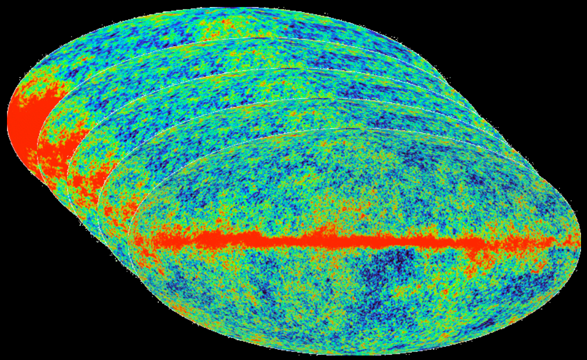
Foreground removal with Commander

Hans Kristian Eriksen
Institute of Theoretical Astrophysics, Oslo
June 24, 2008

See Eriksen et al. 2008, ApJ, 676, 10 (astro-ph:0709.1058) for full details

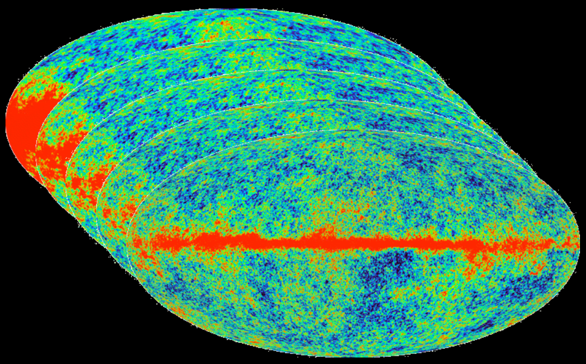
Traditional CMB analysis

Traditional CMB analysis

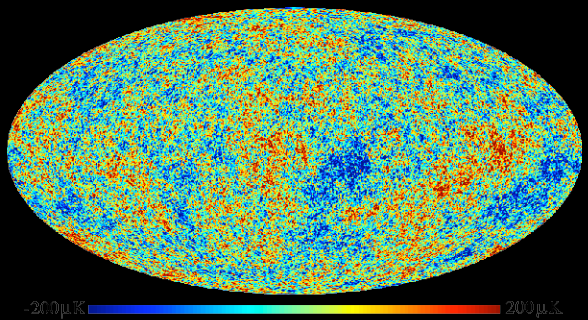


Frequency maps

Traditional CMB analysis



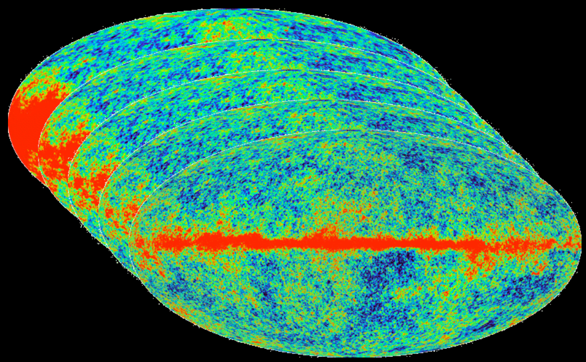
Simple foreground removal (e.g., template fitting)



Frequency maps

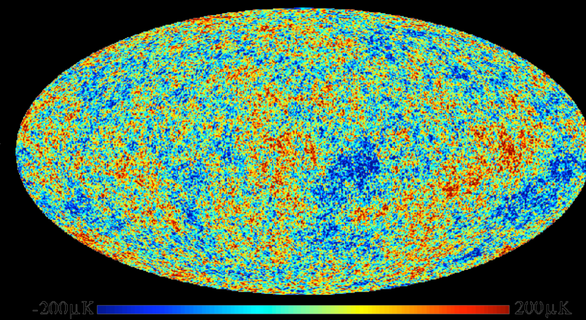
Cleaned CMB map

Traditional CMB analysis



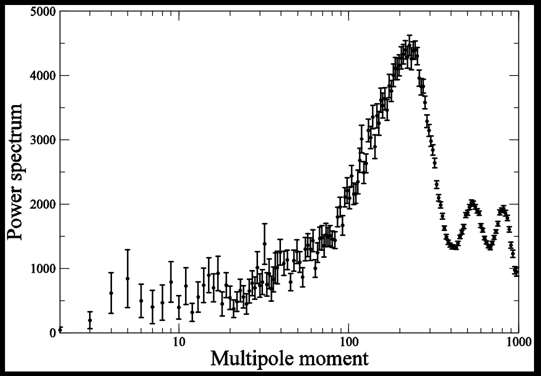
Frequency maps

Simple foreground removal (e.g., template fitting)



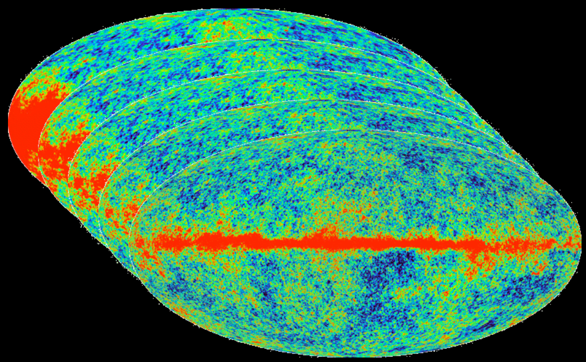
Cleaned CMB map

"Generalized FFT"



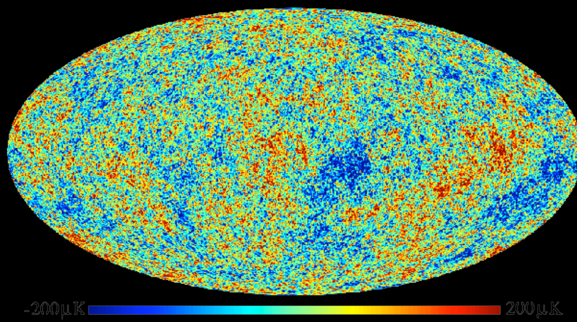
Power spectrum

Traditional CMB analysis



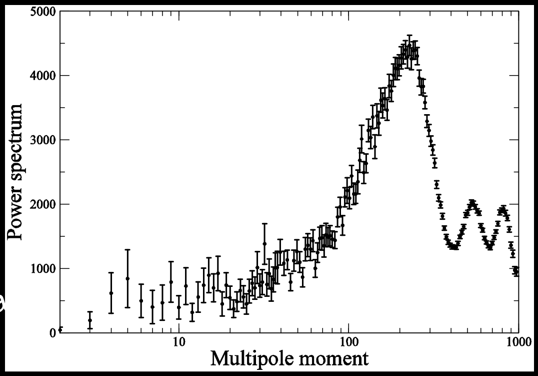
Frequency maps

Simple foreground removal (e.g., template fitting)



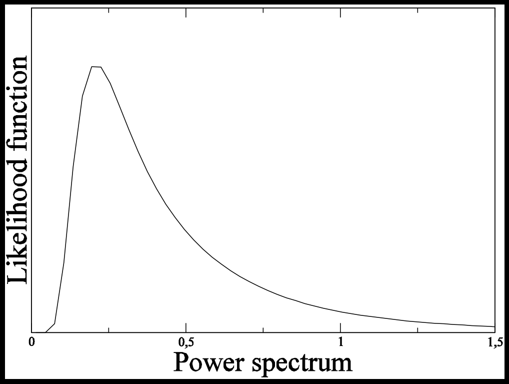
Cleaned CMB map

”Generalized FFT”



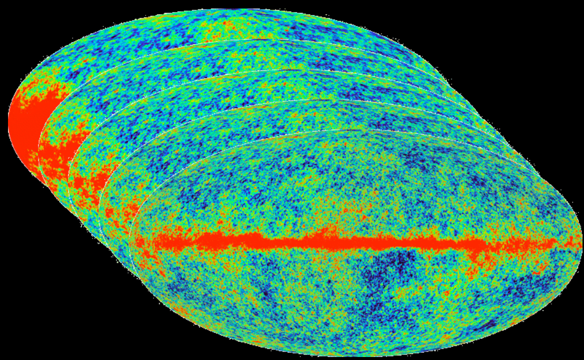
Power spectrum

Likelihood fitting formula



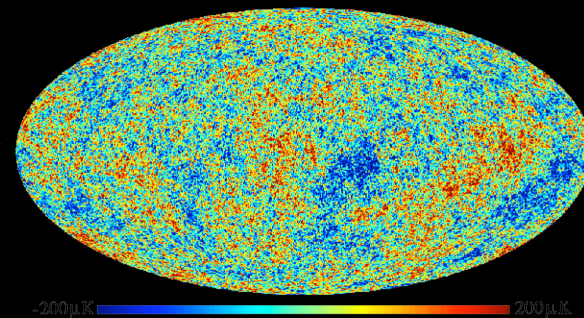
Likelihood

Traditional CMB analysis



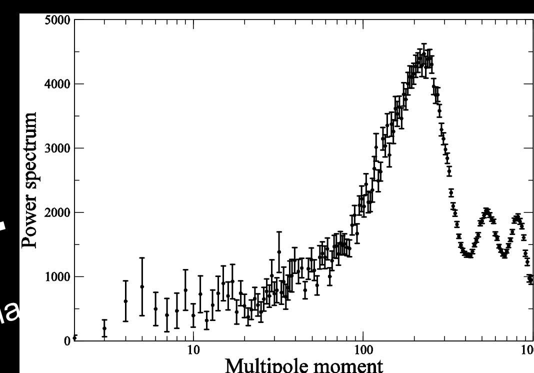
Frequency maps

Simple foreground removal (e.g., template fitting)



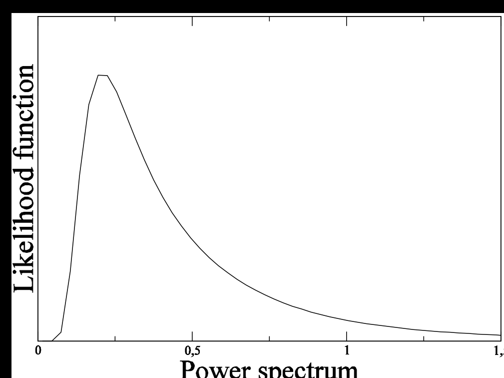
Cleaned CMB map

"Generalized FFT"



Power spectrum

Likelihood fitting formula



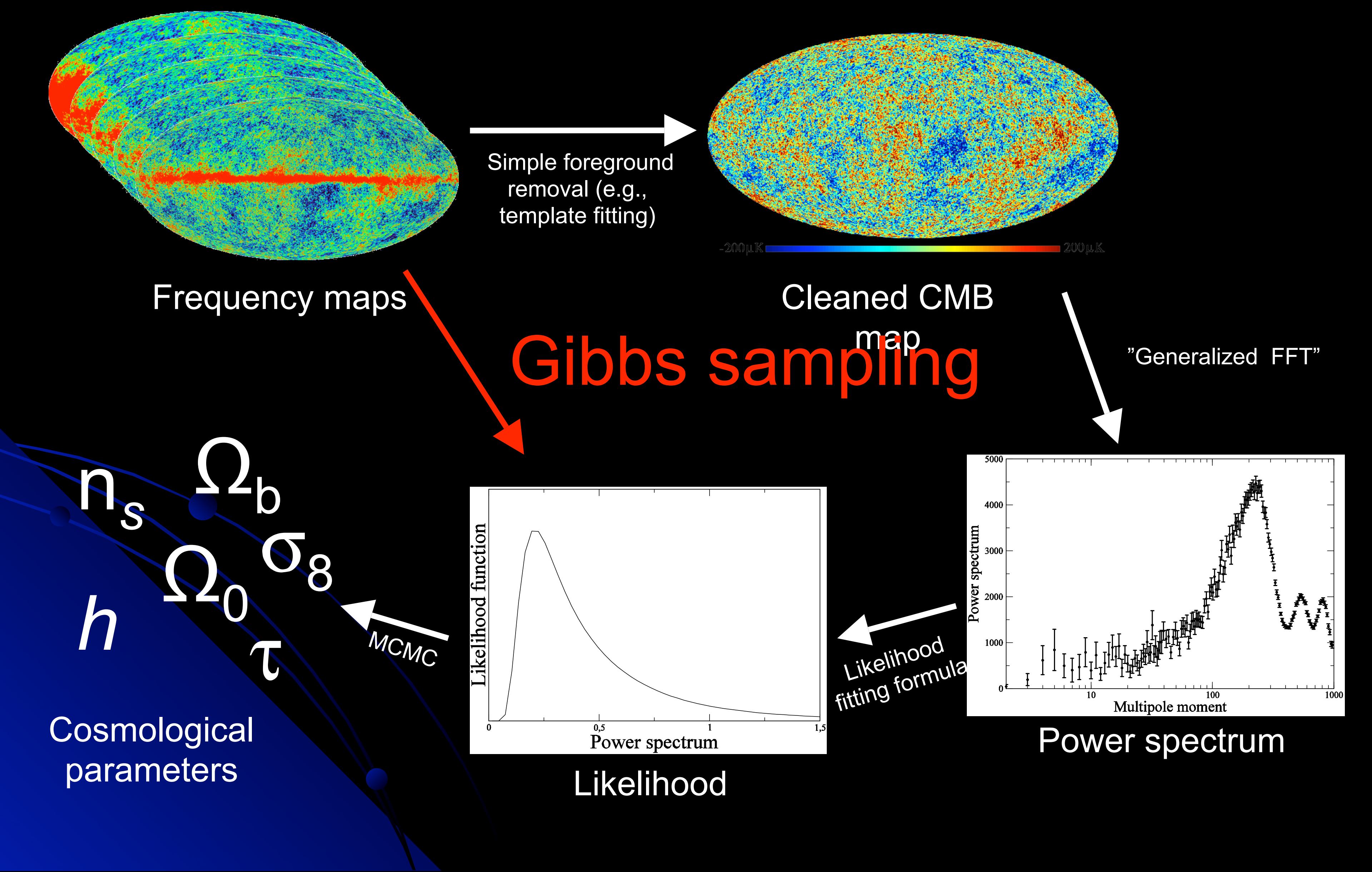
Likelihood

MCMC

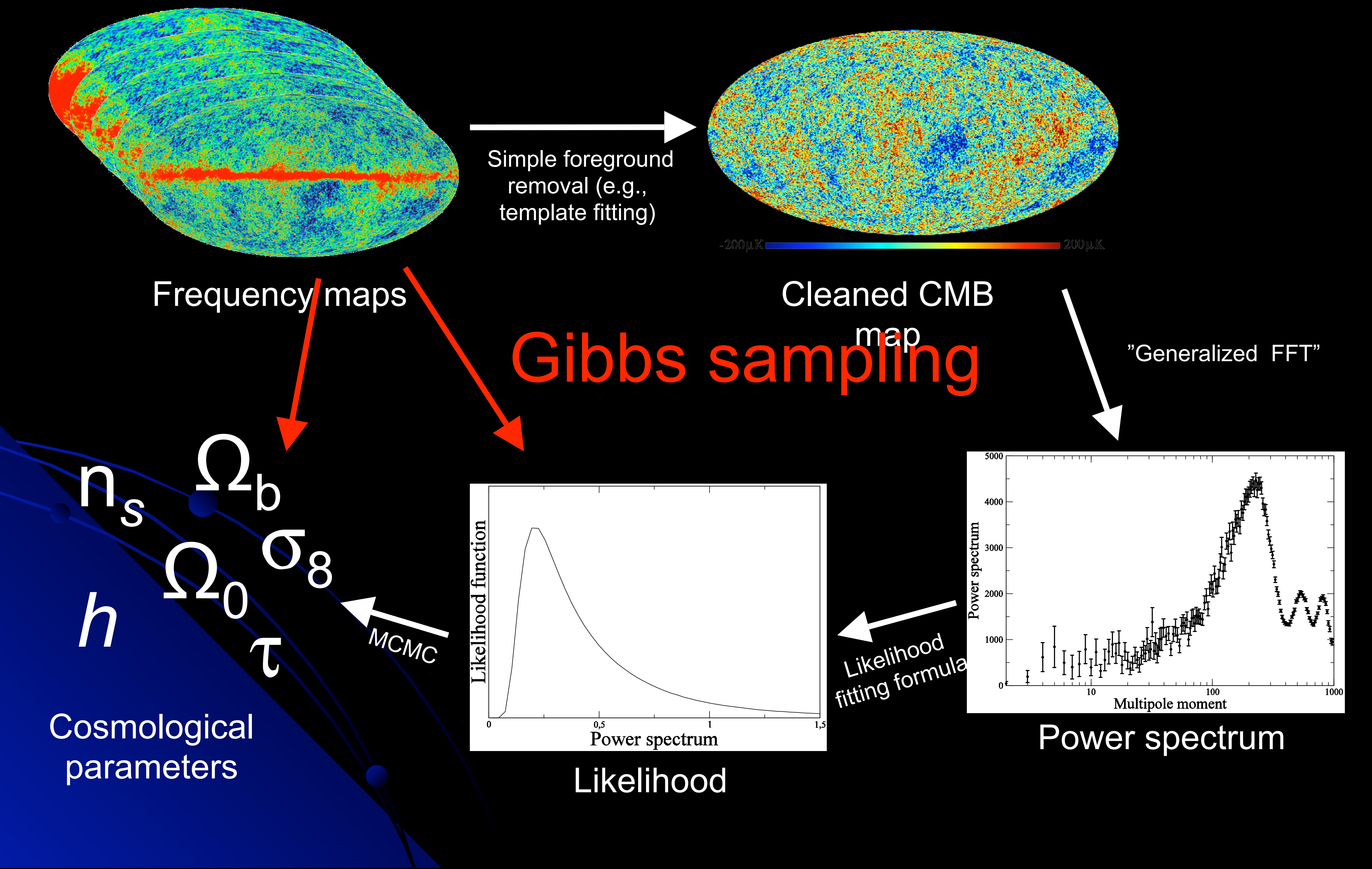
n_s Ω_b σ_8 Ω_0 h τ

Cosmological parameters

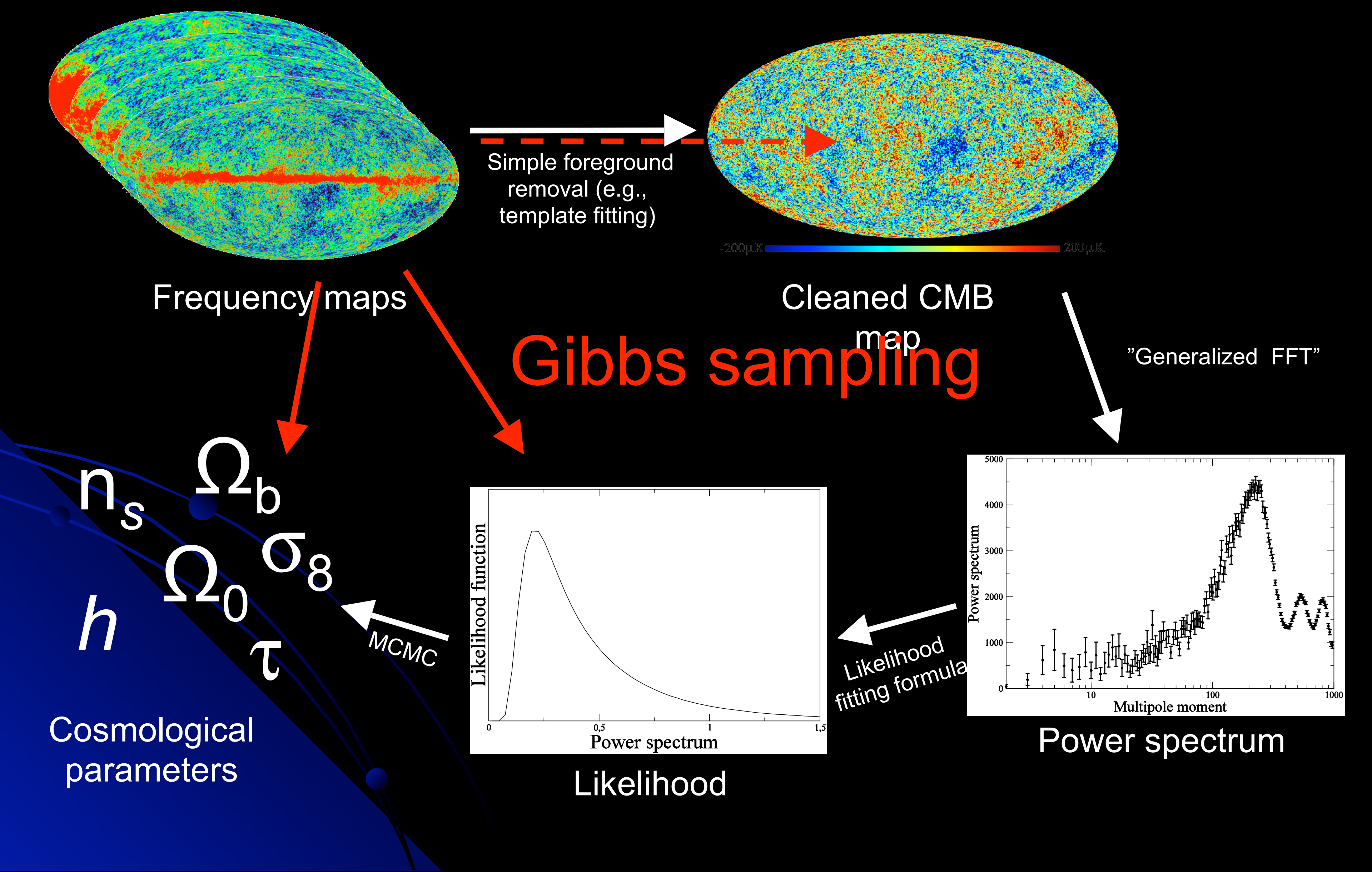
CMB analysis by Gibbs sampling



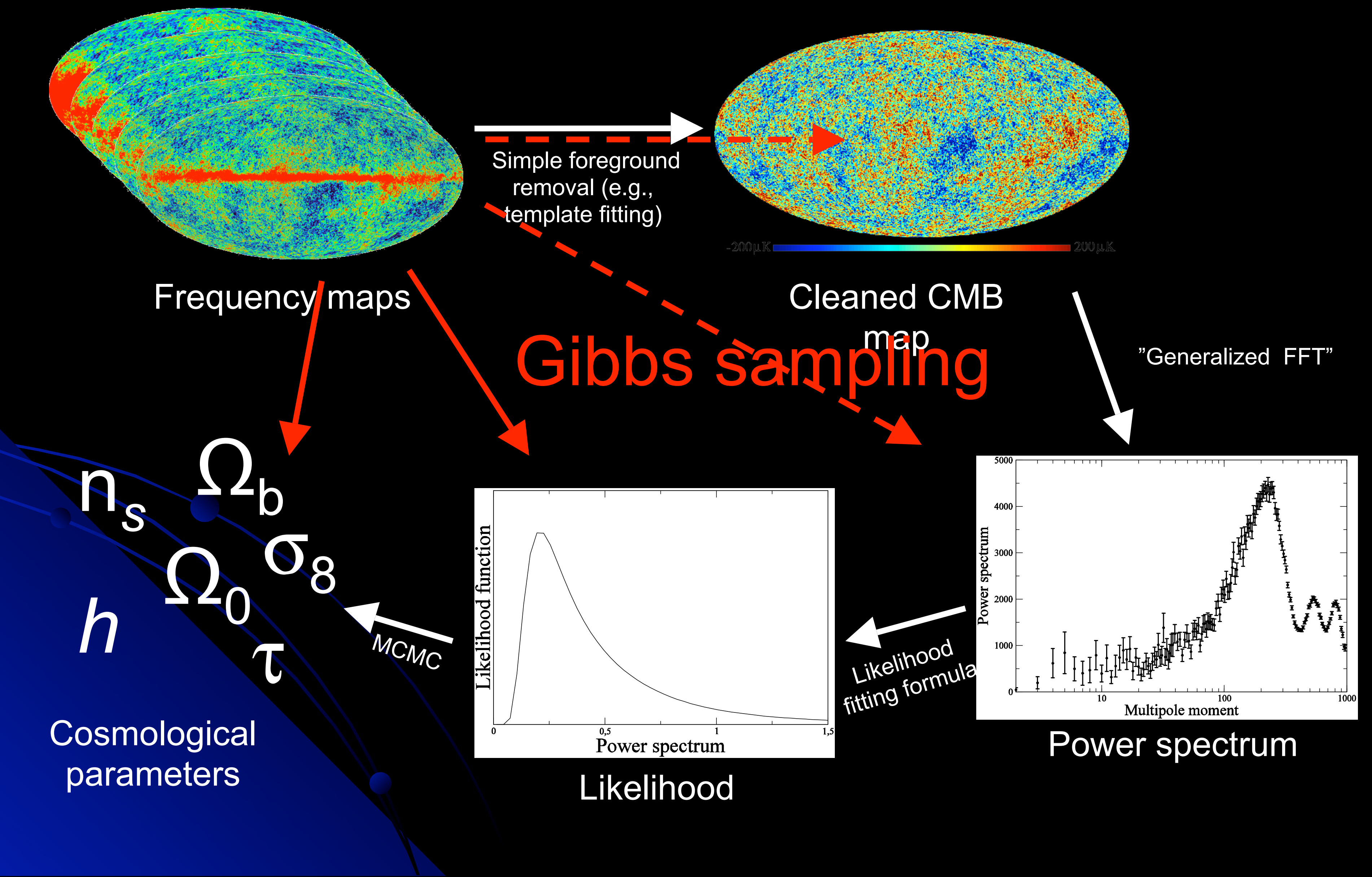
CMB analysis by Gibbs sampling



CMB analysis by Gibbs sampling



CMB analysis by Gibbs sampling



MCMC and the Metropolis-Hastings rule

MCMC and the Metropolis-Hastings rule

- In 1953, Metropolis et al. presented a very general algorithm for sampling from arbitrary probability distributions

MCMC and the Metropolis-Hastings rule

- In 1953, Metropolis et al. presented a very general algorithm for sampling from arbitrary probability distributions
  - Generalized by W. K. Hastings in 1970 to allow for even more ”cases”

MCMC and the Metropolis-Hastings rule

- In 1953, Metropolis et al. presented a very general algorithm for sampling from arbitrary probability distributions
  - Generalized by W. K. Hastings in 1970 to allow for even more ”cases”

MCMC and the Metropolis-Hastings rule

- In 1953, Metropolis et al. presented a very general algorithm for sampling from arbitrary probability distributions
  - Generalized by W. K. Hastings in 1970 to allow for even more ”cases”
- Most easily explained by the actual algorithm. Suppose you want to sample from a distribution P(x):

MCMC and the Metropolis-Hastings rule

- In 1953, Metropolis et al. presented a very general algorithm for sampling from arbitrary probability distributions
  - Generalized by W. K. Hastings in 1970 to allow for even more ”cases”

- Most easily explained by the actual algorithm. Suppose you want to sample from a distribution P(x):

1. Choose an arbitrary starting point, x_0

MCMC and the Metropolis-Hastings rule

- In 1953, Metropolis et al. presented a very general algorithm for sampling from arbitrary probability distributions
  - Generalized by W. K. Hastings in 1970 to allow for even more "cases"

- Most easily explained by the actual algorithm. Suppose you want to sample from a distribution P(x):

1. Choose an arbitrary starting point, x_0
2. Define a stochastic transition rule that proposes a new state given the current state, $T(x_i \rightarrow x_p)$, and propose a new point x_p

$$p = \frac{P(x_p)}{P(x_i)} \frac{T(x_p \rightarrow x_i)}{T(x_i \rightarrow x_p)}$$

MCMC and the Metropolis-Hastings rule

- In 1953, Metropolis et al. presented a very general algorithm for sampling from arbitrary probability distributions
  - Generalized by W. K. Hastings in 1970 to allow for even more "cases"

- Most easily explained by the actual algorithm. Suppose you want to sample from a distribution P(x):
  1. Choose an arbitrary starting point, x_0
  2. Define a stochastic transition rule that proposes a new state given the current state, $T(x_i \to x_p)$, and propose a new point x_p
  3. Compute the (Metropolis-Hastings) acceptance probability

$$p = \frac{P(x_p)}{P(x_i)} \frac{T(x_p \to x_i)}{T(x_i \to x_p)}$$

MCMC and the Metropolis-Hastings rule

- In 1953, Metropolis et al. presented a very general algorithm for sampling from arbitrary probability distributions
  - Generalized by W. K. Hastings in 1970 to allow for even more "cases"

- Most easily explained by the actual algorithm. Suppose you want to sample from a distribution P(x):
  1. Choose an arbitrary starting point, x_0
  2. Define a stochastic transition rule that proposes a new state given the current state, $T(x_i \rightarrow x_p)$, and propose a new point x_p
  3. Compute the (Metropolis-Hastings) acceptance probability

$$p = \frac{P(x_p)}{P(x_i)} \frac{T(x_p \rightarrow x_i)}{T(x_i \rightarrow x_p)}$$

  4. Accept proposed point with probability min(1, p)

MCMC and the Metropolis-Hastings rule

- In 1953, Metropolis et al. presented a very general algorithm for sampling from arbitrary probability distributions
  - Generalized by W. K. Hastings in 1970 to allow for even more "cases"

- Most easily explained by the actual algorithm. Suppose you want to sample from a distribution P(x):
  1. Choose an arbitrary starting point, x_0
  2. Define a stochastic transition rule that proposes a new state given the current state, $T(x_i \rightarrow x_p)$, and propose a new point x_p
  3. Compute the (Metropolis-Hastings) acceptance probability

$$p = \frac{P(x_p)}{P(x_i)} \frac{T(x_p \rightarrow x_i)}{T(x_i \rightarrow x_p)}$$

  4. Accept proposed point with probability min(1, p)
  5. Repeat steps 2-4

MCMC and the Metropolis-Hastings rule

- In 1953, Metropolis et al. presented a very general algorithm for sampling from arbitrary probability distributions
  - Generalized by W. K. Hastings in 1970 to allow for even more "cases"

- Most easily explained by the actual algorithm. Suppose you want to sample from a distribution P(x):
  1. Choose an arbitrary starting point, x_0
  2. Define a stochastic transition rule that proposes a new state given the current state, $T(x_i \rightarrow x_p)$, and propose a new point x_p
  3. Compute the (Metropolis-Hastings) acceptance probability

$$p = \frac{P(x_p)}{P(x_i)} \frac{T(x_p \rightarrow x_i)}{T(x_i \rightarrow x_p)}$$

  4. Accept proposed point with probability min(1, p)
  5. Repeat steps 2-4

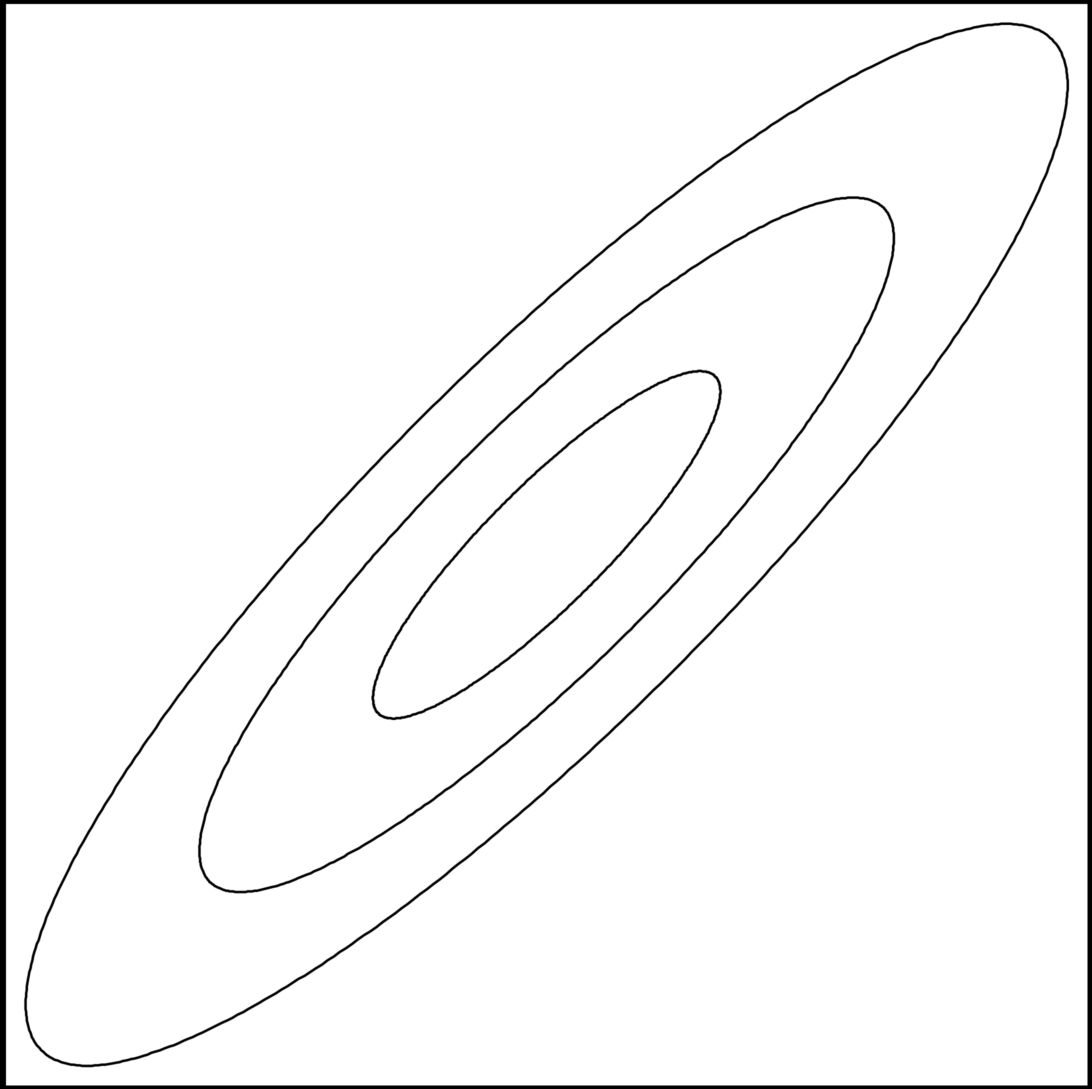
MCMC and the Metropolis-Hastings rule

- In 1953, Metropolis et al. presented a very general algorithm for sampling from arbitrary probability distributions
  - Generalized by W. K. Hastings in 1970 to allow for even more "cases"

- Most easily explained by the actual algorithm. Suppose you want to sample from a distribution P(x):
  1. Choose an arbitrary starting point, x_0
  2. Define a stochastic transition rule that proposes a new state given the current state, $T(x_i \rightarrow x_p)$, and propose a new point x_p
  3. Compute the (Metropolis-Hastings) acceptance probability
  $$p = \frac{P(x_p)}{P(x_i)} \frac{T(x_p \rightarrow x_i)}{T(x_i \rightarrow x_p)}$$
  4. Accept proposed point with probability min(1, p)
  5. Repeat steps 2-4

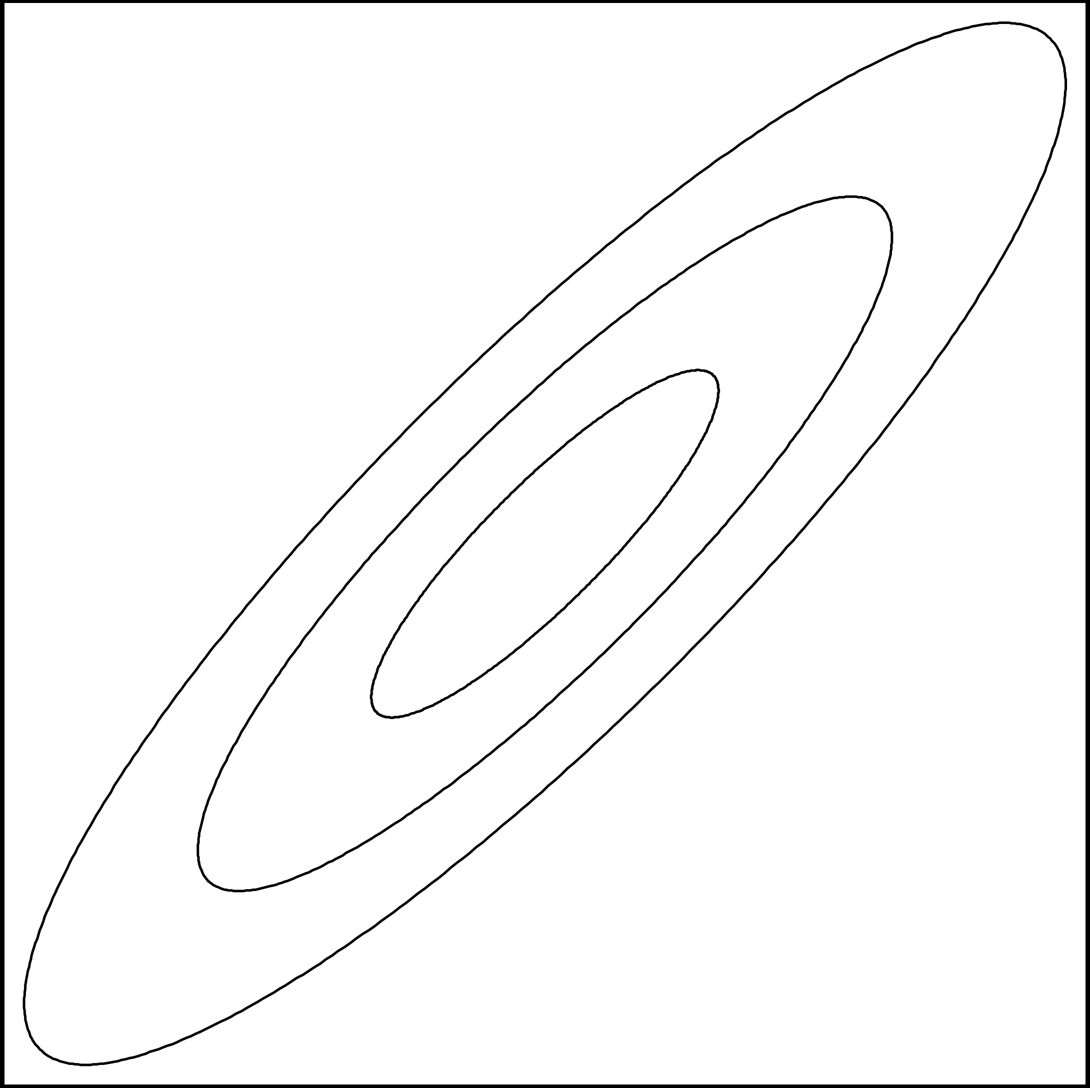
- After some burn-in period, x_i will be drawn from the correct distribution P(x)

Schematic view of MH MCMC



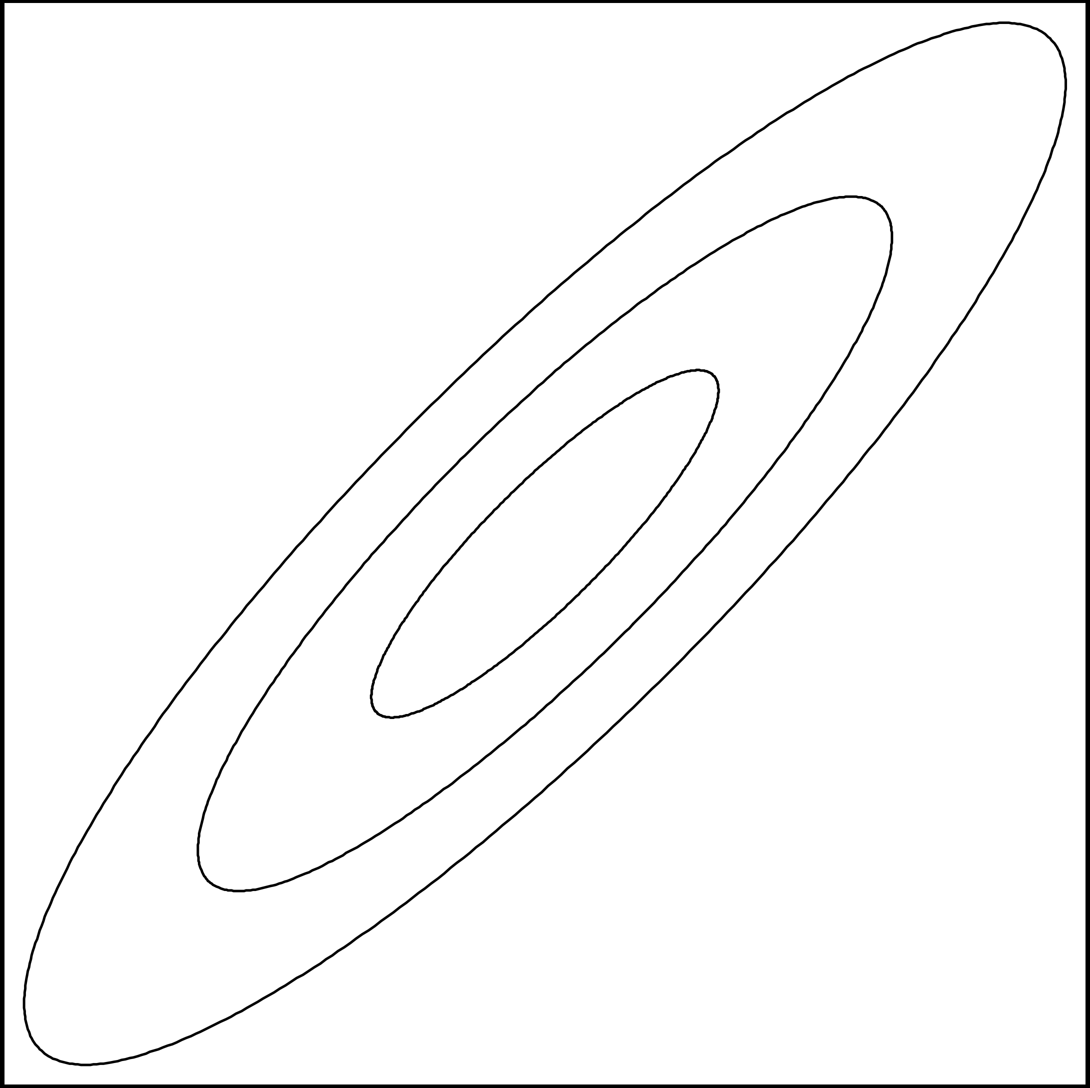
Schematic view of MH MCMC

- Consider a two-dimensional example



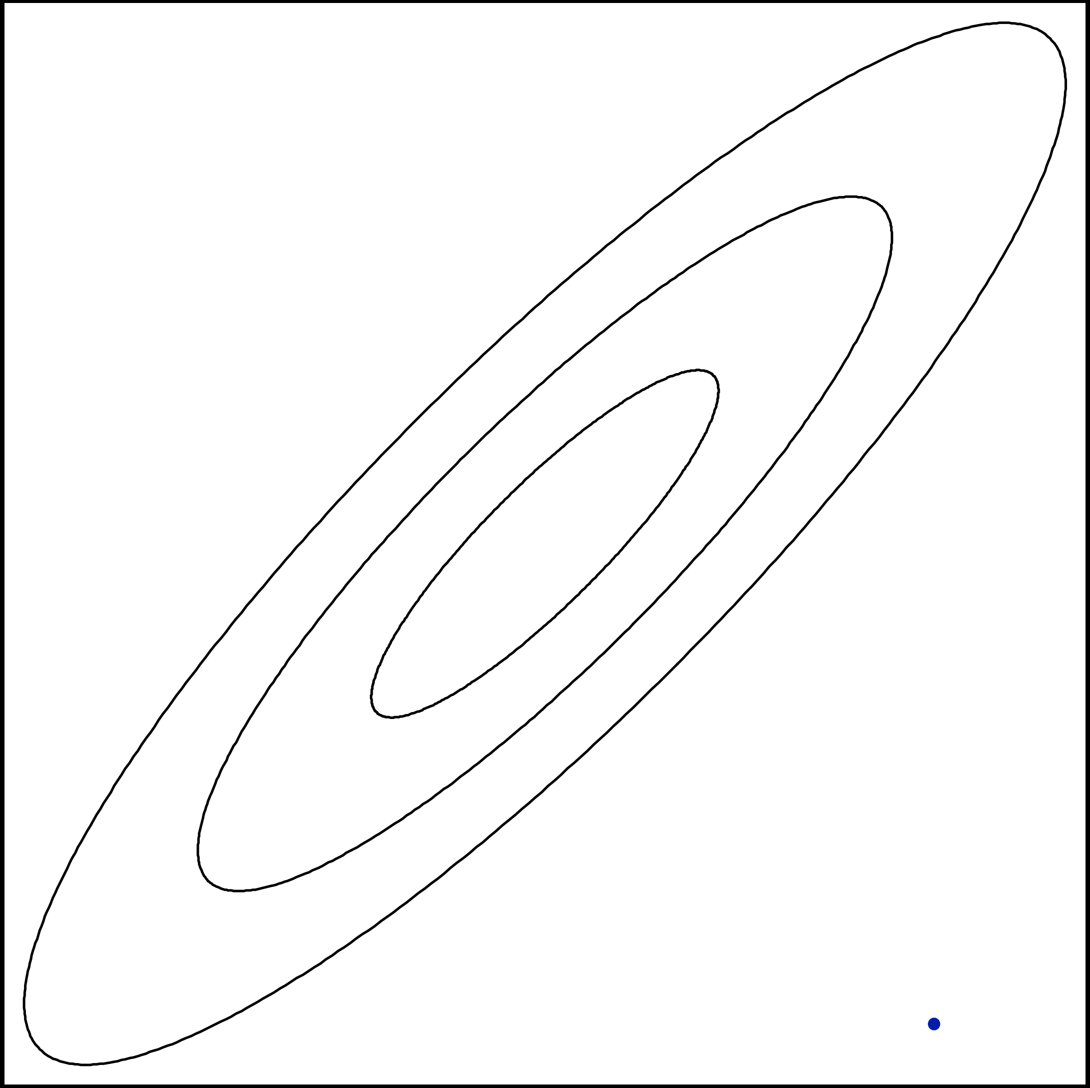
Schematic view of MH MCMC

- Consider a two-dimensional example
- Metropolis algorithm:



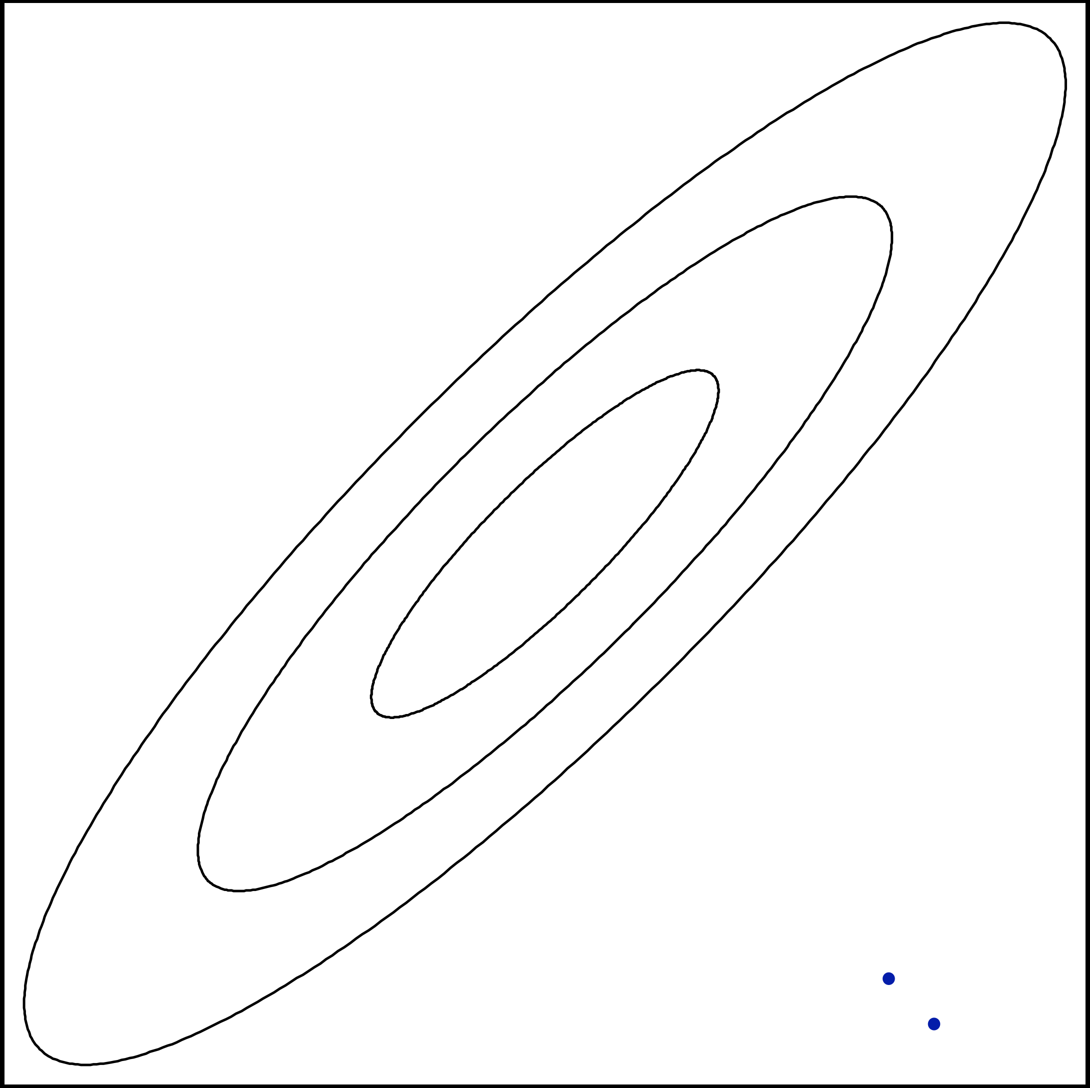
Schematic view of MH MCMC

- Consider a two-dimensional example
- Metropolis algorithm:
  - Choose a symmetric proposal density, such that $T_1 = T_0$



Schematic view of MH MCMC

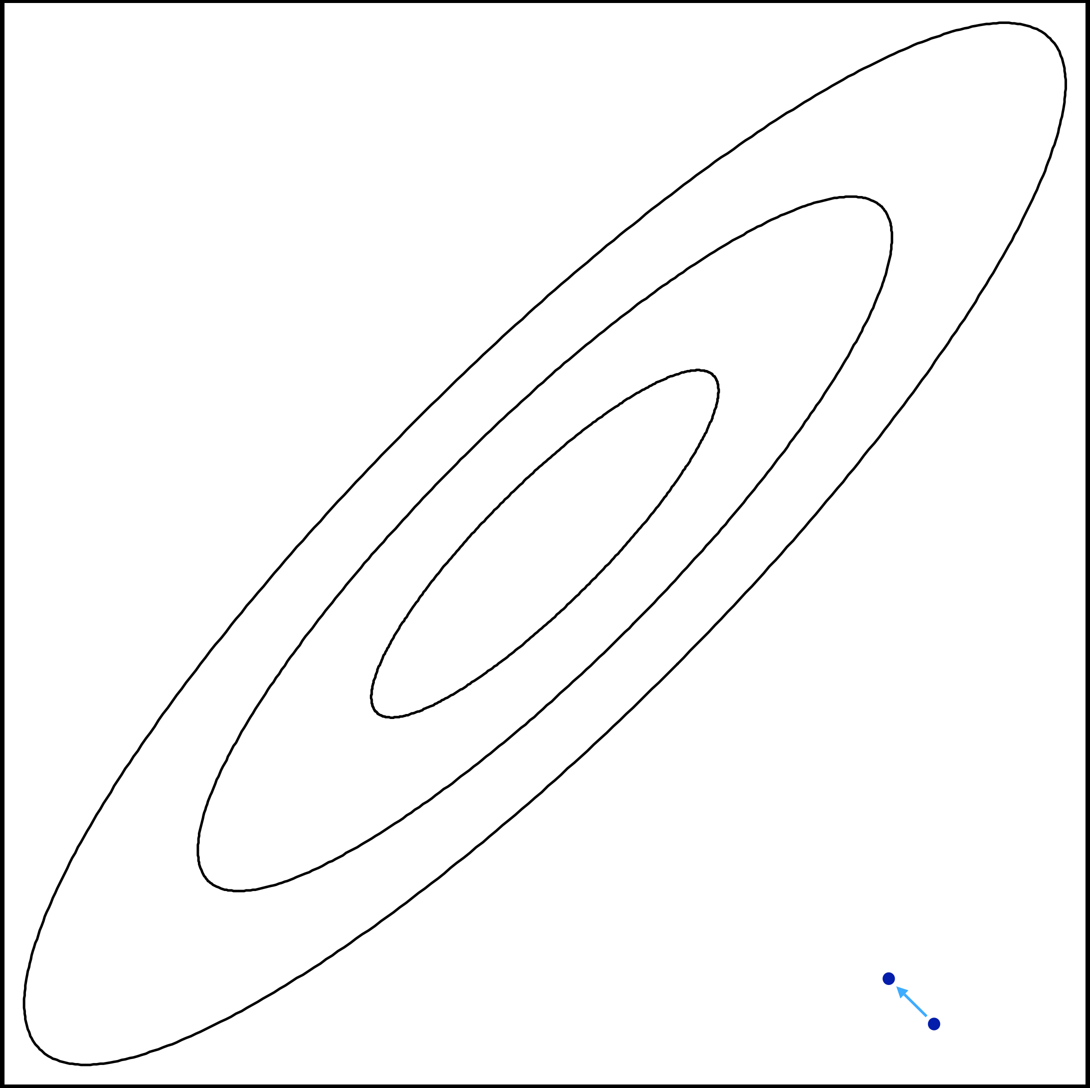
- Consider a two-dimensional example
- Metropolis algorithm:
  - Choose a symmetric proposal density, such that $T_1 = T_0$
  - Choose any initial point



Schematic view of MH MCMC

- Consider a two-dimensional example
- Metropolis algorithm:
  - Choose a symmetric proposal density, such that $T_1 = T_0$
  - Choose any initial point
  - Propose new point according to rule

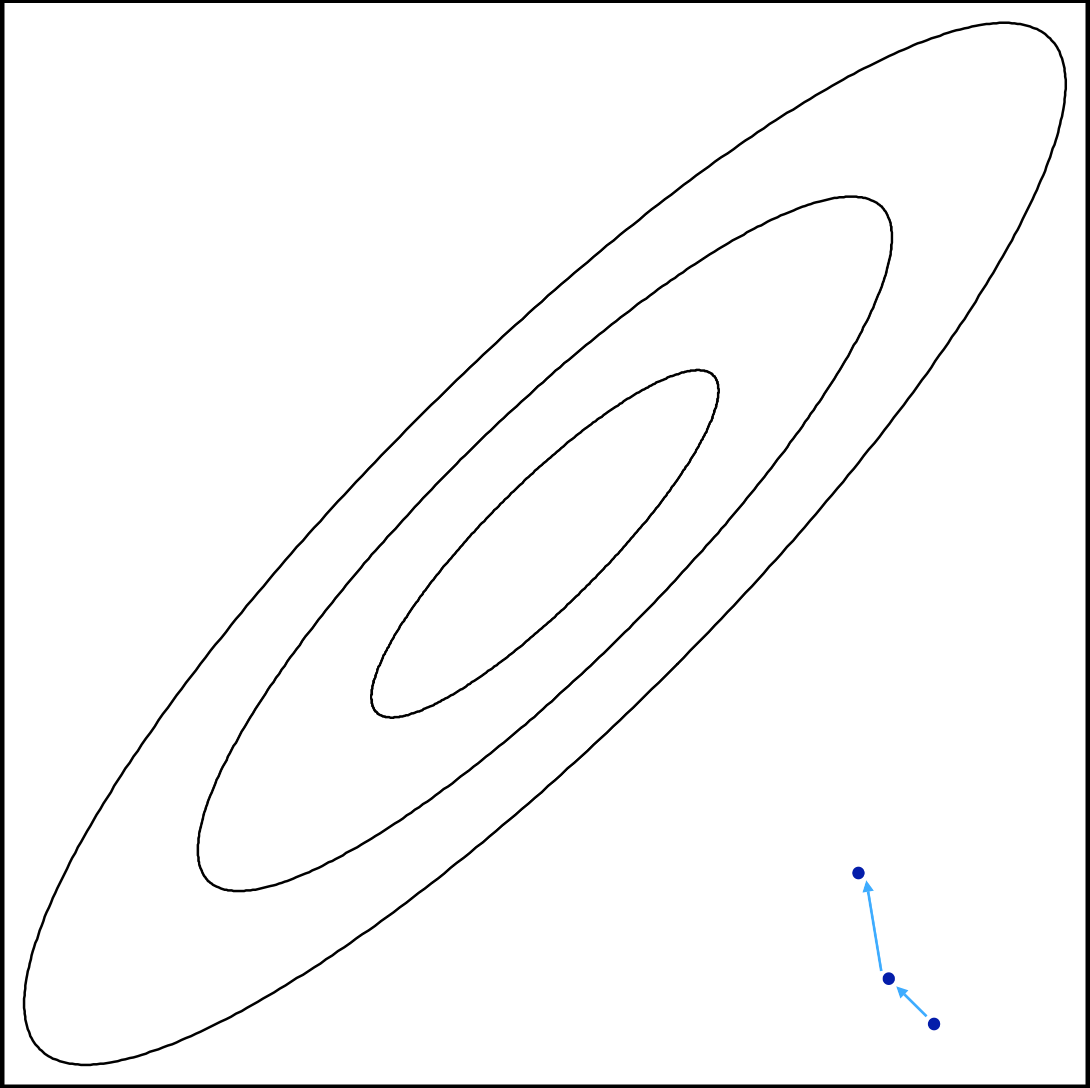
$$P = \min\left(1, \frac{\mathcal{L}(\text{new})}{\mathcal{L}(\text{old})}\right)$$



Schematic view of MH MCMC

- Consider a two-dimensional example
- Metropolis algorithm:
  - Choose a symmetric proposal density, such that $T_1 = T_0$
  - Choose any initial point
  - Propose new point according to rule
  - Move to proposed point with probability

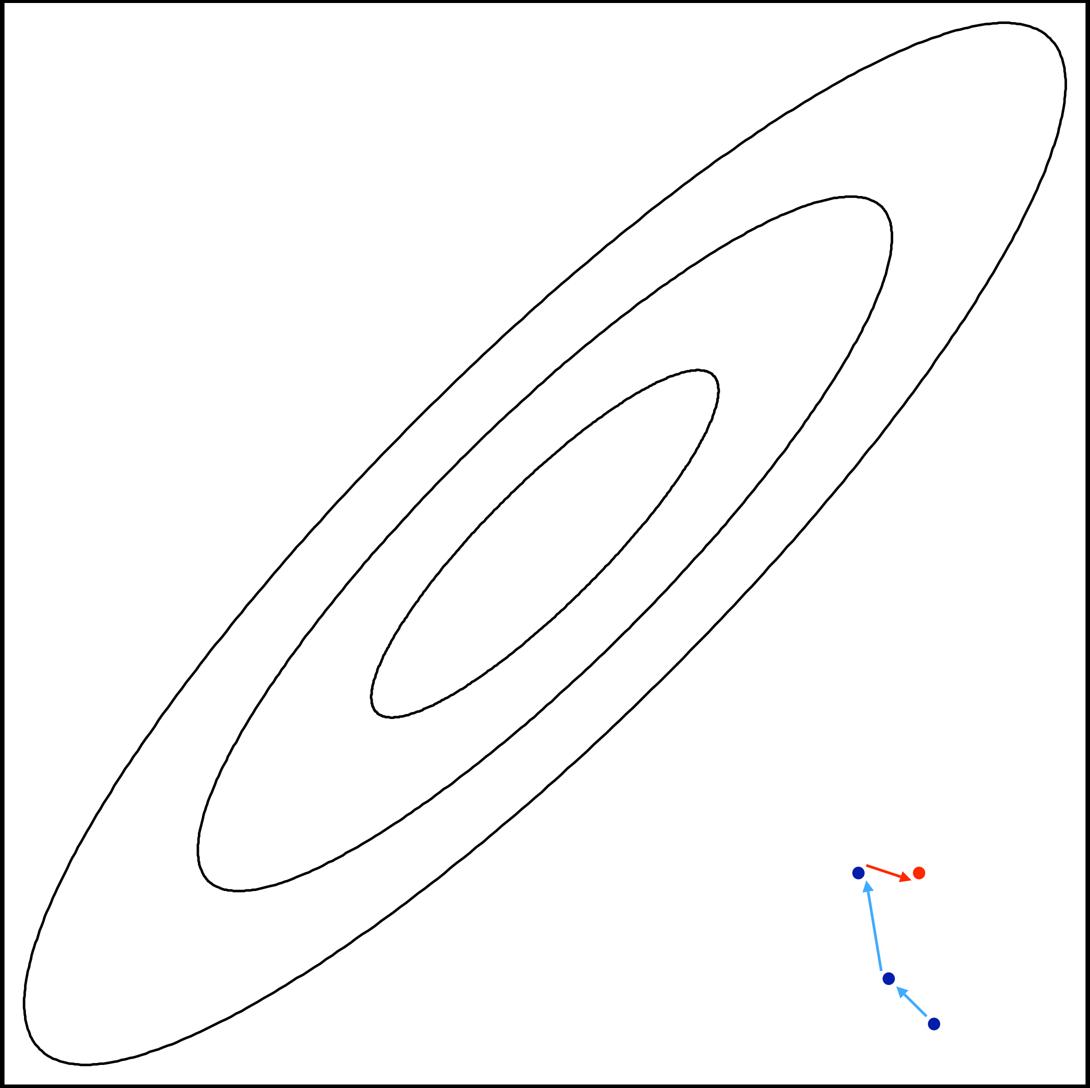
$$P = \min\left(1, \frac{\mathcal{L}(\text{new})}{\mathcal{L}(\text{old})}\right)$$



Schematic view of MH MCMC

- Consider a two-dimensional example
- Metropolis algorithm:
  - Choose a symmetric proposal density, such that $T_1 = T_0$
  - Choose any initial point
  - Propose new point according to rule
  - Move to proposed point with probability

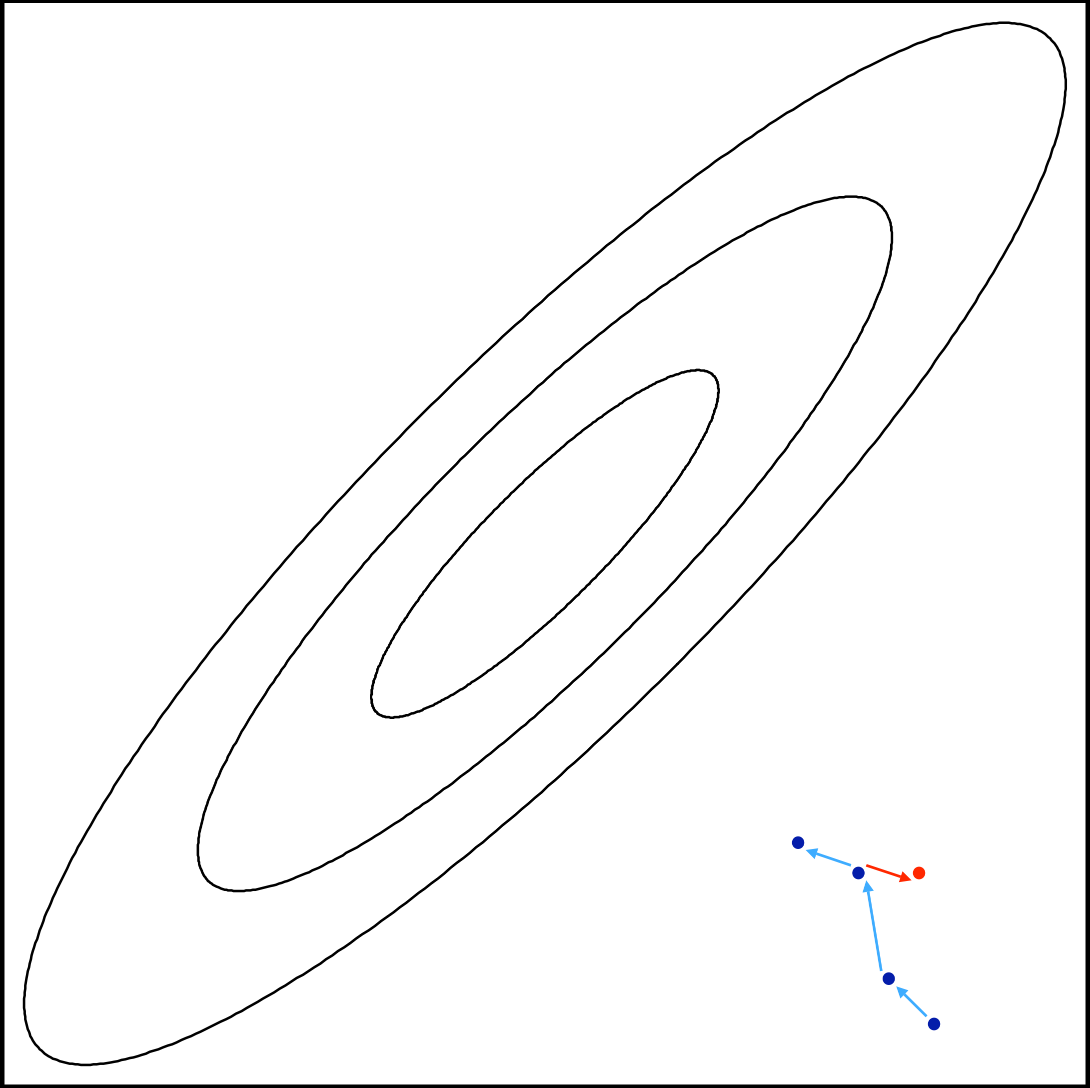
$$P = \min\left(1, \frac{\mathcal{L}(\text{new})}{\mathcal{L}(\text{old})}\right)$$



Schematic view of MH MCMC

- Consider a two-dimensional example
- Metropolis algorithm:
  - Choose a symmetric proposal density, such that $T_1 = T_0$
  - Choose any initial point
  - Propose new point according to rule
  - Move to proposed point with probability

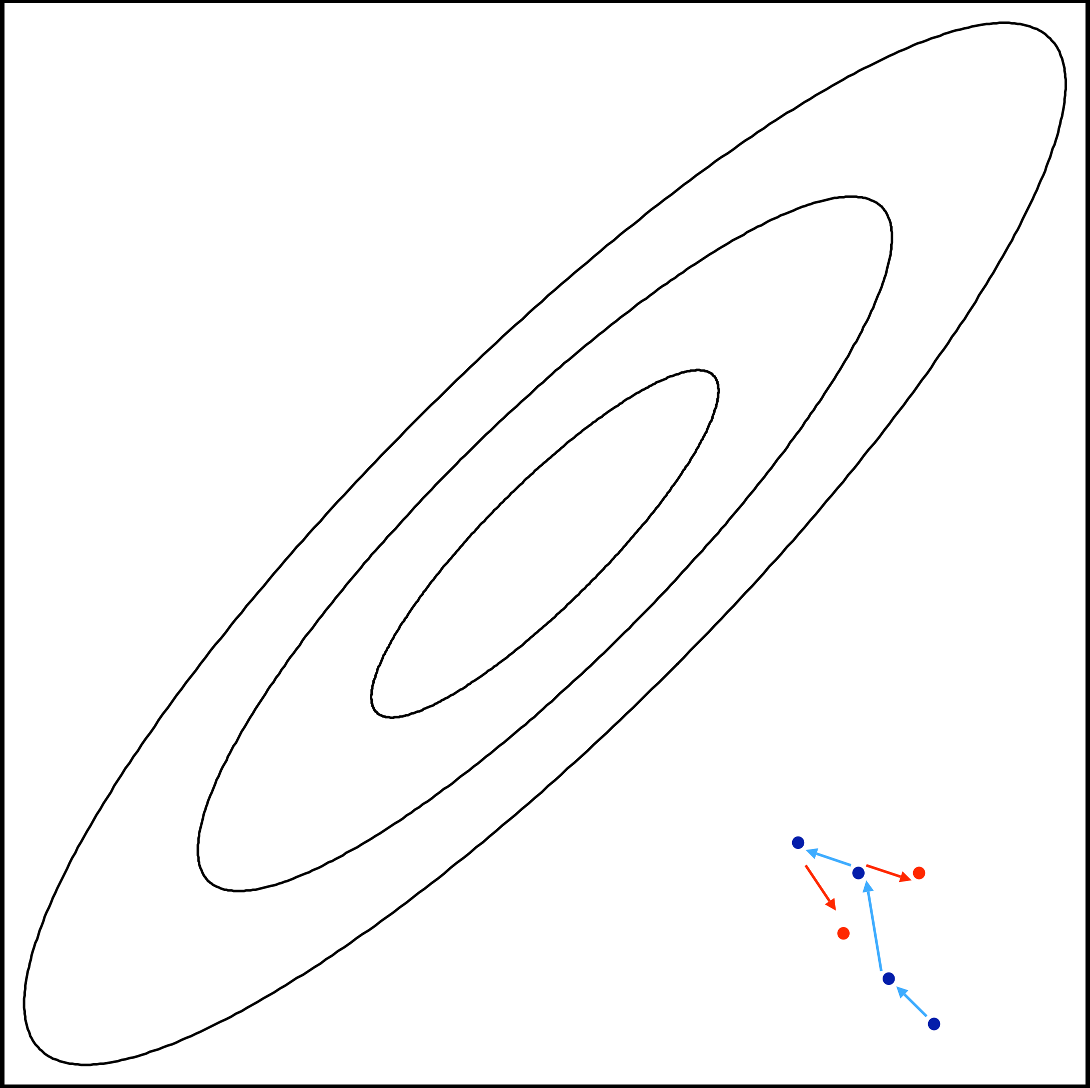
$$P = \min\left(1, \frac{\mathcal{L}(\text{new})}{\mathcal{L}(\text{old})}\right)$$



Schematic view of MH MCMC

- Consider a two-dimensional example
- Metropolis algorithm:
  - Choose a symmetric proposal density, such that $T_1 = T_0$
  - Choose any initial point
  - Propose new point according to rule
  - Move to proposed point with probability

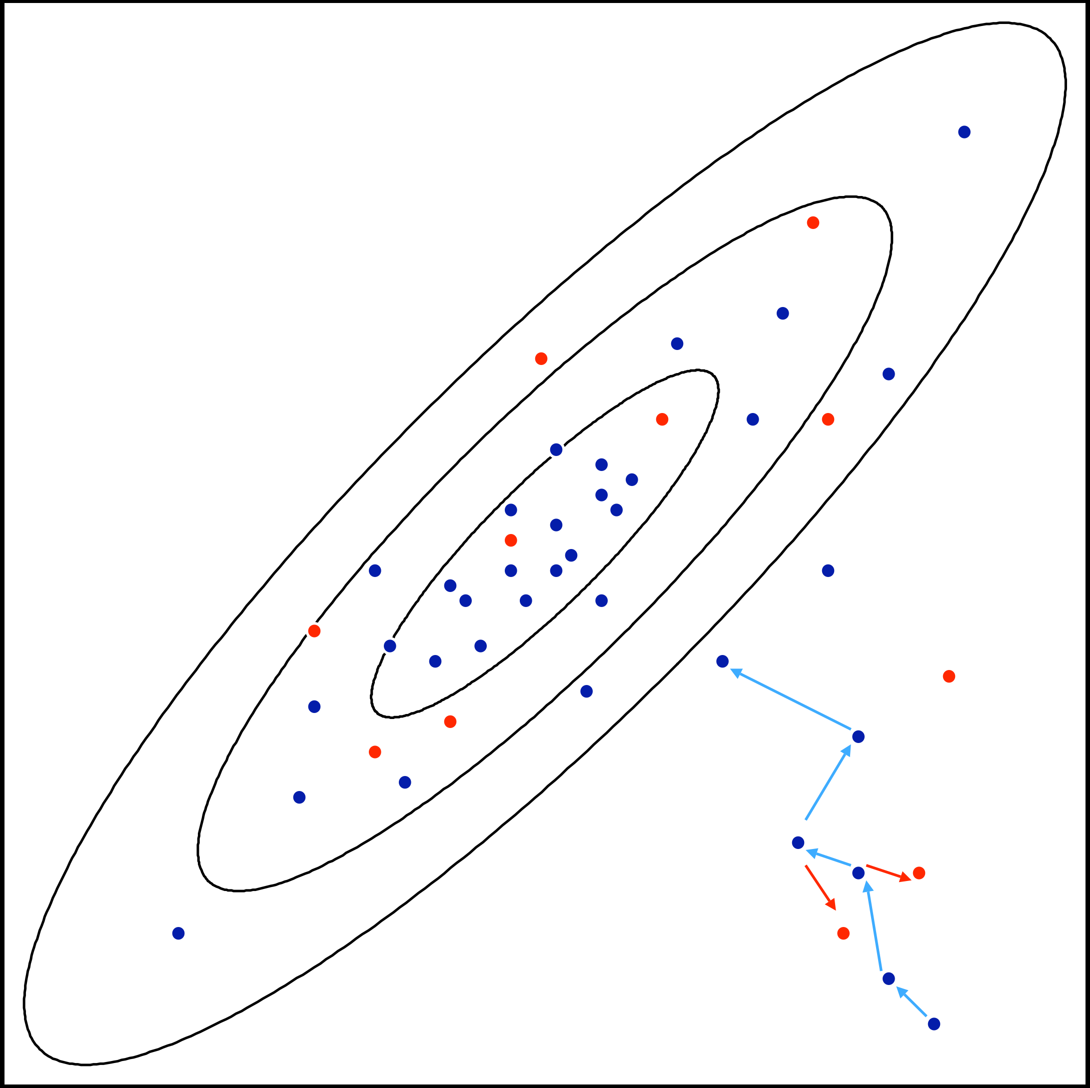
$$P = \min\left(1, \frac{\mathcal{L}(\text{new})}{\mathcal{L}(\text{old})}\right)$$



Schematic view of MH MCMC

- Consider a two-dimensional example
- Metropolis algorithm:
  - Choose a symmetric proposal density, such that $T_1 = T_0$
  - Choose any initial point
  - Propose new point according to rule
  - Move to proposed point with probability

$$P = \min\left(1, \frac{\mathcal{L}(\text{new})}{\mathcal{L}(\text{old})}\right)$$

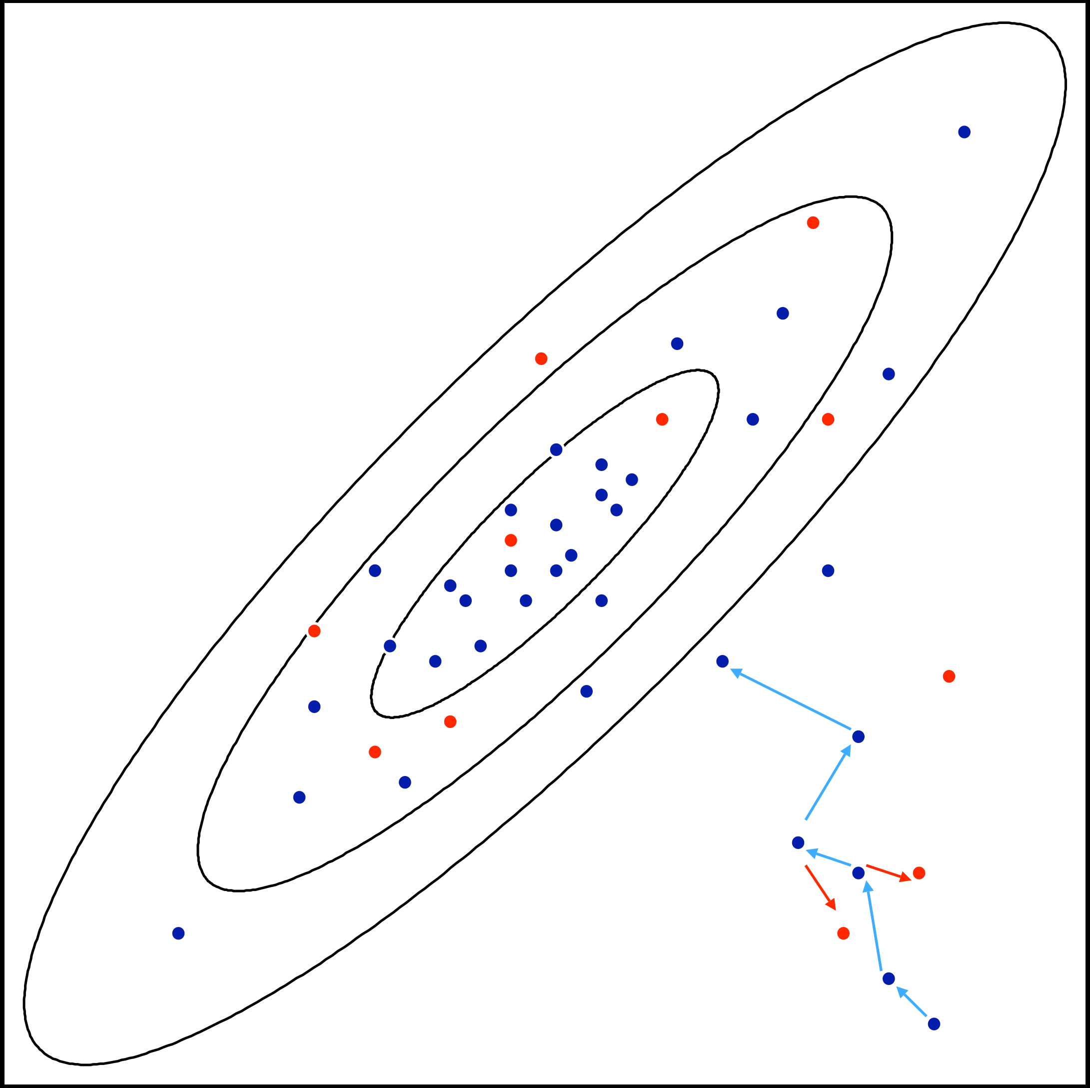


Schematic view of MH MCMC

- Consider a two-dimensional example
- Metropolis algorithm:
  - Choose a symmetric proposal density, such that $T_1 = T_0$
  - Choose any initial point
  - Propose new point according to rule
  - Move to proposed point with probability

$$P = \min\left(1, \frac{\mathcal{L}(\text{new})}{\mathcal{L}(\text{old})}\right)$$

  - Iterate 3 and 4

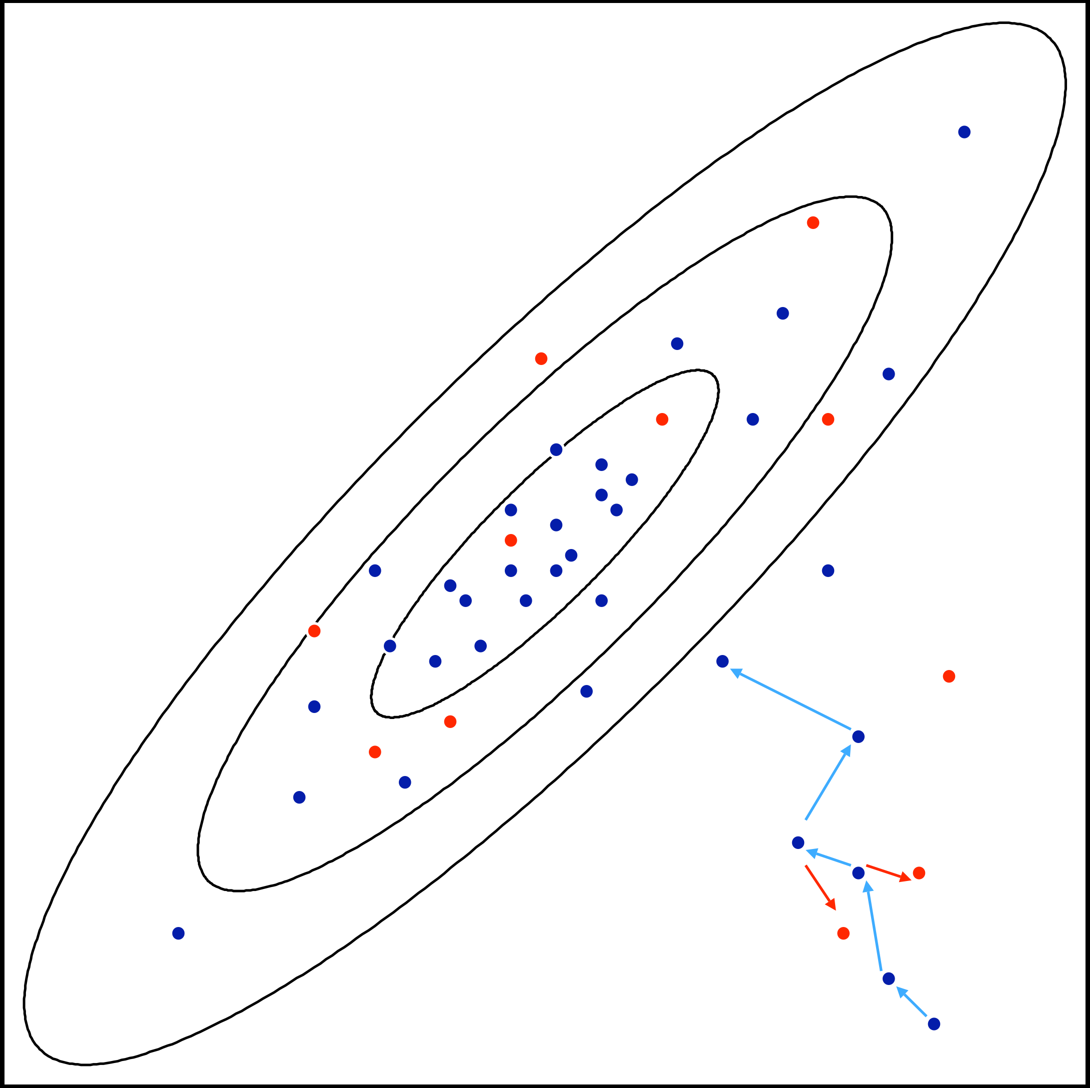


Schematic view of MH MCMC

- Consider a two-dimensional example
- Metropolis algorithm:
  - Choose a symmetric proposal density, such that $T_1 = T_0$
  - Choose any initial point
  - Propose new point according to rule
  - Move to proposed point with probability

$$P = \min\left(1, \frac{\mathcal{L}(\text{new})}{\mathcal{L}(\text{old})}\right)$$

  - Iterate 3 and 4
- Works beautifully for many complex problems, but can be inefficient for many parameters



The Gibbs sampler

The Gibbs sampler

- Gibbs sampling is a special case of the Metropolis-Hastings algorithm

The Gibbs sampler

- Gibbs sampling is a special case of the Metropolis-Hastings algorithm
  - Uses knowledge about the target distribution to propose samples

The Gibbs sampler

- Gibbs sampling is a special case of the Metropolis-Hastings algorithm
  - Uses knowledge about the target distribution to propose samples
  - Applicable to problems with more than one parameter (eg., P(A, B))

The Gibbs sampler

- Gibbs sampling is a special case of the Metropolis-Hastings algorithm
  - Uses knowledge about the target distribution to propose samples
  - Applicable to problems with more than one parameter (eg., P(A, B))

$$T((A,B)^i \rightarrow (A,B)^p) = P(A|B^i)\,\delta(B^p - B^i)$$

The Gibbs sampler

- Gibbs sampling is a special case of the Metropolis-Hastings algorithm
  - Uses knowledge about the target distribution to propose samples
  - Applicable to problems with more than one parameter (eg., P(A, B))
- Defined by a very special transition rule, namely

$$T((A,B)^i \to (A,B)^p) = P(A|B^i)\,\delta(B^p - B^i)$$

The Gibbs sampler

- Gibbs sampling is a special case of the Metropolis-Hastings algorithm
  - Uses knowledge about the target distribution to propose samples
  - Applicable to problems with more than one parameter (eg., P(A, B))
- Defined by a very special transition rule, namely

$$T((A,B)^i \to (A,B)^p) = P(A|B^i)\,\delta(B^p - B^i)$$

  - That is, fix one parameter at the previous state, and sample the other from the corresponding *conditional* distribution

The Gibbs sampler

- Gibbs sampling is a special case of the Metropolis-Hastings algorithm
  - Uses knowledge about the target distribution to propose samples
  - Applicable to problems with more than one parameter (eg., P(A, B))
- Defined by a very special transition rule, namely

$$T((A,B)^i \to (A,B)^p) = P(A|B^i)\,\delta(B^p - B^i)$$

  - That is, fix one parameter at the previous state, and sample the other from the corresponding *conditional* distribution

$$p = \frac{P(A^p, B^p)}{P(A^i, B^i)} \frac{T((A,B)^p \to (A,B)^i)}{T((A,B)^i \to (A,B)^p)}$$

The Gibbs sampler

- Gibbs sampling is a special case of the Metropolis-Hastings algorithm
  - Uses knowledge about the target distribution to propose samples
  - Applicable to problems with more than one parameter (eg., P(A, B))
- Defined by a very special transition rule, namely

$$T((A,B)^i \to (A,B)^p) = P(A|B^i)\,\delta(B^p - B^i)$$

  - That is, fix one parameter at the previous state, and sample the other from the corresponding *conditional* distribution

$$p = \frac{P(A^p, B^p)}{P(A^i, B^i)} \frac{T((A,B)^p \to (A,B)^i)}{T((A,B)^i \to (A,B)^p)} = \frac{P(A^p|B^p)P(B^p)}{P(A^i|B^i)P(B^i)} \frac{P(A^i|B^i)\delta(B^i - B^p)}{P(A^p|B^p)\delta(B^p - B^i)}$$

The Gibbs sampler

- Gibbs sampling is a special case of the Metropolis-Hastings algorithm
  - Uses knowledge about the target distribution to propose samples
  - Applicable to problems with more than one parameter (eg., P(A, B))
- Defined by a very special transition rule, namely

$$T((A,B)^i \to (A,B)^p) = P(A|B^i)\,\delta(B^p - B^i)$$

  - That is, fix one parameter at the previous state, and sample the other from the corresponding *conditional* distribution

Conditional distrib.

$$p = \frac{P(A^p,B^p)}{P(A^i,B^i)}\frac{T((A,B)^p \to (A,B)^i)}{T((A,B)^i \to (A,B)^p)} = \frac{P(A^p|B^p)P(B^p)}{P(A^i|B^i)P(B^i)}\,\frac{P(A^i|B^i)\delta(B^i-B^p)}{P(A^p|B^p)\delta(B^p-B^i)}$$

The Gibbs sampler

- Gibbs sampling is a special case of the Metropolis-Hastings algorithm
  - Uses knowledge about the target distribution to propose samples
  - Applicable to problems with more than one parameter (eg., P(A, B))

- Defined by a very special transition rule, namely

$$T((A,B)^i \to (A,B)^p) = P(A|B^i)\,\delta(B^p - B^i)$$

  - That is, fix one parameter at the previous state, and sample the other from the corresponding *conditional* distribution

Conditional distrib. Proposal density

$$p = \frac{P(A^p,B^p)}{P(A^i,B^i)}\frac{T((A,B)^p \to (A,B)^i)}{T((A,B)^i \to (A,B)^p)} = \frac{P(A^p|B^p)P(B^p)}{P(A^i|B^i)P(B^i)}\frac{P(A^i|B^i)\delta(B^i - B^p)}{P(A^p|B^p)\delta(B^p - B^i)}$$

The Gibbs sampler

- Gibbs sampling is a special case of the Metropolis-Hastings algorithm
  - Uses knowledge about the target distribution to propose samples
  - Applicable to problems with more than one parameter (eg., P(A, B))
- Defined by a very special transition rule, namely

$$T((A,B)^i \to (A,B)^p) = P(A|B^i)\,\delta(B^p - B^i)$$

  - That is, fix one parameter at the previous state, and sample the other from the corresponding *conditional* distribution

Conditional distrib. Proposal density

$$p = \frac{P(A^p, B^p)}{P(A^i, B^i)} \frac{T((A,B)^p \to (A,B)^i)}{T((A,B)^i \to (A,B)^p)} = \frac{P(A^p|B^p)P(B^p)}{P(A^i|B^i)P(B^i)} \frac{P(A^i|B^i)\delta(B^i - B^p)}{P(A^p|B^p)\delta(B^p - B^i)}$$

$$\overset{B^p = B^i}{=} \frac{P(A^p|B^i)P(B^i)}{P(A^i|B^i)P(B^i)} \frac{P(A^i|B^i)}{P(A^p|B^i)}$$

The Gibbs sampler

- Gibbs sampling is a special case of the Metropolis-Hastings algorithm
  - Uses knowledge about the target distribution to propose samples
  - Applicable to problems with more than one parameter (eg., P(A, B))
- Defined by a very special transition rule, namely

$$T((A,B)^i \to (A,B)^p) = P(A|B^i)\,\delta(B^p - B^i)$$

  - That is, fix one parameter at the previous state, and sample the other from the corresponding *conditional* distribution

$$p = \frac{P(A^p,B^p)}{P(A^i,B^i)}\,\frac{T((A,B)^p \to (A,B)^i)}{T((A,B)^i \to (A,B)^p)} = \underbrace{\frac{P(A^p|B^p)P(B^p)}{P(A^i|B^i)P(B^i)}}_{\text{Conditional distrib.}}\,\underbrace{\frac{P(A^i|B^i)\delta(B^i-B^p)}{P(A^p|B^p)\delta(B^p-B^i)}}_{\text{Proposal density}}$$

$$\stackrel{B^p=B^i}{=} \frac{P(A^p|B^i)\cancel{P(B^i)}}{P(A^i|B^i)\cancel{P(B^i)}}\,\frac{P(A^i|B^i)}{P(A^p|B^i)}$$

The Gibbs sampler

- Gibbs sampling is a special case of the Metropolis-Hastings algorithm
  - Uses knowledge about the target distribution to propose samples
  - Applicable to problems with more than one parameter (eg., P(A, B))

- Defined by a very special transition rule, namely

$$T((A,B)^i \to (A,B)^p) = P(A|B^i)\,\delta(B^p - B^i)$$

  - That is, fix one parameter at the previous state, and sample the other from the corresponding *conditional* distribution

Conditional distrib. Proposal density

$$p = \frac{P(A^p, B^p)}{P(A^i, B^i)}\,\frac{T((A,B)^p \to (A,B)^i)}{T((A,B)^i \to (A,B)^p)} = \frac{P(A^p|B^p)P(B^p)}{P(A^i|B^i)P(B^i)}\,\frac{P(A^i|B^i)\delta(B^i - B^p)}{P(A^p|B^p)\delta(B^p - B^i)}$$

$$\overset{B^p = B^i}{=} \frac{\cancel{P(A^p|B^i)}\cancel{P(B^i)}}{P(A^i|B^i)\cancel{P(B^i)}}\,\frac{P(A^i|B^i)}{\cancel{P(A^p|B^i)}}$$

The Gibbs sampler

- Gibbs sampling is a special case of the Metropolis-Hastings algorithm
  - Uses knowledge about the target distribution to propose samples
  - Applicable to problems with more than one parameter (eg., P(A, B))
- Defined by a very special transition rule, namely

$$T((A,B)^i \to (A,B)^p) = P(A|B^i)\,\delta(B^p - B^i)$$

  - That is, fix one parameter at the previous state, and sample the other from the corresponding *conditional* distribution

$$p = \frac{P(A^p,B^p)}{P(A^i,B^i)}\,\frac{T((A,B)^p \to (A,B)^i)}{T((A,B)^i \to (A,B)^p)} = \overbrace{\frac{P(A^p|B^p)P(B^p)}{P(A^i|B^i)P(B^i)}}^{\text{Conditional distrib.}}\;\overbrace{\frac{P(A^i|B^i)\delta(B^i - B^p)}{P(A^p|B^p)\delta(B^p - B^i)}}^{\text{Proposal density}}$$

$$\overset{B^p = B^i}{=} \frac{\cancel{P(A^p|B^i)}\cancel{P(B^i)}}{\cancel{P(A^i|B^i)}\cancel{P(B^i)}}\,\frac{\cancel{P(A^i|B^i)}}{\cancel{P(A^p|B^i)}}$$

The Gibbs sampler

- Gibbs sampling is a special case of the Metropolis-Hastings algorithm
  - Uses knowledge about the target distribution to propose samples
  - Applicable to problems with more than one parameter (eg., P(A, B))
- Defined by a very special transition rule, namely

$$T((A,B)^i \rightarrow (A,B)^p) = P(A|B^i)\,\delta(B^p - B^i)$$

  - That is, fix one parameter at the previous state, and sample the other from the corresponding *conditional* distribution

Conditional distrib. Proposal density

$$p = \frac{P(A^p,B^p)}{P(A^i,B^i)}\frac{T((A,B)^p \rightarrow (A,B)^i)}{T((A,B)^i \rightarrow (A,B)^p)} = \frac{P(A^p|B^p)P(B^p)}{P(A^i|B^i)P(B^i)}\,\frac{P(A^i|B^i)\delta(B^i - B^p)}{P(A^p|B^p)\delta(B^p - B^i)}$$

$$\overset{B^p = B^i}{=} \frac{\cancel{P(A^p|B^i)}\cancel{P(B^i)}}{\cancel{P(A^i|B^i)}\cancel{P(B^i)}}\,\frac{\cancel{P(A^i|B^i)}}{\cancel{P(A^p|B^i)}} = 1!$$

The Gibbs sampler

- Gibbs sampling is a special case of the Metropolis-Hastings algorithm
  - Uses knowledge about the target distribution to propose samples
  - Applicable to problems with more than one parameter (eg., P(A, B))
- Defined by a very special transition rule, namely

$$T((A,B)^i \to (A,B)^p) = P(A|B^i)\,\delta(B^p - B^i)$$

  - That is, fix one parameter at the previous state, and sample the other from the corresponding *conditional* distribution

Conditional distrib. Proposal density

$$p = \frac{P(A^p,B^p)}{P(A^i,B^i)}\frac{T((A,B)^p \to (A,B)^i)}{T((A,B)^i \to (A,B)^p)} = \frac{P(A^p|B^p)P(B^p)}{P(A^i|B^i)P(B^i)}\,\frac{P(A^i|B^i)\delta(B^i - B^p)}{P(A^p|B^p)\delta(B^p - B^i)}$$

$$\stackrel{B^p = B^i}{=} \frac{\cancel{P(A^p|B^i)}\cancel{P(B^i)}}{\cancel{P(A^i|B^i)}\cancel{P(B^i)}}\,\frac{\cancel{P(A^i|B^i)}}{\cancel{P(A^p|B^i)}} = 1!$$

The Gibbs sampler

- Gibbs sampling is a special case of the Metropolis-Hastings algorithm
  - Uses knowledge about the target distribution to propose samples
  - Applicable to problems with more than one parameter (eg., P(A, B))
- Defined by a very special transition rule, namely

$$T((A,B)^i \to (A,B)^p) = P(A|B^i)\,\delta(B^p - B^i)$$

  - That is, fix one parameter at the previous state, and sample the other from the corresponding *conditional* distribution
- The MH probability is then

Conditional distrib. Proposal density

$$p = \frac{P(A^p,B^p)}{P(A^i,B^i)}\frac{T((A,B)^p \to (A,B)^i)}{T((A,B)^i \to (A,B)^p)} = \frac{P(A^p|B^p)P(B^p)}{P(A^i|B^i)P(B^i)}\frac{P(A^i|B^i)\delta(B^i-B^p)}{P(A^p|B^p)\delta(B^p-B^i)}$$

$$\stackrel{B^p=B^i}{=} \frac{\cancel{P(A^p|B^i)}\cancel{P(B^i)}}{\cancel{P(A^i|B^i)}\cancel{P(B^i)}}\frac{\cancel{P(A^i|B^i)}}{\cancel{P(A^p|B^i)}} = 1!$$

The Gibbs sampler

- Gibbs sampling is a special case of the Metropolis-Hastings algorithm
  - Uses knowledge about the target distribution to propose samples
  - Applicable to problems with more than one parameter (eg., P(A, B))

- Defined by a very special transition rule, namely

$$T((A,B)^i \to (A,B)^p) = P(A|B^i)\,\delta(B^p - B^i)$$

  - That is, fix one parameter at the previous state, and sample the other from the corresponding *conditional* distribution

- The MH probability is then

Conditional distrib. Proposal density

$$p = \frac{P(A^p,B^p)}{P(A^i,B^i)}\,\frac{T((A,B)^p \to (A,B)^i)}{T((A,B)^i \to (A,B)^p)} = \frac{P(A^p|B^p)P(B^p)}{P(A^i|B^i)P(B^i)}\,\frac{P(A^i|B^i)\delta(B^i - B^p)}{P(A^p|B^p)\delta(B^p - B^i)}$$

$$\overset{B^p = B^i}{=} \frac{\cancel{P(A^p|B^i)}\cancel{P(B^i)}}{\cancel{P(A^i|B^i)}\cancel{P(B^i)}}\,\frac{\cancel{P(A^i|B^i)}}{\cancel{P(A^p|B^i)}} = 1!$$

The Gibbs sampler

- Gibbs sampling is a special case of the Metropolis-Hastings algorithm
  - Uses knowledge about the target distribution to propose samples
  - Applicable to problems with more than one parameter (eg., P(A, B))

- Defined by a very special transition rule, namely

$$T((A,B)^i \to (A,B)^p) = P(A|B^i)\,\delta(B^p - B^i)$$

  - That is, fix one parameter at the previous state, and sample the other from the corresponding *conditional* distribution

- The MH probability is then

Conditional distrib. Proposal density

$$p = \frac{P(A^p,B^p)}{P(A^i,B^i)}\,\frac{T((A,B)^p \to (A,B)^i)}{T((A,B)^i \to (A,B)^p)} = \frac{P(A^p|B^p)P(B^p)}{P(A^i|B^i)P(B^i)}\,\frac{P(A^i|B^i)\delta(B^i - B^p)}{P(A^p|B^p)\delta(B^p - B^i)}$$

$$\overset{B^p = B^i}{=} \frac{\cancel{P(A^p|B^i)}\cancel{P(B^i)}}{\cancel{P(A^i|B^i)}\cancel{P(B^i)}}\,\frac{\cancel{P(A^i|B^i)}}{\cancel{P(A^p|B^i)}} = 1!$$

- We *always* accept the proposed point!

The Gibbs sampler

- Gibbs sampling is a special case of the Metropolis-Hastings algorithm
  - Uses knowledge about the target distribution to propose samples
  - Applicable to problems with more than one parameter (eg., P(A, B))
- Defined by a very special transition rule, namely

$$T((A,B)^i \to (A,B)^p) = P(A|B^i)\,\delta(B^p - B^i)$$

  - That is, fix one parameter at the previous state, and sample the other from the corresponding *conditional* distribution
- The MH probability is then

$$p = \frac{P(A^p,B^p)}{P(A^i,B^i)}\frac{T((A,B)^p \to (A,B)^i)}{T((A,B)^i \to (A,B)^p)} = \overbrace{\frac{P(A^p|B^p)P(B^p)}{P(A^i|B^i)P(B^i)}}^{\text{Conditional distrib.}}\;\overbrace{\frac{P(A^i|B^i)\delta(B^i - B^p)}{P(A^p|B^p)\delta(B^p - B^i)}}^{\text{Proposal density}}$$

$$\overset{B^p=B^i}{=} \frac{\cancel{P(A^p|B^i)}\cancel{P(B^i)}}{\cancel{P(A^i|B^i)}\cancel{P(B^i)}}\;\frac{\cancel{P(A^i|B^i)}}{\cancel{P(A^p|B^i)}} = 1!$$

- We *always* accept the proposed point!
  - We don’t even need to compute the acceptance probability!

The Gibbs sampler

- Gibbs sampling is a special case of the Metropolis-Hastings algorithm
  - Uses knowledge about the target distribution to propose samples
  - Applicable to problems with more than one parameter (eg., P(A, B))
- Defined by a very special transition rule, namely

$$T((A,B)^i \to (A,B)^p) = P(A|B^i)\,\delta(B^p - B^i)$$

  - That is, fix one parameter at the previous state, and sample the other from the corresponding *conditional* distribution
- The MH probability is then

$$p = \frac{P(A^p,B^p)}{P(A^i,B^i)}\frac{T((A,B)^p \to (A,B)^i)}{T((A,B)^i \to (A,B)^p)} = \overbrace{\frac{P(A^p|B^p)P(B^p)}{P(A^i|B^i)P(B^i)}}^{\text{Conditional distrib.}}\ \overbrace{\frac{P(A^i|B^i)\delta(B^i-B^p)}{P(A^p|B^p)\delta(B^p-B^i)}}^{\text{Proposal density}}$$

$$\stackrel{B^p=B^i}{=} \frac{\cancel{P(A^p|B^i)}\cancel{P(B^i)}}{\cancel{P(A^i|B^i)}\cancel{P(B^i)}}\ \frac{\cancel{P(A^i|B^i)}}{\cancel{P(A^p|B^i)}} = 1!$$

- We *always* accept the proposed point!
  - We don't even need to compute the acceptance probability!

The Gibbs sampler

- Gibbs sampling is a special case of the Metropolis-Hastings algorithm
  - Uses knowledge about the target distribution to propose samples
  - Applicable to problems with more than one parameter (eg., P(A, B))
- Defined by a very special transition rule, namely

$$T((A,B)^i \rightarrow (A,B)^p) = P(A|B^i)\,\delta(B^p - B^i)$$

  - That is, fix one parameter at the previous state, and sample the other from the corresponding *conditional* distribution
- The MH probability is then

Conditional distrib. Proposal density

$$p = \frac{P(A^p,B^p)}{P(A^i,B^i)}\frac{T((A,B)^p \rightarrow (A,B)^i)}{T((A,B)^i \rightarrow (A,B)^p)} = \frac{P(A^p|B^p)P(B^p)}{P(A^i|B^i)P(B^i)}\frac{P(A^i|B^i)\delta(B^i-B^p)}{P(A^p|B^p)\delta(B^p-B^i)}$$

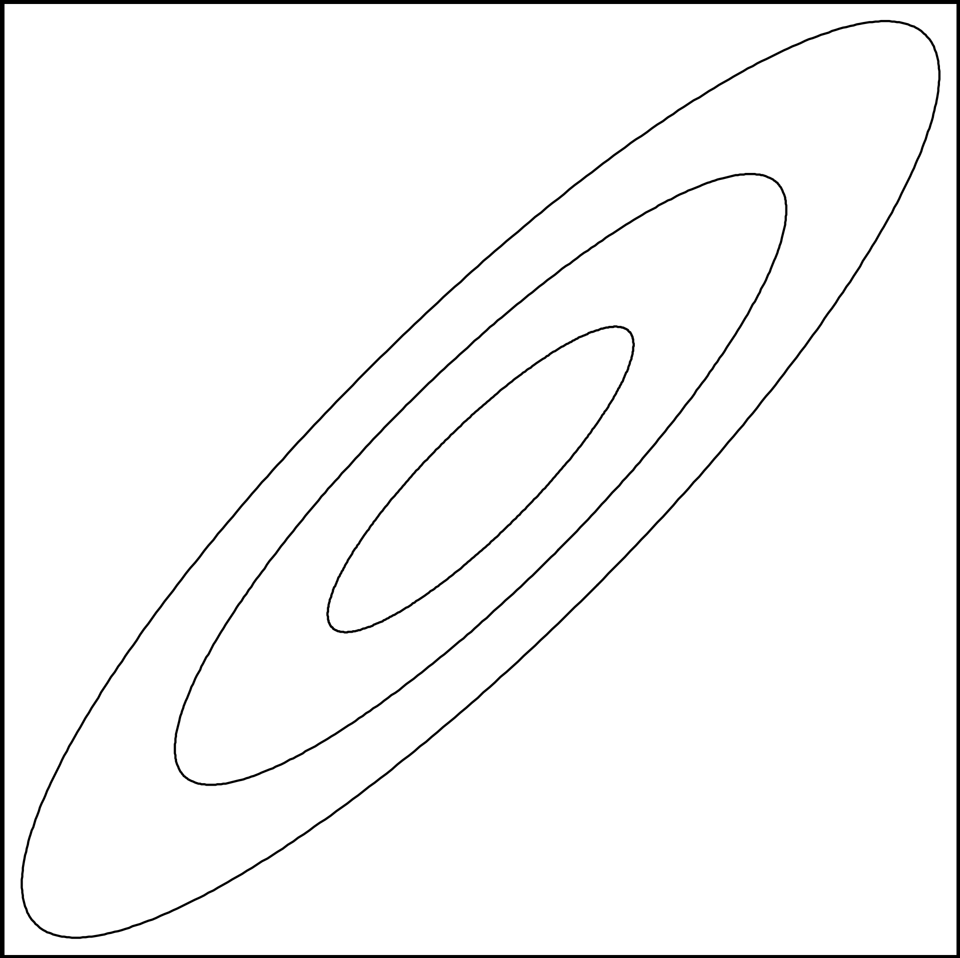
$$\overset{B^p=B^i}{=} \frac{\cancel{P(A^p|B^i)}\cancel{P(B^i)}}{\cancel{P(A^i|B^i)}\cancel{P(B^i)}}\frac{\cancel{P(A^i|B^i)}}{\cancel{P(A^p|B^i)}} = 1!$$

- We *always* accept the proposed point!
  - We don’t even need to compute the acceptance probability!
- After sampling A given B, we then turn around, and sample B given A

Schematic view of the Gibbs sampler

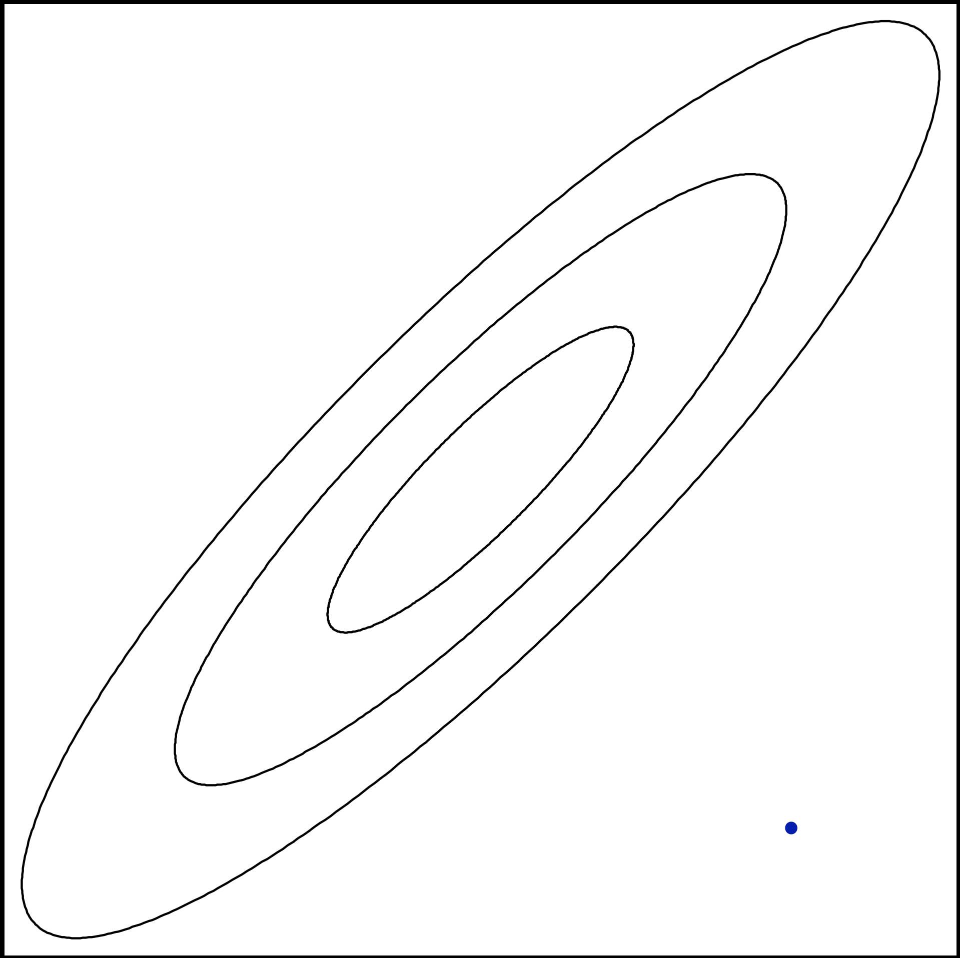
Schematic view of the Gibbs sampler

- Algorithm (for two parameters):



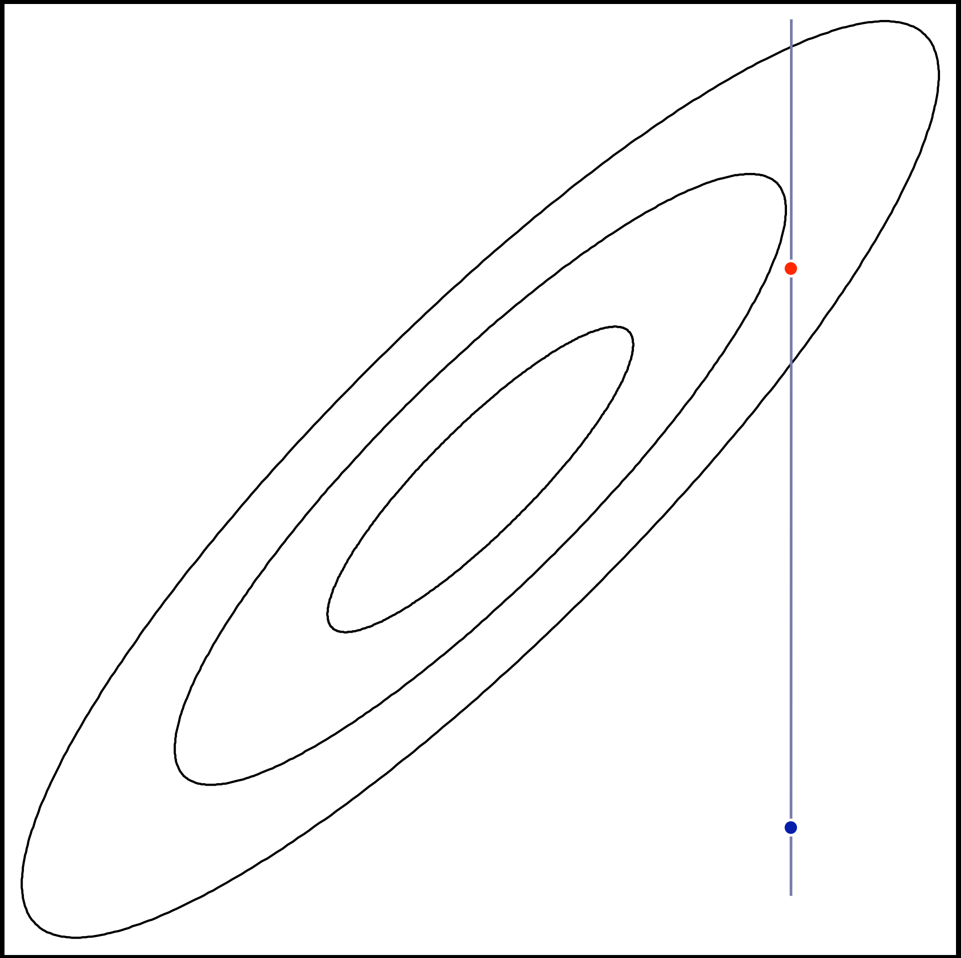
Schematic view of the Gibbs sampler

- Algorithm (for two para-meters):
  1. Choose an initial point



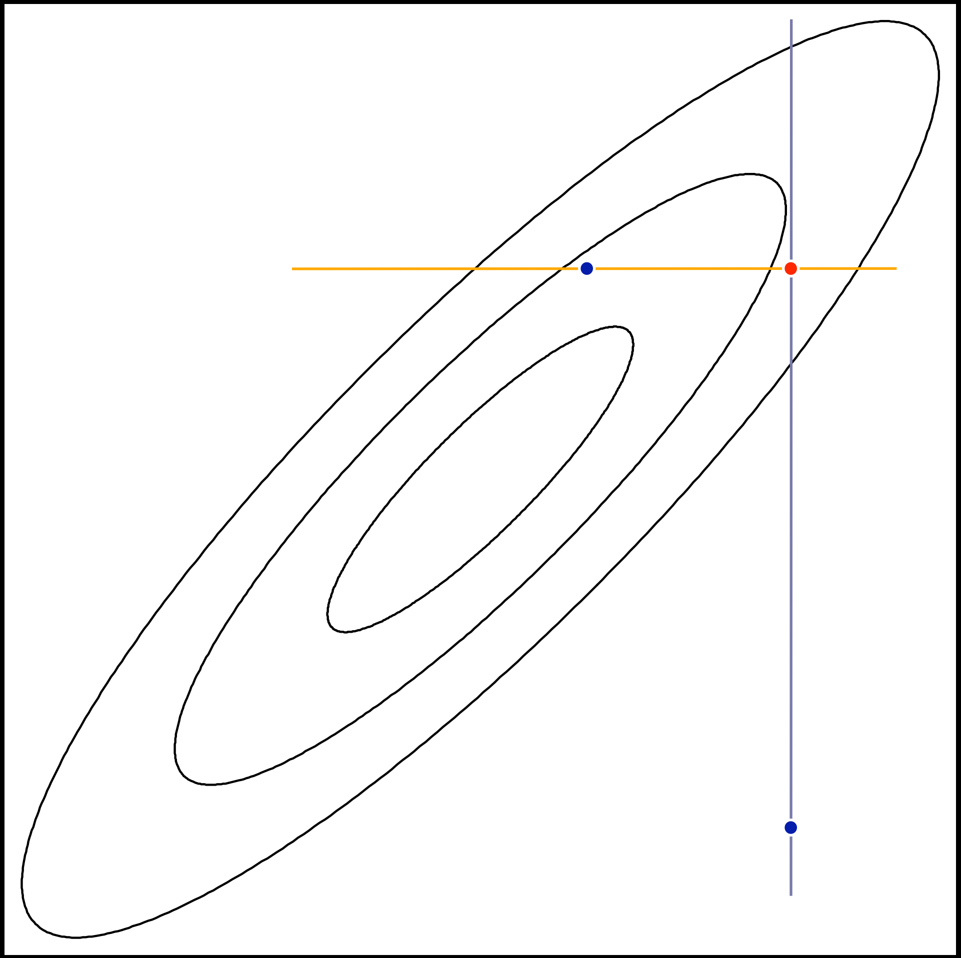
Schematic view of the Gibbs sampler

- Algorithm (for two parameters):
  1. Choose an initial point
  2. Draw A ← P(A|B)



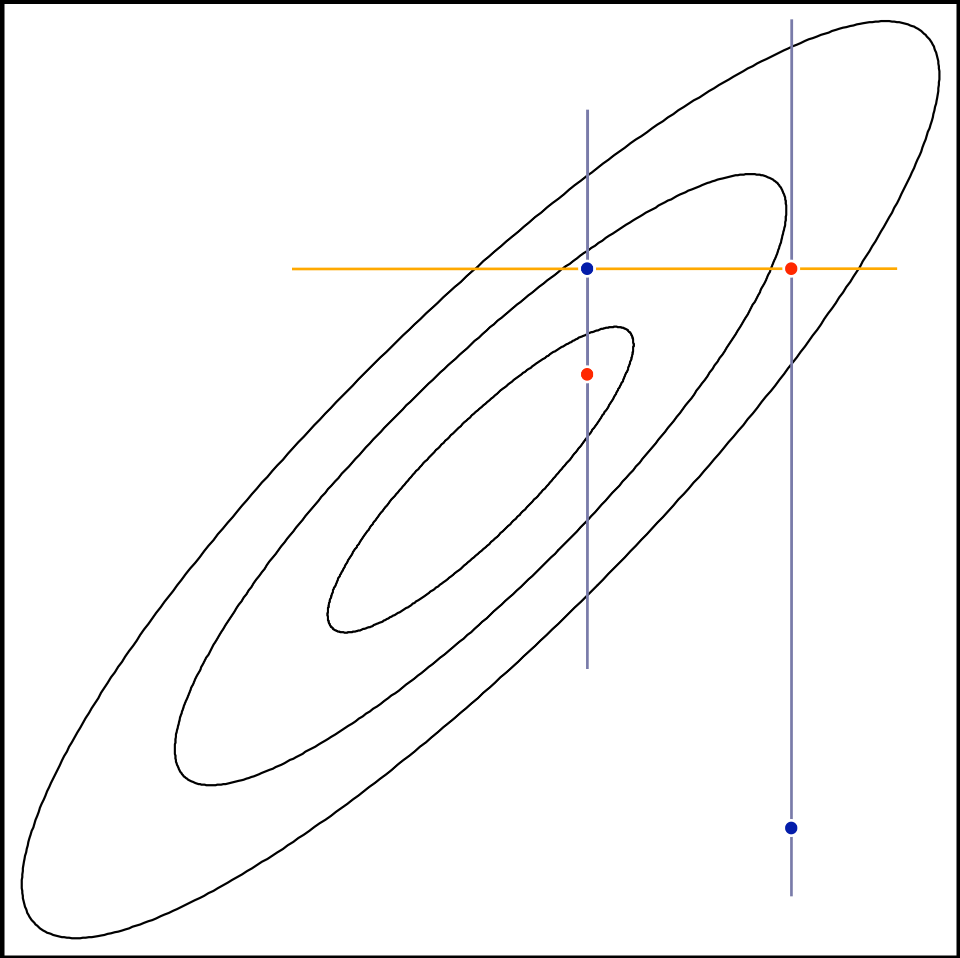
Schematic view of the Gibbs sampler

- Algorithm (for two parameters):
  1. Choose an initial point
  2. Draw A ← P(A|B)
  3. Draw B ← P(B|A)



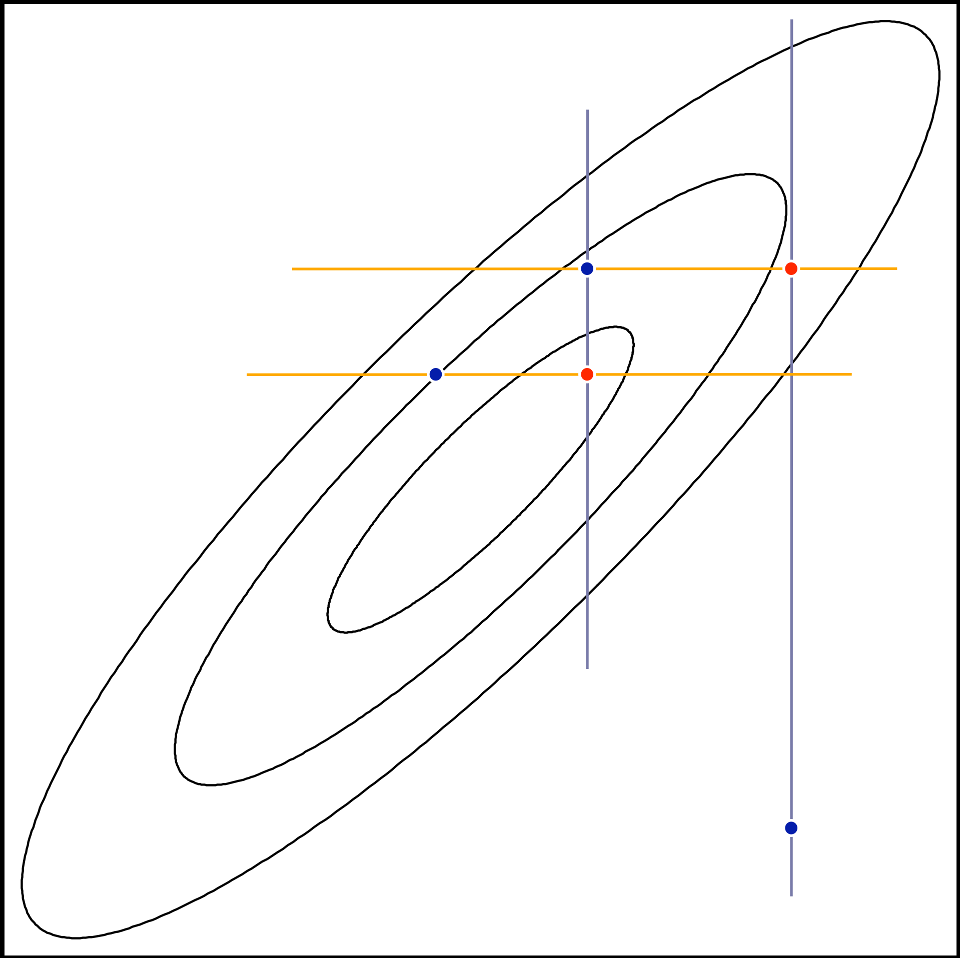
Schematic view of the Gibbs sampler

- Algorithm (for two parameters):
  1. Choose an initial point
  2. Draw A ← P(A|B)
  3. Draw B ← P(B|A)
  4. Iterate



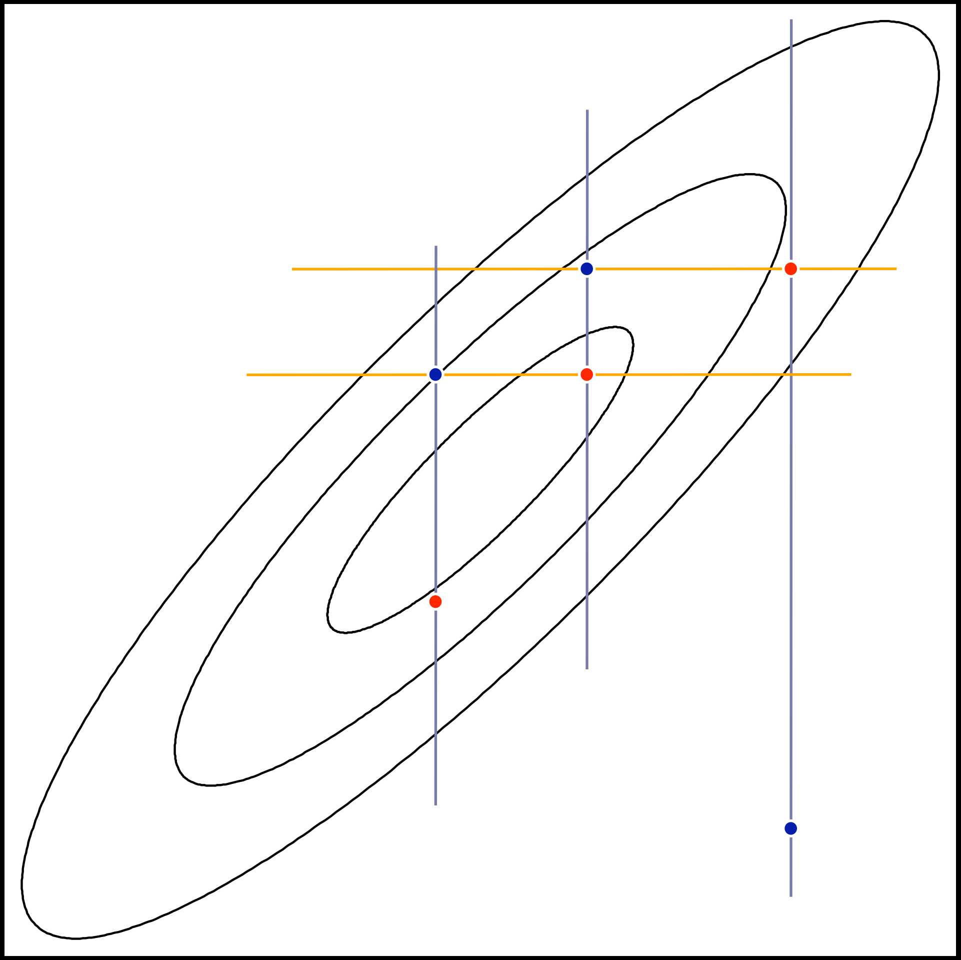
Schematic view of the Gibbs sampler

- Algorithm (for two parameters):
  1. Choose an initial point
  2. Draw A ← P(A|B)
  3. Draw B ← P(B|A)
  4. Iterate



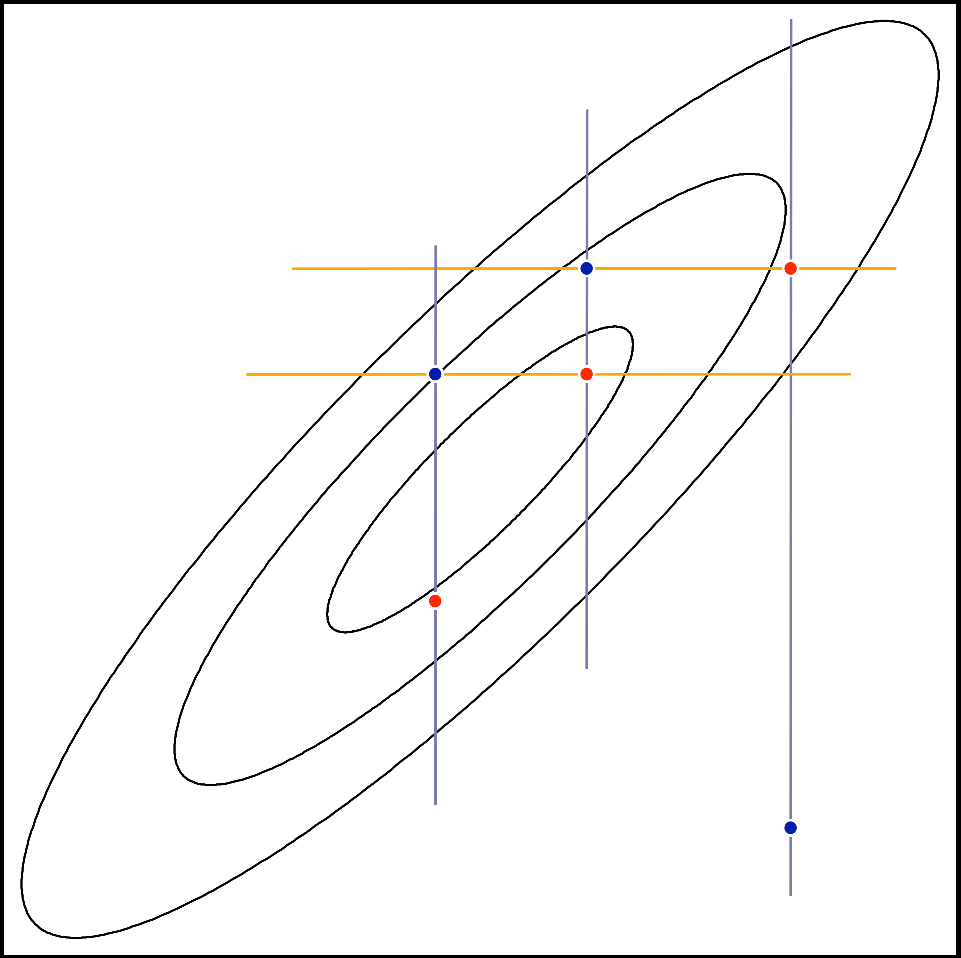
Schematic view of the Gibbs sampler

- Algorithm (for two parameters):
  1. Choose an initial point
  2. Draw A ← P(A|B)
  3. Draw B ← P(B|A)
  4. Iterate



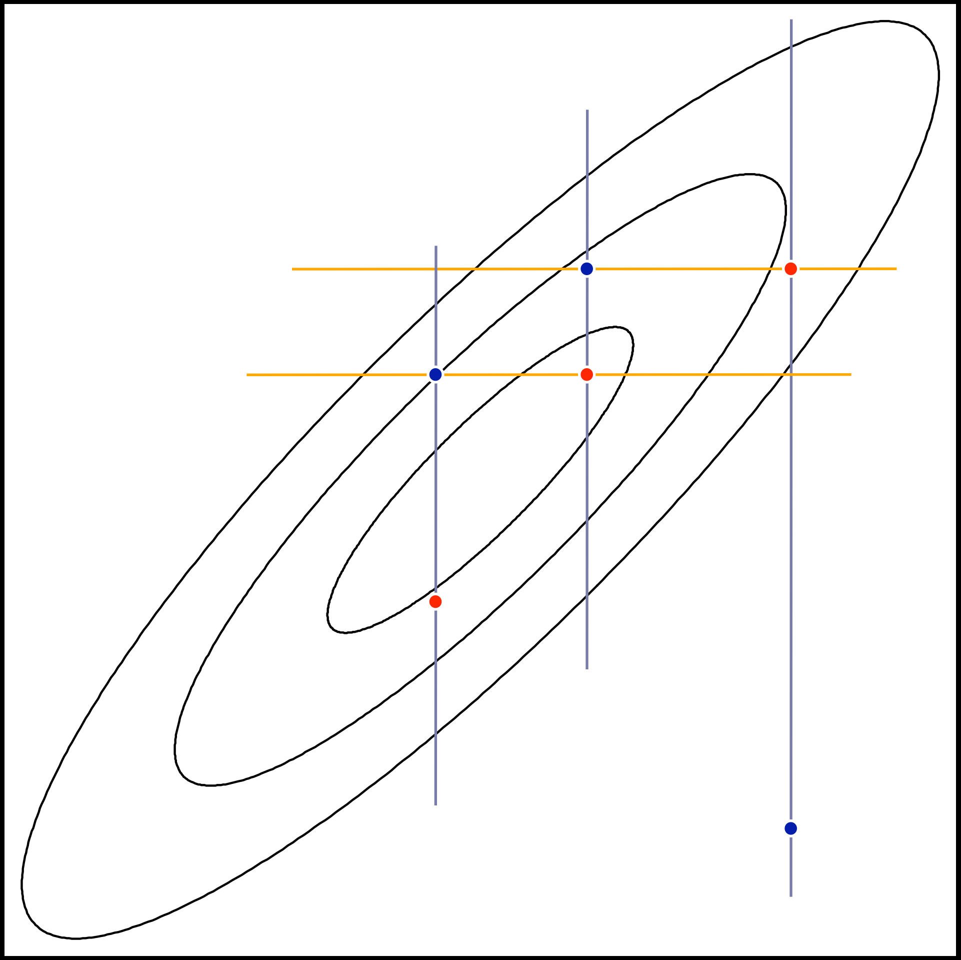
Schematic view of the Gibbs sampler

- Algorithm (for two para-meters):
  1. Choose an initial point
  2. Draw A ← P(A|B)
  3. Draw B ← P(B|A)
  4. Iterate
- Extremely useful in a wide range of applications, but...



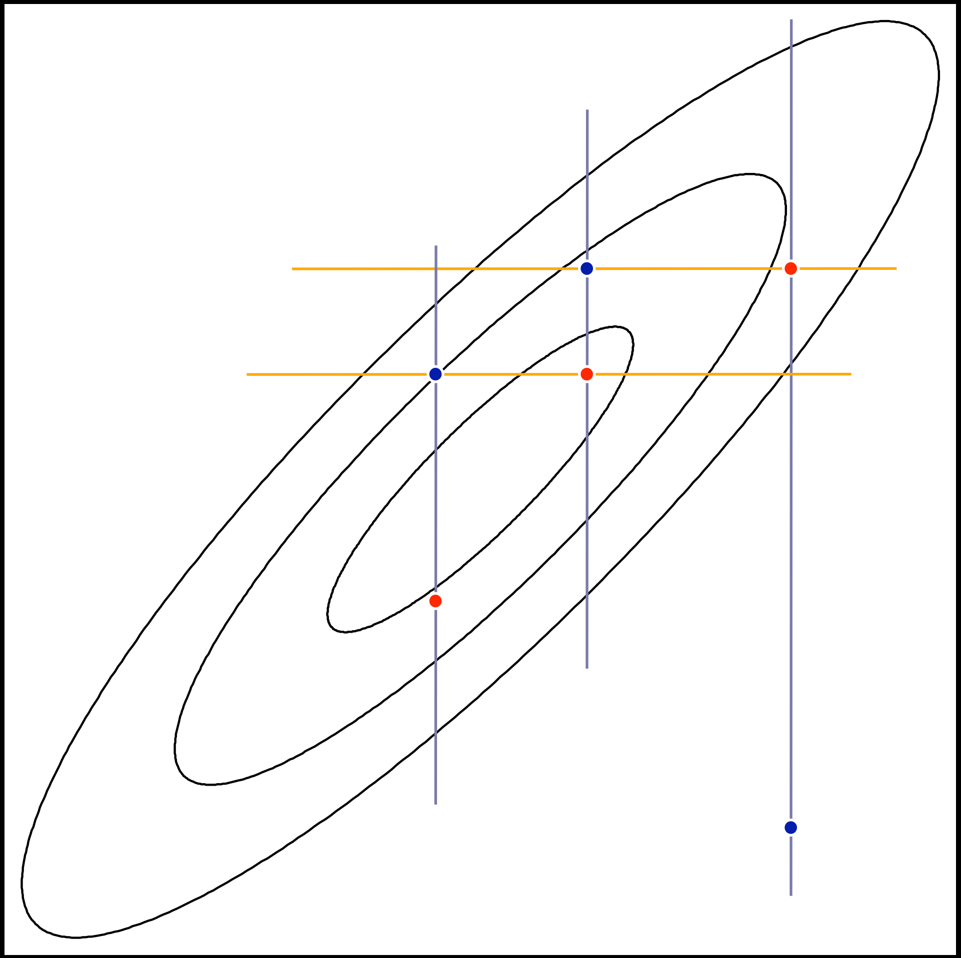
Schematic view of the Gibbs sampler

- Algorithm (for two para-meters):
  1. Choose an initial point
  2. Draw A ← P(A|B)
  3. Draw B ← P(B|A)
  4. Iterate
- Extremely useful in a wide range of applications, but...
  - you do need conditional sampling algorithms



Schematic view of the Gibbs sampler

- Algorithm (for two parameters):
  1. Choose an initial point
  2. Draw A ← P(A|B)
  3. Draw B ← P(B|A)
  4. Iterate
- Extremely useful in a wide range of applications, but...
  - you do need conditional sampling algorithms
  - it is often inefficient for strongly correlated parameters



Elementary CMB Gibbs sampling

Elementary CMB Gibbs sampling

Elementary CMB Gibbs sampling

- The overall main goal of CMB analysis is to establish an estimate of the CMB sky signal and power spectrum, and their uncertainties

Elementary CMB Gibbs sampling

- The overall main goal of CMB analysis is to establish an estimate of the CMB sky signal and power spectrum, and their uncertainties

$$\mathbf{s}^{i+1} \leftarrow P(\mathbf{s}|C_\ell^i, \mathbf{d})$$
$$C_\ell^{i+1} \leftarrow P(C_\ell|\mathbf{s}^{i+1}, \mathbf{d})$$

Elementary CMB Gibbs sampling

- The overall main goal of CMB analysis is to establish an estimate of the CMB sky signal and power spectrum, and their uncertainties

- Technically, this information is most conveniently summarized in terms of the *CMB posterior* P(**s**, C_l | **d**)

$$\mathbf{s}^{i+1} \leftarrow P(\mathbf{s}|C_\ell^i, \mathbf{d})$$
$$C_\ell^{i+1} \leftarrow P(C_\ell|\mathbf{s}^{i+1}, \mathbf{d})$$

Elementary CMB Gibbs sampling

- The overall main goal of CMB analysis is to establish an estimate of the CMB sky signal and power spectrum, and their uncertainties

- Technically, this information is most conveniently summarized in terms of the *CMB posterior* $P(\mathbf{s}, C_l \mid \mathbf{d})$

$$\mathbf{s}^{i+1} \leftarrow P(\mathbf{s}|C_\ell^i, \mathbf{d})$$
$$C_\ell^{i+1} \leftarrow P(C_\ell|\mathbf{s}^{i+1}, \mathbf{d})$$

Elementary CMB Gibbs sampling

- The overall main goal of CMB analysis is to establish an estimate of the CMB sky signal and power spectrum, and their uncertainties

- Technically, this information is most conveniently summarized in terms of the *CMB posterior* $P(\mathbf{s}, C_l \mid \mathbf{d})$

- Currently, the most efficient method for computing this posterior is through sampling, and in particular the Gibbs sampling algorithm (Jewell et at., Wandelt et al. 2004):

$$\mathbf{s}^{i+1} \leftarrow P(\mathbf{s}|C_\ell^i, \mathbf{d})$$
$$C_\ell^{i+1} \leftarrow P(C_\ell|\mathbf{s}^{i+1}, \mathbf{d})$$

Elementary CMB Gibbs sampling

- The overall main goal of CMB analysis is to establish an estimate of the CMB sky signal and power spectrum, and their uncertainties

- Technically, this information is most conveniently summarized in terms of the *CMB posterior* $P(\mathbf{s}, C_l \mid \mathbf{d})$

- Currently, the most efficient method for computing this posterior is through sampling, and in particular the Gibbs sampling algorithm (Jewell et at., Wandelt et al. 2004):

$$\mathbf{s}^{i+1} \leftarrow P(\mathbf{s}|C_\ell^i, \mathbf{d})$$
$$C_\ell^{i+1} \leftarrow P(C_\ell|\mathbf{s}^{i+1}, \mathbf{d})$$

Elementary CMB Gibbs sampling

- The overall main goal of CMB analysis is to establish an estimate of the CMB sky signal and power spectrum, and their uncertainties

- Technically, this information is most conveniently summarized in terms of the *CMB posterior* $P(\mathbf{s}, C_l \mid \mathbf{d})$

- Currently, the most efficient method for computing this posterior is through sampling, and in particular the Gibbs sampling algorithm (Jewell et at., Wandelt et al. 2004):

$$\mathbf{s}^{i+1} \leftarrow P(\mathbf{s}|C_\ell^i, \mathbf{d})$$
$$C_\ell^{i+1} \leftarrow P(C_\ell|\mathbf{s}^{i+1}, \mathbf{d})$$

Elementary CMB Gibbs sampling

- The overall main goal of CMB analysis is to establish an estimate of the CMB sky signal and power spectrum, and their uncertainties

- Technically, this information is most conveniently summarized in terms of the *CMB posterior* $P(\mathbf{s}, C_l \mid \mathbf{d})$

- Currently, the most efficient method for computing this posterior is through sampling, and in particular the Gibbs sampling algorithm (Jewell et at., Wandelt et al. 2004):

$$\mathbf{s}^{i+1} \leftarrow P(\mathbf{s}|C_\ell^i, \mathbf{d})$$
$$C_\ell^{i+1} \leftarrow P(C_\ell|\mathbf{s}^{i+1}, \mathbf{d})$$

Elementary CMB Gibbs sampling

- The overall main goal of CMB analysis is to establish an estimate of the CMB sky signal and power spectrum, and their uncertainties

- Technically, this information is most conveniently summarized in terms of the *CMB posterior* $P(\mathbf{s}, C_l \mid \mathbf{d})$

- Currently, the most efficient method for computing this posterior is through sampling, and in particular the Gibbs sampling algorithm (Jewell et at., Wandelt et al. 2004):

$$\mathbf{s}^{i+1} \leftarrow P(\mathbf{s}|C_\ell^i, \mathbf{d})$$
$$C_\ell^{i+1} \leftarrow P(C_\ell|\mathbf{s}^{i+1}, \mathbf{d})$$

Elementary CMB Gibbs sampling

- The overall main goal of CMB analysis is to establish an estimate of the CMB sky signal and power spectrum, and their uncertainties

- Technically, this information is most conveniently summarized in terms of the *CMB posterior* $P(\mathbf{s}, C_l \mid \mathbf{d})$

- Currently, the most efficient method for computing this posterior is through sampling, and in particular the Gibbs sampling algorithm (Jewell et at., Wandelt et al. 2004):

$$\mathbf{s}^{i+1} \leftarrow P(\mathbf{s}|C_\ell^i, \mathbf{d})$$
$$C_\ell^{i+1} \leftarrow P(C_\ell|\mathbf{s}^{i+1}, \mathbf{d})$$

- The first conditional is simply a Gaussian distribution, and the second is known as the inverse Wishart (Gamma) distribution

Elementary CMB Gibbs sampling

- The overall main goal of CMB analysis is to establish an estimate of the CMB sky signal and power spectrum, and their uncertainties

- Technically, this information is most conveniently summarized in terms of the *CMB posterior* $P(\mathbf{s}, C_l \mid \mathbf{d})$

- Currently, the most efficient method for computing this posterior is through sampling, and in particular the Gibbs sampling algorithm (Jewell et at., Wandelt et al. 2004):

$$\mathbf{s}^{i+1} \leftarrow P(\mathbf{s}|C_\ell^i, \mathbf{d})$$
$$C_\ell^{i+1} \leftarrow P(C_\ell|\mathbf{s}^{i+1}, \mathbf{d})$$

- The first conditional is simply a Gaussian distribution, and the second is known as the inverse Wishart (Gamma) distribution

- Our goal is to extend this framework to take into account all relevant sources of uncertainty, and especially non-cosmological foregrounds

Data model including foregrounds

Data model including foregrounds

- A fairly general model of the observed CMB data may be written on the form:

$$\mathbf{d}_\nu = \mathbf{A}_\nu(\mathbf{s} + \mathbf{f}_\nu(\theta)) + \mathbf{n}_\nu$$

Data model including foregrounds

- A fairly general model of the observed CMB data may be written on the form:

$$\mathbf{d}_\nu = \mathbf{A}_\nu(\mathbf{s} + \mathbf{f}_\nu(\theta)) + \mathbf{n}_\nu$$

where

  - **d** = Observed data at frequency υ

Data model including foregrounds

- A fairly general model of the observed CMB data may be written on the form:

$$\mathbf{d}_\nu = \mathbf{A}_\nu(\mathbf{s} + \mathbf{f}_\nu(\theta)) + \mathbf{n}_\nu$$

where

  - **d** = Observed data at frequency υ
  - **s** = CMB sky signal

Data model including foregrounds

- A fairly general model of the observed CMB data may be written on the form:

$$\mathbf{d}_\nu = \mathbf{A}_\nu(\mathbf{s} + \mathbf{f}_\nu(\theta)) + \mathbf{n}_\nu$$

where

  - **d** = Observed data at frequency υ
  - **s** = CMB sky signal
  - **f** = Foreground sky signal with parameters θ

Data model including foregrounds

- A fairly general model of the observed CMB data may be written on the form:

$$\mathbf{d}_\nu = \mathbf{A}_\nu(\mathbf{s} + \mathbf{f}_\nu(\theta)) + \mathbf{n}_\nu$$

where

  - **d** = Observed data at frequency υ
  - **s** = CMB sky signal
  - **f** = Foreground sky signal with parameters θ
  - **n** = Instrumental noise

Data model including foregrounds

- A fairly general model of the observed CMB data may be written on the form:

$$\mathbf{d}_\nu = \mathbf{A}_\nu(\mathbf{s} + \mathbf{f}_\nu(\theta)) + \mathbf{n}_\nu$$

where

- **d** = Observed data at frequency υ
- **s** = CMB sky signal
- **f** = Foreground sky signal with parameters θ
- **n** = Instrumental noise
- **A** = Point spread function (beam)

Data model including foregrounds

- A fairly general model of the observed CMB data may be written on the form:

$$\mathbf{d}_\nu = \mathbf{A}_\nu(\mathbf{s} + \mathbf{f}_\nu(\theta)) + \mathbf{n}_\nu$$

where

- **d** = Observed data at frequency υ
- **s** = CMB sky signal
- **f** = Foreground sky signal with parameters θ
- **n** = Instrumental noise
- **A** = Point spread function (beam)

Data model including foregrounds

- A fairly general model of the observed CMB data may be written on the form:

$$\mathbf{d}_\nu = \mathbf{A}_\nu(\mathbf{s} + \mathbf{f}_\nu(\theta)) + \mathbf{n}_\nu$$

  where

  - **d** = Observed data at frequency υ
  - **s** = CMB sky signal
  - **f** = Foreground sky signal with parameters θ
  - **n** = Instrumental noise
  - **A** = Point spread function (beam)

- All quantities are (T, Q, U) vectors

Data model including foregrounds

- A fairly general model of the observed CMB data may be written on the form:

$$\mathbf{d}_\nu = \mathbf{A}_\nu(\mathbf{s} + \mathbf{f}_\nu(\theta)) + \mathbf{n}_\nu$$

  where
  - **d** = Observed data at frequency υ
  - **s** = CMB sky signal
  - **f** = Foreground sky signal with parameters θ
  - **n** = Instrumental noise
  - **A** = Point spread function (beam)

- All quantities are (T, Q, U) vectors

Data model including foregrounds

- A fairly general model of the observed CMB data may be written on the form:

$$\mathbf{d}_\nu = \mathbf{A}_\nu(\mathbf{s} + \mathbf{f}_\nu(\theta)) + \mathbf{n}_\nu$$

  where
  - **d** = Observed data at frequency υ
  - **s** = CMB sky signal
  - **f** = Foreground sky signal with parameters θ
  - **n** = Instrumental noise
  - **A** = Point spread function (beam)

- All quantities are (T, Q, U) vectors

- We assume both the CMB and noise to be Gaussian distributed, with a CMB covariance matrices defined in harmonic space by

$$\langle \mathbf{s}_\ell \mathbf{s}_\ell^t \rangle = \begin{pmatrix} C_\ell^{\mathrm{TT}} & C_\ell^{\mathrm{TE}} & C_\ell^{\mathrm{TB}} \\ C_\ell^{\mathrm{TE}} & C_\ell^{\mathrm{EE}} & C_\ell^{\mathrm{EB}} \\ C_\ell^{\mathrm{TB}} & C_\ell^{\mathrm{EB}} & C_\ell^{\mathrm{BB}} \end{pmatrix} \delta_{\ell\ell}$$

The CMB-foreground Gibbs sampler

The CMB-foreground Gibbs sampler

- Again, recall that it is possible to map out the full CMB posterior P(s, C_l | d) by Gibbs sampling:

$$\mathbf{s}^{i+1} \leftarrow P(\mathbf{s}|C_\ell^i, \mathbf{d})$$

$$C_\ell^{i+1} \leftarrow P(C_\ell|\mathbf{s}^{i+1}, \mathbf{d})$$

The CMB-foreground Gibbs sampler

- Again, recall that it is possible to map out the full CMB posterior P(s, C_l | d) by Gibbs sampling:

$$\mathbf{s}^{i+1} \leftarrow P(\mathbf{s}|C_\ell^i, \mathbf{d})$$

$$C_\ell^{i+1} \leftarrow P(C_\ell|\mathbf{s}^{i+1}, \mathbf{d})$$

$$\mathbf{s}^{i+1} \leftarrow P(\mathbf{s}|C_\ell^i, \mathbf{f}^i, \mathbf{d})$$

Gaussian; sampled as before

The CMB-foreground Gibbs sampler

- Again, recall that it is possible to map out the full CMB posterior P(s, C_l | d) by Gibbs sampling:

$$\mathbf{s}^{i+1} \leftarrow P(\mathbf{s}|C_\ell^i, \mathbf{d})$$
$$C_\ell^{i+1} \leftarrow P(C_\ell|\mathbf{s}^{i+1}, \mathbf{d})$$

$$\mathbf{s}^{i+1} \leftarrow P(\mathbf{s}|C_\ell^i, \mathbf{f}^i, \mathbf{d})$$ Gaussian; sampled as before

$$C_\ell^{i+1} \leftarrow P(C_\ell|\mathbf{s}^{i+1}, \mathbf{f}^i, \mathbf{d})$$ Inverse Wishart; sampled as before

The CMB-foreground Gibbs sampler

- Again, recall that it is possible to map out the full CMB posterior P(s, C_l | d) by Gibbs sampling:

$$\mathbf{s}^{i+1} \leftarrow P(\mathbf{s}|C_\ell^i, \mathbf{d})$$

$$C_\ell^{i+1} \leftarrow P(C_\ell|\mathbf{s}^{i+1}, \mathbf{d})$$

$\mathbf{s}^{i+1} \leftarrow P(\mathbf{s}\|C_\ell^i, \mathbf{f}^i, \mathbf{d})$	Gaussian; sampled as before
$C_\ell^{i+1} \leftarrow P(C_\ell\|\mathbf{s}^{i+1}, \mathbf{f}^i, \mathbf{d})$	Inverse Wishart; sampled as before
$\mathbf{f}^{i+1} \leftarrow P(\mathbf{f}\|C_\ell^i, \mathbf{s}^{i+1}, \mathbf{d})$	Arbitrary; χ^2 inversion sampler

The CMB-foreground Gibbs sampler

- Again, recall that it is possible to map out the full CMB posterior P(s, C_l | d) by Gibbs sampling:

$$\mathbf{s}^{i+1} \leftarrow P(\mathbf{s}|C_\ell^i, \mathbf{d})$$
$$C_\ell^{i+1} \leftarrow P(C_\ell|\mathbf{s}^{i+1}, \mathbf{d})$$

$$\mathbf{s}^{i+1} \leftarrow P(\mathbf{s}|C_\ell^i, \mathbf{f}^i, \mathbf{d})$$ Gaussian; sampled as before

$$C_\ell^{i+1} \leftarrow P(C_\ell|\mathbf{s}^{i+1}, \mathbf{f}^i, \mathbf{d})$$ Inverse Wishart; sampled as before

$$\mathbf{f}^{i+1} \leftarrow P(\mathbf{f}|C_\ell^i, \mathbf{s}^{i+1}, \mathbf{d})$$ Arbitrary; χ^2 inversion sampler

The CMB-foreground Gibbs sampler

- Again, recall that it is possible to map out the full CMB posterior P(s, C_l | d) by Gibbs sampling:

$$\mathbf{s}^{i+1} \leftarrow P(\mathbf{s}|C_\ell^i, \mathbf{d})$$
$$C_\ell^{i+1} \leftarrow P(C_\ell|\mathbf{s}^{i+1}, \mathbf{d})$$

$$\mathbf{s}^{i+1} \leftarrow P(\mathbf{s}|C_\ell^i, \mathbf{f}^i, \mathbf{d})$$
Gaussian; sampled as before

$$C_\ell^{i+1} \leftarrow P(C_\ell|\mathbf{s}^{i+1}, \mathbf{f}^i, \mathbf{d})$$
Inverse Wishart; sampled as before

$$\mathbf{f}^{i+1} \leftarrow P(\mathbf{f}|C_\ell^i, \mathbf{s}^{i+1}, \mathbf{d})$$
Arbitrary; χ^2 inversion sampler

The CMB-foreground Gibbs sampler

- Again, recall that it is possible to map out the full CMB posterior P(s, C_l | d) by Gibbs sampling:

$$\mathbf{s}^{i+1} \leftarrow P(\mathbf{s}|C_\ell^i, \mathbf{d})$$
$$C_\ell^{i+1} \leftarrow P(C_\ell|\mathbf{s}^{i+1}, \mathbf{d})$$

- Introducing foregrounds, the corresponding Gibbs sampler simply looks as follows

$$\mathbf{s}^{i+1} \leftarrow P(\mathbf{s}|C_\ell^i, \mathbf{f}^i, \mathbf{d})$$ Gaussian; sampled as before

$$C_\ell^{i+1} \leftarrow P(C_\ell|\mathbf{s}^{i+1}, \mathbf{f}^i, \mathbf{d})$$ Inverse Wishart; sampled as before

$$\mathbf{f}^{i+1} \leftarrow P(\mathbf{f}|C_\ell^i, \mathbf{s}^{i+1}, \mathbf{d})$$ Arbitrary; χ^2 inversion sampler

The CMB-foreground Gibbs sampler

- Again, recall that it is possible to map out the full CMB posterior P(s, C_l | d) by Gibbs sampling:

$$\mathbf{s}^{i+1} \leftarrow P(\mathbf{s}|C_\ell^i, \mathbf{d})$$
$$C_\ell^{i+1} \leftarrow P(C_\ell|\mathbf{s}^{i+1}, \mathbf{d})$$

- Introducing foregrounds, the corresponding Gibbs sampler simply looks as follows

$$\mathbf{s}^{i+1} \leftarrow P(\mathbf{s}|C_\ell^i, \mathbf{f}^i, \mathbf{d})$$ Gaussian; sampled as before

$$C_\ell^{i+1} \leftarrow P(C_\ell|\mathbf{s}^{i+1}, \mathbf{f}^i, \mathbf{d})$$ Inverse Wishart; sampled as before

$$\mathbf{f}^{i+1} \leftarrow P(\mathbf{f}|C_\ell^i, \mathbf{s}^{i+1}, \mathbf{d})$$ Arbitrary; χ^2 inversion sampler

The CMB-foreground Gibbs sampler

- Again, recall that it is possible to map out the full CMB posterior P(s, C_l | d) by Gibbs sampling:

$$\mathbf{s}^{i+1} \leftarrow P(\mathbf{s}|C_\ell^i, \mathbf{d})$$
$$C_\ell^{i+1} \leftarrow P(C_\ell|\mathbf{s}^{i+1}, \mathbf{d})$$

- Introducing foregrounds, the corresponding Gibbs sampler simply looks as follows

$$\mathbf{s}^{i+1} \leftarrow P(\mathbf{s}|C_\ell^i, \mathbf{f}^i, \mathbf{d})$$ Gaussian; sampled as before

$$C_\ell^{i+1} \leftarrow P(C_\ell|\mathbf{s}^{i+1}, \mathbf{f}^i, \mathbf{d})$$ Inverse Wishart; sampled as before

$$\mathbf{f}^{i+1} \leftarrow P(\mathbf{f}|C_\ell^i, \mathbf{s}^{i+1}, \mathbf{d})$$ Arbitrary; χ^2 inversion sampler

The CMB-foreground Gibbs sampler

- Again, recall that it is possible to map out the full CMB posterior P(s, C_l | d) by Gibbs sampling:

$$\mathbf{s}^{i+1} \leftarrow P(\mathbf{s}|C_\ell^i, \mathbf{d})$$
$$C_\ell^{i+1} \leftarrow P(C_\ell|\mathbf{s}^{i+1}, \mathbf{d})$$

- Introducing foregrounds, the corresponding Gibbs sampler simply looks as follows

$$\mathbf{s}^{i+1} \leftarrow P(\mathbf{s}|C_\ell^i, \mathbf{f}^i, \mathbf{d})$$ Gaussian; sampled as before

$$C_\ell^{i+1} \leftarrow P(C_\ell|\mathbf{s}^{i+1}, \mathbf{f}^i, \mathbf{d})$$ Inverse Wishart; sampled as before

$$\mathbf{f}^{i+1} \leftarrow P(\mathbf{f}|C_\ell^i, \mathbf{s}^{i+1}, \mathbf{d})$$ Arbitrary; χ^2 inversion sampler

The CMB-foreground Gibbs sampler

- Again, recall that it is possible to map out the full CMB posterior P(s, C_l | d) by Gibbs sampling:

$$\mathbf{s}^{i+1} \leftarrow P(\mathbf{s}|C_\ell^i, \mathbf{d})$$
$$C_\ell^{i+1} \leftarrow P(C_\ell|\mathbf{s}^{i+1}, \mathbf{d})$$

- Introducing foregrounds, the corresponding Gibbs sampler simply looks as follows

$$\mathbf{s}^{i+1} \leftarrow P(\mathbf{s}|C_\ell^i, \mathbf{f}^i, \mathbf{d})$$ Gaussian; sampled as before

$$C_\ell^{i+1} \leftarrow P(C_\ell|\mathbf{s}^{i+1}, \mathbf{f}^i, \mathbf{d})$$ Inverse Wishart; sampled as before

$$\mathbf{f}^{i+1} \leftarrow P(\mathbf{f}|C_\ell^i, \mathbf{s}^{i+1}, \mathbf{d})$$ Arbitrary; χ^2 inversion sampler

The CMB-foreground Gibbs sampler

- Again, recall that it is possible to map out the full CMB posterior P(s, C_l | d) by Gibbs sampling:

$$\mathbf{s}^{i+1} \leftarrow P(\mathbf{s}|C_\ell^i, \mathbf{d})$$
$$C_\ell^{i+1} \leftarrow P(C_\ell|\mathbf{s}^{i+1}, \mathbf{d})$$

- Introducing foregrounds, the corresponding Gibbs sampler simply looks as follows

$$\mathbf{s}^{i+1} \leftarrow P(\mathbf{s}|C_\ell^i, \mathbf{f}^i, \mathbf{d})$$ Gaussian; sampled as before

$$C_\ell^{i+1} \leftarrow P(C_\ell|\mathbf{s}^{i+1}, \mathbf{f}^i, \mathbf{d})$$ Inverse Wishart; sampled as before

$$\mathbf{f}^{i+1} \leftarrow P(\mathbf{f}|C_\ell^i, \mathbf{s}^{i+1}, \mathbf{d})$$ Arbitrary; χ^2 inversion sampler

The CMB-foreground Gibbs sampler

- Again, recall that it is possible to map out the full CMB posterior P(s, C_l | d) by Gibbs sampling:

$$\mathbf{s}^{i+1} \leftarrow P(\mathbf{s}|C_\ell^i, \mathbf{d})$$
$$C_\ell^{i+1} \leftarrow P(C_\ell|\mathbf{s}^{i+1}, \mathbf{d})$$

- Introducing foregrounds, the corresponding Gibbs sampler simply looks as follows

$$\mathbf{s}^{i+1} \leftarrow P(\mathbf{s}|C_\ell^i, \mathbf{f}^i, \mathbf{d})$$ Gaussian; sampled as before

$$C_\ell^{i+1} \leftarrow P(C_\ell|\mathbf{s}^{i+1}, \mathbf{f}^i, \mathbf{d})$$ Inverse Wishart; sampled as before

$$\mathbf{f}^{i+1} \leftarrow P(\mathbf{f}|C_\ell^i, \mathbf{s}^{i+1}, \mathbf{d})$$ Arbitrary; χ^2 inversion sampler

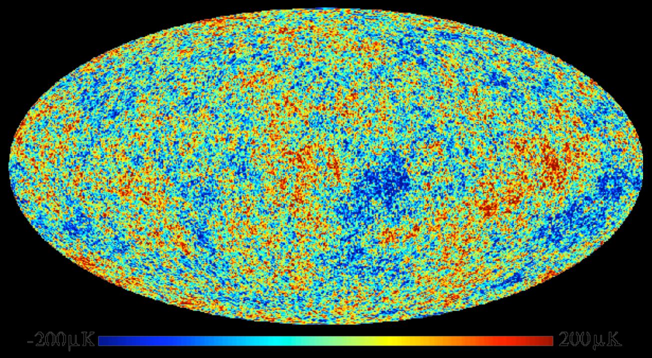
- Explicitly, while the CMB sampler is virtually unchanged, the foreground component is sampled based on the χ^2

Example: Simulated Planck data

Example: Simulated Planck data

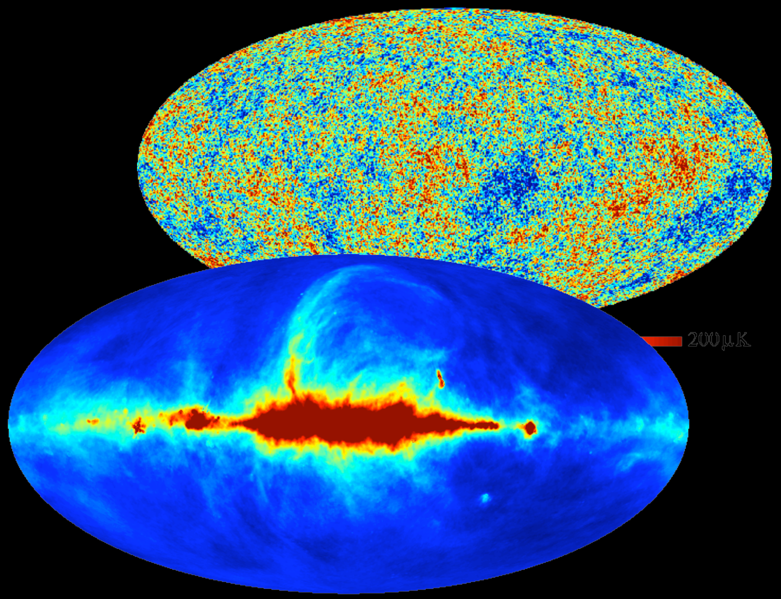
Example: Simulated Planck data

- Sky signal includes all ”known” components (”Planck sky model”):



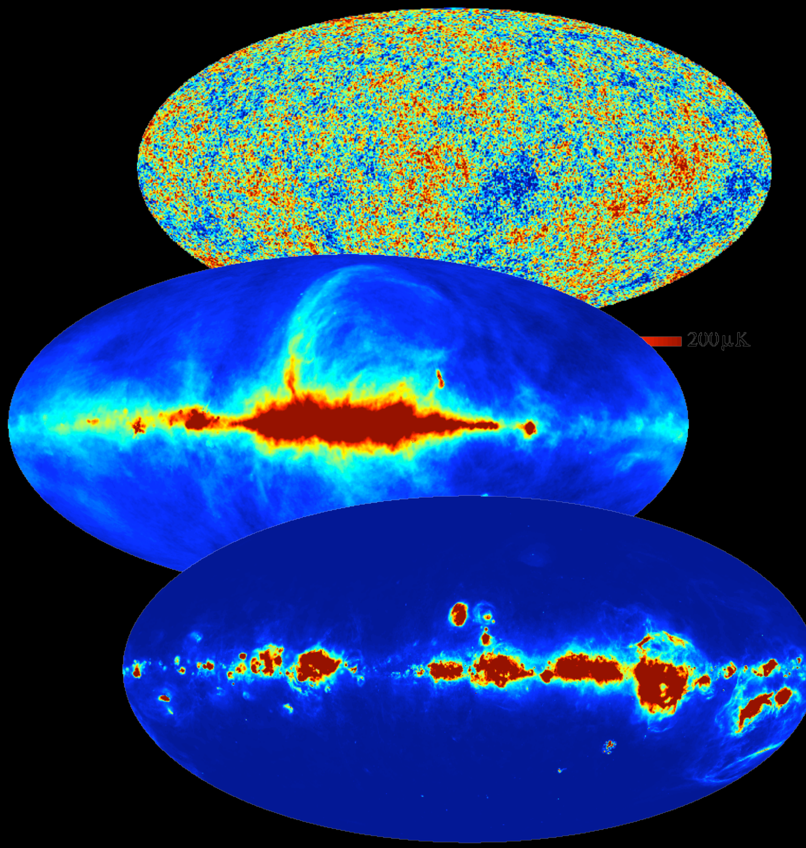
Example: Simulated Planck data

- Sky signal includes all ”known” components (”Planck sky model”):
  - Gaussian CMB signal



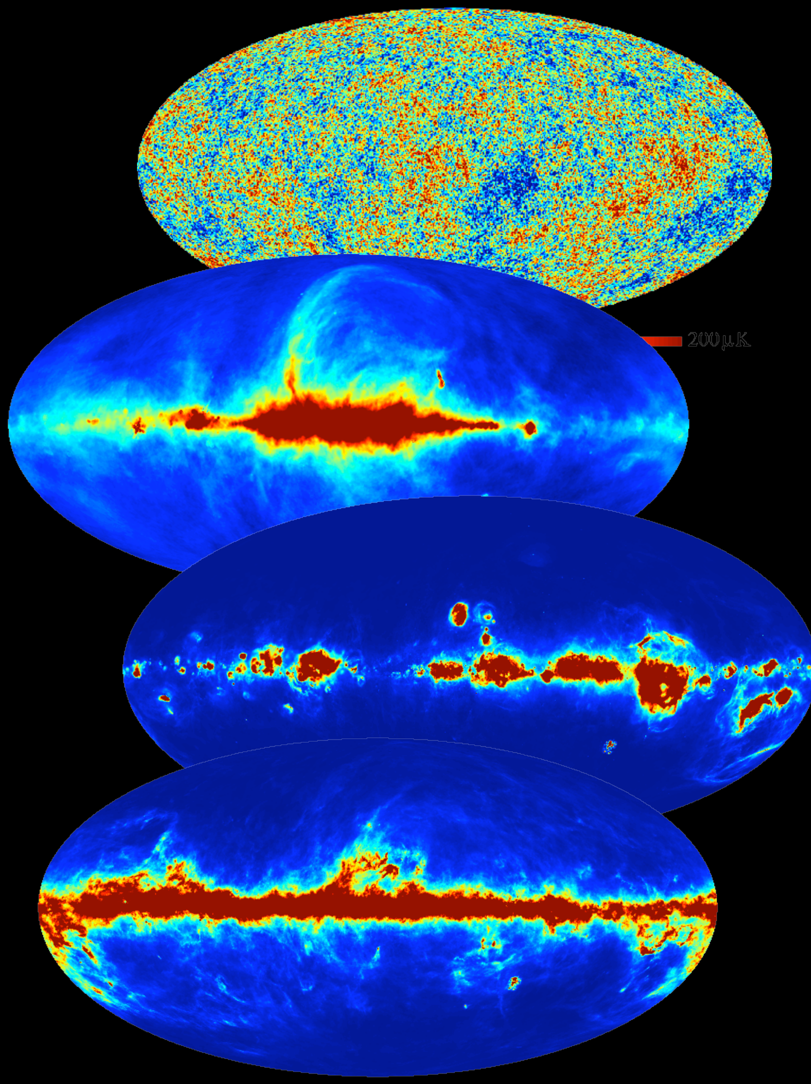
Example: Simulated Planck data

- Sky signal includes all ”known” components (”Planck sky model”):
  - Gaussian CMB signal
  - Synchrotron emission, with a spatially varying spectral index at each pixel



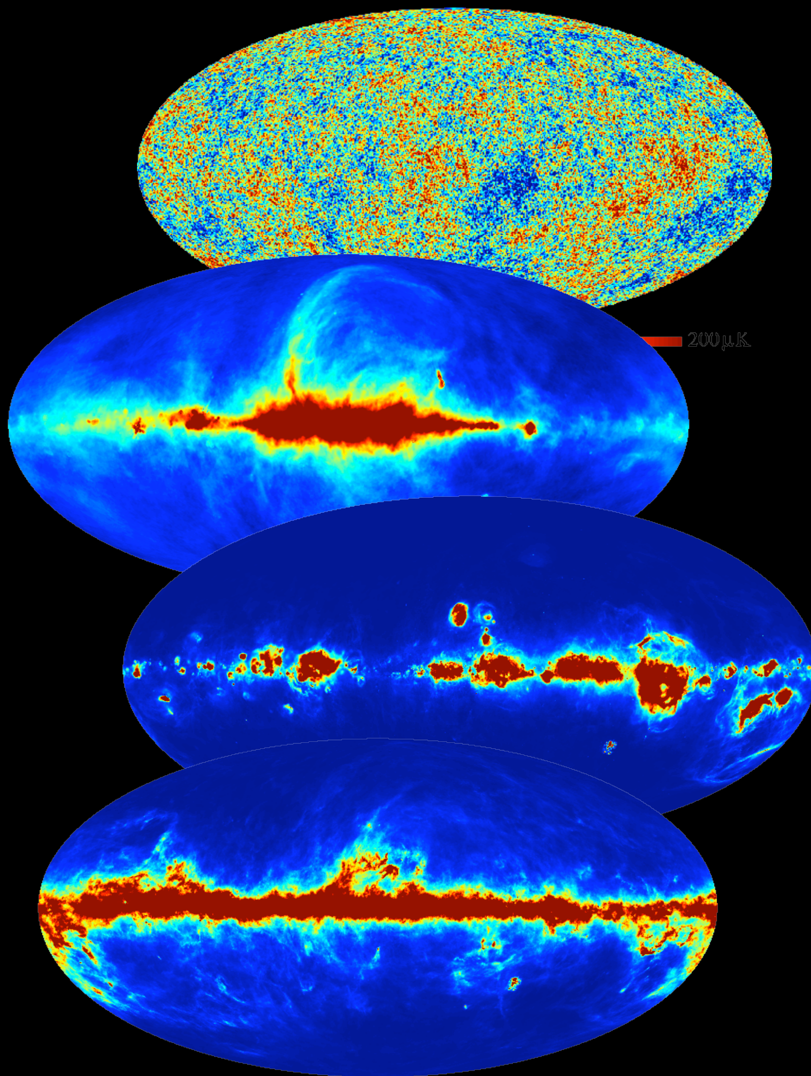
Example: Simulated Planck data

- Sky signal includes all ”known” components (”Planck sky model”):
  - Gaussian CMB signal
  - Synchrotron emission, with a spatially varying spectral index at each pixel
  - Free-free emission with a fixed spectral index of β = -2.15



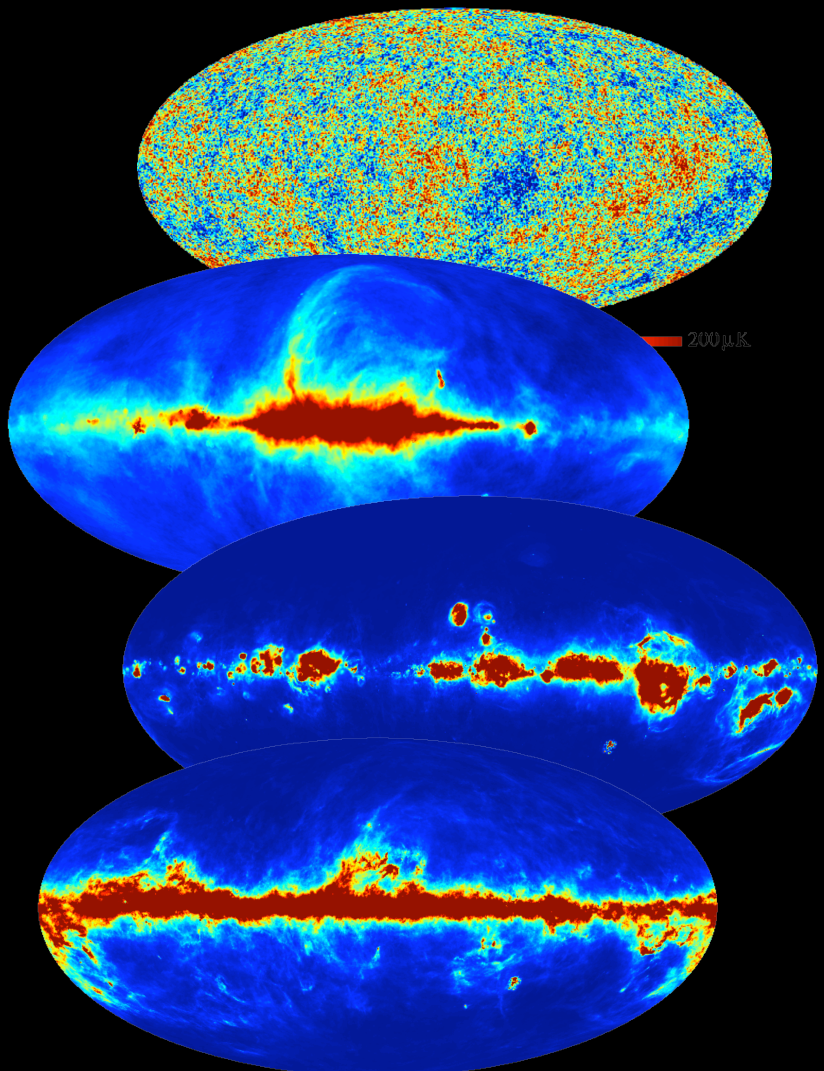
Example: Simulated Planck data

- Sky signal includes all ”known” components (”Planck sky model”):
  - Gaussian CMB signal
  - Synchrotron emission, with a spatially varying spectral index at each pixel
  - Free-free emission with a fixed spectral index of $\beta = -2.15$
  - Thermal dust based on FDS model 8



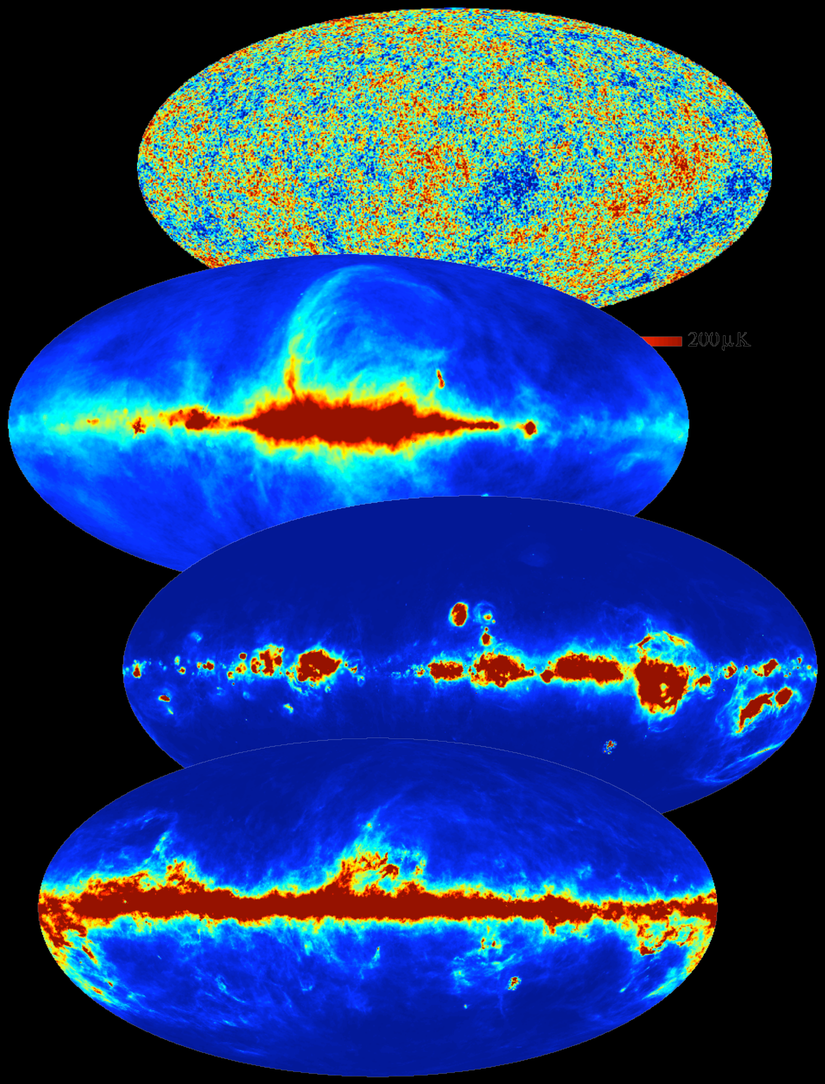
Example: Simulated Planck data

- Sky signal includes all ”known” components (”Planck sky model”):
  - Gaussian CMB signal
  - Synchrotron emission, with a spatially varying spectral index at each pixel
  - Free-free emission with a fixed spectral index of β = -2.15
  - Thermal dust based on FDS model 8
  - Spinning dust



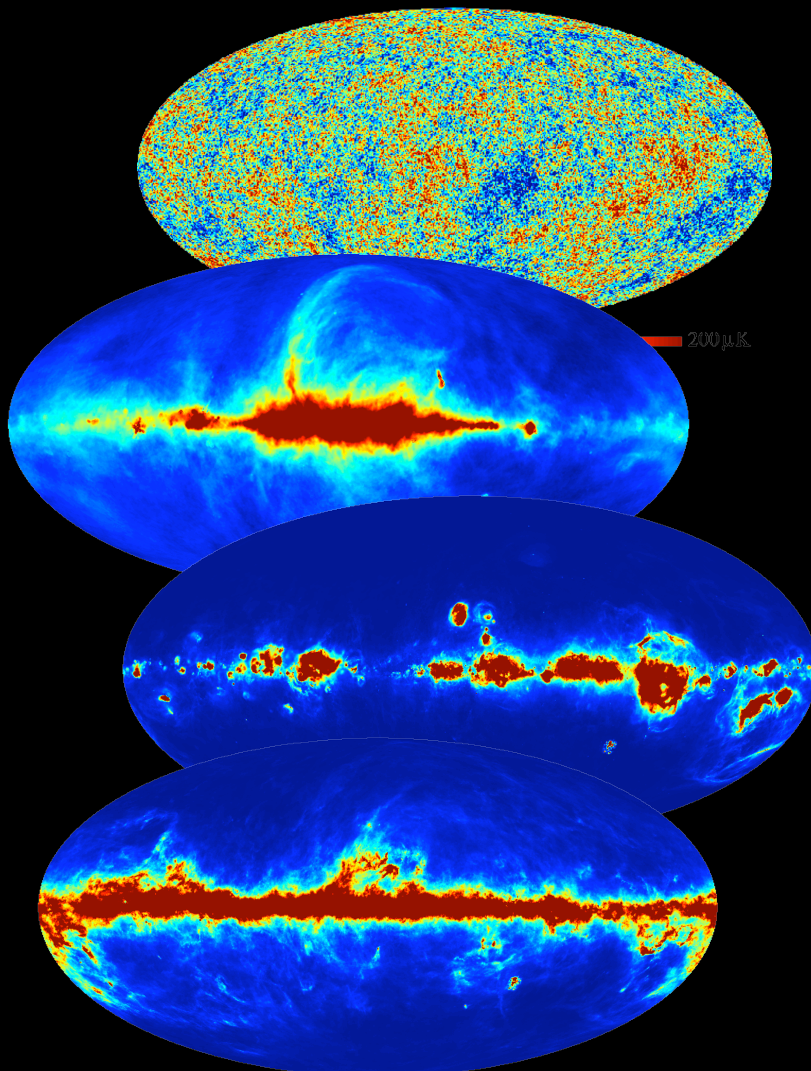
Example: Simulated Planck data

- Sky signal includes all ”known” components (”Planck sky model”):
  - Gaussian CMB signal
  - Synchrotron emission, with a spatially varying spectral index at each pixel
  - Free-free emission with a fixed spectral index of β = -2.15
  - Thermal dust based on FDS model 8
  - Spinning dust
  - The Sunyaev-Zeldovich effect



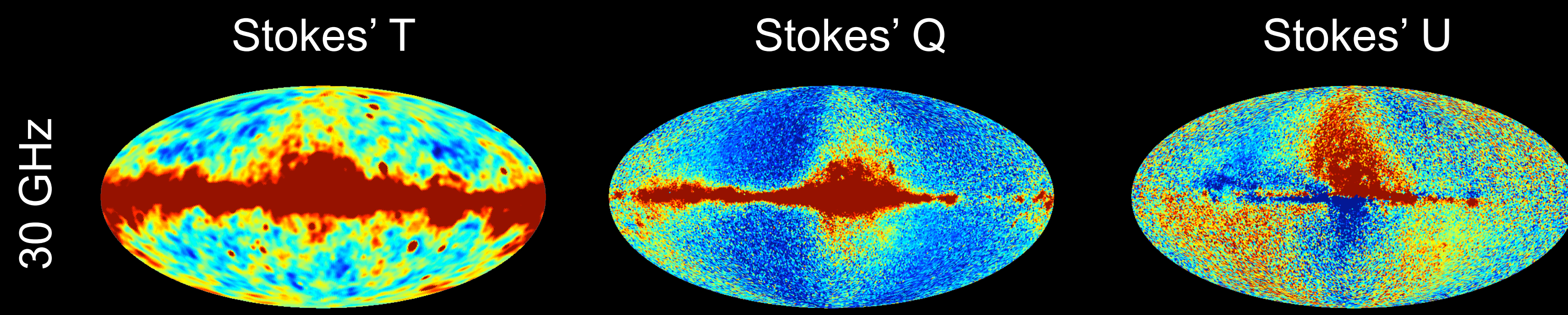
Example: Simulated Planck data

- Sky signal includes all ”known” components (”Planck sky model”):
  - Gaussian CMB signal
  - Synchrotron emission, with a spatially varying spectral index at each pixel
  - Free-free emission with a fixed spectral index of β = -2.15
  - Thermal dust based on FDS model 8
  - Spinning dust
  - The Sunyaev-Zeldovich effect
  - Three populations of point sources



Simulated sky maps

Simulated sky maps

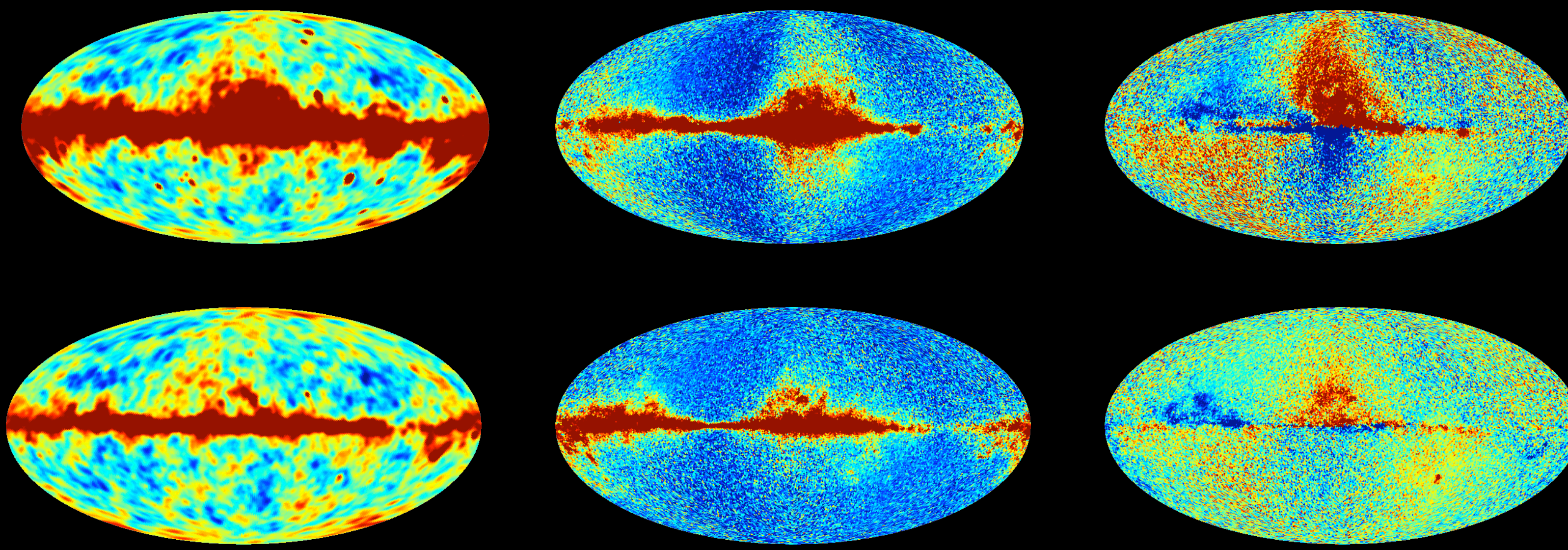


Simulated sky maps

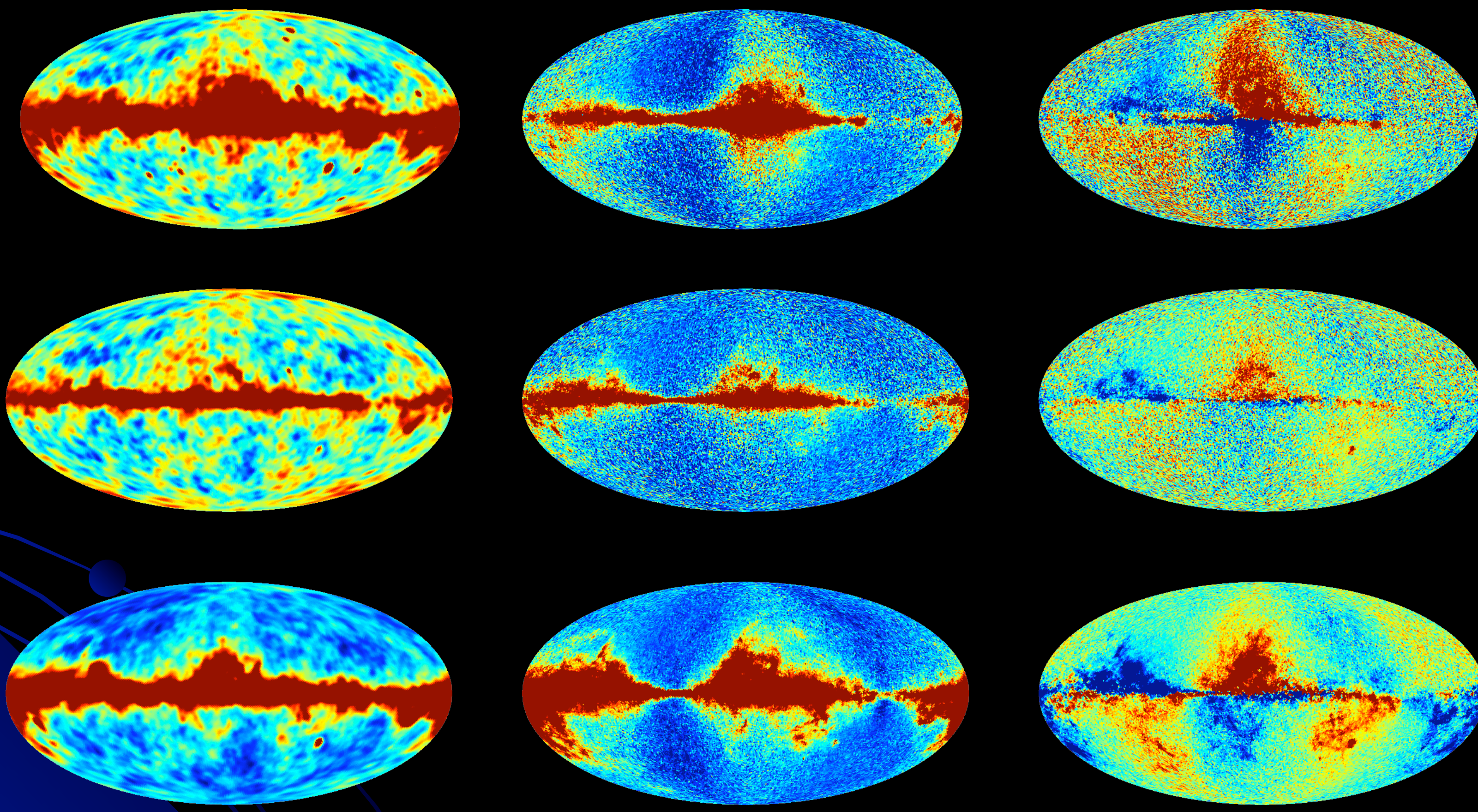
Stokes' T Stokes' Q Stokes' U

30 GHz

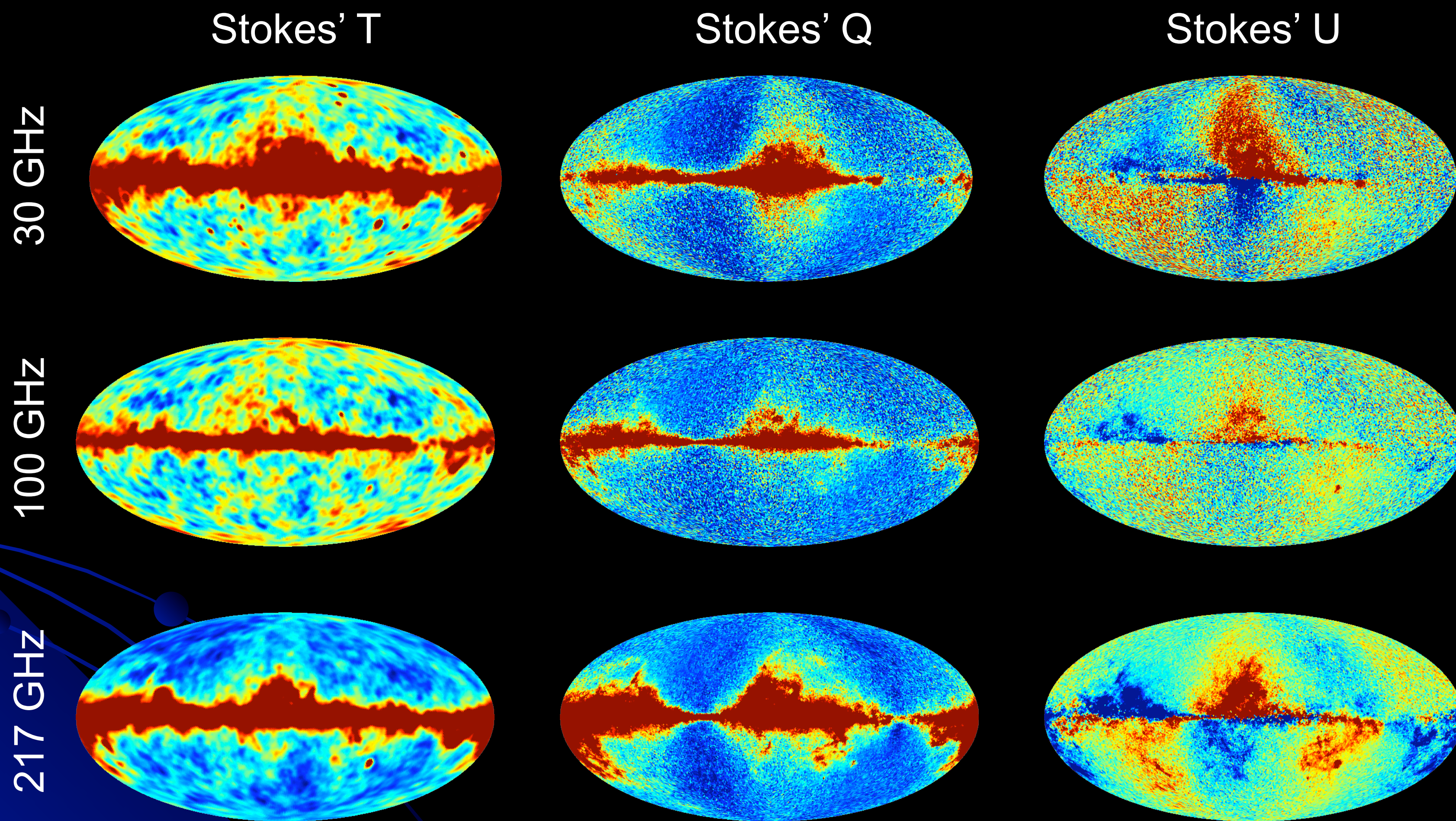
100 GHz



Simulated sky maps

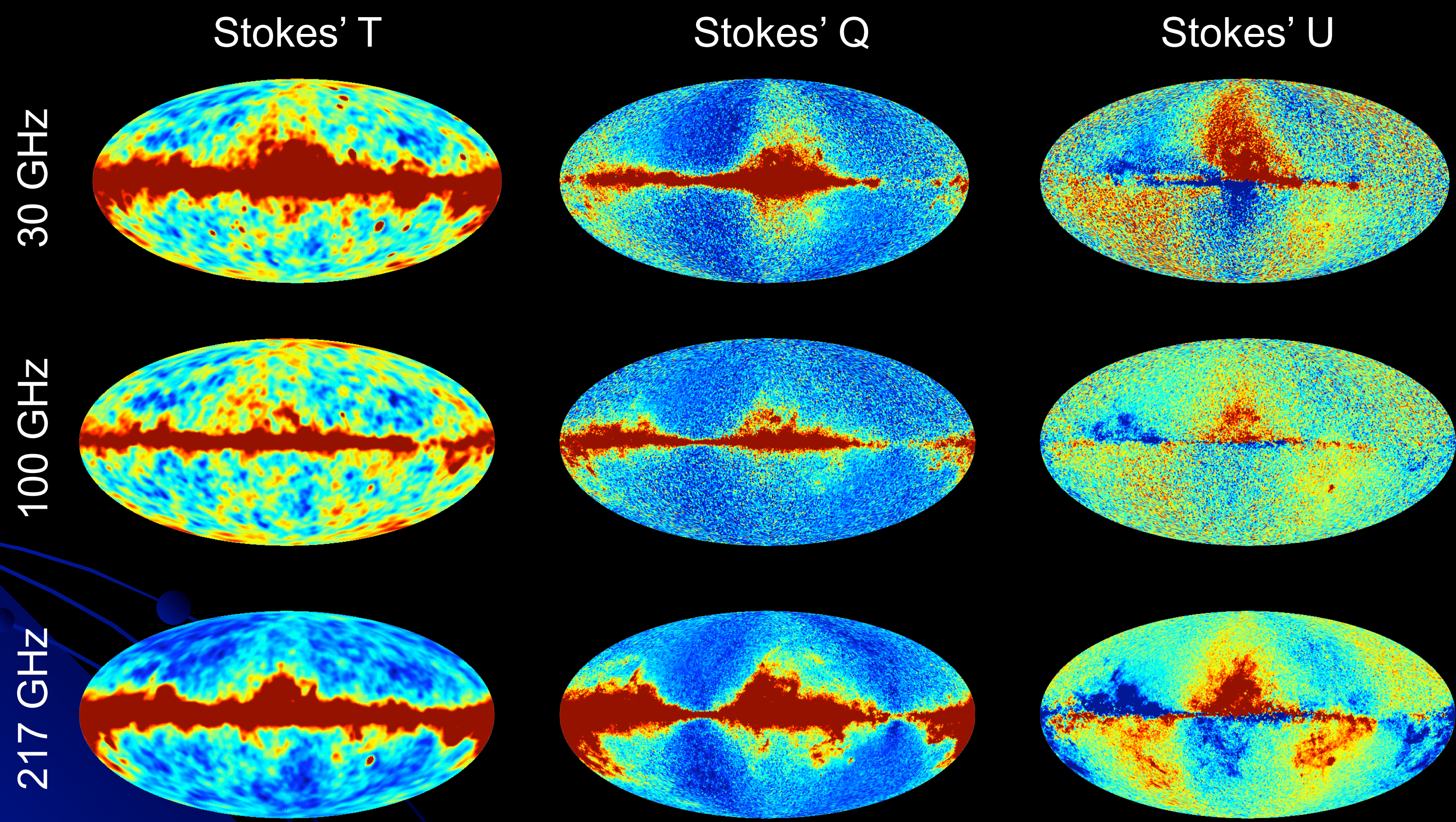


Simulated sky maps



- All maps downgraded to Nside = 64 (55' pixels) to enhance S/N

Simulated sky maps



- All maps downgraded to Nside = 64 (55' pixels) to enhance S/N
- Temperature maps additionally smoothed to 3° FWHM by post-processing, to ensure proper bandwidth limitation

The data model

The data model

- The parametric model adopted for this analysis was the following:

$$S_\nu(p) = \mathbf{s}(p) + m_\nu + \sum_{i=1}^{3} d_i(\hat{p} \cdot \mathbf{e}_i) +$$
$$+ A_s(p)a(\nu)\left(\frac{\nu}{\nu_0}\right)^{\beta_s(p)} + A_d(p)a(\nu)\left(\frac{\nu}{\nu_0}\right)^{\beta_d(p)}$$

The data model

- The parametric model adopted for this analysis was the following:

CMB

$$S_\nu(p) = \mathbf{s}(p) + m_\nu + \sum_{i=1}^{3} d_i(\hat{p}\cdot\mathbf{e}_i) +$$

$$+ A_s(p)a(\nu)\left(\frac{\nu}{\nu_0}\right)^{\beta_s(p)} + A_d(p)a(\nu)\left(\frac{\nu}{\nu_0}\right)^{\beta_d(p)}$$

The data model

- The parametric model adopted for this analysis was the following:

CMB Monopoles*

$$S_\nu(p) = \mathbf{s}(p) + m_\nu + \sum_{i=1}^{3} d_i(\hat{p} \cdot \mathbf{e}_i) +$$

$$+ A_s(p) a(\nu) \left(\frac{\nu}{\nu_0}\right)^{\beta_s(p)} + A_d(p) a(\nu) \left(\frac{\nu}{\nu_0}\right)^{\beta_d(p)}$$

* Only temperature, not polarization

The data model

- The parametric model adopted for this analysis was the following:

CMB Monopoles* Dipoles*

$$S_\nu(p) = \mathbf{s}(p) + m_\nu + \sum_{i=1}^{3} d_i(\hat{p}\cdot\mathbf{e}_i) + A_s(p)a(\nu)\left(\frac{\nu}{\nu_0}\right)^{\beta_s(p)} + A_d(p)a(\nu)\left(\frac{\nu}{\nu_0}\right)^{\beta_d(p)}$$

* Only temperature, not polarization

The data model

- The parametric model adopted for this analysis was the following:

CMB

Monopoles*

Dipoles*

$$S_\nu(p) = \mathbf{s}(p) + m_\nu + \sum_{i=1}^{3} d_i(\hat{p}\cdot\mathbf{e}_i) +$$

$$+ A_s(p)a(\nu)\left(\frac{\nu}{\nu_0}\right)^{\beta_s(p)} + A_d(p)a(\nu)\left(\frac{\nu}{\nu_0}\right)^{\beta_d(p)}$$

Low-frequency foreground

* Only temperature, not polarization

The data model

- The parametric model adopted for this analysis was the following:

CMB Monopoles* Dipoles*

$$S_\nu(p) = \mathbf{s}(p) + m_\nu + \sum_{i=1}^{3} d_i(\hat{p}\cdot\mathbf{e}_i) +$$

$$+ A_s(p)a(\nu)\left(\frac{\nu}{\nu_0}\right)^{\beta_s(p)} + A_d(p)a(\nu)\left(\frac{\nu}{\nu_0}\right)^{\beta_d(p)}$$

Low-frequency foreground

Thermal dust

* Only temperature, not polarization

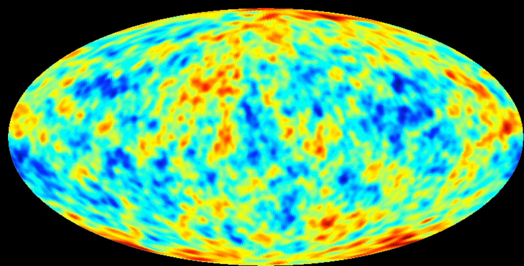
Reconstructed CMB signal maps

Reconstructed CMB signal maps

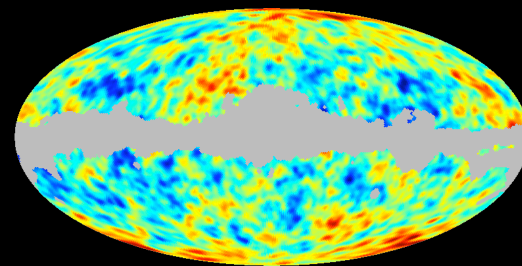
Input signal

Estimated signal

Stokes' T



Scale: -175 μK to 175μK

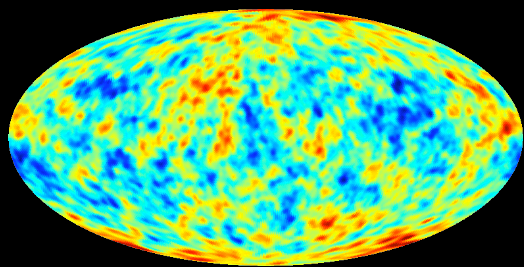


Scale: -175 μK to 175μK

Reconstructed CMB signal maps

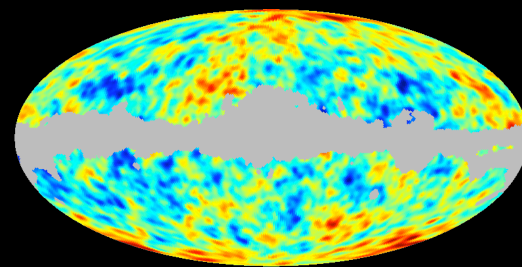
Stokes' T

Input signal



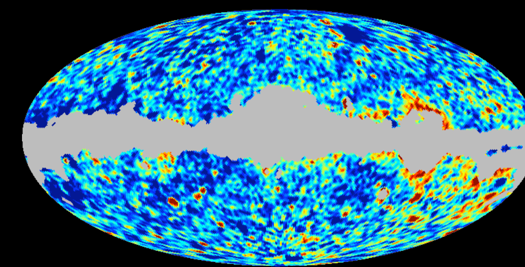
Scale: -175 μK to 175μK

Estimated signal



Scale: -175 μK to 175μK

$\frac{\text{Estimate} - \text{Input}}{\text{Estimated error}}$

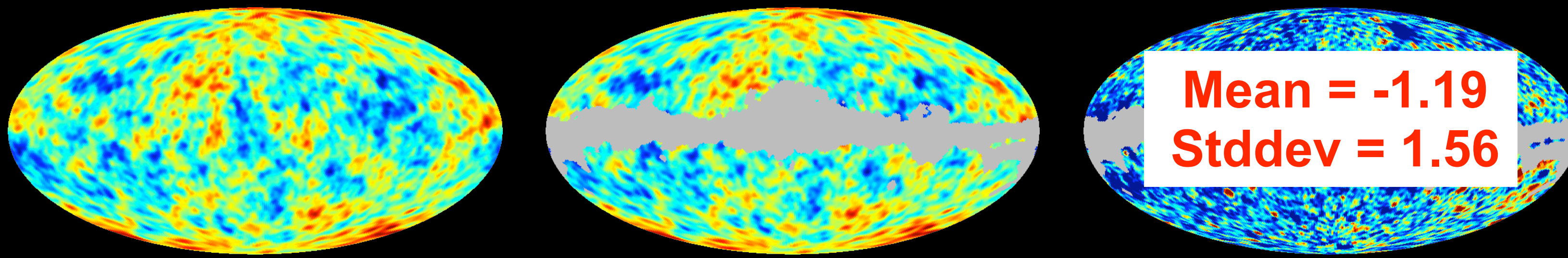


Scale: -3σ to 3σ

Reconstructed CMB signal maps

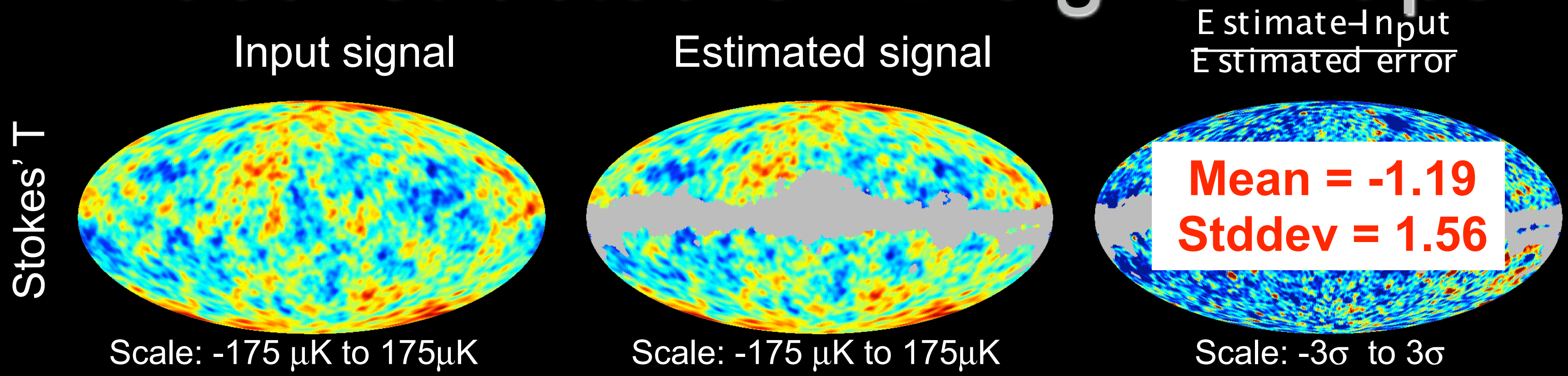
Input signal | Estimated signal | $\frac{\text{Estimate} - \text{Input}}{\text{Estimated error}}$

Stokes' T



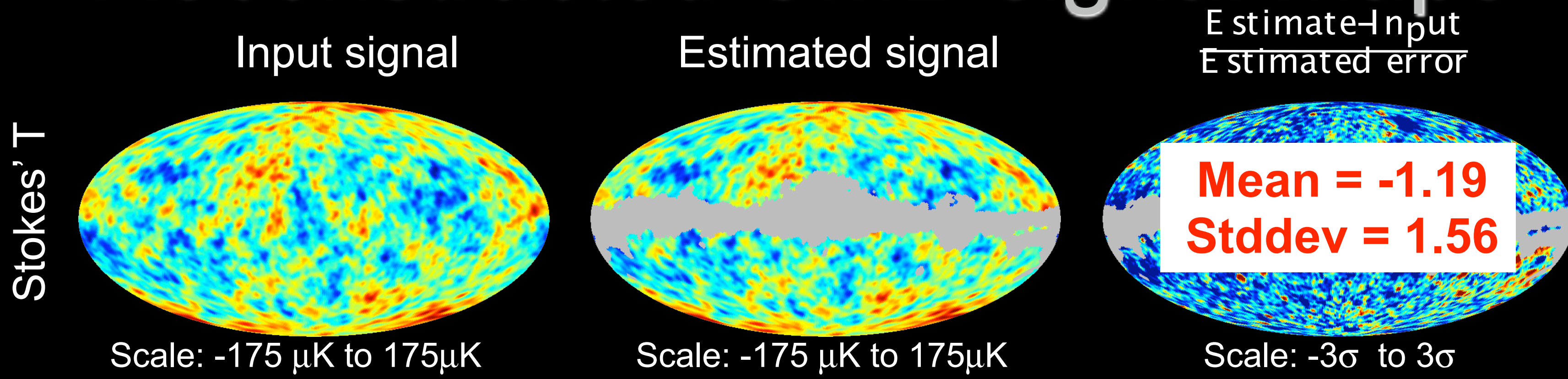
Scale: -175 μK to 175μK | Scale: -175 μK to 175μK | Scale: -3σ to 3σ

Reconstructed CMB signal maps



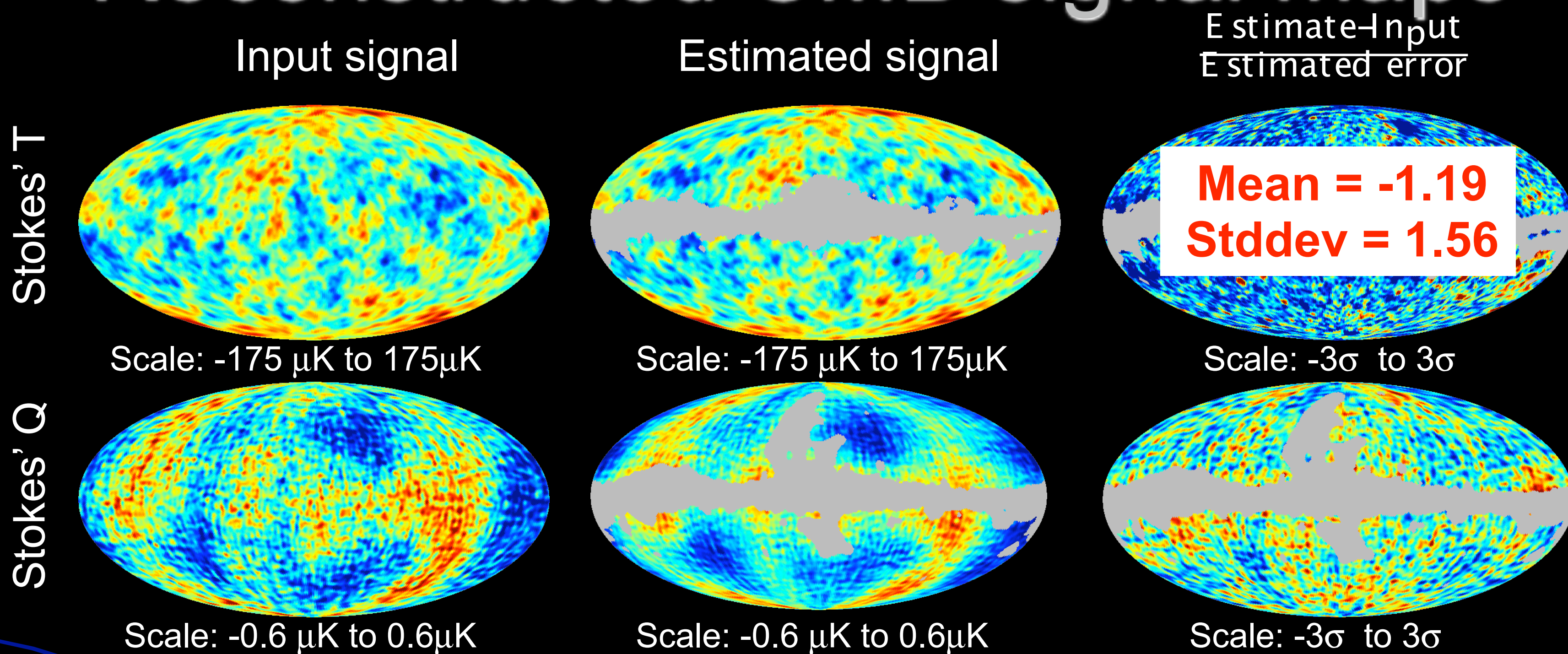
- Temperature is statistically "difficult" because of (1) multiple low-frequency components and (2) very high S/N

Reconstructed CMB signal maps



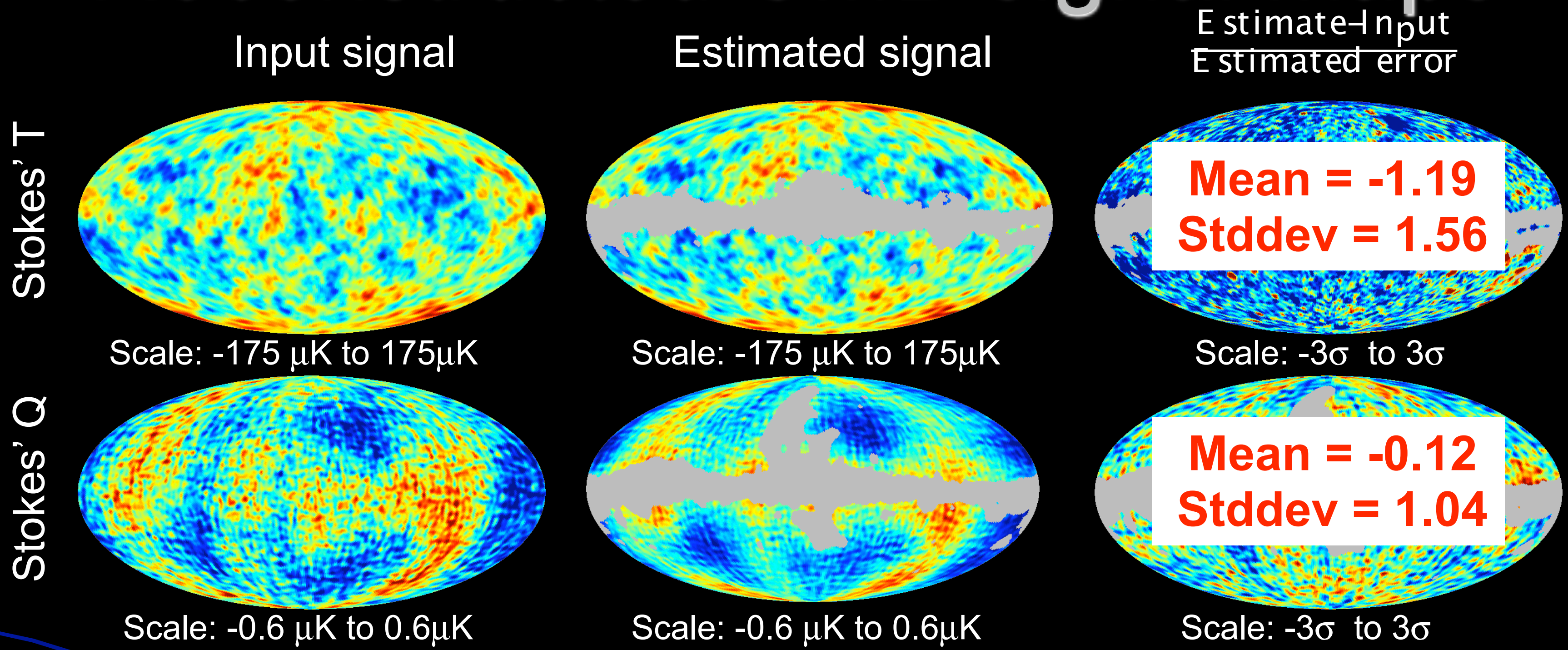
- Temperature is statistically "difficult" because of (1) multiple low-frequency components and (2) very high S/N
- However, these small (~5-10 μK) temperature errors don't matter to the overall large-scale likelihood because of tremendous cosmic variance

Reconstructed CMB signal maps



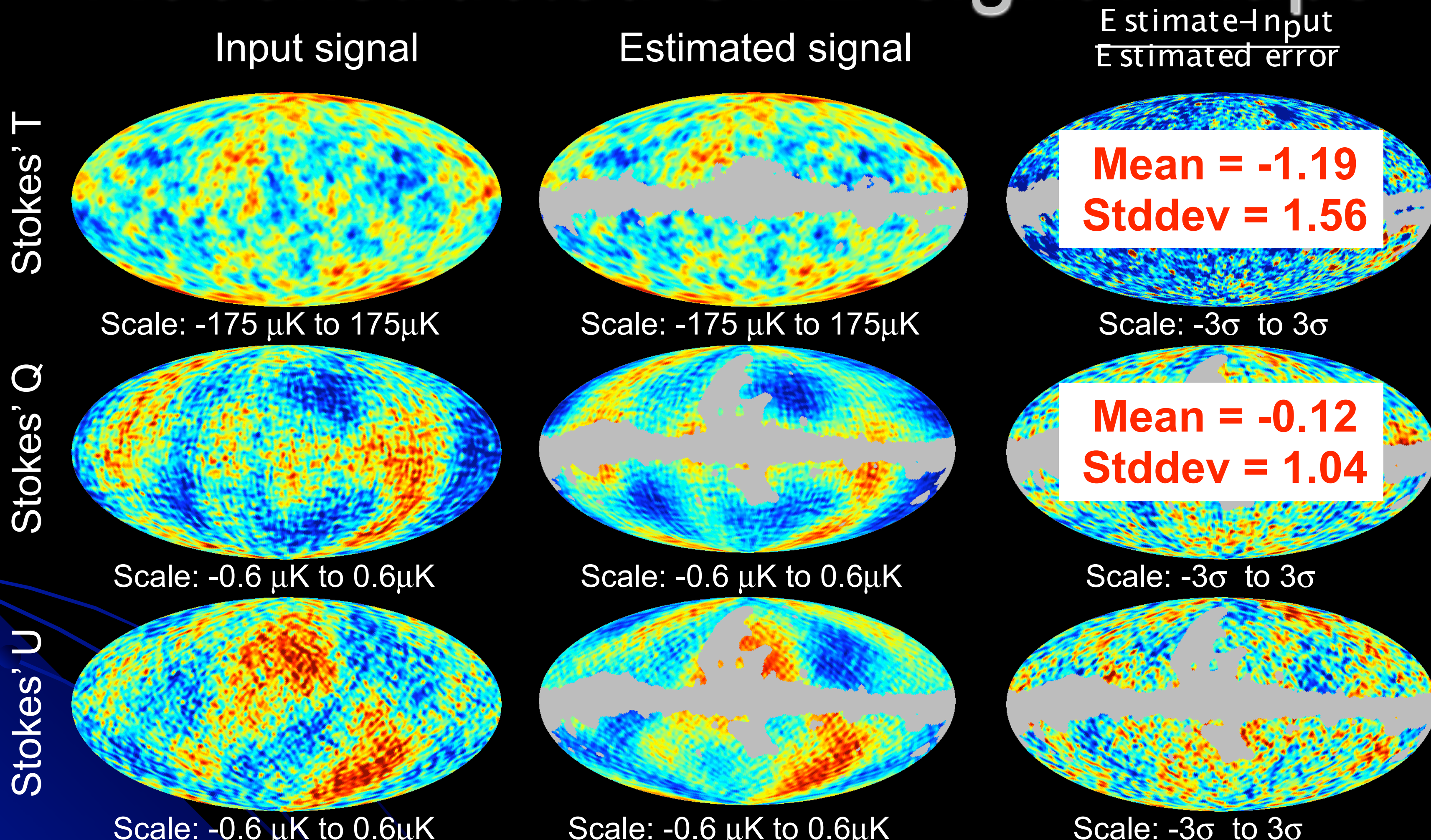
- Temperature is statistically "difficult" because of (1) multiple low-frequency components and (2) very high S/N
- However, these small (~5-10 μK) temperature errors don't matter to the overall large-scale likelihood because of tremendous cosmic variance

Reconstructed CMB signal maps



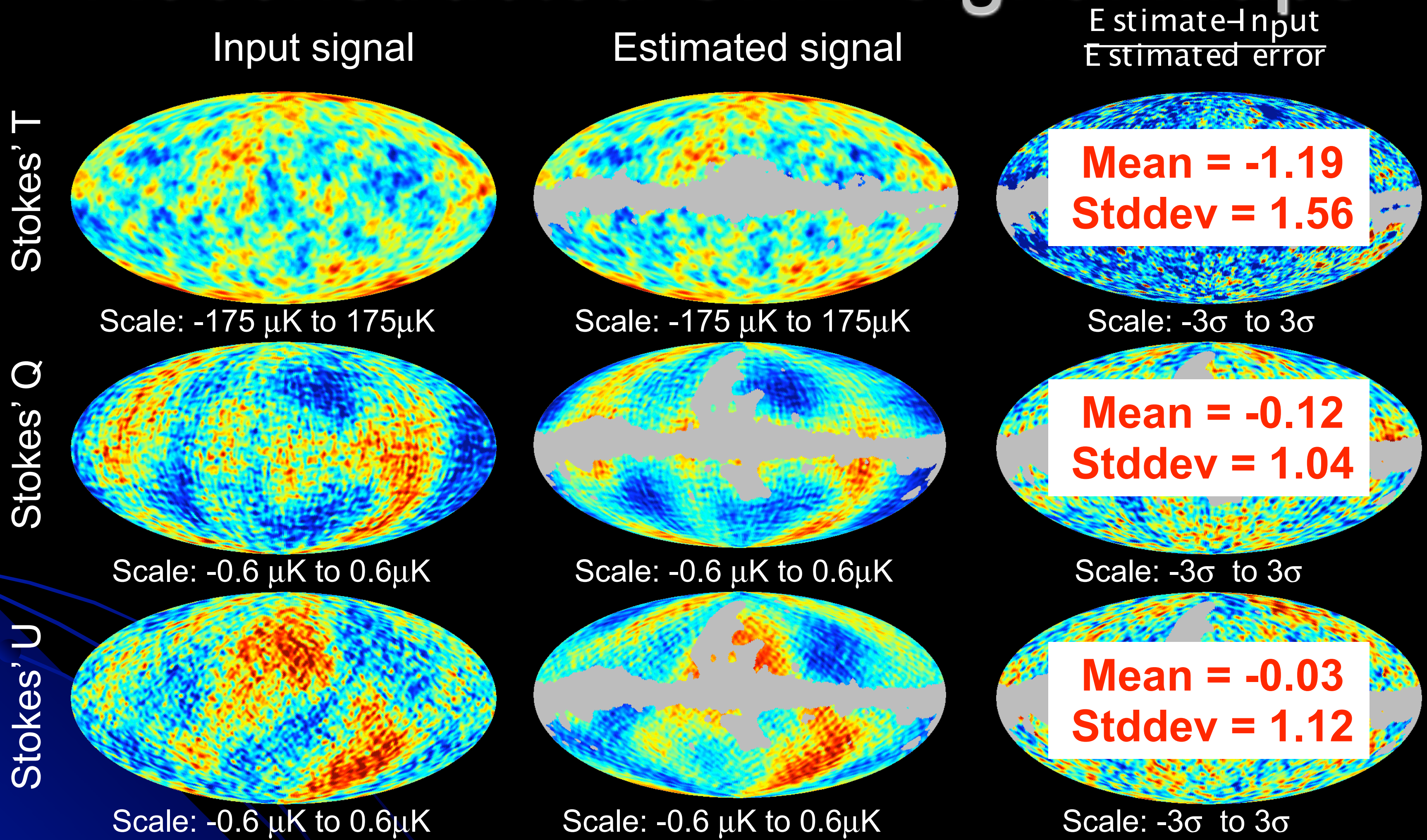
- Temperature is statistically "difficult" because of (1) multiple low-frequency components and (2) very high S/N
- However, these small (~5-10 μK) temperature errors don't matter to the overall large-scale likelihood because of tremendous cosmic variance

Reconstructed CMB signal maps



- Temperature is statistically "difficult" because of (1) multiple low-frequency components and (2) very high S/N
- However, these small (~5-10 μK) temperature errors don't matter to the overall large-scale likelihood because of tremendous cosmic variance

Reconstructed CMB signal maps

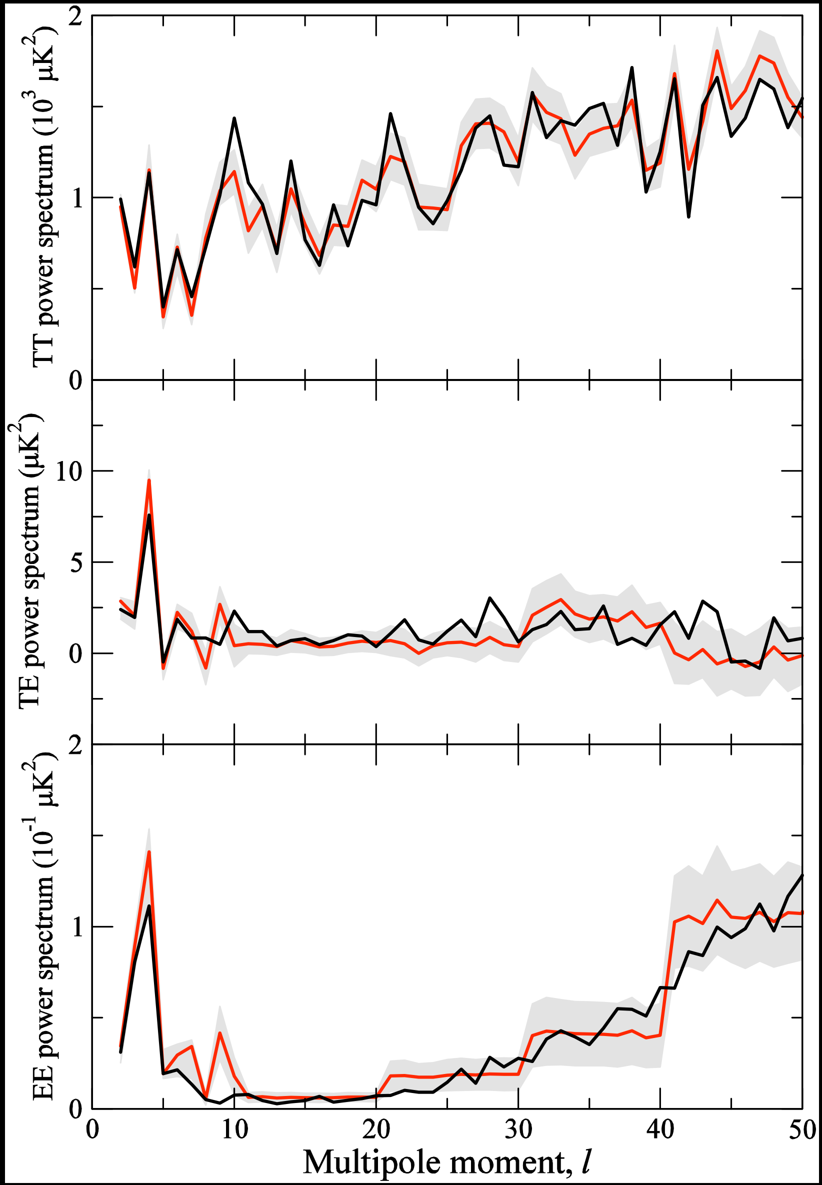


- Temperature is statistically "difficult" because of (1) multiple low-frequency components and (2) very high S/N
- However, these small (~5-10 μK) temperature errors don't matter to the overall large-scale likelihood because of tremendous cosmic variance

Reconstructed power spectrum

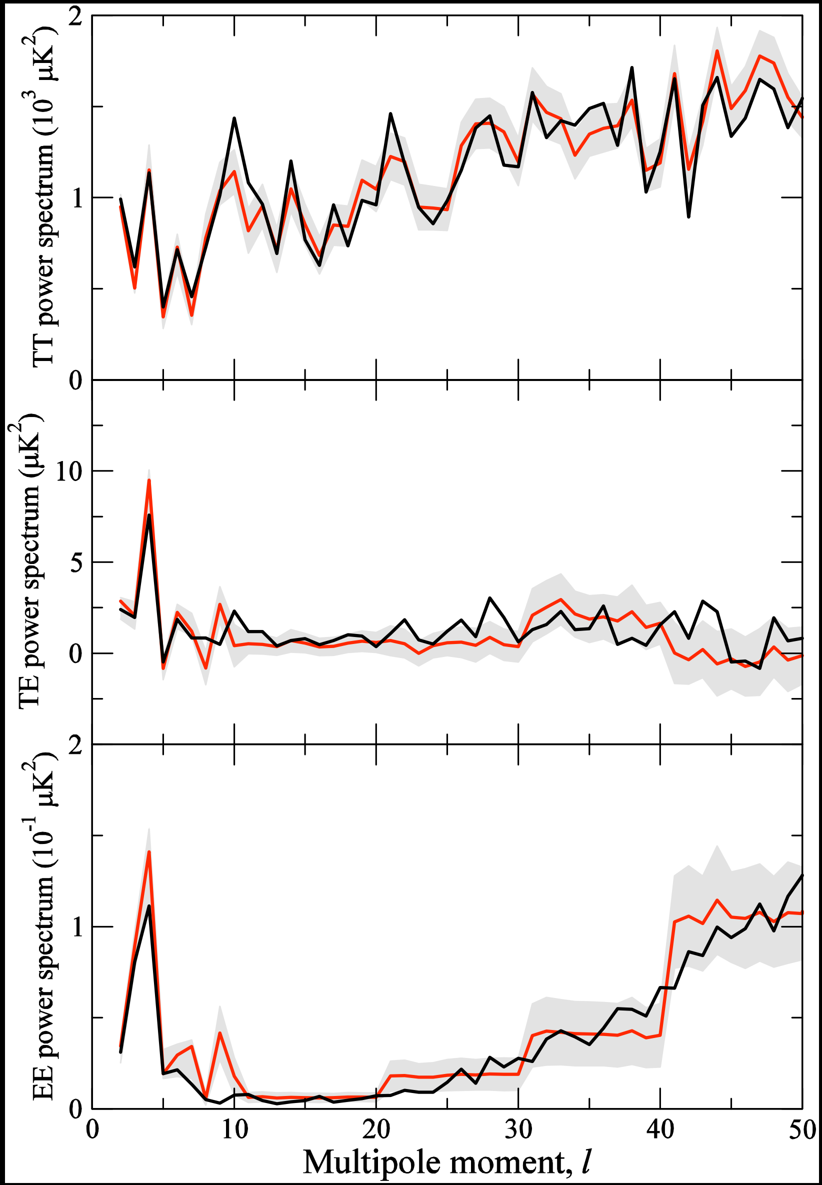
Reconstructed power spectrum

- Legend:



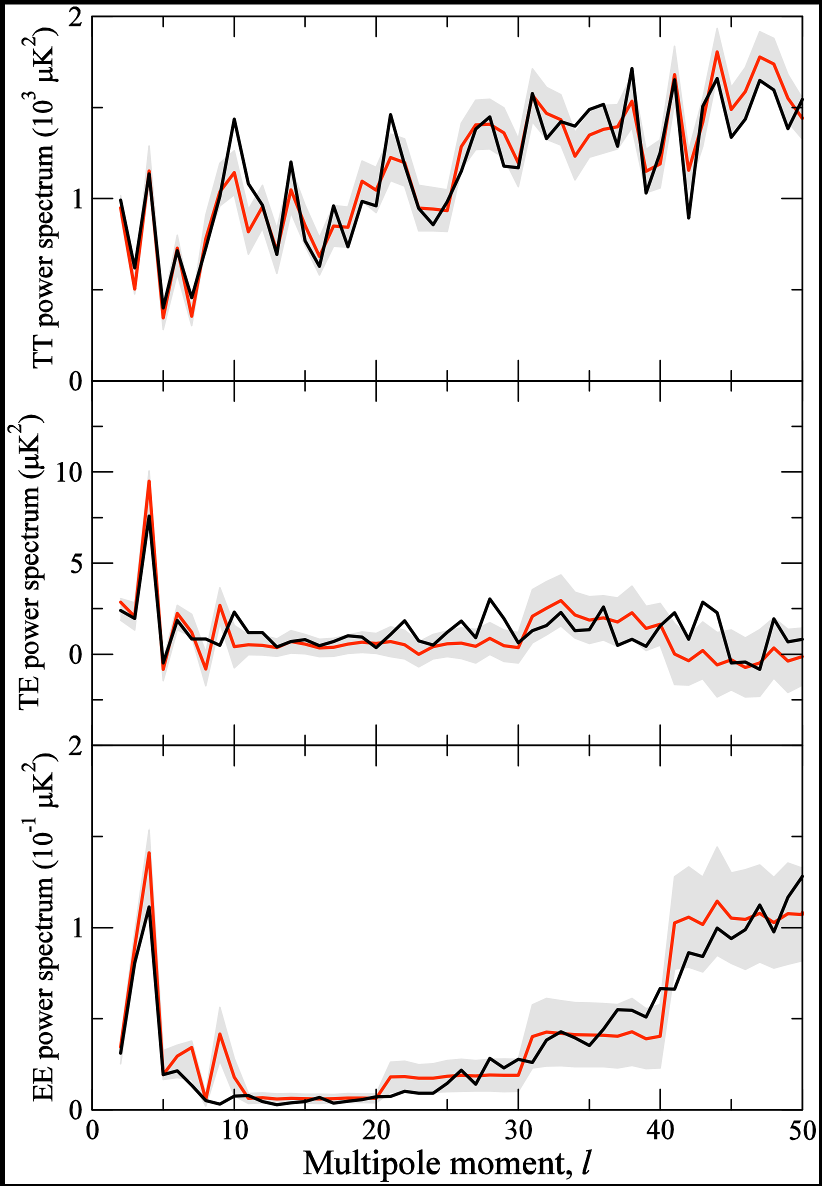
Reconstructed power spectrum

- Legend:
  - Black curve: True ("unknown") input full-sky spectrum



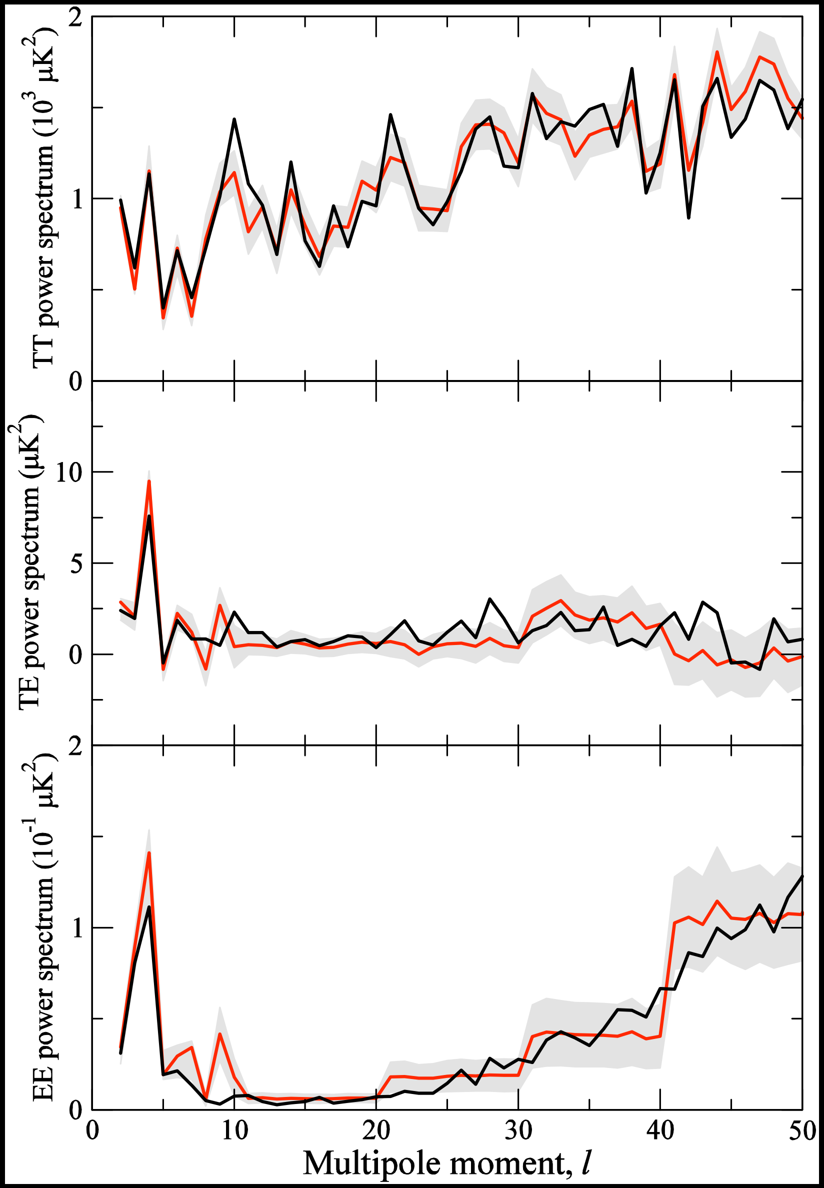
Reconstructed power spectrum

- Legend:
  - Black curve: True ("unknown") input full-sky spectrum
  - Red curve: Reconstructed spectrum



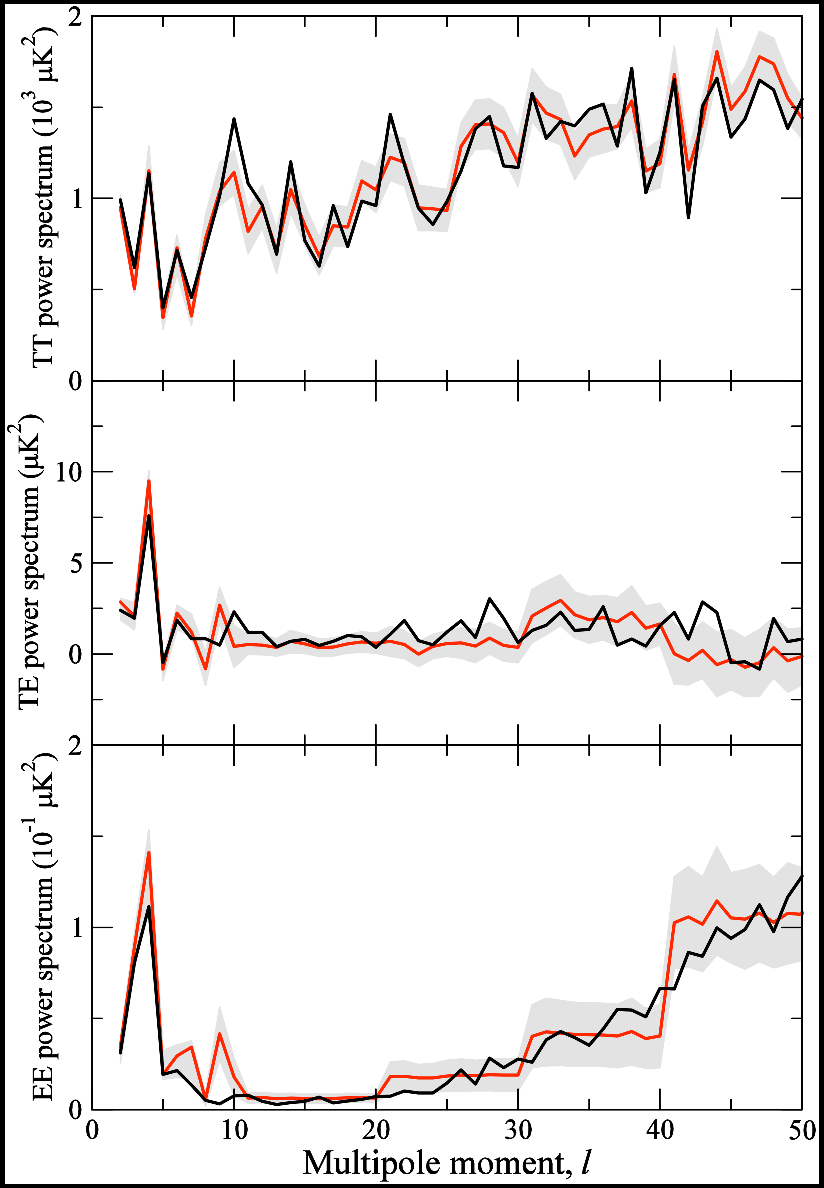
Reconstructed power spectrum

- Legend:
  - Black curve: True ("unknown") input full-sky spectrum
  - Red curve: Reconstructed spectrum
  - Gray region: 68% confi-dence region, excluding cosmic variance



Reconstructed power spectrum

- Legend:
  - Black curve: True ("unknown") input full-sky spectrum
  - Red curve: Reconstructed spectrum
  - Gray region: 68% confi-dence region, excluding cosmic variance
- No striking biases, and error bars are reasonable



CMBPol simulations

CMBPol simulations

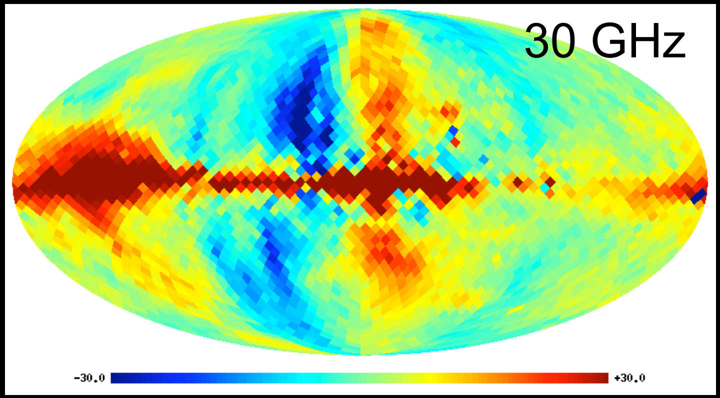
- Simulations made by Jo Dunkley:

CMBPol simulations

- Simulations made by Jo Dunkley:
  - 7 channels between 30 and 300 GHz (30, 40, 60, 90, 135, 200 and 300 GHz)

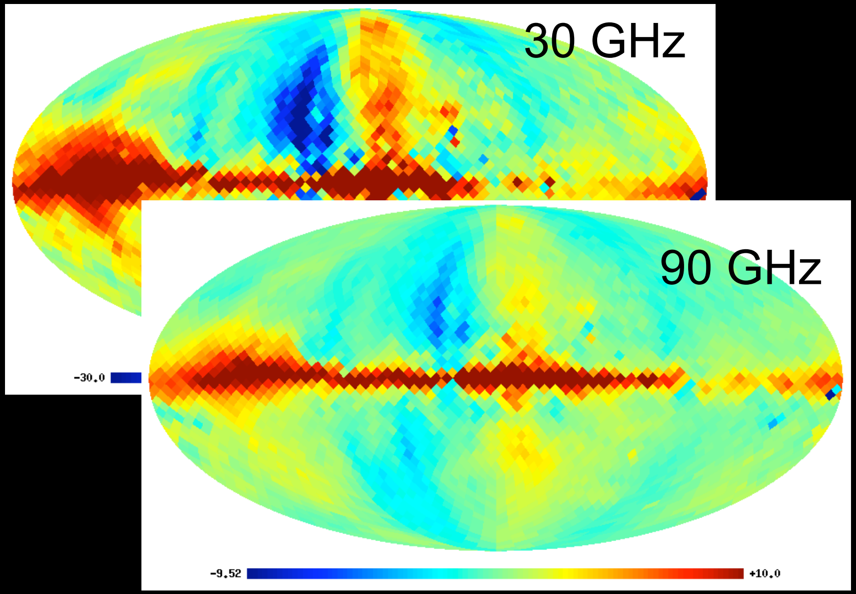
CMBPol simulations

- Simulations made by Jo Dunkley:
  - 7 channels between 30 and 300 GHz (30, 40, 60, 90, 135, 200 and 300 GHz)



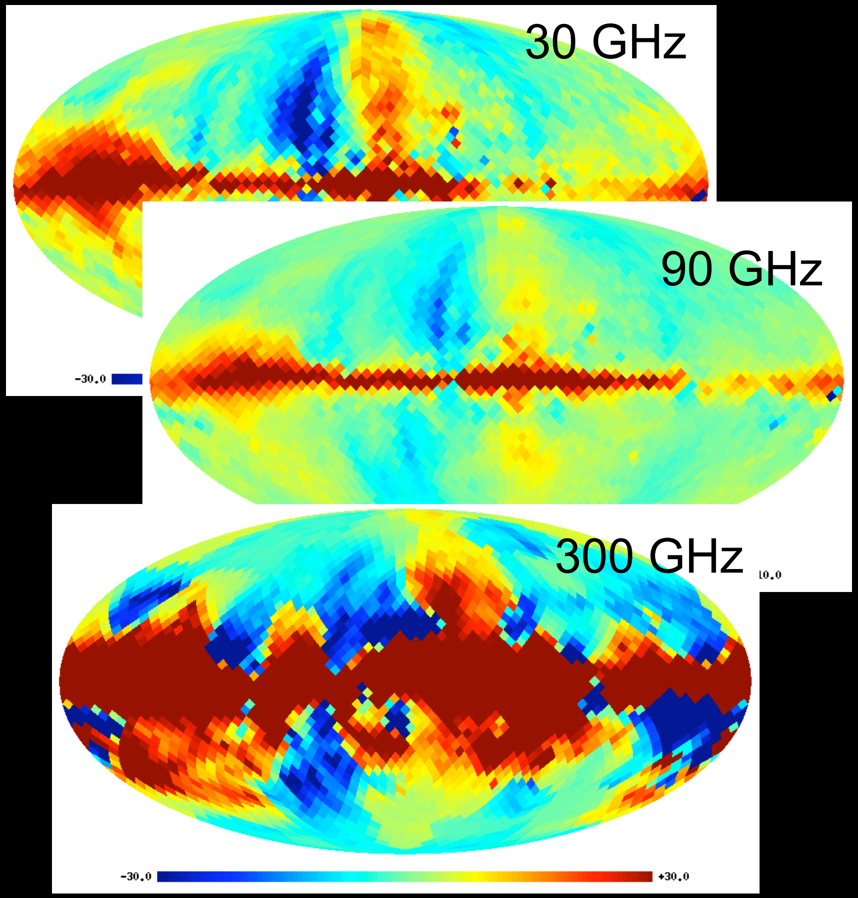
CMBPol simulations

- Simulations made by Jo Dunkley:
  - 7 channels between 30 and 300 GHz (30, 40, 60, 90, 135, 200 and 300 GHz)



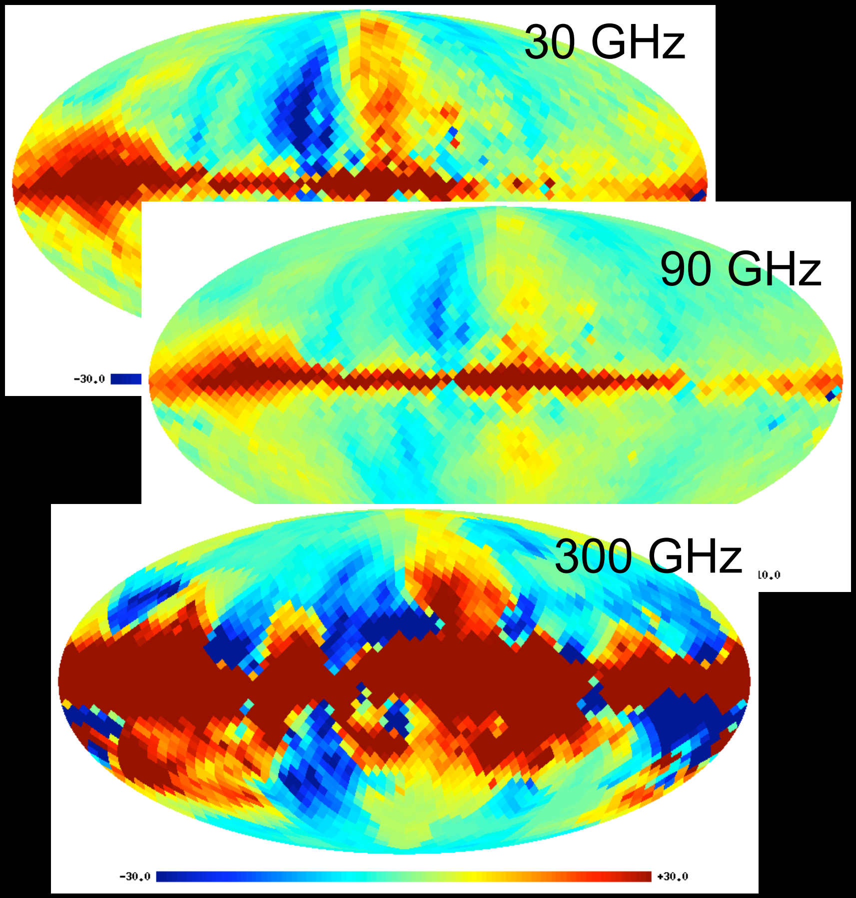
CMBPol simulations

- Simulations made by Jo Dunkley:
  - 7 channels between 30 and 300 GHz (30, 40, 60, 90, 135, 200 and 300 GHz)



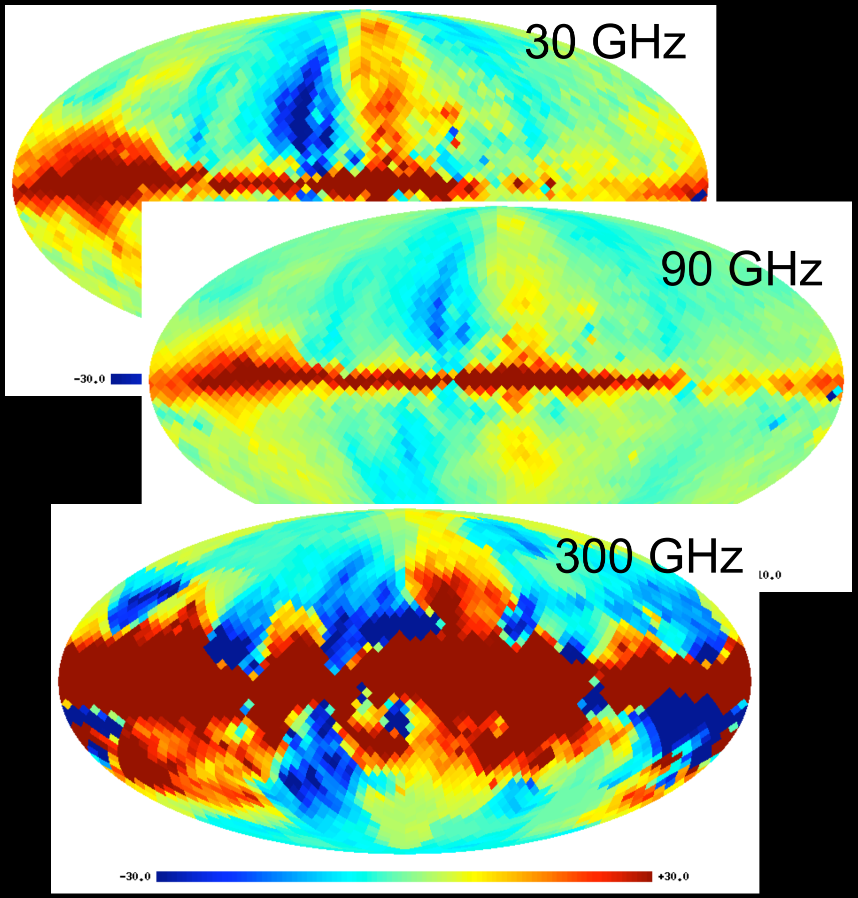
CMBPol simulations

- Simulations made by Jo Dunkley:
  - 7 channels between 30 and 300 GHz (30, 40, 60, 90, 135, 200 and 300 GHz)
  - CMB + power law synchrotron (beta = -3.0) + power law dust (beta=1.7) + noise



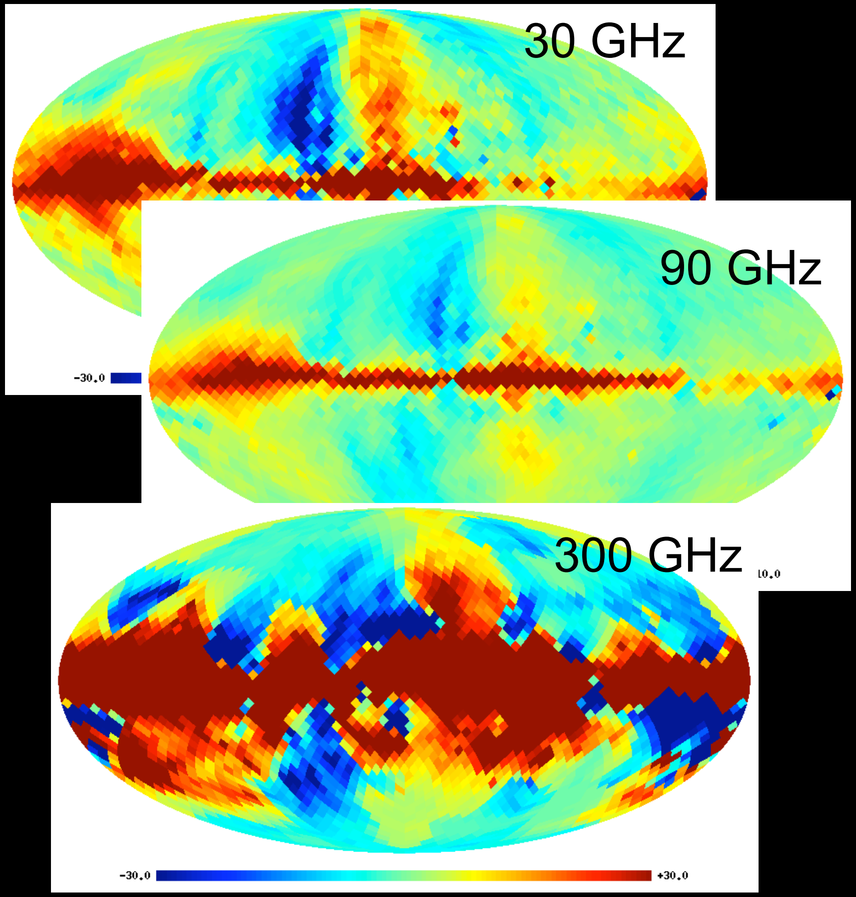
CMBPol simulations

- Simulations made by Jo Dunkley:
  - 7 channels between 30 and 300 GHz (30, 40, 60, 90, 135, 200 and 300 GHz)
  - CMB + power law synchrotron (beta = -3.0) + power law dust (beta=1.7) + noise
  - Two sets of noise levels, corresponding to two versions of EPIC LC (LC1 and LC2)



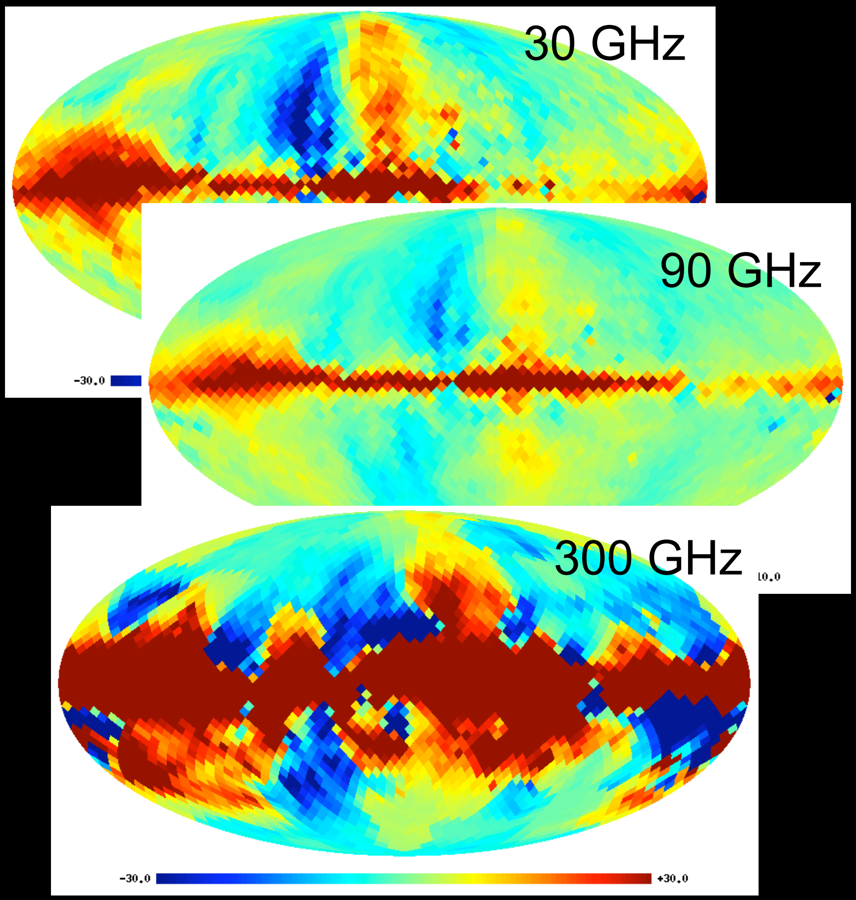
CMBPol simulations

- Simulations made by Jo Dunkley:
  - 7 channels between 30 and 300 GHz (30, 40, 60, 90, 135, 200 and 300 GHz)
  - CMB + power law synchrotron (beta = -3.0) + power law dust (beta=1.7) + noise
  - Two sets of noise levels, corresponding to two versions of EPIC LC (LC1 and LC2)



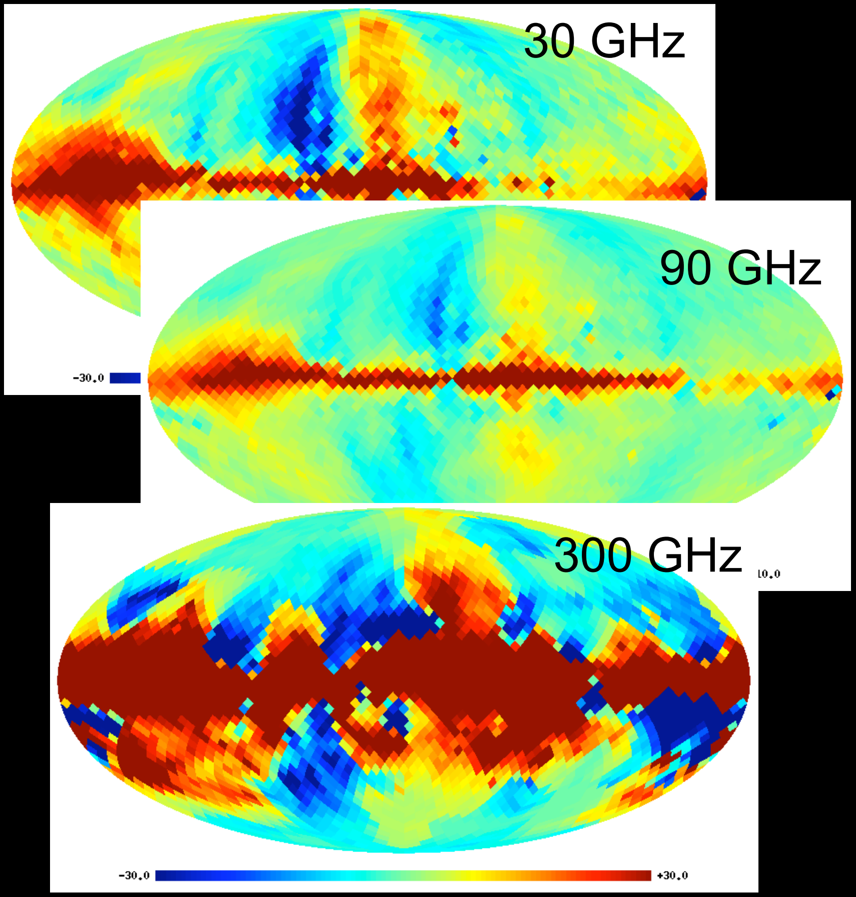
CMBPol simulations

- Simulations made by Jo Dunkley:
  - 7 channels between 30 and 300 GHz (30, 40, 60, 90, 135, 200 and 300 GHz)
  - CMB + power law synchrotron (beta = -3.0) + power law dust (beta=1.7) + noise
  - Two sets of noise levels, corresponding to two versions of EPIC LC (LC1 and LC2)
- 5-year WMAP polarization mask



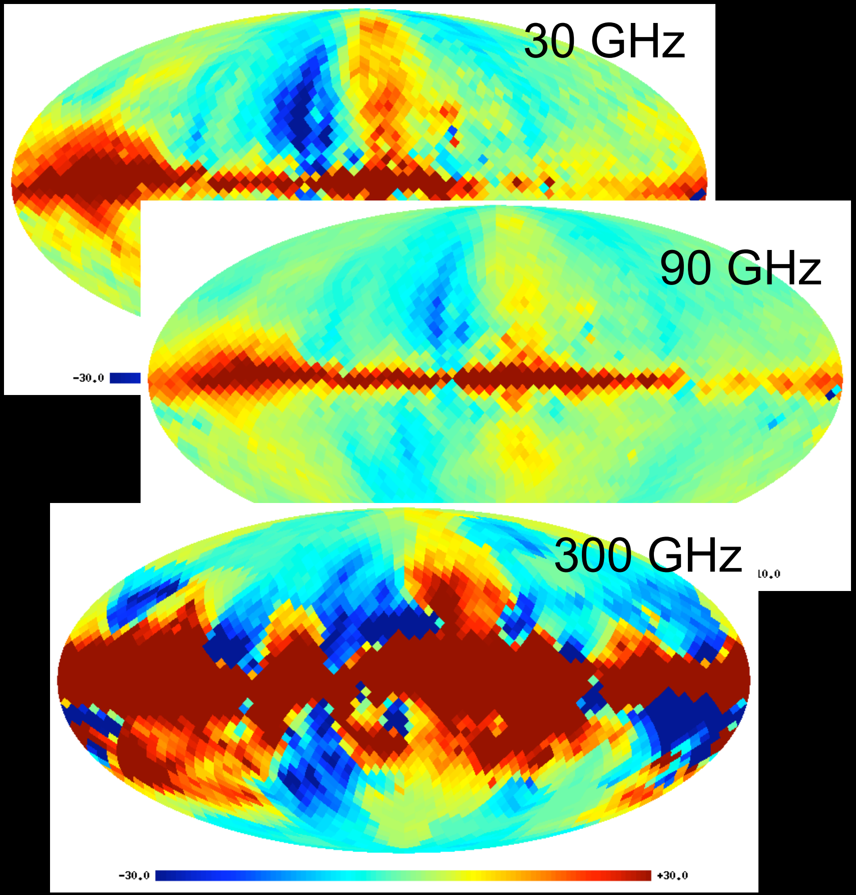
CMBPol simulations

- Simulations made by Jo Dunkley:
  - 7 channels between 30 and 300 GHz (30, 40, 60, 90, 135, 200 and 300 GHz)
  - CMB + power law synchrotron (beta = -3.0) + power law dust (beta=1.7) + noise
  - Two sets of noise levels, corresponding to two versions of EPIC LC (LC1 and LC2)
- 5-year WMAP polarization mask



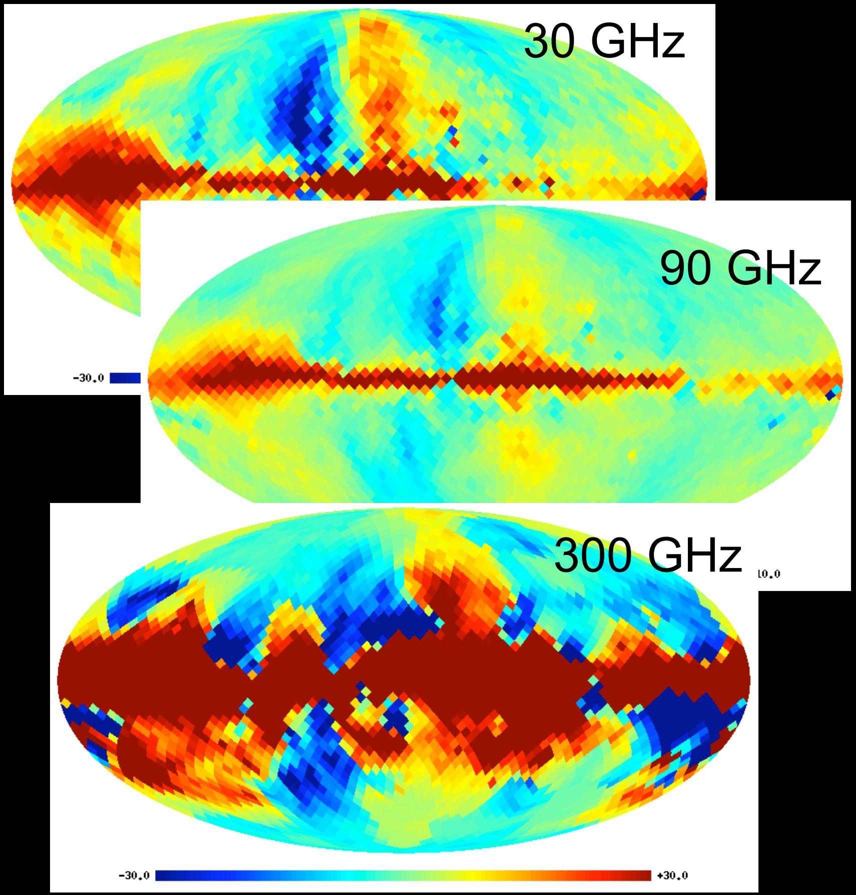
CMBPol simulations

- Simulations made by Jo Dunkley:
  - 7 channels between 30 and 300 GHz (30, 40, 60, 90, 135, 200 and 300 GHz)
  - CMB + power law synchrotron (beta = -3.0) + power law dust (beta=1.7) + noise
  - Two sets of noise levels, corresponding to two versions of EPIC LC (LC1 and LC2)
- 5-year WMAP polarization mask
- Analysis at Nside=16, lmax=47, FWHM = 7 degrees, to speed up things during test phase



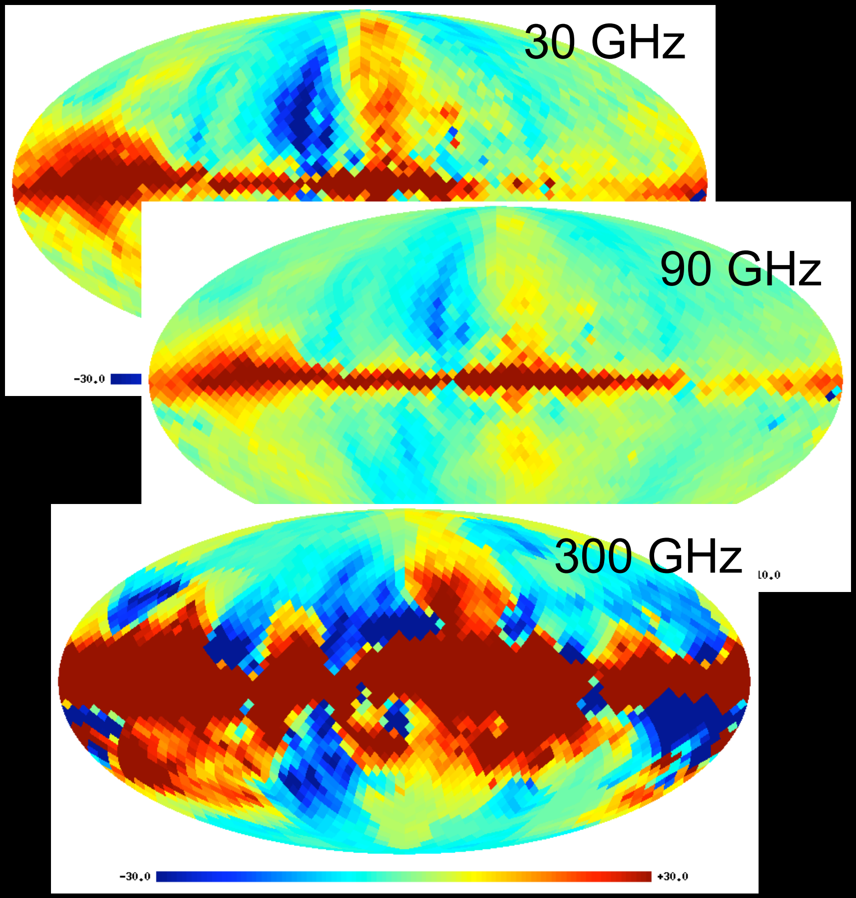
CMBPol simulations

- Simulations made by Jo Dunkley:
  - 7 channels between 30 and 300 GHz (30, 40, 60, 90, 135, 200 and 300 GHz)
  - CMB + power law synchrotron (beta = -3.0) + power law dust (beta=1.7) + noise
  - Two sets of noise levels, corresponding to two versions of EPIC LC (LC1 and LC2)
- 5-year WMAP polarization mask
- Analysis at Nside=16, lmax=47, FWHM = 7 degrees, to speed up things during test phase



CMBPol simulations

- Simulations made by Jo Dunkley:
  - 7 channels between 30 and 300 GHz (30, 40, 60, 90, 135, 200 and 300 GHz)
  - CMB + power law synchrotron (beta = -3.0) + power law dust (beta=1.7) + noise
  - Two sets of noise levels, corresponding to two versions of EPIC LC (LC1 and LC2)
- 5-year WMAP polarization mask
- Analysis at Nside=16, lmax=47, FWHM = 7 degrees, to speed up things during test phase
- (Note: These results are a first test, hastily computed last week, and intended only to demonstrate the machinery)



Foreground model and free parameters

Foreground model and free parameters

- Simplest possible model adopted:

$$S_o(p) = s(p) + A_s\, g(o)\, 3 \left(\frac{o}{30\,\mathrm{GHz}}\right)^{i\,3} + A_d\, g(o)\, 3 \left(\frac{o}{300\,\mathrm{GHz}}\right)^{1:7}$$

Foreground model and free parameters

- Simplest possible model adopted:

$$S_\nu(p) = s(p) + A_s\, g(\nu) \left(\frac{\nu}{30\mathrm{GHz}}\right)^{-3} + A_d\, g(\nu) \left(\frac{\nu}{300\mathrm{GHz}}\right)^{1.7}$$

Foreground model and free parameters

- Simplest possible model adopted:

$$S_\nu(p) = s(p) + A_s\, g(\nu) \left(\frac{\nu}{30\mathrm{GHz}}\right)^{-3} + A_d\, g(\nu) \left(\frac{\nu}{300\mathrm{GHz}}\right)^{1.7}$$

$$\langle s_{\ell m} s^t_{\ell m} \rangle = \begin{pmatrix} 0 & 0 & 0 \\ 0 & aC_\ell^{EE,in} & 0 \\ 0 & 0 & bC_\ell^{BB,in} \end{pmatrix} \delta_{\ell\ell}\delta_{mm}$$

Foreground model and free parameters

- Simplest possible model adopted:

$$S_\nu(p) = s(p) + A_s\, g(\nu) \left(\frac{\nu}{30\mathrm{GHz}}\right)^{-3} + A_d\, g(\nu) \left(\frac{\nu}{300\mathrm{GHz}}\right)^{1.7}$$

  - Only marginalization over foreground amplitudes, not spectral indices

$$\langle s_{\ell m} s^t_{\ell m} \rangle = \begin{pmatrix} 0 & 0 & 0 \\ 0 & aC_\ell^{EE,in} & 0 \\ 0 & 0 & bC_\ell^{BB,in} \end{pmatrix} \delta_{\ell} \delta_{mm}$$

$$r_2 = \frac{C_2^{BB}}{C_2^{EE}}$$

Foreground model and free parameters

- Simplest possible model adopted:

$$S_\nu(p) = s(p) + A_s\, g(\nu) \left(\frac{\nu}{30\,\mathrm{GHz}}\right)^{-3} + A_d\, g(\nu) \left(\frac{\nu}{300\,\mathrm{GHz}}\right)^{1.7}$$

  - Only marginalization over foreground amplitudes, not spectral indices

$$\langle s_{\ell m} s^t_{\ell m} \rangle = \begin{pmatrix} 0 & 0 & 0 \\ 0 & aC_\ell^{EE,in} & 0 \\ 0 & 0 & bC_\ell^{BB,in} \end{pmatrix} \delta_{\ell\ell'} \delta_{mm'}$$

$$r_2 = \frac{C_2^{BB}}{C_2^{EE}}$$

Foreground model and free parameters

- Simplest possible model adopted:

$$S_\nu(p) = s(p) + A_s\, g(\nu) \left(\frac{\nu}{30\mathrm{GHz}}\right)^{-3} + A_d\, g(\nu) \left(\frac{\nu}{300\mathrm{GHz}}\right)^{1.7}$$

  - Only marginalization over foreground amplitudes, not spectral indices

$$\langle s_{\ell m} s^{t}_{\ell m} \rangle = \begin{pmatrix} 0 & 0 & 0 \\ 0 & aC_\ell^{EE,in} & 0 \\ 0 & 0 & bC_\ell^{BB,in} \end{pmatrix} \delta_{\ell \ell'} \delta_{mm'}$$

$$r_\ell = \frac{C_\ell^{BB}}{C_\ell^{EE}}$$

Foreground model and free parameters

- Simplest possible model adopted:

$$S_\nu(p) = s(p) + A_s\, g(\nu) \left(\frac{\nu}{30\mathrm{GHz}}\right)^{-3} + A_d\, g(\nu) \left(\frac{\nu}{300\mathrm{GHz}}\right)^{1.7}$$

  - Only marginalization over foreground amplitudes, not spectral indices

- CMB covariance parameterized by two overall scaling amplitudes, *a* and *b*, only:

$$\langle s_{\ell m} s^t_{\ell m} \rangle = \begin{pmatrix} 0 & 0 & 0 \\ 0 & aC_\ell^{EE,\mathrm{in}} & 0 \\ 0 & 0 & bC_\ell^{BB,\mathrm{in}} \end{pmatrix} \delta_{\ell\ell'}\delta_{mm'}$$

$$r_2 = \frac{C_2^{BB}}{C_2^{EE}}$$

Foreground model and free parameters

- Simplest possible model adopted:

$$S_\nu(p) = s(p) + A_s\, g(\nu) \left(\frac{\nu}{30\,\mathrm{GHz}}\right)^{-3} + A_d\, g(\nu) \left(\frac{\nu}{300\,\mathrm{GHz}}\right)^{1.7}$$

  - Only marginalization over foreground amplitudes, not spectral indices

- CMB covariance parameterized by two overall scaling amplitudes, *a* and *b*, only:

$$\langle s_{\ell m} s^t_{\ell' m'} \rangle = \begin{pmatrix} 0 & 0 & 0 \\ 0 & aC_\ell^{EE,\mathrm{in}} & 0 \\ 0 & 0 & bC_\ell^{BB,\mathrm{in}} \end{pmatrix} \delta_{\ell\ell'}\delta_{mm'}$$

$$r_2 = \frac{C_2^{BB}}{C_2^{EE}}$$

Foreground model and free parameters

- Simplest possible model adopted:

$$S_\nu(p) = s(p) + A_s\, g(\nu) \left(\frac{\nu}{30\,\mathrm{GHz}}\right)^{-3} + A_d\, g(\nu) \left(\frac{\nu}{300\,\mathrm{GHz}}\right)^{1.7}$$

  - Only marginalization over foreground amplitudes, not spectral indices

- CMB covariance parameterized by two overall scaling amplitudes, *a* and *b*, only:

$$\langle s_{\ell m} s^t_{\ell m} \rangle = \begin{pmatrix} 0 & 0 & 0 \\ 0 & aC_\ell^{EE,\mathrm{in}} & 0 \\ 0 & 0 & bC_\ell^{BB,\mathrm{in}} \end{pmatrix} \delta_{\ell\ell}\delta_{mm}$$

$$r_2 = \frac{C_2^{BB}}{C_2^{EE}}$$

Foreground model and free parameters

- Simplest possible model adopted:

$$S_\nu(p) = s(p) + A_s\, g(\nu) \left(\frac{\nu}{30\,\mathrm{GHz}}\right)^{-3} + A_d\, g(\nu) \left(\frac{\nu}{300\,\mathrm{GHz}}\right)^{1.7}$$

  - Only marginalization over foreground amplitudes, not spectral indices

- CMB covariance parameterized by two overall scaling amplitudes, *a* and *b*, only:

$$\langle s_{\ell m} s^t_{\ell' m'} \rangle = \begin{pmatrix} 0 & 0 & 0 \\ 0 & aC_\ell^{EE,\mathrm{in}} & 0 \\ 0 & 0 & bC_\ell^{BB,\mathrm{in}} \end{pmatrix} \delta_{\ell\ell'} \delta_{mm'}$$

$$r_\ell = \frac{C_\ell^{BB}}{C_\ell^{EE}}$$

Foreground model and free parameters

- Simplest possible model adopted:

$$S_\nu(p) = s(p) + A_s\, g(\nu)\left(\frac{\nu}{30\,\mathrm{GHz}}\right)^{-3} + A_d\, g(\nu)\left(\frac{\nu}{300\,\mathrm{GHz}}\right)^{1.7}$$

  - Only marginalization over foreground amplitudes, not spectral indices

- CMB covariance parameterized by two overall scaling amplitudes, *a* and *b*, only:

$$\langle s_{\ell m} s^t_{\ell' m'}\rangle = \begin{pmatrix} 0 & 0 & 0 \\ 0 & aC_\ell^{EE,\mathrm{in}} & 0 \\ 0 & 0 & bC_\ell^{BB,\mathrm{in}} \end{pmatrix} \delta_{\ell\ell'}\delta_{mm'}$$

- Defined for simplicity a quadrupole-based tensor-to-scalar ratio, r_2:

$$r_2 = \frac{C_2^{BB}}{C_2^{EE}}$$

Foreground model and free parameters

- Simplest possible model adopted:

$$S_\nu(p) = s(p) + A_s\, g(\nu) \left(\frac{\nu}{30\mathrm{GHz}}\right)^{-3} + A_d\, g(\nu) \left(\frac{\nu}{300\mathrm{GHz}}\right)^{1.7}$$

  - Only marginalization over foreground amplitudes, not spectral indices

- CMB covariance parameterized by two overall scaling amplitudes, *a* and *b*, only:

$$\langle s_{\ell m} s^t_{\ell' m'} \rangle = \begin{pmatrix} 0 & 0 & 0 \\ 0 & aC_\ell^{EE,\mathrm{in}} & 0 \\ 0 & 0 & bC_\ell^{BB,\mathrm{in}} \end{pmatrix} \delta_{\ell\ell'}\delta_{mm'}$$

- Defined for simplicity a quadrupole-based tensor-to-scalar ratio, r_2:

$$r_2 = \frac{C_2^{BB}}{C_2^{EE}}$$

Foreground model and free parameters

- Simplest possible model adopted:

$$S_\nu(p) = s(p) + A_s\, g(\nu) \left(\frac{\nu}{30\,\mathrm{GHz}}\right)^{-3} + A_d\, g(\nu) \left(\frac{\nu}{300\,\mathrm{GHz}}\right)^{1.7}$$

  - Only marginalization over foreground amplitudes, not spectral indices

- CMB covariance parameterized by two overall scaling amplitudes, *a* and *b*, only:

$$\langle s_{\ell m} s^t_{\ell' m'} \rangle = \begin{pmatrix} 0 & 0 & 0 \\ 0 & aC_\ell^{EE,\mathrm{in}} & 0 \\ 0 & 0 & bC_\ell^{BB,\mathrm{in}} \end{pmatrix} \delta_{\ell\ell'}\delta_{mm'}$$

- Defined for simplicity a quadrupole-based tensor-to-scalar ratio, r_2:

$$r_2 = \frac{C_2^{BB}}{C_2^{EE}}$$

Foreground model and free parameters

- Simplest possible model adopted:

$$S_\nu(p) = s(p) + A_s\, g(\nu) \left(\frac{\nu}{30\,\mathrm{GHz}}\right)^{-3} + A_d\, g(\nu) \left(\frac{\nu}{300\,\mathrm{GHz}}\right)^{1.7}$$

  - Only marginalization over foreground amplitudes, not spectral indices

- CMB covariance parameterized by two overall scaling amplitudes, *a* and *b*, only:

$$\langle s_{\ell m} s^t_{\ell' m'} \rangle = \begin{pmatrix} 0 & 0 & 0 \\ 0 & aC_\ell^{EE,\mathrm{in}} & 0 \\ 0 & 0 & bC_\ell^{BB,\mathrm{in}} \end{pmatrix} \delta_{\ell\ell'} \delta_{mm'}$$

- Defined for simplicity a quadrupole-based tensor-to-scalar ratio, r_2:

$$r_2 = \frac{C_2^{BB}}{C_2^{EE}}$$

Foreground model and free parameters

- Simplest possible model adopted:

$$S_\nu(p) = s(p) + A_s\, g(\nu) \left(\frac{\nu}{30\mathrm{GHz}}\right)^{-3} + A_d\, g(\nu) \left(\frac{\nu}{300\mathrm{GHz}}\right)^{1.7}$$

  - Only marginalization over foreground amplitudes, not spectral indices

- CMB covariance parameterized by two overall scaling amplitudes, *a* and *b*, only:

$$\langle s_{\ell m} s^t_{\ell m} \rangle = \begin{pmatrix} 0 & 0 & 0 \\ 0 & aC_\ell^{EE,in} & 0 \\ 0 & 0 & bC_\ell^{BB,in} \end{pmatrix} \delta_{\ell \ell'} \delta_{mm'}$$

- Defined for simplicity a quadrupole-based tensor-to-scalar ratio, r_2:

$$r_2 = \frac{C_2^{BB}}{C_2^{EE}}$$

  - Only a few thousand samples is required to get a robust estimate of r_2

Foreground model and free parameters

- Simplest possible model adopted:

$$S_\nu(p) = s(p) + A_s\, g(\nu) \left(\frac{\nu}{30\mathrm{GHz}}\right)^{-3} + A_d\, g(\nu) \left(\frac{\nu}{300\mathrm{GHz}}\right)^{1.7}$$

  - Only marginalization over foreground amplitudes, not spectral indices

- CMB covariance parameterized by two overall scaling amplitudes, *a* and *b*, only:

$$\langle s_{\ell m} s^t_{\ell m} \rangle = \begin{pmatrix} 0 & 0 & 0 \\ 0 & aC_\ell^{EE,\mathrm{in}} & 0 \\ 0 & 0 & bC_\ell^{BB,\mathrm{in}} \end{pmatrix} \delta_{\ell\ell'} \delta_{mm'}$$

- Defined for simplicity a quadrupole-based tensor-to-scalar ratio, r_2:

$$r_2 = \frac{C_2^{BB}}{C_2^{EE}}$$

  - Only a few thousand samples is required to get a robust estimate of r_2
  - No explicit likelihood or post-processing parameter MCMC necessary

Foreground model and free parameters

- Simplest possible model adopted:

$$S_\nu(p) = s(p) + A_s\, g(\nu) \left(\frac{\nu}{30\,\mathrm{GHz}}\right)^{-3} + A_d\, g(\nu) \left(\frac{\nu}{300\,\mathrm{GHz}}\right)^{1.7}$$

  - Only marginalization over foreground amplitudes, not spectral indices

- CMB covariance parameterized by two overall scaling amplitudes, *a* and *b*, only:

$$\langle s_{\ell m} s^t_{\ell m} \rangle = \begin{pmatrix} 0 & 0 & 0 \\ 0 & aC_\ell^{EE,\mathrm{in}} & 0 \\ 0 & 0 & bC_\ell^{BB,\mathrm{in}} \end{pmatrix} \delta_{\ell\ell'} \delta_{mm'}$$

- Defined for simplicity a quadrupole-based tensor-to-scalar ratio, r_2:

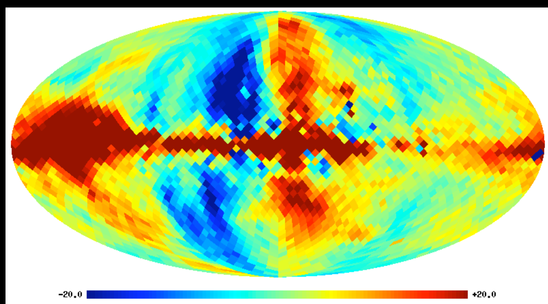
$$r_2 = \frac{C_2^{BB}}{C_2^{EE}}$$

  - Only a few thousand samples is required to get a robust estimate of r_2
  - No explicit likelihood or post-processing parameter MCMC necessary
  - But not *really* what we want, since different from standard (primordial based) *r*

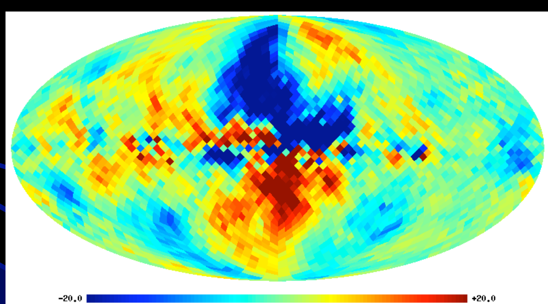
Reconstructed sky maps

Posterior mean

Stokes' Q



Stokes' U



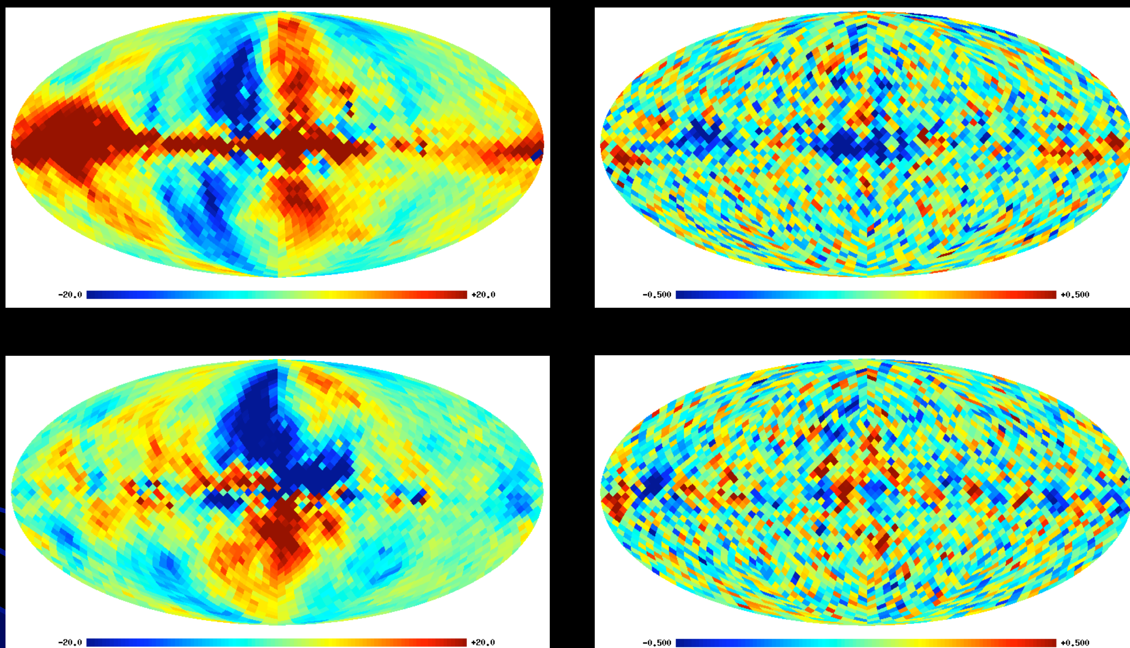
Scale: -20 to 20 μK

Reconstructed sky maps

Posterior mean　　Mean - Input

Stokes' Q

Stokes' U



Scale: -20 to 20 μK

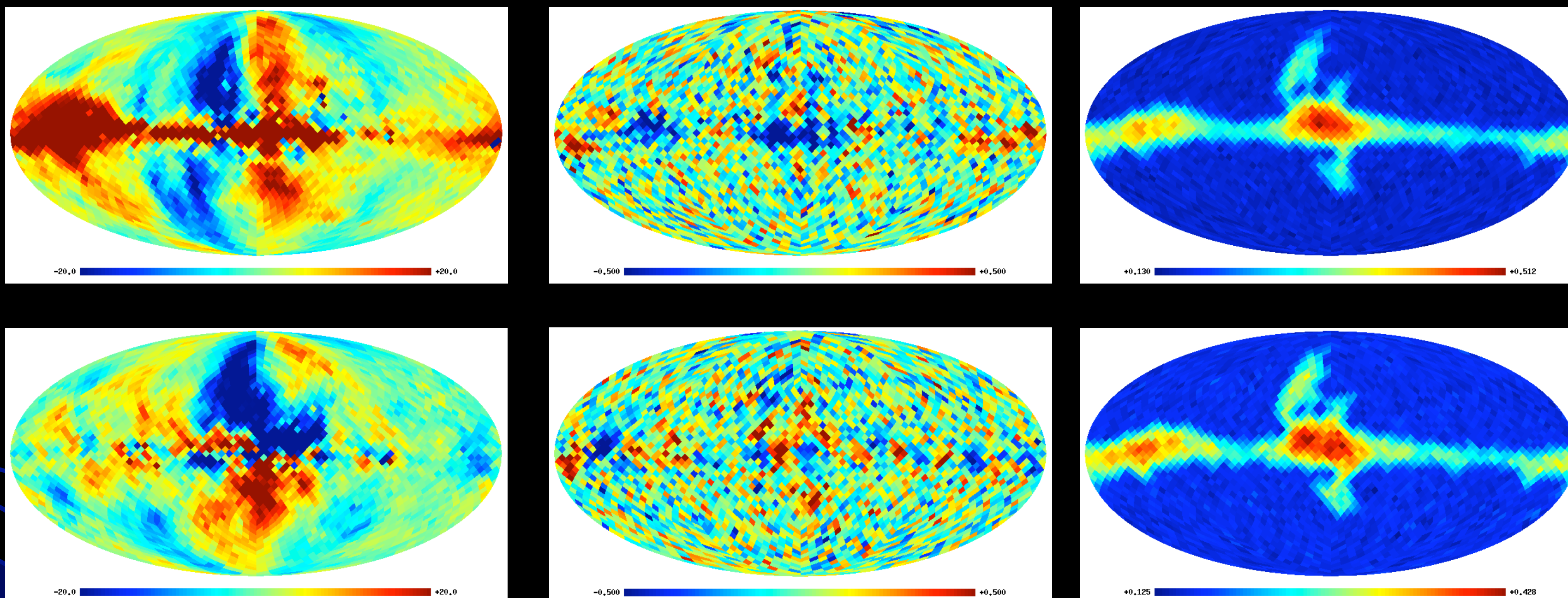
Scale: -0.5 to 0.5 μK

Reconstructed sky maps

Posterior mean | Mean - Input | Posterior RMS

Stokes' Q

Stokes' U



Scale: -20 to 20 μK

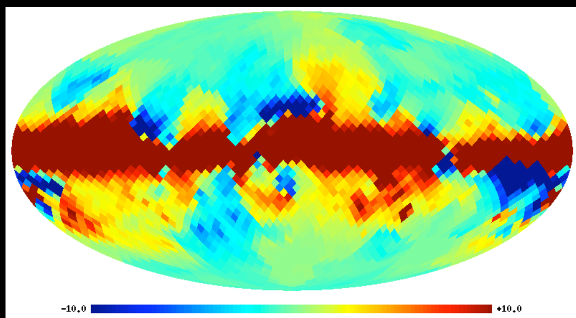
Scale: -0.5 to 0.5 μK

Scale: 0.13 to 0.5 μK

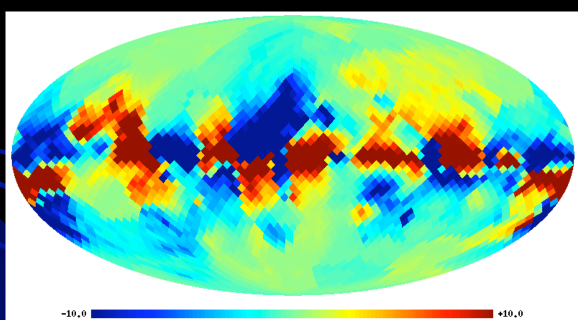
Reconstructed dust maps

Posterior mean

Stokes' Q



Stokes' U



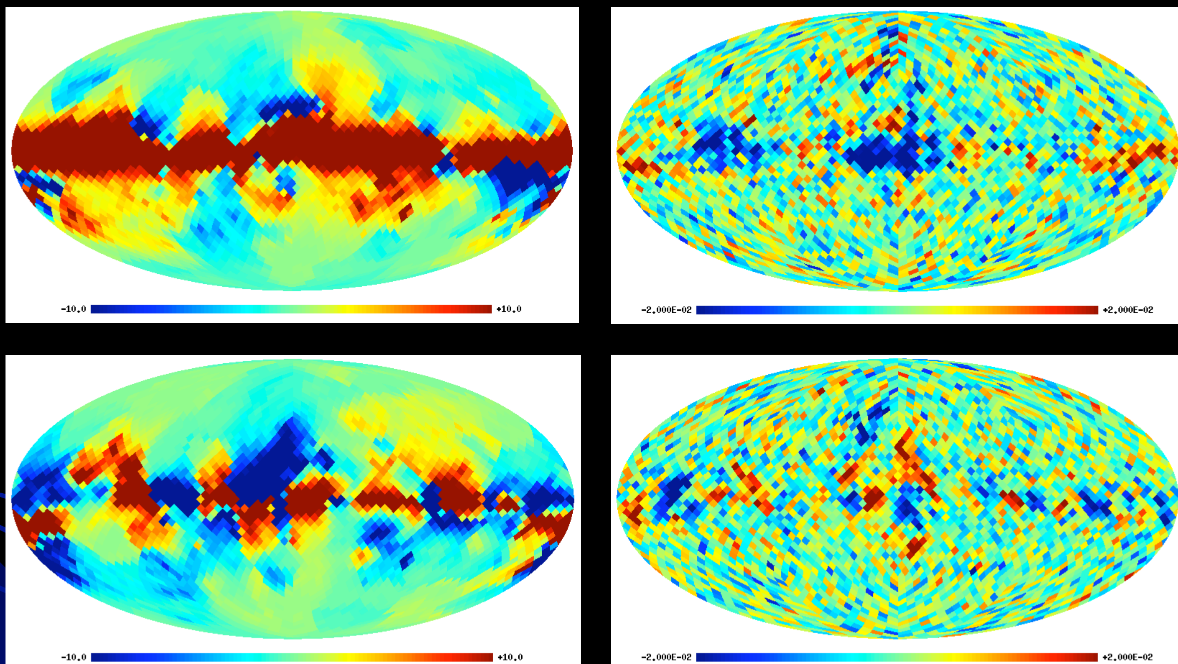
Scale: -10 to 10 μK

Reconstructed dust maps

Posterior mean | Mean - Input

Stokes' Q

Stokes' U



Scale: -10 to 10 μK

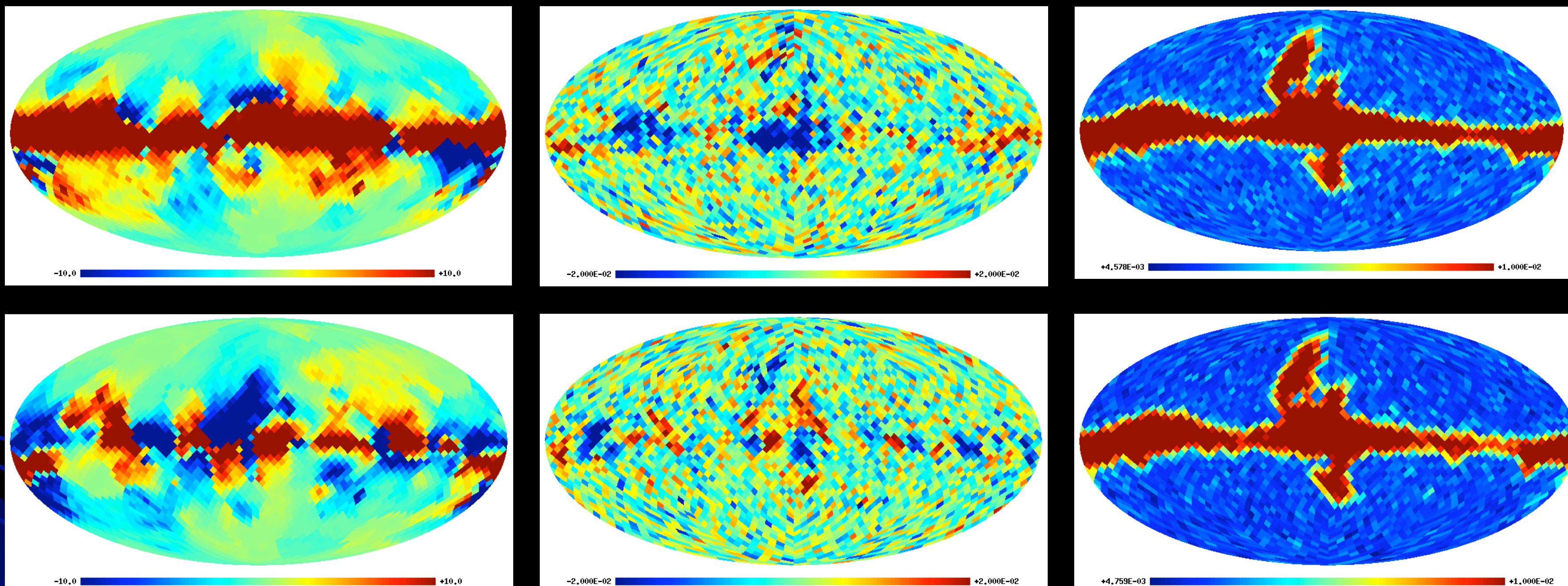
Scale: -0.02 to 0.02 μK

Reconstructed dust maps

Posterior mean | Mean - Input | Posterior RMS

Stokes' Q

Stokes' U

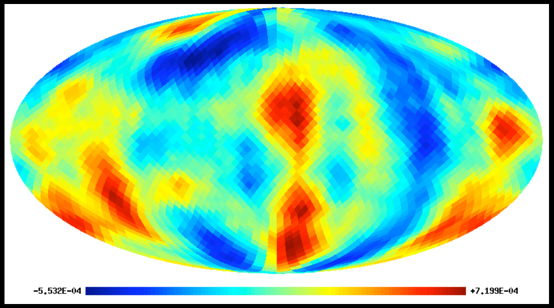


Scale: -10 to 10 μK | Scale: -0.02 to 0.02 μK | Scale: 0.005 to 0.02μK

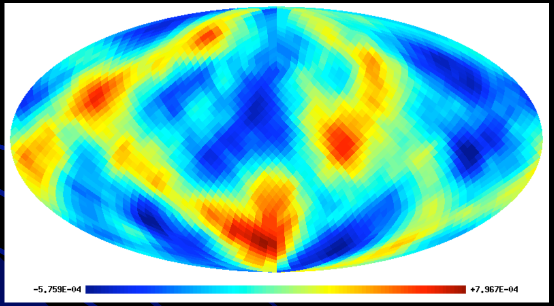
Reconstructed CMB maps

Posterior mean

Stokes' Q



Stokes' U

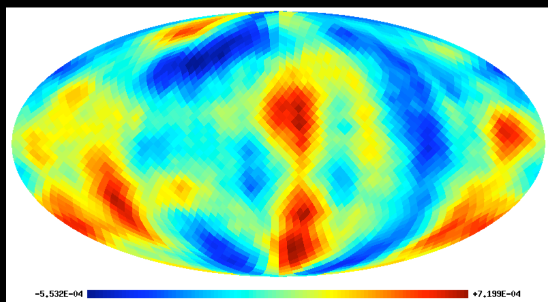


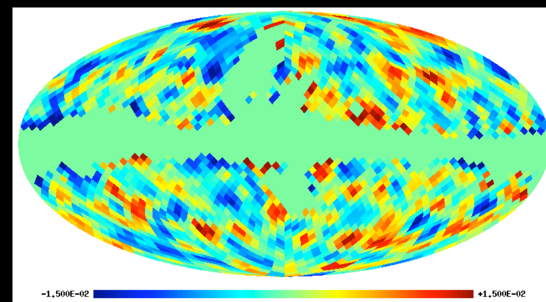
Scale: -0.5 to 0.7 μK

Reconstructed CMB maps

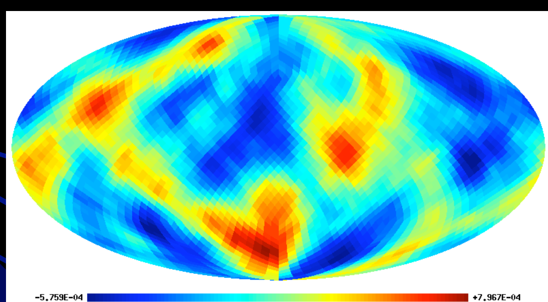
Posterior mean | Mean - Input

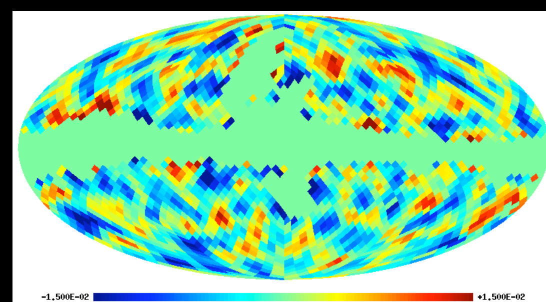
Stokes' Q





Stokes' U





Scale: -0.5 to 0.7 μK

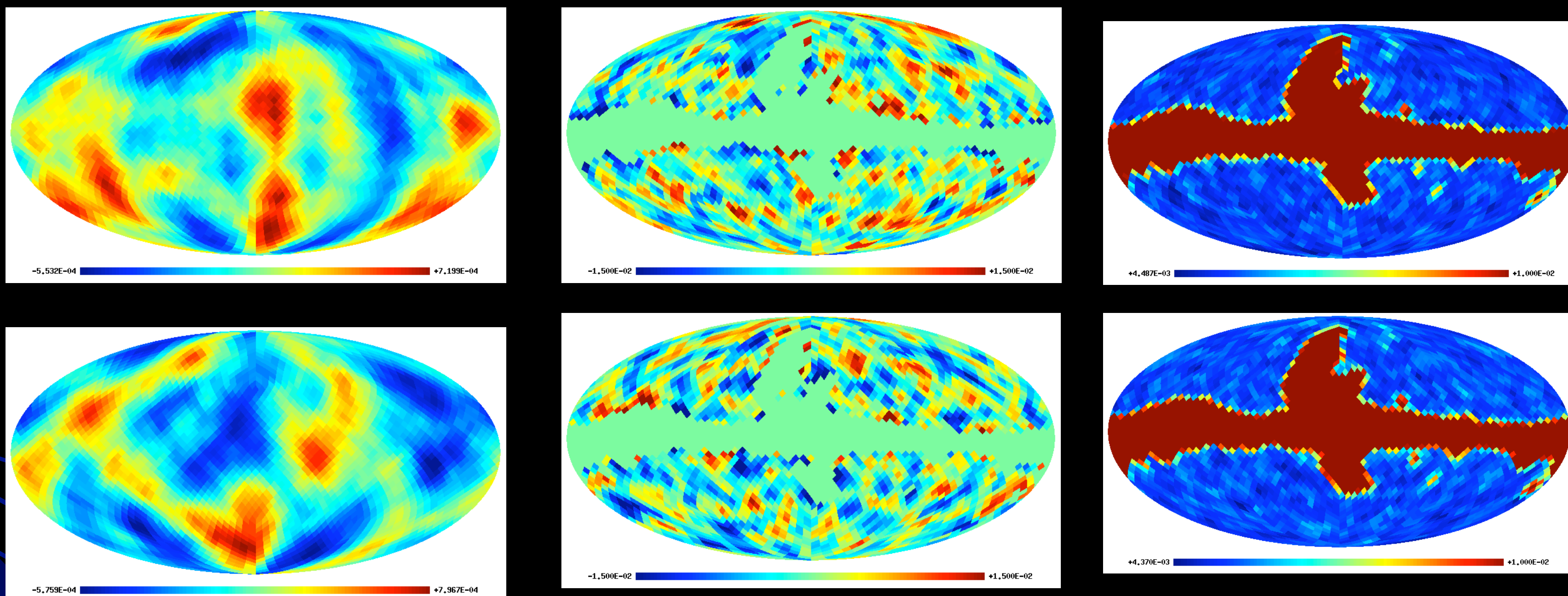
Scale: -0.015 to 0.015 μK

Reconstructed CMB maps

Posterior mean | Mean - Input | Posterior RMS

Stokes' Q

Stokes' U



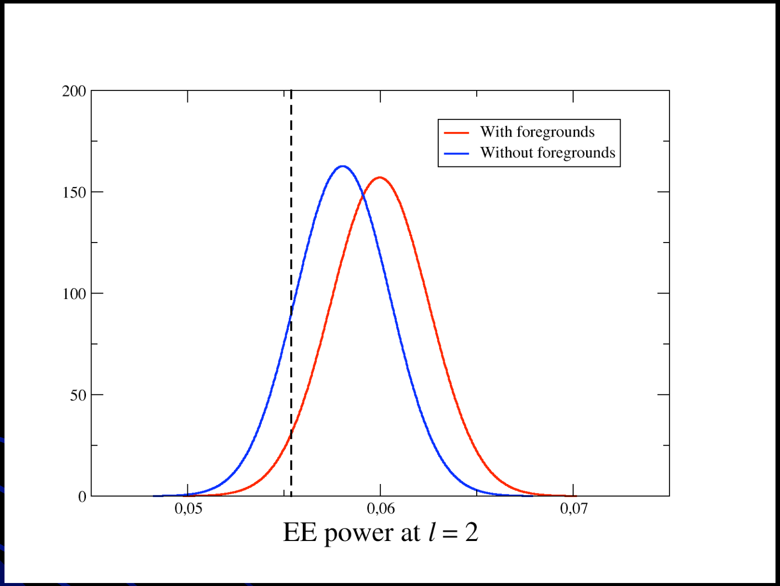
Scale: -0.5 to 0.7 μK

Scale: -0.015 to 0.015 μK

Scale: 0.005 to 0.01 μK

Reconstructed EE and BB power

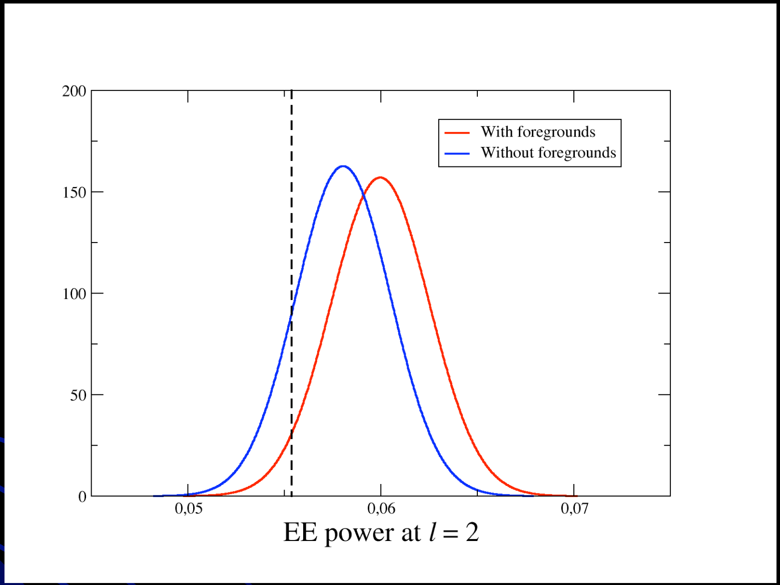
EE power at l = 2



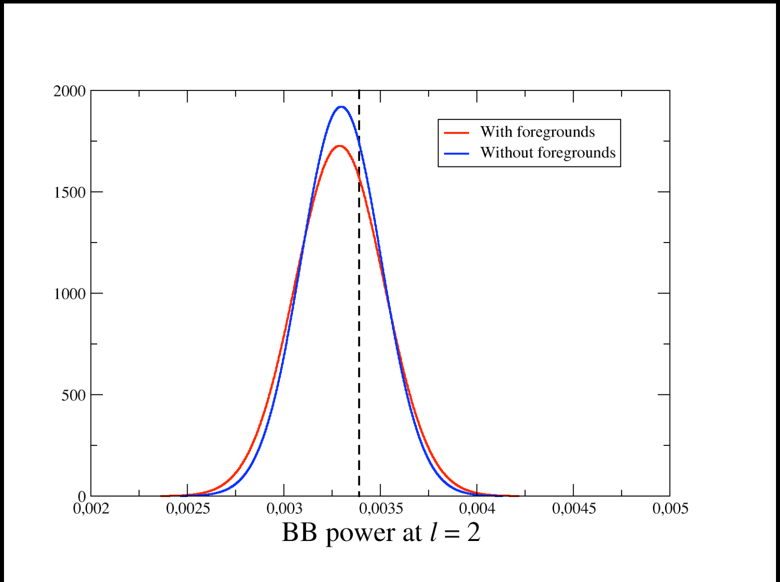
- Blue: Marginal C_2 posterior *without* foregrounds
- Red: Marginal C_2 posterior *with* foregrounds
- Vertical dashed: True input value

Reconstructed EE and BB power

EE power at l = 2

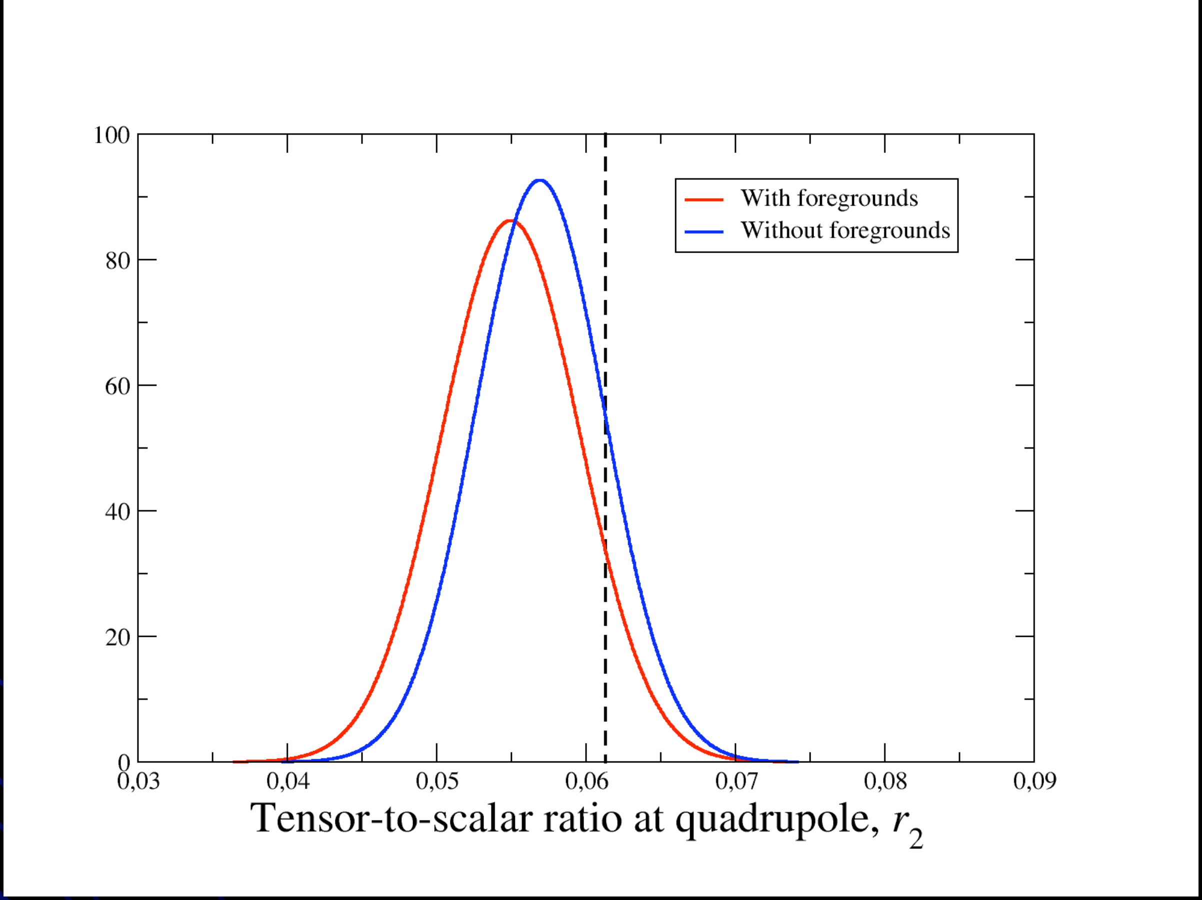


BB power at l = 2



- Blue: Marginal C_2 posterior *without* foregrounds
- Red: Marginal C_2 posterior *with* foregrounds
- Vertical dashed: True input value

Reconstructed tensor-to-scalar ratio



- Blue: Marginal C_2 posterior *without* foregrounds
- Red: Marginal C_2 posterior *with* foregrounds

Experiment comparison

Experiment comparison

- Recall again that
  - the simulation is based on r = 0.1 (r_2 = 0.061), and Nside=16, lmax = 47, 7 degrees FWHM beam

Experiment comparison

- Recall again that
  - the simulation is based on r = 0.1 (r_2 = 0.061), and Nside=16, lmax = 47, 7 degrees FWHM beam
  - the analysis included marginalization over foreground amplitudes only, not spectral indices or other cosmological parameters

Experiment comparison

Experiment	μ_r	σ_r	μ_r / σ_r	$\sigma_r / \sigma_r^{\text{no fg, LC1}}$
LC1; no foregrounds	0.0569	0.0043	13.2	1
LC1; foregrounds	0.0550	0.0046	12.0	1.07
LC2; no foregrounds	0.0609	0.0056	10.9	1.30
LC2; foregrounds	0.0589	0.0062	9.5	1.44

- Recall again that
  - the simulation is based on r = 0.1 (r_2 = 0.061), and Nside=16, lmax = 47, 7 degrees FWHM beam
  - the analysis included marginalization over foreground amplitudes only, not spectral indices or other cosmological parameters

Experiment comparison

Experiment	μ_r	σ_r	μ_r / σ_r	$\sigma_r / \sigma_r^{\text{no fg, LC1}}$
LC1; no foregrounds	0.0569	0.0043	13.2	1
LC1; foregrounds	0.0550	0.0046	12.0	1.07
LC2; no foregrounds	0.0609	0.0056	10.9	1.30
LC2; foregrounds	0.0589	0.0062	9.5	1.44

- Recall again that
  - the simulation is based on r = 0.1 (r_2 = 0.061), and Nside=16, lmax = 47, 7 degrees FWHM beam
  - the analysis included marginalization over foreground amplitudes only, not spectral indices or other cosmological parameters
- So, for this preliminary (demonstration-purpose only) analysis, we find that an amplitude of r = 0.1 gives a 9.5σ detection with LC1, and 12.0σ for LC2

Experiment comparison

Experiment	μ_r	σ_r	μ_r / σ_r	$\sigma_r / \sigma_r^{\text{no fg, LC1}}$
LC1; no foregrounds	0.0569	0.0043	13.2	1
LC1; foregrounds	0.0550	0.0046	12.0	1.07
LC2; no foregrounds	0.0609	0.0056	10.9	1.30
LC2; foregrounds	0.0589	0.0062	9.5	1.44

- Recall again that
  - the simulation is based on r = 0.1 (r_2 = 0.061), and Nside=16, lmax = 47, 7 degrees FWHM beam
  - the analysis included marginalization over foreground amplitudes only, not spectral indices or other cosmological parameters
- So, for this preliminary (demonstration-purpose only) analysis, we find that an amplitude of r = 0.1 gives a 9.5σ detection with LC1, and 12.0σ for LC2
- Next steps are 1) to marginalize over spectral indices, and 2) go to higher resolution (N_{side} = 128, lmax = 300, 90 arcmin FWHM)

Commander as a forecasting tool?

Commander as a forecasting tool?

- Commander is currently being prepared for Planck, not for CMBPol

Commander as a forecasting tool?

- Commander is currently being prepared for Planck, not for CMBPol
  - Target for Planck is Nside=64 with dense noise covariance (~150,000 pixels), and lmax = 150. Other tools will be used at higher l's

Commander as a forecasting tool?

- Commander is currently being prepared for Planck, not for CMBPol
  - Target for Planck is Nside=64 with dense noise covariance (~150,000 pixels), and lmax = 150. Other tools will be used at higher l's
  - Computational time is expected at ~few CPU hours per sample

Commander as a forecasting tool?

- Commander is currently being prepared for Planck, not for CMBPol
  - Target for Planck is Nside=64 with dense noise covariance (~150,000 pixels), and lmax = 150. Other tools will be used at higher l's
  - Computational time is expected at ~few CPU hours per sample
    - Corresponding number is ~five to ten times higher for CMBPol because of higher signal-to-noise

Commander as a forecasting tool?

- Commander is currently being prepared for Planck, not for CMBPol
  - Target for Planck is Nside=64 with dense noise covariance (~150,000 pixels), and lmax = 150. Other tools will be used at higher l's
  - Computational time is expected at ~few CPU hours per sample
    - Corresponding number is ~five to ten times higher for CMBPol because of higher signal-to-noise
  - So, while Gibbs sampling is exact, it is NOT cheap!

Commander as a forecasting tool?

- Commander is currently being prepared for Planck, not for CMBPol
  - Target for Planck is Nside=64 with dense noise covariance (~150,000 pixels), and lmax = 150. Other tools will be used at higher l's
  - Computational time is expected at ~few CPU hours per sample
    - Corresponding number is ~five to ten times higher for CMBPol because of higher signal-to-noise
  - So, while Gibbs sampling is exact, it is NOT cheap!
    - I've recently done an Nside=512, lmax=1024 analysis for a WMAP V1 only simulation, and the cost was 20,000 CPU hours

Commander as a forecasting tool?

- Commander is currently being prepared for Planck, not for CMBPol
  - Target for Planck is Nside=64 with dense noise covariance (~150,000 pixels), and lmax = 150. Other tools will be used at higher l's
  - Computational time is expected at ~few CPU hours per sample
    - Corresponding number is ~five to ten times higher for CMBPol because of higher signal-to-noise
  - So, while Gibbs sampling is exact, it is NOT cheap!
    - I've recently done an Nside=512, lmax=1024 analysis for a WMAP V1 only simulation, and the cost was 20,000 CPU hours
  - In short, do *not* expect lmax ~ 2000 for CMBPol anytime soon!

Commander as a forecasting tool?

- Commander is currently being prepared for Planck, not for CMBPol
  - Target for Planck is Nside=64 with dense noise covariance (~150,000 pixels), and lmax = 150. Other tools will be used at higher l's
  - Computational time is expected at ~few CPU hours per sample
    - Corresponding number is ~five to ten times higher for CMBPol because of higher signal-to-noise
  - So, while Gibbs sampling is exact, it is NOT cheap!
    - I've recently done an Nside=512, lmax=1024 analysis for a WMAP V1 only simulation, and the cost was 20,000 CPU hours
  - In short, do *not* expect lmax ~ 2000 for CMBPol anytime soon!

Commander as a forecasting tool?

- Commander is currently being prepared for Planck, not for CMBPol
  - Target for Planck is Nside=64 with dense noise covariance (~150,000 pixels), and lmax = 150. Other tools will be used at higher l's
  - Computational time is expected at ~few CPU hours per sample
    - Corresponding number is ~five to ten times higher for CMBPol because of higher signal-to-noise
  - So, while Gibbs sampling is exact, it is NOT cheap!
    - I've recently done an Nside=512, lmax=1024 analysis for a WMAP V1 only simulation, and the cost was 20,000 CPU hours
  - In short, do *not* expect lmax ~ 2000 for CMBPol anytime soon!

- The Gibbs sampler is (in my opinion) the tool of choice for a final analysis, but not for forecasting

Commander as a forecasting tool?

- Commander is currently being prepared for Planck, not for CMBPol
  - Target for Planck is Nside=64 with dense noise covariance (~150,000 pixels), and lmax = 150. Other tools will be used at higher l's
  - Computational time is expected at ~few CPU hours per sample
    - Corresponding number is ~five to ten times higher for CMBPol because of higher signal-to-noise
  - So, while Gibbs sampling is exact, it is NOT cheap!
    - I've recently done an Nside=512, lmax=1024 analysis for a WMAP V1 only simulation, and the cost was 20,000 CPU hours
  - In short, do *not* expect lmax ~ 2000 for CMBPol anytime soon!

- The Gibbs sampler is (in my opinion) the tool of choice for a final analysis, but not for forecasting
  - Accuracy is of less importance; speed is essential

Summary and conclusions

Summary and conclusions

- The CMB Gibbs sampler provides us with the full joint CMB posterior

Summary and conclusions

- The CMB Gibbs sampler provides us with the full joint CMB posterior
  - Allows for seamless marginalization over numerous systematics, most importantly non-cosmological foregrounds

Summary and conclusions

- The CMB Gibbs sampler provides us with the full joint CMB posterior
  - Allows for seamless marginalization over numerous systematics, most importantly non-cosmological foregrounds
  - For mid- to low-l's the comptuational expense is quite tractable

Summary and conclusions

- The CMB Gibbs sampler provides us with the full joint CMB posterior
  - Allows for seamless marginalization over numerous systematics, most importantly non-cosmological foregrounds
  - For mid- to low-l's the comptuational expense is quite tractable

Summary and conclusions

- The CMB Gibbs sampler provides us with the full joint CMB posterior
  - Allows for seamless marginalization over numerous systematics, most importantly non-cosmological foregrounds
  - For mid- to low-l's the comptuational expense is quite tractable

Summary and conclusions

- The CMB Gibbs sampler provides us with the full joint CMB posterior
  - Allows for seamless marginalization over numerous systematics, most importantly non-cosmological foregrounds
  - For mid- to low-l's the comptuational expense is quite tractable
- So, as briefly demonstrated here, it *can* be used for forecasting and experiment design...

Summary and conclusions

- The CMB Gibbs sampler provides us with the full joint CMB posterior
  - Allows for seamless marginalization over numerous systematics, most importantly non-cosmological foregrounds
  - For mid- to low-l's the comptuational expense is quite tractable
- So, as briefly demonstrated here, it *can* be used for forecasting and experiment design...

Summary and conclusions

- The CMB Gibbs sampler provides us with the full joint CMB posterior
  - Allows for seamless marginalization over numerous systematics, most importantly non-cosmological foregrounds
  - For mid- to low-l's the comptuational expense is quite tractable

- So, as briefly demonstrated here, it *can* be used for forecasting and experiment design...

Summary and conclusions

- The CMB Gibbs sampler provides us with the full joint CMB posterior
  - Allows for seamless marginalization over numerous systematics, most importantly non-cosmological foregrounds
  - For mid- to low-l's the comptuational expense is quite tractable
- So, as briefly demonstrated here, it *can* be used for forecasting and experiment design...
- ... but I would argue that this is *not* its main purpose:

Summary and conclusions

- The CMB Gibbs sampler provides us with the full joint CMB posterior
  - Allows for seamless marginalization over numerous systematics, most importantly non-cosmological foregrounds
  - For mid- to low-l's the comptuational expense is quite tractable
- So, as briefly demonstrated here, it *can* be used for forecasting and experiment design...
- ... but I would argue that this is *not* its main purpose:
  - The Gibbs sampler provides exact answers, but it is not cheap

Summary and conclusions

- The CMB Gibbs sampler provides us with the full joint CMB posterior
  - Allows for seamless marginalization over numerous systematics, most importantly non-cosmological foregrounds
  - For mid- to low-l's the comptuational expense is quite tractable
- So, as briefly demonstrated here, it *can* be used for forecasting and experiment design...
- ... but I would argue that this is *not* its main purpose:
  - The Gibbs sampler provides exact answers, but it is not cheap
  - For experimental design, it's better to use cheaper methods (e.g., Fisher or single-pixel analysis)

Summary and conclusions

- The CMB Gibbs sampler provides us with the full joint CMB posterior
  - Allows for seamless marginalization over numerous systematics, most importantly non-cosmological foregrounds
  - For mid- to low-l's the comptuational expense is quite tractable

- So, as briefly demonstrated here, it *can* be used for forecasting and experiment design...

- ... but I would argue that this is *not* its main purpose:
  - The Gibbs sampler provides exact answers, but it is not cheap
  - For experimental design, it's better to use cheaper methods (e.g., Fisher or single-pixel analysis)
  - Then analyse a few select cases with the full Gibbs machinery to verify the above findings, and demonstrate our general analysis capability