LASER INTERFEROMETER GRAVITATIONAL WAVE OBSERVATORY
- LIGO -
CALIFORNIA INSTITUTE OF TECHNOLOGY
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

Technical Note **LIGO-T16XXXX-v1** 2016/01/19

Multi-channel coherence

Gabriele Vajente

Distribution of this document:
LIGO Scientific Collaboration

California Institute of Technology
LIGO Project, MS 18-34
Pasadena, CA 91125
Phone (626) 395-2129
Fax (626) 304-9834
E-mail: info@ligo.caltech.edu

Massachusetts Institute of Technology
LIGO Project, Room NW17-161
Cambridge, MA 02139
Phone (617) 253-4824
Fax (617) 253-7014
E-mail: info@ligo.mit.edu

LIGO Hanford Observatory
Route 10, Mile Marker 2
Richland, WA 99352
Phone (509) 372-8106
Fax (509) 372-8137
E-mail: info@ligo.caltech.edu

LIGO Livingston Observatory
19100 LIGO Lane
Livingston, LA 70754
Phone (225) 686-3100
Fax (225) 686-7189
E-mail: info@ligo.caltech.edu

http://www.ligo.caltech.edu/

Abstract

We describe a technique to estimate the coherence of a target signal with a set of correlated auxiliary channels. This is an extension of the more common two-channel magnitude square coherence.

1 Multi-channel coherence

Let $y(\omega)$ be the target channel, which is assumed to be a linear combination of a set of auxiliary channels $x_i(\omega)$ plus some uncorrelated additive noise $n(\omega)$:

$$y(\omega) = \sum_{i=1}^{N} \alpha_i(\omega) x_i(\omega) + n(\omega) \tag{1}$$

where the coupling coefficients $\alpha_i(\omega)$ are unknown and frequency dependent. We allow the auxiliary channels $x_i(\omega)$ to be correlated with each other, but uncorrelated with the additional noise in the target signal.

The task at hand is to find how much of the target signal $y(\omega)$ can be attributed to a linear combination of the auxiliary channels $x_i(\omega)$, without over-couting due to the cross correlations in the auxiliary channels.

Let $S[a,b]$ be the cross-spectral density (CSD) of the two signals a and b. It can be computed for example with the well known Welch method. In general, the CSD has the following properties:

$$\begin{aligned} S[a+b,c] &= S[a,c]+S[b,c] \\ S[a,b] &= S^*[b,a] \\ S[\alpha(\omega)a,b] &= \alpha^*(\omega)S[a,b] \end{aligned}$$

where a,b and c are signals, * indicates complex conjugation and $\alpha(\omega)$ is a generic transfer function. The cross spectral density is in generale a function of frequency ω.

The power spectral density (PSD) of the target signal is given by:

$$\begin{aligned} S[y,y] &= S[\sum_{i=1}^{N} \alpha_i(\omega) x_i(\omega) + n(\omega), \sum_{j=1}^{N} \alpha_j(\omega) x_j(\omega) + n(\omega)] \\ &= \sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_i^*(\omega)\, S[x_i,x_j]\, \alpha_j(\omega) + S[n,n] \end{aligned}$$

where we used eq. 1 and the fact that the auxiliary signals are uncorrelated to the additive noise, so that $S[x_i, n] = 0$. We can rewrite the last equation in matrix form with the following definitions:

$$\mathbf{A} = \begin{pmatrix} \alpha_1(\omega) \\ \vdots \\ \alpha_N(\omega) \end{pmatrix}$$

$$\mathbf{S_X} = \begin{pmatrix} S[x_1,x_1] & S[x_1,x_2] & \dots & S[x_1,x_N] \\ \vdots & \vdots & \ddots & \vdots \\ S[x_N,x_1] & S[x_N,x_2] & \dots & S[x_N,x_N] \end{pmatrix}$$

Note that $\mathbf{S_X}$ is a Hemitean matrix: $\mathbf{S_X}^H = \mathbf{S_X}$ where ^{H} indicate the operation of transposition and complex conjugation. Using this notation, the target signal PSD is simply:

$$S[y,y] = \mathbf{A}^H \mathbf{S_X} \mathbf{A} + S[n,n] \tag{2}$$

The CSD of the target signal with one of the auxiliary channels can be written as

$$S[y,x_i] = \sum_{i=1}^{N} \alpha_i^*(\omega) S[x_i, x_j] \tag{3}$$

and in vector notation

$$\mathbf{S_Y} = (S[y,x_1], \dots S[y,x_N])$$
$$\mathbf{S_Y} = \mathbf{A}^H \mathbf{S_X} \tag{4}$$

Given the time series of y and x_i, we can estimate both the vector $\mathbf{S_Y}$ and the matrix $\mathbf{S_X}$. We can then solve eq. 4 for the unknown coefficients $\alpha_i(\omega)$:

$$\mathbf{A} = \mathbf{S_X}^{-1} \mathbf{S_Y}^H \tag{5}$$

and we can substitute this result into eq. 2 to get:

$$S[y,y] = S[n,n] + \mathbf{S_Y} \mathbf{S_X}^{-1} \mathbf{S_Y}^H \tag{6}$$

We finally define the *multi channel coherence* as the following measurement of the fraction of the target signal PSD that can be explained with a linear combination of the auxiliary signals:

$$C(\omega) = \frac{S[y,y] - S[n,n]}{S[y,y]} = \frac{\mathbf{S_Y} \mathbf{S_X}^{-1} \mathbf{S_Y}^H}{S[y,y]} \tag{7}$$

In general, since the auxiliary channels can be strongly coupled, one cannot ask which of them is the largest contribution to the noise. However, it makes sense to find which is the combination of channels that contributes most at a given frequency. One way to do this is to diagonalize the CSD matrix $\mathbf{S_X}$ at each frequency. This is in general possible, since this matrix is Hermitian:

$$\mathbf{S_X}(\omega) = \mathbf{U}(\omega)\, \mathbf{D}(\omega)\, \mathbf{U}(\omega)^{-\mathbf{1}} \tag{8}$$

where $\mathbf{D}$ is the diagonal matrix of the eigenvalues $D_i(\omega)$ of $\mathbf{S_X}$. We can then rewrite the total multi-channel coherence of eq. 7 as a sum of terms, each one giving the contribution to the coherence from a linear combination of the auxiliary channels that corresponds to one of the eigenvectors:

$$C(\omega) = \frac{1}{S[y,y]} \sum_{i=1}^{N} \frac{|(\mathbf{S_Y}(\omega)\mathbf{U}(\omega))_i|^2}{D_i(\omega)} \quad (9)$$

where we used that $\mathbf{S_Y}(\omega)\mathbf{U}(\omega)$ is a vector and used the subscript i to indicate the i-th component. Additionally, each column of the matrix $\mathbf{U}(\omega)$ gives the coefficient of the linear combination of the auxiliary channels corresponding to the i-th contribution to the multi-channel coherence.

A MATLAB code

```
function [c, A, fr, U, e, ccum] = multicoherence(y, X, npp, fs)
    % Input:
    % y = target channel
    % X = matrix of auxiliary channel (one per row)
    % npp = number of points for FFT computation
    % fs = sampling frequency
    %
    % Output:
    % c = multi-channel total coherence
    % A = reconstructed coupling coefficients (frequency dependent)
    % fr = vector of frequency values
    % U = matrix that diagonalize the cross spectral density matrix of
    %        the aux channels
    % e = eigenvalues of the cross spectral density matrix of the
    %        aux channels
    % ccum = contributions of each eigenvector to the
    %               multi-channel coherence

    %% compute multicoherence and coefficient estimates
    N = size(X,1);
    % compute cross-spectral-densities
    SX = zeros(N,N,npp/2+1);
    SY = zeros(npp/2+1,N);
    fprintf(2, 'Computing CSD matrix\n');
    for i=1:N
        fprintf(2, '\r%d/%d      ', i, N);
        SY(:,i) = cpsd(X(i,:), y, hanning(npp), npp/2, npp, fs);
        for j=i:N
            SX(i,j,:) = cpsd(X(j,:), X(i,:), hanning(npp), npp/2, npp, fs);
            SX(j,i,:) = conj(SX(i,j,:));
        end
    end
    [Syy,fr] = pwelch(y, hanning(npp), npp/2, npp, fs);
```

```
    % solve for coherence and coefficients
    fprintf(2, '\nComputing multicoherence\n');
    c = zeros(npp/2+1,1);
    A = zeros(N, npp/2+1);
    U = zeros(N,N,npp/2+1);
    e = zeros(N,npp/2+1);
    ccum = zeros(N,npp/2);
    for i=1:npp/2+1
        %A(:,i) = squeeze(SX(:,:,i)) \ SY(i,:)';
        A(:,i) = pinv(squeeze(SX(:,:,i))) * SY(i,:)';
        c(i) = (SY(i,:) * A(:,i)) / Syy(i);
        [U(:,:,i), ee] = eig(squeeze(SX(:,:,i)));
        e(:,i) = diag(ee);
        ccum(:,i) = abs(SY(i,:)*squeeze(U(:,:,i))).^2 ./ e(:,i).' / Syy(i);
    end
    % remove small imaginary part due to numerical errors
    c = real(c);
end
```