Julian Merten

SaWLens 1.4: Non-parametric lensing reconstruction

INAF - Osservatorio Astronomico di Bologna
Institut für Theoretische Astrophysik
Zentrum für Astronomie
Universität Heidelberg

December 17^{th}, 2009

Overview

1. The theory behind the algorithm
   JM et al. (2009) arXiv:0806.1967
2. The implementation
3. A very recent example: CL0024+1654

SaWLens 1.4: A practical manual

Julian Merten
ITA Heidelberg
INAF-OA Bologna

November 11, 2009

The manual is available together with the source code:

https://www.ita.uni-heidelberg.de/svn/sawlens

What we want from the method

- Combine multiple observational constraints into a joint reconstruction method in a consistent way.
- The method should be fully non-parametric.
- It should perform well compared to specialised routines in their regimes.
- The demandings regarding CPU-time and resources should stay on an acceptable level.

The basic idea: Bartelmann et al. (1996)

- Divide the reconstruction field into a grid.
- Reconstruction quantity is the lensing potential ψ.
- Find a relation between the observed quantities and ψ.
- Define χ^2-functions on the data grid.

$$\chi^2(\psi) = \chi_1^2(\psi) + \chi_2^2(\psi) + \chi_3^2(\psi) + ...$$

- Minimise the overall χ^2 with respect to the potential.
- Possible constraints: shear, flexion, magnification, critical curve position, mulitple image positions, velocity dispersions, X-ray temperature, SZ, ...

The basic idea: Bartelmann et al. (1996)

- Divide the reconstruction field into a grid.
- Reconstruction quantity is the lensing potential ψ.
- Find a relation between the observed quantities and ψ.
- Define χ^2-functions on the data grid.

$$\chi^2(\psi) = \chi_1^2(\psi) + \chi_2^2(\psi) + \chi_3^2(\psi) + ...$$

- Minimise the overall χ^2 with respect to the potential.
- Possible constraints: shear, flexion, magnification, critical curve position, mulitple image positions, velocity dispersions, X-ray temperature, SZ, ...

The basic idea: Bartelmann et al. (1996)

- Divide the reconstruction field into a grid.
- Reconstruction quantity is the lensing potential ψ.
- Find a relation between the observed quantities and ψ.
- Define χ^2-functions on the data grid.

$$\chi^2(\psi) = \chi_1^2(\psi) + \chi_2^2(\psi) + \chi_3^2(\psi) + \ldots$$

- Minimise the overall χ^2 with respect to the potential.
- Possible constraints: shear, flexion, magnification, critical curve position, mulitple image positions, velocity dispersions, X-ray temperature, SZ, ...

The basic idea: Bartelmann et al. (1996)

- Divide the reconstruction field into a grid.
- Reconstruction quantity is the lensing potential ψ.
- Find a relation between the observed quantities and ψ.
- Define χ^2-functions on the data grid.

$$\chi^2(\psi) = \chi_1^2(\psi) + \chi_2^2(\psi) + \chi_3^2(\psi) + ...$$

- Minimise the overall χ^2 with respect to the potential.
- Possible constraints: shear, flexion, magnification, critical curve position, mulitple image positions, velocity dispersions, X-ray temperature, SZ, ...

The basic idea: Bartelmann et al. (1996)

- Divide the reconstruction field into a grid.
- Reconstruction quantity is the lensing potential ψ.
- Find a relation between the observed quantities and ψ.
- Define χ^2-functions on the data grid.

$$\chi^2(\psi) = \chi_1^2(\psi) + \chi_2^2(\psi) + \chi_3^2(\psi) + ...$$

- Minimise the overall χ^2 with respect to the potential.
- Possible constraints: shear, flexion, magnification, critical curve position, mulitple image positions, velocity dispersions, X-ray temperature, SZ, ...

The basic idea: Bartelmann et al. (1996)

- Divide the reconstruction field into a grid.
- Reconstruction quantity is the lensing potential ψ.
- Find a relation between the observed quantities and ψ.
- Define χ^2-functions on the data grid.

$$\chi^2(\psi) = \chi_1^2(\psi) + \chi_2^2(\psi) + \chi_3^2(\psi) + ...$$

- Minimise the overall χ^2 with respect to the potential.
- Possible constraints: shear, flexion, magnification, critical curve position, mulitple image positions, velocity dispersions, X-ray temperature, SZ, ...

The basic idea: Bartelmann et al. (1996)

- Divide the reconstruction field into a grid.
- Reconstruction quantity is the lensing potential ψ.
- Find a relation between the observed quantities and ψ.
- Define χ^2-functions on the data grid.

$$\chi^2(\psi) = \chi_1^2(\psi) + \chi_2^2(\psi) + \chi_3^2(\psi) + ...$$

- Minimise the overall χ^2 with respect to the potential.
- Possible constraints: shear, flexion, magnification, critical curve position, mulitple image positions, velocity dispersions, X-ray temperature, SZ, ...

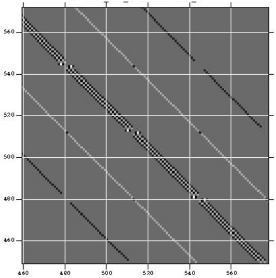
Numerics on a grid: Finite differencing

We need to relate every observed quantity to the ψ-values on the grid, e.g. the convergence:

1	-1	1/2				1/2	-1/2	1/2	+
-1/2							-1		
1/2							1/2		
									1/2
+							1/2	-1	-1/2
			1/3	-1/6	1/3				1/2
			-1/6	-2/3	-1/6				
			1/3	-1/6	1/3				
+						+			+

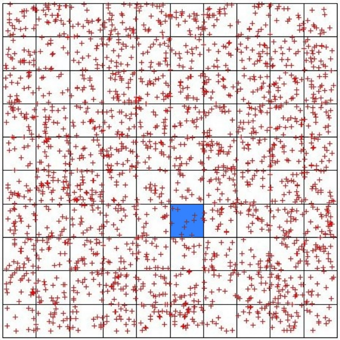
$$\kappa = \frac{1}{2}\left(\psi_{,11} + \psi_{,22}\right)$$

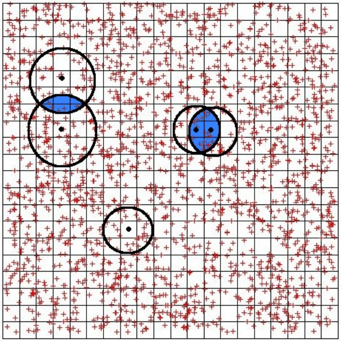
$$\kappa_i = \mathcal{K}_{ij}\psi_j$$



One example: weak lensing

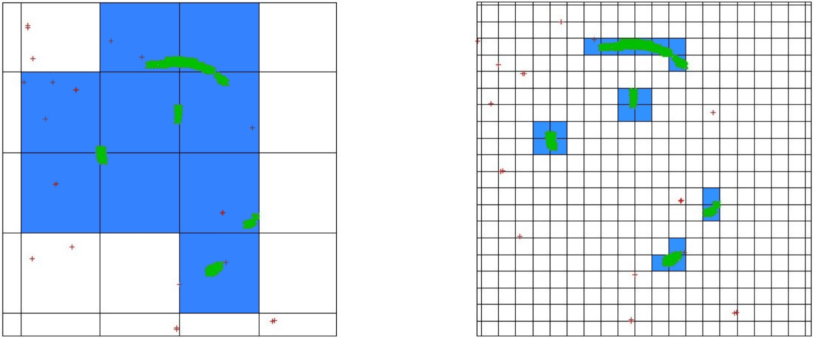
$$\chi^2_w(\psi) = \sum_{i,j} \left(\varepsilon - \frac{Z(z)\gamma(\psi)}{1 - Z(z)\kappa(\psi)} \right)_i \mathcal{C}^{-1}_{ij} \left(\varepsilon - \frac{Z(z)\gamma(\psi)}{1 - Z(z)\kappa(\psi)} \right)_j$$





Another example: strong lensing

$$\chi_s^2(\psi) = \sum_i \frac{\left((1 - Z(z)\kappa(\psi))^2 - (Z(z)\gamma(\psi))^2\right)_i^2}{\sigma_i^2}$$



Obtaining a non-parametric result

$$\frac{\partial \chi^2(\psi)}{\partial \psi_l} \stackrel{!}{=} 0$$

Using $\gamma_i = \mathcal{G}_{ik}\psi_k$, $\kappa_i = \mathcal{K}_{ik}\psi_k$ and $\frac{\partial}{\partial \psi_l}\mathcal{K}_{ik}\psi_k = \mathcal{K}_{ik}\delta_{kl}$

$$\mathcal{B}_{lk}\psi_k = \mathcal{V}_l$$

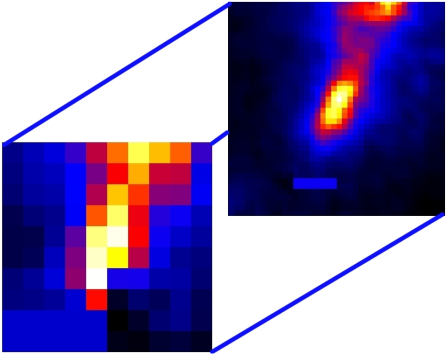
With ψ_k being the final result as the potential on every grid position.

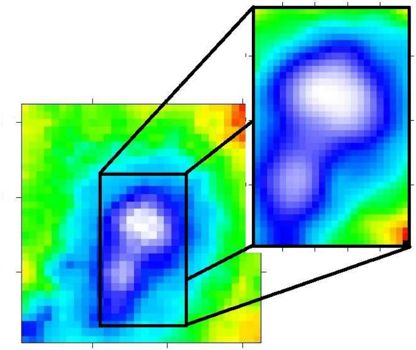
$$\begin{aligned}
\mathcal{B}_{lk} = & \sum_{i,j} \mathcal{F}^1_{ij} Z_i Z_j \\
& \cdot \left[\varepsilon^1_i \varepsilon^1_j \mathcal{K}_{ik}\mathcal{K}_{jl} + \varepsilon^1_i \varepsilon^1_j \mathcal{K}_{jk}\mathcal{K}_{il} + \varepsilon^1_i \mathcal{K}_{ik}\mathcal{G}^1_{jl} + \varepsilon^1_i \mathcal{G}^1_{jk}\mathcal{K}_{il} \right. \\
& \left. + \varepsilon^1_j \mathcal{K}_{jk}\mathcal{G}^1_{il} + \varepsilon^1_j \mathcal{G}^1_{ik}\mathcal{K}_{jl} + \mathcal{G}^1_{ik}\mathcal{G}^1_{jl} + \mathcal{G}^1_{jk}\mathcal{G}^1_{il}\right] \\
& + \sum_{i,j} \mathcal{F}^2_{ij} Z_i Z_j \\
& \cdot \left[\varepsilon^2_i \varepsilon^2_j \mathcal{K}_{ik}\mathcal{K}_{jl} + \varepsilon^2_i \varepsilon^2_j \mathcal{K}_{jk}\mathcal{K}_{il} + \varepsilon^2_i \mathcal{K}_{ik}\mathcal{G}^2_{jl} + \varepsilon^2_i \mathcal{G}^2_{jk}\mathcal{K}_{il} \right. \\
& \left. + \varepsilon^2_j \mathcal{K}_{jk}\mathcal{G}^2_{il} + \varepsilon^2_j \mathcal{G}^2_{ik}\mathcal{K}_{jl} + \mathcal{G}^2_{ik}\mathcal{G}^2_{jl} + \mathcal{G}^2_{jk}\mathcal{G}^2_{il}\right] \\
& + \sum_m \frac{4(\det \mathcal{A})_m}{\sigma^2_m} Z^2_m \\
& \cdot \left[\mathcal{K}_{mk}\mathcal{K}_{ml} - \mathcal{G}^1_{mk}\mathcal{G}^1_{ml} - \mathcal{G}^2_{mk}\mathcal{G}^2_{ml}\right] + \sum_n \eta_n \mathcal{K}_{nk}\mathcal{K}_{nl}
\end{aligned} \tag{A.17}$$

and the result vector

$$\begin{aligned}
\mathcal{V}_l = & \sum_{i,j} \mathcal{F}^1_{ij}\left[\varepsilon^1_i \varepsilon^1_j \mathcal{K}_{jl} + \varepsilon^1_i Z_j \mathcal{G}^1_{jl} + \varepsilon^1_i \varepsilon^1_j Z_i \mathcal{K}_{il} + \varepsilon^1_j Z_i \mathcal{G}^1_{il}\right] \\
& + \sum_{i,j} \mathcal{F}^2_{ij}\left[\varepsilon^2_i \varepsilon^2_j \mathcal{K}_{jl} + \varepsilon^2_i Z_j \mathcal{G}^2_{jl} + \varepsilon^2_i \varepsilon^2_j Z_i \mathcal{K}_{il} + \varepsilon^2_j Z_i \mathcal{G}^2_{il}\right] \\
& + \sum_m \frac{4(\det \mathcal{A})_m}{\sigma^2_m} Z_m \mathcal{K}_{ml} + \sum_n \eta_n \kappa^b_n \mathcal{K}_{nl},
\end{aligned} \tag{A.18}$$

A convenient trick: Iterations





SaWLens 1.4

General info

- Written in C++
- $\sim$ 12000 lines
- MPI-implementation
- CUDA on the way
- 3 independent steps

Requirements

- At least $\sim$ 8 cores
- $\sim$ 1 GB of main memory per core
- negligible disk space per reconstruction: $<$ 6 GB

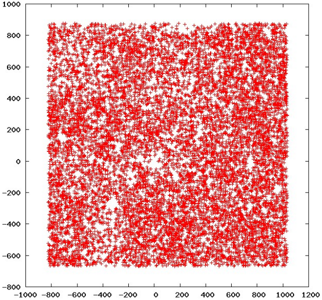
Features

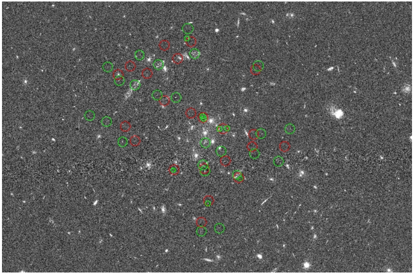
- WL: Ellipticity, Flexion
- SL: Ccurve, Msystems
- Field masking

Dependencies

- GSL
- MPI
- ATLAS
- LAPACK
- CFITSIO
- CCfits
- LibAstroC++

CL0024: input catalogues





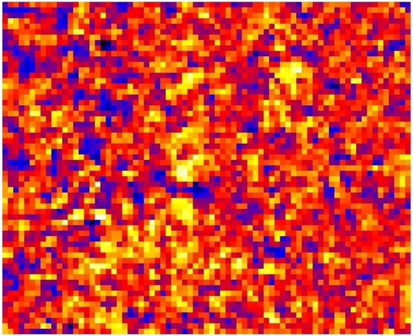
Ellipticity catalogue

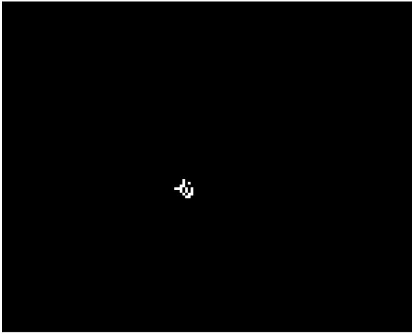
x-pos	y-pos	γ_1	γ_2	weight

Critical curve catalogue

x-pos	y-pos	redshift

Step1: Data grids (Runtime ∼ 1hr)





fv: Summary of ellip_75.fits in /home/jmerten/files/talks/roma_clustermeeting_171209/

File Edit Tools Help

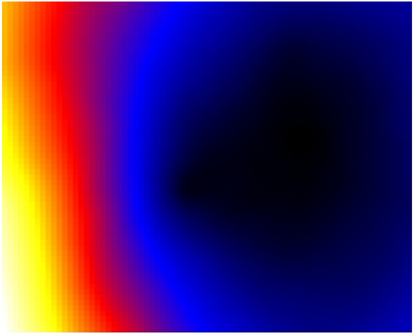
Index	Extension	Type	Dimension	View
0	Primary	Image	75 X 61	Header Image Table
1	mean_ellip2	Image	75 X 61	Header Image Table
2	ellip1_sd	Image	75 X 61	Header Image Table
3	ellip2_sd	Image	75 X 61	Header Image Table
4	overlap	Image	4575 X 4575	Header Image Table
5	ellip1_covariance	Image	4575 X 4575	Header Image Table
6	ellip2_covariance	Image	4575 X 4575	Header Image Table
7	field_mask	Image	75 X 61	Header Image Table

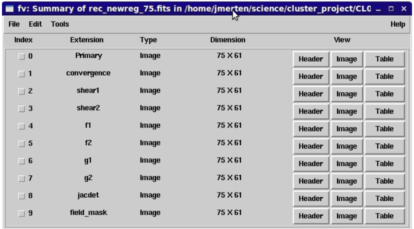
fv: Summary of ccurve_150.fits in /home/jmerten/science/cluster_project/CL0024/elli

File Edit Tools Help

Index	Extension	Type	Dimension	View
0	Primary	Image	150 X 121	Header Image Table
1	pixel_relevance	Image	150 X 121	Header Image Table
2	error	Image	150 X 121	Header Image Table
3	redshift_info	Image	150 X 121	Header Image Table
4	field_mask	Image	150 X 121	Header Image Table

Step2: Joint reconstruction *(Runtime ~ 3hrs)*

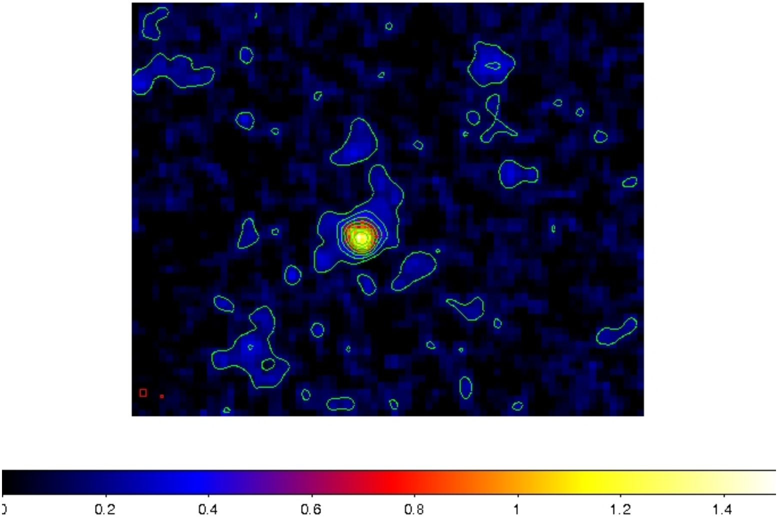



fv: Summary of rec_newreg_75.fits in /home/jmerten/science/cluster_project/CL0

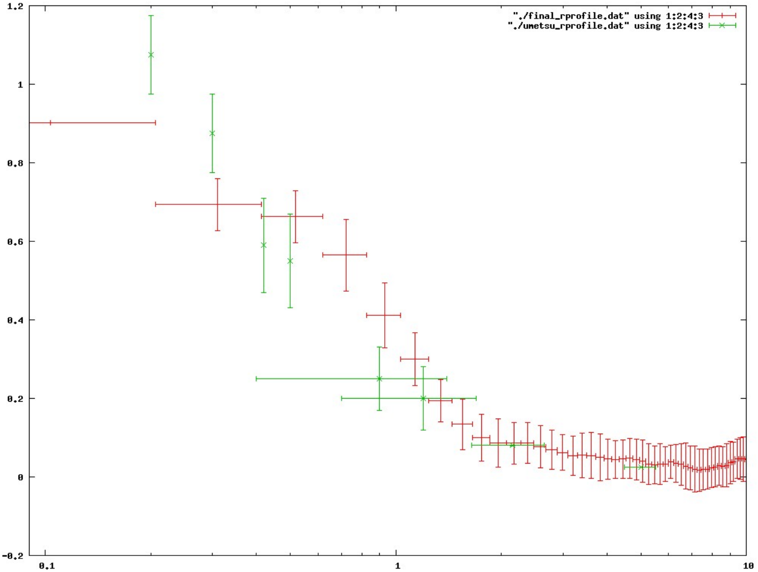
File Edit Tools Help

Index	Extension	Type	Dimension	View		
0	Primary	Image	75 X 61	Header	Image	Table
1	convergence	Image	75 X 61	Header	Image	Table
2	shear1	Image	75 X 61	Header	Image	Table
3	shear2	Image	75 X 61	Header	Image	Table
4	f1	Image	75 X 61	Header	Image	Table
5	f2	Image	75 X 61	Header	Image	Table
6	g1	Image	75 X 61	Header	Image	Table
7	g2	Image	75 X 61	Header	Image	Table
8	jacdet	Image	75 X 61	Header	Image	Table
9	field_mask	Image	75 X 61	Header	Image	Table

Step3 (optional): Highly resolved core (Runtime ~ 1min)



Step3 (optional): Highly resolved core (Runtime ∼ 1min)



Work in progress and what is missing

1. Physics
   - Improved usage of strong-lensing constraints
   - Flexion implementation has to be tested carefully
   - Inclusion of cluster-member dynamics
   - Incorporation of parametric strong-lensing models
   - X-Ray constraints
2. Numerics
   - `CUDA`-implementation
   - Documentation
   - User interface