Fermi *Gamma-ray Space Telescope*

fermi
Gamma-ray
Space Telescope

New Software Tools and Analysis Methods Designed for an All-Sky Analysis of Gamma-Ray Emission from Dark Matter Interactions

Eric Charles

on Behalf of the Fermi-LAT Collaboration

6th Fermi Symposium

Arlington, VA

Nov. 11, 2015

Outline

- Dark matter catalog (DMCat) project
- Important analysis tools for the DMCat
  - HEALPix-based maps and data analysis in the framework of the LAT *Science Tools*
  - Likelihood v. flux SED scans (aka "Castro" plots)
  - Incorporating systematic uncertainties in likelihood fitting
  - Constraining model components
- Current status and plans

DMCat: Dark Matter Catalog Project

- Goal: create a set of all-sky maps of the likelihood v. $\langle\sigma v\rangle$ for a set of dark matter particle models (i.e., decay channel & masses) and use those maps together with sets of known DM targets to constrain DM annihilation / decay and *make it all publicly available as a Fermi data product*

- Considerations:
  - Systematic uncertainties from diffuse emission modeling
  - Treatment of catalog sources
  - Spatial extent (point sources vs. extended sources)
  - Data size / processing speed / parallelization

- Tool development is well underway

- Many of the tools have wider applicability beyond the DMCat project
 - Extra details available in posted slides

HEALPIX-BASED MAPS AND DATA ANALYSIS

HEALPix

- HEALPix = **H**ierarchical **E**qual **A**rea iso**L**atitude **Pix**elization

  - http://healpix.jpl.nasa.gov/

- Some very desirable features:

  - **H**ierarchical: easy to change resolution, sub-divide or merge ROIs

    - Also useful for adaptive numerical integration

  - **E**qual **A**rea iso**L**atitude: vastly improves computational efficiency of convolutions using spherical harmonics

- HEALPix is already widely used within LAT data analysis:

  - *Science Tools* currently can produce HEALPix counts cubes and exposure cubes

  - Livetime cubes (i.e., *gtltcube* output) are produced in non-standard HEALPix format

  - *Gadget*, *GaRDiAN* diffuse fitting packages

HEALPix Basics

Pixel Index: NESTED scheme, nside=2
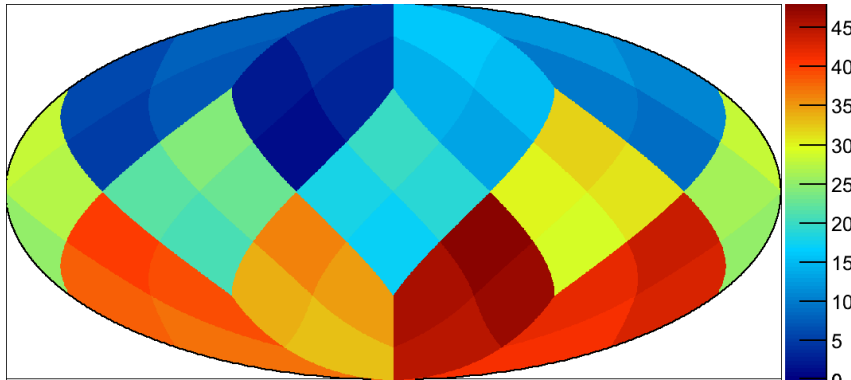
Pixel Index: RING scheme, nside=2
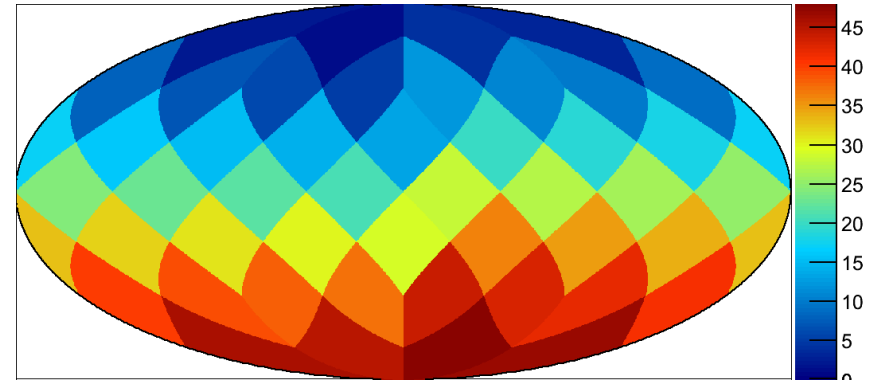
- HEALPix divides sphere in 12 pixels, then allows for subdivisions
- Two parameters specify a HEALPix pixelization
  - Nside: How many sub-divisions per side of each of the 12 original pixels
    - Total number of pixels = $12 \ast Nside^2$
    - Order: Nside = 2^Order gives a hierarchy of pixelizations, each with twice the resolution of the previous one
  - Scheme: "RING" or "NESTED" tells how to order the pixel indices
    - The actual pixels are the same either way
  - Coordinate system not specified, that is a separate question

Software Interface Changes for HEALPix

- As much as possible we tried to avoid changing any software interfaces in the *Science Tools*

- C++ classes (and python bindings) have the same functions and take the same inputs

- All the *Science Tools*, except for *gtbin,* take exactly the same input parameters whether the counts map is binned with WCS-based or HEALPix-based maps (see extra slides)

  - *gtbin* works largely as before, with additional input parameters to specify partial-sky ROIs for HEALPix binning (see extra slides)

- Developed partial-sky representation using HEALPix for parallel processing of small (~10°) regions

LIKELIHOOD V. FLUX SED GRID SCANS

Based on the Dwarf Spheroidal DM searches: e.g., Ackermann+ (2015, arXiv:1503.02641), see also talk by Alex Drlica-Wagner at the 5th Fermi symposium, and the LMC DM search, Buckley+ (2015, arXiv:1502.01020), see also talk by Regina Caputo
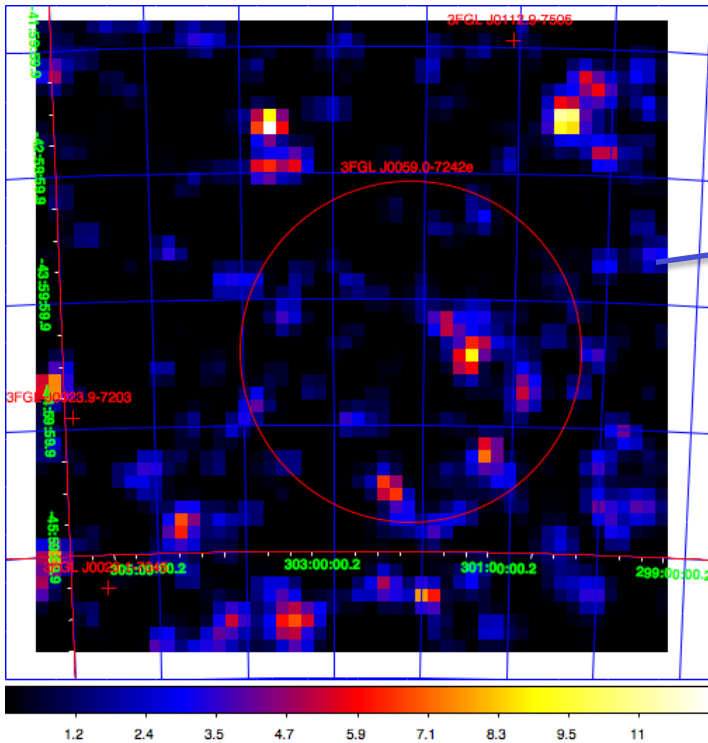
Likelihood v. Flux Scan

Energy bin-by-bin likelihood as a function energy flux for a simulation of a dsph
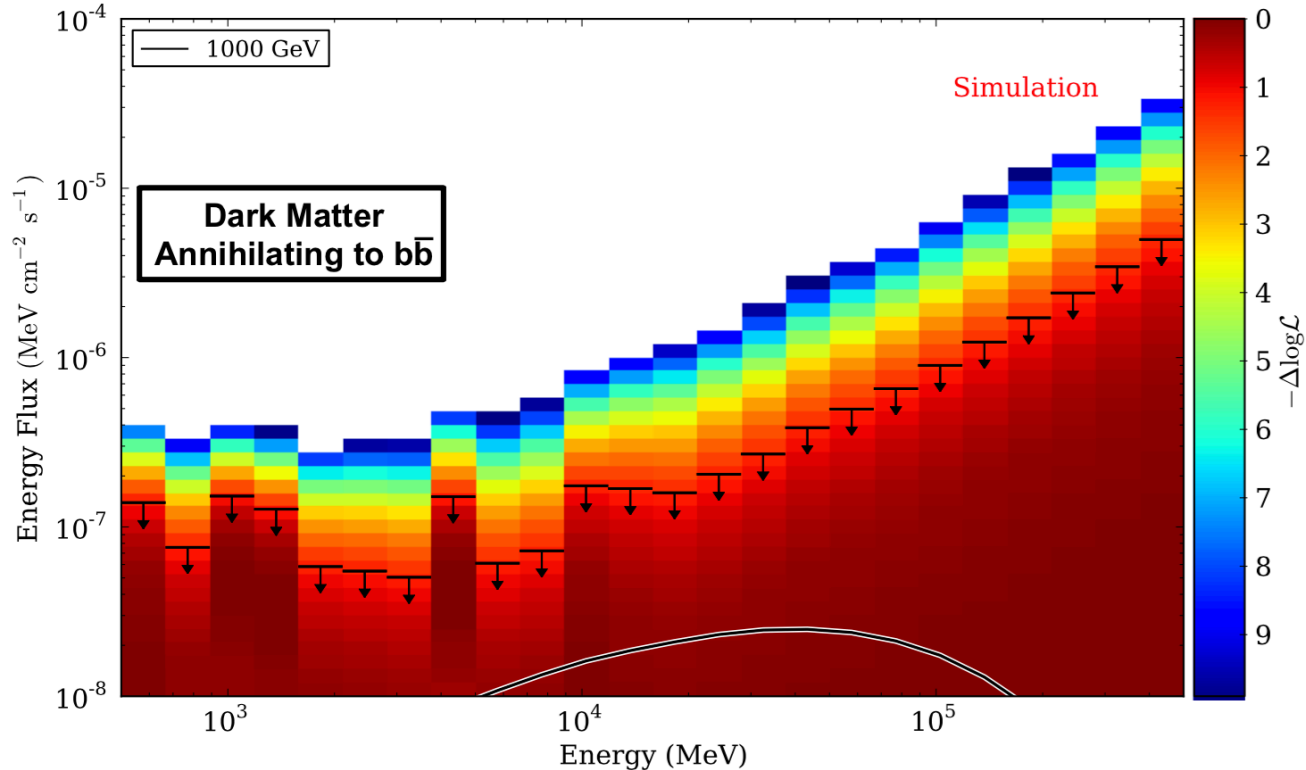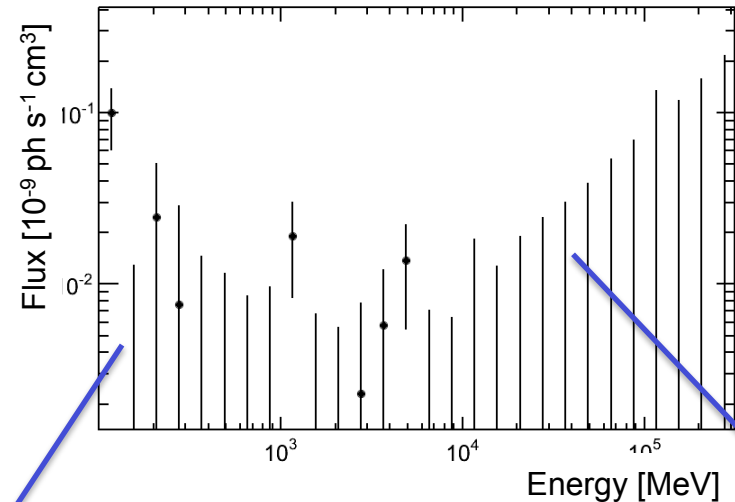
- So called "Castro plot" shows the likelihood as a function of Energy flux energy bin-by-bin for a specific DM spatial profile
- For each DM spectrum under consideration we vary the normalization (i.e., <σv>) and sum the bin-by-bin likelihood to obtain the likelihood as a function of <σv>
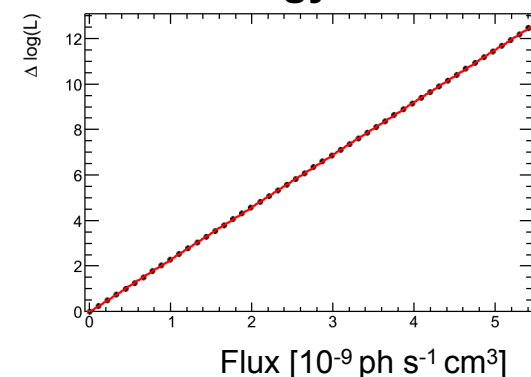
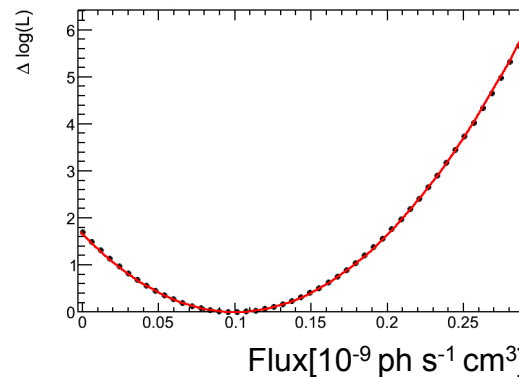# Test Statistic Cube

TS Map of 5° x 5° Region

SED for test source in one pixel

Likelihood v. Flux for each energy bin

$$TS = 2\log\frac{\mathcal{L}(M(\hat{\alpha}); N)}{\mathcal{L}_{\text{null}}(M(\hat{\alpha}); N)}.$$

- Plots are for a *gtmodel*-based simulation of the 10° x 10° ROI around the Small Magellanic Cloud (with the SMC removed from the model)
- P8R2_SOURCE_V6 IRFs

Gttscube Application

- *Science Tools* compatible application to make TS cube files
  - Creates TS cube file
  - Very fast (up to 100x faster than *gttsmap*, see extra slides)
  - Map region and scan parameters configurable
  - Performs highly-optimized and specialized fitting

INCORPORATING SYSTEMATIC UNCERTAINTIES IN LIKELIHOOD FITTING

Methodology from Line Searches & LMC DM Search:
Albert+ (2014 arXiv:1406.3430), Ackermann+ (2015 arXiv:1506.00013),
Buckley+ (2015 arXiv:1502.01020)
see also talk by Andrea Albert

Accounting for Systematic Uncertainties in Fits

$$\mathcal{L}(\alpha, \Gamma, n_{\text{sig}}, n_{\text{syst}}) = P_{\mathcal{F}}(n_{\text{syst}}, b_{\text{eff}}) \prod_i P(c_i | \mu_i(\alpha, \Gamma, n_{\text{sig}} + n_{\text{syst}}))$$

$$P_{\mathcal{F}}(n_{\text{syst}}, b_{\text{eff}}) = \frac{1}{\sigma_{\text{syst}}\sqrt{2\pi}} \exp\left(-\frac{(n_{\text{syst}} - \mu_{\text{syst}})^2}{2\sigma_{\text{syst}}^2}\right)$$

- Assume that the "apparent" signal has contributions from the "real" signal, n_{sig} and from the "systematically induced" signal, n_{syst}
  - Treat them as having exactly the same shape PDF, so in principle they would be degenerate
  - Except that we put a Gaussian prior, $P_{\mathcal{F}}$, on n_{syst}, the width of which is determined from control samples
  - For now we do not try to take out any average bias observed in the control samples, i.e., $\mu_{\text{syst}} = 0$

Effect of Systematic Uncertainties

Likelihood curves with different levels of systematic uncertainty
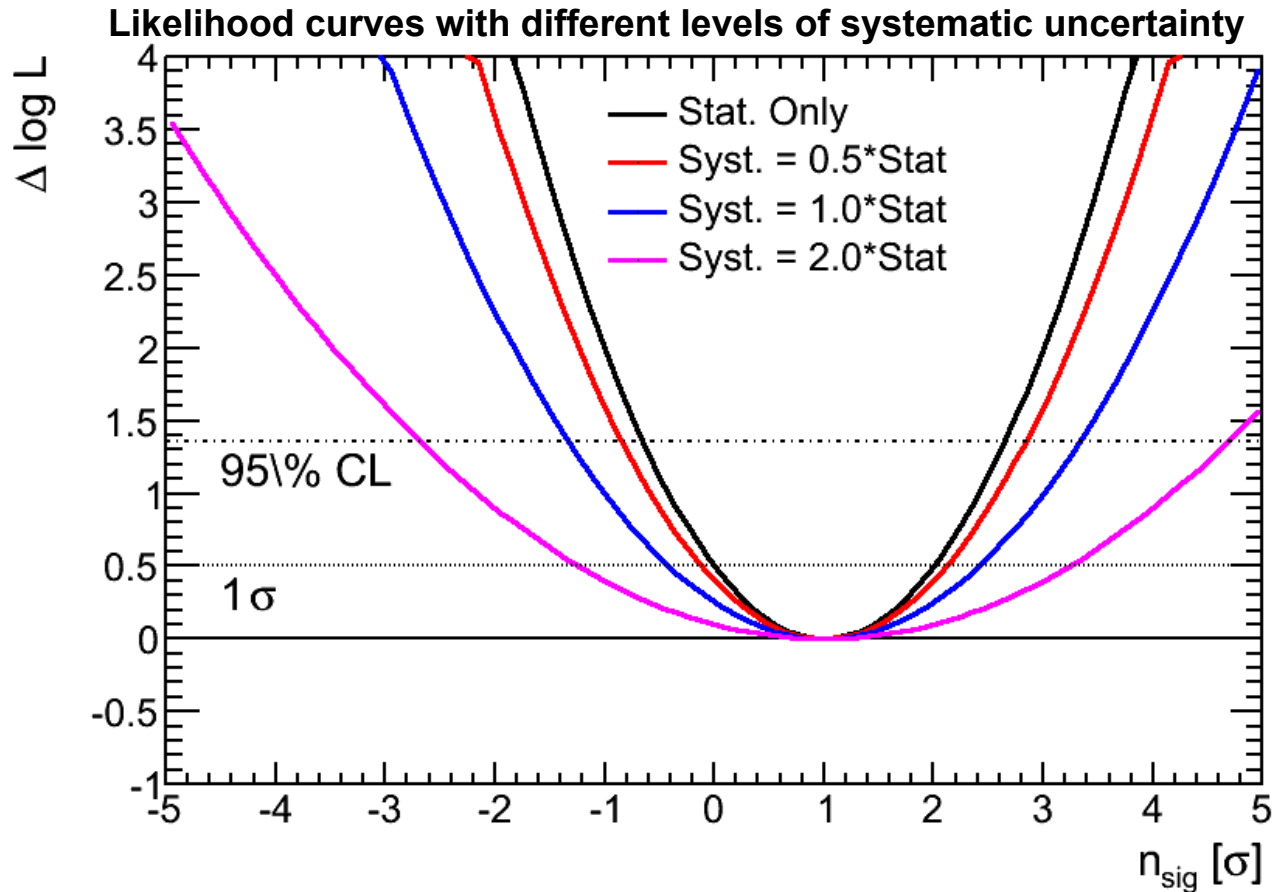

- This formulation allows the n_{syst}, component to absorb signal at a cost to the likelihood determined by σ_{syst}
- This will make the likelihood curve shallower, increasing the uncertainty on n_{sig} (and consequently also increasing the ULs)

CONSTRAINING MODEL COMPONENTS IN FITTING

Many Analyses Have Several Correlated Background Components

Correlation factor between background components and DM signal

Buckley+ (2015, arXiv:1502.01020)
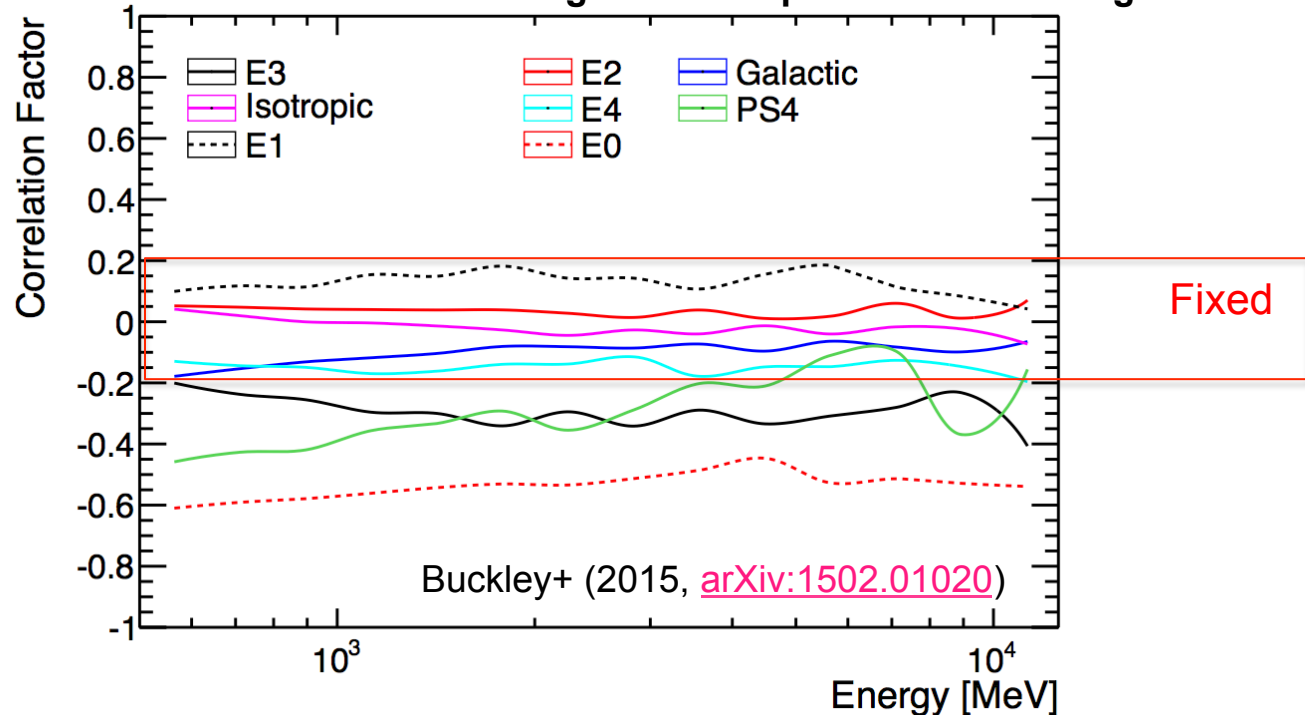
- Basic idea: fix the components that are least correlated with the test source to reduce degeneracy, improve convergence and speed
  - Estimate correlation between model components and test source
  - Apply a threshold (e.g., | Correl. Fact. | > 0.2) to select which components are to be constrained instead of fixed

Adding a Multivariate Prior to the Likelihood

- We assume that we have a prior constraint on some of the baseline model component normalizations (more on that on next slide)
  - And that the prior can be expressed as favored values ĉ and a covariance matrix V_{prior}
- Essentially the constraint is a multivariate Gaussian:

$$\mathcal{L}_{\mathrm{prior}} = \exp[-(\vec{c} - \tilde{c})^T H_{\mathrm{prior}}(\vec{c} - \tilde{c})]$$

$$-\log \mathcal{L}_{\mathrm{prior}} = (\vec{c} - \tilde{c})^T H_{\mathrm{prior}}(\vec{c} - \tilde{c})$$

$$H_{\mathrm{prior}} = V_{\mathrm{prior}}^{-1}$$

- And we can calculate contributions to the 1st and 2nd derivatives:

$$\vec{g}_{\mathrm{prior}} = \frac{\partial(-\log \mathcal{L}_{\mathrm{prior}})}{\partial c_{\mathrm{a}}} = H_{\mathrm{prior}}(\vec{c} - \tilde{c})$$

$$H_{\mathrm{prior}} = \frac{\partial^2(-\log \mathcal{L}_{\mathrm{prior}})}{\partial c_{\mathrm{a}} \partial c_{\mathrm{b}}} = H_{\mathrm{prior}}$$

CURRENT STATUS AND PLANS

Current Status and Plans

- Status of tools and plans for public release
  - HEALPix-based fitting is tested in *Science Tools* framework and in the queue to be integrated into the *Science Tools*
  - *gttscube* is integrated in *Science Tools* and in the queue for eventual release
  - Tools to incorporate systematic uncertainties and constrain background components have been incorporated into Dwarf analysis pipeline (not public) and will be released with DMCat software

EXTRA DETAILS

HEALPix PSF Convolutions

- Three cases to consider (which were already dealt with very cleverly by *Gadget* / *GaRDiAN* / *pointlike*):

  - Point sources: use hierarchical feature of HEALPix to perform adaptive integration of PSF by subdividing central pixels more and more finely

    - Stole the code from *GaRDiAN* (1 function)

  - All-sky diffuse sources: use fast-convolution with HEALPix spherical harmonic decomposition

    - Stole the code from *GaRDiAN* (2 functions: 1st to put PSF-image at the pole, 2nd to do convolution)

  - Extended, but not all-sky, sources: (i.e., LMC, SMC, SNeR, etc..)

    - Use existing *Science Tools* code to do convolution in native map projection, then convert the convolved map to HEALPix (copied parts from *pointlike*)

HEALPix partial skymaps

7.5° radius ROI
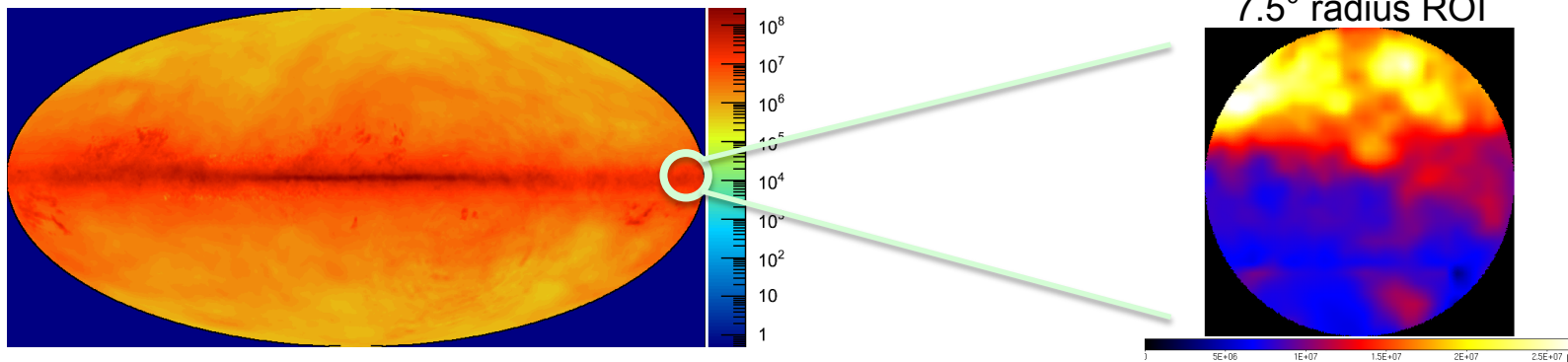
Name	Type	Dimensions (R X C)
SKYMAP	BinTableHDU	n_pixel X n_energy+1

KEYWORD	Value
INDXSCHM	'EXPLICIT'
HPXREGION	'DISK(184.5298,-5.8361,7.5)'
TTYPE1	'PIX' % Name of column with pixel indices
TFORM1	J % 32 bit signed integer

This can be generalized for more complicated ROIs

Extra col. with pixel indices

This is based on how DES used HEALPix for the dSphs search (2015ApJ...807...50B). These extra keywords and extra column specify the ROI for partial skymaps, and are present in each table with a HEALPix skymap

Test Statistic Cube File

```
No. Type          EXTNAME    BITPIX Dimensions(columns)
  0  PRIMARY                  -32       50 50      # TS Map
  1  BINTABLE SCANDATA     8        11200(2) 2500

     Column Name    Format       Dims
     1 NORMSCAN     1400E     (28,50,) # Normalization values for likelihood scan
     2 NLL_SCAN     1400E     (28,50,) # Delta log-likelihood values from scan


  2  BINTABLE EBOUNDS      8        10(3) 28   # Standard energy bins HDU
  3  IMAGE    N_MAP     -32       50 50       # Map of best-fit norm. values
  4  IMAGE    ERRP_MAP  -32       50 50       # Positive errors on norm. value
  5  IMAGE    ERRN_MAP  -32       50 50       # Negative errors on norm. value
  6  IMAGE    TSCUBE    -32       50 50 28   # TS Map for each energy bin
  7  IMAGE    N_CUBE    -32       50 50 28   # Best-fit norm for each energy bin
  8  IMAGE    ERRPCUBE  -32       50 50 28   # Positive errors on bin norms
  9  IMAGE    ERRNCUBE  -32       50 50 28   # Negative errors on bin norms
 10  IMAGE    NLL_CUBE  -32       50 50 28   # Delta log-likelihood for test-fit
```

Extension of TS map to 4 dimensions (x,y,energy,normalization)

Gttscube Input and Fitting Parameters

```
#
# Input data set parameters
#
cmap,f,a,"",,,"Counts map file"
bexpmap,f,a,"",,,"Binned exposure map"
expcube,f,a,"",,,"Exposure hypercube file"
evtype,i,h,INDEF,,,"Event type selections"
irfs,s,a,"CALDB",,,"Response functions to use"


# Input model parameters
srcmdl,f,a,"",,,"Source model file"
psfcorr,b,h,yes,,,"apply psf integral corrections"
remakesrc,b,h,no,,,"Remake the test source to each grid point?"


# Fitter parameters
optimizer,s,a,"MINUIT",DRMNFB|NEWMINUIT|MINUIT|DRMNGB|LBFGS,,"Optimizer"
ftol,r,h,1e-3,,,"Fit tolerance"
toltype,s,h,"ABS","ABS|REL",,"Fit tolerance convergence type"
maxiter,i,h,30,,,"Maximum number of iterations for Newton's method fitting"
stlevel,i,h,1,0,4,"Standard optimizer fitting up to what scan loop"
```

Gttscube Output Parameters

```
# Output file parameters
outfile,f,a,"smc_p8src_gal_tscube_v6.fits",,,"TS cube file name"
nxpix,i,a,10,1,,Number of X axis pixels
nypix,i,a,10,1,,Number of Y axis pixels
binsz,r,a,0.1,,,Image scale (in degrees/pixel)
coordsys,s,a,"GAL",CEL|GAL,,"Coordinate system"
xref,r,a,,0,360,"X-coord of image center in degrees (RA or l)"
yref,r,a,,-90,90,"Y-coordinate of image center in degrees (Dec or b)"
proj,s,a,"CAR",AIT|ARC|CAR|GLS|MER|NCP|SIN|STG|TAN,,Projection method
nnorm,i,h,10,2,100,"Number of normalization points"
nsigma,r,h,5.0,,,"Width of normalization scan, in sigma"
covscale,r,h,-1.,,,"Scale factor to apply to broadband errors in bin-
  by-bin fits ( < 0 -> fixed )"
```

gttscube Timing Data

Run Parameters	Timing (s)
gttsmap (baseline model indices free)	> 10k
gttsmap (baseline model indices fixed)	5980.5
gttscube (Newton's method baseline re-fit, nnorm=10)	17.6
gttscube (Newton's method baseline re-fit, nnorm=50)	28.2
gttscube (Newton's method baseline re-fit, nnorm=10, remake sources)	253.4
gttscube (Newton's method baseline re-fit, nnorm=50, 50 x 50 map)	514.1
gttscube (MINUIT baseline re-fit, nnorm=10)	46.4
gttscube (MINUIT baseline re-fit, nnorm=50)	54.3
gttscube (MINUIT baseline re-fit, nnorm=10, constrained)	70.7
gttscube (MINUIT broadband fits for each pixel, test source fixed Γ=2)	3215.5

- Fits to simulated data of SMC region (with SMC removed)
- 10 x 10 pixel TS Map, (0.1° pixel size -> 1° x 1°)
- 4 point sources and isotropic & Gal. diffuse with free normalizations

Optimization I: Fitting Normalization Only

- Extract model cubes (i.e., predicted counts) from *BinnedLikelihood* object
- Fit for normalization scaling constants, i.e., $P_a = c_a M_a$ ← Baseline counts model
- Calculate 1st and 2nd derivatives analytically:

$$\vec{g} = \frac{\partial(-\log \mathcal{L})}{\partial c_a} = \sum_i (1 - \frac{n_i}{P_i}) M_{a,i}$$

Scale w.r.t. baseline model

$$V_{ab}^{-1} = H_{ab} = \frac{\partial^2(-\log \mathcal{L})}{\partial c_a \partial c_b} = \sum_i (\frac{n_i}{P_i^2}) M_{a,i} M_{b,i}$$

- Iterate fit parameters with Newton's method:

$$\vec{\delta c} = V \vec{g} \qquad \vec{c}_{n+1} = \vec{c}_n + \delta \vec{c}$$

- Convergence based on vertical step size:

$$\Delta(-\log \mathcal{L}) \simeq \vec{g} \cdot \delta \vec{c}$$

Optimization II: Simplify Parameter Bounds

- Since we are fitting normalizations only, the only real constraint on the parameters is that they be positive

- Simply clip δc at current parameter values to avoid any scale factor ever going negative

  - Much simpler than remapping parameter (as per MINUIT) or treating convergence criteria for parameters near bounds specially

Optimization III: Only Loop Over Bins with Counts

- For sparse data (i.e., lots of pixels with no counts) it is much more efficient only to track the model contributions from pixels that have counts and also keep track of the total model prediction

$$- \log \mathcal{L} = \sum_i n_i \log P_i - P_{\text{tot}}$$

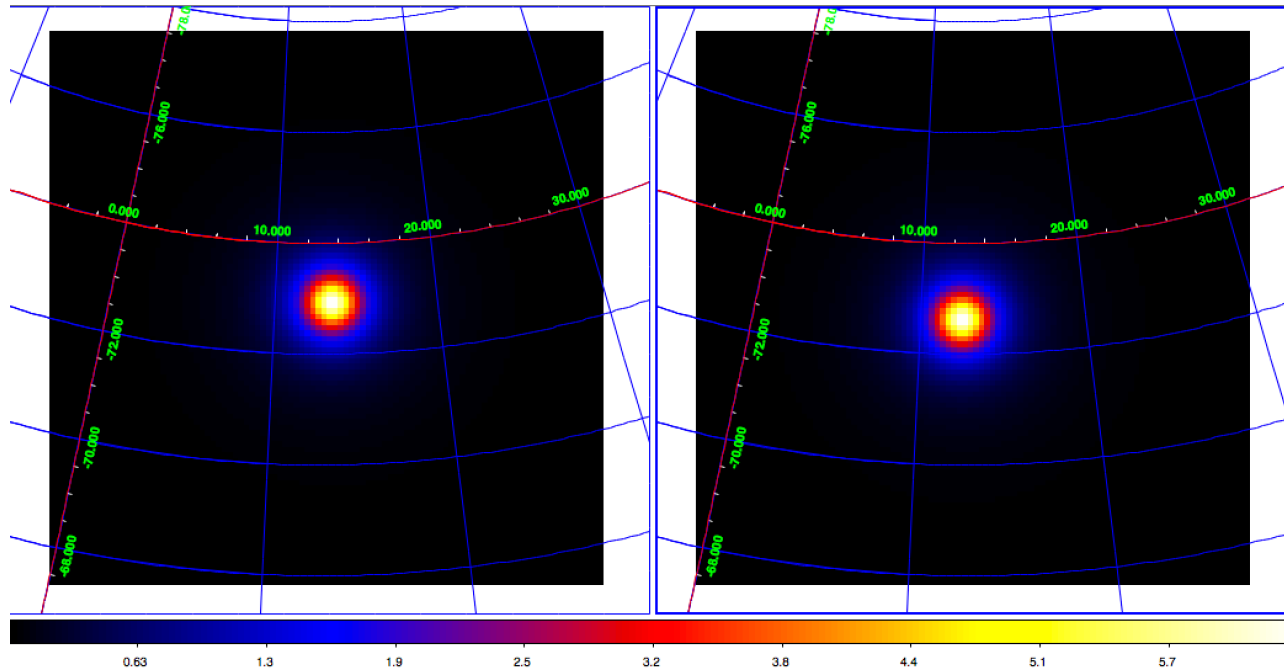- For the 1st and 2nd derivatives of the log-likelihood we have:

$$\vec{g} = \frac{\partial(-\log \mathcal{L})}{\partial c_a} = M_{a,tot} - \sum_i (\frac{n_i}{P_i}) M_{a,i}$$

$$V_{ab}^{-1} = H_{ab} = \frac{\partial^2(-\log \mathcal{L})}{\partial c_a \partial c_b} = \sum_i (\frac{n_i}{P_i^2}) M_{a,i} M_{b,i}$$

- This is implemented in the *BinnedLikelihood* class, but not yet in the specialized Newton's method fitter

Optimization IV: Shift the Test Source Image

**PSF image, 560-750 MeV, pixel-shifted by -0.3°,+0.3° between images**

- The optimized fitting is fast enough that computing the expected counts test source image is the speed-limiting factor (by a factor of ~5-10)

- We are usually working with locally flat projections and relatively small ROIs, so we can shift the image by a fixed number of pixels instead of re-computing it

- Still need to quantify errors, establish working parameters

Estimating the Systematics in the LMC DM Search

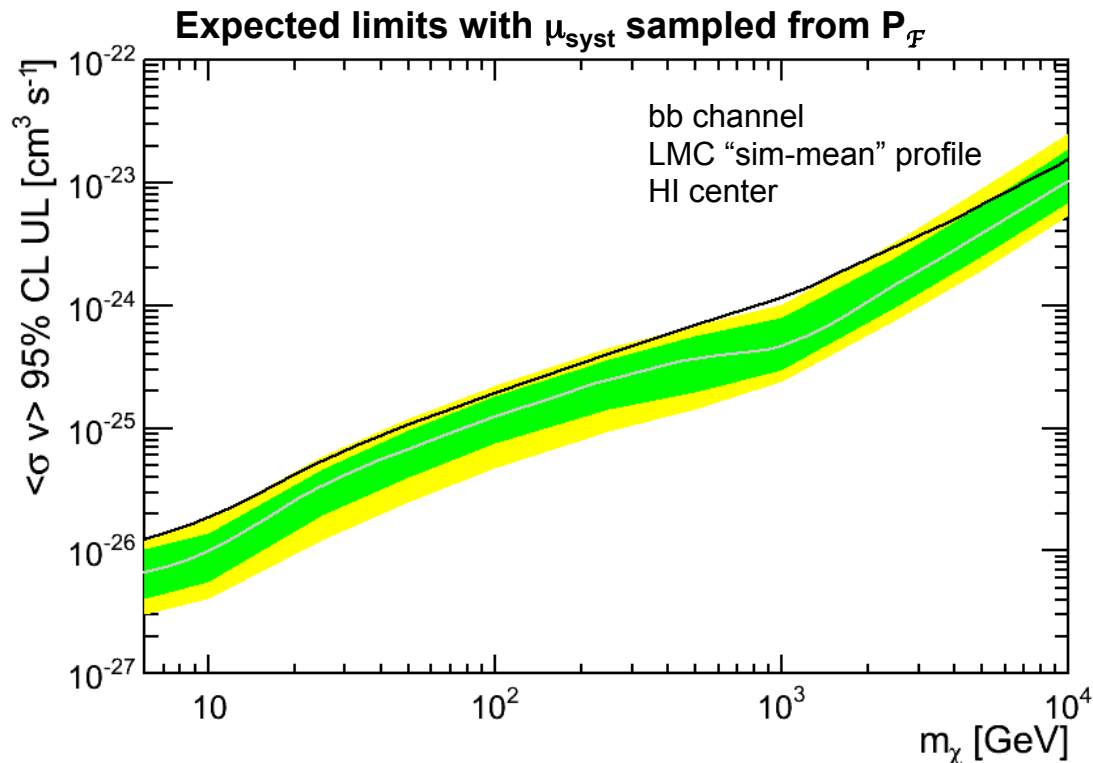**"TS Map" for LMC, showing control region**
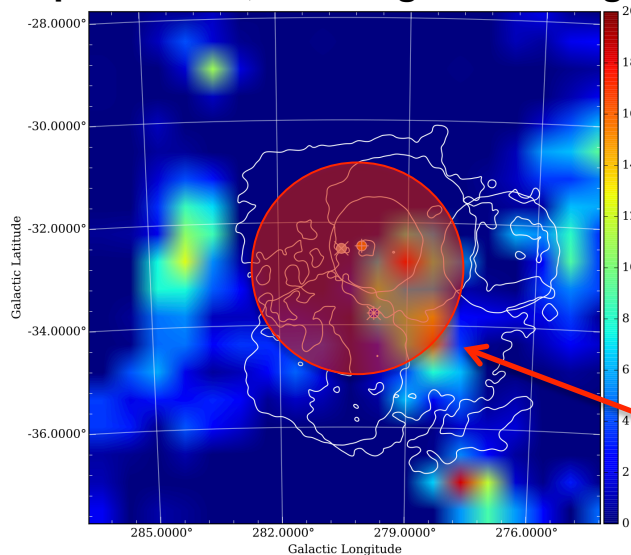
Distribution of f_{stat} and f_{sig} for LMC control regions

Difference between total error and statistical error is a measure of syst. error
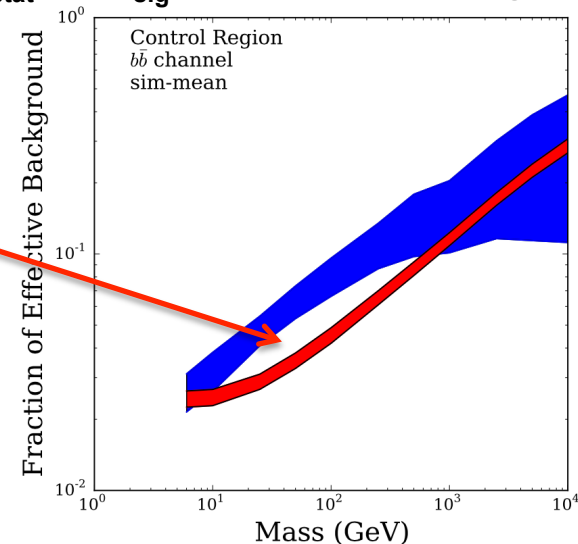
- For the LMC we use the area away from the signal region near the kinematic center of the LMC as our control region

  - Calculation of b_{eff} includes spatial information

  - Unlike line-search, signal is not allowed to go negative

    - Use the 84% (i.e., +1σ) quantiles of f_{sig} and f_{stat} to characterize error
    - Take the difference in quadrature of f_{sig} and f_{stat} as f_{syst}

Buckley+ (2015, arXiv:1502.01020)

Effect of Systematic Uncertainties on Upper Limits

Expected limits with μ_{syst} sampled from $P_{\mathcal{F}}$


bb channel
LMC "sim-mean" profile
HI center

- Estimate σ_{syst} from control regions (details provided in posted slides)
- Sample μ_{syst} from $P_{\mathcal{F}}$ when doing MC simulations to establish expected limit bands (yellow and green bands)
  - This gives the same results as combining the statistical and systematic uncertainties in quadrature

Buckley+ (2015, arXiv:1502.01020)

Building the Multivariate Prior

- Prior depends on context
  - Going from all-sky fitting to small ROI:
    - Constrain all-sky diffuse model components
  - Going from broadband SED to energy bin-by-bin fitting:
    - Constrain baseline components to broadband values
- Constrain or fix nuisance parameter values?
  - It depends…
- Scale covariance matrix to allow parameter values from fits to move around within errors of broadband fit (e.g., when the statistical uncertainties from the fit are very large compared to the prior constraint)

Dark Matter Catalog Overview
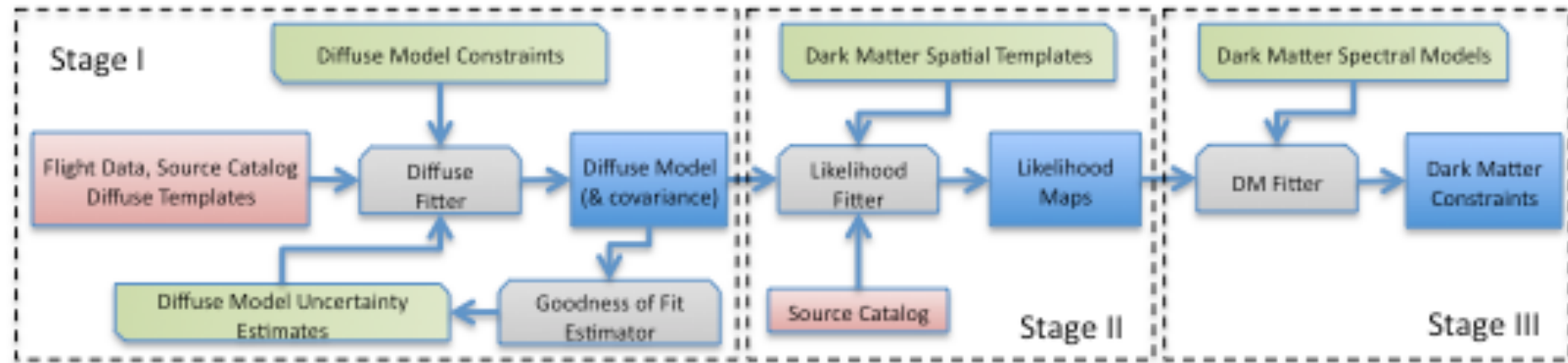
- DMCat stage I data product: all-sky fit to diffuse model (and covariances between model components)
  - HEALPix based all-sky fitting
- DMCat stage II data product: all-sky TS cubes with diffuse correlations included
  - all-sky map for point sources and small extended sources
  - specialized template maps for specific regions
- DMCat final data product: DM limits, including systematic uncertainties for given target lists

DMCat Software/ Analysis Tools Overview

Software tool/ package	Purpose	Status
HEALPix-based fitting in *Science Tools*	Interface to all-sky fitting tools such as Gadget, *GaRDiAN*	Written, tested
TS cube generation	Making TS cube scans of small ROIs	Written, tested
TS cube-based spectral fitting	Scanning over DM channels/ masses	Adapt from dSph pipeline
DMCat stage I pipeline	Data preparation, book-keeping for all-sky fitting, simulations	Adapt from dSph pipeline/ all-fitting
DMCat stage II pipeline	Parallelization & book-keeping for ROI fitting. TS cube generation, simulations	Adapt from dsph pipeline
DMCat stage III pipeline	Converting from TS cube to DM constraints, book-keeping for DM target lists, application of systematic errors	Adapt from dsph pipeline