx \Delta m_{31}^2 + \frac{1}{2}\Delta m_{ee}^2 \left(x - 1 + \sqrt{(1-x)^2 + 4xs_{13}^2} \right)$$

$$x \equiv \frac{a}{\Delta m_{ee}^2} \qquad \Delta m_{ee}^2 \equiv \Delta m_{31}^2 - s_{12}^2 \Delta m_{21}^2$$

H. Minakata, S. Parke 1505.01826
PBD, H. Minakata, S. Parke 1604.08167
H. Nunokawa, S. Parke, R. Funchal hep-ph/0503283

3. Get other two eigenvalues by picking two of A, B, C conditions

Requires one more $\sqrt{\ \ }$

The approximation

- This is the only approximation used in the entire approach
- In vacuum the approximation returns to the correct value

  Many approximations in the literature are not correct in vacuum limit
  See G. Barenboim, PBD, S. Parke, C. Ternes 1902.00517

- In fact can iteratively improve λ_3 with rapid convergence via Newton's method:

$$\lambda_3 \to \lambda_3 - \frac{X(\lambda_3)}{X'(\lambda_3)}$$

$$X(\lambda) = \lambda^3 - A\lambda^2 + B\lambda - C = 0$$

  - Precision improvement starts at 10^{-5} for the first step
  - The improvement is quadratic thereafter
- One line of code, just loop as many times as desired

All 9 channels

Given P_{ee}, $P_{\mu\mu}$, $P_{\mu e}^{CPC}$, and $P_{\mu e}^{CPV}$:

	$P_{\alpha e}$	$P_{\alpha\mu}$	$P_{\alpha\tau}$
$P_{e\beta}$	P_{ee}	$P_{\mu e}^{CPC} - P_{\mu e}^{CPV}$	$1 - P_{ee} - P_{\mu e}^{CPC} + P_{\mu e}^{CPV}$
$P_{\mu\beta}$	$P_{\mu e}^{CPC} + P_{\mu e}^{CPV}$	$P_{\mu\mu}$	$1 - P_{\mu\mu} - P_{\mu e}^{CPC} - P_{\mu e}^{CPV}$
$P_{\tau\beta}$	$1 - P_{ee} - P_{\mu e}^{CPC} - P_{\mu e}^{CPV}$	$1 - P_{\mu\mu} - P_{\mu e}^{CPC} + P_{\mu e}^{CPV}$	$-1 + P_{ee} + P_{\mu\mu} + 2P_{\mu e}^{CPC}$

Total approach

1. Inputs: 6 oscillation parameters, experimental details (L, E, ρ, Y_e)
2. Calculate λ_3 approximately
   - Iteratively improve with Newton's method, if desired
3. Calculate the $|V_{\alpha i}|^2$'s with the Eigenvector-Eigenvalue Identity
4. Calculate the sines of the kinematic terms
5. Calculate the CPV term with the NHS identity
6. Calculate key probabilities: P_{ee}, $P_{\mu\mu}$, and $P_{\mu e}$
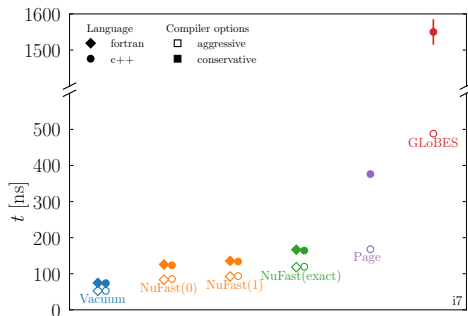7. Calculate remaining probabilities

Precision

Is this approximation okay?
DUNE requires $\lesssim 1\%$ level precision
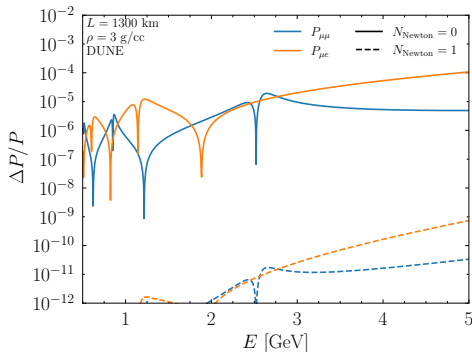
Slightly better for HK
$\sim 10^{-10}$ for JUNO
See backups

Speed

Is this algorithm fast?
How does it compare?

- "Conservative" = default, `-O0`
- "Aggressive" = `-Ofast` and `-ffast-math`
- Some variation expected due to architecture

See also J. Page 2309.06900
and P. Huber, et al. hep-ph/0701187

NuFast: the code

- Code is on github: github.com/PeterDenton/NuFast
- Implementations in c++ and f90
- Easy to use and there are comments!

```cpp
// NuFast.cpp
// Calculate the probabilities:
Probability_Matter_LBL(s12sq, s13sq, s23sq, delta,
    Dmsq21, Dmsq31, L, E, rho, Ye, N_Newton,
    &probs_returned);
// Print out the probabilities to terminal
for (int alpha = 0; alpha < 3; alpha++)
{
    for (int beta = 0; beta < 3; beta++)
    {
        printf("%d %d %g\n", alpha, beta,
            probs_returned[alpha][beta]);
    } // beta, 3
} // alpha, 3
```

- $\bar{\nu}$: $E < 0$; IO: $\Delta m_{31}^2 < 0$
- Folder called Benchmarks to make the plots in the paper

Summary

- Computing neutrino oscillations fast is important

- New algorithm contains several innovations:
  - Approximate λ_3 and rapid convergence, if desired
  - Get all eigenvalues from λ_3 via A, B, C
  - Get CPV part from NHS identity
  - Structure CPC parts to not require phase information
  - Get $|V_{\alpha i}|^2$ from Eigenvalue-Eigenvector Identity
  - Careful implementation in code to reduce expensive and repeated computations

- People are already trying out the code!

- Atmospherics and nighttime solar in the works!

Backups

References
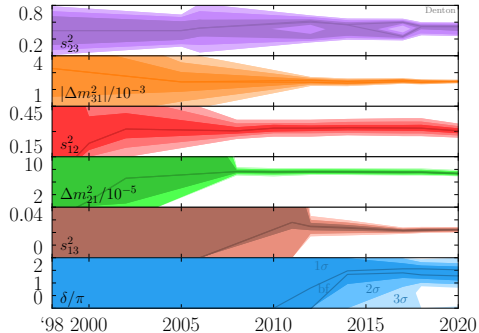
SK `hep-ex/9807003`

M. Gonzalez-Garcia, et al. `hep-ph/0009350`

M. Maltoni, et al. `hep-ph/0207227`

SK `hep-ex/0501064`

SK `hep-ex/0604011`

T. Schwetz, M. Tortola, J. Valle `0808.2016`

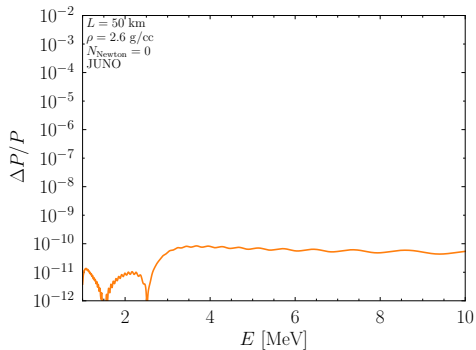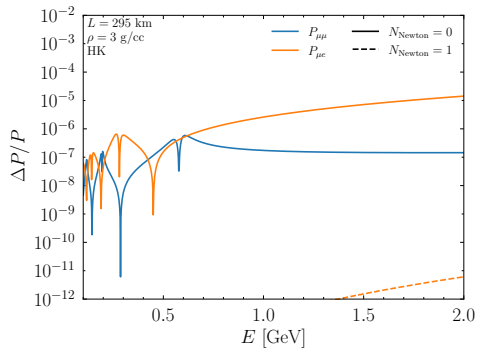M. Gonzalez-Garcia, M. Maltoni, J. Salvado `1001.4524`

T2K `1106.2822`

D. Forero, M. Tortola, J. Valle `1205.4018`
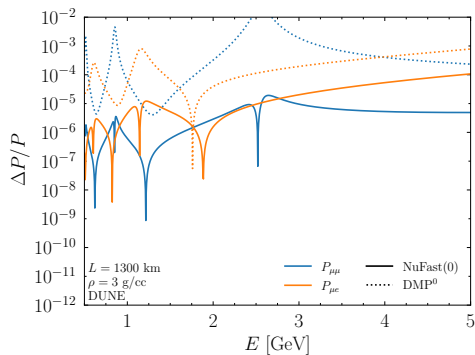
D. Forero, M. Tortola, J. Valle `1405.7540`

P. de Salas, et al. `1708.01186`

F. Capozzi et al. `2003.08511`

Precision

Comparison to DMP

DMP: PBD, H. Minakata, S. Parke 1604.08167