

fiTQun RJMCMC Documentation

Richard Calland

October 13, 2016

This document describes the modelling of a neutrino event inside a water cherenkov detector in a Bayesian construction, using fiTQun to calculate the likelihood terms. First, the motivation for such an analysis is discussed, along with the introduction of the concepts involved. Then, the model will be explained along with choices of priors, and finally the specific implementation of the sampler will be described, along with the derivations of acceptance probabilities.

1 Motivation

1.1 Current fiTQun method

fiTQun in its current state constructs a joint likelihood from charge and time information, and maximizes this using the SIMPLEX routine [?]. In the case when there are multiple rings inside an event, the current strategy is to perform a series of fits increasing the number of rings each time, and using the resulting likelihood to decide which fit performed better. This method runs into problems as the number of rings increases, and in order to feasibly run this fit, the number of parameters are reduced by assuming all rings share a common vertex.

1.2 Proposed changes

Another approach to solving the multi-ring problem is to replace SIMPLEX with another method. By constructing the problem as a Mixture model, where each ring is a component of the mixture, we can sample the Posterior probability using a suitable method (that allows the addition/removal of components). The maximum of the Posterior probability will provide the most likely configuration of rings given a data set.

2 The Mixture Model and Sampler

A mixture model is a summation of PDFs that are weighted, such that the total weight always sums to 1:

$$f_{total} = \sum_{i=1}^{i=k} \omega_i f(\theta_i) \quad \sum_{i=1}^{i=k} \omega_i = 1 \quad (1)$$

where ω_i is the weight of component i and θ_i are the parameters describing component i , for a total of k components. The constraint that all weights sum to 1 means that probability is conserved even when the number of components is changed.

2.1 Reversible Jump Sampler

In order to sample the mixture model, a trans-dimensional sampler must be used. This means that the sampler must be able to add or remove components from the mixture, effectively changing the number of parameters. The usual Markov chain Monte Carlo (MCMC) approach to sampling is to use the Metropolis-Hastings acceptance probability, accepting a new step with probability:

$$P_{accept} = \min(1, a) \quad (2)$$

where a is defined as

$$a_{mh} = \frac{l(x|\theta')p(\theta')q(\theta|\theta')}{l(x|\theta)p(\theta)q(\theta'|\theta)} \quad (3)$$

where θ are the parameters of the current model, θ' the parameters of the proposed model, $l(x|...)$ the likelihood of data x , $p(...)$ the prior and $q(a|b)$ the proposal probability of moving to state a from state b .

This holds for any samples that are proposed within the model. The Reversible Jump MCMC sampler allows jumps between models with different numbers of parameters, accepting said moves using:

$$a_{rj} = \frac{l(x|\theta')p(\theta')}{l(x|\theta)p(\theta)q(u)}|J| \quad (4)$$

where there are dimension matching terms in the form of $q(u)$ to account for the different number of parameters of each model. In addition, a Jacobian term J is used to adjust the probability measure according to the parameter transformations, however this term is often 1.

To elaborate further on the reversible jump move, we will consider an example. Typically to add a new ring to the model, the new ring parameters are constructed using random numbers $u \sim g()$ (meaning, u is drawn from the distribution $g()$), which are then turned into the ring parameters via a transformation of the current model parameters $\theta_{new} = f(u, \theta)$. If we draw the new ring parameters direct from the prior distributions, then $\theta_{new} = u \sim p()$, and we can replace $q(u)$ in 4 with $p(\theta_{new})$. Now, we have $p(\theta') = p(\theta) + p(\theta_{new})$, where the $p(\theta_{new})$ term cancels with the proposal function $q(u) = p(\theta_{new})$ on the denominator. As there are no transformation of variables involved, the Jacobian term is 1, thus, the acceptance probability reduces to a simple likelihood ratio for this move. For the cases where new ring parameters are not drawn from the prior distribution, the $q(u)$ and J terms ensure detailed balance in the chain.

3 A Bayesian Model for Cherenkov Rings in a Neutrino Event

In fitQun, the predicted charge and time for each ring are parameterized as having a vertex x, y, z, t , direction θ, ϕ , momentum p and particle identification (PID) PID (discrete variable). The ring also has an energy loss parameter E_{loss} in the case of a MIP particle that undergoes scattering. The E_{loss} parameter is within the range $[0.01, 1.0]$ and describes the fraction of total visible energy that the particle deposited in the detector.

The vertex position parameters have Gaussian priors centered at 0, with a width equal to the diagonal distance across the SK tank for position, and the vertex time parameter has a mean of 1000 ns with a width the size of the time window (usually around 500 ns).

The direction parameters use spherical coordinates, and have a prior that loses probability near the poles.

$$i\text{amasphericalprior} \quad (5)$$

To allow the chain to mix properly between different states of PID and momentum, we reparameterize such that there is a global visible energy E_{vis} parameter, which is distributed amongst the rings using the component weight w_i from 1. Using the rings visible energy $E_{vis}w_i$, the momentum is determined by considering the momentum threshold of the PID:

$$p = p_{thr}(PID) + \text{Energy2Momentum}(E_{vis}w_i, PID) \quad (6)$$

The weights w of all rings must always sum to unity. We choose a Dirichlet prior for w to reflect this:

$$\text{Dir}(w, \alpha) = \frac{1}{B(\alpha)} \prod_{i=1}^k w_i^{\alpha-1} \quad (7)$$

where the concentration parameter α controls the distribution of weight parameters. For example, $\alpha < 1$ would favour uneven distributions of weights, $\alpha > 1$ would favour all weights to have similar values, and $\alpha = 1$ places no preference on weight values.

E_{vis} has a Gaussian prior centered at 1000 with a width of x . The momentum p is now a function of E_{vis} , w_i and PID , so it does not receive its own prior.

In this construction, the number of rings (components) k is not fixed; it is allowed to change depending on the reversible jump sampler to propose correct transitions from k to $k \pm 1$ components. Therefore, as a variable, we choose a Poisson prior with a mean of 1 for k .

Eloss parameter.

The PID parameter is a discrete positive integer, with each value having equal probability.

4 Sampler Move Types

At each MCMC step, a move is selected at random from all possible moves:

- Update x, y, z, t, θ, ϕ
- Update E_{vis}
- Update PID
- Birth/death move
- Split/merge move
- Update E_{loss}

The probability of each move occurring is a tuneable parameter. The move types are *within-model* or *between-model* moves, depending on whether they change the number of rings or not.

4.1 Within Model Move

The within-model moves follows a standard Metropolis-Hastings algorithm, where each parameter is updated using a random sample u from their corresponding proposal distribution $q()$. The proposal distribution is then used in the acceptance probability 3 to ensure detailed balance. If $q()$ is symmetric (e.g. Gaussian), then the proposal terms in 3 cancel.

The $x, y, z, t, \theta, \phi, E_{vis}$ and E_{loss} parameters all follow this simple formula, using Gaussian distributions with tuneable parameters σ_i as widths.

Additionally, to increase the acceptance rate in the case of a large number of rings, the step sizes σ_i are scaled by a factor $1/\sqrt{k}$.

The weights are updated following

$$u_i \sim g() \quad w'_i = w_i e^{u_i}. \quad (8)$$

The weights are then normalized to 1. The proposal term is calculated THIS IS WRONG
 $p_{prop} = \sum_{i=1}^k w'_i - w_i$

When the update PID move is selected, a randomly selected ring has its PID parameter value resampled from the prior. Additionally, to assist the chain in moving between PID hypotheses, the vertex and time parameters are shifted according to a predetermined distribution. This distribution is generated from 1 ring fits for all PID hypotheses to the same set of single ring particle gun samples.

4.2 Between Model Move

4.2.1 Birth/death Move

When the *birth/death move* is chosen, the decision to either add or remove a ring is made probabilistically. The birth probability p_{birth} can be adjusted to improve mixing, however the acceptance probability must be adjusted to reflect that $p_{birth} \neq 0.5$ by multiplying by $p_{birth}/(1 - p_{birth})$. Additionally, when $k = 1(k_{max})$, $p_{birth} = 1(0)$.

The birth move adds a ring to the model using one of the birth moves explained in sections 4.2.2 and 4.2.3. The death move removes a randomly selected ring from the model. The acceptance for the death move is $a_{death} = a_{birth}^{-1}$, where a_{birth} is the acceptance for the *draw from prior* move 9, with w_{birth} replaced with the weight of the removed ring w_{death} .

4.2.2 Draw from Prior Birth Move

This move was described in section 2.1, so there are a few details left to talk about. The Jacobian term is no longer 1 due to the Dirichlet nature of the weight parameter. The acceptance (to be used with equation 2) is defined as

$$a_{prior\text{draw}} = \frac{l(x|\theta')}{l(x|\theta)q(w_{birth})}(1 - w_{birth})^{k-1} \quad (9)$$

where w_{birth} is the weight of the new ring, $q()$ is a Beta distribution $B(1, k)$ and k the number of components before the move was attempted.

4.2.3 Add-on-end Birth Move

There is a high likelihood that multi ring events will have ring vertices downstream from other rings. We can construct a more efficient move to take advantage of this knowledge, by proposing a new vertex position $v' = \{x, y, z\}$ and time t' parameters along the direction $d = \{d_x, d_y, d_z\}$ of a ring chosen at random:

$$u \sim g() \quad v' = v + ud \quad t' = t + (ud)n_{wtr}c^{-1} \quad (10)$$

where u is a random sample from $g()$ that defines exactly how far along the direction we will add the new ring, n_{wtr} is the refractive index of water, and c the speed of light. For this analysis, the vertex proposal distribution from the within-model move is used in place of $g()$.

The new ring weight is decided by sampling from a beta distribution $w^* \sim B(1, k)$, then the existing rings are scaled down according to $w'_i = (1 - w^*)w_i$, to ensure that the weights sum to unity. All other parameters are drawn from their prior distributions.

The Jacobian $|J|$ of this move is simple to calculate. For the vertex parameters, the Jacobian is the direction component:

$$|J| = \begin{vmatrix} \frac{\partial v'}{\partial v} & \frac{\partial v'}{\partial u} \\ \frac{\partial v}{\partial v} & \frac{\partial v}{\partial u} \end{vmatrix} = \begin{vmatrix} 1 & d \\ 1 & 0 \end{vmatrix} = -d \quad (11)$$

and similarly for the time parameter, we find $|J| = -n_{wtr}c^{-1}$. The weight Jacobian term is $(1 - w^*)^{k-1}$. The Jacobian for all other parameters is 1. The acceptance for the add-on-end move is:

$$a_{addonend} = \frac{l(x|\theta')p(\theta')}{l(x|\theta)p(\theta)q_{vertex}(u)^4q_{direction}(\theta, \phi)q_w(w_{birth})}(-d)n_{wtr}c^{-1}(1 - w_{birth})^{k-1} \quad (12)$$

5 Post Processing

The output of the RJMCMC sampler is not immediately useful. In order to extract information to pass onwards to physics analyses, the samples produced by the RJMCMC must be processed.

The RJMCMC output is a set of samples from parameter space, including samples from models with different numbers of rings k ???. We wish to extract the most likely number of rings, and the most likely parameter values for each of those rings. The first step is to determine the most likely number of rings, and condition the Posterior on that. This is done by simply constructing a histogram of k , selecting the bin with the most entries k_{MAP} , and disposing of samples where $k \neq k_{MAP}$ (*MAP = maximum a posteriori*).

Now we have a set of samples that have the same number of rings, so the next step is to group all parameters together that correspond to the same ring. This is non-trivial because the ordering of the rings may change between samples due to the RJMCMC's ability to add and remove rings. The solution taken here is to step through each sample, and reorder the rings in each sample based on some metric. The simplest metric to use would be ordering by weight w , where the ring with the lowest weight is "ring 1", the second lowest is "ring 2" etc. However, this solution has issues when two rings have similar weights, causing samples to be misidentified. As the direction parameters are relatively well constrained, the Posterior probability distributions are narrow and the chances of overlap is much smaller than that for the weight parameter. Thus, the ordering metric chosen is the distance from an arbitrary fixed position in 2D $\theta\phi$ space.

The samples for individual rings have now been grouped together, and so now the ring parameters can be estimated by constructing 1D histograms for each parameter, and finding the maximum.