Gemini++ Updates: v.3.3

Arjun Ray

Undergraduate Research Assistant II
raya@frib.msu.edu

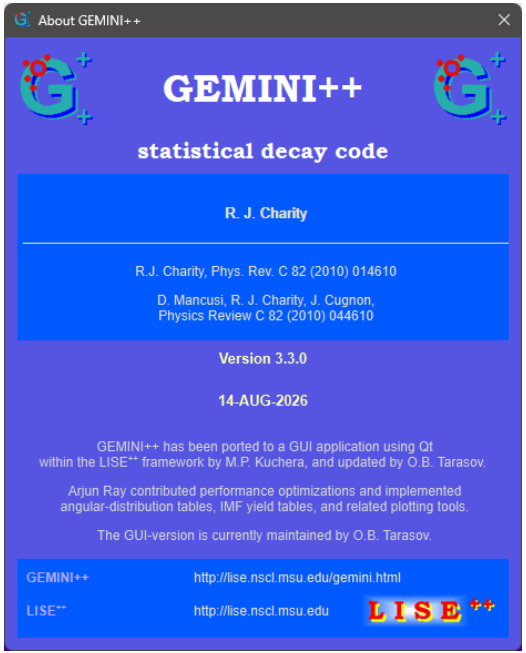
08/13/2026

(updated 08/15/2026 by O.B.Tarasov)

This material is based upon work supported by the U.S. Department of Energy, Office of Science, Office of Nuclear Physics and used resources of the Facility for Rare Isotope Beams (FRIB) Operations, which is a DOE Office of Science User Facility under Award Number DE-SC0023633, and by the US National Science Foundation under Grants No. PHY-20-12040 and 23-10078 “Windows on the Universe: Open Quantum Systems in Atomic Nuclei at FRIB”.

GEMINI Update Details

- **3.1.09–3.1.11 — Arjun:**
  - introduced project-wide optimization,
  - residual angular-distribution tables,
  - plotting windows,
  - Gaussian fits for 1D plots.
- **3.2.0–3.2.7 — Arjun:**
  - consolidated the 2026 GUI development,
  - including gamma angular-distribution tables,
  - center-of-mass spectra,
  - IMF angular tables and plots,
  - Boltzmann overlays,
  - yield-table output updates.
- **3.2.8–3.3.0 — Oleg:**
  - covers the Codex-assisted development changes, including startup defaults,
  - **migration to double precision,**
  - **MinGW/MSVC warning cleanup**,
  - result-window scrollbar support,
  - About-dialog attribution,
  - synchronized version metadata.



Optimization

Gemini profiling to search for bottleneck points

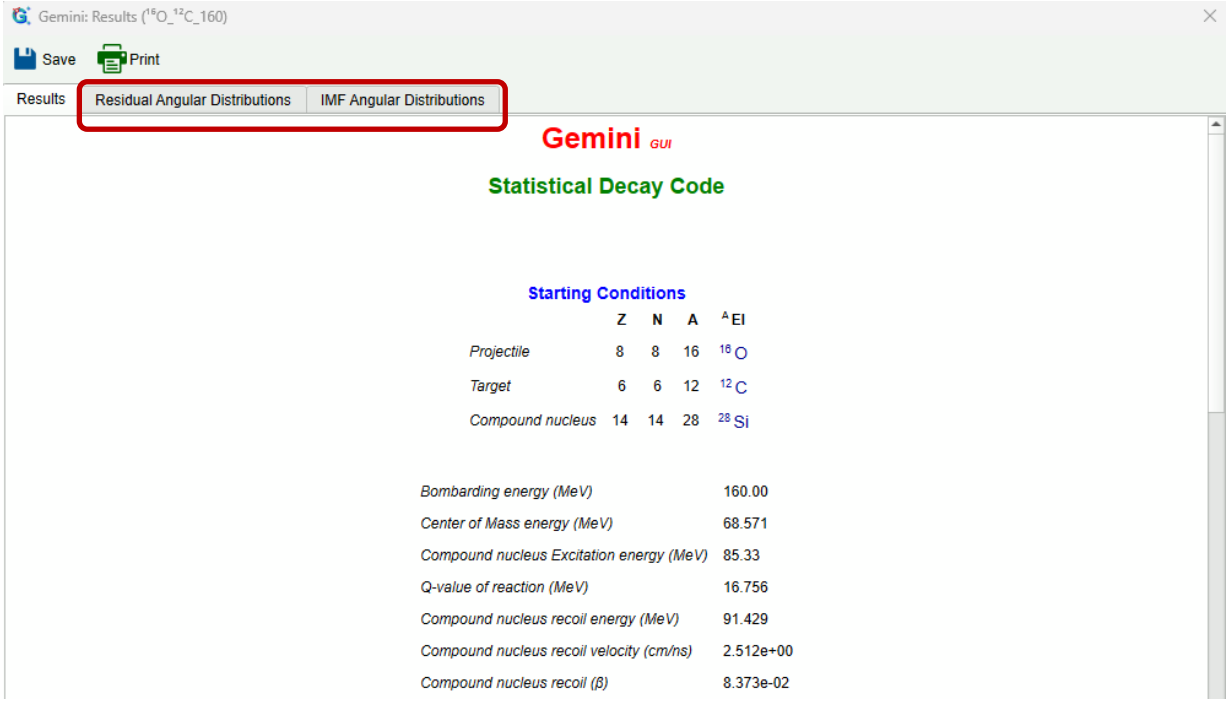
BOTTLENECKS ADDRESSED AND CHANGES MADE

No.	Location	Bottleneck	Change Made
1	Nucleus.cpp(CNucleus::getSumTl())	storeSub.push_back() caused heavy reallocations and copies	Reused a thread_local buffer, reserved capacity using (IMax - IMin + 1), and replaced push_back with emplace_back
2	Nucleus.cpp(CNucleus::getSumTl())	Repeated calls to getTl(L, ek*scale, temp) for same inputs	Added a small local cache to store computed getTl values and reused them
3	Nucleus.cpp (decay-product generation)	Heavy new/delete churn for CNucleus objects during decay	Introduced a recycle pool (acquire() / release()) to reuse CNucleus objects instead of allocating each time
4	Nucleus.cpp (decay-product generation)	Deep recursive decay traversal causing function call overhead	Replaced recursive decay traversal with an explicit stack-based loop
5	Project-wide	Extensive use of pow() in performance-critical code paths	Replaced pow() with equivalent direct arithmetic (e.g., x*x)
6	Nucleus.cpp (constructors & initialization)	Duplicated constructor logic and repeated setup code	Centralized all setup logic into initializeDefaults() and initialize(...)
7	Nucleus.cpp (decay-product handling)	Frequent temporary vector allocations across events	Reused decay-product vectors across events instead of recreating them
8	gm_mwExecFusion.cpp,gm_mwExecCompound.cpp	Throttled progress + cancel handling	Updated progress bar only every 10 iterations to reduce UI overhead

Gemini optimization at bottleneck points and more to result in Gemini being ~1.6x faster

TIMES			
	old time(s)	new time(s)	Time ratio (%)
Compound Nucleus Decay			
Decay events.= 1000	7.07	4..30	39%
No. Of decay events.= 3000	21.51	12.61	41%
A=195,J=50,Events= 3000	23.63	15.43	35%
Average speed increase: 38.4% time spent, 1.6× faster			
Fusion Reaction			
Default	10.23	5.67	45%
Events= 1000	20.5	11.38	44%
A=27,100 MeV	7.46	3.94	47%
A=27,200 MeV	11.28	7.06	37%
Average speed increase: 43.9% time spent, 1.8× faster			

Tabs for Residual and Angular distributions



IMF production table

Gemini: Results (^{16}O_^{12}C_160)

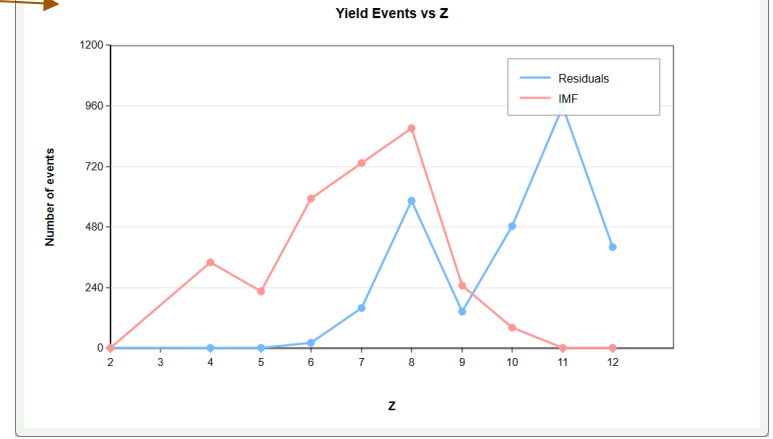
Save Print

Results | Residual Angular Distributions | IMF Angular Distributions

Yields of Residual Nuclei and IMF Particles (plot)

	Residual Nuclei					IMF Particles				
Z	Name	Events	Percent	x-section (mb)	err(mb)	Name	Events	Percent	x-section (mb)	err(mb)
12	^{26}Mg	2	0.1%	0.3749	0.2651	-	-	-	-	-
	^{25}Mg	70	2.5%	13.12	1.568	-	-	-	-	-
	^{24}Mg	279	10.1%	52.29	3.131	-	-	-	-	-
	^{23}Mg	50	1.8%	9.372	1.325	-	-	-	-	-
11	^{24}Na	145	5.3%	27.18	2.257	-	-	-	-	-
	^{23}Na	607	22.0%	113.8	4.618	-	-	-	-	-
	^{22}Na	132	4.8%	24.74	2.153	-	-	-	-	-
	^{21}Na	74	2.7%	13.87	1.612	-	-	-	-	-
10	^{23}Ne	5	0.2%	0.9372	0.4191	^{22}Ne	1	0.0%	0.07487	0.07487
	^{22}Ne	66	2.4%	12.37	1.523	^{21}Ne	18	0.6%	1.348	0.3176
	^{21}Ne	403	14.6%	75.54	3.763	^{20}Ne	39	1.3%	2.92	0.4675
	^{19}Ne	9	0.3%	1.687	0.5623	^{19}Ne	20	0.6%	1.497	0.3348
	-	-	-	-	-	^{18}Ne	5	0.2%	0.3743	0.1674
9	^{21}F	4	0.1%	0.7497	0.3749	^{20}F	22	0.7%	1.647	0.3512
	^{20}F	29	1.1%	5.436	1.009	^{19}F	49	1.6%	3.669	0.5241
	^{19}F	62	2.3%	11.62	1.476	^{18}F	109	3.5%	8.161	0.7816
	^{18}F	49	1.8%	9.184	1.312	^{17}F	67	2.2%	5.016	0.6128
8	^{18}O	28	1.0%	5.248	0.9918	^{18}O	48	1.6%	3.594	0.5187
	^{17}O	342	12.4%	64.1	3.466	^{17}O	263	8.5%	19.69	1.214
	^{16}O	202	7.3%	37.86	2.664	^{16}O	416	13.4%	31.14	1.527
	^{15}O	12	0.4%	2.249	0.6493	^{15}O	140	4.5%	10.48	0.8858

Gemini: Yield events vs Z



Angular Distribution Table

– Implemented PACE style angular distribution tables for Gemini

7. Energy and angular distribution of residual nucleus Z = 11 and N = 12 (23 Na)

Residual velocity/c Vz = 8.29e-02 (sig = 4.77e-03) rms Vxy = 9.30e-03

Energy Range	Angular range (deg)																	
(MeV)	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
54.0 - 60.0				2														
60.0 - 66.0		1			1	1	1				1		1					
66.0 - 72.0		2	4	3	3	1	1	1					1				1	
72.0 - 78.0			1		1	2		2	1									
78.0 - 84.0			2	2	1	3	1		3	1	1	1						
84.0 - 90.0	1	2	3	1						1	1							
Above 174																		
Total	1	5	10	8	6	7	3	3	4	2	3	1	2				1	
dSig/dOmeg	1.1e+03	1.8e+03	2.2e+03	1.2e+03	7.2e+02	6.9e+02	2.5e+02	2.2e+02	2.5e+02	1.1e+02	1.5e+02	47	87	0.00	0.00	0.00	33	0.00
0 - 0																		
0 - 90	1	5	10	8	6	7	3	3	4	2	3	1	2				1	
90 - 180																		
Above 180																		

Plot E vs Theta Plot N vs Theta Plot N vs E Plot dCS/dTheta vs Theta

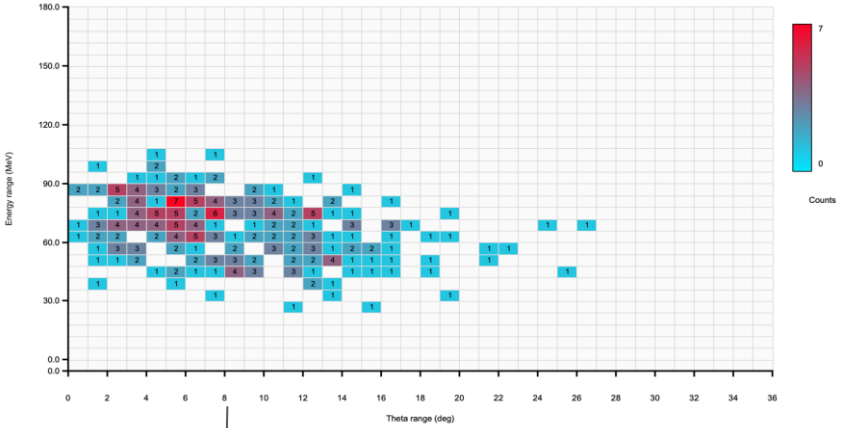
4 types of plots for each table

Plots

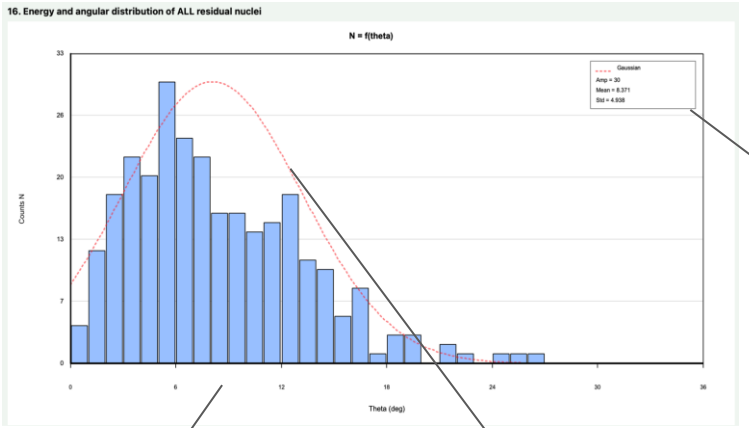
Plot All: E vs Theta Plot All: N vs Theta Plot All: N vs E Plot All: dCS/dTheta vs Theta

*** Spin alignment perpendicular to recoil axis - standard compound nucleus angular distribution

"Plot All" option for each type of plot to display plots for all tables



2D E vs Theta Plot example



1D plot example

Gaussian distribution for all 1D plots with N(count)