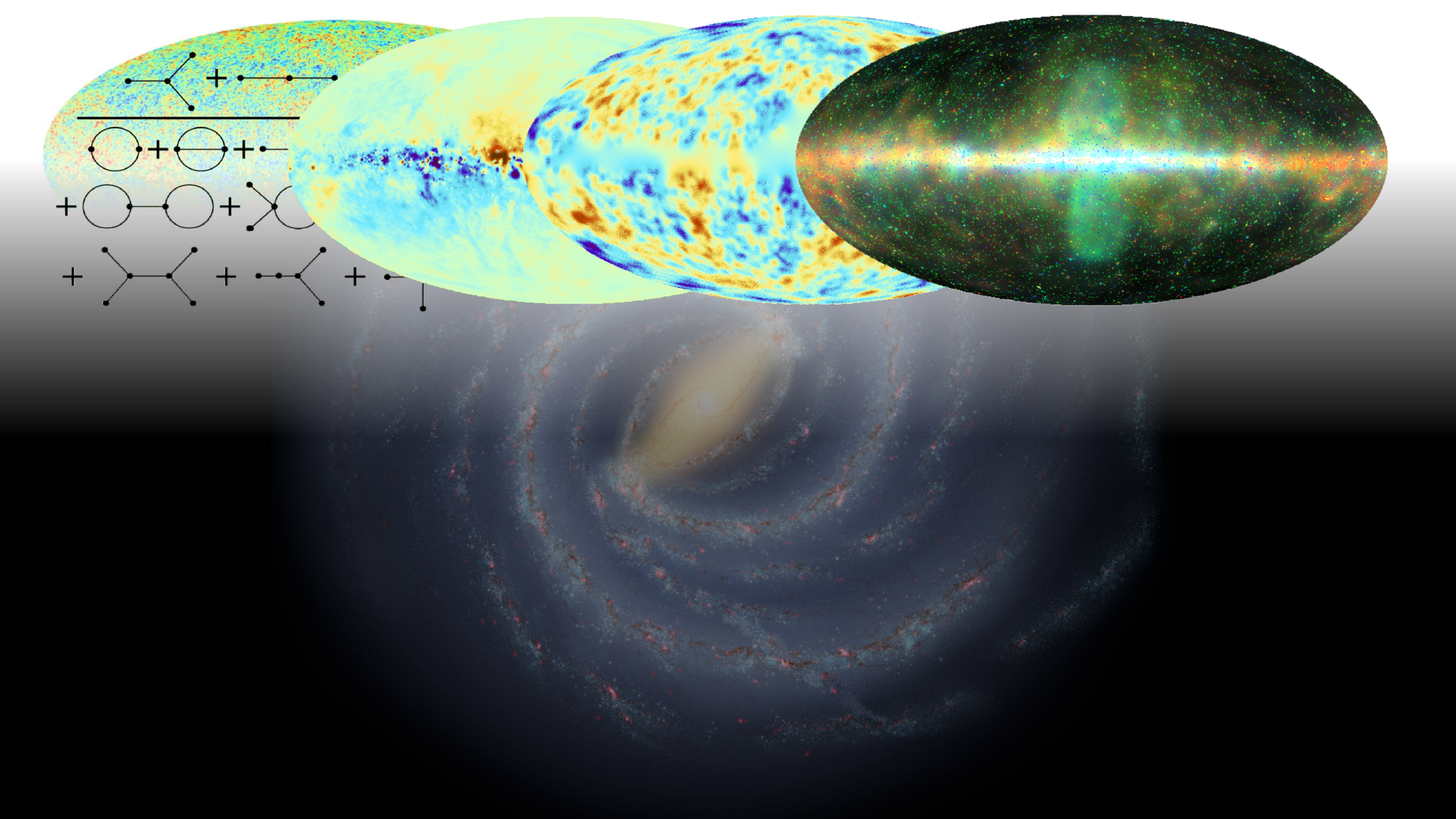
Information Field Theory

+ Numerical Information Field Theory

Torsten Enßlin
Philipp Arras
MPI for Astrophysics

IFT Team: Philipp Arras, Michael Bell, Vanessa Böhm, Sebastian Dorn, Martin Dupont, Mona Frommert, Philipp Frank, Mahsa Ghaempanah, Maksim Greiner, Philipp Haim, Sebastian Hutschenreuter, Henrik Junklewitz, Francisco-Shu Kitaura, Jakob Knollmüller, Christoph Lienhard, Reimar Leike, Ancla Müller, Johannes Oberpriller, Niels Oppermann, Natalia Porquerese, Daniel Pumpe, Tiago Ramalho, Martin Reinecke, Julia Stadler, Marco Selig, Theo Steininger, Valentina Vacca, Cornelius Weig, Margret Westerkamp, & many more



Galactic Tomography

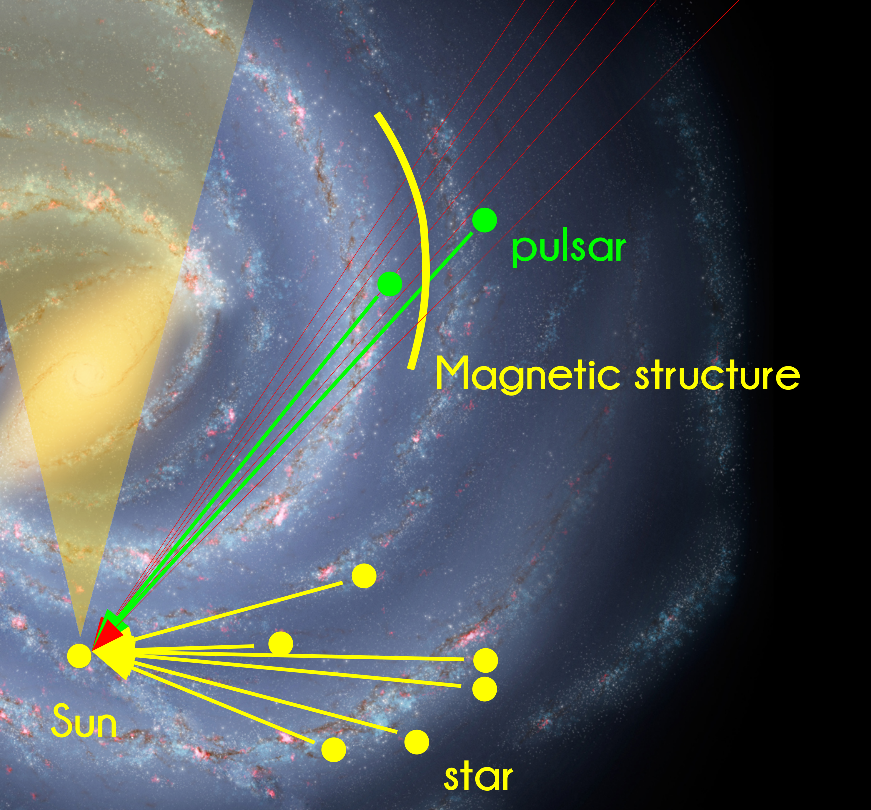
Pulsars:
Dispersion Measure → electron density
Rotation Measure → magnetic field x el. dens.
Scintillation Measure → el. dens. x turbulence

Extragalactic sources:
Rotation Measure → magnetic field x el. dens.
Ultra High Energy Cosmic Rays → mag. fields

Stars:
Dust reddening → dust density & properties
Positions → stellar density & radiation field
Kinematics → gravitational potential

Emission Processes:
Dust emission → dust density & radiation field
Synchrotron → relativistic el. x mag. Fields
Bremsstrahlung → thermal, rel. el. x gas density
Inverse Compton → rel. el. x radiation field
Hadronic interactions → rel. nuclei x gas density
Lines (21 cm, CO, ...) → gas density & kinematics

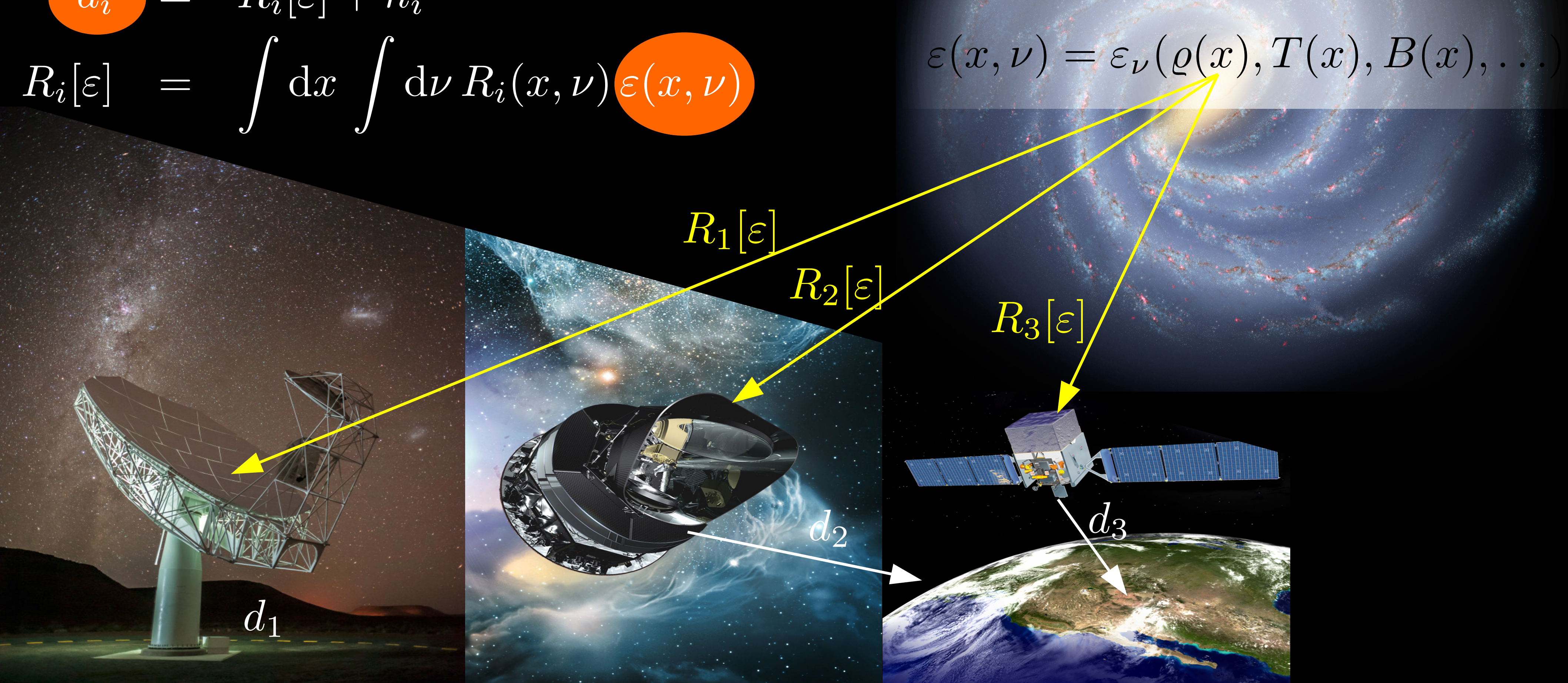
Other information sources:
Correlation structures (auto- & cross-correlations)
Approximate symmetries
Physical laws
Empirical laws, ...



Data Fusion

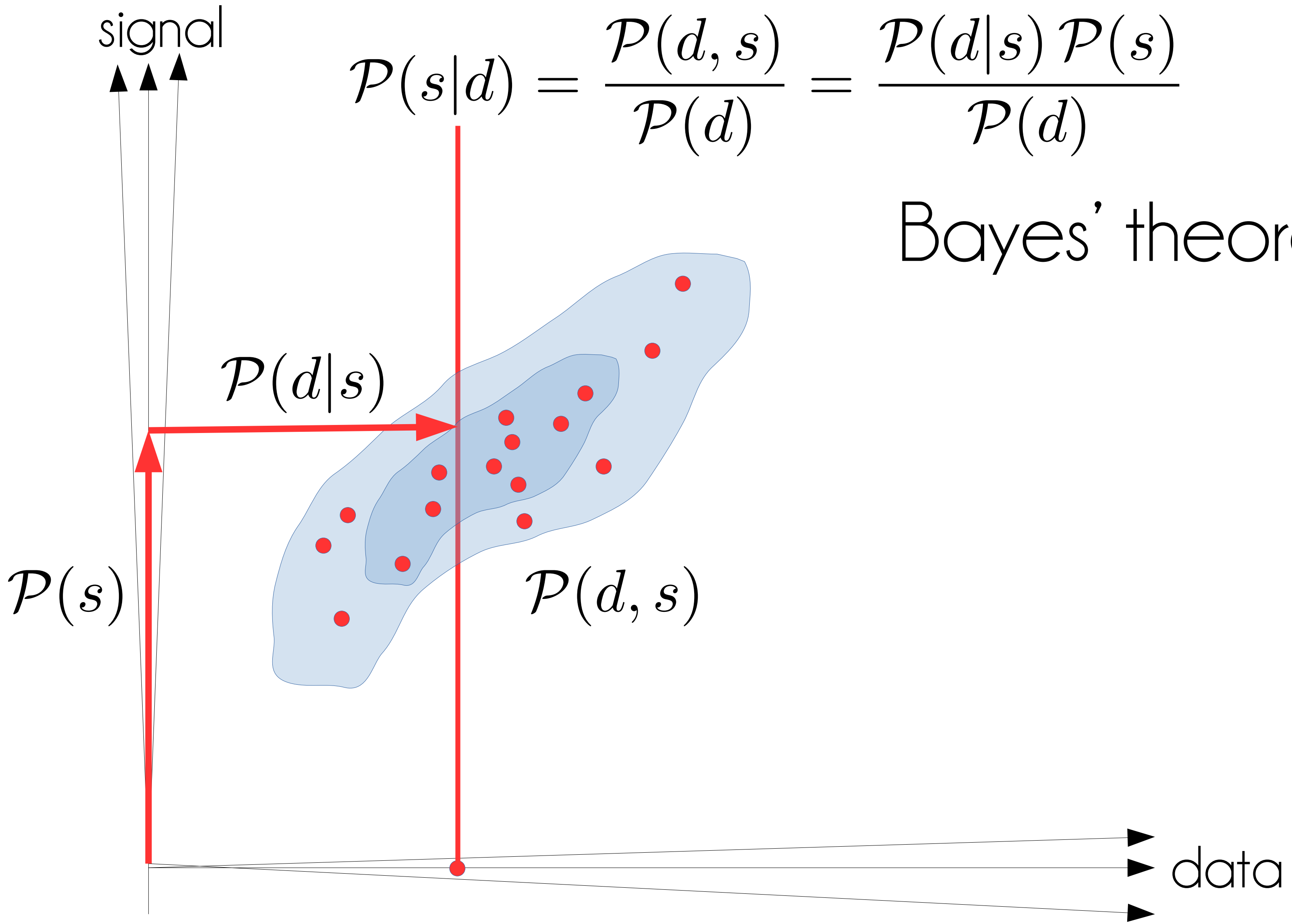
$$d_i = R_i[\varepsilon] + n_i$$

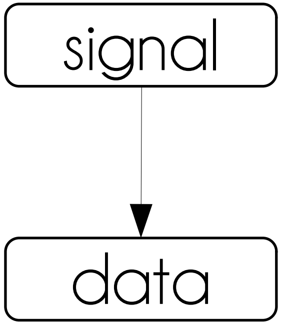
$$R_i[\varepsilon] = \int \mathrm{d}x \int \mathrm{d}\nu \, R_i(x,\nu)\, \varepsilon(x,\nu)$$



$$\mathcal{P}(s|d) = \frac{\mathcal{P}(d,s)}{\mathcal{P}(d)} = \frac{\mathcal{P}(d|s)\,\mathcal{P}(s)}{\mathcal{P}(d)}$$

Bayes' theorem





Information theory

$$\begin{aligned}
\mathcal{P}(s|d) &= \frac{\mathcal{P}(d,s)}{\mathcal{P}(d)} = \frac{e^{-\mathcal{H}(d,s)}}{\mathcal{Z}(d)} \\
\mathcal{H}(d,s) &= -\log \mathcal{P}(d,s) \\
\mathcal{Z}(d) &= \mathcal{P}(d) \\
&= \int \mathcal{D}s\, \mathcal{P}(d,s) \\
\mathcal{P}(d,s) &= \mathcal{P}(d|s)\, \mathcal{P}(s) \\
\mathcal{H}(d,s) &= \mathcal{H}(d|s) + \mathcal{H}(s)
\end{aligned}$$

Information

is additive

`metric` `regularization`

Information theory

$$\mathcal{P}(s|d) = \frac{\mathcal{P}(d,s)}{\mathcal{P}(d)} = \frac{e^{-\mathcal{H}(d,s)}}{\mathcal{Z}(d)}$$

$$\mathcal{H}(d,s) = -\log \mathcal{P}(d,s)$$

Information

$$\mathcal{Z}(d) = \mathcal{P}(d) = \int \mathcal{D}s\, \mathcal{P}(d,s)$$

$$\mathcal{P}(d,s) = \mathcal{P}(d|s)\, \mathcal{P}(s)$$

$$\mathcal{H}(d_1,d_2,s) = \mathcal{H}(d_1|s) + \mathcal{H}(d_2|s) + \mathcal{H}(s)$$

is additive

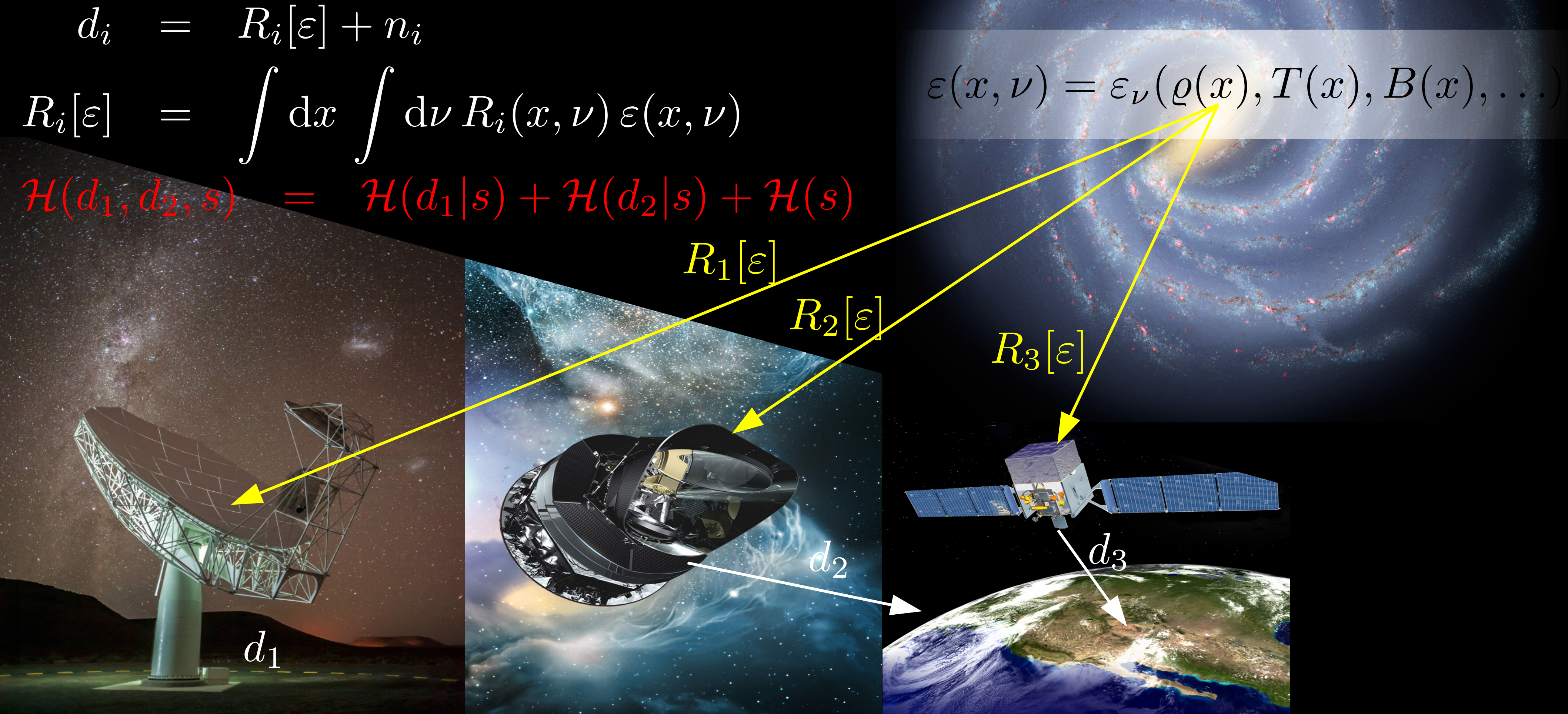
Data Fusion

$$d_i = R_i[\varepsilon] + n_i$$

$$R_i[\varepsilon] = \int \mathrm{d}x \int \mathrm{d}\nu \, R_i(x,\nu)\, \varepsilon(x,\nu)$$

$$\mathcal{H}(d_1, d_2, s) = \mathcal{H}(d_1|s) + \mathcal{H}(d_2|s) + \mathcal{H}(s)$$

$$\varepsilon(x,\nu) = \varepsilon_\nu(\varrho(x), T(x), B(x), \ldots)$$



Probability & Information

$$\mathcal{P}(s) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{s^2}{2\sigma^2}}$$

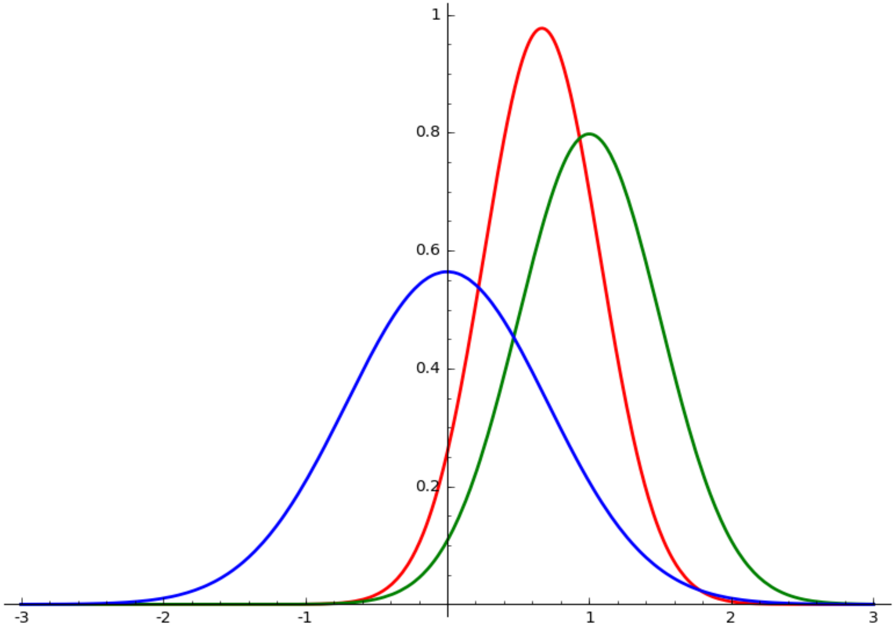
$$\mathcal{P}(d|s) \propto e^{-\frac{(s-d)^2}{2\sigma'^2}}$$

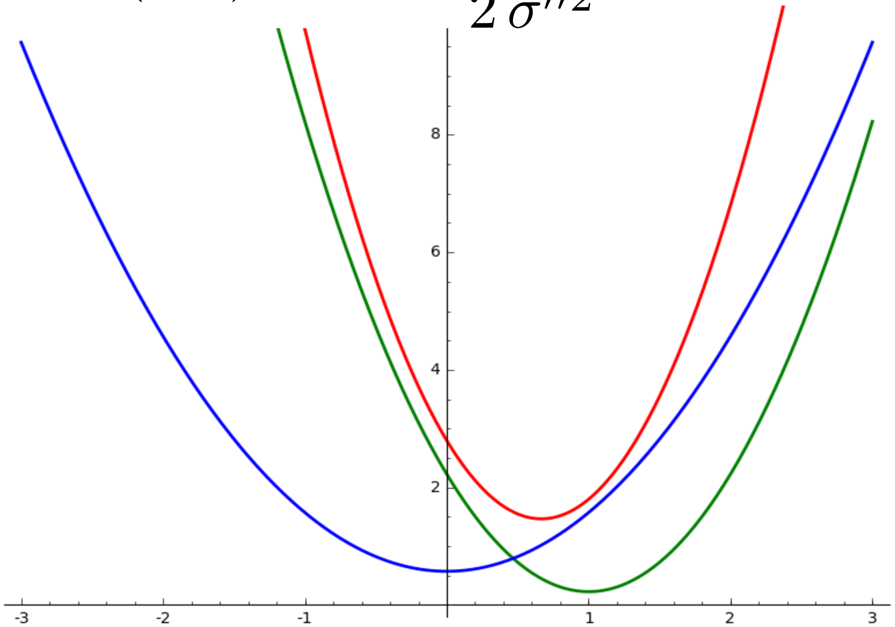
$$\mathcal{P}(s|d) \propto e^{-\frac{(s-m)^2}{2\sigma''^2}}$$

$$\mathcal{H}(s) \,\widehat{=}\, \frac{s^2}{2\,\sigma^2}$$

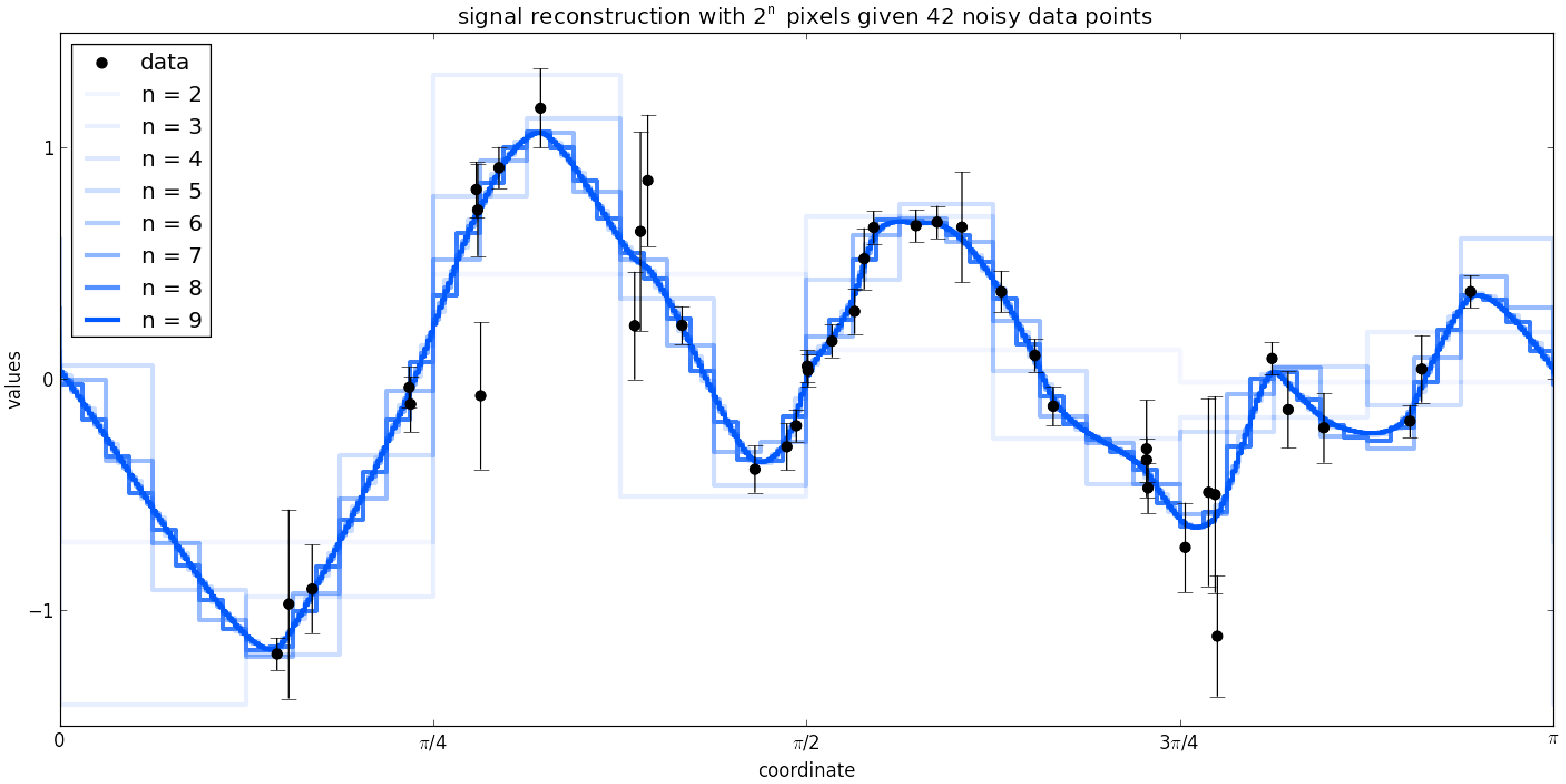
$$\mathcal{H}(d|s) \,\widehat{=}\, \frac{(s-d)^2}{2\,\sigma'^2}$$

$$\mathcal{H}(d,s) \,\widehat{=}\, \frac{(s-m)^2}{2\,\sigma''^2}$$






Correlations



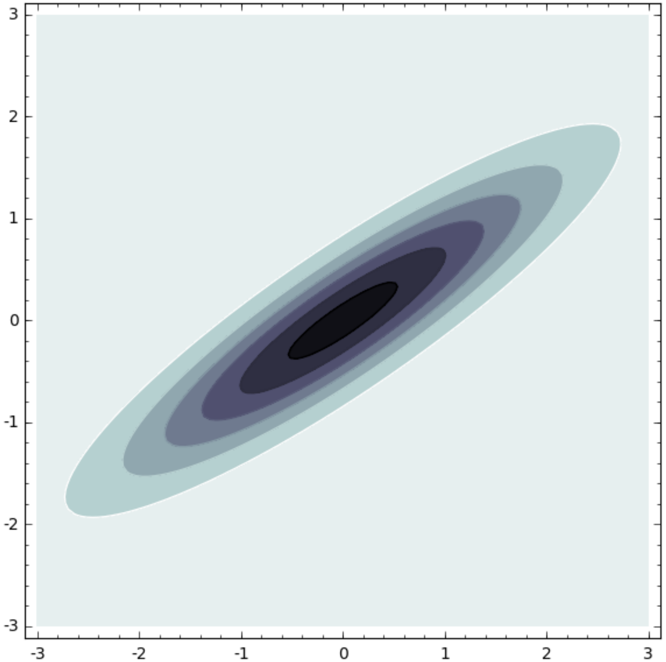
signal reconstruction with 2^n pixels given 42 noisy data points
data
n = 2
n = 3
n = 4
n = 5
n = 6
n = 7
n = 8
n = 9
values
1
0
−1
0
π/4
π/2
3π/4
π
coordinate

Correlations

$\mathcal{P}(s)$

$$s = \begin{pmatrix} s_1 \\ s_2 \end{pmatrix}$$

$d = s_1 + n$



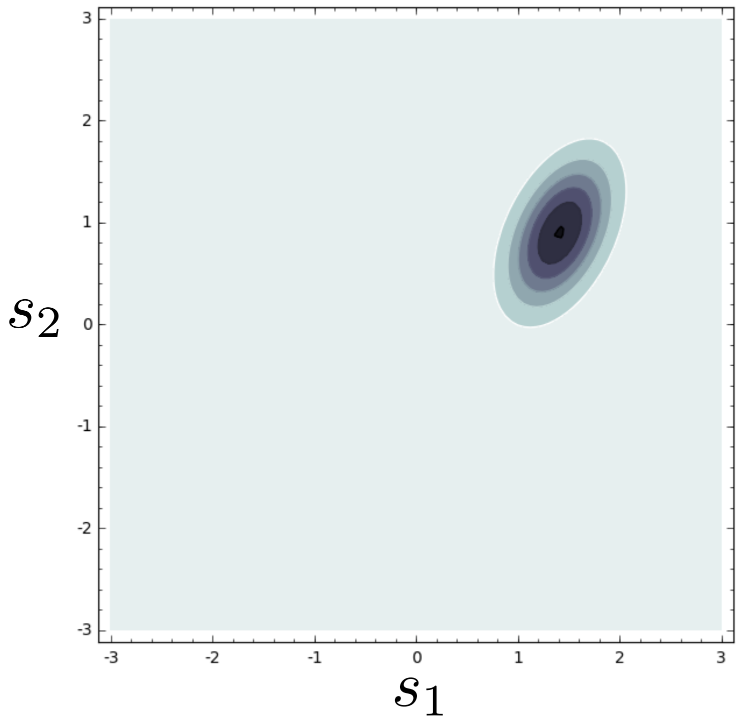
s_2

s_1

Correlations

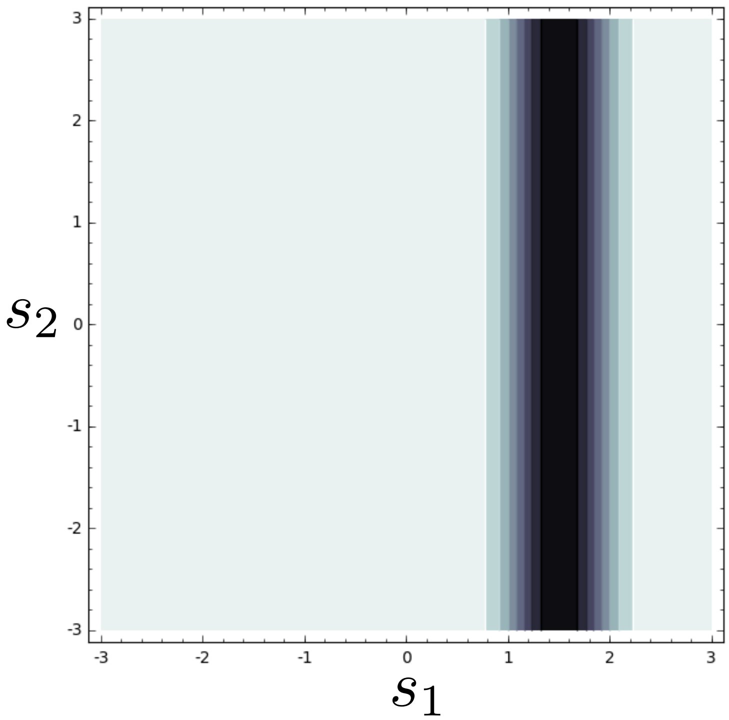
$$s = \begin{pmatrix} s_1 \\ s_2 \end{pmatrix}$$

$\mathcal{P}(s|d)$



$\mathcal{P}(d|s)$

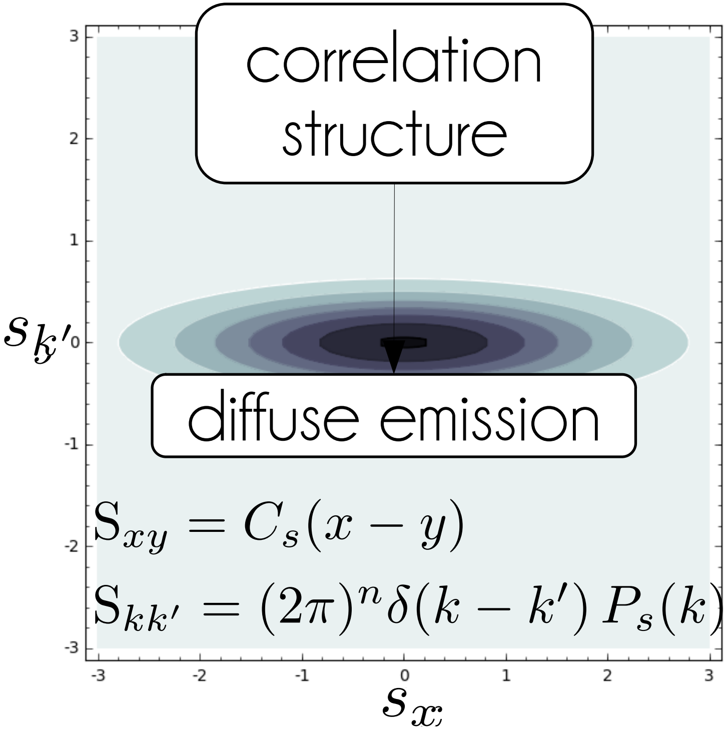
$d = s_1 + n$



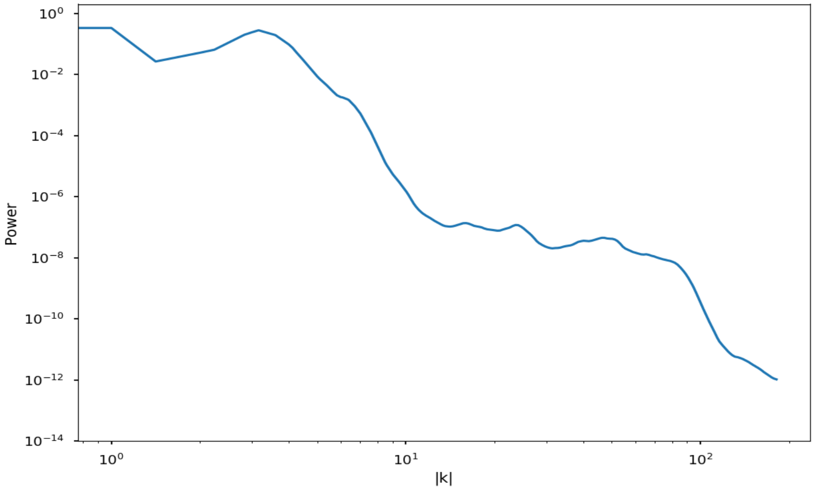
Correlations

$$\mathcal{P}(s)$$

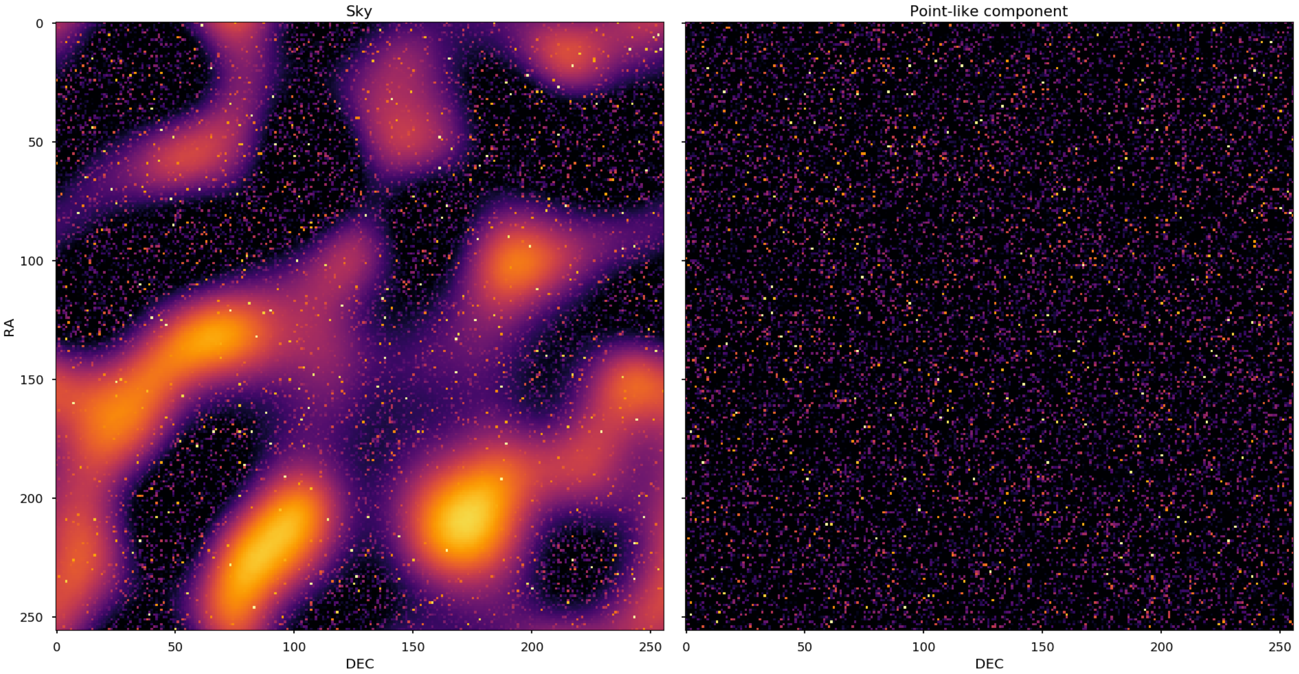
$$s = \begin{pmatrix} s_1 \\ s_2 \end{pmatrix}$$



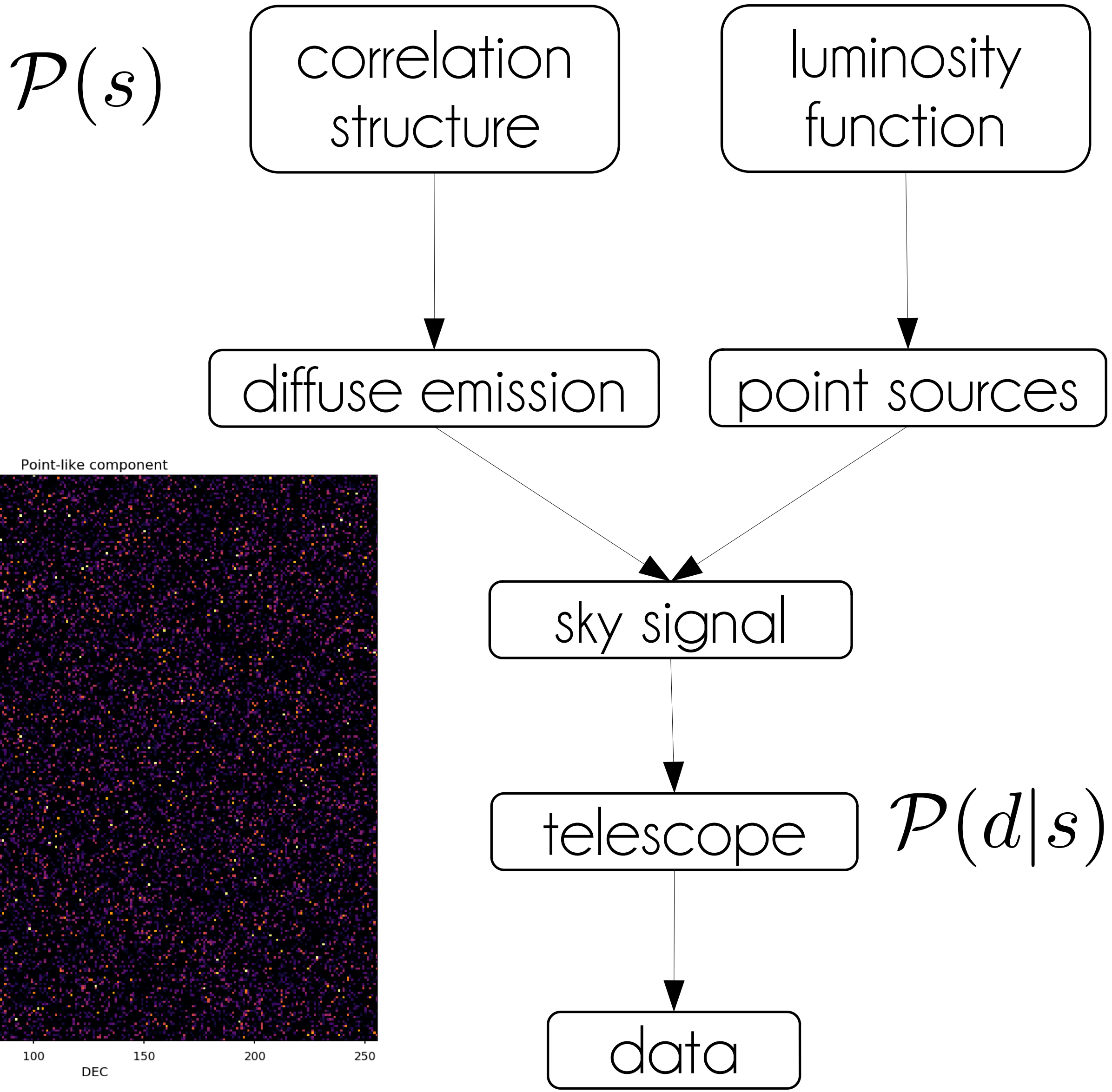
$$\begin{aligned}
\mathcal{P}(s) &= \mathcal{G}(s, S) \\
&= \frac{1}{\sqrt{|2\pi S|}} \exp\left(-\frac{1}{2} s^\dagger S^{-1} s\right) \\
S &= \begin{pmatrix} \langle s_1 s_1 \rangle & \langle s_1 s_2 \rangle \\ \langle s_2 s_1 \rangle & \langle s_2 s_2 \rangle \end{pmatrix} \quad \text{2-dim.} \\
S_{ij} &= \langle s_i s_j \rangle \quad n\text{-dim.} \\
S_{xy} &= \langle s_x s_y \rangle,\ x \in \mathbb{R}^n \quad \infty\text{-dim.}
\end{aligned}$$



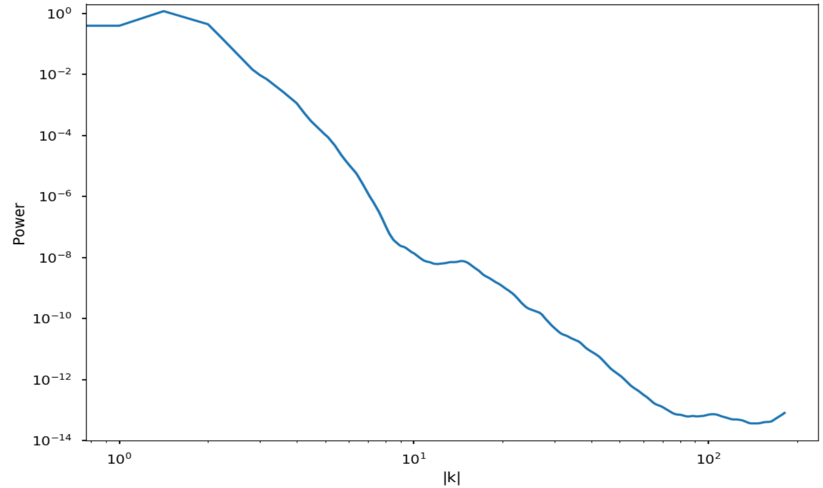
Power
|k|

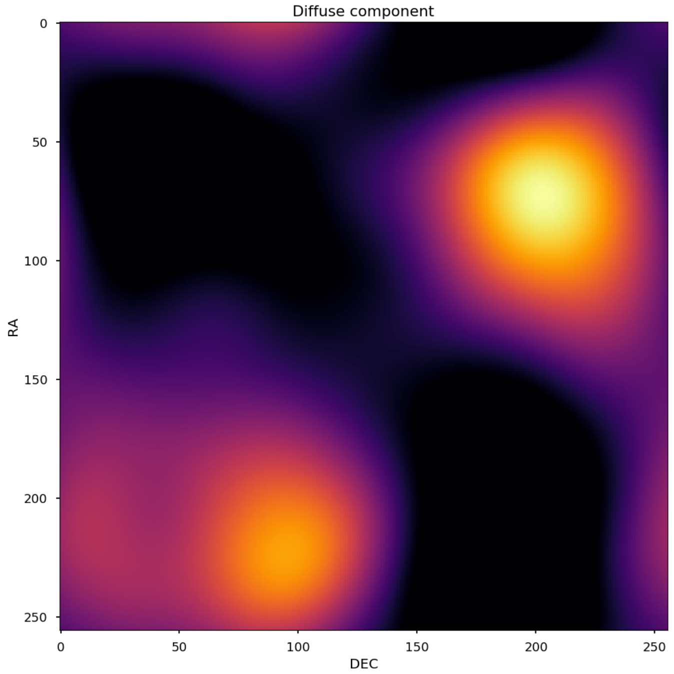


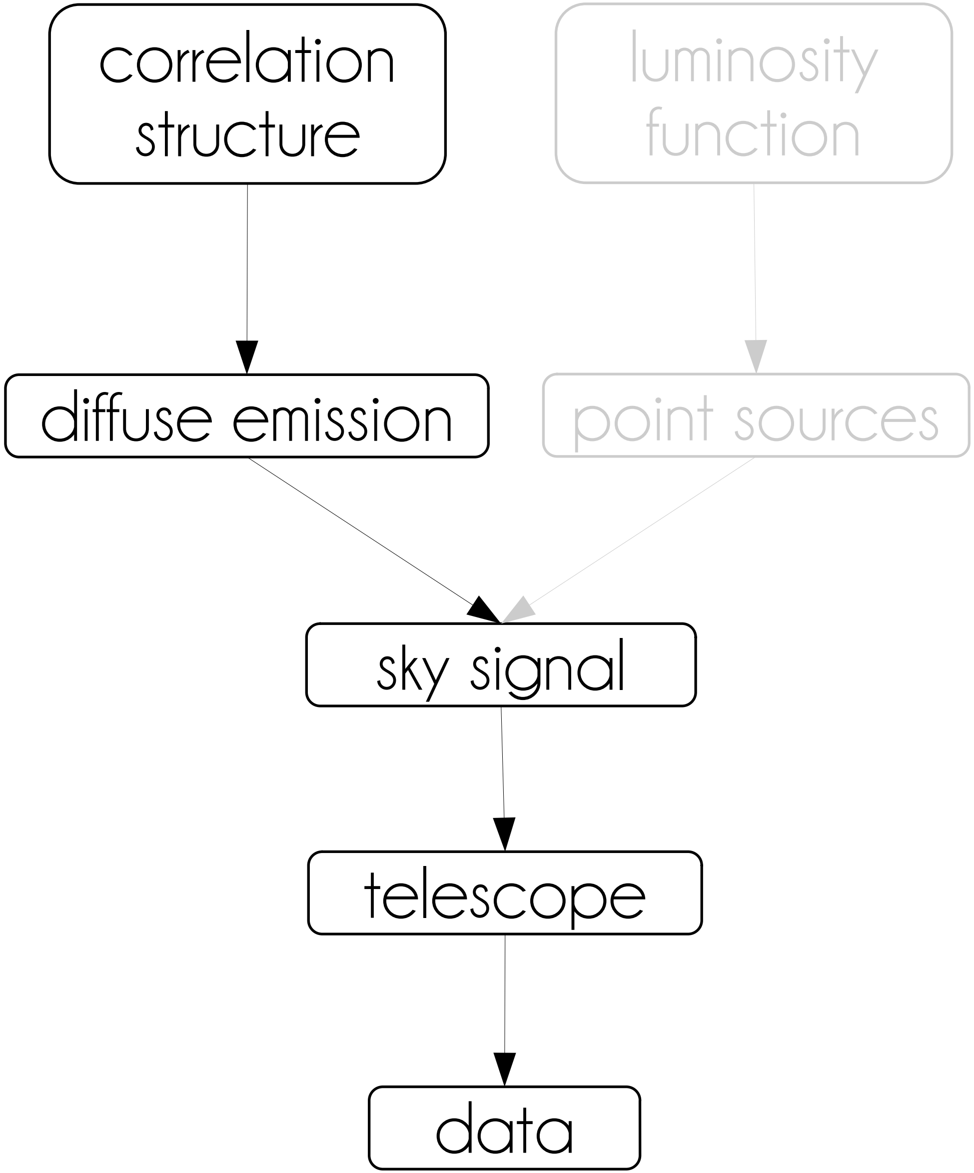
Sky
Point-like component
RA
DEC
DEC



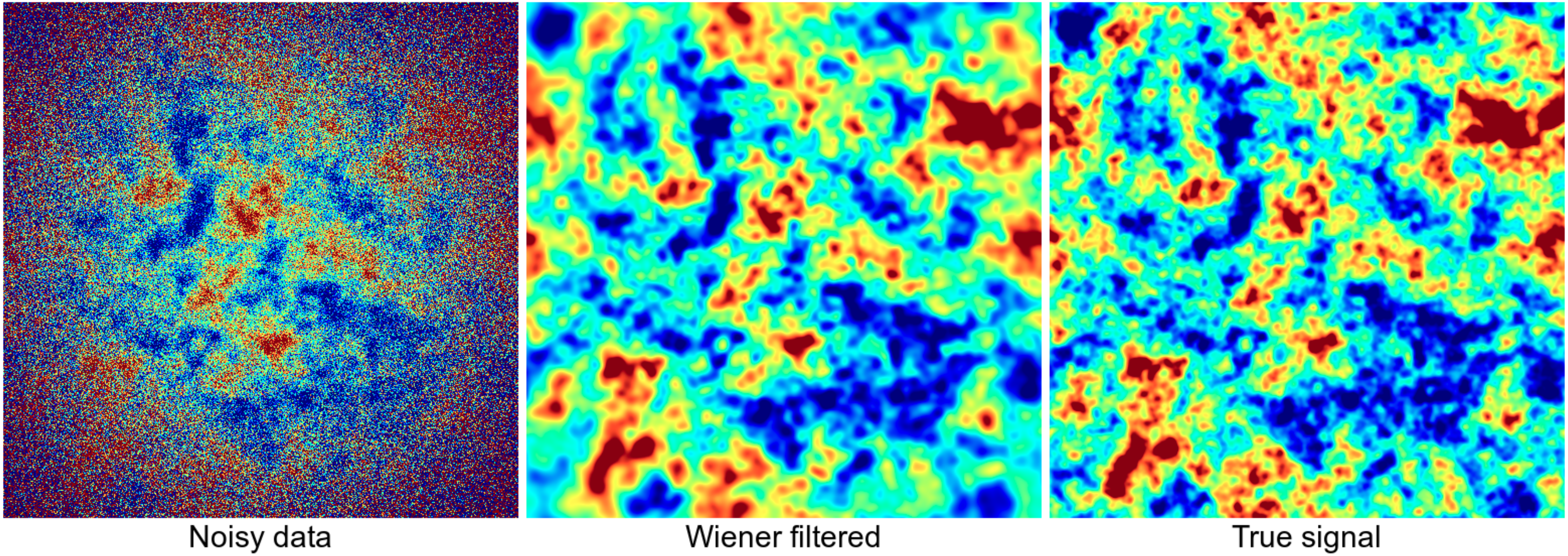
$\mathcal{P}(s)$
correlation structure
luminosity function
diffuse emission
point sources
sky signal
telescope
$\mathcal{P}(d|s)$
data


Power
|k|


Diffuse component
RA
DEC


correlation structure
luminosity function
diffuse emission
point sources
sky signal
telescope
data

Wiener Filter



https://en.wikipedia.org/wiki/Generalized_Wiener_filter

$$
\begin{aligned}
d &= R\,s + n && \text{data}\\
\mathcal{P}(d,s|R,S,N) &= \mathcal{G}(s,S)\,\mathcal{G}(d - R\,s, N) && \text{prior \& likelihood}\\
\mathcal{P}(s|d,R,S,N) &= \mathcal{G}(s-m, D) && \text{posterior}\\
\mathcal{H}(d,s|R,S,N) &\widehat{=} \frac{1}{2}\,s^\dagger S^{-1} s + \frac{1}{2}\,(d - R\,s)^\dagger N^{-1} (d - R\,s)\\
&\widehat{=} \frac{1}{2}\,[s^\dagger \underbrace{\left(S^{-1} + R^\dagger N^{-1} R\right)}_{=D^{-1}} s + s\,\underbrace{R^\dagger N^{-1} d}_{=j} + \underbrace{d^\dagger N^{-1} R}_{=j^\dagger}\; s]\\
&= \frac{1}{2}\,[s^\dagger D^{-1} s + s^\dagger j + j^\dagger s]\\
&= \frac{1}{2}\,[s^\dagger D^{-1} s + s^\dagger D^{-1}\,\underbrace{D\,j}_{=m} + j^\dagger D\,D^{-1} s]\\
&\widehat{=} \frac{1}{2}\,[(s-m)^\dagger D^{-1} (s-m)]
\end{aligned}
$$

$$d = R\,s + n$$
data

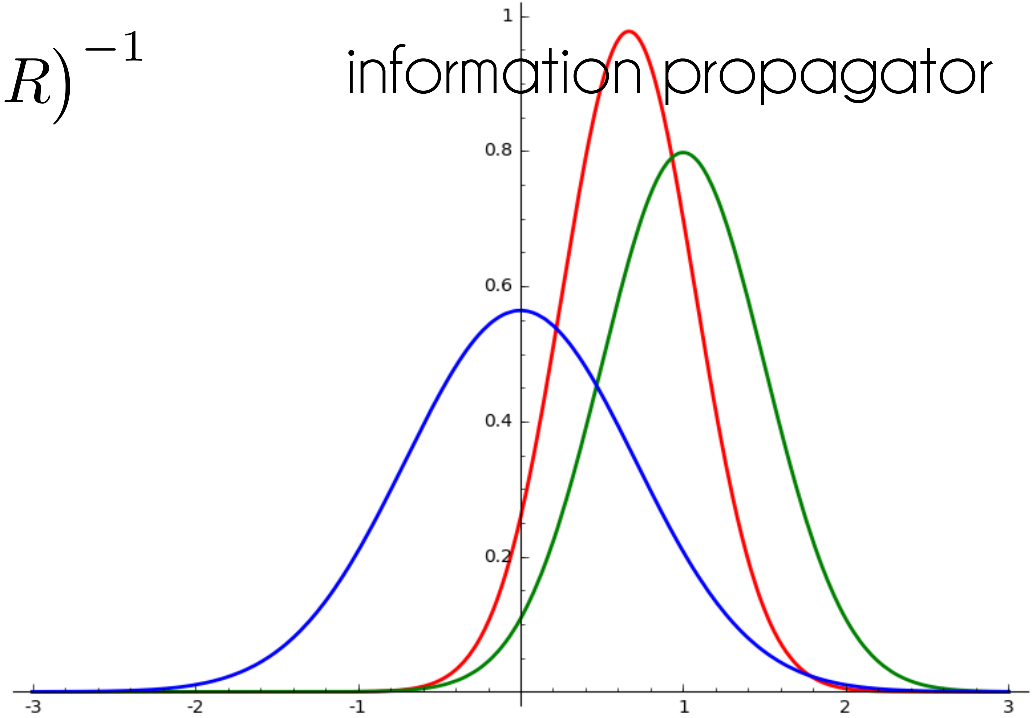
$$\mathcal{P}(d, s|R, S, N) = \mathcal{G}(s, S)\,\mathcal{G}(d - R\,s, N)$$
prior & likelihood

$$\mathcal{P}(s|d, R, S, N) = \mathcal{G}(s - m, D)$$
posterior

$$m = D\,j$$
posterior mean

$$j = R^{\dagger}N^{-1}d$$
information source

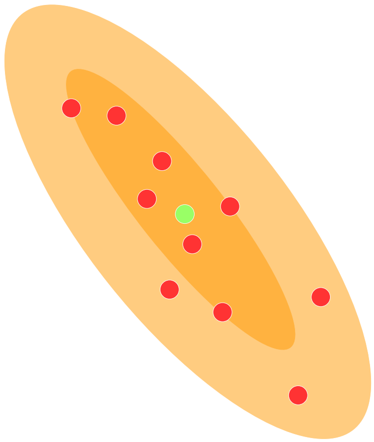
$$D = \left(S^{-1} + R^{\dagger}N^{-1}R\right)^{-1}$$
information propagator



Wiener Filter

$$
\begin{aligned}
\mathcal{P}(s,d) &= \mathcal{G}(s-{\color{red}m}, D) \\
{\color{red}m} &= D\,j \\
j &= R^\dagger N^{-1} d \\
D^{-1} &= S^{-1} + R^\dagger N^{-1} R \\
D^{-1}{\color{red}m} &= j \qquad \leftarrow \text{solve with Conjugate Gradient} \\
S &= F^{-1}\widehat{P_s}\,F \\
S^{-1} &= F^{-1}\widehat{1/P_s}\,F
\end{aligned}
$$

Wiener Filter Samples



$$
\begin{array}{rcl}
s' & \hookleftarrow & \mathcal{G}(s' - m, D) \\
s^* & \hookleftarrow & \mathcal{G}(s^*, S) \\
\xi & \hookleftarrow & \mathcal{G}(\xi, \mathbb{1}) \\
s^* & = & F^{-1}\widehat{\sqrt{P_s}}\,\xi \\
n^* & \hookleftarrow & \mathcal{G}(n^*, N) \\
d^* & = & R\,s^* + n^* \\
m^* & = & D\,R^\dagger N^{-1}\,d^* \\
\delta & = & s^* - m^* \\
s' & = & m \pm \delta \\
\langle \delta^\dagger \delta \rangle_{(s,n)} & = & D
\end{array}
$$

← solve with Conjugate Gradient

NIFTy5

5.0

Search docs

NIFTy – Numerical Information Field Theory

Probabilistic programming with auto-differentiation

NIFTy [1], [2], "**N**umerical **I**nformation **F**ield **T**heor**y**" is a versatile library designed to enable the development of signal inference algorithms that are independent of the underlying grids (spatial, spectral, temporal, …) and their resolutions. Its object-oriented framework is written in Python, although it accesses libraries written in C++ and C for efficiency.

NIFTy offers a toolkit that abstracts discretized representations of continuous spaces, fields in these spaces, and operators acting on these fields into classes. This allows for an abstract formulation and programming of inference algorithms, including those derived within information field theory. NIFTy's interface is designed to resemble IFT formulae in the sense that the user implements algorithms in NIFTy independent of the topology of the underlying spaces and the discretization scheme. Thus, the user can develop algorithms on subsets of problems and on spaces where the detailed performance of the algorithm can be properly evaluated and then easily generalize them to other, more complex spaces and the full problem, respectively.

The set of spaces on which NIFTy operates comprises point sets, n-dimensional regular grids, spherical spaces, their harmonic counterparts, and product spaces constructed as combinations of those. NIFTy takes care of numerical subtleties like the normalization of operations on fields and the numerical representation of model components, allowing the user to focus on formulating the abstract inference procedures and process-specific model properties.

References

[1] Selig et al., "NIFTY - Numerical Information Field Theory. A versatile PYTHON library for signal inference ", 2013, Astronmy and Astrophysics 554, 26; [DOI], [arXiv:1301.4499]

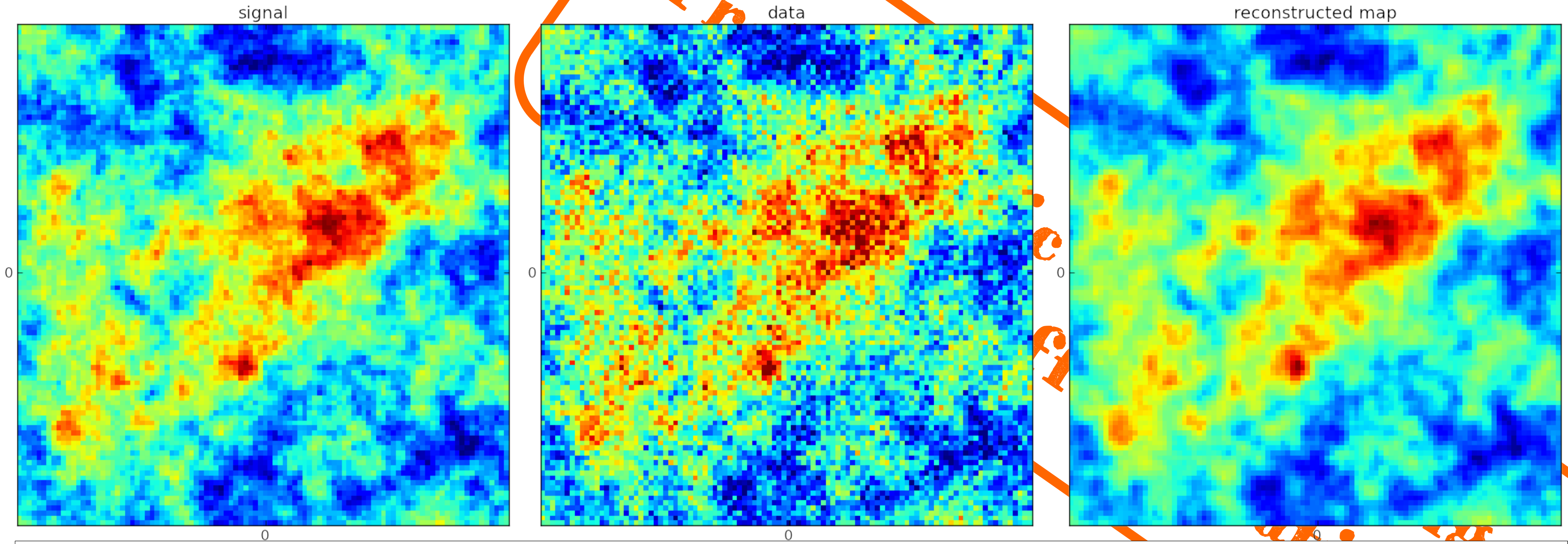
[2] Steininger et al., "NIFTy 3 - Numerical Information Field Theory - A Python framework for multicomponent signal inference on HPC clusters", 2017, accepted by Annalen der Physik; [arXiv:1708.01073]

Contents

NIFTy5

NIFTy – Numerical Information Field Theory

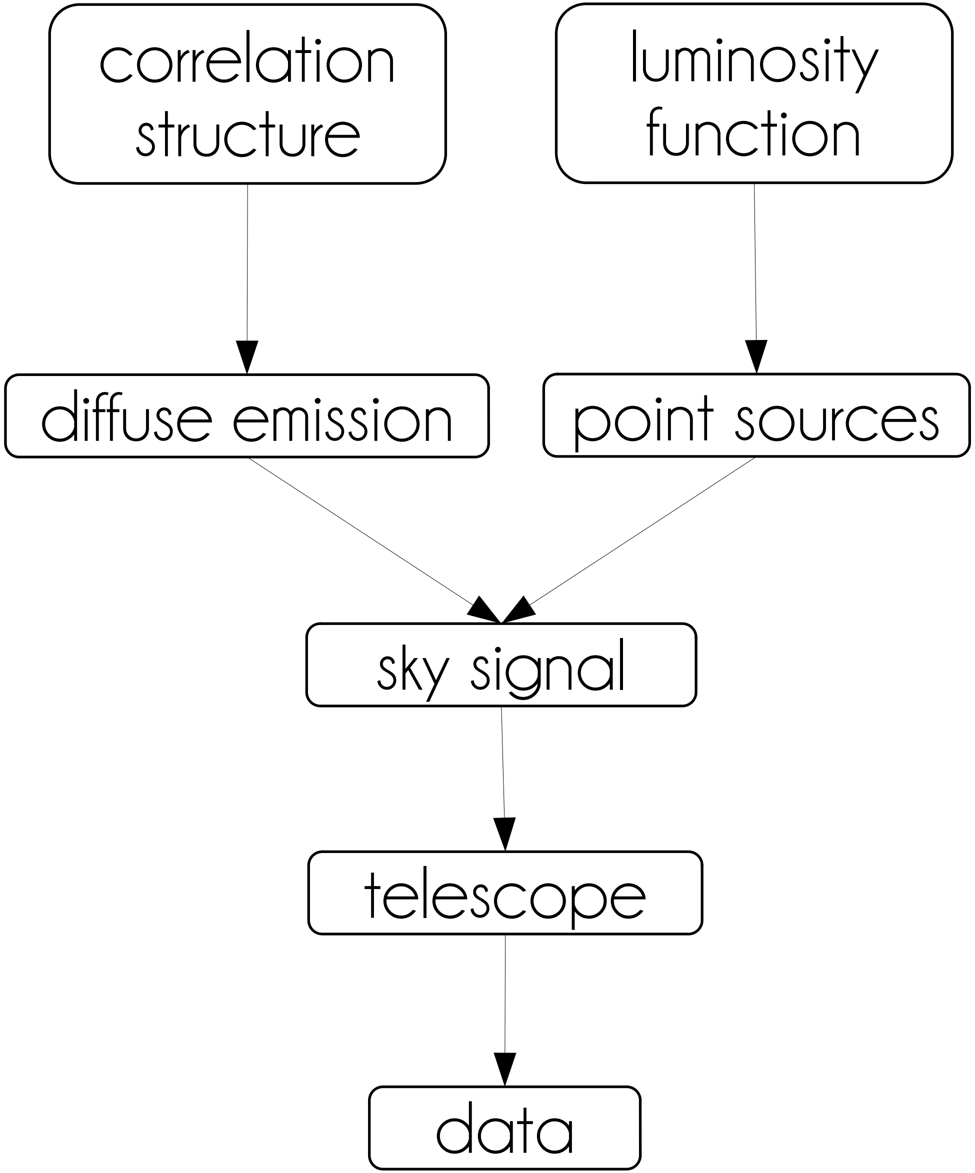
NIFTy [1], [2], “Numerical Information Field Theory” is a versatile library designed to enable the development of signal inference algorithms that are independent of the underlying grids (spatial, spectral, temporal, …) and their resolutions. Its object-oriented framework is written in Pytho




```
import nifty6 as ift
s_space = ift.RGSpace([N,N])
```

NIFTy tutorial part 1

linear reconstructions


correlation structure
luminosity function
diffuse emission
point sources
sky signal
telescope
data

$\mathcal{P}(d|s)$ Data model

known → $$d = R\,e^{s} + n$$

unknown → $$\lambda = R\,e^{s}$$

$$\mathcal{P}(s) = \mathcal{G}(s,\, S)$$ unknown

$$\mathcal{P}(d|\lambda) = \prod_i \frac{\lambda_i^{d_i}}{d_i!}\, e^{-\lambda_i}$$

Information

$$\mathcal{H}(\boldsymbol{d}, \boldsymbol{s}, \boldsymbol{\tau}) = -\log \mathcal{P}(\boldsymbol{d}, \boldsymbol{s}, \boldsymbol{\tau})$$

likelihood

$$= 1^\dagger \left[\log(d!) + \boldsymbol{R}\left(\mathrm{e}^{\boldsymbol{s}} + \mathrm{e}^{\boldsymbol{u}}\right)\right] - \boldsymbol{d}^\dagger \log\left[\boldsymbol{R}\left(\mathrm{e}^{\boldsymbol{s}} + \mathrm{e}^{\boldsymbol{u}}\right)\right]$$

diffuse prior

$$+ \frac{1}{2}\boldsymbol{s}^\dagger \boldsymbol{S}^{-1}\boldsymbol{s} + \frac{1}{2}\log\left(\det\left[\boldsymbol{S}\right]\right)$$

hyperprior

$$+ (\boldsymbol{\alpha} - \boldsymbol{1})^\dagger \boldsymbol{\tau} + \boldsymbol{q}^\dagger \mathrm{e}^{-\boldsymbol{\tau}} + \frac{1}{2}\boldsymbol{\tau}^\dagger \boldsymbol{T}\boldsymbol{\tau}$$

point prior

$$+ (\boldsymbol{\beta} - \boldsymbol{1})^\dagger \boldsymbol{u} + \boldsymbol{\eta}^\dagger \mathrm{e}^{-\boldsymbol{u}}$$

correlation structure

$$\boldsymbol{S} = \sum_k \mathrm{e}^{\tau_k} \boldsymbol{S}_k$$

Bayesian Sampling

Sampling the unknown signal according to its posterior probability

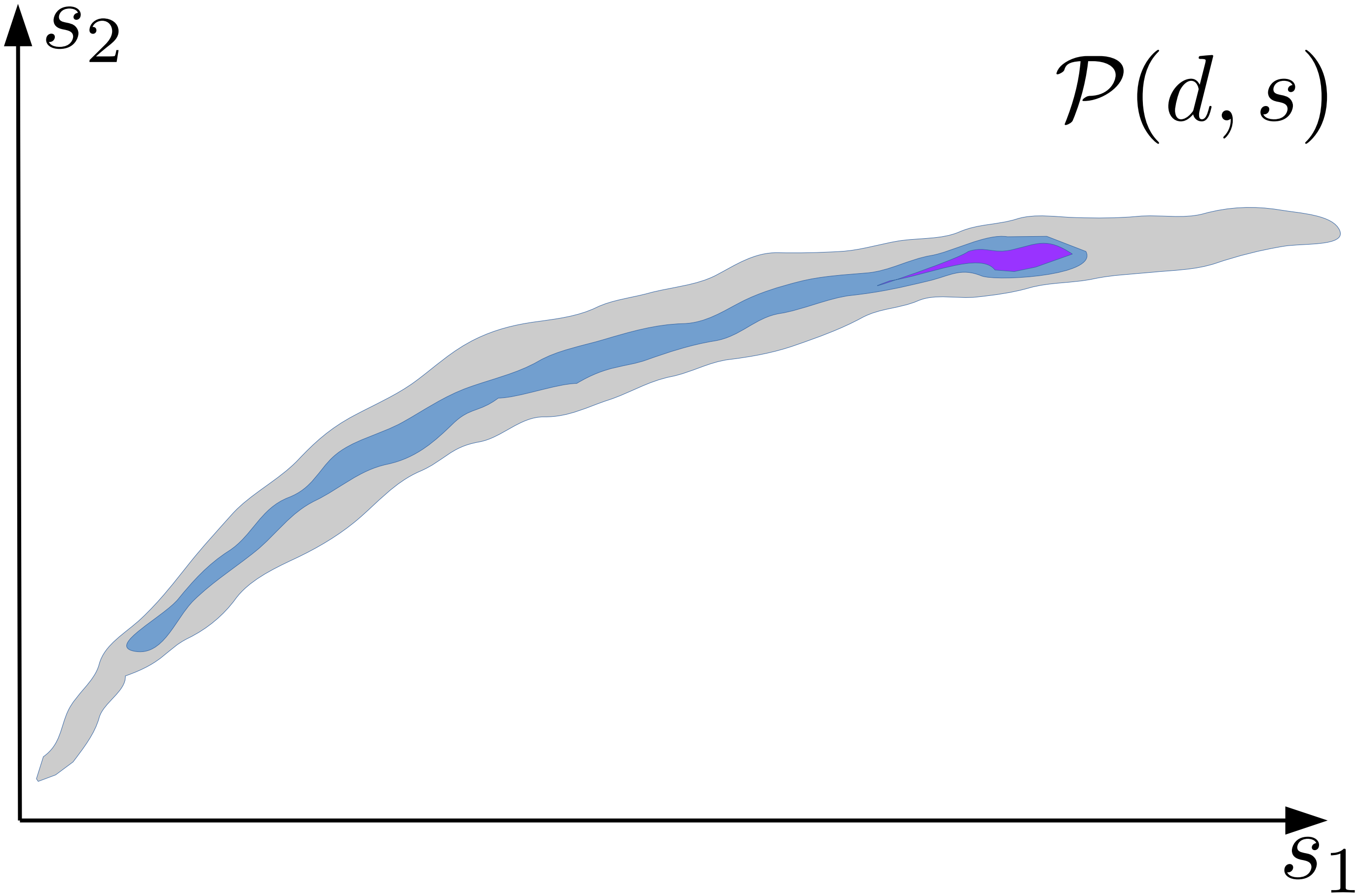
.

$$\mathcal{P}(s|d) = \mathcal{P}(d,s)/\mathcal{P}(d)$$

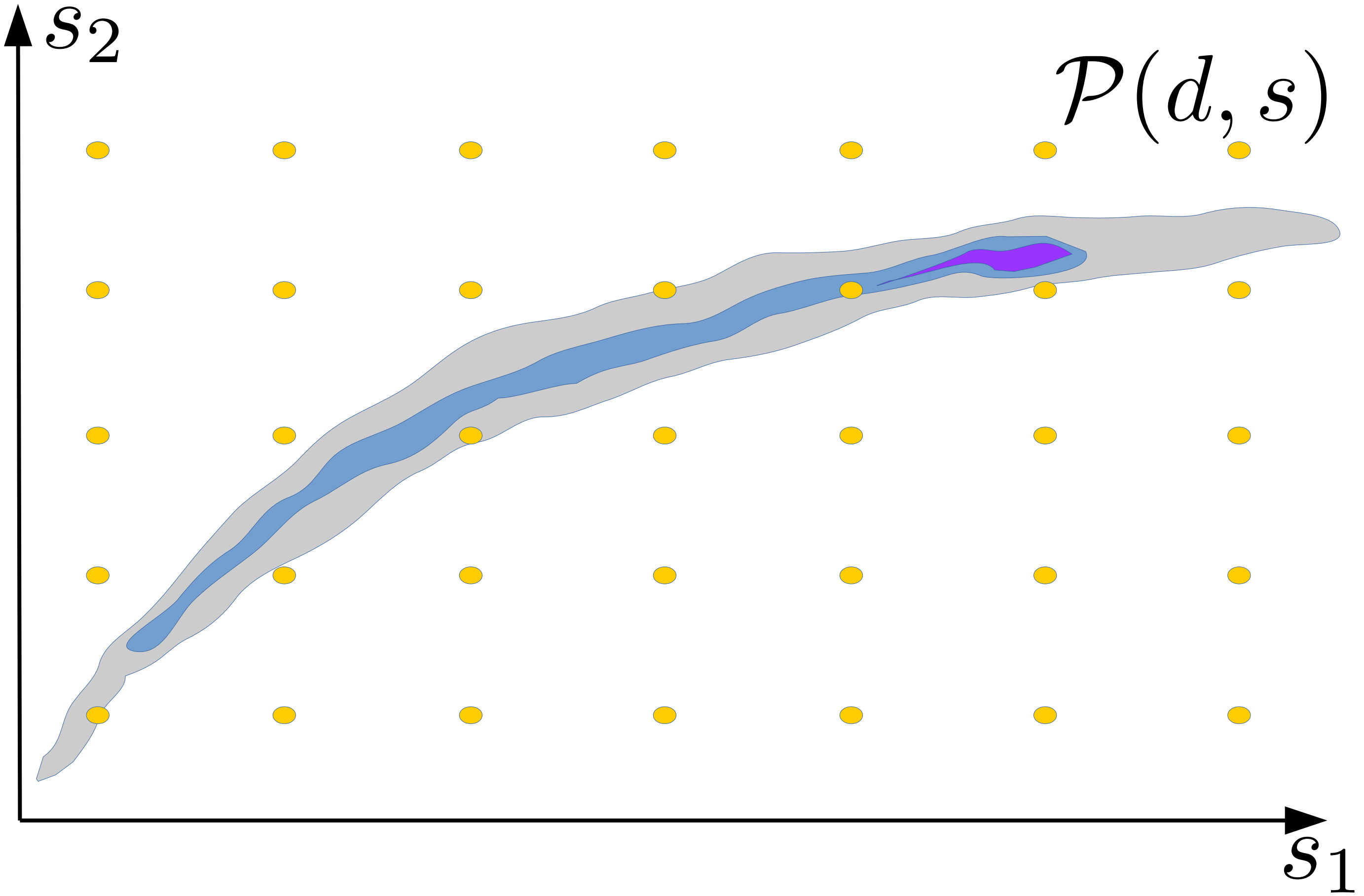
Goal is to calculate posterior expectation values:

$$\begin{aligned}
\langle f(s)\rangle_{\mathcal{P}(s|d)} &= \int ds\,\mathcal{P}(s|d)\,f(s) \\
&= \frac{\int ds\,\mathcal{P}(d,s)\,f(s)}{\int ds\,\mathcal{P}(d,s)} \approx \frac{\sum_i w_i\,f(s_i)}{\sum_i w_i}
\end{aligned}$$

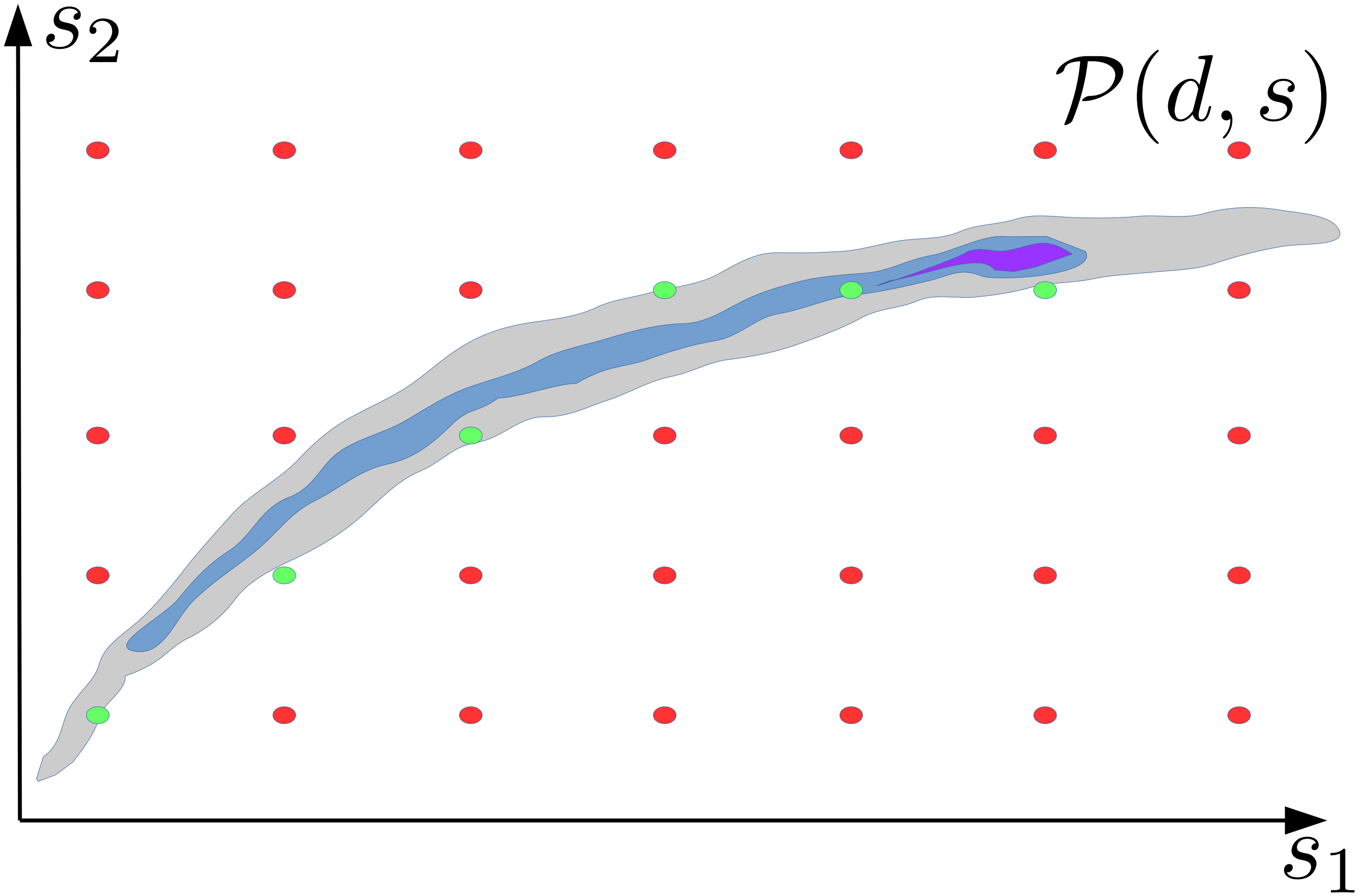
Sampling



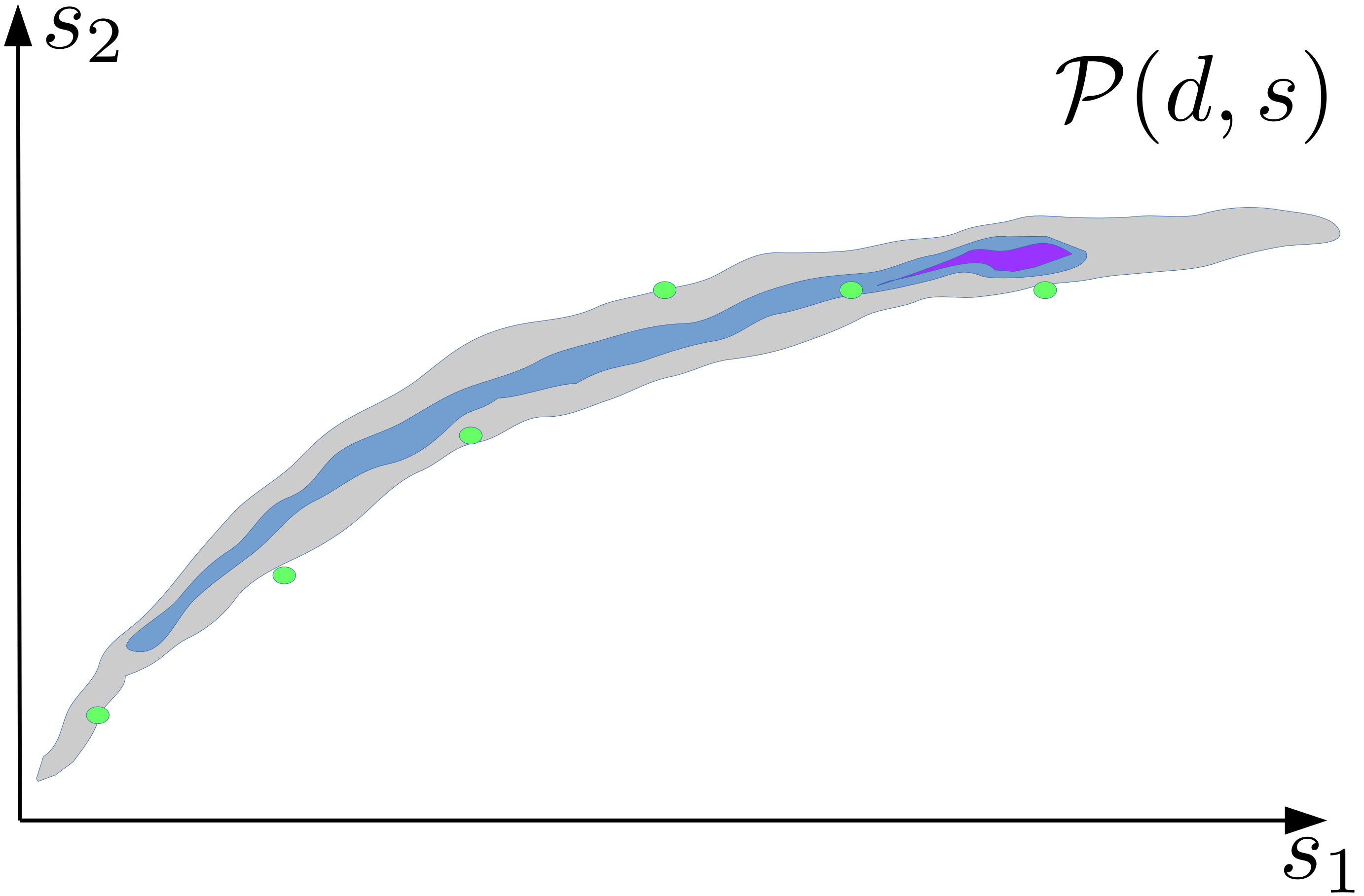
Uniform Sampling



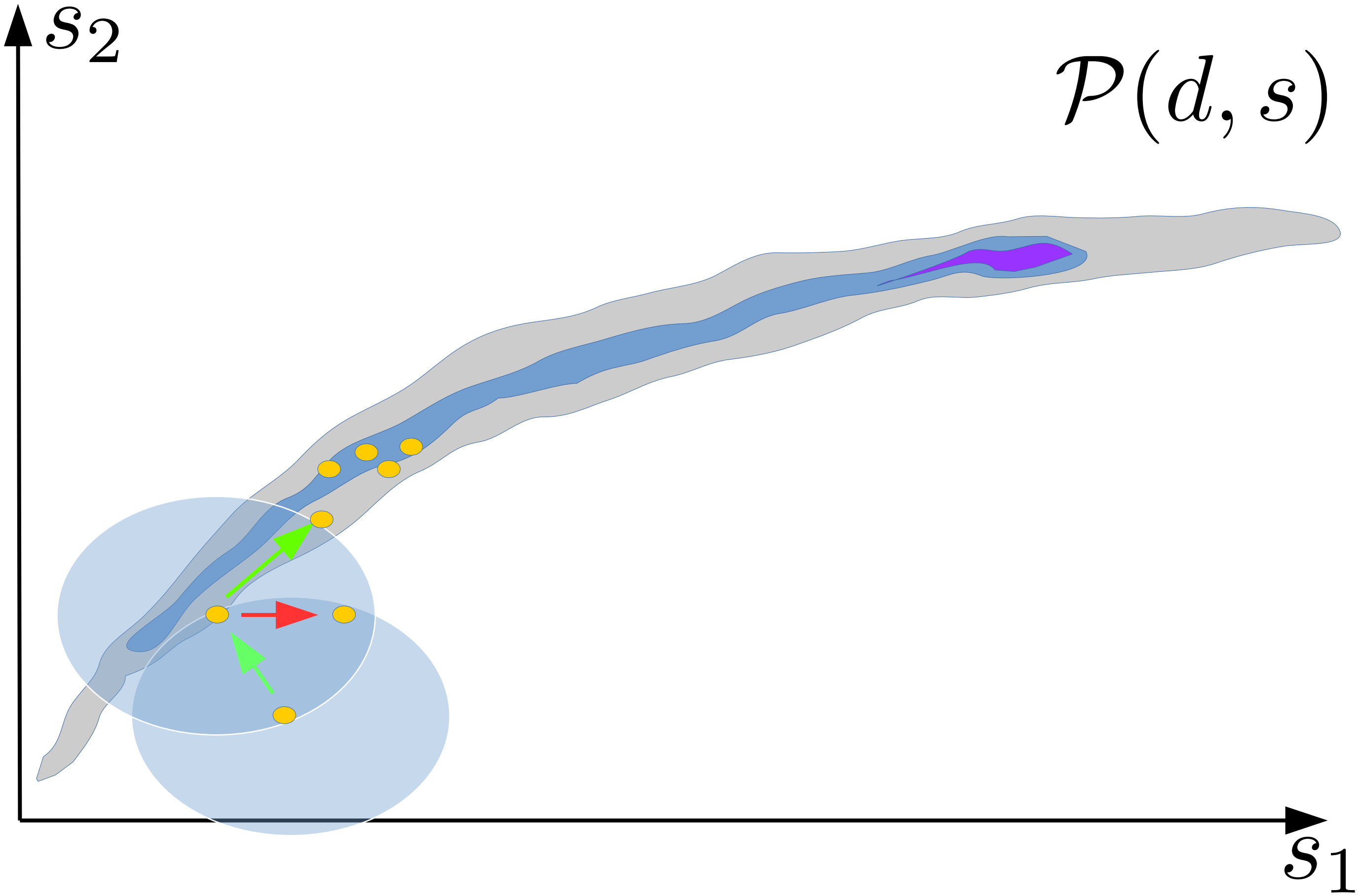
Rejection Sampling



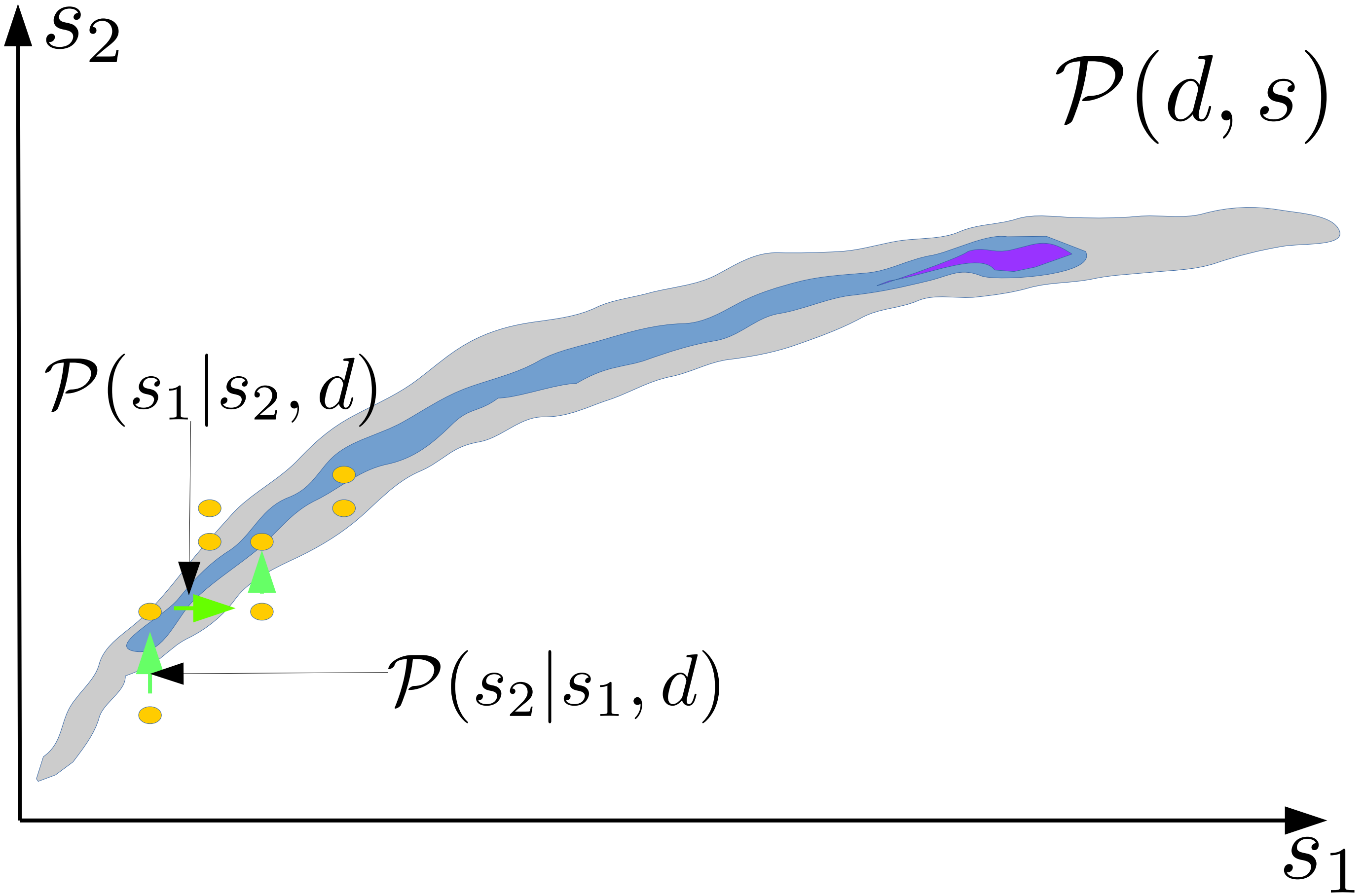
Rejection Sampling



Metropolis Hasting Sampling

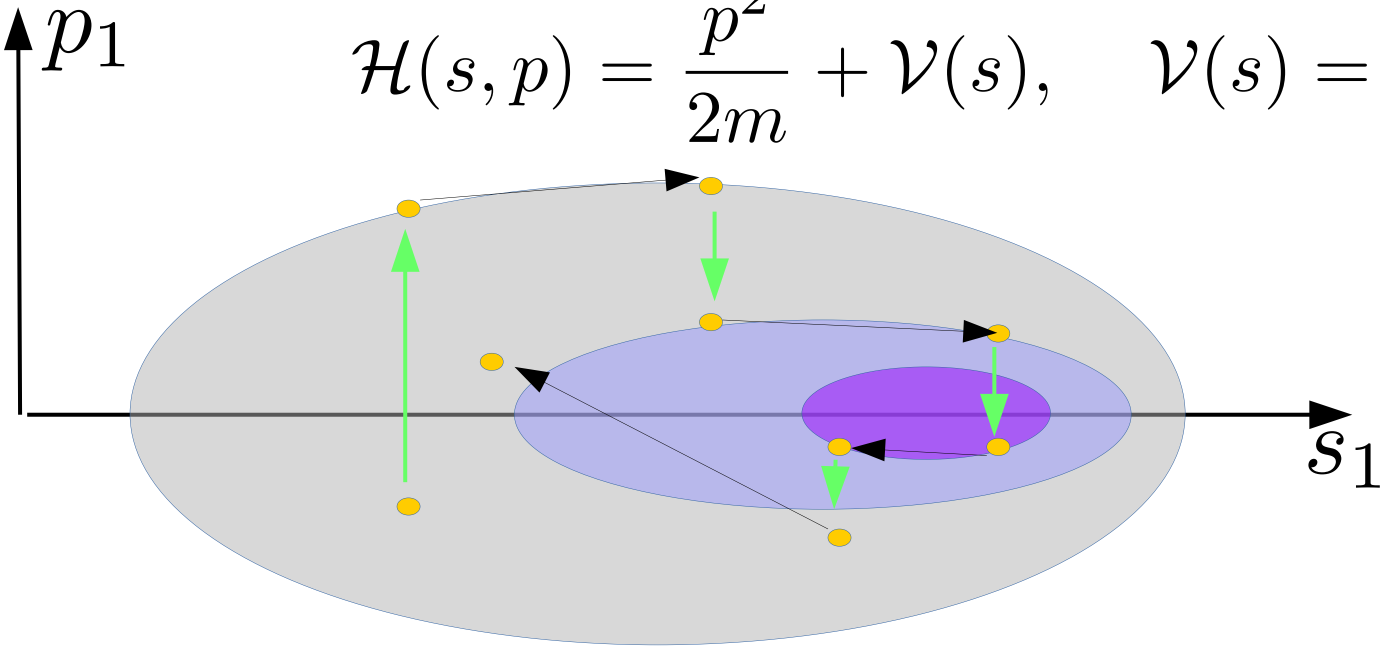


Gibbs Sampling



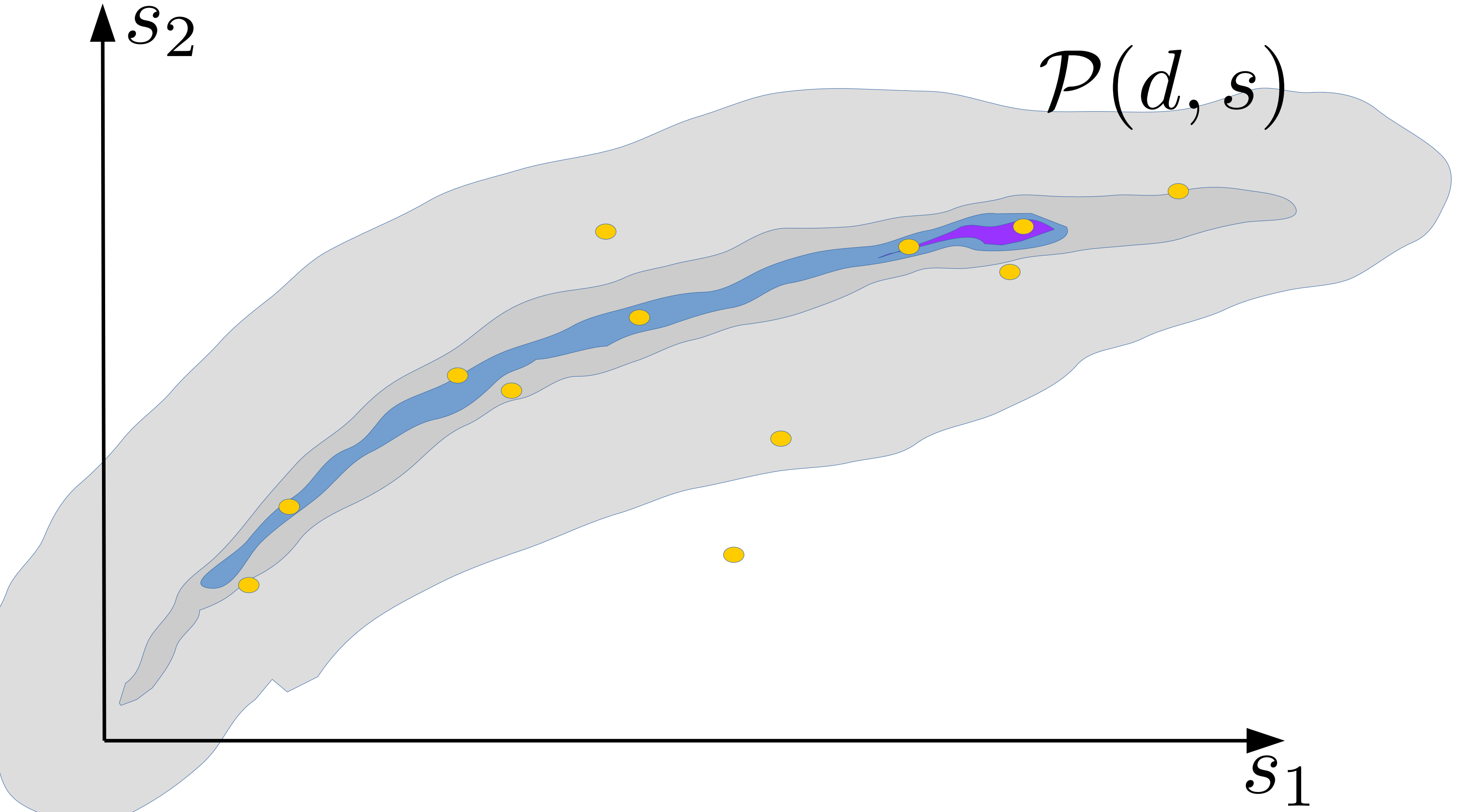
Hamiltonian Sampling

$$\mathcal{H}(s,p) = \frac{p^2}{2m} + \mathcal{V}(s), \quad \mathcal{V}(s) = -\ln \mathcal{P}(d,s)$$

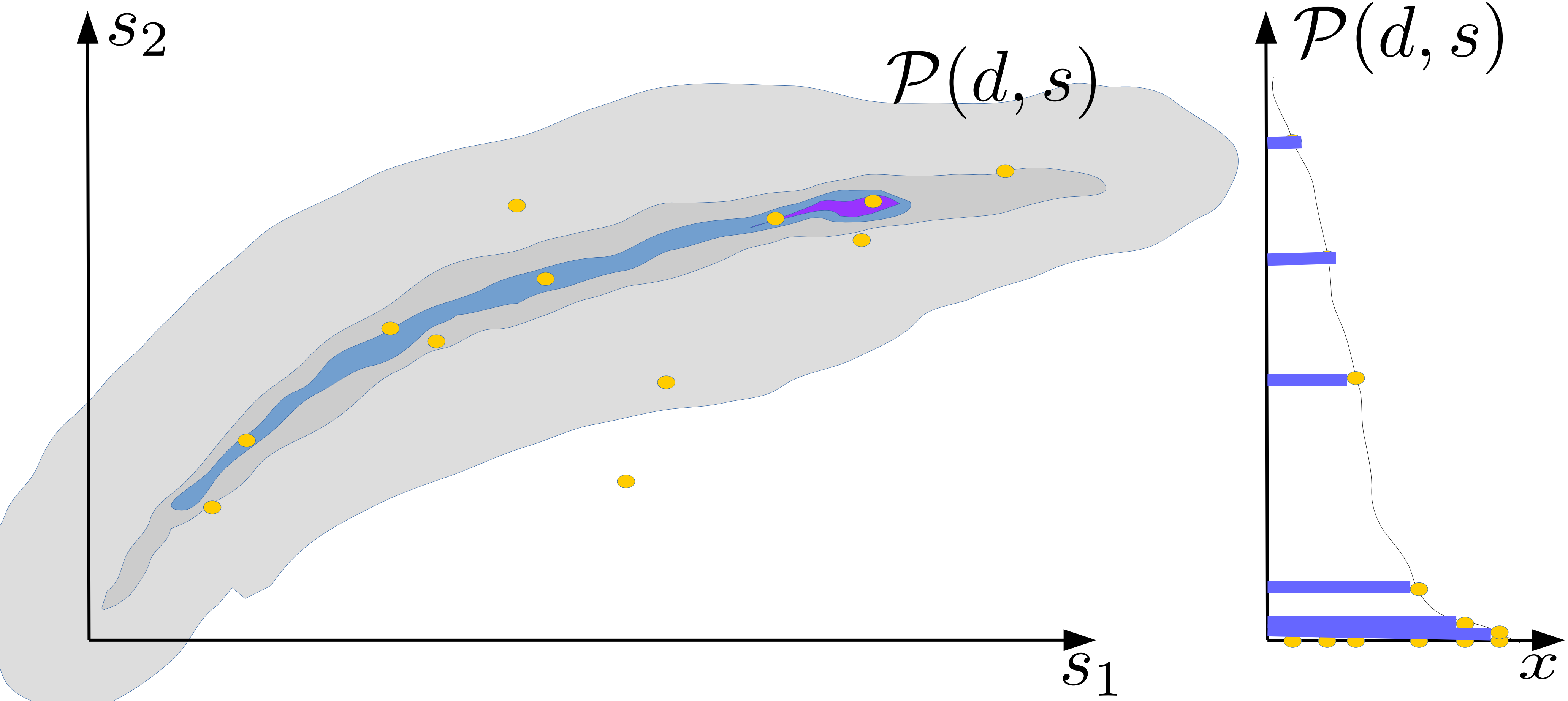


$$\mathcal{P}(d,s,p) = \mathcal{P}(d,s) \times \mathcal{P}(p)$$

Nested Sampling



Nested Sampling

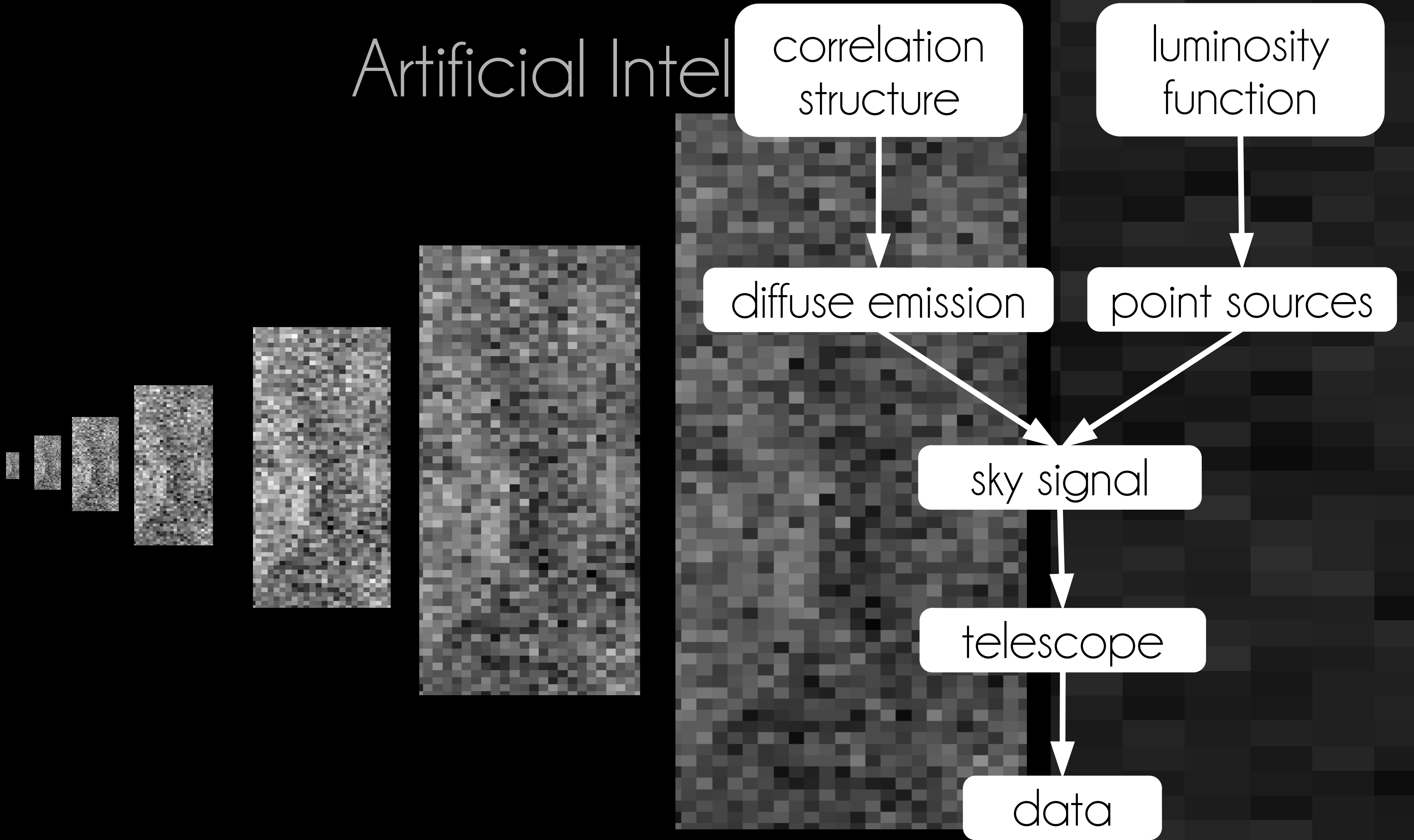


Information

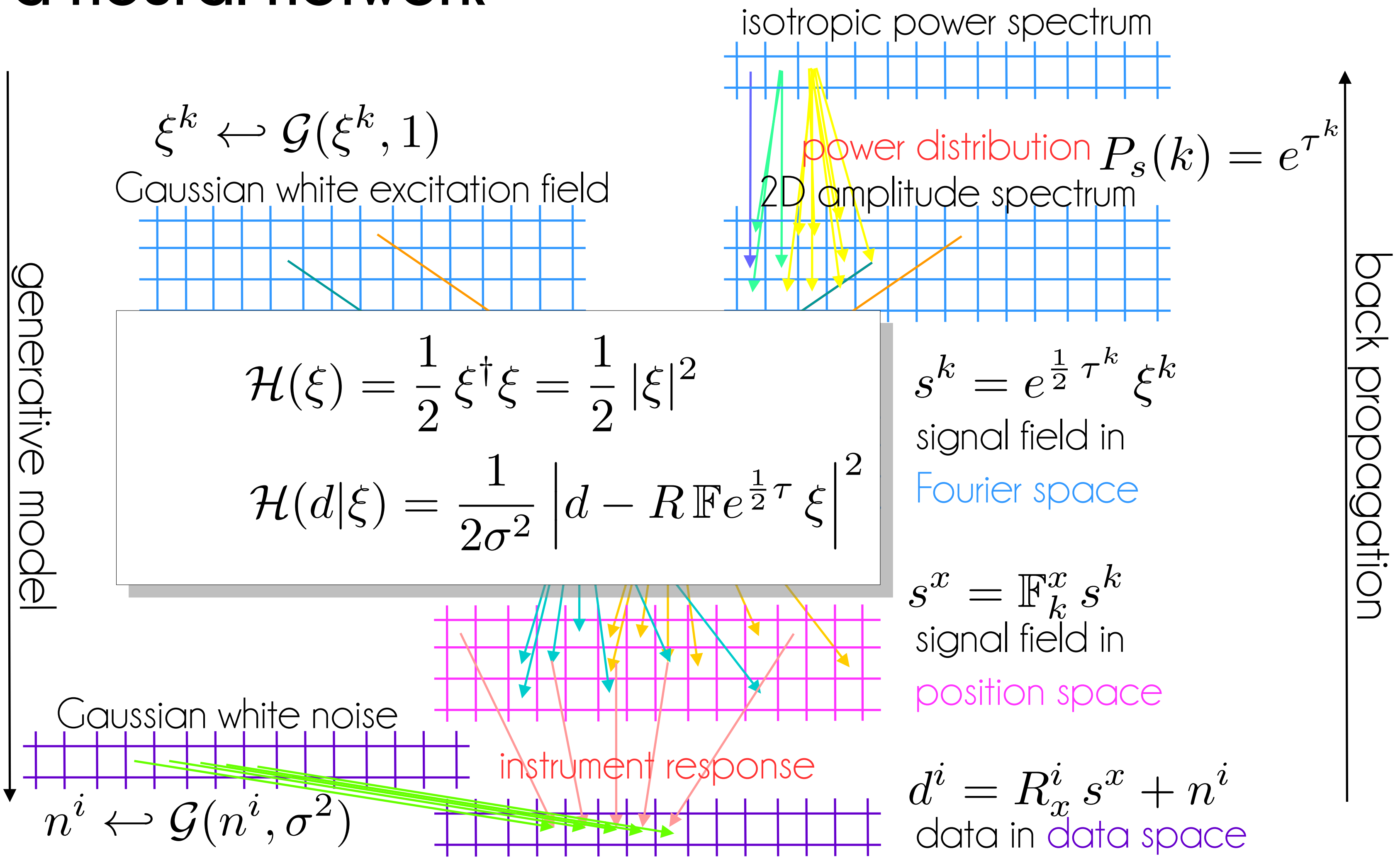
$$
\begin{aligned}
\mathcal{H}(\boldsymbol{d},\boldsymbol{s},\boldsymbol{\tau}) &= -\log \mathcal{P}(\boldsymbol{d},\boldsymbol{s},\boldsymbol{\tau}) \\
&= 1^{\dagger}\left[\log(d!) + \boldsymbol{R}\left(\mathrm{e}^{\boldsymbol{s}} + \mathrm{e}^{\boldsymbol{u}}\right)\right] - \boldsymbol{d}^{\dagger}\log\left[\boldsymbol{R}\left(\mathrm{e}^{\boldsymbol{s}} + \mathrm{e}^{\boldsymbol{u}}\right)\right] \\
&\quad + \frac{1}{2}\boldsymbol{s}^{\dagger}\boldsymbol{S}^{-1}\boldsymbol{s} + \frac{1}{2}\log\left(\det\left[\boldsymbol{S}\right]\right) \\
&\quad + (\boldsymbol{\alpha} - \mathbf{1})^{\dagger}\boldsymbol{\tau} + \boldsymbol{q}^{\dagger}\mathrm{e}^{-\boldsymbol{\tau}} + \frac{1}{2}\boldsymbol{\tau}^{\dagger}\boldsymbol{T}\boldsymbol{\tau} \\
&\quad + (\boldsymbol{\beta} - \mathbf{1})^{\dagger}\boldsymbol{u} + \boldsymbol{\eta}^{\dagger}\mathrm{e}^{-\boldsymbol{u}}
\end{aligned}
$$

$$
\boldsymbol{S} = \sum_{k} \mathrm{e}^{\tau_k}\boldsymbol{S}_k
$$

Artificial Intel



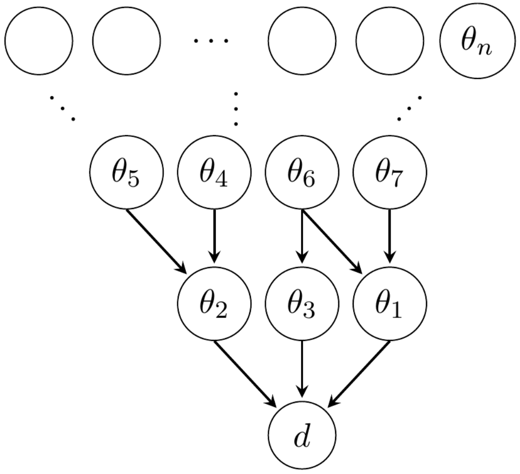
IFT as a neural network

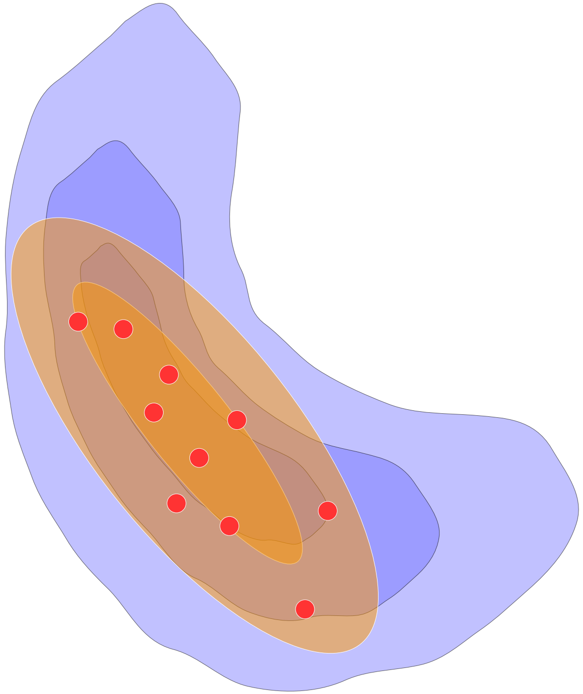


Metric Gaussian Variational Inference

Knollmüller & Enßlin (arXiv:1901.11033)

Hierarchical Model





// inference
initialize $\bar\xi$
while $\bar\xi$ *not converged* **do**
  $\Xi^{-1} = J^\dagger_{\bar\xi} M_{d|\bar\xi} J_{\bar\xi} + \mathbb{1}$
  // draw N samples
  for *N samples* **do**
    $\xi' \curvearrowleft \mathcal{G}(\xi, \mathbb{1})$
    $n' \curvearrowleft \mathcal{G}(n, M^{-1}_{d|\bar\xi})$
    $d' = J_{\bar\xi}\xi' + n'$
    $j' = J^\dagger_{\bar\xi} M_{d|\bar\xi} d'$
    solve $j' = \Xi^{-1}\bar\xi'$ for $\bar\xi'$ with conjugate gradient
    store sample $\Delta\xi_i = \xi' - \bar\xi'$
  end
  // Use these samples to minimize KL with respect to the mean
  while D_{KL} *not converged* **do**
    // Stochastically estimate KL and its gradient
    $\mathcal{D}_{KL} = \frac{1}{N}\sum_{i=0}^{N} \mathcal{H}(d, \bar\xi + \Delta\xi_i)$
    $\frac{\partial \mathcal{D}_{KL}}{\partial \bar\xi} = \frac{1}{N}\sum_{i=0}^{N} \frac{\partial \mathcal{H}}{\partial \bar\xi}(d, \bar\xi + \Delta\xi_i)$
    solve $\frac{\partial \mathcal{D}_{KL}}{\partial \bar\xi} = \Xi^{-1}\Delta_{\bar\xi}$ for natural gradient $\Delta_{\bar\xi}$ with conjugate gradient
    use $\Delta_{\bar\xi}$ to update $\bar\xi$ such that D_{KL} is minimized
  end
  // now the mean is updated
end
// Preparing posterior analysis
for *N samples* **do**
  $\xi' \curvearrowleft \mathcal{G}(\xi, \mathbb{1})$
  $n' \curvearrowleft \mathcal{G}(n, M^{-1}_{d|\bar\xi})$
  $d' = J_{\bar\xi}\xi' + n'$
  $j' = J^\dagger_{\bar\xi} M_{d|\bar\xi} d'$
  solve $j' = \Xi^{-1}\bar\xi'$ for $\bar\xi'$ with conjugate gradient
  store sample $\Delta\xi_i = \xi' - \bar\xi$ to use for posterior analysis
end

Variational Bayes

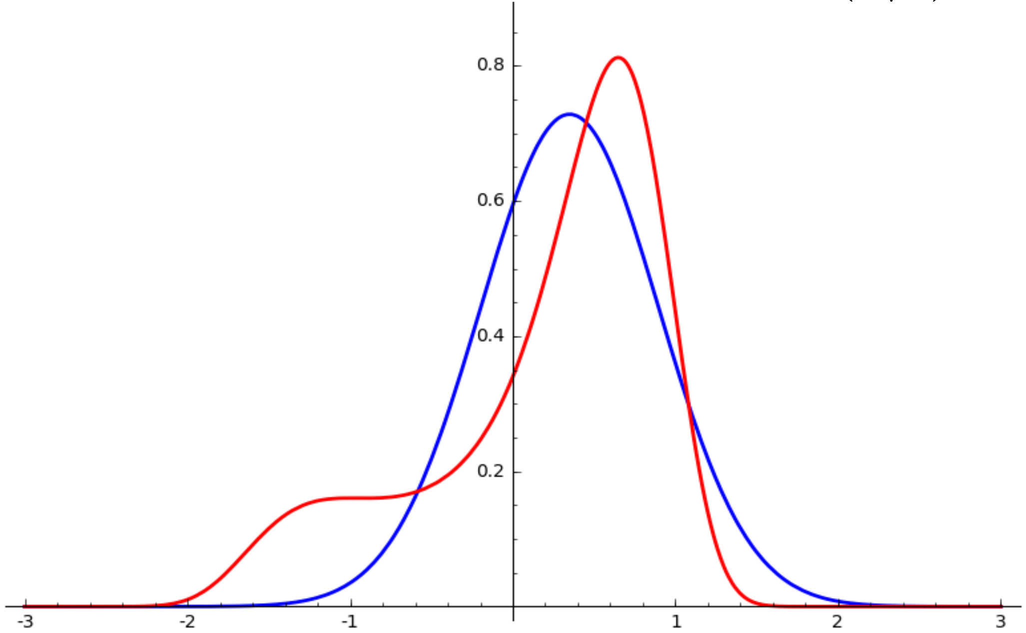
$\mathcal{P}(s|d)$

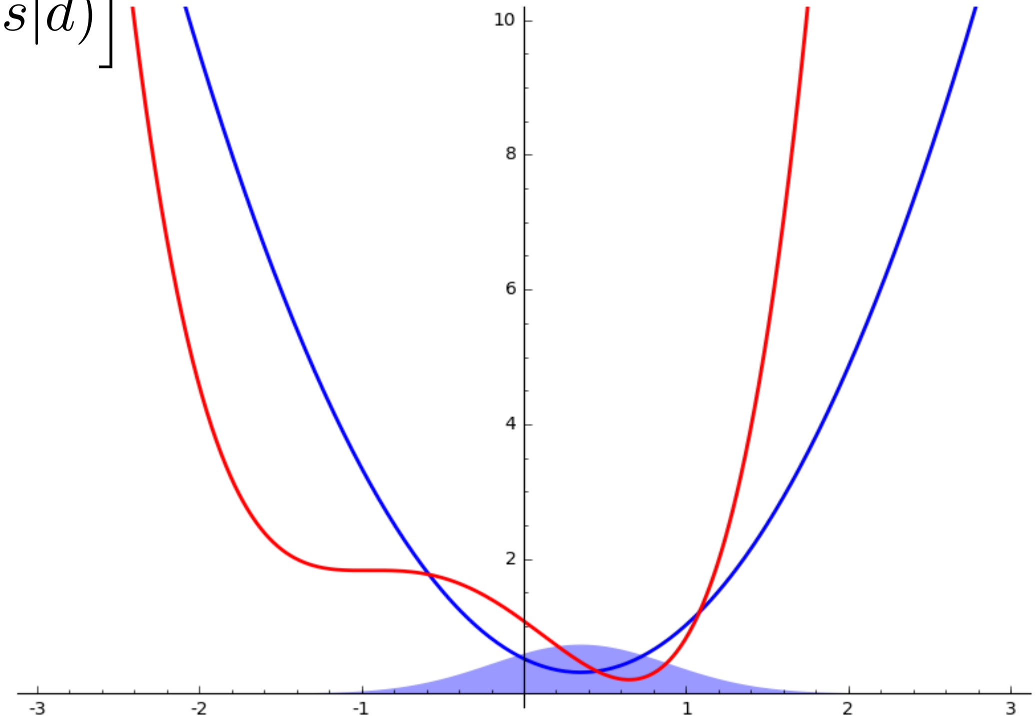
$\widetilde{\mathcal{P}}(s|d) = \mathcal{G}(s-m, D)$

$\mathcal{H}(s|d)$

$\widetilde{\mathcal{H}}(s|d) \widehat{=} \frac{1}{2}(s-m)^\dagger D^{-1}(s-m)$

$$\mathrm{KL}(\widetilde{\mathcal{P}}, \mathcal{P}) = \int \mathcal{D}s\, \widetilde{\mathcal{P}}(s|d)\, \left[\mathcal{H}(s|d) - \widetilde{\mathcal{H}}(s|d)\right]$$





Metric Gaussian Variational Bayes

$\mathcal{P}(s|d)$

$\widetilde{\mathcal{P}}(s|d) = \mathcal{G}(s-m, D)$

$\mathcal{H}(s|d)$

$\widetilde{\mathcal{H}}(s|d) \widehat{=} \frac{1}{2}(s-m)^\dagger D^{-1}(s-m)$

Knollmüller & Enßlin (2019)

$$\mathrm{KL}(\widetilde{\mathcal{P}}, \mathcal{P}) = \int \mathcal{D}s\, \widetilde{\mathcal{P}}(s|d) \left[\mathcal{H}(s|d) - \widetilde{\mathcal{H}}(s|d)\right]$$

$$D \approx \left\langle \frac{\partial \mathcal{H}(d,s)}{\partial s} \frac{\partial \mathcal{H}(d,s)}{\partial s}^\dagger \right\rangle_{(d|s=m)}^{-1}$$

Information 1 dim

$$\mathcal{H}(\boldsymbol{d},\boldsymbol{s},\boldsymbol{\tau}) = -\log \mathcal{P}(\boldsymbol{d},\boldsymbol{s},\boldsymbol{\tau})$$

$$= 1^{\dagger}\left[\log(d!) + \boldsymbol{R}\left(\mathrm{e}^{\boldsymbol{s}} + \mathrm{e}^{\boldsymbol{u}}\right)\right] - \boldsymbol{d}^{\dagger}\log\left[\boldsymbol{R}\left(\mathrm{e}^{\boldsymbol{s}} + \mathrm{e}^{\boldsymbol{u}}\right)\right]$$

$$+ \frac{1}{2}\boldsymbol{s}^{\dagger}\boldsymbol{S}^{-1}\boldsymbol{s} + \frac{1}{2}\log\left(\det\left[\boldsymbol{S}\right]\right)$$

$$+ (\boldsymbol{\alpha} - \mathbf{1})^{\dagger}\boldsymbol{\tau} + \boldsymbol{q}^{\dagger}\mathrm{e}^{-\boldsymbol{\tau}} + \frac{1}{2}\boldsymbol{\tau}^{\dagger}\boldsymbol{T}\boldsymbol{\tau}$$

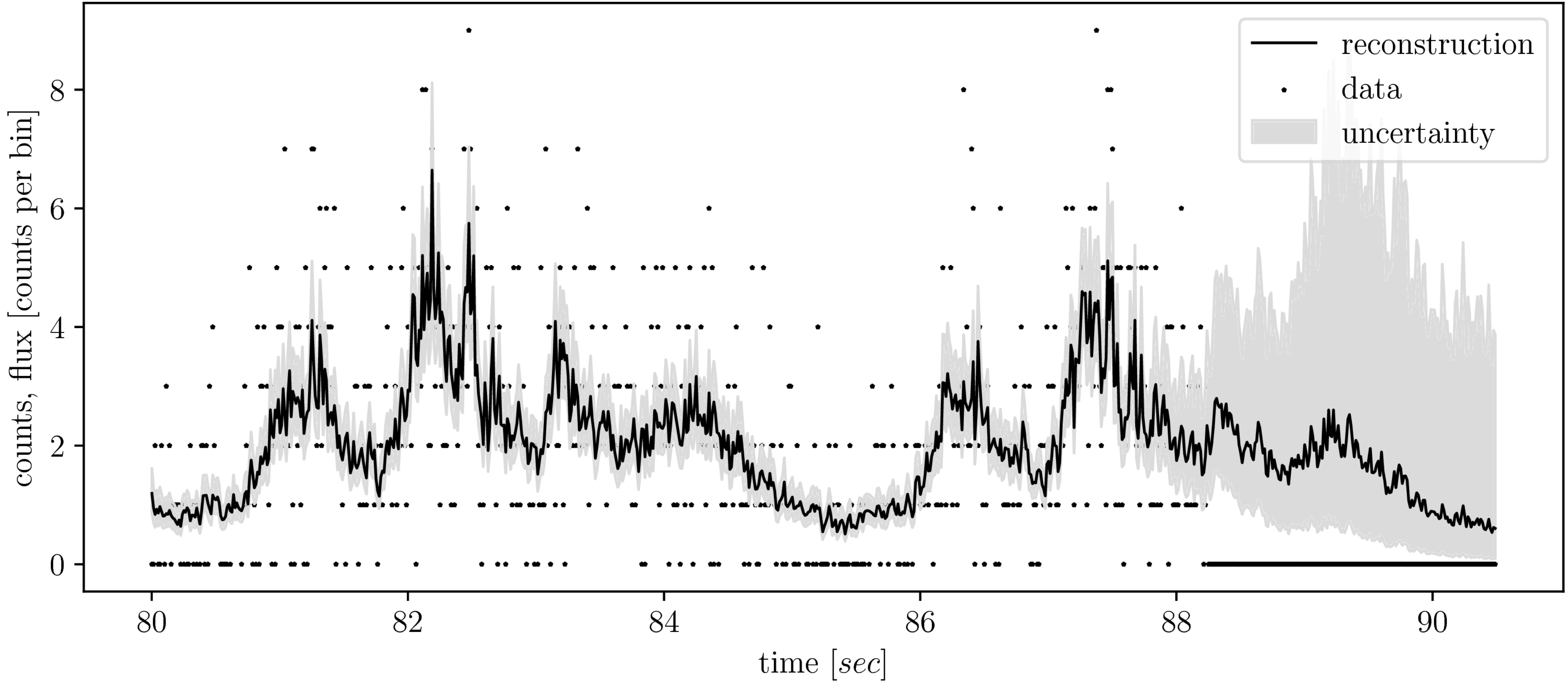
$$+ (\boldsymbol{\beta} - \mathbf{1})^{\dagger}\boldsymbol{u} + \boldsymbol{\eta}^{\dagger}\mathrm{e}^{-\boldsymbol{u}}$$

$$\boldsymbol{S} = \sum_{k} \mathrm{e}^{\tau_k}\boldsymbol{S}_k$$

Magnetar flare SGR 1900+14

1 dim

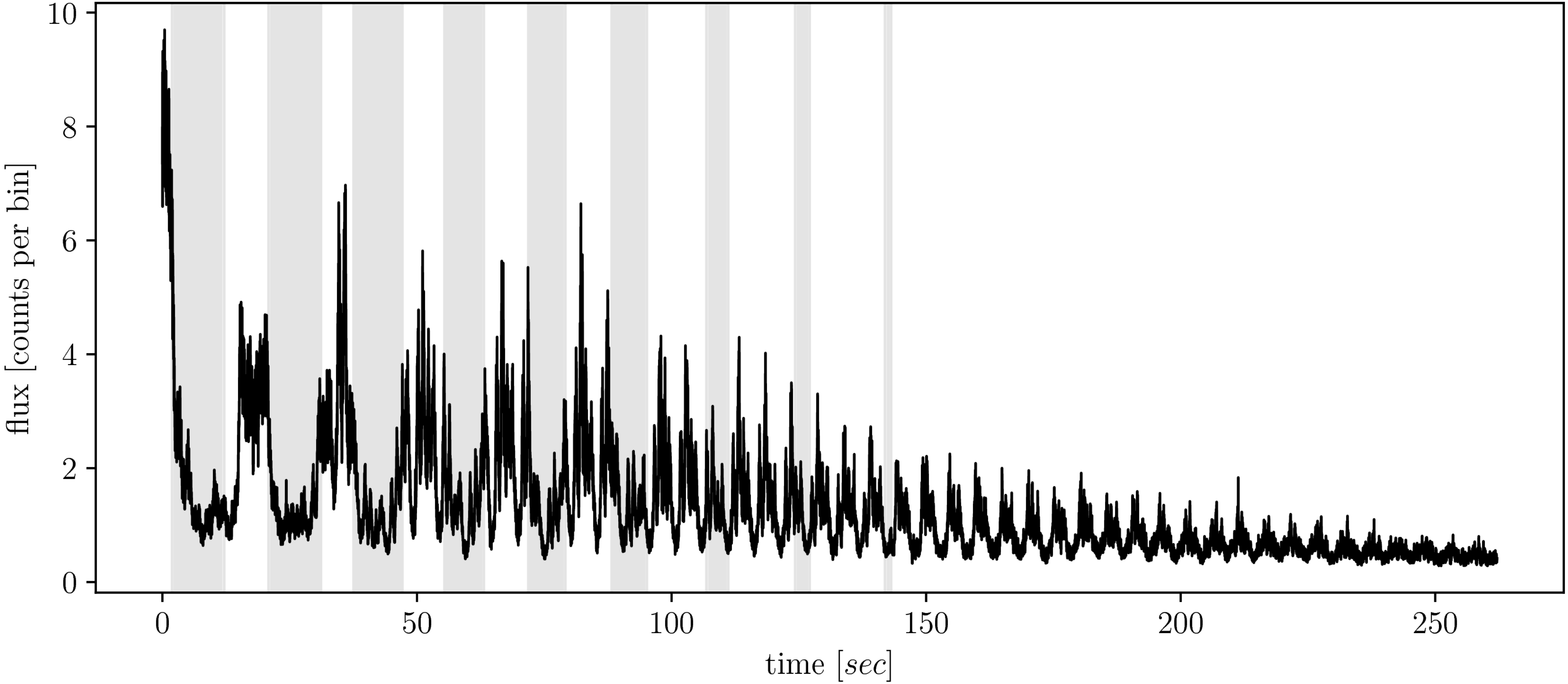
Pumpe et al. (2018)



Magnetar flare SGR 1900+14

Pumpe et al. (2018)

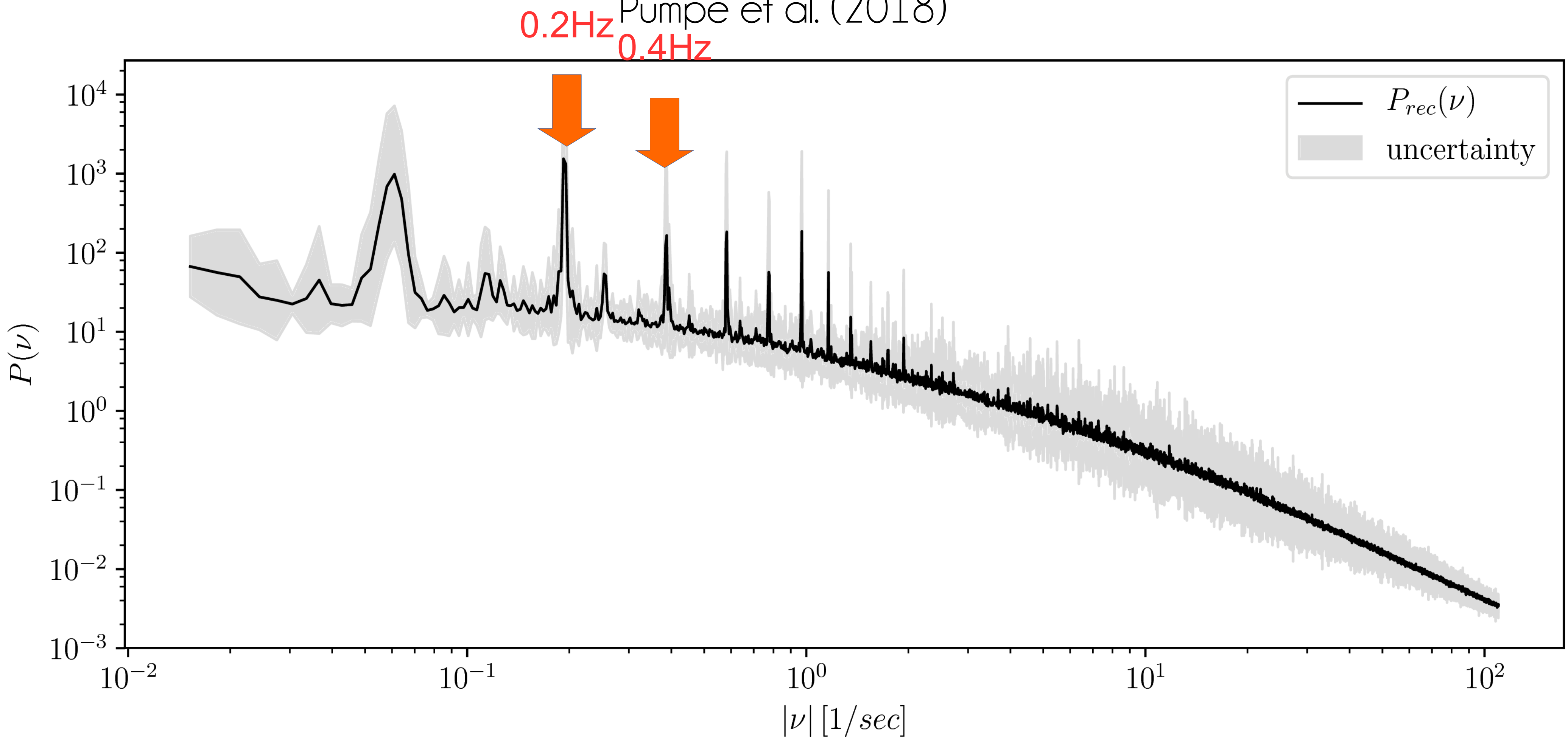
1 dim



Magnetar flare SGR 1900+14

1 dim

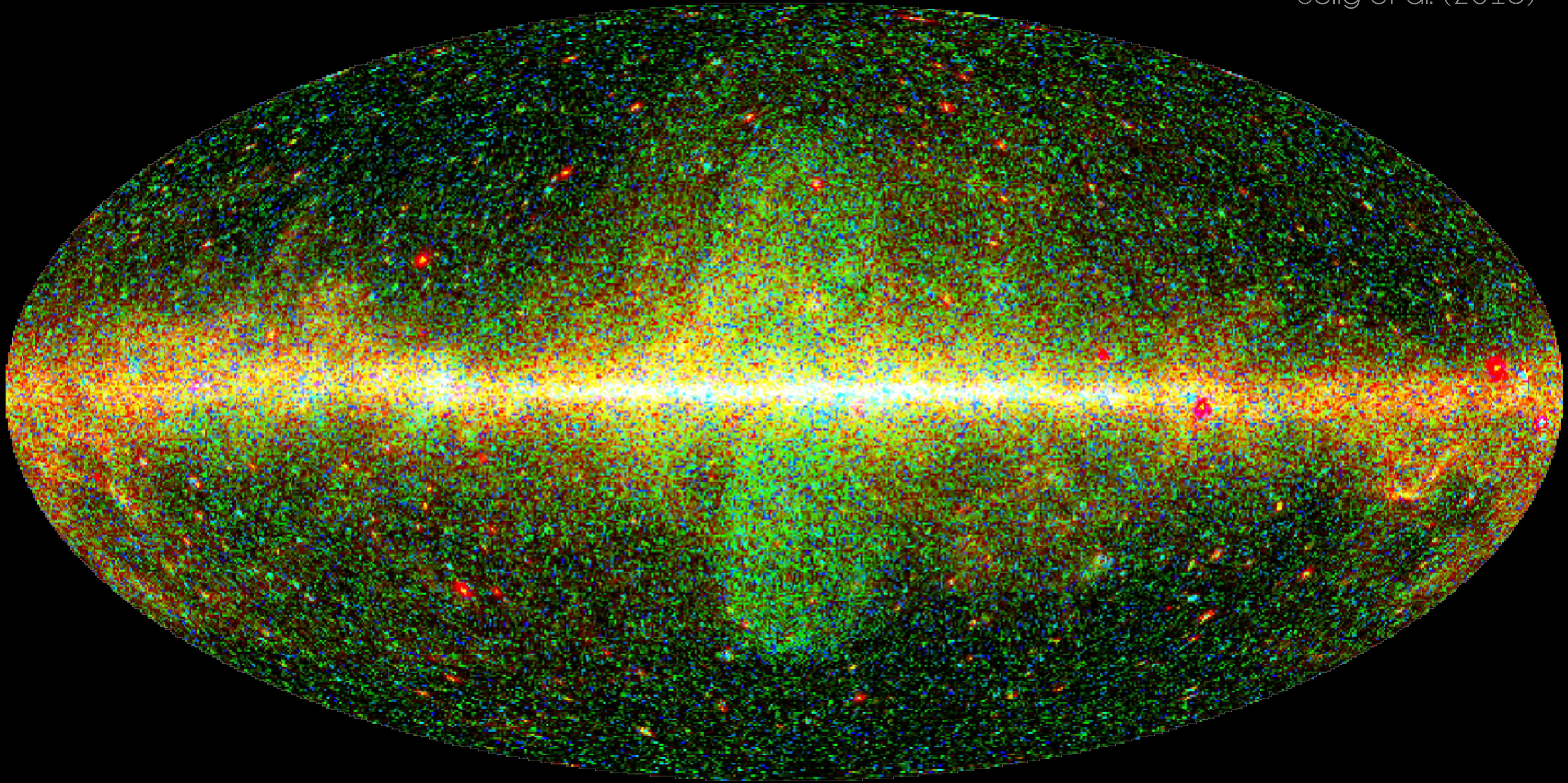
Pumpe et al. (2018)



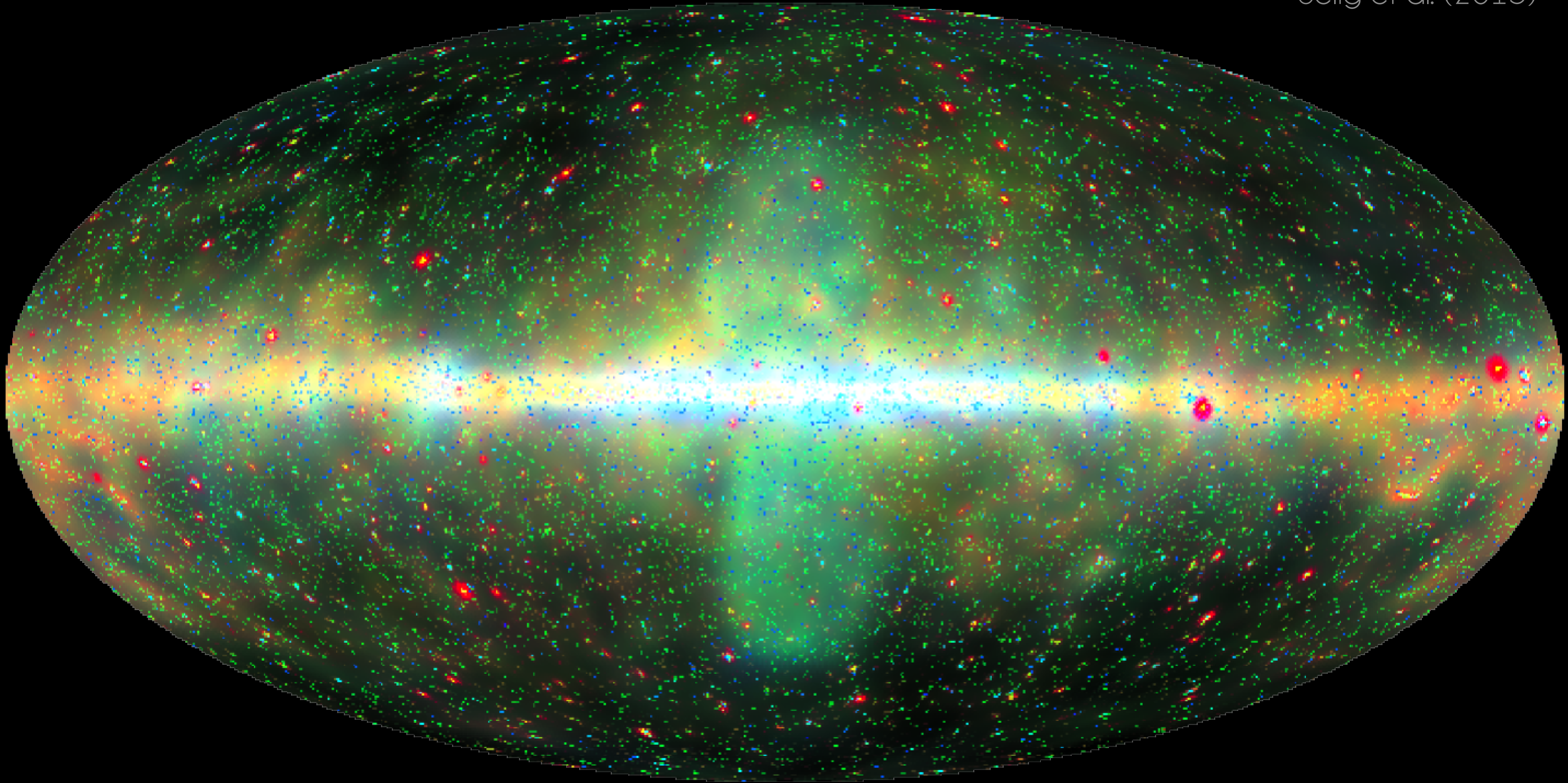
Information 2 dim

$$\mathcal{H}(\boldsymbol{d}, \boldsymbol{s}, \boldsymbol{\tau}) = -\log \mathcal{P}(\boldsymbol{d}, \boldsymbol{s}, \boldsymbol{\tau})$$
$$= 1^\dagger \left[\log(d!) + \boldsymbol{R}\left(\mathrm{e}^{\boldsymbol{s}} + \mathrm{e}^{\boldsymbol{u}}\right)\right] - \boldsymbol{d}^\dagger \log\left[\boldsymbol{R}\left(\mathrm{e}^{\boldsymbol{s}} + \mathrm{e}^{\boldsymbol{u}}\right)\right]$$
$$+ \frac{1}{2}\boldsymbol{s}^\dagger \boldsymbol{S}^{-1}\boldsymbol{s} + \frac{1}{2}\log\left(\det\left[\boldsymbol{S}\right]\right)$$
$$+ (\boldsymbol{\alpha} - \mathbf{1})^\dagger \boldsymbol{\tau} + \boldsymbol{q}^\dagger \mathrm{e}^{-\boldsymbol{\tau}} + \frac{1}{2}\boldsymbol{\tau}^\dagger \boldsymbol{T}\boldsymbol{\tau}$$
$$+ (\boldsymbol{\beta} - \mathbf{1})^\dagger \boldsymbol{u} + \boldsymbol{\eta}^\dagger \mathrm{e}^{-\boldsymbol{u}}$$

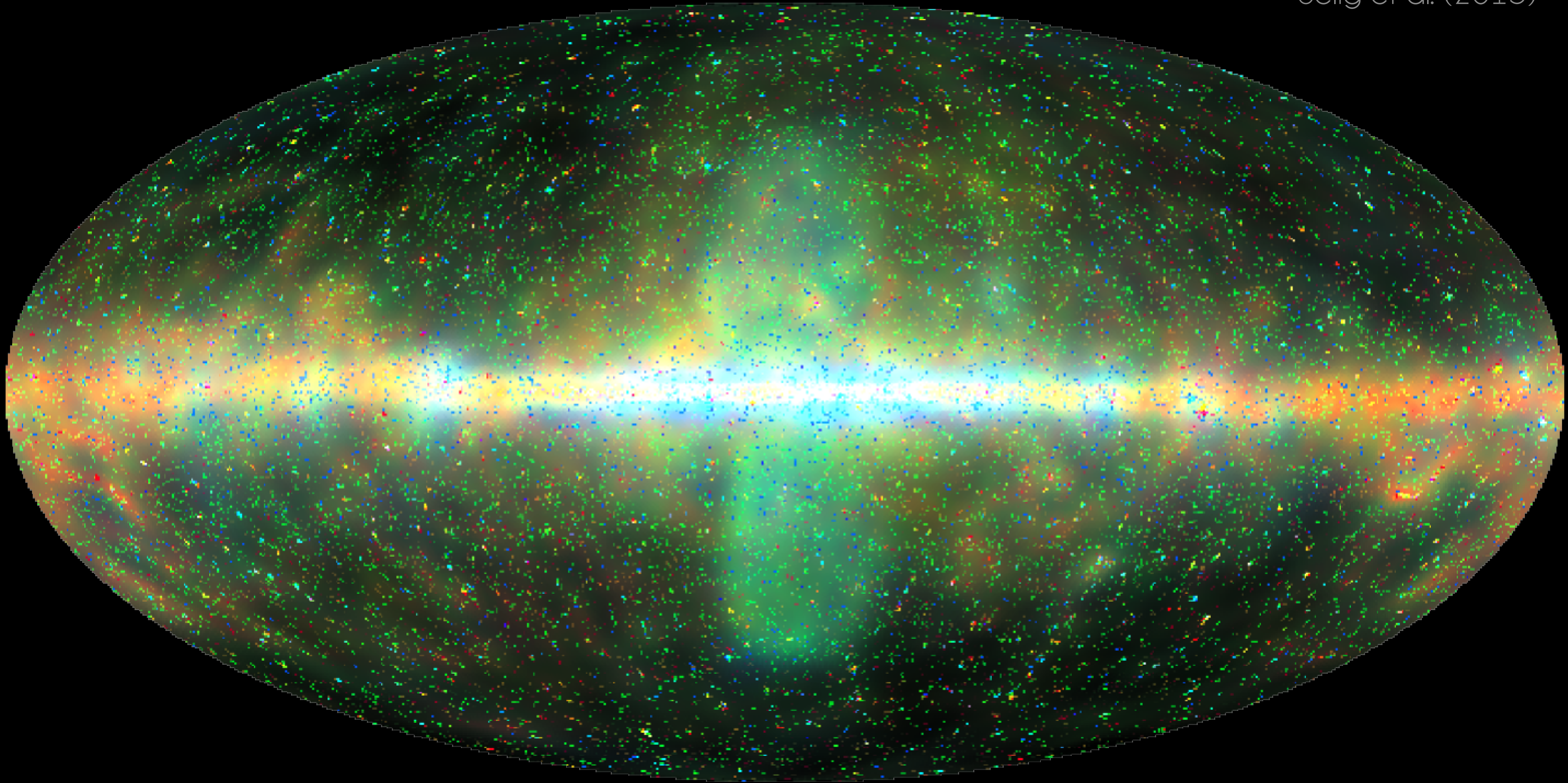
$$\boldsymbol{S} = \sum_k \mathrm{e}^{\tau_k}\boldsymbol{S}_k$$



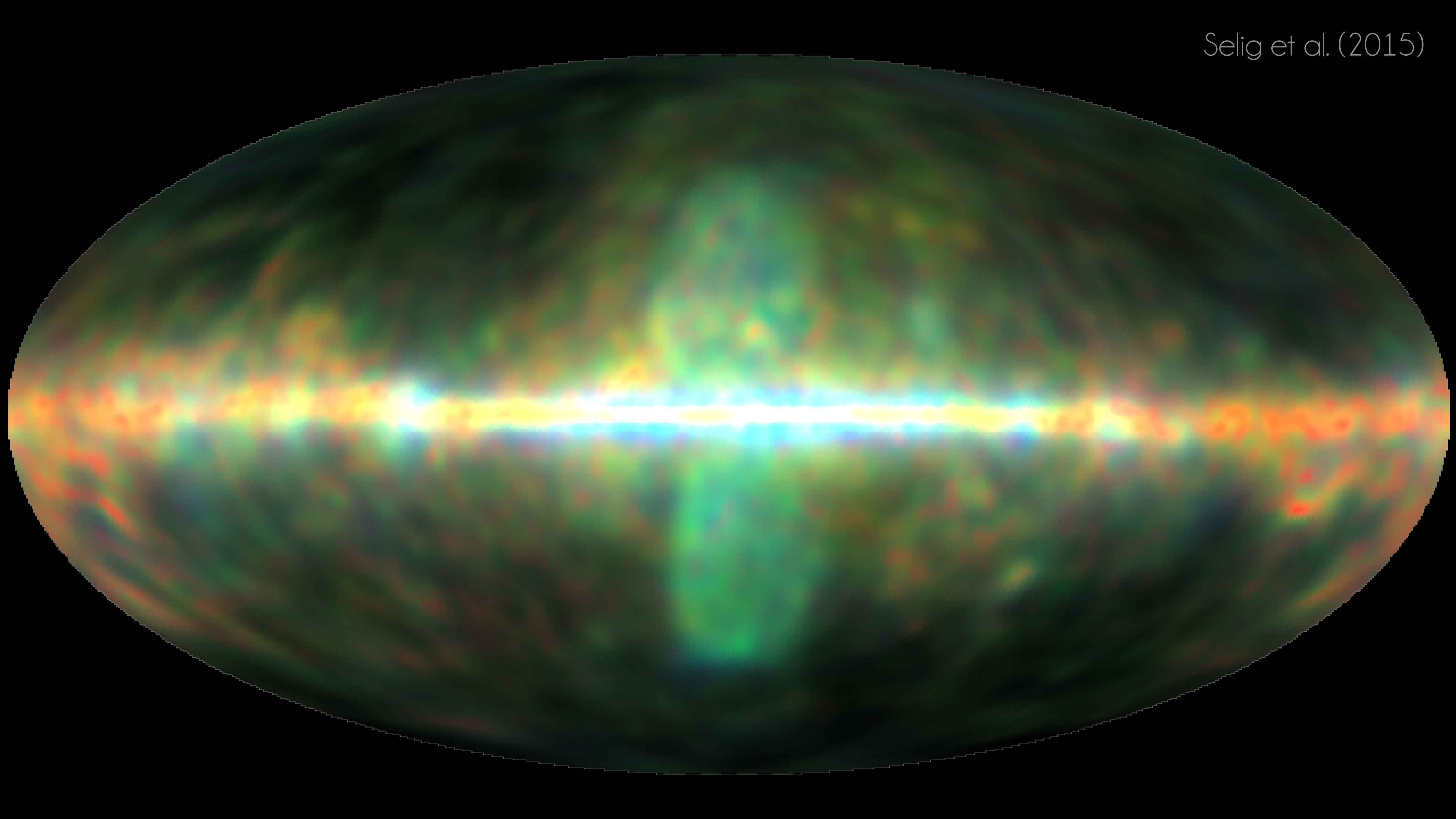
Selig et al. (2015)



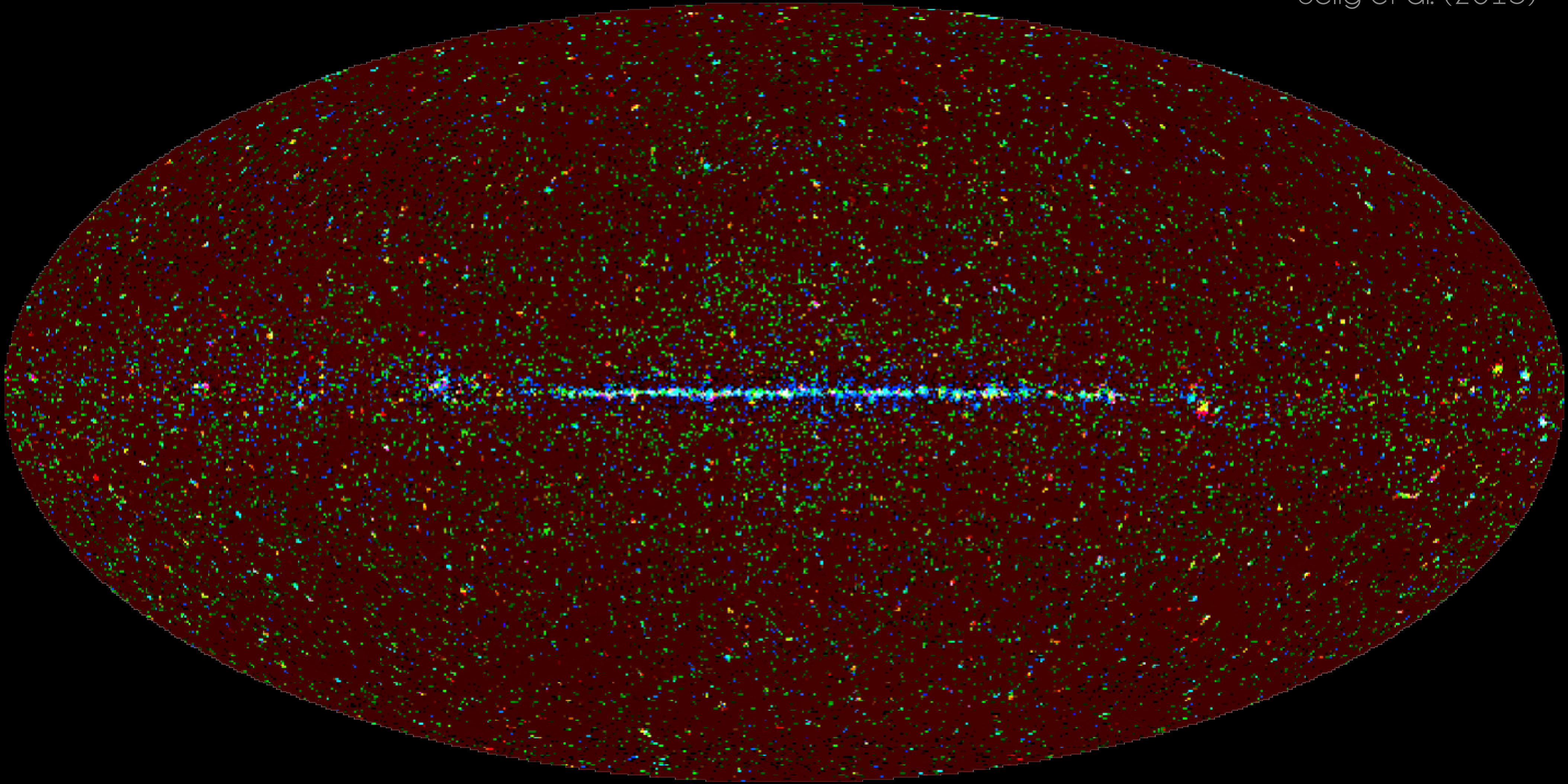
Selig et al. (2015)

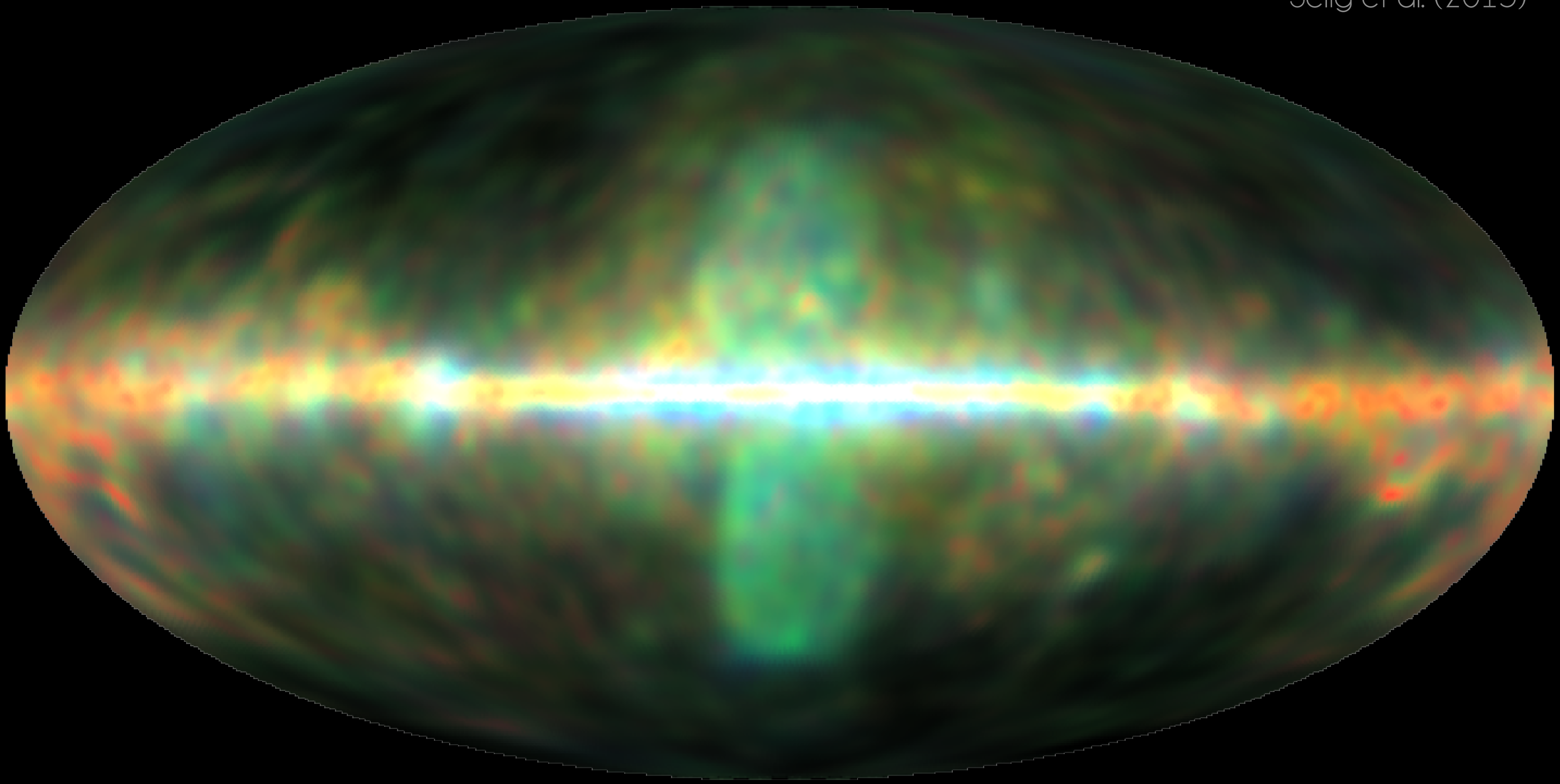


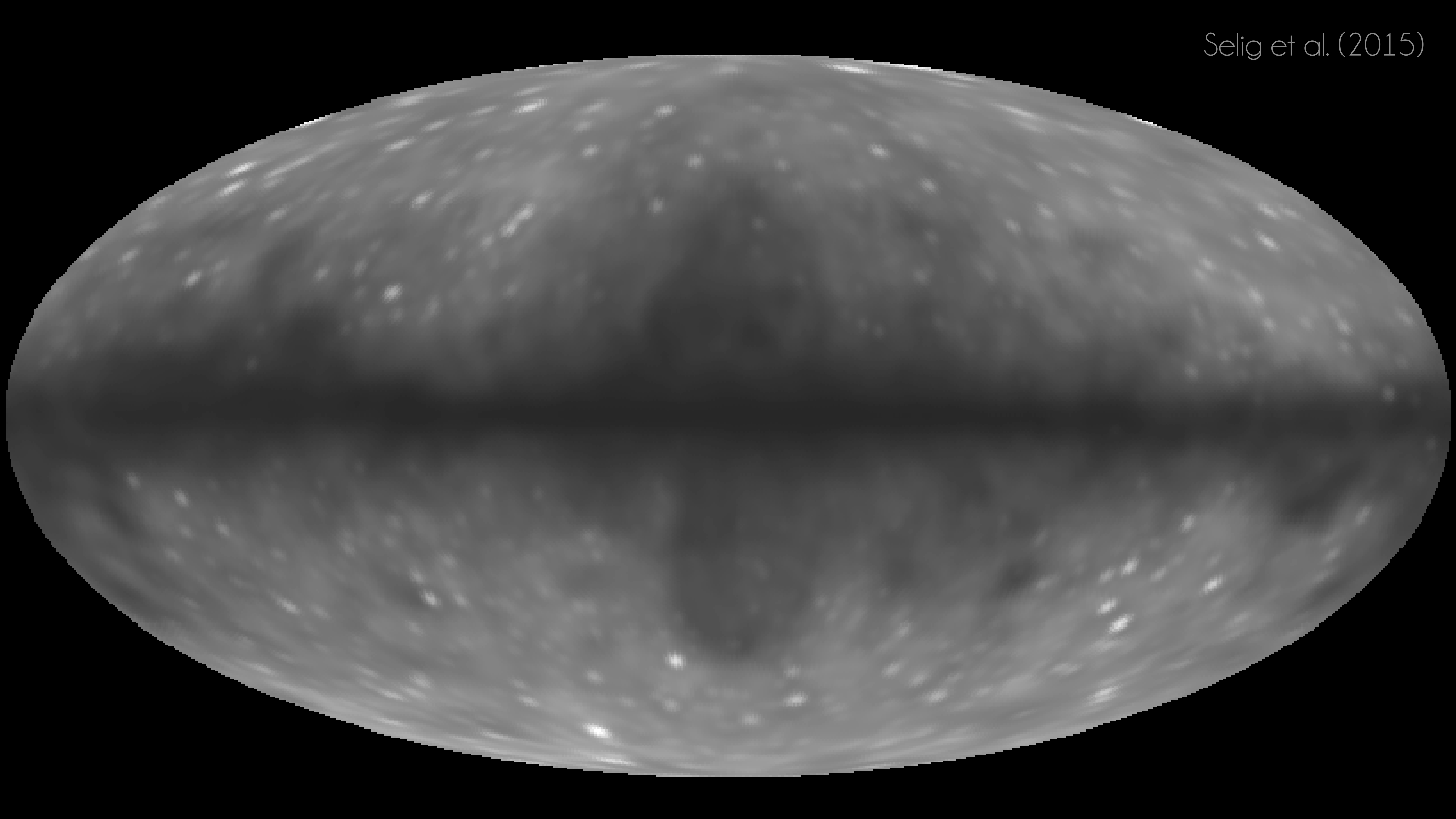
Selig et al. (2015)

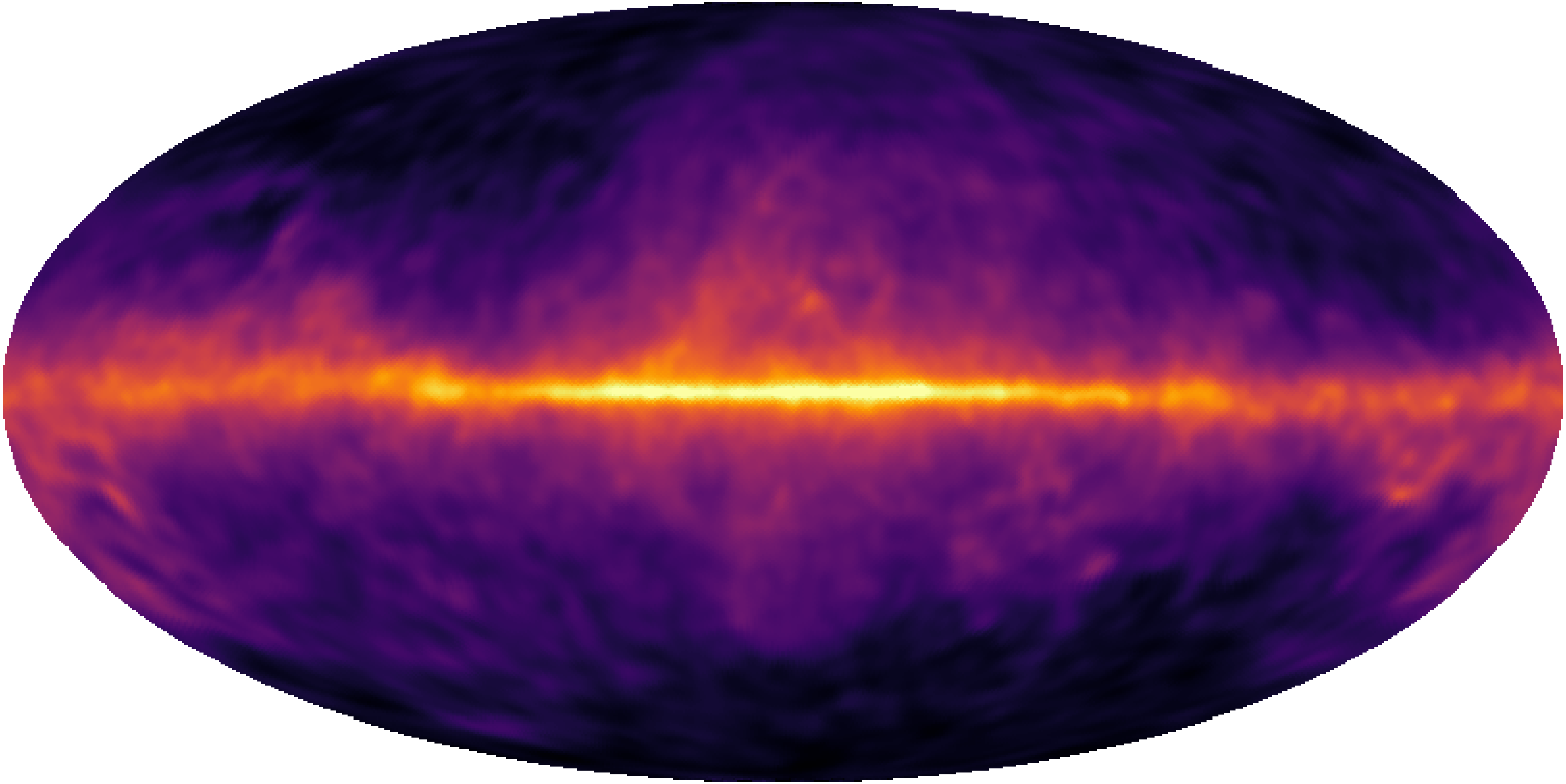

Selig et al. (2015)

Selig et al. (2015)

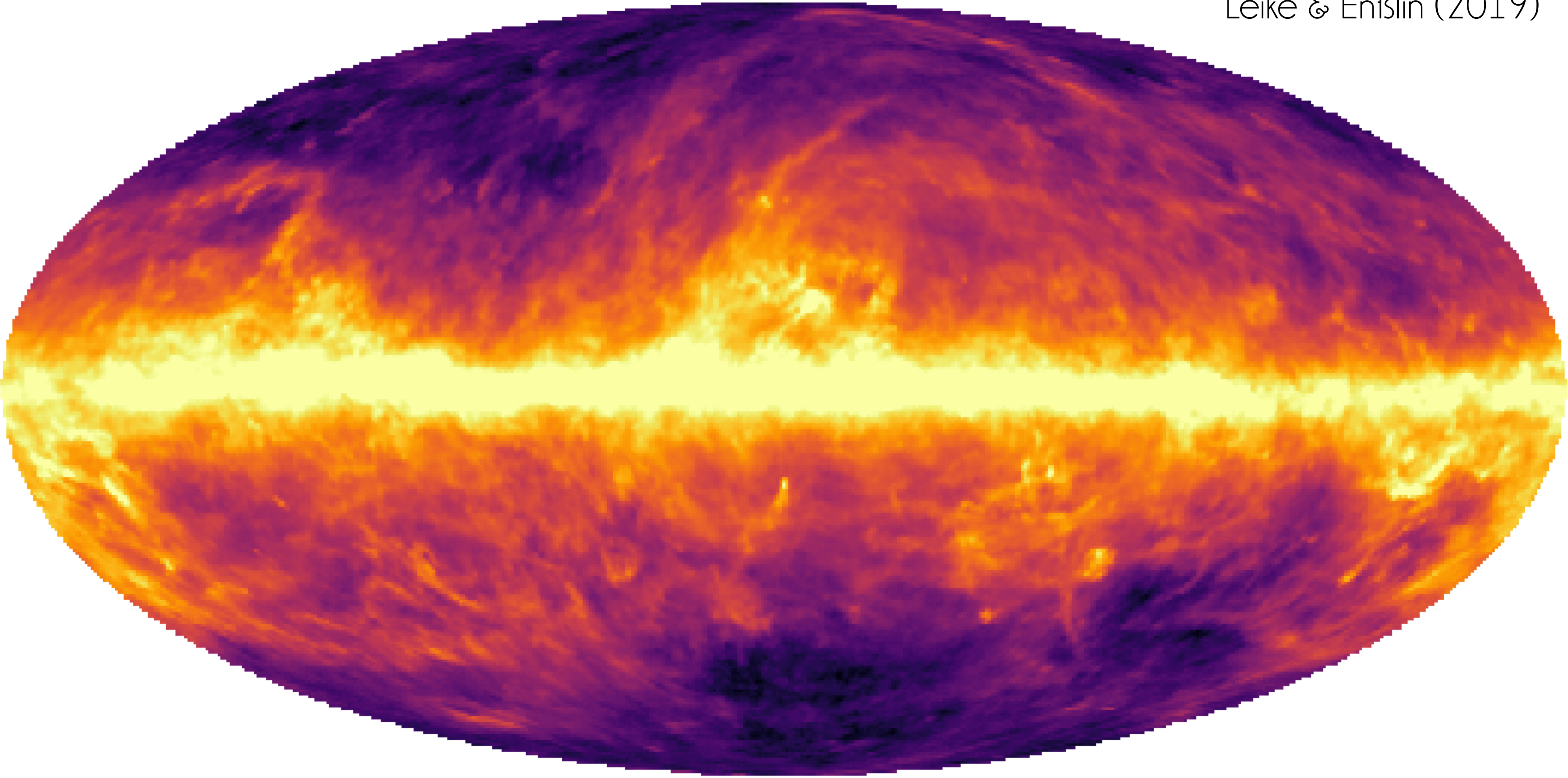



Selig et al. (2015)


Selig et al. (2015)

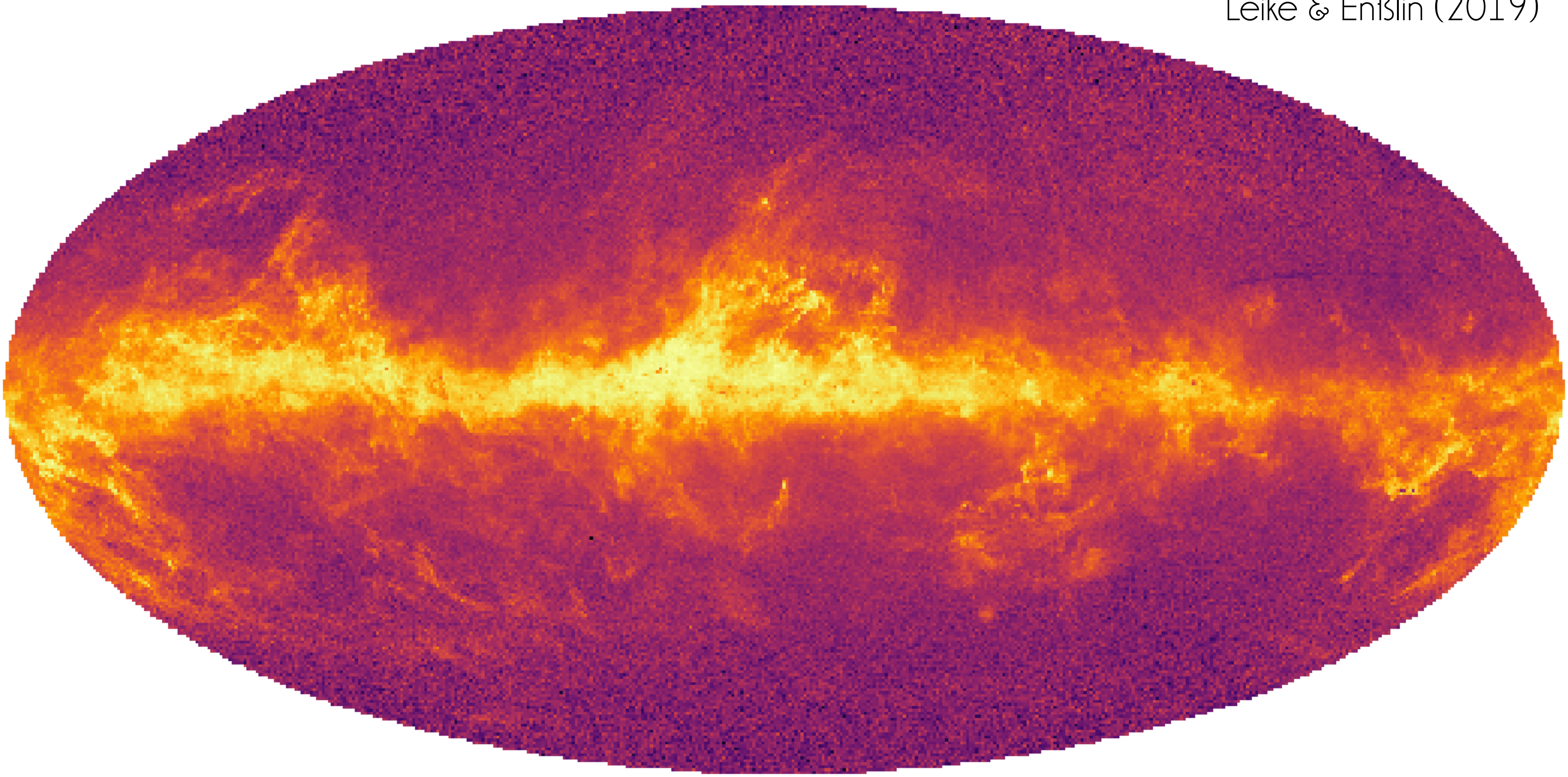


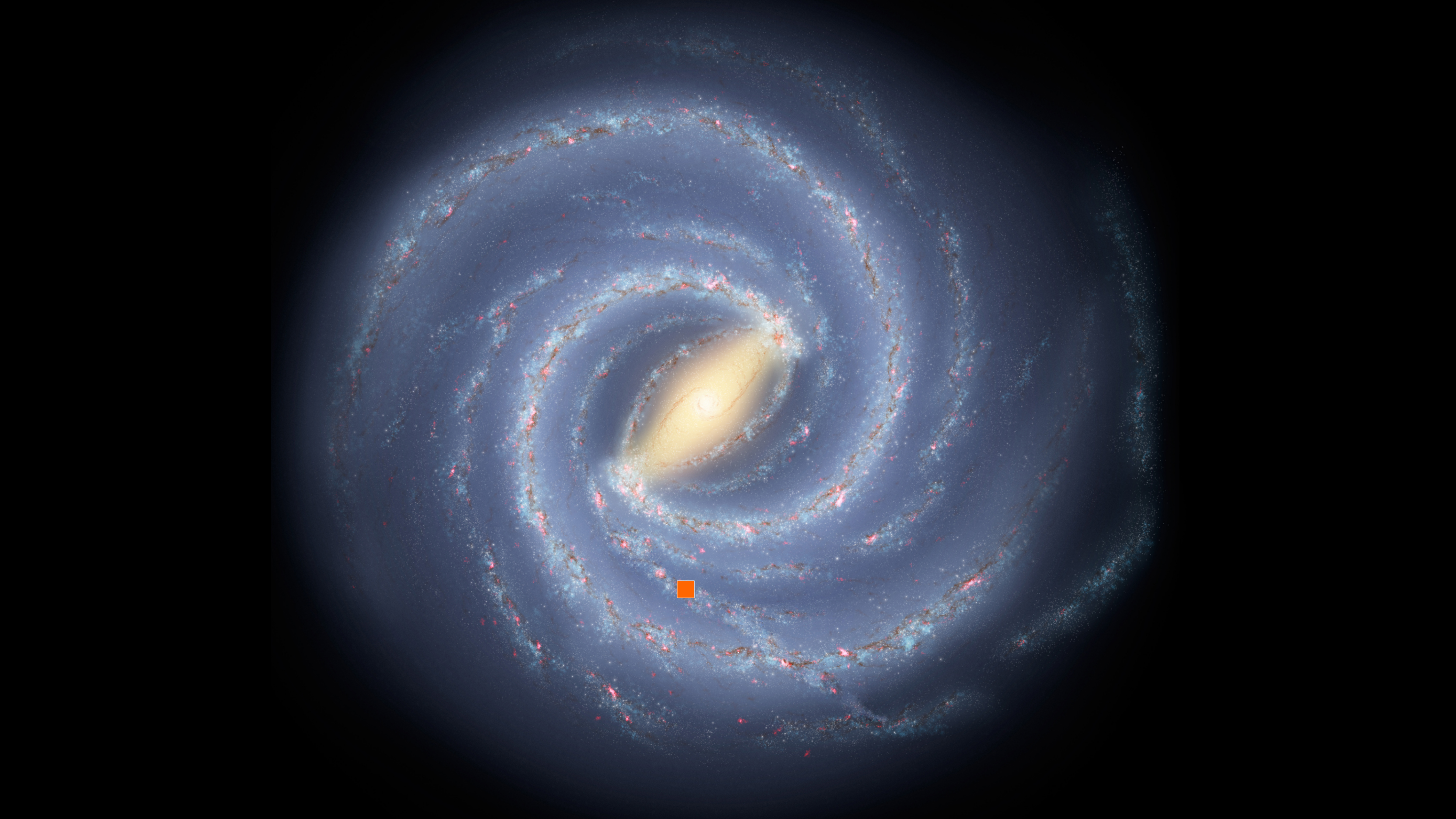
dust emission by IRAS + Planck
Leike & Enßlin (2019)



dust absorption by Gaia

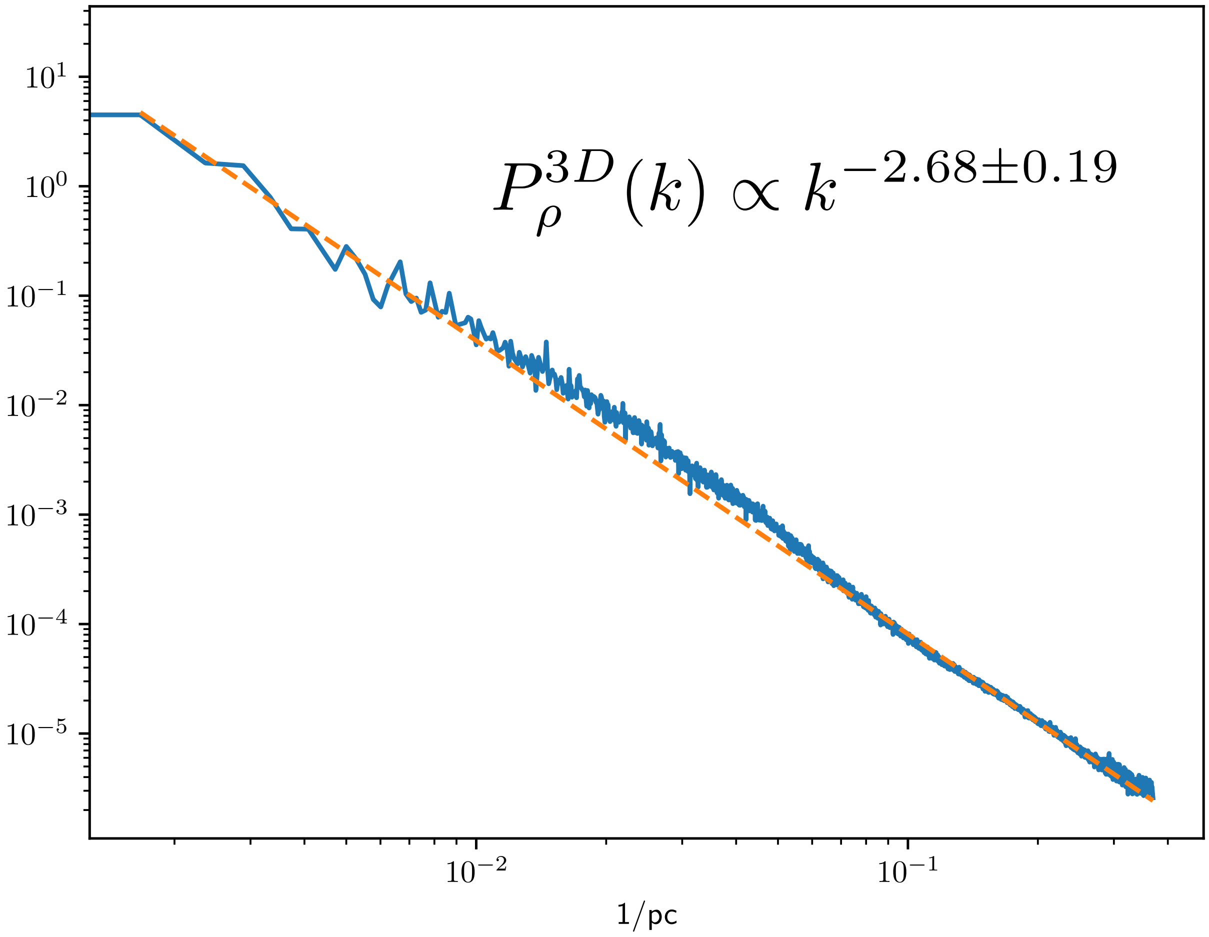
Leike & Enßlin (2019)





power spectrum

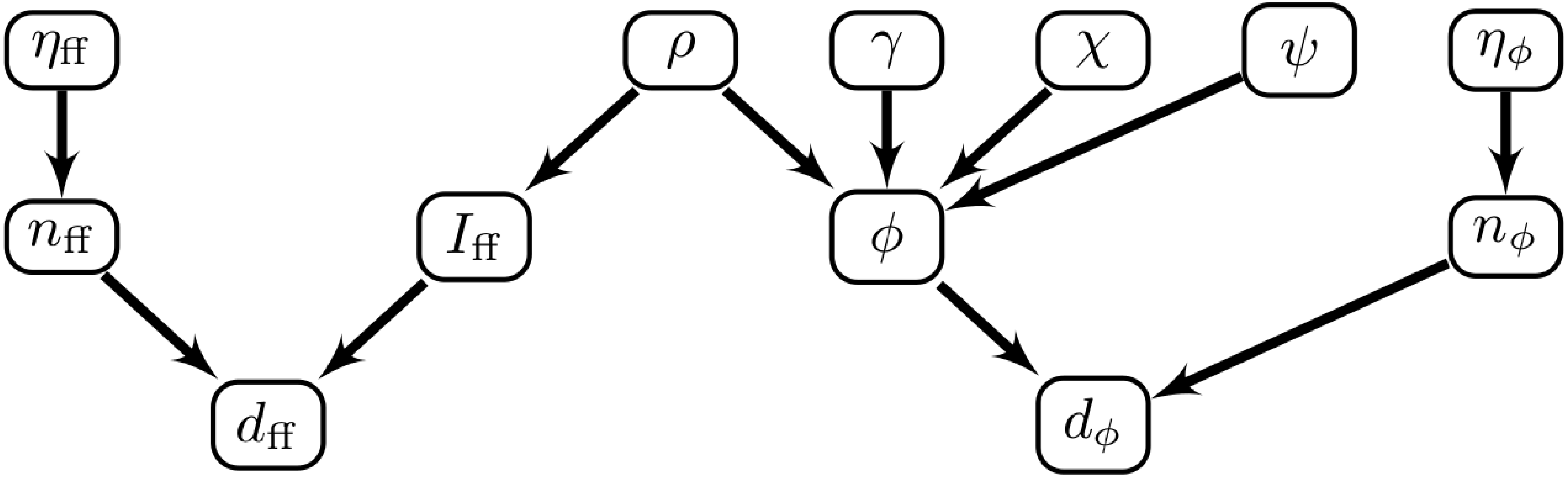
Leike & Enßlin. (2019)

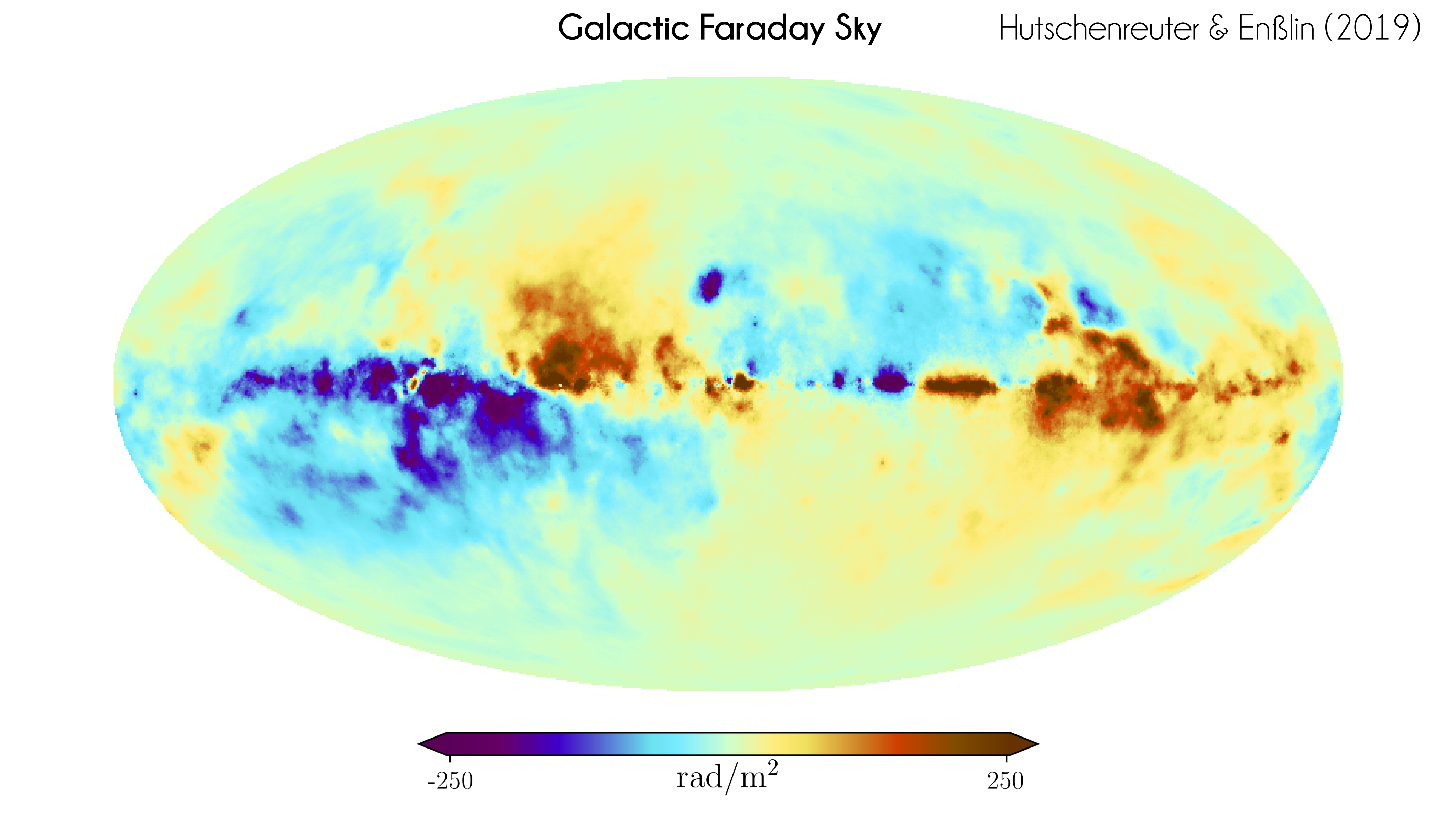


The physics of multiphase gas flows: fragmentation of a radiatively cooling gas cloud in a hot wind

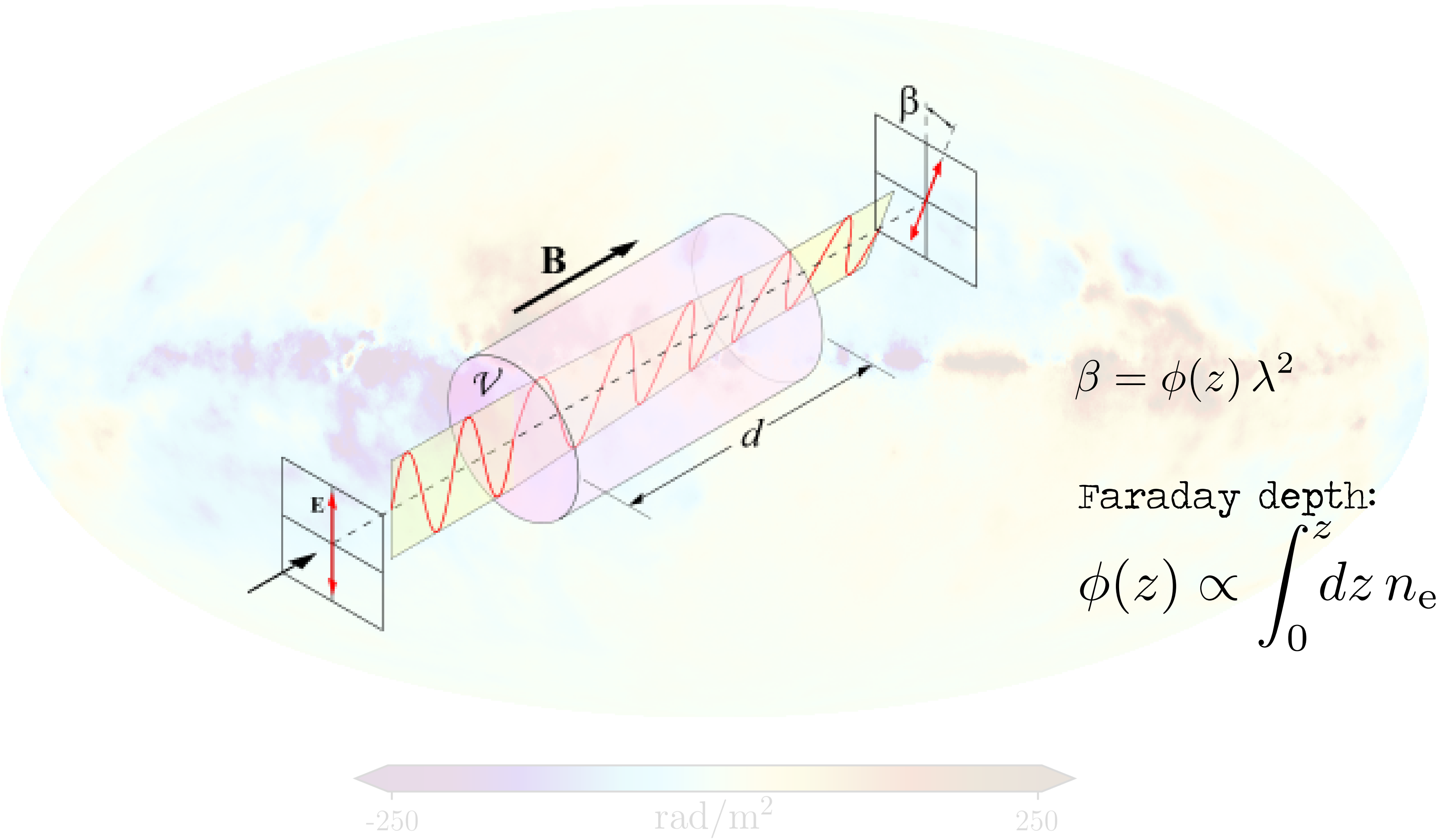
Martin Sparre[1,2,3★], Christoph Pfrommer[2,1] and Mark Vogelsberger[3]

Hierarchical Bayesian Model




Galactic Faraday Sky
Hutschenreuter & Enßlin (2019)
-250
rad/m²
250

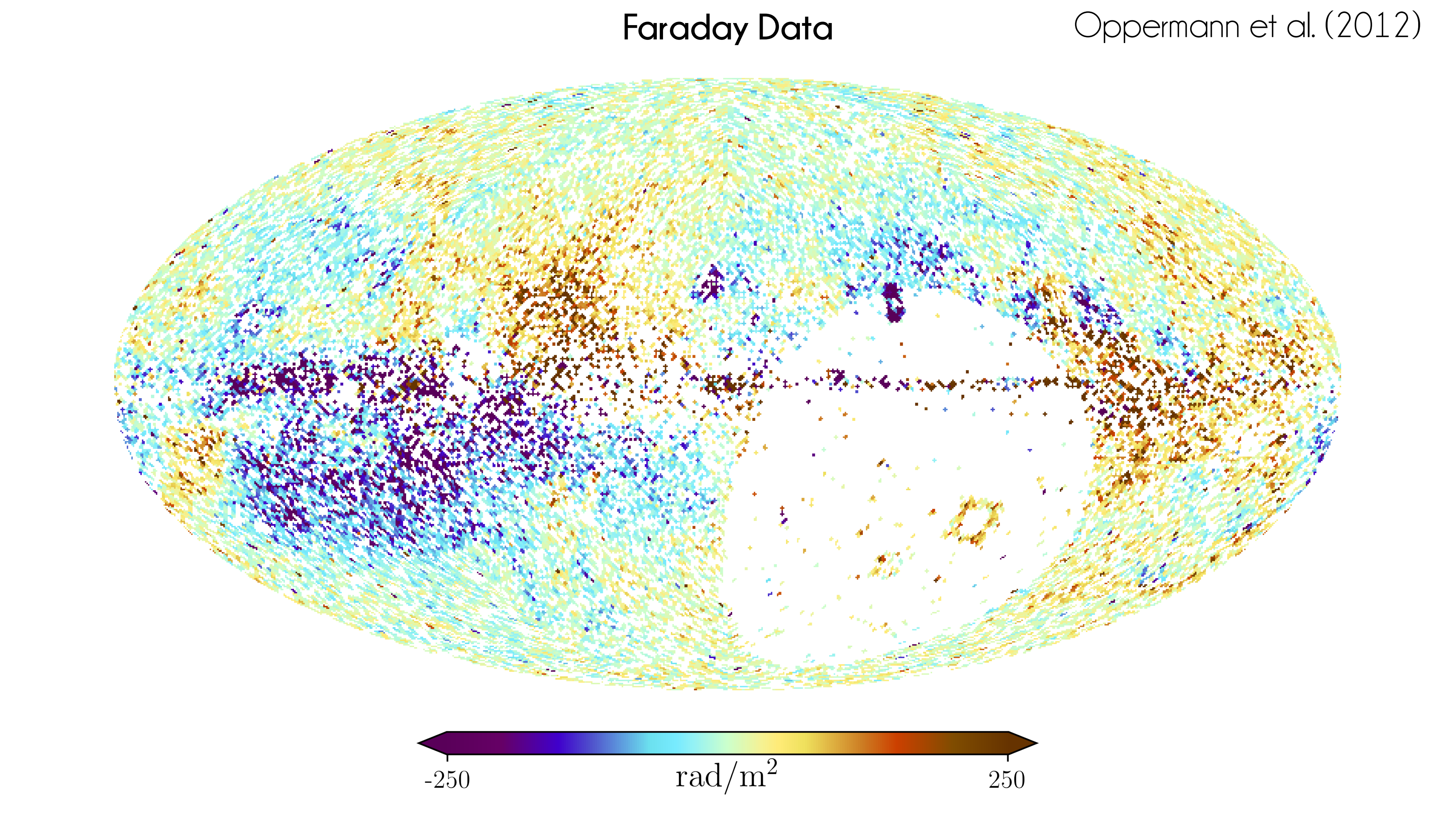
Faraday Effect

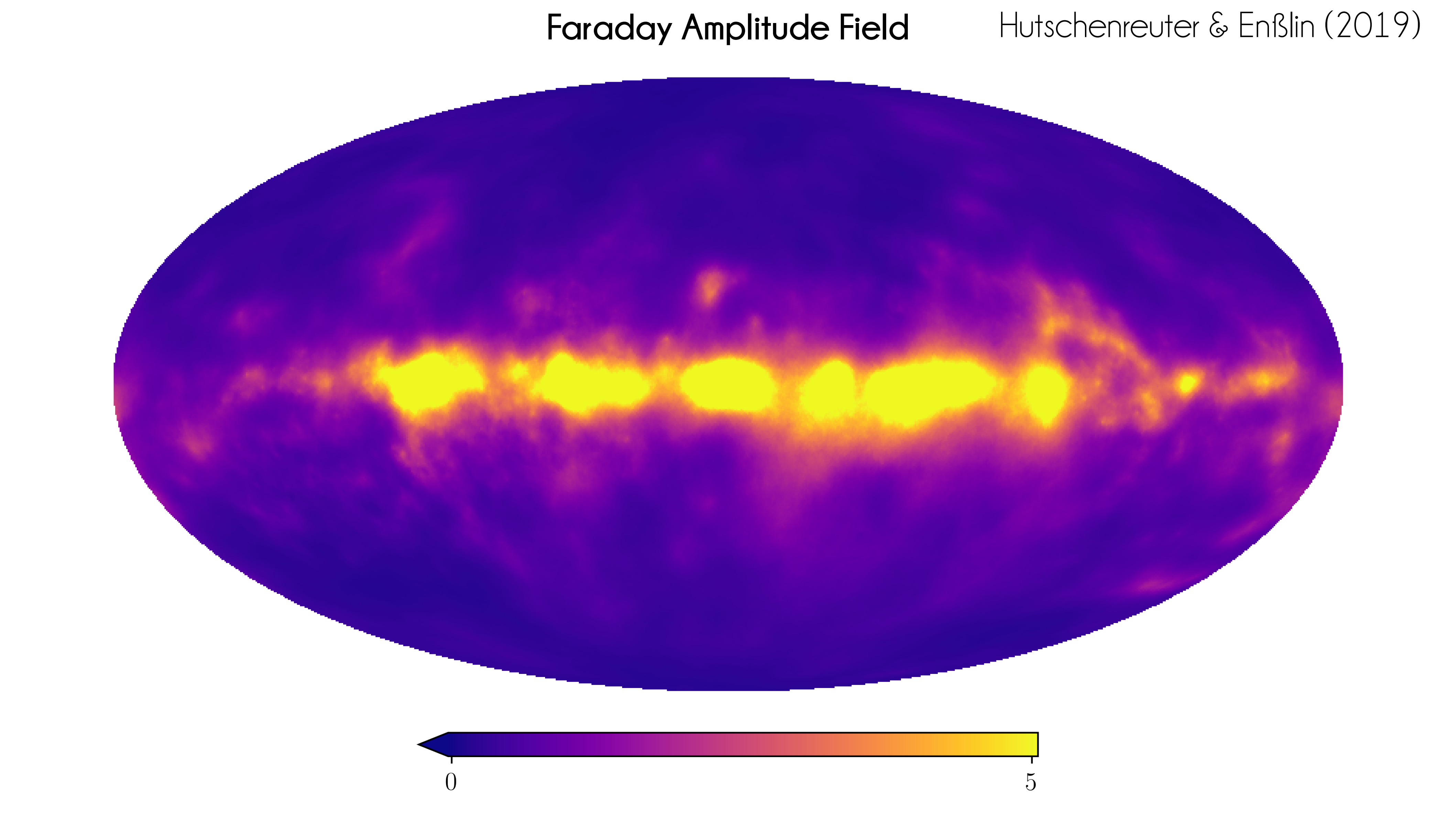


$$\beta = \phi(z)\,\lambda^2$$

Faraday depth:

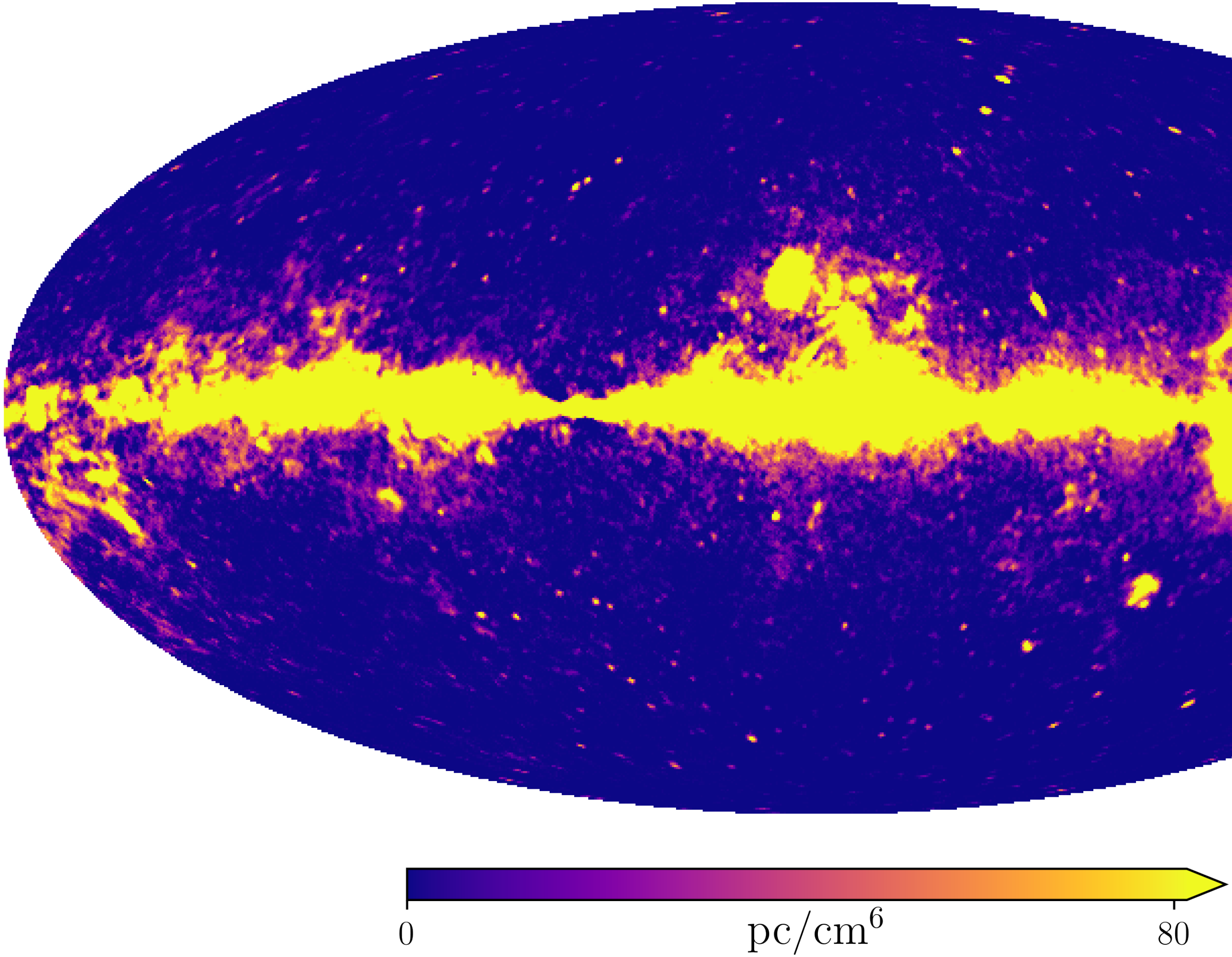
$$\phi(z) \propto \int_0^z dz\, n_{\mathrm{e}}\, B_z$$


Faraday Data
Oppermann et al. (2012)
-250
rad/m²
250


Faraday Amplitude Field
Hutschenreuter & Enßlin (2019)
0
5

Planck Free-Free Emission

Hutschenreuter & Enßlin (2019)

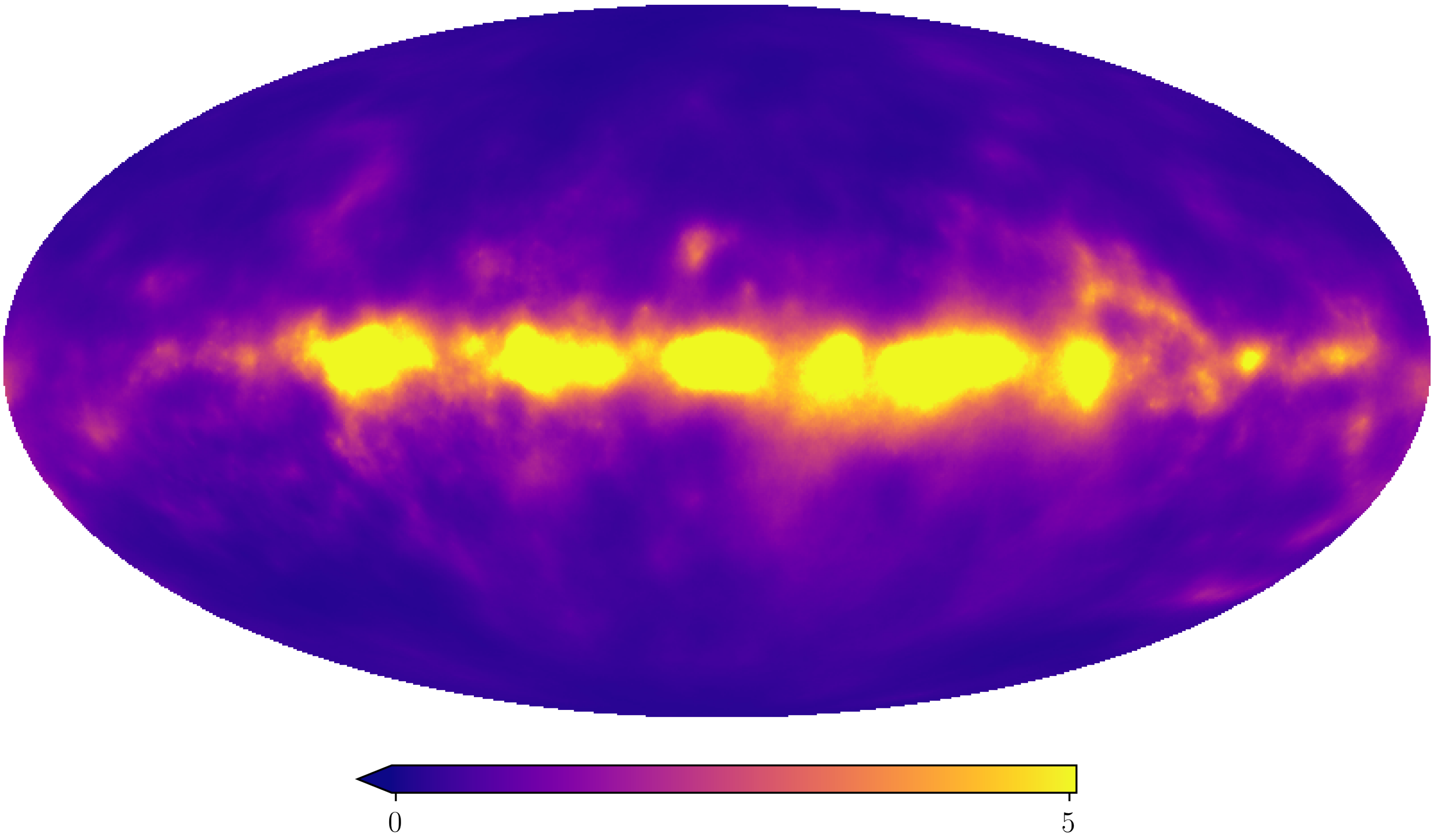


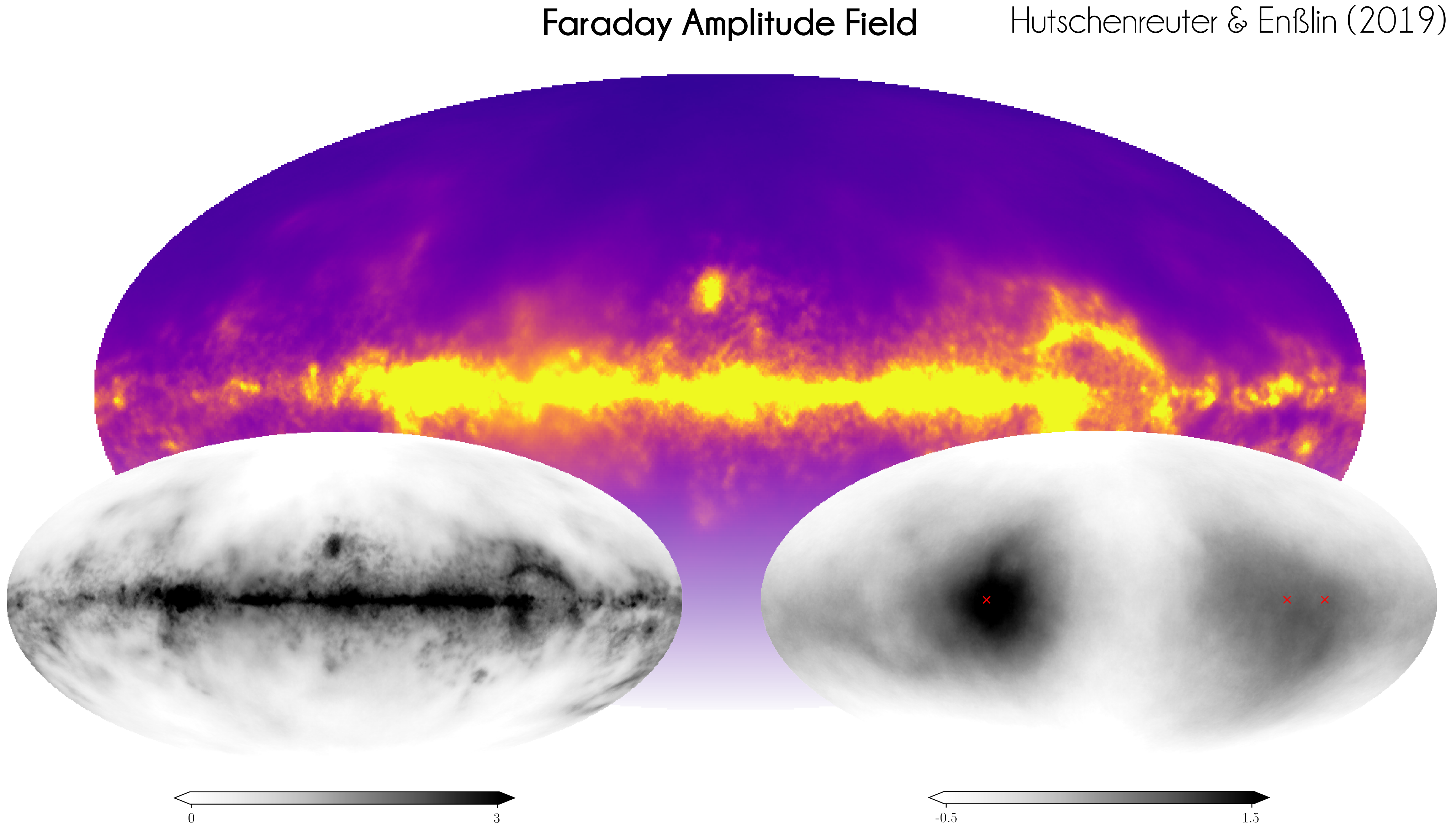
Faraday depth:

$$\phi(z) \propto \int_0^z dz\, n_e B_z$$

Faraday Amplitude Field

Hutschenreuter & Enßlin (2019)

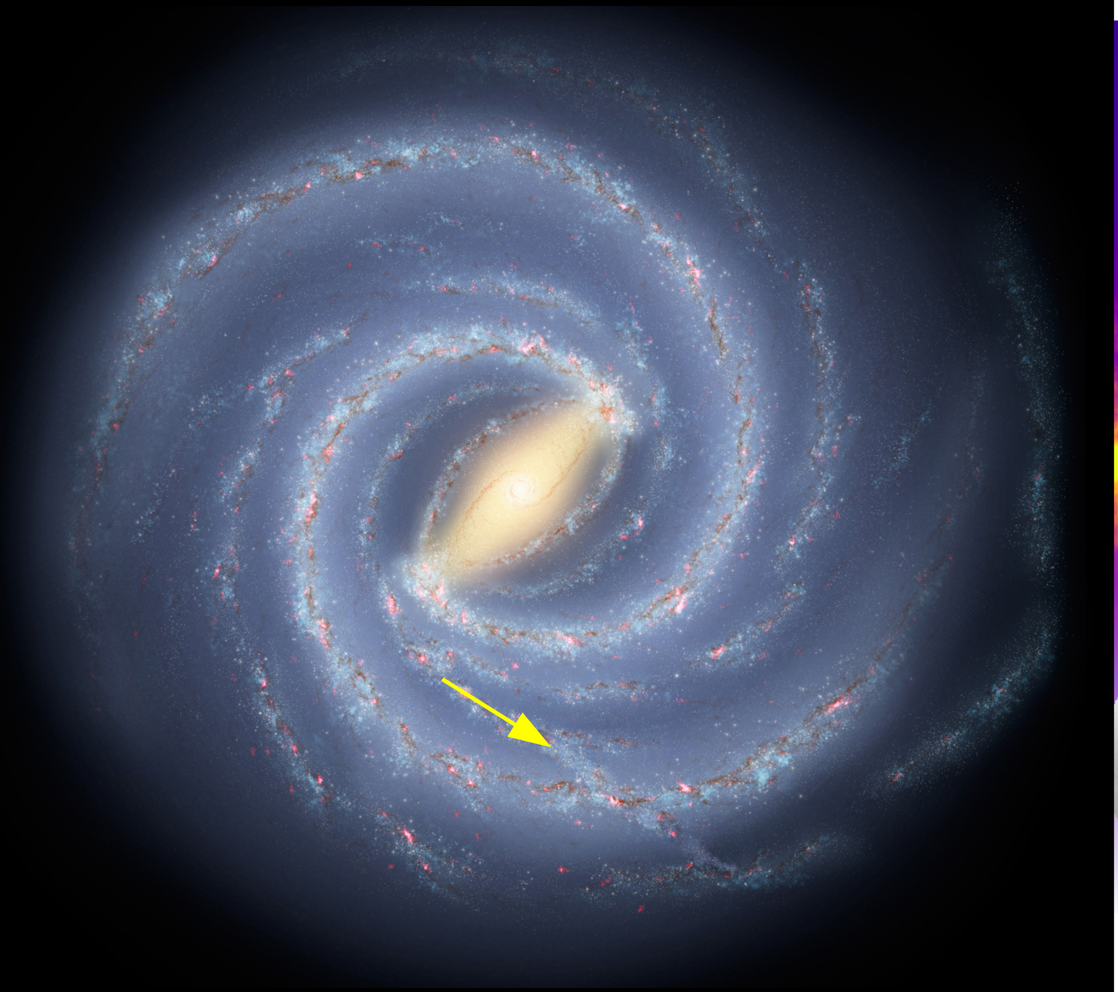


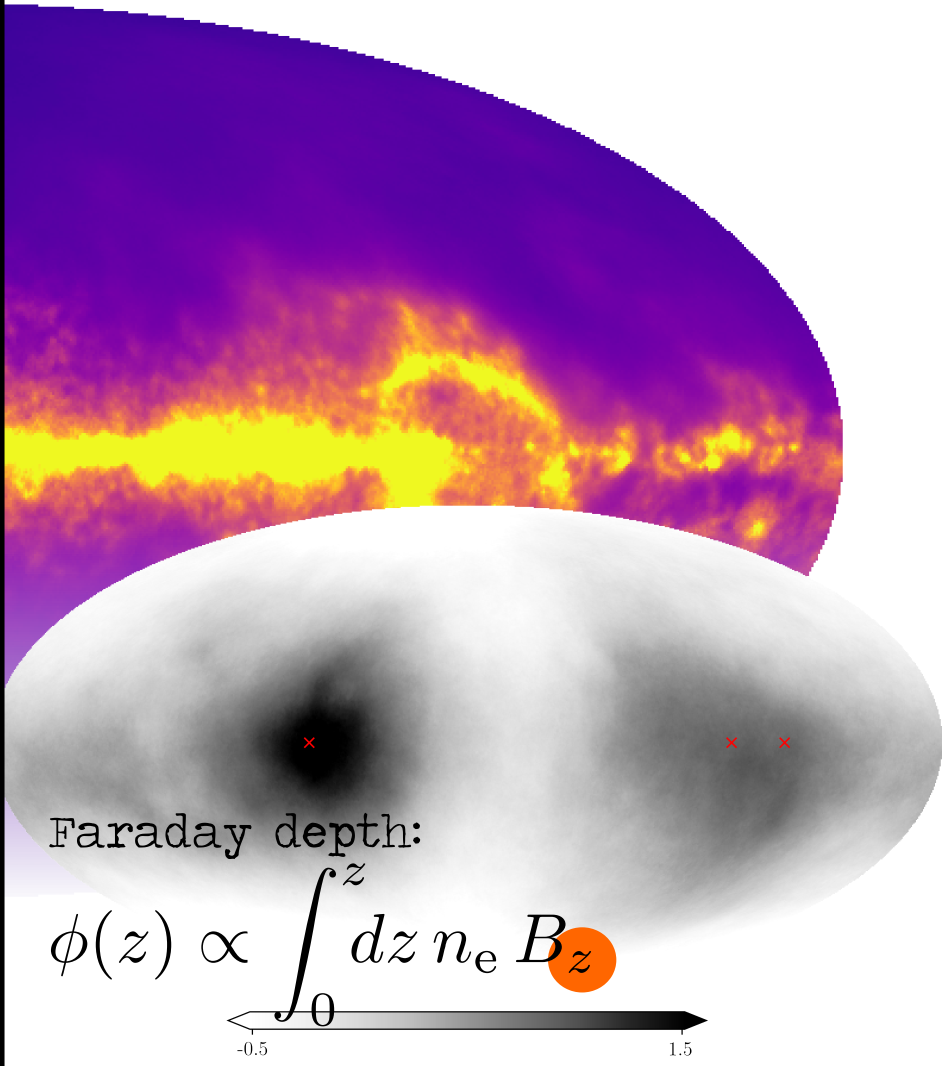


Faraday Amplitude Field
Hutschenreuter & Enßlin (2019)
0
3
-0.5
1.5

tude Field

Hutschenreuter & Enßlin (2019)



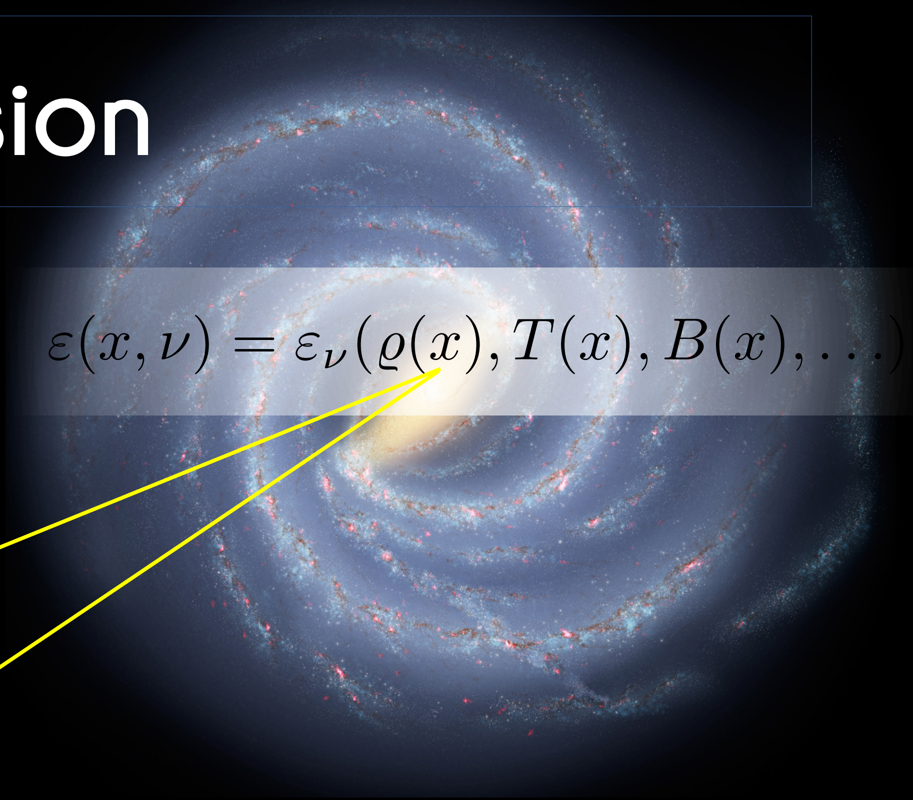


Faraday depth:

$$\phi(z) \propto \int_0^z dz\, n_\mathrm{e}\, B_z$$

Data Fusion

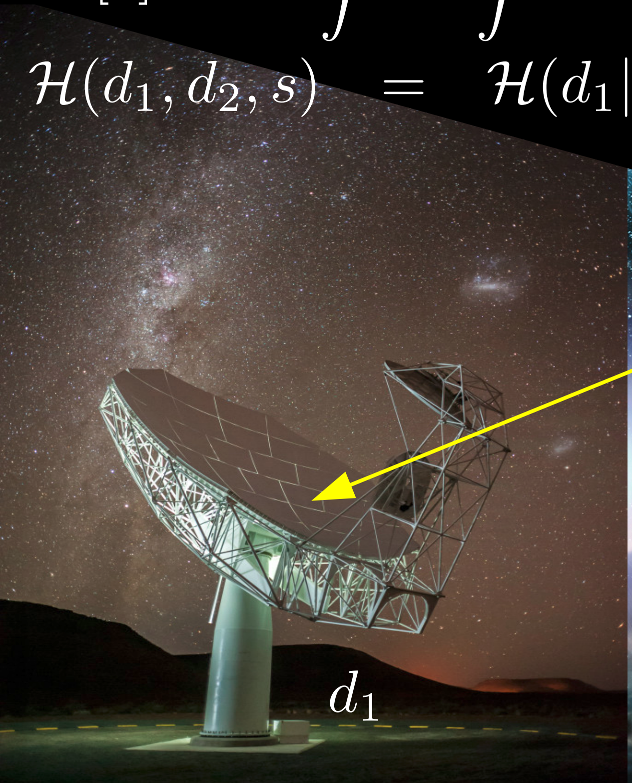
$$
\begin{aligned}
d_i &= R_i[\varepsilon] + n_i \\
R_i[\varepsilon] &= \int \mathrm{d}x \int \mathrm{d}\nu \, R_i(x,\nu)\, \varepsilon(x,\nu) \\
\mathcal{H}(d_1, d_2, s) &= \mathcal{H}(d_1|s) + \mathcal{H}(d_2|s) + \mathcal{H}(s)
\end{aligned}
$$



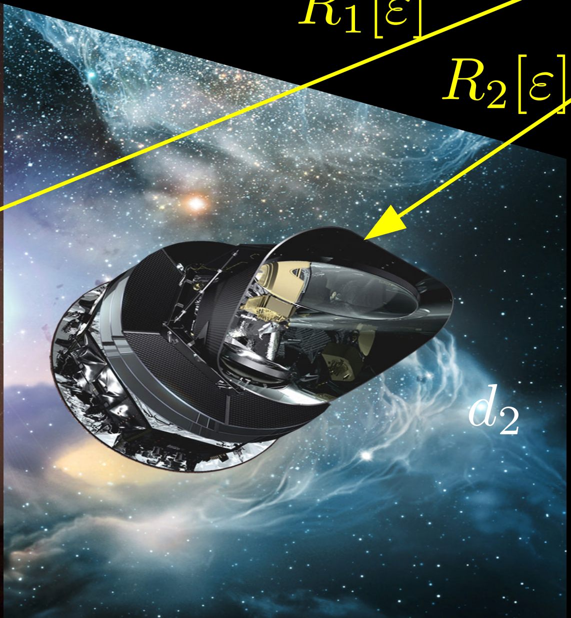
$$\varepsilon(x,\nu) = \varepsilon_\nu(\varrho(x), T(x), B(x), \ldots)$$

$R_1[\varepsilon]$

$R_2[\varepsilon]$

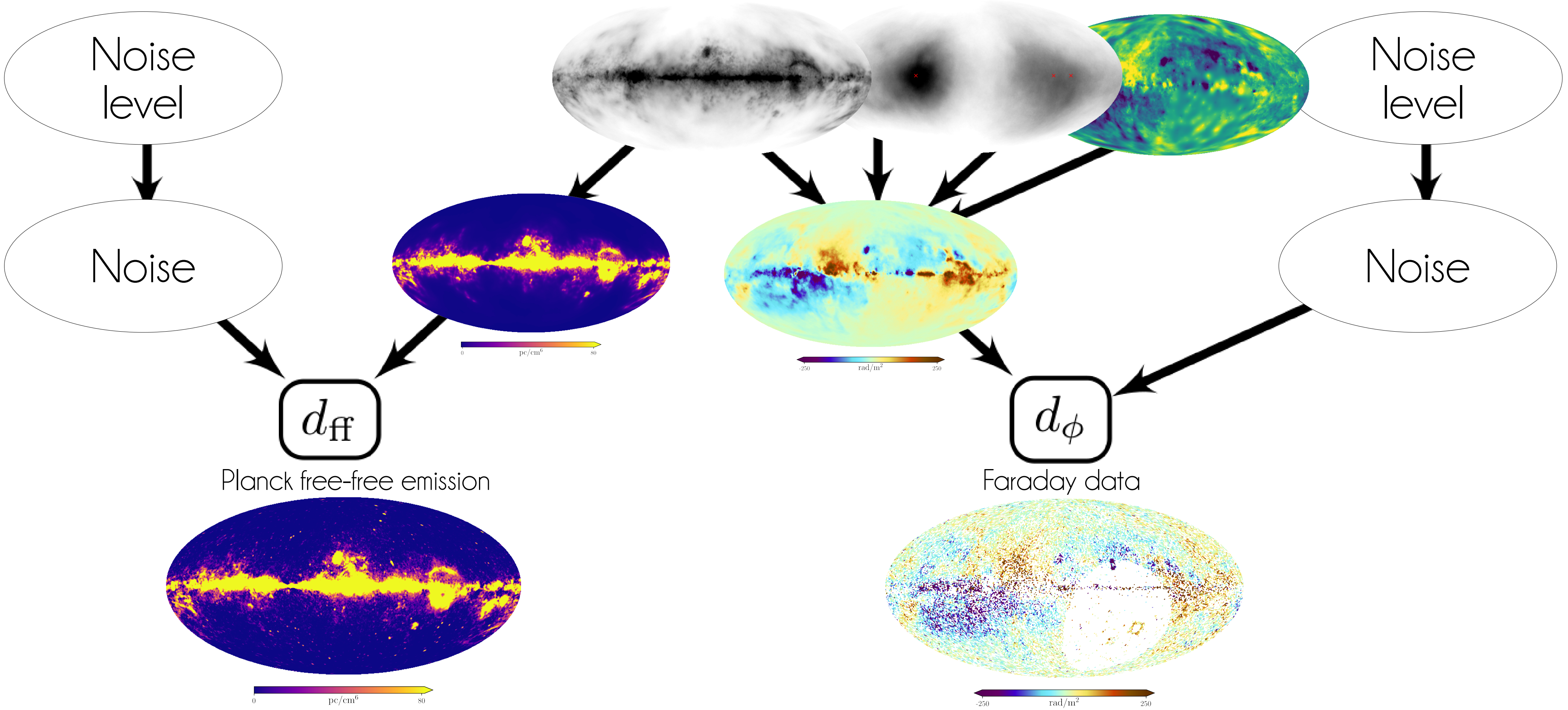


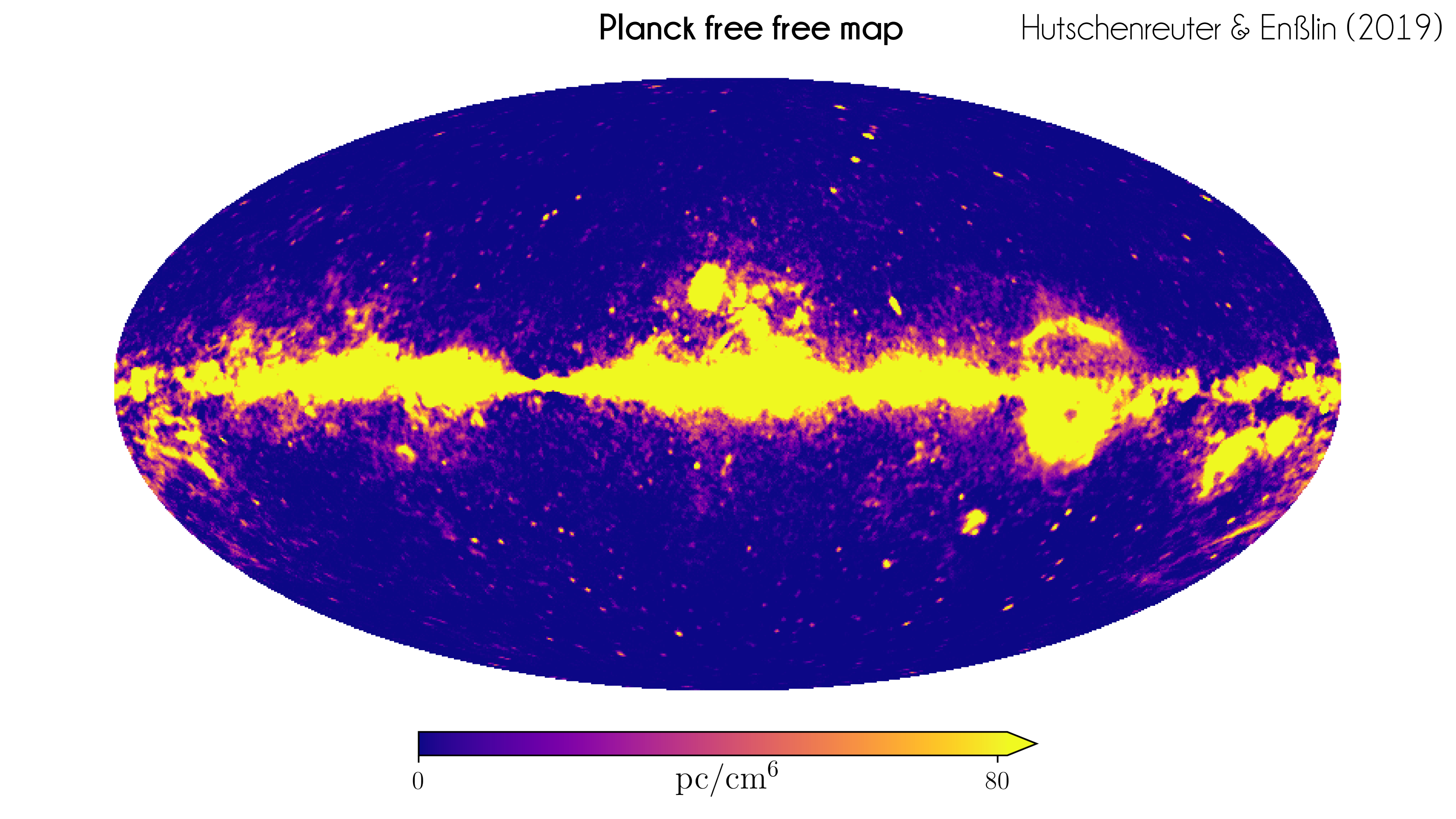
d_1

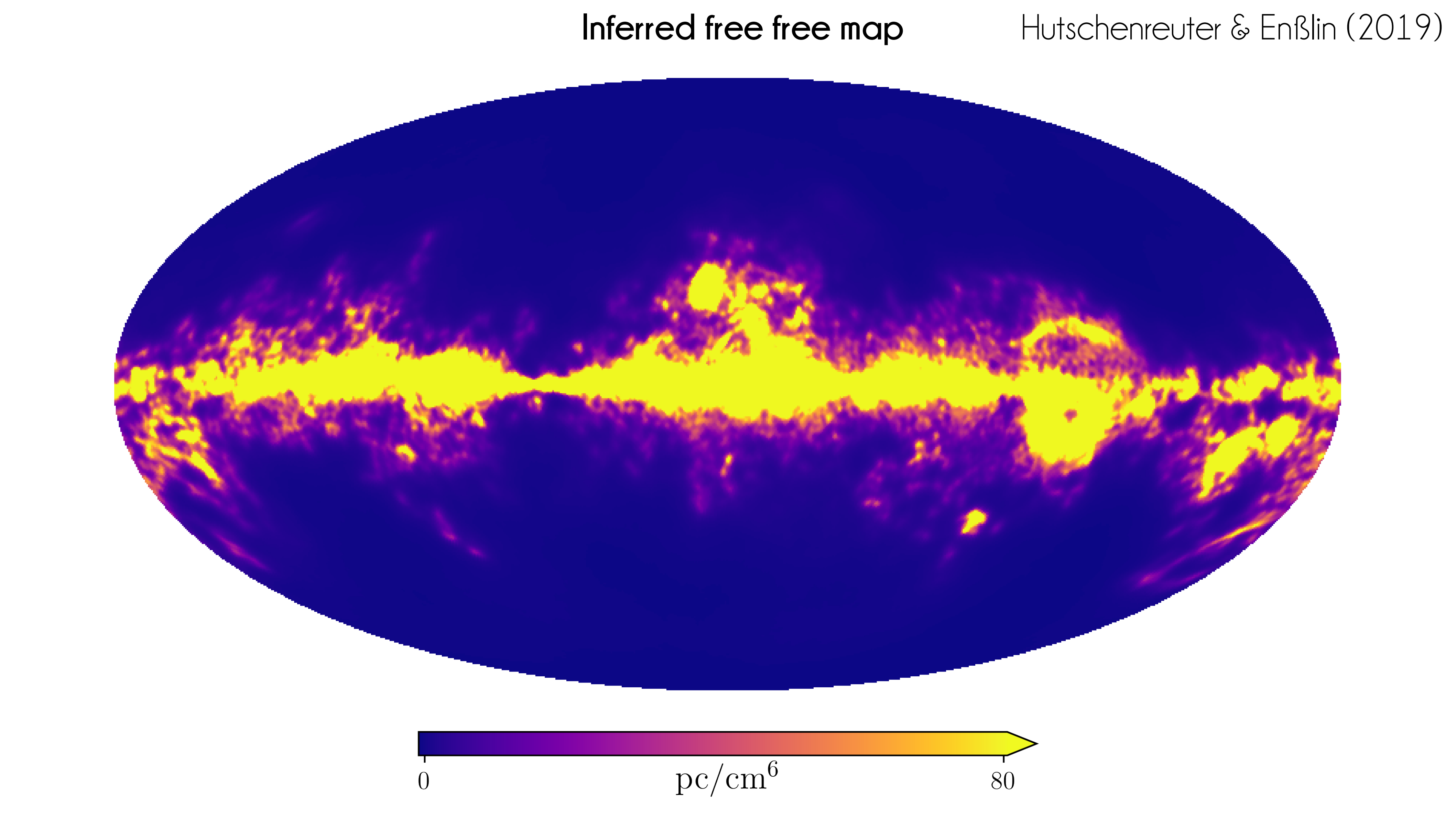


d_2

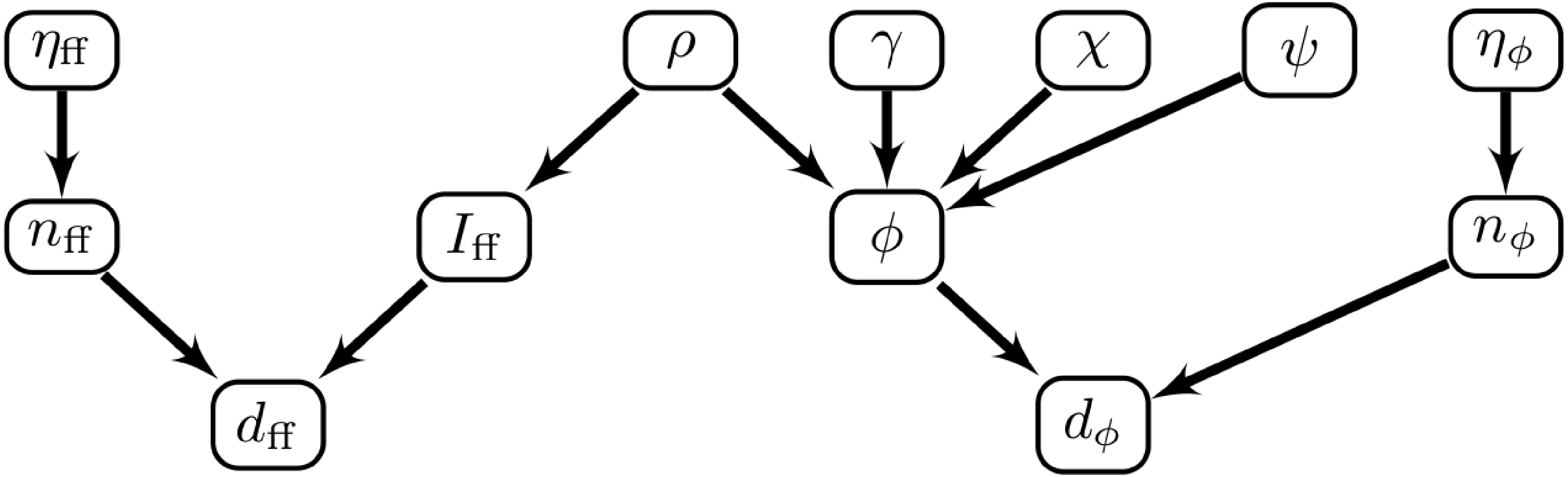
Hierarchical Bayesian Model




Planck free free map
Hutschenreuter & Enßlin (2019)
0
pc/cm6
80


Inferred free free map
Hutschenreuter & Enßlin (2019)
0
pc/cm^6
80

Hierarchical Bayesian Model



NIFTy tutorial part 2

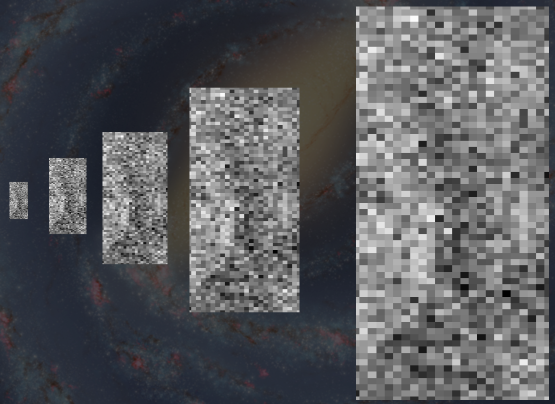
nonlinear reconstructions

Information Field Theory

+ Numerical Information Field Theory

IFT is imaging

IFT is AI



IFT is for you

NIFTy software package:
ift.pages.mpcdf.de/nifty

IFT resource page:
wwwmpa.mpa-garching.mpg.de/ift