Developing a method for measuring wing-beat frequency and flight durations of Fig Wasps

Amitabh Shrivastava

December 9, 2013

The fig-fig wasp system presents a wonderful platform for measuring comparative flight characteristics of a community of fig wasps dependant on a single host plant. The primary aim of the study was to characterize wing beat frequencies of all wasp species of the community. The set up also simultaneously allowed for the measurement of flight durations of the wasps. Wing-beat frequency and flight duration are good parameters for estimating dispersal propensities of the different fig wasp species.

Generally, acoustic systems are used for measuring wing-beat frequency of insects. But, for very small insects, acoustic systems are impractical because of low signal to noise ratio. Video recording the wasps at high fps presents challenges due to the the unnaturally high light intensity, high data storage demands, manual data analysis and high uncertainty. The simplest device to measure the wing-beat frequency is a strobe. But, it needs to be controlled manually to match the frequency of the wasp which makes data recording for long periods difficult.

An optical tachometer for measuring wing beat frequency of fig wasps based on the design by Unwin and Ellington (1979) is ideal for measuring real time wing beat frequencies and is presented below. A PC with USB sound-cards is used as data logger and analysis is done through Fourier transform in MATLAB. The associated hardware and software is described.

1 Hardware

Cold anaesthetised fig wasps are tethered on a fine needle at approximately constant angles and are allowed to flap their wings by lifting them off the ground. The wasps are positioned such that the flapping left wing crosses a fine (0.2mm diameter) laser beam during a wing beat. As the wasp flaps its wings, the laser light passing through gets amplitude modulated. This light is intercepted by a photo-transistor where it gets converted to a modulated voltage.
The modulated voltage passes through a high pass filter with a -3dB gain at 22.5Hz. The signal then gets amplified with an op-Amp with a gain of 1800. Finally the signal passes through a low pass filter with a -3dB gain at 300Hz. The signal is then captured with a USB sound-card and gets transferred to a PC.

2 Software

The signal is captured as sound at $44KHz$ sample rate using the software package Cool Edit Pro and saved in .wav format. The data is analysed with MATLAB. The Discrete Fast Fourier

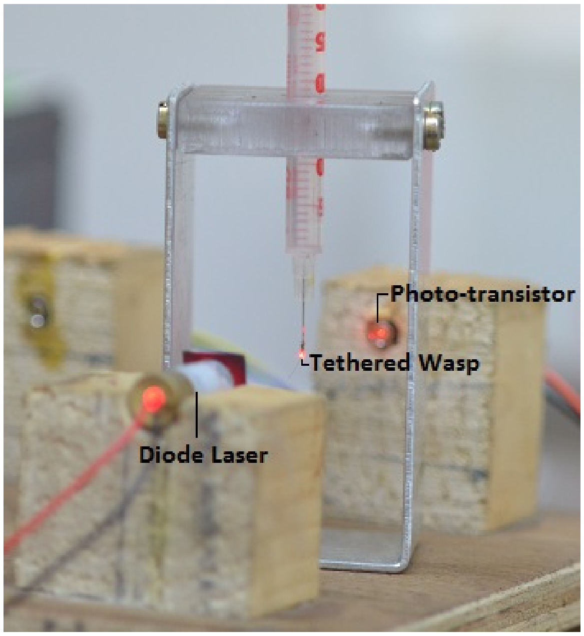

Figure 1: Wing-beat frequency analyser, set up

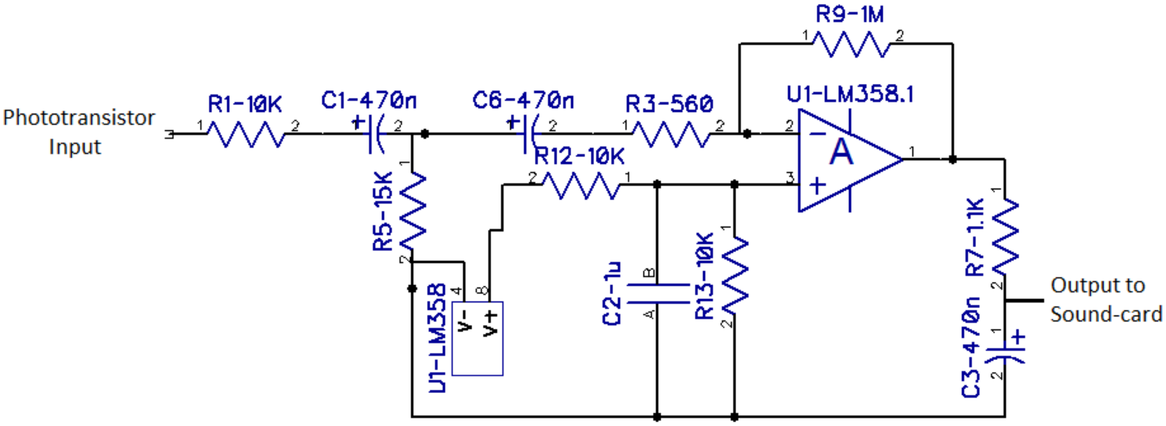

Figure 2: Wing-beat frequency analyser, amplifier circuit

transform is applied to a 44.3*sec* stretch of data and a Gaussian fit around the dominant frequency is used to determine the mean frequency and deviation. The code used to analyse can be found in the supplementary.

3 Sample Data

The wing-beat frequency shows some variability with time, presumably due to fatigue in the muscles. The wing-beat frequency also seems to change on a per beat basis although this change is only of the order of one percent.

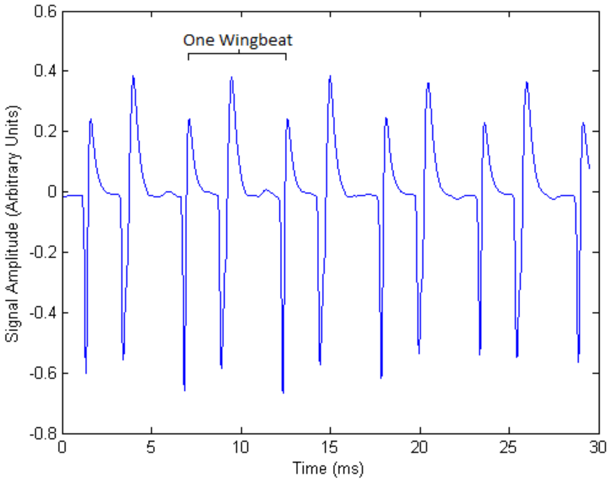

Figure 3: Raw signal for 30 milliseconds of flight

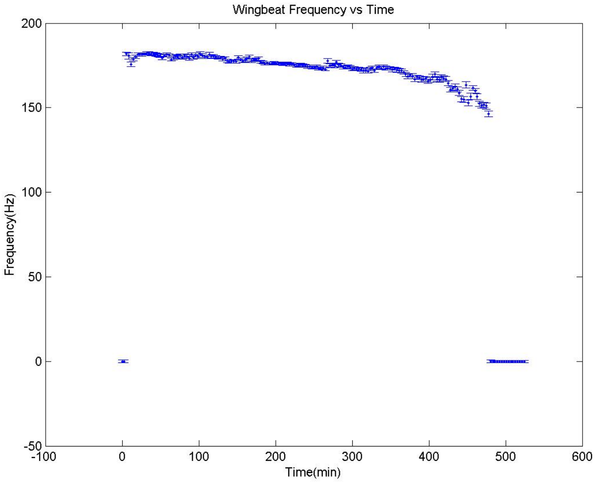

Figure 4: Code output, Frequency(±sd) vs time

4 References

- Fourier Analysis of Wing Beat Signals: Assessing the Effects of Genetic Alterations of Flight Muscle Structure in Diptera Christopher J. Hyatt and David W. Maughan *Biophysical Journal Volume 67 September 1994 1149-1154*
- An optical tachometer for measurement of wing-beat frequency of free-flying insects by D B Unwin and C P Ellington *J. Exp. Bio. 1979*

5 Supplementary

The code for analysing wing-beat frequency:

```
function mean_freq= Wingbeat_Final_2(file, len, fmin, fmax)

if(nargin < 1)
    error('File name is required');
end
if(nargin < 2)
    len=20;
end
if(nargin < 3)
    fmin=150;
end
if(nargin < 4)
    fmax=230;
end

[~,Fs] =wavread(file, [1, 100]);
fprintf('Sample Rate = %d samples/sec\n', Fs);
time_step= 2^len/Fs;
fprintf('Time Resolution = %.1f seconds\n', time_step);
wav_size=wavread(file, 'size');
tot_time=wav_size(1, 1)/Fs/60;
hours=floor(tot_time/60);
minutes=tot_time-hours*60;
fprintf('File Size = %.0f hours %.1f minutes \n', hours(1, 1), minutes);
max_index=floor(wav_size(1, 1)/2^len);

beat_freq=zeros(max_index, 1);
freq_err=zeros(max_index, 1);
time=zeros(max_index, 1);
rsquare_gauss=zeros(max_index, 1);
elapsed_time=zeros(max_index, 1);
amplitude=zeros(max_index, 1);

index_per_hertz=(2^(len-1)+1)/(Fs/2);
index_fmin=fmin*index_per_hertz;
index_fmax=fmax*index_per_hertz;
index_fmin=ceil(index_fmin);
index_fmax=ceil(index_fmax);

F=Fs/2*linspace(0, 1, 2^(len-1)+1);
F=transpose(F);
m=mod(max_index, 200)+1;
str_size=0;

```

```matlab
for i=0:(max_index-1)
    t=cputime;
    k=i+1;
    if(mod(k, m) && i)
        while(str_size)
            fprintf('\b');
            str_size=str_size-1;
        end
        str=strcat('Progress :', num2str(ceil(k/max_index*100)),'%% ETC= ',  num2str(ceil((max_index-k)*e)));
        str_size=fprintf(str);
    end
    time(k, 1)=i;
    dat=wavread(file, [2^len*i+1, 2^len*k]);
    Y=fft(dat, 2^len)/2^len;
    dat=abs(dat);
    dat_mean=mean(dat);
    amplitude(k, 1)=dat_mean;
    X=2*abs(Y(1:2^(len-1)+1));

    [c, index_fpeak]=max(X(index_fmin:index_fmax, 1));
    index_flow=index_fpeak-10*index_per_hertz;
    index_fhigh=index_fpeak+10*index_per_hertz;
    index_flow=floor(index_flow);
    index_fhigh=ceil(index_fhigh);
    index_flow=index_flow+index_fmin-1;
    index_fhigh=index_fhigh+index_fmin-1;
    [a, b]=fit(F(index_flow:index_fhigh, 1), X(index_flow:index_fhigh, 1), 'gauss1');

    beat_freq(k, 1)=a.b1;
    freq_err(k, 1)=sqrt(a.c1/2);
    rsquare_gauss(k, 1)=b.rsquare;

    elapsed_time(k, 1)=cputime-t;
    e=mean(elapsed_time(1:k, 1));
end
fprintf(' ...Done\n\n');
time=time.*(2^len/Fs)/60;

mean_freq=0;
ontime_index=0;
for i=1:max_index
    if(amplitude(i, 1)<0.01)
        beat_freq(i, 1)=0;
    else
        mean_freq= mean_freq+beat_freq(i, 1);
        ontime_index=ontime_index+1;
    end
end

file_size=size(file);
file(file_size(2))='t';
file_size(2)=file_size(2)-1;
file(file_size(2))='x';
file_size(2)=file_size(2)-1;
file(file_size(2))='t';
save(file, 'beat_freq', 'freq_err', 'amplitude', 'rsquare_gauss');

```

```
mean_freq=mean_freq/ontime_index;
fprintf('Mean Frequency = %.2f Hz\n', mean_freq);
ontime=ontime_index*time_step/60;
fprintf('Flight Time = %.1f min \n', ontime);
endurance=ontime*60*mean_freq;
fprintf('Endurance = %.0f \n', endurance);

figure1 =figure;
errorbar(time, beat_freq, freq_err, '.');
title('Wingbeat Frequency vs Time');
xlabel('Time(min)');
ylabel('Frequency(Hz)');

file_size=size(file);
file(file_size(2))='g';
file_size(2)=file_size(2)-1;
file(file_size(2))='p';
file_size(2)=file_size(2)-1;
file(file_size(2))='j';
saveas(figure1, file);
```