LASER INTERFEROMETER GRAVITATIONAL WAVE OBSERVATORY

LIGO Laboratory / LIGO Scientific Collaboration

LIGO- T080096-00-K *ADVANCED LIGO* 12 May 2008

Wire attachment points and flexure corrections

Mark Barton, Norna Robertson

Distribution of this document:
DCC

This is an internal working note
of the LIGO Project.

California Institute of Technology
LIGO Project – MS 18-34
1200 E. California Blvd.
Pasadena, CA 91125
Phone (626) 395-2129
Fax (626) 304-9834
E-mail: info@ligo.caltech.edu

LIGO Hanford Observatory
P.O. Box 1970
Mail Stop S9-02
Richland WA 99352
Phone 509-372-8106
Fax 509-372-8137

Institute for Gravitational Research
University of Glasgow
Kelvin Building
Glasgow G12 8QQ
Phone: +44 (0)141 330 3340
Fax: +44 (0)141 330 6833
Web: www.physics.gla.ac.uk/igr

Massachusetts Institute of Technology
LIGO Project – NW22-295
185 Albany St
Cambridge, MA 02139
Phone (617) 253-4824
Fax (617) 253-7014
E-mail: info@ligo.mit.edu

LIGO Livingston Observatory
P.O. Box 940
Livingston, LA 70754
Phone 225-686-3100
Fax 225-686-7189

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

Table of Contents

1 Introduction

1.1 Purpose and Scope

Terminology for specifying wire attachment points has evolved with increased understanding of the relevant physics and more sophisticated suspension models. This document outlines the history and recommends standard usage for the future.

1.2 Version History

4/16/08: Pre-rev-00 draft #1.

4/21/08: Pre-rev-00 draft #2.

4/30/08: Pre-rev-00 draft #3. Input from Norna, new references, new section on blades.

5/08/08: Pre-rev-00 draft #4. Improved figures, clarified discussion of Cagnoli et al. 2000.

5/12/08: Rev-00. Revised recommended terminology per suggestion of Justin, added section on realistic ribbon/fibre necks.

1.3 References

T010103-05, Advanced LIGO Suspension System Conceptual Design, N.A. Robertson et al.

G. Cagnoli et al., Phys. Lett. A, 272: 39 – 45, 2000.

Calum Torrie, PhD Thesis, 1999, University of Glasgow. On-line at

http://www.physics.gla.ac.uk/~ctorrie/ALIGO/C_Torrie_thesis.pdf .

P000039-00, Matthew Husman, PhD Thesis, 1999, University of Glasgow.

T020205-02, Models of the Advanced LIGO Suspensions in Mathematica™, M.A. Barton.

T020011-00, Suspension model comparisons, M.A. Barton.

T070101-00, Dissipation dilution, M.A. Barton.

T050255-07, Investigation into blade torsion, blade lateral flexibility, and the effect they have on blade and wire performance, I. Wilmut et al.

T050172-00 , Modifications to Quad Controls Prototype to fix the "d's", C.I. Torrie.

T050257-01, Quad Pendulum Controls Prototype - Identification of Modes, M.A. Barton

T070138-01, Ribbon/Fibre Length Budget, M.A. Barton et al.

Ultralow frequency oscillator using a pendulum with crossed suspension wires. M.A. Barton and K. Kuroda, Rev. Sci. Instrum. 65(12): 3775, 1994.

A low-frequency vibration isolation table using multiple crossed-wire suspensions. M.A. Barton et al., Rev. Sci. Instrum. 67(11): 3994, November 1996

T050255-07, Investigation into blade torsion, blade lateral flexibility, and the effect they have on blade and wire performance. I. Wilmut, J. Greenhalgh, N.A. Robertson, M.A. Barton, C.I. Torrie

2 History

2.1 Models

There have been three different families of models for GEO/AdvLIGO-style pendulums, which are nowadays known by the respective computer language used.

For his PhD thesis on the GEO triple pendulum, Calum Torrie developed a state-space model in Matlab using hand-derived matrix elements for the A, B, C and D matrices. It assumed a symmetrical pendulum and wires with no bending or torsional stiffness, and did not model the blades directly but allowed for them by adding their compliance in series with that of the associated wires. (This was tantamount to assuming that the blades were oriented with their working direction in line with the wires, which was not exactly the case but a quite usable approximation.) This triple model was later extended for the quad suspension proposed for AdvLIGO.

At about the same time and also as research for a PhD, Matt Husman developed a more sophisticated model in Maple using the symbolic algebra features to include more of the physics than was feasible with hand derivation. The Maple model included explicit cantilever blades, wires with bending and torsional (as well as longitudinal) elasticity, an option for asymmetry, and arbitrary frequency-dependent loss on the wire bending (which allowed for the calculation of thermal noise).

Somewhat earlier, Mark Barton had developed a model in Mathematica of the "X-pendulum", a device developed as a 2D vibration isolation system for possible use in TAMA (Barton et. al, 1994, 1996). Then, somewhat later than the Torrie/Husman work, he recycled much of this code to create a Mathematica model of the AdvLIGO quad. Initially this was as a double-check on the Matlab quad model, which was still in regular use, but it was extended to include all the features of the Maple model, which was no longer being maintained. Because the Mathematica is structured as a model of a specific pendulum on top of a general-purpose pendulum modeling toolkit, it has been used to produce many other models including a triple, a LIGO-I-style simple pendulum and assorted toy models, as well as more complicated quad models with blade lateral compliance, dumbbell fibres, etc.

Because the Matlab model is still in frequent use, being familiar to key personnel, faster to run, and integrated with Simulink, it has been given regular updates. As part of the original validation of the Mathematica, the Matlab matrix elements were rederived symbolically within Mathematica, and it was established that the Matlab triple was exactly correct and a few minor errors had crept into the extension to a quad (T020011-00). Subsequently the core of the Matlab model was replaced by Matlab code generated programmatically from the Mathematica model, and this has allowed many advanced features to be transplanted, including blades oriented vertically (rather than in line with the wires), blades with lateral compliance, wire bending elasticity, non-diagonal MOI tensors, etc.

2.2 Nomenclature

The standard nomenclature for parameters of the AdvLIGO triple and quad suspensions is based on the variable names in the Matlab model. (The Maple names are somewhat different.) With a few minor exceptions to do with the elasticity of blades and wires, the Mathematica model copied this exactly. (The blade elasticities are `kcn`, `kn1` and `kc2` in the Matlab but `kbuz`, `kbiz` and `kblz` in the Mathematica. The wire elasticities, `kwn`, `kw1`, `kw2` and `kw3`, are specified per side in the Matlab but per wire in the Mathematica.) The ones of particular interest for the present purposes are shown in Figure 1, which is taken from T010103-05. The parameters `dn`, `dm`, `d0`, `d1`, `d2`, `d3` and `d4` specify the vertical positions of the endpoints of the wires relative to the centres of mass of the respective masses (modulo fineprint discussed below). Collectively these are known as the `d`'s. Similarly the `n`'s give the horizontal positions (in the transverse plane).

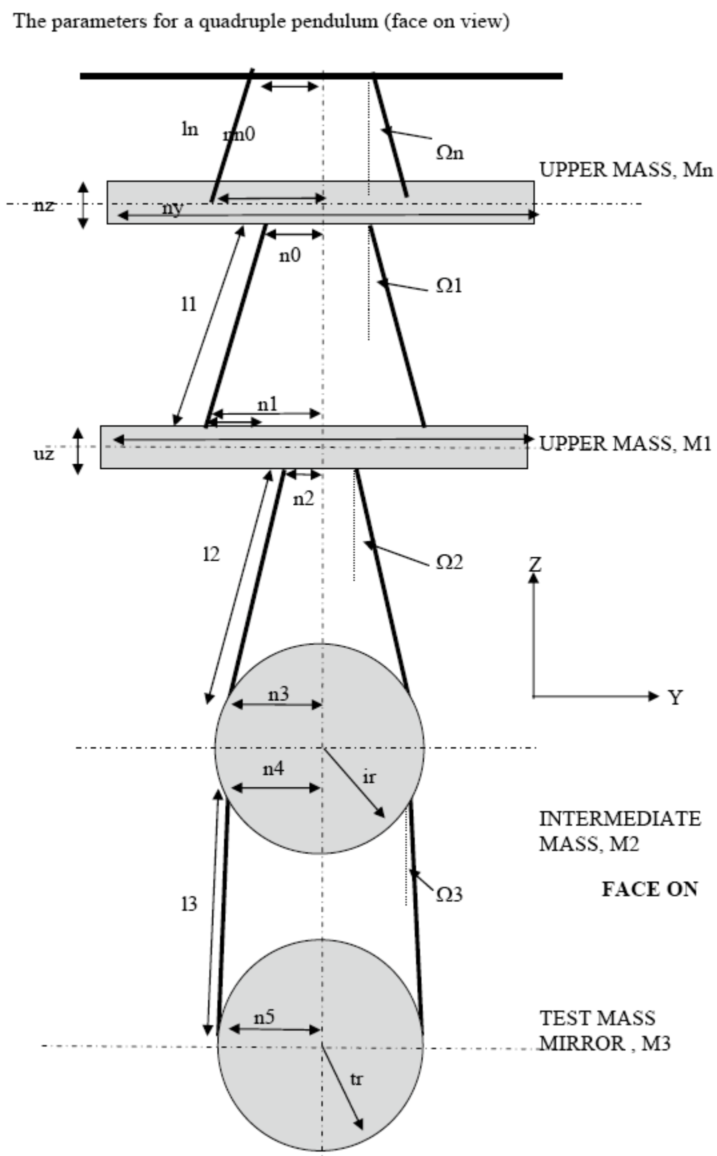

Figure 1: Nomenclature, from C2.3 of T010103-05

As just alluded to, there is a subtlety in the interpretation of the "`d`" parameters (and to a lesser extent the `n`'s) to do with the flexure of the wire. In the Matlab model as derived, the wire is taken to be infinitely flexible with respect to bending, so that the effective flexure point is the same as the attachment point. The conceptual designs of pendulums in successive editions of T010103 are specified in terms of parameter sets to the Matlab code, so in some sense they have adopted this assumption.

However it has also long been appreciated that a real wire subject to a lateral force curves gradually as it emerges the clamp, and that the effective flexure point is some distance away – a distance that can be significant compared to the nominal d values. Specifically, a real wire near the clamp has an exponential shape given by (Cagnoli et al. Eq. 1 with modified notation adapted for the case of a pendulum; see Figure 2)

$$x(z) = \frac{F}{T}\left(z + ae^{-z/a} - a\right) \tag{1}$$

where z is vertical distance (positive down), $x(z)$ is lateral displacement, the wire is clamped at $x = z = 0$, $T = mg$ is a vertical tension, F is a horizontal force, and

$$a = \sqrt{\frac{YI}{T}}$$

(where Y is Young's modulus and I is the moment of area in the bending direction) is a characteristic length, the "flexure length".

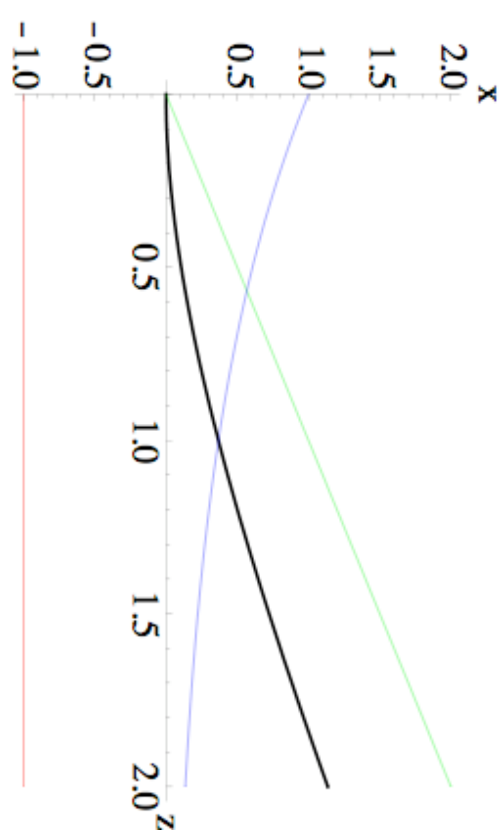

Figure 2: Shape of a bent wire, $x(z)$ as given by Eq. 1 for $a = 1$ (black curve, with individual terms in colours: constant – red; linear – green; exponential - blue)

For example, in the conceptual design of T010103-05, the quoted `d`'s are all 1 mm but the flexure lengths are (from the top down) 4.1, 2.7, 2.8 and 0.3 mm, i.e., rather larger in all but one case. This is important because the frequency of the fundamental pitch mode is a sensitive function of the `d`'s, and if no allowance is made the pendulum will be too stiff in pitch, or worse, if too much allowance is made, the pendulum will be unstable. Such corrections have traditionally been left implicit in the mathematical model and are made at the detailed mechanical design stage as described in the next section. However the Mathematica model upset this convention by interpreting the `d`'s as wire attachment points pure and simple and treating the flexure correction as something to be done explicitly in the model. Recent updates to the Matlab allow for stiff wire and adopt the new convention if a switch `pend.stage2` is defined and non-zero. (The name of the switch could perhaps have been better chosen – it refers somewhat obscurely to the stage of the calculation in the Mathematica where the wire stiffness potential terms are added in.)

2.3 Flexure length and effective flexure point

There are five important ways in which effective flexure point could be defined. For *four* of those ways, it turns out that the effective flexure point is exactly one flexure length a from the clamp:

1. The point at which a line straight out from the jaws of the clamp intersects the line of the asymptotically straight section of the wire.

2, 3, and 4. The points where an infinitely flexible wire would have to be terminated to give the same frequency for pendulum, pitch and violin modes.

Bizarrely however, a seemingly equivalent fifth definition describes a point only $a/2$ from the clamp (see Figure 3):

5. The centre of curvature of the locus of the end of the wire as it swings.

Definition 5 will be important later for the calculation of energy flow and dissipation dilution. However we set it aside for the moment and focus on definition 3 in terms of the pitch mode.

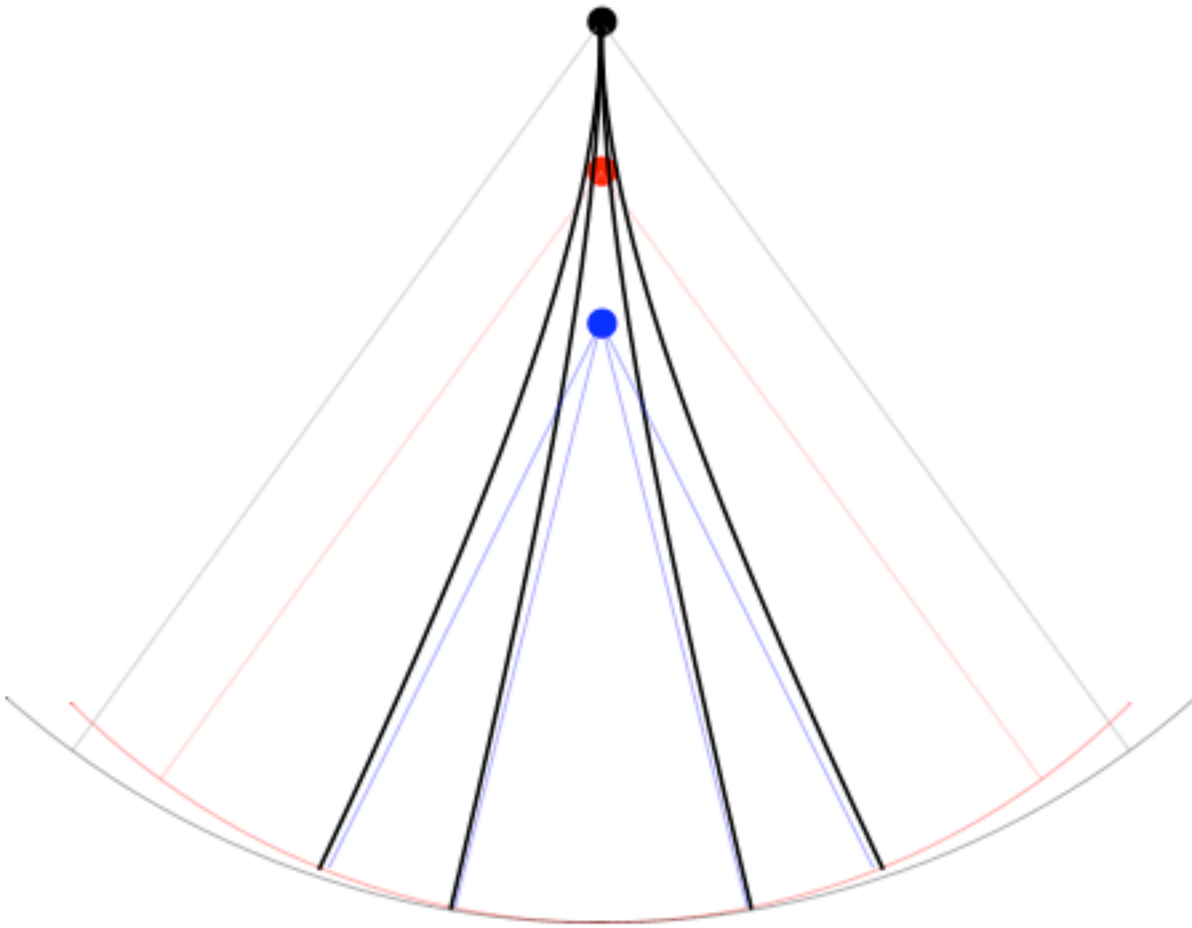

Figure 3: Relationship of effective flexure points according to different definitions. The black dot is the wire clamp and the thick black lines are the wire in different positions. The blue dot at distance *a* is the standard flexure point defined by the asymptotes to the straight section of the wire (blue lines). The red dot at *a*/2 is the alternative flexure point defined as the centre of curvature of the locus of the wire endpoint (red arc). The thin black lines and the black arc are the wire positions and mass locus for an infinitely flexible wire of the same length.

2.4 Flexure length and flexure correction

To apply the results of the previous section to implementing a conceptual design with real wire is then simply a matter of moving each wire attachment point along the direction of the wire away from the centre by an amount a. (The length of the wire then needs to be increased by $2a$ to compensate, and keep the heights of the hanging masses the same.) In principle, for a diagonal wire at θ to the vertical, this should be a diagonal offset of $-a\cos\theta$ in the associated `d`'s and $a\sin\theta$ in the associated `n`'s – see Figure 4. However since the mode frequencies are very sensitive to the `d`'s but hardly at all to the `n`'s, only the `d` correction is worth applying or applied in practice. (The full correction is used internally in recent versions of the Matlab model to derive the mode frequencies for stiff wire from matrix elements derived for flexible wire and the agreement with the Mathematica is essentially exact: four figures or more of correspondence in the frequencies and mode shapes.)

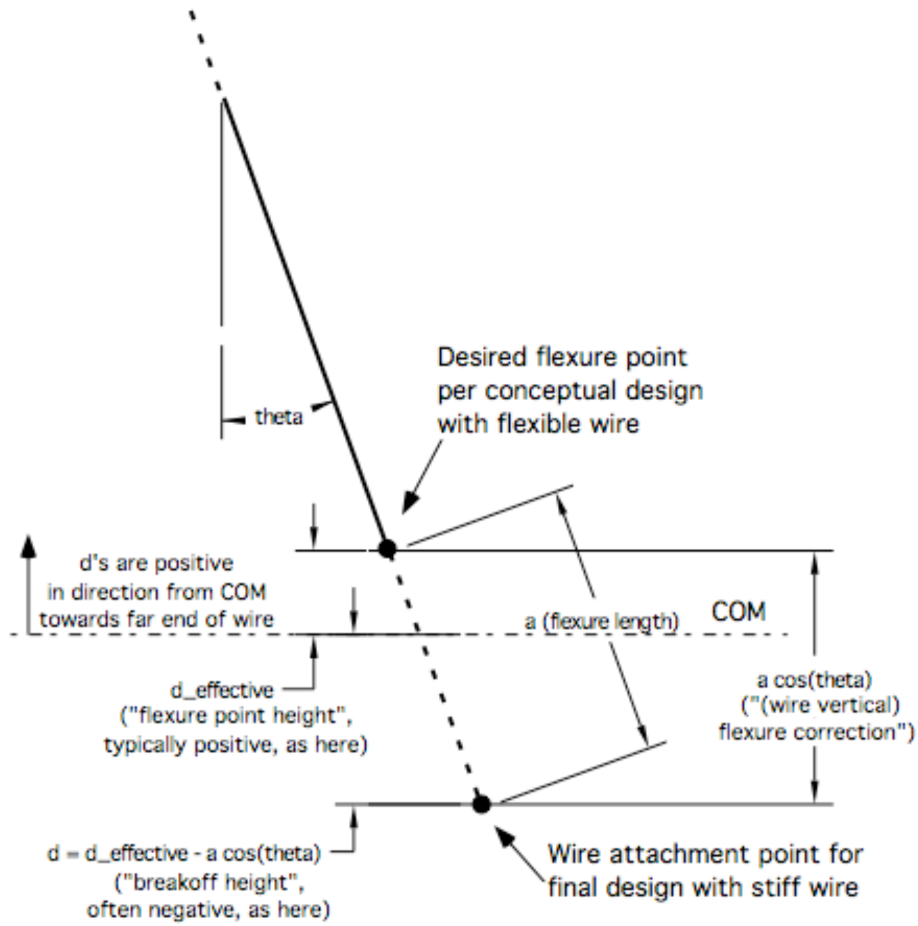

Figure 4: Geometry of ideal flexure correction (in practice the horizontal offset is not worth bothering with, since θ is typically small and frequencies are insensitive to the n's).

2.5 Flexure length and dissipation dilution

As is well known, one of the main reasons for using a pendulum as the final stage in the suspension is to exploit the phenomenon of dissipation dilution to reduce thermal noise. If the final stage used only a horizontal spring, then the mechanical loss would be equal to the material loss angle of the spring material, which even for fused silica would not meet AdvLIGO requirements. However in a pendulum, the majority of the restoring force is gravitational and the material loss angle ϕ is "diluted" by the ratio of elastic energy at the pivot to total energy, typically an improvement of from several hundred to several thousand.

There is some subtlety in how this applies in practice. As noted above, in terms of frequency, a simple stiff-wire pendulum of length l is equivalent to a flexible-wire pendulum of length $l ! a$. That is, considered as a horizontal oscillator it has extra restoring force of

$$k_{stiff} = \frac{T}{l-a} \simeq \frac{T}{l} + a\frac{T}{l^2} = k_{flex} + a\frac{T}{l^2}$$

relative to a flexible-wire pendulum of the same length (i.e., l). However, integrating the elastic energy of the wire over the flexure region (see Cagnoli et al., 2000) accounts for only half the above value, i.e., $aT/2l^2$. The other half of the energy comes from the fact that, due to a "yo-yo" or "capstan" effect that changes the effective length of the pendulum by a second order amount as it swings, the locus of the mass has a radius of curvature of $l - a/2$. That is, the mass rides slightly higher at the extremities of the swing than for a flexible-wire pendulum of length l, giving extra gravitational restoring force of the other $aT/2l^2$.[1]

Note also that the difference between definitions 5 and 2 is the resolution of the mystery as to how energy loss at the pivot can damp the mass when the wire is seemingly at right angles to the motion of the bob – in fact the straight section of the wire *isn't* at right angles to the locus, but angled very slightly in towards the centre.

2.6 Blade lateral compliance

It turns out there is another effect that needs to be considered when tuning the fundamental pitch mode, which originates in the blades rather than the wires but mimics a wire flexure effect and is also best compensated for by adjusting the wire attachment points: the compliance of the blade springs in the "lateral" direction (at right angles to the length of the blade). This arises because, as the top masses tilt in the course of the pitch mode, the force from the wires is no longer purely in the working direction, but has a lateral component. If the blade has compliance in this direction the tip moves to the side, which increases the lever arm from wire tension to pitch and increases the pitch, possibly to the point of instability.

Justin Greenhalgh pointed out that if the blade moves a distance x to the side under a lateral force F, it is as if the wire flexure length was less by an amount h such that

$$h = \frac{Wx}{F} = \frac{W}{k_{lateral}}$$

where W is the weight supported and $k_{lateral}$ is the spring constant in the lateral direction. See Figure 5, reproduced from T050255-07 (p. 10). According to finite-element analysis, typical quad blade springs have lateral stiffnesses only about 30-50 times that in the working direction, which turns out to imply that about 5 mm *less* wire flexure correction needed to be applied at the `dn` and

[1] Note however that while Cagnoli et al. correctly identified this extra potential term in the context of an attempt to resolve an apparent error in the use of complex elastic constants to calculate damping, it's purely coincidental that the problem appears to be resolved. The complex elastic constant method attributes half the correct damping to the bending of the wire, and the remaining half to the new gravitational term. This is spurious – all the damping is due to the bending and none to the gravitational term. More generally, the complex elastic constant method is unreliable for problems with static tensions, and it is easy to give examples (e.g., violin modes) where it gives totally the wrong answer. See T070101-00.

d1 levels (the top blade springs don't contribute because the support structure doesn't rotate). This was enough to capsize the first build of the controls prototype, which had been designed without knowledge of this effect. The controls prototype was later gotten to work (T050257-01), but in an ad hoc manner (T050172-00) not as described above. The noise prototype, which took account of the blades from the beginning, behaved exactly as expected.

The effect of the blades on other modes is not exactly the same as for the fundamental pitch mode but versions of the Matlab and Mathematica models that take it into account have been developed.

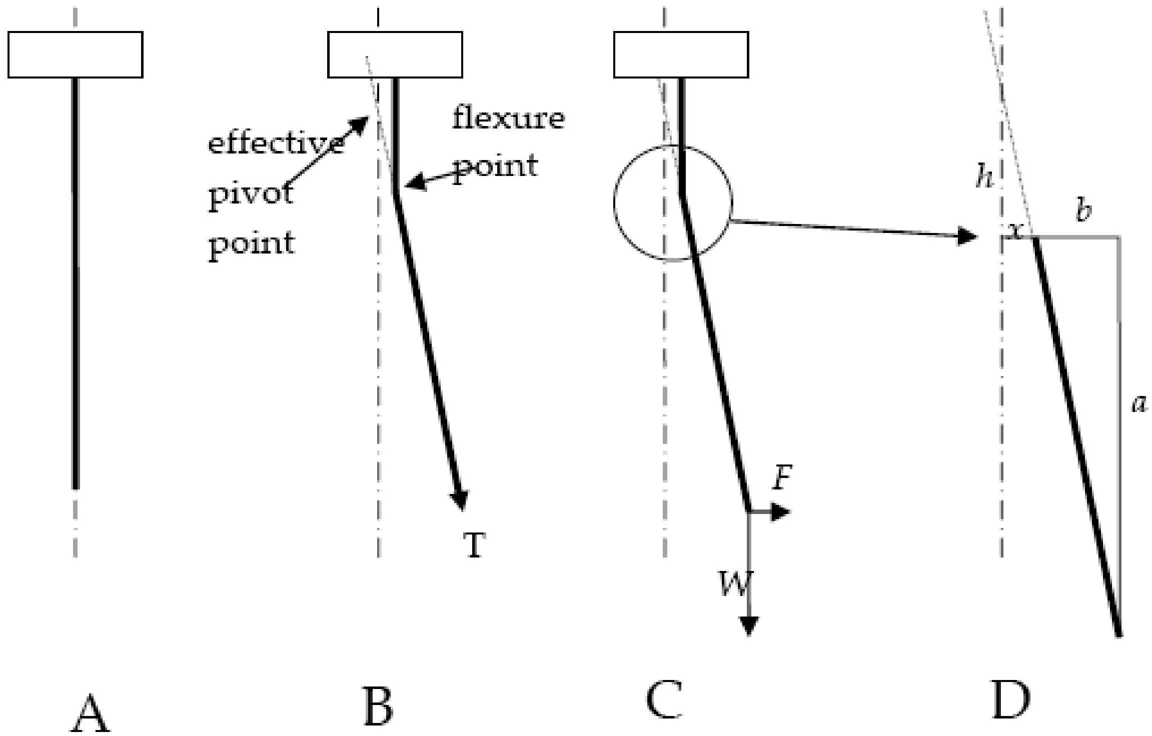

Figure 5: From T050255-07, effect of blade lateral compliance and its similarity to a wire flexure effect

3 Conclusion

3.1 Interim recommendations

We make the following recommendations for the future:

- Any unqualified references to d values should be understood as referring to the heights of actual (physical) breakoff positions relative to the centre of mass. These can also be called breakoff heights.
- Any designs specified in terms of a flexible-wire equivalent model should refer to "effective" d's or flexure point heights.

- "Flexure length" should be used and understood to be a distance along the wire, i.e., diagonally if the wire is slanted. "(Wire vertical) flexure correction" is the vertical component of flexure length".
- The sign convention for all the above quantities is positive from the COM towards the far end of the wire, i.e., up for `d0`, `d2` and `d4`, and down for `dn`, `d1` and `d3`.
- The Matlab model should always be run with the switch `pend.stage2=1`, and any flexure correction should be done explicitly with code like the following fragment, which assumes all effective `d`'s are to be `0.001` (i.e., 1 mm) and adds the appropriate corrections to derive the actual `d`'s to be stored in the `pend` data structure describing the pendulum, `pend.dm`, `pend.dn`, etc.

```
pend.stage2 = 1;
cn = sqrt(pend.ln^2-(pend.nn1-pend.nn0)^2)/pend.ln;
c1 = sqrt(pend.l1^2-(pend.n1-pend.n0)^2)/pend.l1;
c2 = sqrt(pend.l2^2-(pend.n3-pend.n2)^2)/pend.l2;
c3 = sqrt(pend.l3^2-(pend.n5-pend.n4)^2)/pend.l3;
Mn1 = (1/4)*pi*pend.rn^4;
M11 = (1/4)*pi*pend.r1^4;
M21 = (1/4)*pi*pend.r2^4;
M31 = (1/4)*pi*pend.r3^4;
flexn =
sqrt(pend.nwn*Mn1*pend.Yn/(pend.mn+pend.m1+pend.m2+pend.m3)/pend.g)*c1^(3
/2);
flex1 =
sqrt(pend.nw1*M11*pend.Y1/(pend.m1+pend.m2+pend.m3)/pend.g)*c1^(3/2);
flex2 = sqrt(pend.nw2*M21*pend.Y2/(pend.m2+pend.m3)/pend.g)*c2^(3/2);
flex3 = sqrt(pend.nw3*M31*pend.Y3/pend.m3/pend.g)*c3^(3/2);
pend.dm = 0.001-flexn;
pend.dn = 0.001-flex1+pend.g*(pend.m1+pend.m2+pend.m3)/(2*pend.kx1);
pend.d0 = 0.001-flex1;
pend.d1 = 0.001-flex2+pend.g*(pend.m2+pend.m3)/(2*pend.kx2);
pend.d2 = 0.001-flex2;
pend.d3 = 0.001-flex3;
pend.d4 = 0.001-flex3;
```

Note: the above code is for a quad model with (round) fibres and also illustrates the adjustment of `dn` and `d1` to compensate for the destabilizing effect of lateral compliance of the blades – `pend.kx1` and `pend.kx2` are the lateral stiffness values.

- Any Mathematica models should also have flexure corrections applied explicitly with code analogous to the above in the `overrides` list, and the results after `Stage2` of the calculation should be used.

3.2 Note on neck shapes

All of the preceding will need to be revisited when more is known about the behaviour of ribbons/fibres with realistic neck shapes. Alan Cumming and Rahul Kumar have developed techniques for taking measured ribbon/fibre profile data, creating an ANSYS finite-element model and determining the standard flexure point (i.e., the one in terms of the wire asymptotic direction). Their calculations were used to establish the ear positions for the noise prototype (see T070138-01). As in the uniform wire/fibre/ribbon case the standard flexure point should be the point of equivalence with the flexible wire equivalent system because it is the point around which the body of the wire can transmit no torque.

The endpoint locus centre of curvature flexure point has not been studied for realistic necks yet, and its relationship to the standard flexure point is not obvious except that the neat 1:2 ratio of distances from the nominal attachment point is unlikely to continue to hold. (This is not least because the attachment point is somewhat arbitrary for a fibre welded to a less than infinitely rigid ear.) However it is likely to be of interest as a part of a scheme for simply parameterizing neck shapes in the Matlab and Mathematica models, because the distance between it and the standard flexure point will be a measure of the component of the neck energy which is *not* gravitational, and so contributes damping. Stay tuned for revisions of this document and of the Matlab and Mathematica models.