A global analysis of decaying cosmological ALPs

Tomás Gonzalo

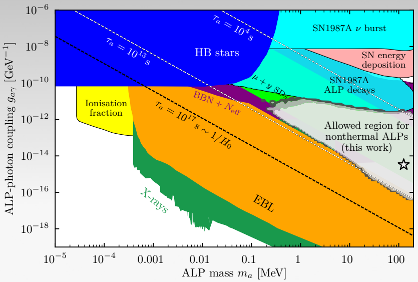
Karlsruhe Institute for Technology

Light Dark World 2023, 21 September 2023

[P.Stoecker, TG, S. Hoof, F, Kahlhoefer et al, JCAP 12 (2022) 027]

Motivation

- Axion-like-particles are well-motivated
  - → They appear in UV completions (e.g. string compactifications, etc)
- ALPs can be a sucessful DM candidate
  - → Produced by realignment mechanism ($m_a <$ keV)
  - → Thermally produced and decay ($m_a \lesssim$ keV or $m_a \gtrsim$ GeV)
- MeV ALPs can be produced out of equilibrium
  - → decay between BBN and CMB 10^6 s $< \tau < 10^{13}$ s



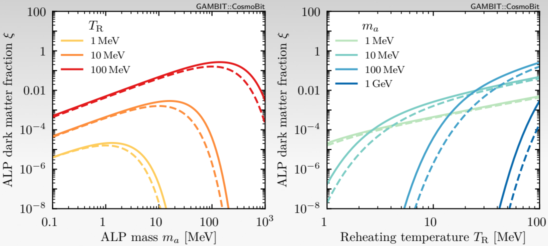
Cosmological ALPs

- ALPs with effective coupling to photons $\mathcal{L} = \frac{g_{a\gamma}}{4} a F^{\mu\nu} \tilde{F}_{\mu\nu}$
- ALP lifetime

$$\Gamma_a = \frac{g_{a\gamma}^2 m_a^3}{64\pi} = \frac{1}{1.32 \times 10^8 \text{ s}} \left(\frac{g_{a\gamma}}{10^{-12} \text{ GeV}^{-1}} \right)^2 \left(\frac{m_a}{10 \text{ MeV}} \right)^3$$

- ALP abundance $\xi = m_a n_{a,0} / \rho_{\text{DM},0}$

$$\xi > 7.82 \times 10^{-5} \frac{m_a g_{a\gamma}^2 \bar{m}_P T_R s_0}{\rho_{\text{DM},0}} = 0.022 \left(\frac{m_a}{1 \text{ MeV}} \right) \left(\frac{T_R}{5 \text{ MeV}} \right) \left(\frac{g_{a\gamma}}{10^{-10} \text{ GeV}^{-1}} \right)^2$$

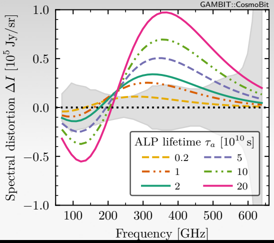


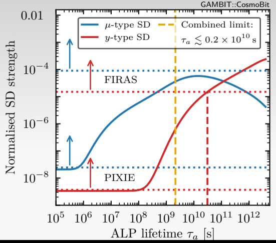
CMB constraints

- Energy injected into pre-recombination plasma from ALP decays
  - → Inject non-thermalized photons cause anisotropies in the CMB

$$\frac{dT_\gamma}{dt} = \frac{15}{4\pi^2}\frac{m_a n_a(t) T_\gamma^{-3}}{\tau_a} - H(t) T_\gamma$$

  - → Variations on $N_{\rm eff} = N_{\rm eff,BBN}(T_\nu/T_\gamma)^4(11/4)^{4/3}$
  - → Variations on $\eta_b = \eta_{b,\rm BBN}(T_{\gamma,\rm BBN}/T_\gamma)^3(a_0/a_{\rm BBN})^3$
- Spectral distortions on the CMB caused by ALP decays





Cosmological ALPs

Primordial nucleosynthesis

- ALPs decay after BBN $\rightsquigarrow$ vanilla nucleosynthesis (ΛCDM)
- Injected γ spectrum causes photodisintegration

$$\frac{\mathrm{d}Y_N}{\mathrm{d}t} = \sum_j \int_0^\infty \mathrm{d}E f_\gamma \left(Y_j\, \sigma_{j\gamma\to N} - Y_N\, \sigma_{N\gamma\to j} \right) ,$$

- Measurements of relevant abundances

$$\begin{aligned} \mathrm{D/H} &= (2.547 \pm 0.025) \times 10^{-5} \\ {}^3\mathrm{He/D} &< 1.03\ (95\%\ \mathrm{CL}) \\ Y_p &= 0.245 \pm 0.003 \\ {}^7\mathrm{Li/H} &= (1.6 \pm 0.3) \times 10^{-10} \end{aligned}$$

- Likelihood computation

$$-2 \ln \mathcal{L} = f(\mathcal{P}, \mathcal{O})\, \mathcal{C}^{-1}\, f(\mathcal{P}, \mathcal{O})^T + (2\pi)^n \det(\mathcal{C}),$$

Cosmological ALPs

Astrophysical constraints

- SN1987A
  → Modified photon fluence from ALP decays, with mean distance

$$\ell = \frac{64\pi}{g_{a\gamma}^2 m_a^3}\sqrt{\frac{E_a^2}{m_a^2} - 1}$$

- $R_{\rm GC} = N_{\rm HB}/N_{\rm RGB}$
  → ALPs carry out energy from stellar interiors affecting their evolution

$$R_{\rm GC} \approx 0.022 - 0.443(1 + 0.965|g_{a\gamma}|)^{1/2} + 7.331 Y_{\rm GC}$$

Other

- Baryon Accoustic Oscillations (BOSS DR12)
- Type IA supernovea (Pantheon)
- Neutron lifetime (bottle measurement)
- Nuisance parameters

Global fit

Karlsruher Institut für Technologie

Model parameter		Scan range	
ALP mass	m_a	[0.001, 200]	MeV
ALP lifetime	τ_a	$[10^4, 10^{13}]$	s
ALP abundance	ξ	$[10^{-12}, 10^2]$	
Baryon abundance	ω_b	[0.020, 0.024]	
Dark matter abundance	$\omega_{\rm DM}$	[0.10, 0.13]	
Hubble constant	H_0	[62, 74]	km $\mathrm{s}^{-1}\mathrm{Mpc}^{-1}$
Redshift of reionisation	$z_{\rm reio}$	[4.5, 9.5]	
Primordial curvature	$\ln(10^{10} A_s)$	[2.9, 3.2]	
Scalar spectral index	n_s	[0.9, 1.1]	
Neutron lifetime	τ_n	[875, 895]	s
Planck nuisance parameter	$A_{\rm Planck}$	[0.9, 1.1]	
Pantheon nuisance parameter	M	$[-20, -18]$	

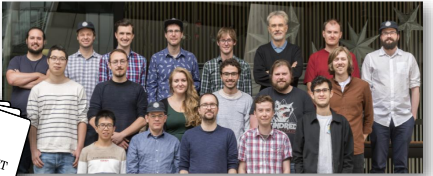
Global Fit

Karlsruher Institut für Technologie

GAMBIT: The Global And Modular BSM Inference Tool

gambit.hepforge.org   github.com/GambitBSM   EPJC 77 (2017) 784   arXiv:1705.07908

- Extensive model database, beyond SUSY
- Fast definition of new datasets, theories
- Extensive observable/data libraries
- Plug&play scanning/physics/likelihood packages
- Various statistical options (frequentist /Bayesian)
- Fast LHC likelihood calculator
- Massively parallel
- Fully open-source



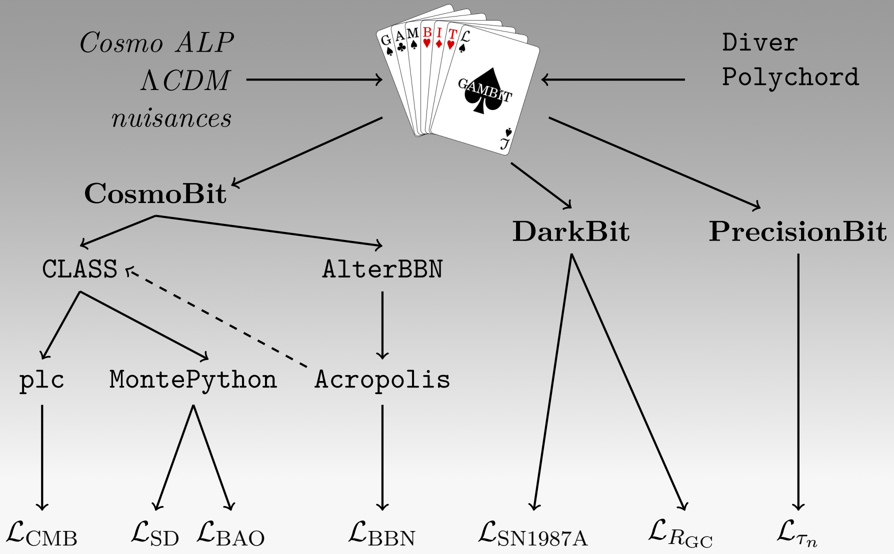
Members of: ATLAS, Belle-II, CLiC, CMS, CTA, Fermi-LAT, DARWIN, IceCube, LHCb, SHiP, XENON

Authors of: BubbleProfiler, Capt'n General, Contur, DarkAges, DarkSUSY, DDCalc, DirectDM, Diver, EasyScanHEP, ExoCLASS, FlexibleSUSY, gamLike, GM2Calc, HEPLike, IsaTools, MARTY, nuLike, PhaseTracer, PolyChord, Rivet, SOFTSUSY, Superlso, SUSY-AI, xsec, Vevacious, WIMPSim

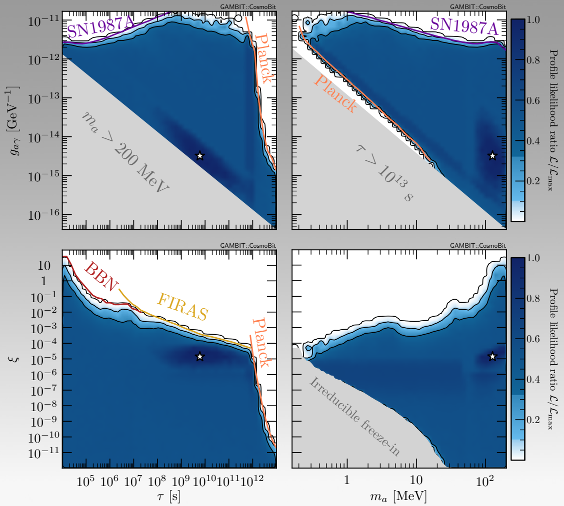
Recent collaborators: V Ananyev, P Athron, N Avis-Kozar, C Balázs, A Beniwal, S Bloor, LL Braseth, T Bringmann, A Buckley, J Butterworth, J-E Camargo-Molina, C Chang, M Chrzaszcz, J Conrad, J Cornell, M Danninger, J Edsjö, T Emken, A Fowlie, T Gonzalo, W Handley, J Harz, S Hoof, F Kahlhoefer, A Kvellestad, M Lecroq, P Jackson, D Jacob, C Lin, FN Mahmoudi, G Martinez, H Pacey, MT Prim, T Procter, F Rajec, A Raklev, JJ Renk, R Ruiz, A Scaffidi, P Scott, N Serra, P Stöcker, W. Su, J Van den Abeele, A Vincent, C Weniger, A Woodcock, M White, Y Zhang ++

80+ participants in many experiments and numerous major theory codes

Global Fit


Cosmo ALP
ΛCDM
nuisances
Diver
Polychord
GAMBIT
CosmoBit
DarkBit
PrecisionBit
CLASS
AlterBBN
plc
MontePython
Acropolis
$\mathcal{L}_{\mathrm{CMB}}$
$\mathcal{L}_{\mathrm{SD}}$
$\mathcal{L}_{\mathrm{BAO}}$
$\mathcal{L}_{\mathrm{BBN}}$
$\mathcal{L}_{\mathrm{SN1987A}}$
$\mathcal{L}_{R_{\mathrm{GC}}}$
$\mathcal{L}_{\tau_n}$

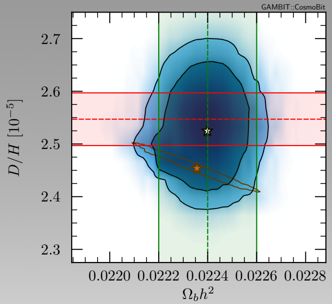
Results

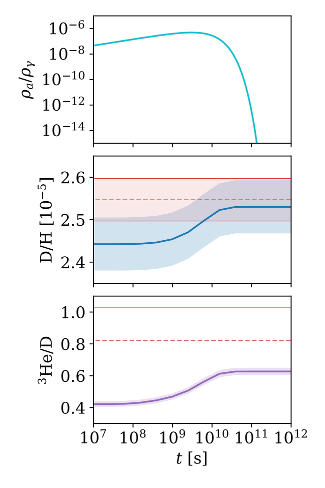


- Frequentist results
- Independent pars $\{m_a, \tau_a, \xi\}$
- Small abundance $\xi \ll 1$
- Mass lower bound $m_a > 300$ keV
- No effect from $N_{\rm eff}$, η_b or R parameter
- Mostly flat $\Delta\mathcal{L}$
- Small excess at

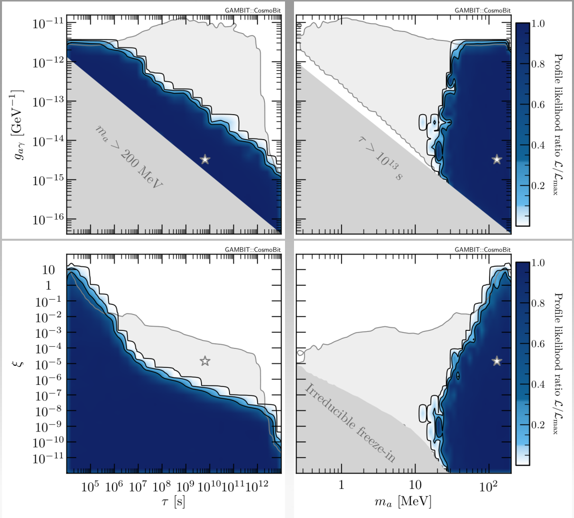
$$m_a = 126.1 \text{ MeV}$$
$$\tau_a = 6.04 \times 10^9 \text{ s}$$
$$\xi = 4.18 \times 10^{-5}$$

Results





- In ΛCDM there is a correlation between $\Omega_b h^2$ and D/H
- No correlation in ALP model because photodisintegration
- Improved fit to observations
- ΛCDM within 1σ of ALP model

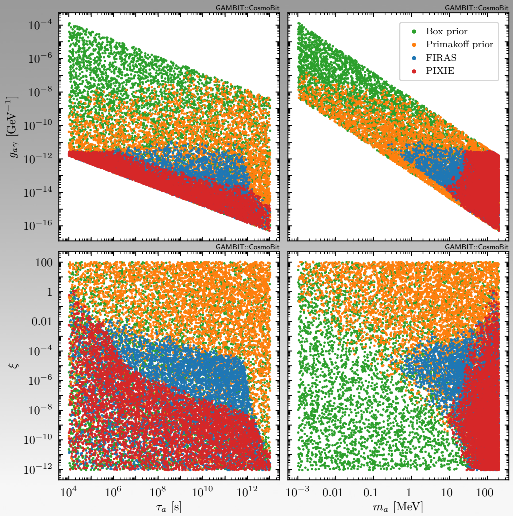


- Mock PIXIE $\Delta\mathcal{L}$
- Assumes null result
- Best fit region completely explored
- Would mean $[D/H]$ discrepancy cannot be explained with ALPs

^{7}Li problem

- Prediction does not match observation
- Cannot be explained by ALPs because of SD

Results



- Bayesian results
- Two priors: box and Primakoff
- Bayesian evidence

$$\begin{aligned}
\ln \mathcal{Z} &= \quad \langle \ln \mathcal{L} \rangle_{\mathcal{P}} & - \mathcal{D}_{\rm KL} \\
\ln \mathcal{Z}^{\rm box}_{\rm ALPs} &= -1012.27 & - 26.82 \\
\ln \mathcal{Z}^{\rm Primakoff}_{\rm ALPs} &= -1012.27 & - 25.85 \\
\ln \mathcal{Z}_{\Lambda{\rm CDM}} &= -1012.38 & - 24.72
\end{aligned}$$

- Penalisation comes from additional parameters
- Compression smaller for Primakoff prior

Summary and Conclusions

- Summary
  - → Explored ALPs in range keV $< m_a <$ 100 MeV
  - → ALPs decay between BBN and recombination 10^4 s $< \tau < 10^{13}$s
  - → Non thermal abundance relaxes cosmological constraints $\xi < 10^2$
- Conclusions
  - → Confirmed that cosmological constraints cannot constrain window
  - → ALP decays cause photodisintegration of light elements
  - → Fits better D/H abundance ($< 1\sigma$ excess in frequentist results)
  - → Bayesian results show preference for ΛCDM
  - → Not able to fit ^7Li result (mostly due to SD)
  - → COBE/FIRAS results (20 year old) very constraining
  - → Future mission PIXIE promise much stronger constraints and discovery potential
- Outlook
  - → ALP-electron and ALP-hadron couplings unexplored
  - → Possible solution to ^7Li problem

Backup

Bayesian statistics

- Bayesian evidence (marginal likelihood)

$$\mathcal{Z} = \int \mathcal{L}\pi d\theta = \langle \mathcal{L} \rangle_\pi$$

- Posterior distribution

$$\mathcal{P} = \frac{\mathcal{L}\pi}{\mathcal{Z}}$$

- Kullback–Leibler divergence (Occam's penalty)

$$\mathcal{D}_{\text{KL}} = \int \mathcal{P} \ln \frac{\mathcal{P}}{\pi} d\theta = \left\langle \ln \frac{\mathcal{P}}{\pi} \right\rangle_{\mathcal{P}}$$

Photodisintegration

Reaction	E_{th} [MeV]
$D + \gamma \to n + p$	2.22
$^3H + \gamma \to D + n$	6.26
$^3H + \gamma \to 2n + p$	8.48
$^3He + \gamma \to D + p$	5.49
$^3He + \gamma \to n + 2p$	7.12
$^4He + \gamma \to {}^3H + p$	19.81
$^4He + \gamma \to {}^3He + n$	20.58
$^4He + \gamma \to 2D$	23.84
$^4He + \gamma \to D + n + p$	26.07

Reaction	E_{th} [MeV]
$^6Li + \gamma \to {}^4He + n + p$	3.70
$^6Li + \gamma \to {}^3H + {}^3He$	15.79
$^7Li + \gamma \to {}^3H + {}^4He$	2.47
$^7Li + \gamma \to {}^6Li + n$	7.25
$^7Li + \gamma \to {}^4He + 2n + p$	10.95
$^7Be + \gamma \to {}^3He + {}^4He$	1.59
$^7Be + \gamma \to {}^6Li + p$	5.61
$^7Be + \gamma \to {}^4He + n + 2p$	9.30

Global fits

- Combine all constraints into a **composite likelihood**

$$\mathcal{L} = \mathcal{L}_{Collider}\mathcal{L}_{Higgs}\mathcal{L}_{DM}\mathcal{L}_{Flavour}\ldots$$

- Perform an extensive **parameter scan**

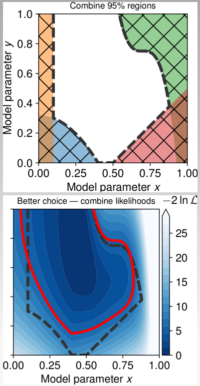
→ Old-school sampling methods (random, grid) are inefficient

→ Harder to make statement about statistics

→ Need **smart sampling strategies** (differential, nested, genetic,...)

→ **Rigorous** statistical interpretation (frequentist/Bayesian)

  - Goodness-of-fit
  - Parameter estimation
  - Model comparison



[arXiv:2012.09874 [hep-ph]]

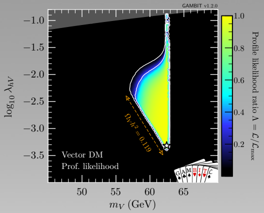
Global Fit

- Physics Modules
  - → **ColliderBit**: collider searches [Eur.Phys.J. C77 (2017) no.11, 795]
  - → **DarkBit**: relic density, dd,... [Eur.Phys.J. C77 (2017) no.12, 831]
  - → **FlavBit**: flavour observables [Eur.Phys.J. C77 (2017) no.11, 786]
  - → **SpecBit**: spectra, RGE running [Eur.Phys.J. C78 (2018) no.1, 22]
  - → **DecayBit**: decay widths [Eur.Phys.J. C78 (2018) no.1, 22]
  - → **PrecisionBit**: precision tests [Eur.Phys.J. C78 (2018) no.1, 22]
  - → **NeutrinoBit**: neutrino likelihoods [Eur.Phys.J.C 80 (2020) no.6, 569]
  - → **CosmoBit**: cosmological constraints [JCAP 02 (2021) 022]
- **ScannerBit** : stats and sampling [Eur.Phys.J. C77 (2017) no.11, 761]
  - → Diver, GreAT, Multinest, Polychord, ...
- **Models**: hierarchical model database
- **Core** : dependency resolution [Eur.Phys.J. C78 (2018) no.2, 98]
- **Backends** : External tools to calculate observables
- **GUM**: Autogeneration of code [S. Bloor, TG, P. Scott et. al., soon]

Global Fit

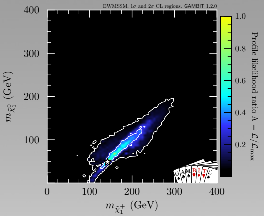
Higgs-portal DM

[Eur.Phys.J.C 79 (2019) 1, 38]



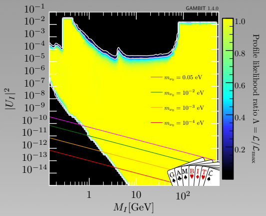
MSSM-EW

[Eur.Phys.J.C 79 (2019) 5, 395]



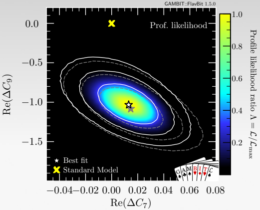
Right-Handed Neutrinos

[Eur.Phys.J.C 80 (2020) 6, 569]



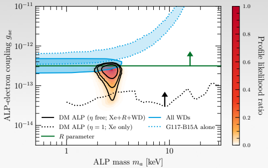
Flavour EFT

[arXiv:2006.03489 hep-ph]



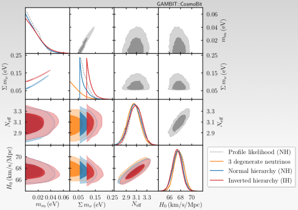
DM ALPs

[arXiv:2007.05517 astro-ph.CO]



Neutrino Masses

[arXiv:2009.03287 astro-ph.CO]



Karlsruher Institut für Technologie

Core

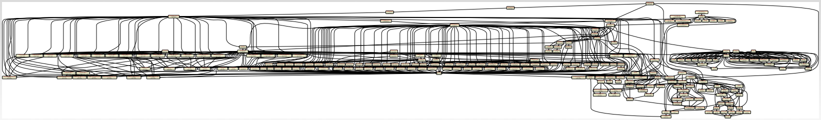
- Each module contains a collection of module functions
- Module functions provide a *capability*
- They have dependencies and backend requirements
- Allowed for specific models
- At run time a dependency tree is generated and resolved

```
// SM-like Higgs mass with theoretical uncertainties
#define CAPABILITY prec_mh
START_CAPABILITY

  #define FUNCTION FH_HiggsMass
  START_FUNCTION(triplet<double>)
  DEPENDENCY(unimproved_MSSM_spectrum, Spectrum)
  DEPENDENCY(FH_HiggsMasses, fh_HiggsMassObs)
  ALLOW_MODELS(MSSM63atQ, MSSM63atMGUT)
  #undef FUNCTION

  #define FUNCTION SHD_HiggsMass
  START_FUNCTION(triplet<double>)
  DEPENDENCY(unimproved_MSSM_spectrum, Spectrum)
  BACKEND_REQ(SUSYHD_MHiggs, (), MReal, (const MList<MReal>&))
  BACKEND_REQ(SUSYHD_DeltaMHiggs, (), MReal, (const MList<MReal>&))
  ALLOW_MODELS(MSSM63atQ, MSSM63atMGUT)
  #undef FUNCTION

#undef CAPABILITY
```



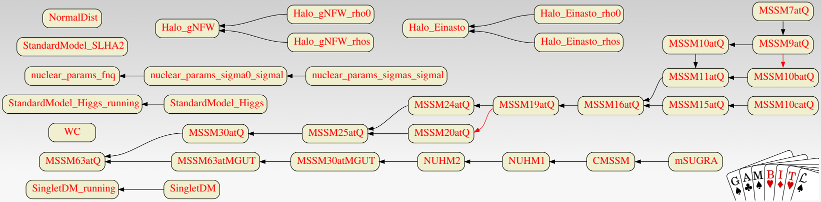
Models

Karlsruher Institut für Technologie

- Extensive model database

SUSY	DM	Cosmo	Others
CMSSM NUHM1,2 MSSM63atQ	Scalar Singlet Fermionic Singlet Vector Singlet Axions	ΛCDM ΔN_{eff} Power-law inflation	SM RH neutrinos WC nuisance models

- Parent-daughter hierarchy
- Module functions are activated for each model



Backends

- C, Fortran ⇝ POSIX `dl`
- C++ ⇝ BOSS + POSIX `dl`
- Mathematica ⇝ WSTP
- Python ⇝ pybind11

CosmoBit

AlterBBN 2.2
DarkAges 1.2.0
MontePythonLike 3.3.0
MultiModeCode 2.0.0
classy 2.9.4
plc 3.0

PrecisionBit

FeynHiggs 2.12.0
SUSYHD 1.0.2
gm2calc 1.3.0

DarkBit

CaptnGeneral 1.0
DDCalc 2.2.0
DarkSUSY 6.2.2
MicrOmegas 3.6.9.2
gamLike 1.0.1
nulike 1.0.9

SpecBit

FlexibleSUSY 2.0.1
SPheno 4.0.3

ColliderBit

HiggsBounds 4.3.1
HiggsSignals 1.4
Pythia 8.212
nulike 1.0.9

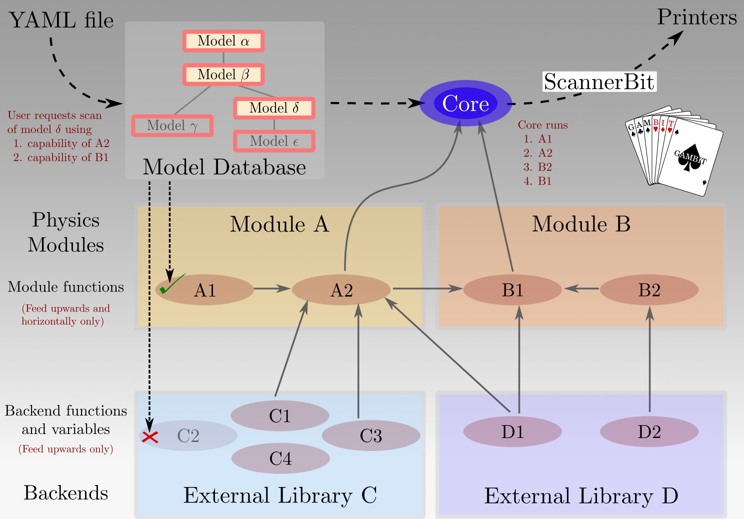
FlavBit

SuperISO 3.6

DecayBit

SUSY_HIT 1.5

Karlsruher Institut für Technologie



But...

How do I use GAMBIT with my favourite model?
⇝ Adding a model
⇝ Sorting out hierarchy
⇝ Making physics computations work with that model

How do I add a new physical observable or likelihood?
⇝ Create capabilities
⇝ Declare dependencies
⇝ and models
⇝ and backend requirements

1. Add the model to the **model hierarchy**:
   - Choose a model name, and declare any **parent model**
   - Declare the model's parameters
   - Declare any **translation function** to the parent model

```
#define MODEL NUHM1
#define PARENT NUHM2
  START_MODEL
  DEFINEPARS(M0,M12,mH,A0,TanBeta,SignMu)
  INTERPRET_AS_PARENT_FUNCTION(NUHM1_to_NUHM2)
#undef PARENT
#undef MODEL
```

2. Write the translation function as a standard C++ function:

```
void MODEL_NAMESPACE::NUHM1_to_NUHM2 (const ModelParameters &myP, ModelParameters &targetP)
{
   // Set M0, M12, A0, TanBeta and SignMu in the NUHM2 to the same values as in the NUHM1
   targetP.setValues(myP,false);
   // Set the values of mHu and mHd in the NUHM2 to the value of mH in the NUHM1
   targetP.setValue("mHu", myP["mH"]);
   targetP.setValue("mHd", myP["mH"]);
}
```

3. If needed, declare that existing module functions work with the new model, or add new functions that do.

Adding a new module function is easy:

1. Declare the function to GAMBIT in a module's **rollcall header**
   - Choose a capability
   - Declare any **backend requirements**
   - Declare any **dependencies**
   - Declare any specific **allowed models**
   - other more advanced declarations also available

```
#define MODULE FlavBit                               // A tasty GAMBIT module.
START_MODULE

  #define CAPABILITY Rmu                             // Observable: BR(K->mu nu)/BR(pi->mu nu)
  START_CAPABILITY
    #define FUNCTION SI_Rmu                          // Name of a function that can compute Rmu
    START_FUNCTION(double)                           // Function computes a double precision result
    BACKEND_REQ(Kmunu_pimunu, (my_tag), double, (const parameters*)) // Needs function from a backend
    BACKEND_OPTION( (SuperIso, 3.6), (my_tag) )                      // Backend must be SuperIso 3.6
    DEPENDENCY(SuperIso_modelinfo, parameters)       // Needs another function to calculate SuperIso info
    ALLOW_MODELS(MSSM63atQ, MSSM63atMGUT)            // Works with weak/GUT-scale MSSM and descendents
    #undef FUNCTION
  #undef CAPABILITY
```

2. Write the function as a standard C++ function (one argument: the result)

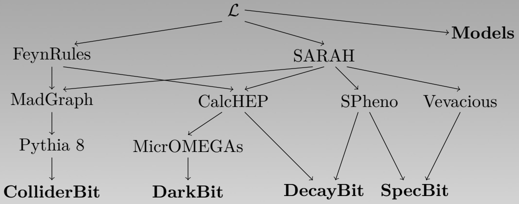
from P.Scott

Solution

The GAMBIT Universal Model Machine

GUM

- GUM interfaces LLT SARAH and FeynRules with GAMBIT
- Uses existing HEP toolchains

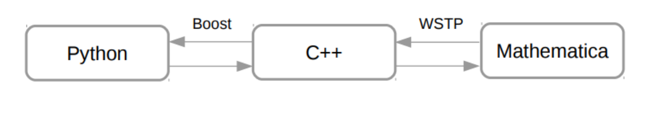


- GAMBIT-compatible outputs from GUM

Generated output	FeynRules	SARAH	Usage in GAMBIT
CalcHEP	✓	✓	Decays, cross-sections
micrOMEGAs (via CalcHEP)	✓	✓	DM observables
Pythia (via MadGraph)	✓	✓	Collider physics
SPheno	✗	✓	Particle mass spectra, decay widths
Vevacious	✗	✓	Vacuum stability

GUM

- Primarily written in Python, with interface to Mathematica via Boost and WSTP



- Automatically generates GAMBIT code
  - → Particles → particle database and parameters → Models
  - → Module functions for ColliderBit, DarkBit, DecayBit and SpecBit
  - → Writes interfaces to requested backends
- GUM will release with GAMBIT 2.0 **VERY SOON**

An example

- Majorana DM χ with scalar mediator Y

$$\mathcal{L} = \mathcal{L}_{\rm SM} + \frac{1}{2}\overline{\chi}\left(i\not{\partial} - m_\chi\right)\chi + \frac{1}{2}\partial_\mu Y \partial^\mu Y - \frac{1}{2}m_Y^2 Y^2 - \frac{g_\chi}{2}\overline{\chi}\chi Y - \frac{c_Y}{2}\sum y_f \overline{f} f Y \,.$$

```
math:
  # Choose FeynRules
  package: feynrules
  # Name of the model
  model: MDMSM
  # Model builds on the Standard Model FeynRules file
  base_model: SM
  # The Lagrangian is defined by the DM sector (LDM),
  # defined in MDMSM.fr, plus the SM Lagrangian (LSM)
  # imported from the 'base model', SM.fr
  Lagrangian: LDM + LSM
  # Make CKM matrix = identity to simplify output
  restriction: DiagonalCKM

# PDG code of the annihilating DM candidate in
↪FeynRules file
wimp_candidate: 52

# Select outputs for DM physics.
# Collider physics is not as important in this model.
output:
  pythia: false
  calchep: true
  micromegas: true
```