The boloSource() algorithm

Gábor Marton (Konkoly), Roland Vavrek (HSC),
Csaba Kiss (Konkoly), Thomas Müller (MPE), Zoltán Balog (MPIA)

Outline

- The boloSource() algorithm
- Photometry with boloSource()
- Conclusions
- Installation & Settings
- Demo

The boloSource() algorithm

- Extended emission analysis requires clean maps
- Compact objects contribute to the image power spectra with a significant power at a broad range of spatial frequencies
- Image analysis techniques are difficult to compare if sources are not subtracted, because their sensitivity to discrete sub-structures may be quite different
- For extended emission analysis we need a technique to subtract sources that fall within a well defined range of spatial frequencies
- A major requirement: preserve noise properties of the image!
  - ➡ Classical way: try modeling the source intensity $I_{(x,y)}$ in the position-position space and subtract from the image
    - ➡This is not easy, but one could reduce the problem to 1D in the detector timeline
- Subtract sources from the detector timeline and re-project the image

Source subtraction in the timeline

- Looks difficult but we are mainly interested in to subtract high-frequency components. In the masked part of the timeline one could interpolate with simulated noise + sky background:

$$I_{(t)} = \underbrace{N^{1/f}_{(t)} + N^{D}_{(t)} + N^{det}_{(t)}}_{\text{Simulated noise}} + \underbrace{I^{S(lowfreq)}_{(t)}}_{\text{Baseline estimate from data}}$$

Interpolated intensity
in masked timeline

- Stationary wavelet transform SWT (á trous algorithm)

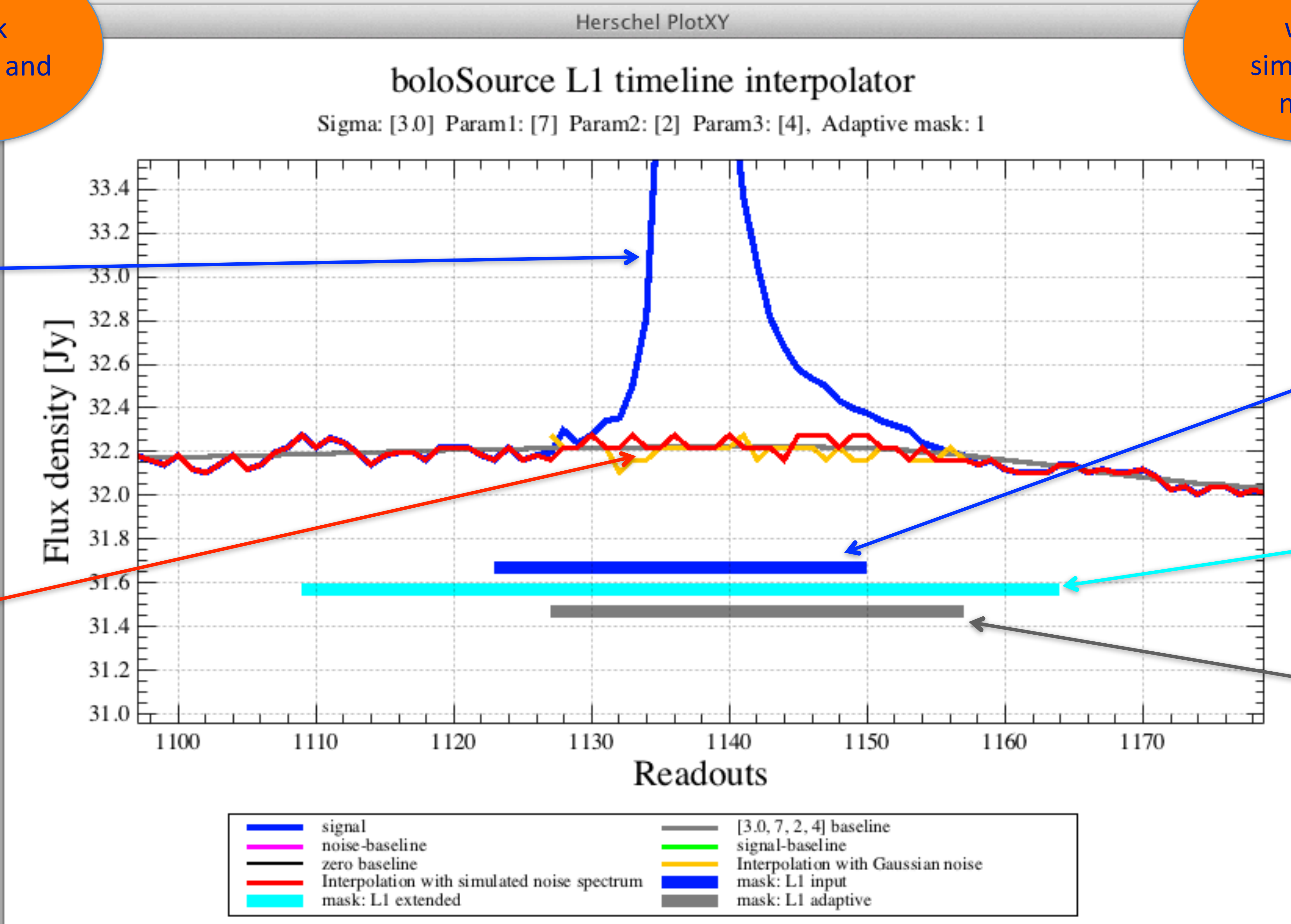
- The objective is to simulate intensity distribution of a single scan-leg with similar noise power spectrum as we experience in the observed data.

Timeline interpolation

L1 bolometer timeline example (red band, pixel [8,8])

Optimize L1 mask location and size

Interpolate with simulated noise



L1 bolometer signal in the timeline

Interpolated signal

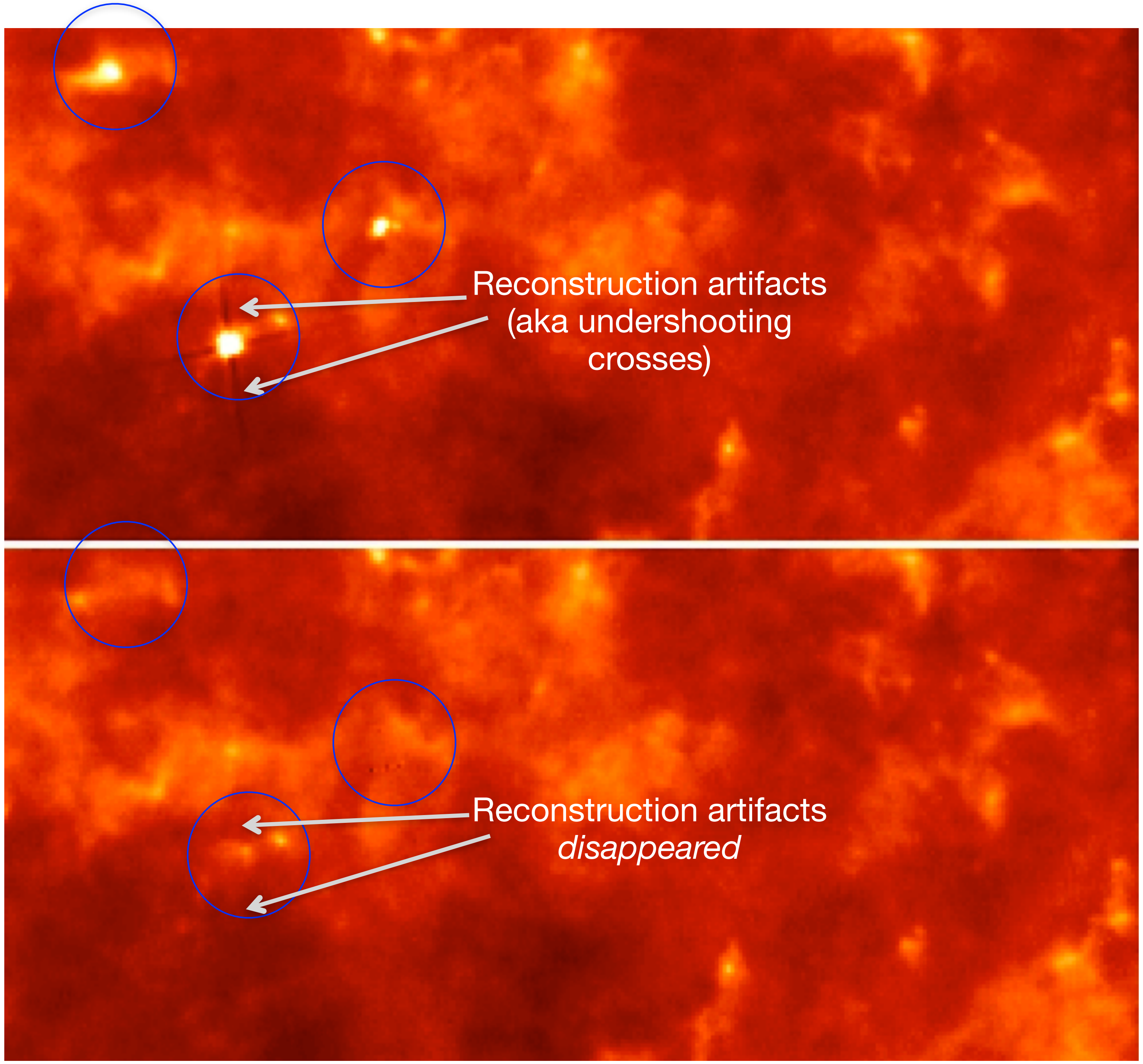
Back projected input mask

Intermediate working mask

Optimized adaptive mask

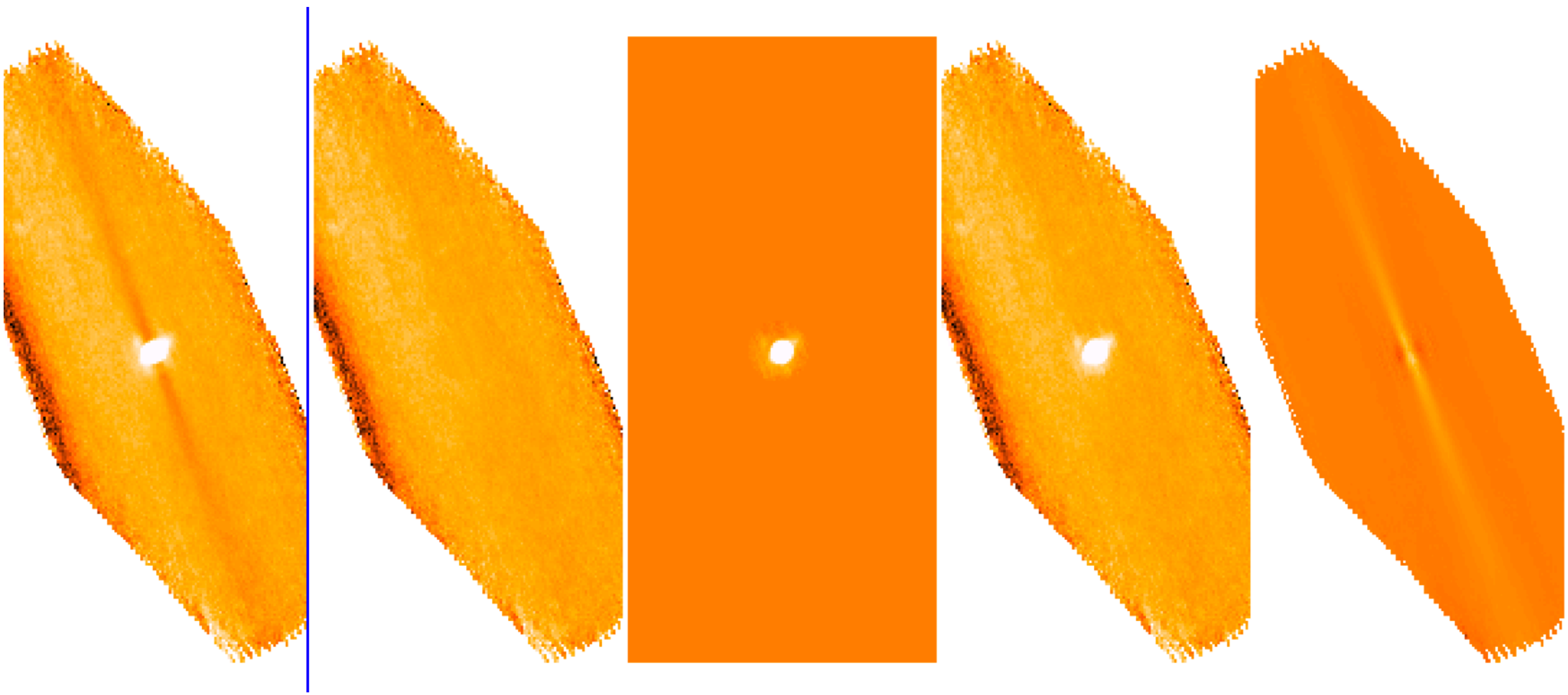
First run of boloSource()

- First run of boloSource (Hi-GAL Field297_0)
- Background is quite well preserved
- By product: MadMap reconstruction noise (undershooting artifacts) could be eliminated
- For the analysis of extended emission there is no strict need for other cleaning techniques



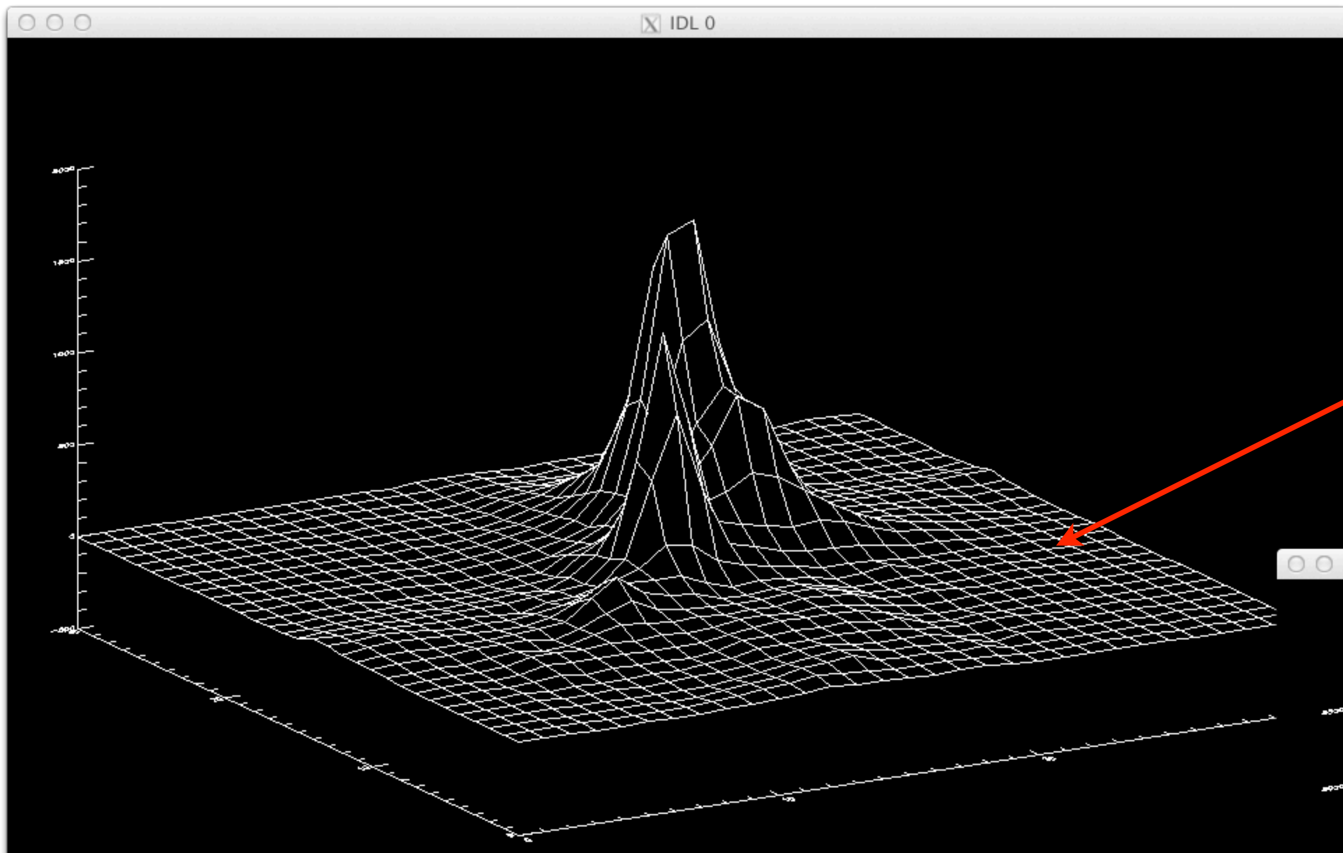
Side effect

original Unimap	interpolated Unimap	differential photProject	interpolated+ differential	interpolated+ differential- original

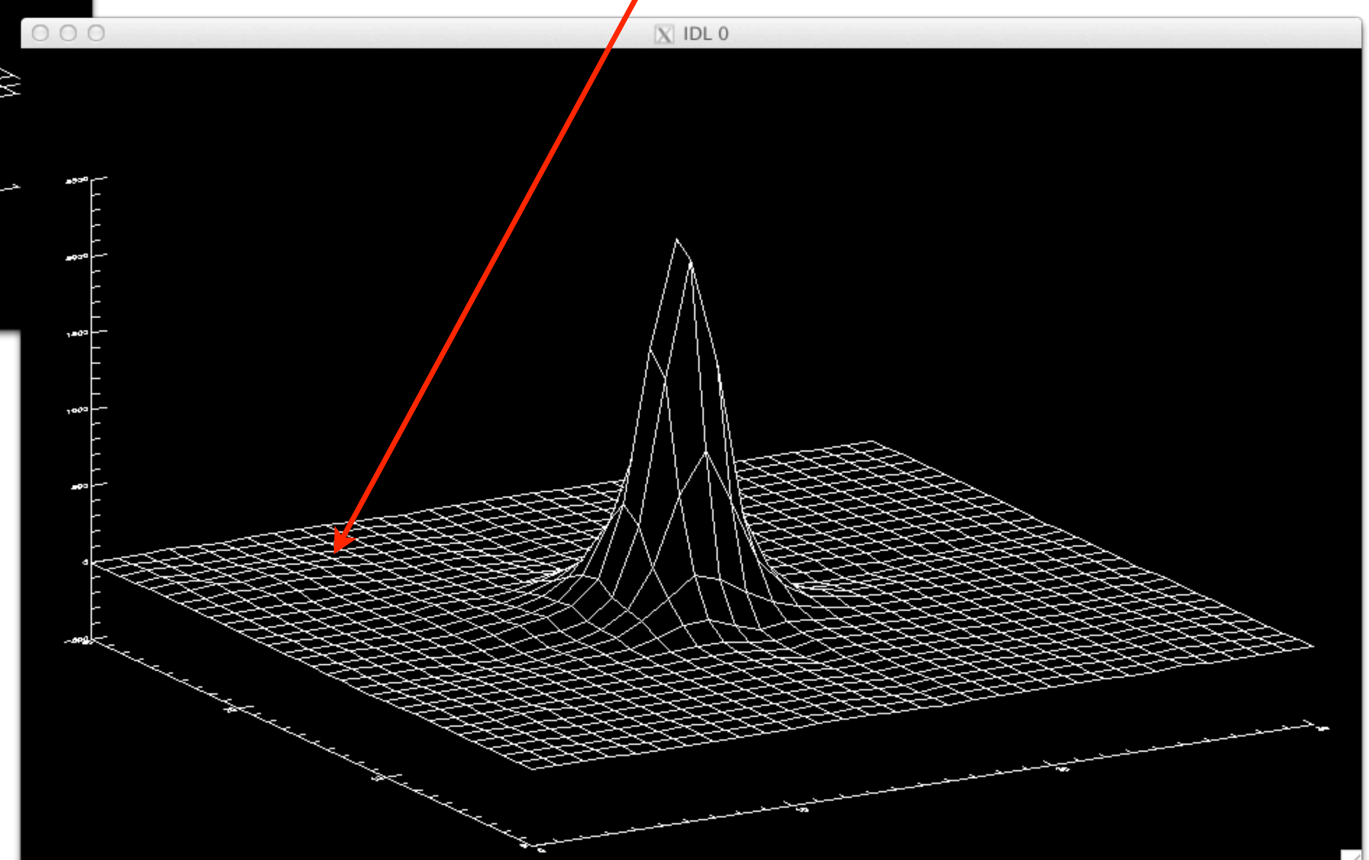


Side effect

sources from boloSource() differential



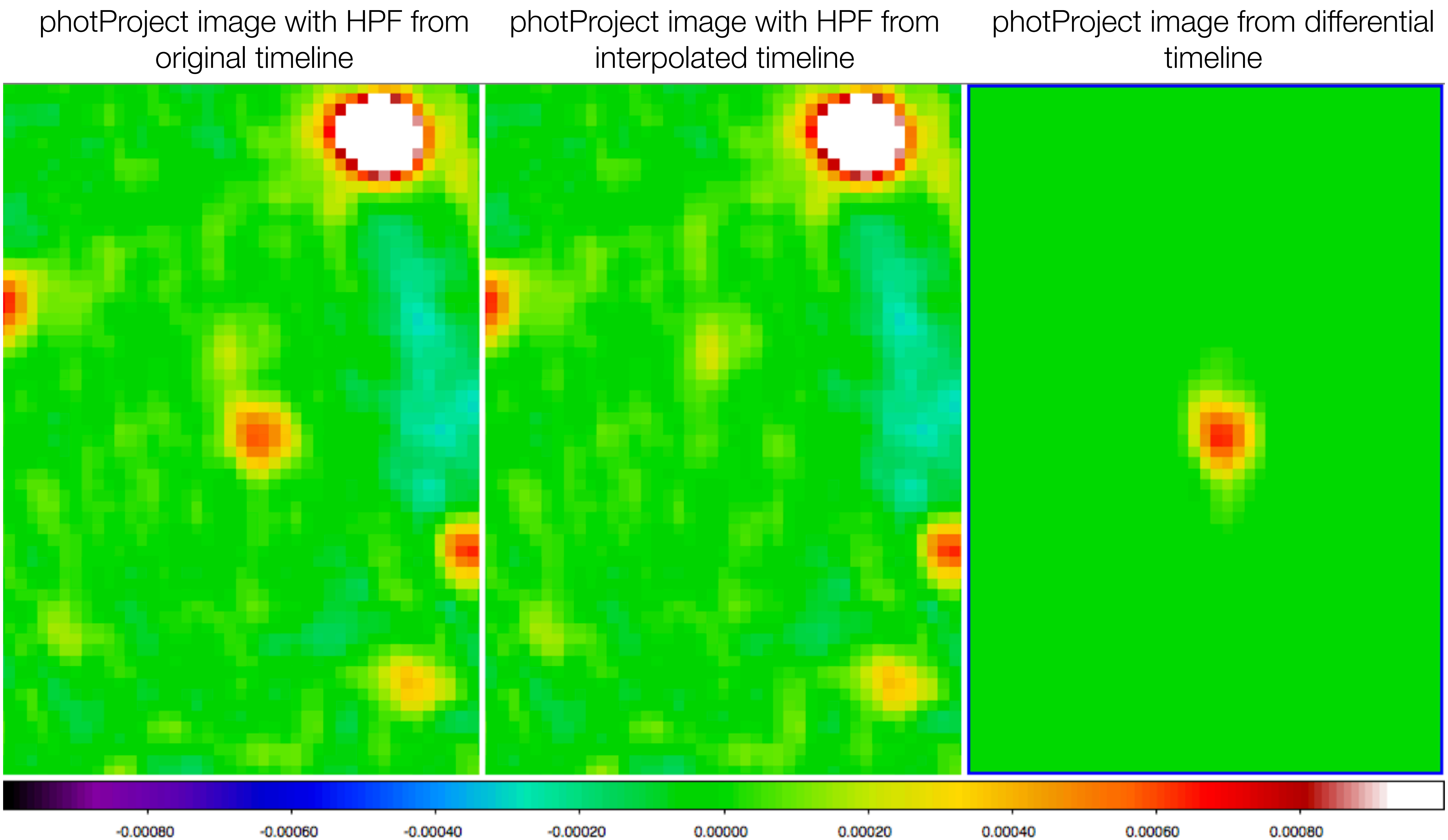
Another byproduct: source-only maps with flat, zero-level background



Photometry - The standard star dataset

Name	Blue [mJy]	Green [mJy]	Red [mJy]
HD15008	22.0	10.8	4.2
HD152222	37.9	18.6	7.3
HD39608	29.7	14.6	5.7
HD159330	61.7	30.2	11.8
HD139669	286	140	54
HD138265	111.4	54.6	21.3
HD170693	147.7	72.4	28.3

Photometry with boloSource() - HD 170 693 RED



Regular photometry

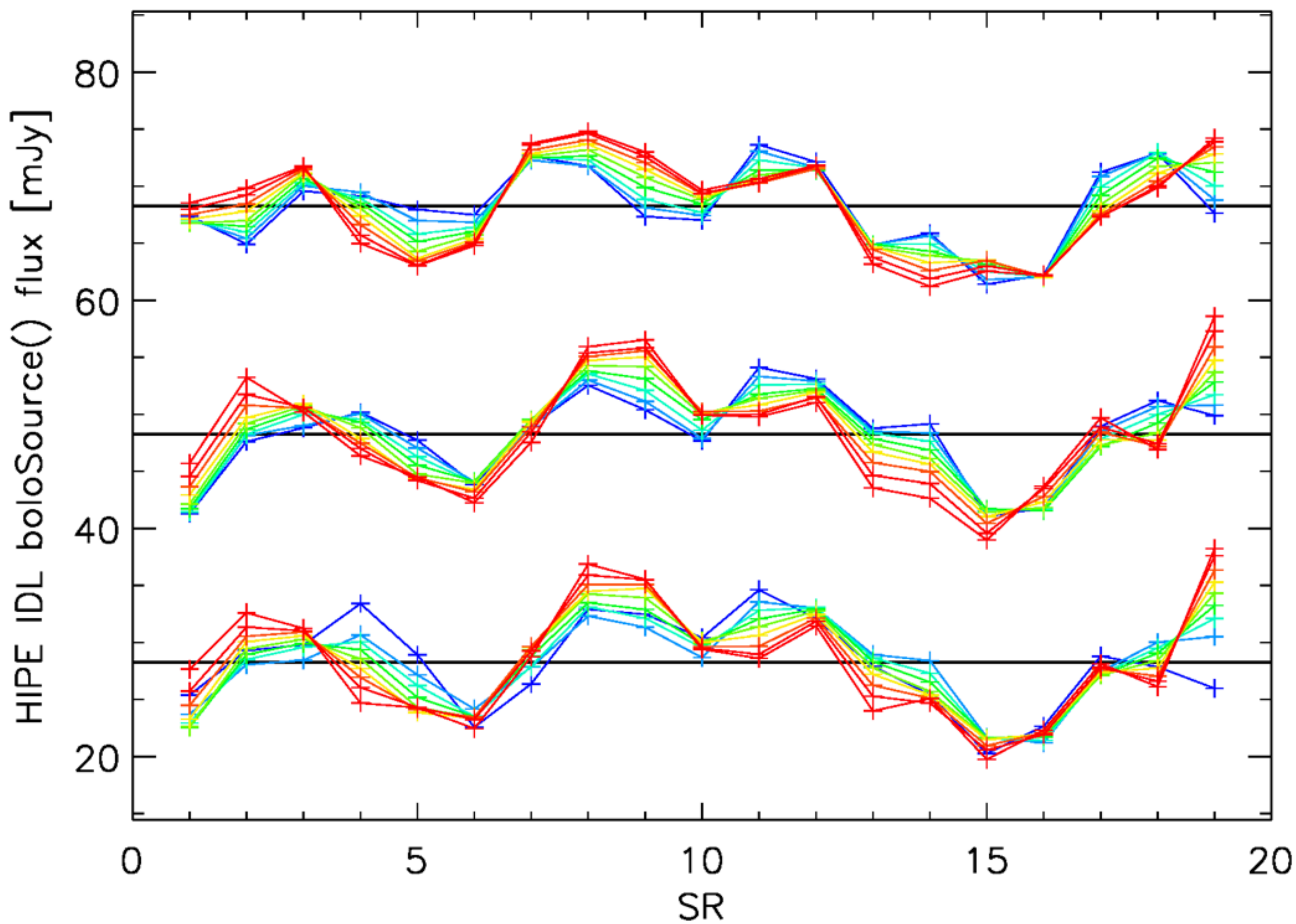
L1 timeline data
↓
high-pass filtering
↓
photProject
↓
centroid fitting
↓
aperture photometry

boloSource() photometry

L1 timeline data
↓
boloSource() interpolation
↓
source-only timeline
↓
photProject
↓
centroid fitting
↓
aperture photometry

Photometry - regular & boloSource()

- HD170693 - 20 repetitions
- Light curve constructed by merging 2 consecutive repetitions
- Aperture photometry with HIPE and IDL with aperture radius ranging from 2” to 10”
- The average of the light curve needs to be corrected to match the predicted brightness
- Characterization with the σ standard deviation and with the C correction factor

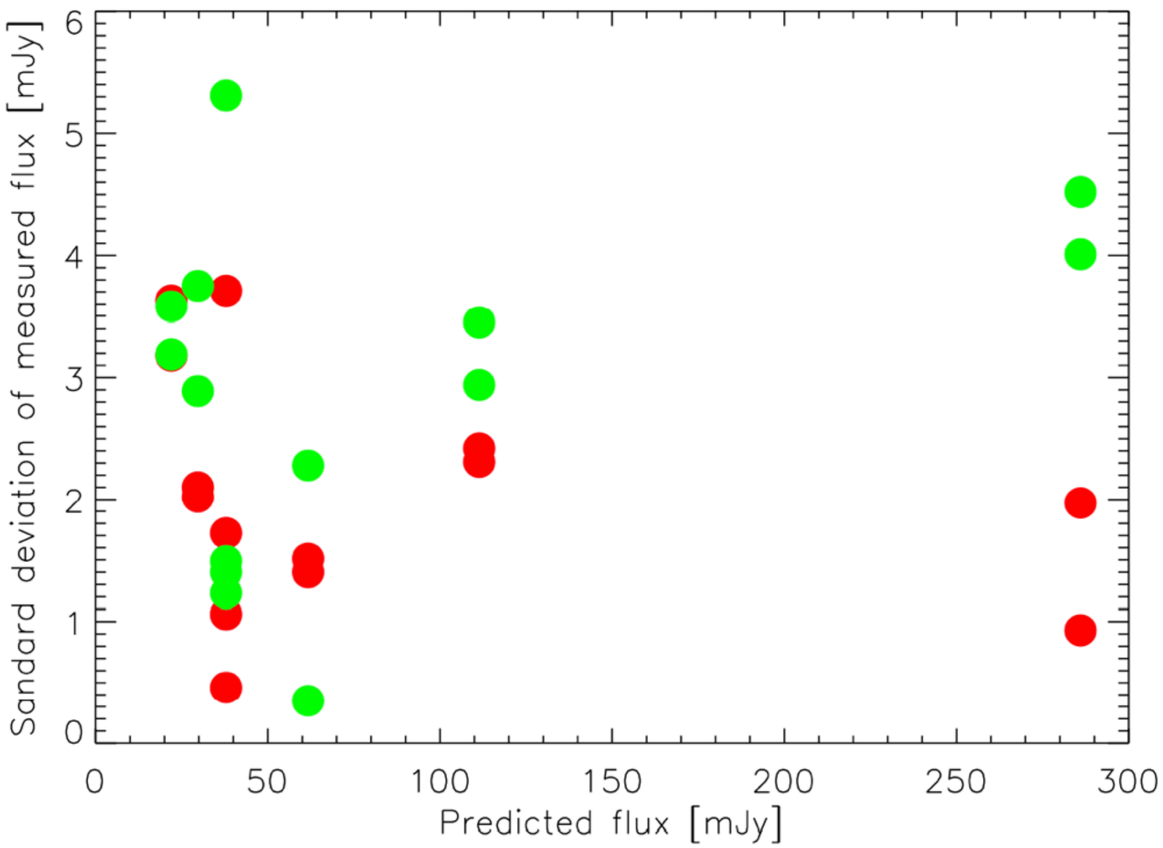


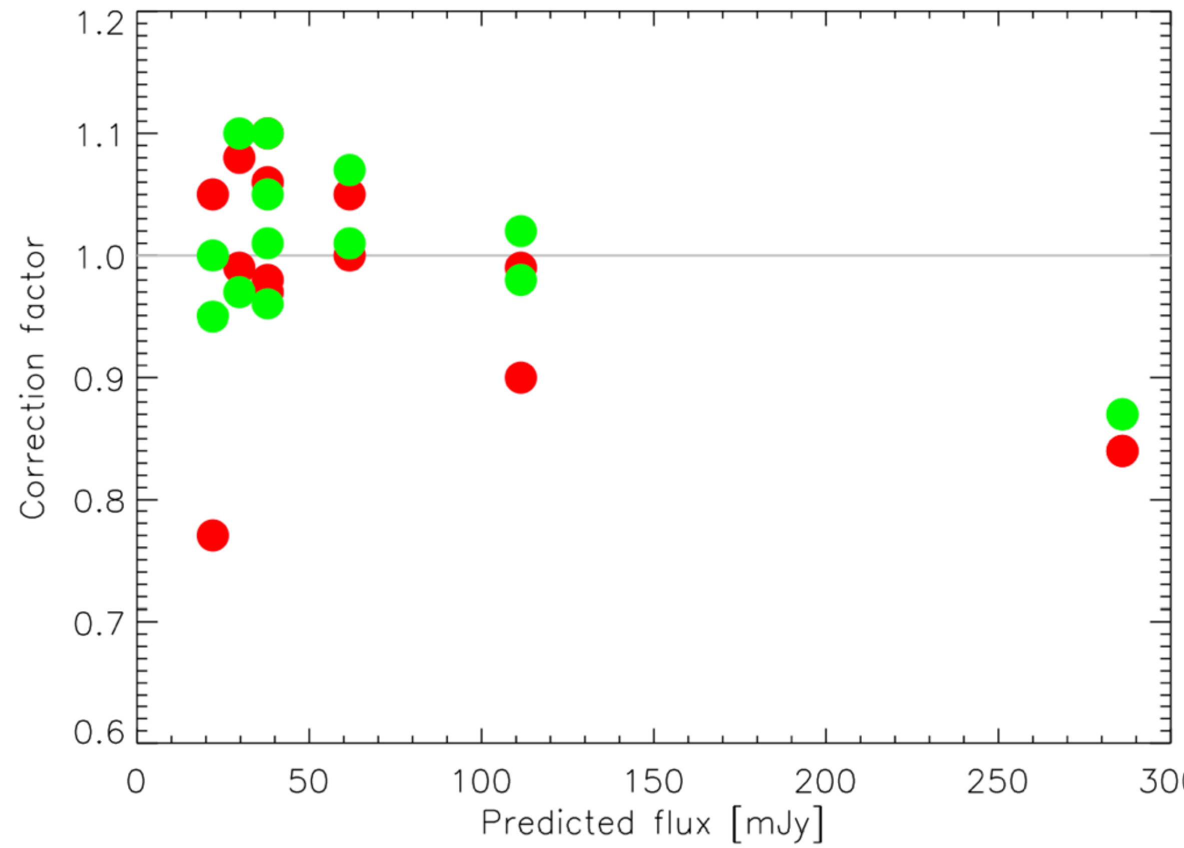
Photometry - regular & boloSource() - 70um

Aperture radius = 5.0″

Name	OBSID	Repetitions	Brightness [mJy]	σ_{IDL}	C_{IDL}	$\sigma_{bS()}$	$C_{bS()}$
HD 15008	1342189130	9	22.0	3.63	1.05	3.58	1.00
HD 15008	1342189131	9	22.0	3.18	0.77	3.19	0.95
HD 152222	1342240702	6	37.9	3.71	0.98	5.31	1.01
HD 152222	1342240703	6	37.9	1.72	1.10	1.40	1.10
HD 152222	1342191964	6	37.9	1.06	0.97	1.23	0.96
HD 152222	1342191965	6	37.9	0.46	1.06	1.49	1.05
HD 39608	1342198535	10	29.7	2.10	1.08	3.75	1.10
HD 39608	1342198536	10	29.7	2.02	0.99	2.89	0.97
HD 159330	1342213585	6	61.7	1.40	1.00	0.35	1.01
HD 159330	1342213586	6	61.7	1.51	1.05	2.28	1.07
HD 139669	1342191982	6	286	0.93	0.84	4.01	0.87
HD 139669	1342191983	6	286	1.97	0.84	4.52	0.87
HD 138265	1342188841	21	111.4	2.31	0.90	2.94	0.98
HD 138265	1342188842	21	111.4	2.42	0.99	3.45	1.02

Photometry - regular & boloSource() - 70um





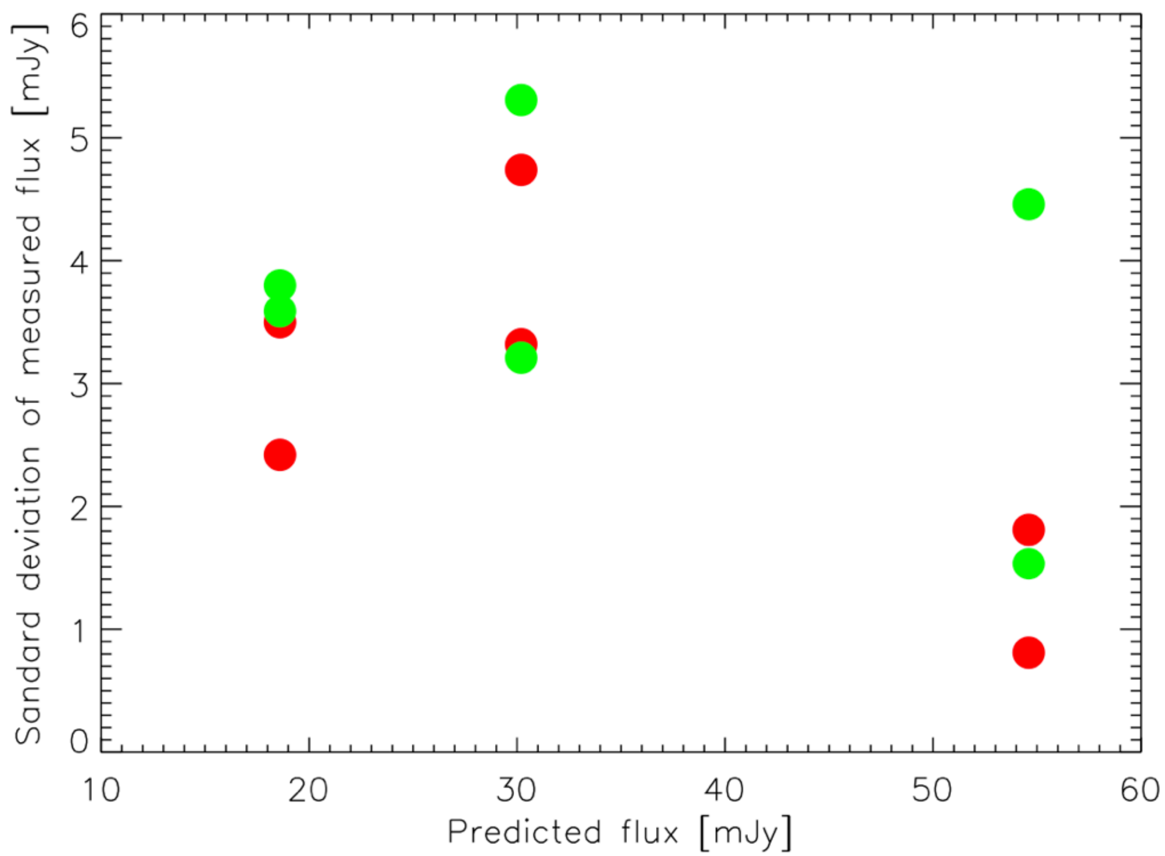
σ				C			
IDL		bS()		IDL		bS()	
Median	AVG	Median	AVG	Median	AVG	Median	AVG
2.02	2.03	3.19	2.89	0.99	0.97	1.01	1.00

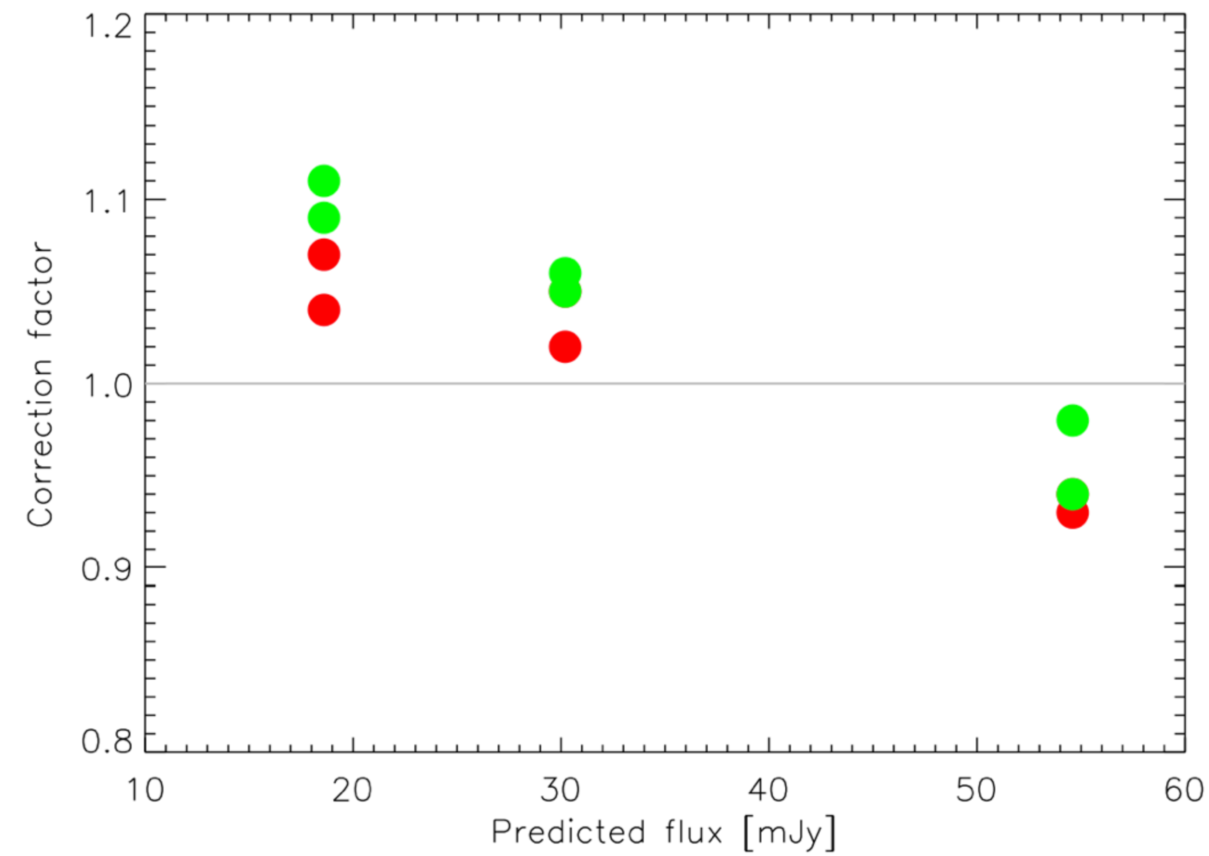
Photometry - regular & boloSource() - 100um

Aperture radius = 6.0″

Name	OBSID	Repetitions	Brightness [mJy]	σ_{IDL}	C_{IDL}	$\sigma_{bS()}$	$C_{bS()}$
HD 152222	1342227973	33	18.6	2.42	1.04	3.59	1.09
HD 152222	1342227974	33	18.6	3.50	1.07	3.80	1.11
HD 159330	1342188839	10	30.2	4.74	1.02	5.30	1.05
HD 159330	1342188840	10	30.2	3.32	1.05	3.21	1.06
HD 138265	1342191986	6	54.6	1.81	0.94	1.53	0.98
HD 138265	1342191987	6	54.6	0.81	0.93	4.46	0.94

Photometry - regular & boloSource() - 100um





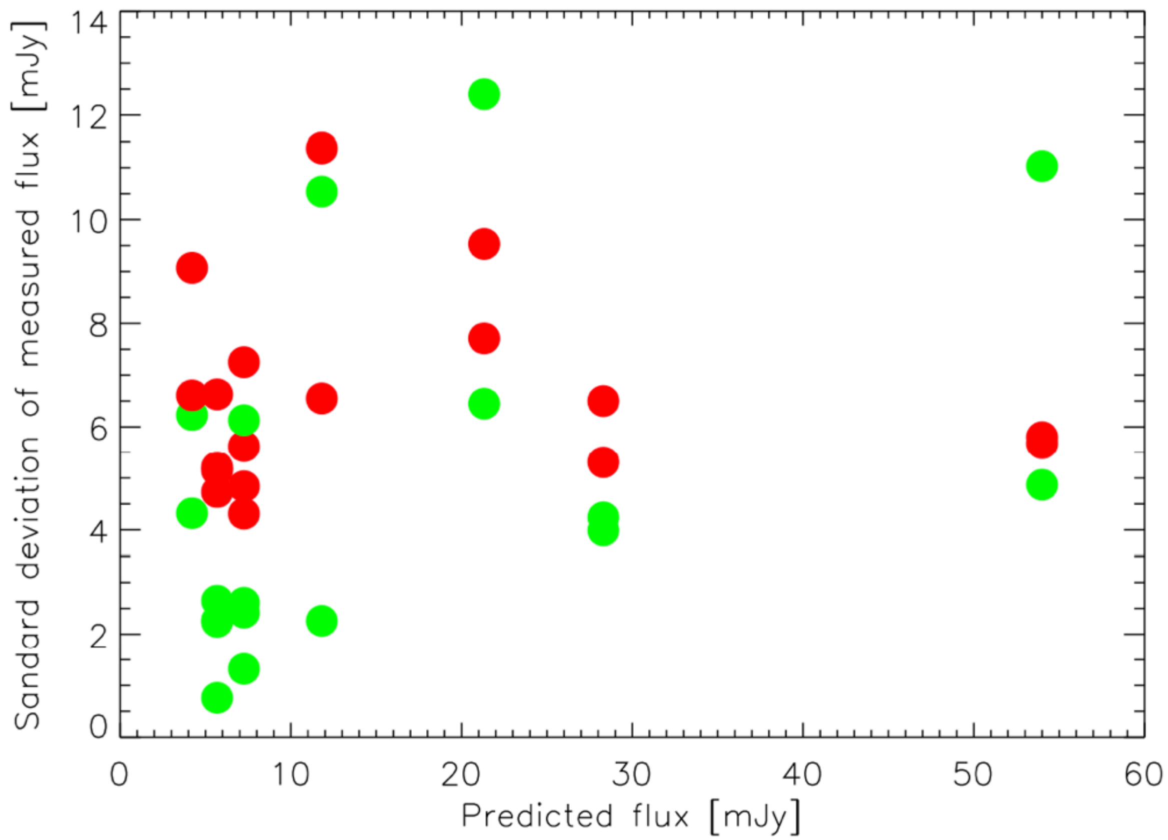
σ				C			
IDL		bS()		IDL		bS()	
Median	AVG	Median	AVG	Median	AVG	Median	AVG
3.32	2.77	3.80	3.65	1.04	1.01	1.06	1.04

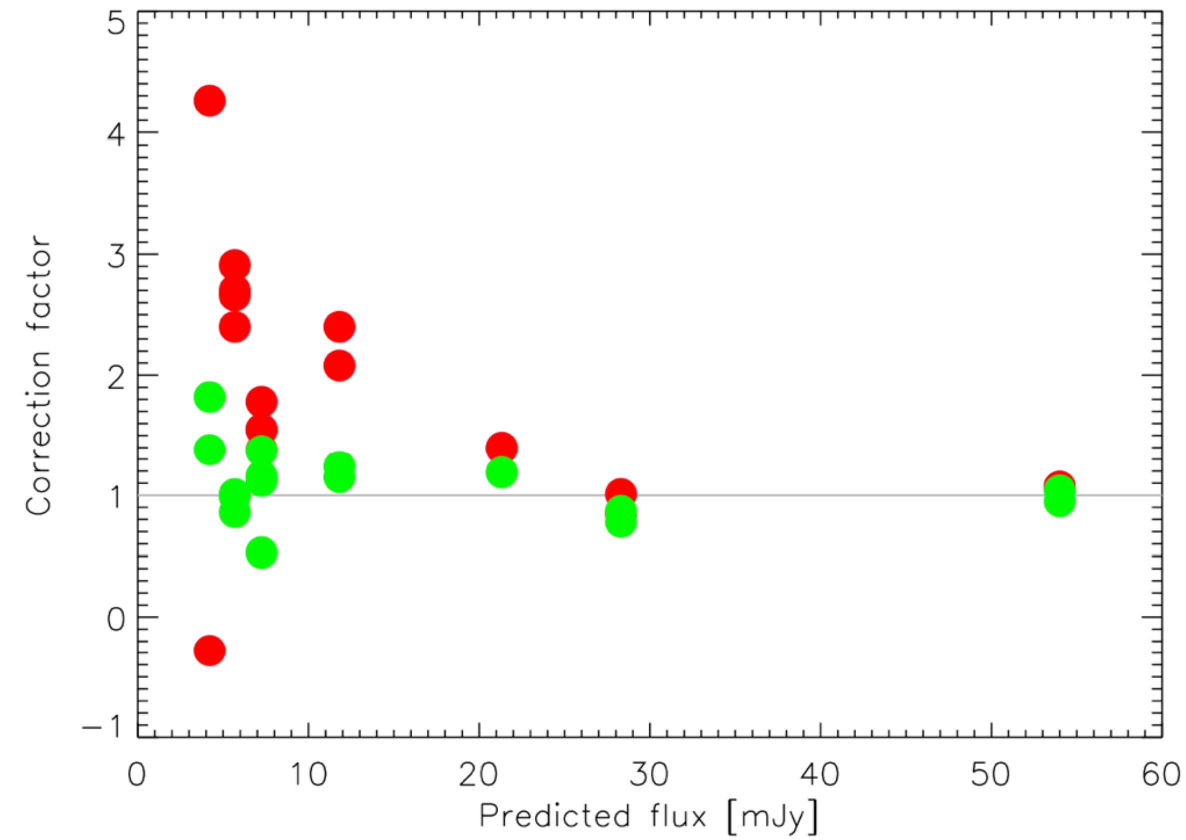
Photometry - regular & boloSource() - 160um

Aperture radius = 10.0″

Name	OBSID	Repetitions	Brightness [mJy]	σ_{IDL}	C_{IDL}	$\sigma_{bS()}$	$C_{bS()}$
HD 15008	1342189130	9	4.21	6.58	-0.05	7.78	1.07
HD 15008	1342189131	9	4.21	7.57	0.93	6.20	1.00
HD 152222	1342240702	6	7.25	0.05	0.90	2.21	0.67
HD 152222	1342240703	6	7.25	5.79	1.37	3.80	1.38
HD 152222	1342191964	6	7.25	3.98	1.26	1.58	1.30
HD 152222	1342191965	6	7.25	4.49	1.41	2.71	1.26
HD 39608	1342198535	10	5.68	4.21	1.87	0.82	1.28
HD 39608	1342198536	10	5.68	5.21	1.90	2.33	1.15
HD 39608	1342198537	35	5.68	6.66	1.61	2.32	1.18
HD 39608	1342198538	35	5.68	5.14	1.99	2.76	1.05
HD 159330	1342213585	6	11.81	3.17	1.26	1.97	1.09
HD 159330	1342213586	6	11.81	4.04	0.72	7.57	0.91
HD 139669	1342191982	6	54.0	5.31	1.01	7.72	1.00
HD 139669	1342191983	6	54.0	7.57	0.97	7.54	0.89
HD 138265	1342188841	21	21.32	8.22	0.83	9.93	1.11
HD 138265	1342188842	21	21.32	7.38	0.81	4.79	1.11
HD 170693	1342243730	20	28.3	6.50	0.85	3.99	0.78
HD 170693	1342243731	20	28.3	5.31	1.01	4.24	0.87

Photometry - regular & boloSource() - 160um

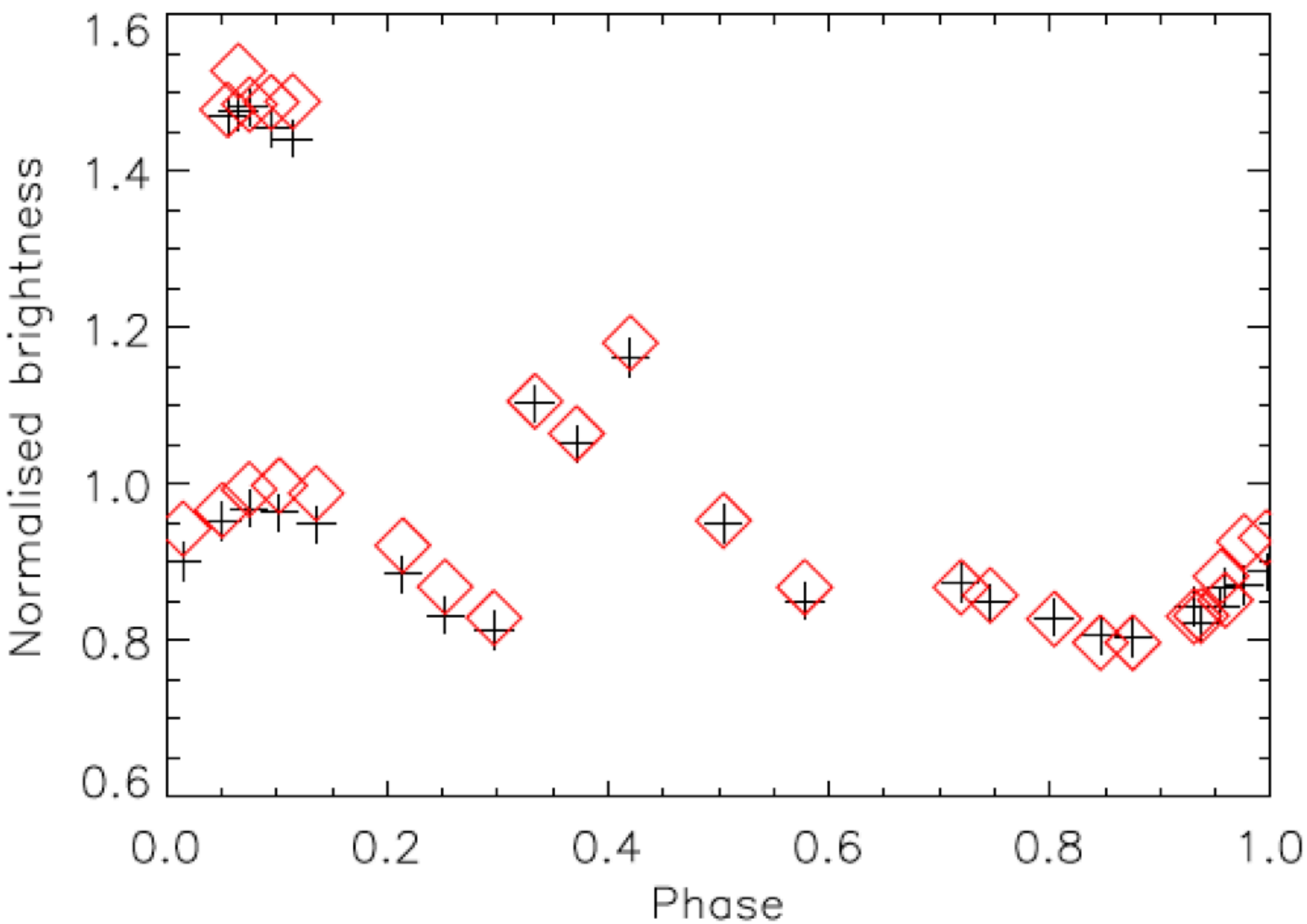




σ				C			
IDL		bS()		IDL		bS()	
Median	AVG	Median	AVG	Median	AVG	Median	AVG
6.50	6.54	4.24	4.81	1.55	1.78	1.12	1.09

Photometry - something more realistic

IC348 (Z. Balog et al.) - with an IR variable star



Average difference:
1.9 %

STDEV of difference:
2.0 %

Installation & Settings

Installation: unpack the script files wherever you want

What you need to run the code:

- HIPE
- PACS L1 cube (frames)
- List of sources (RA, Dec, r)/L2 mask

Installation & Settings

```
# Setup environment and demo data
ver           = "v2.7"
dirHome           = "/Users/user/Astro/HIPErutins/"
dirScripts  = dirHome + "boloSource_"+ver+"/scripts/definitions/"
dirScripts2 = dirHome + "boloSource_"+ver+"/scripts/benchmark/"

# Compile
execfile(dirScripts + "photBaselineEstimator.py")
execfile(dirScripts2 + "photFFT.py")
execfile(dirScripts + "photBoloSourceInterpolator.py")
execfile(dirScripts + "photBoloSourceTools.py")
execfile(dirScripts + "photBoloSource.py")

calTree=getCalTree()

#Define the Level1 frames filename
originalFrames=simpleFitsReader('/Users/user/Astro/obsid_simple_level1Frames.fits')

#Define the mask file name
fileName='/Users/user/Astro/mask.fits'

#Define the output path
outputPath='/Users/user/Astro/BS/'

sourceMask = boloSourceReadMask(fileName)
framesOut  = boloSource(originalFrames, sourceMask = sourceMask, calTree = calTree, verbose = 0)
```

(Near) Future plans

- Execute the algorithm on simulations
- Test the power spectra
- Compact source photometry
    - Comparison with getsources and/or other routines
- Make it available within HIPE
- Extend the code to SPIRE
- Timeline source fitting similar to SPIRE(?)