



2016-02-11

The Bambow: A Bamboo-Cored Carbon-Fiber Composite Bow

Ezekiel G. Merriam

Brigham Young University - Provo, ezekieltovarish@gmail.com

Follow this and additional works at: <https://scholarsarchive.byu.edu/studentpub>

 Part of the [Mechanical Engineering Commons](#)

BYU ScholarsArchive Citation

Merriam, Ezekiel G., "The Bambow: A Bamboo-Cored Carbon-Fiber Composite Bow" (2016). *All Student Publications*. 148.
<https://scholarsarchive.byu.edu/studentpub/148>

This Class Project or Paper is brought to you for free and open access by BYU ScholarsArchive. It has been accepted for inclusion in All Student Publications by an authorized administrator of BYU ScholarsArchive. For more information, please contact scholarsarchive@byu.edu, ellen_amatangelo@byu.edu.

THE BAMBOW: A BAMBOO-CORED CARBON-FIBER COMPOSITE BOW

Ezekiel G. Merriam*

Dept. of Mechanical Engineering
Brigham Young University
Provo, UT 84602

ABSTRACT

Archery is a sport that has benefited from thousands of years of development. Starting with primitive stick-bows, ancient people found they could propel projectiles larger distances and with more effectiveness than by hand or with other mechanical aids. Over the centuries different cultures adopted and advanced different aspects of the technology, including materials, geometry, and manufacturing methods. In the Mechanical Engineering Department at BYU, Nephi's Steel Bow Competition is held annually to challenge engineering students to test their knowledge of mechanics and materials. As a past participant in this competition, I felt building an advanced composite bow would be an ideal class project to fulfill the requirements of MFG 555. I won the Grand Prize with my BamBow, with a distance of 67 yards. This report details my design and fabrication efforts.

1 Background

A bow is one of the most ancient examples of a compliant mechanism. A compliant mechanism is one that elastically deforms to provide motion and store and release energy. A bow functions by storing energy in the bow's limbs, then rapidly converting that potential energy into kinetic energy as it launches an arrow. In the Nephi's Steel Bow Competition the bows are judged by the distance they are able to fire an arrow. Therefore, the arrow must have as high an initial velocity as possible. In turn, this indicates that the bow's primary function is to store as much energy as possible and release it into the arrow as efficiently as possible.

The amount of energy a bow stores is primarily governed

by the bow's shape. It is determined by first generating a graph where the draw distance is indicated on the x-axis and the draw force is indicated on the y-axis. The area under the curve is the energy stored in the bow. Stated more formally:

$$E = \int_0^D F(x)dx \quad (1)$$

where D is the maximum draw distance, F is the draw force (a function of draw distance), and E is the total energy stored. Part of the design effort is dedicated to maximizing E , subject to constraints on F_{\max} and D . Examples of various force-draw curves are shown in Figure 1.

The efficiency with which the stored energy E is converted to kinetic energy (and thus, arrow speed) is known as dynamic efficiency. Dynamic efficiency is defined as the ratio of arrow kinetic energy to potential energy stored in the bow. Dynamic efficiency is a function of arrow mass and shape, and the mass of the bow limbs and its damping characteristics. For our purposes arrow mass and shape are fixed. Bow materials and geometry must be chosen to minimize mass and damping in the bow limbs.

Finally, other factors that will affect bow performance are the workmanship of its construction and design features that may aid in the smooth launch of the arrow.

2 Design

This section discusses the methods used to obtain a highly efficient, high performance design. The force-draw curve of the final design is shown in Figure 2. All code referenced is included

*The author may be contacted with questions at ezekieltovarish@gmail.com

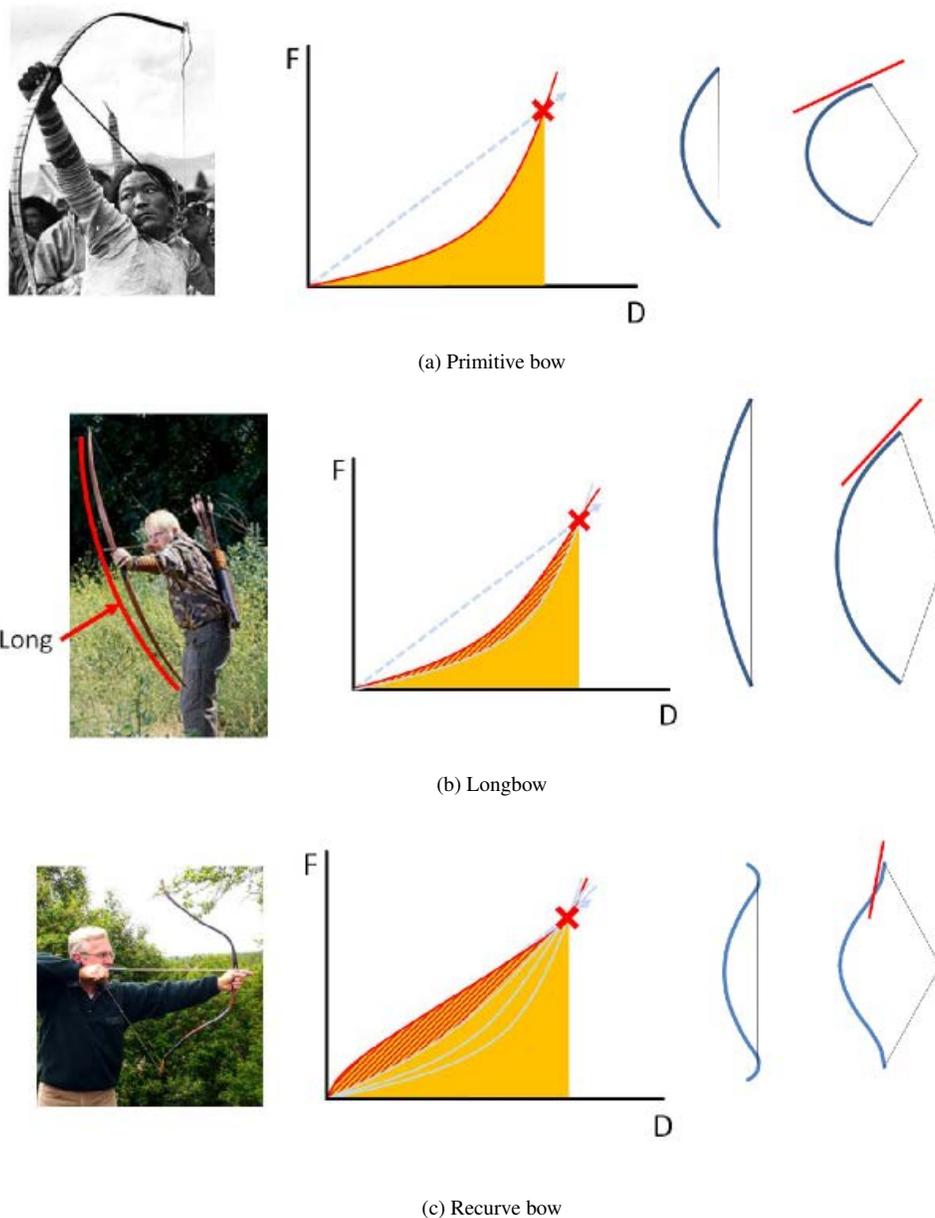


Figure 1: Force-draw curves of various bow topologies. Images courtesy of Kim and Kim of the Korea National Sport University.

in the Appendix.

Material failure criteria were not considered in the design for three reasons. Firstly, no data was available for the pre-preg materials I had available (even if it was available the material is so old and outdated that any data would be overly optimistic). Secondly, observed failures were dominated by manufacturing imperfections, so failure avoidance focused on eliminating these imperfections. Finally, carbon fiber is commonly used in commercially produced bows, so its strength should be adequate for

this application as long as limb dimensions do not deviate far from the norm.

2.1 Material Selection

Carbon fiber is widely used in the sporting goods industry and in the archery industry in particular. Its specific stiffness is unparalleled and so it can be used to make a high draw-weight bow very efficiently. Less-costly fiberglass is also widely used

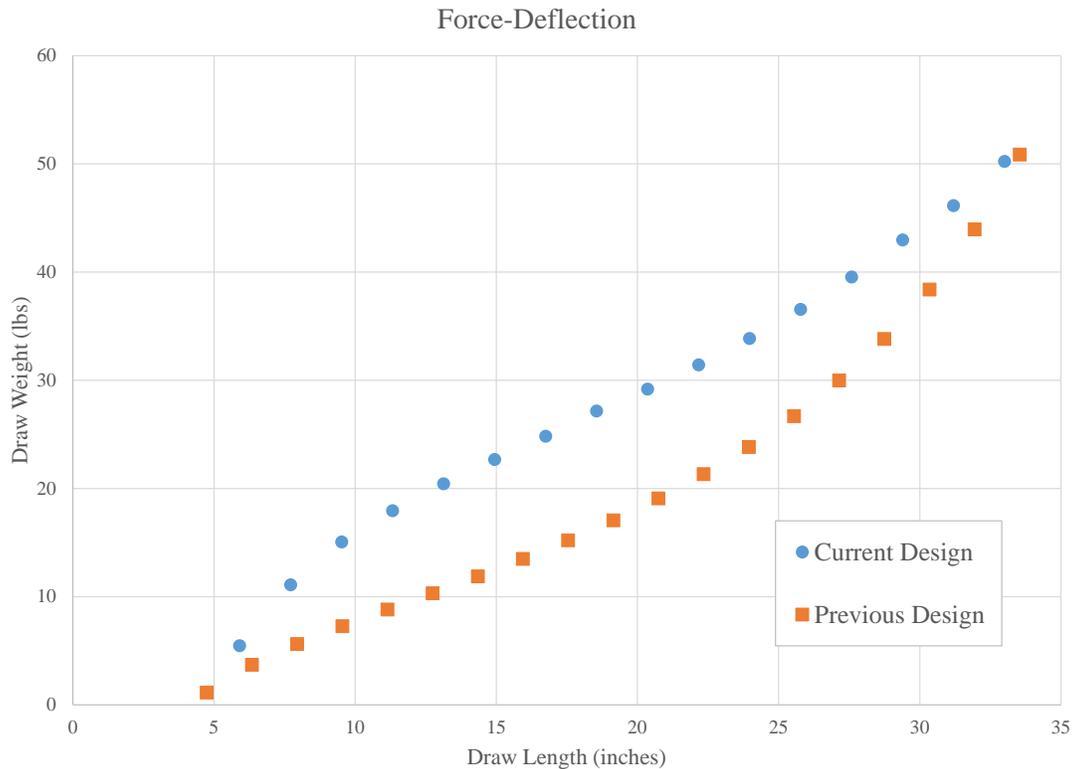


Figure 2: The force-draw curve of the final design. The entry labeled "Previous Design" is the force-draw curve from the design I used last year. It consisted of pultruded fiberglass machined down to prescribed dimensions. However, its shape was not highly optimized. Note the difference in the area under each curve.

in bow construction, but is heavier. In our specific situation, pre-preg carbon fiber provides excellent performance with zero out-of-pocket expense to the student. Therefore, carbon fiber pre-preg was chosen as the primary load-carrying material in this design.

Bending loads are primarily supported by the portion of a beam that is farthest from the neutral plane of bending. Therefore, it makes sense to try to concentrate our load-carrying carbon fibers as far from the neutral plane as possible. This is the principle behind sandwich panel composites, except that in this application we want to still be able to undergo high strains. Therefore, it was decided to use some kind of core material sandwiched with carbon fiber to reduce limb mass and increase dynamic efficiency.

When considering core materials, several are worth mentioning. Firstly, foam is available with extremely low density, but bonding carbon fiber to open-cell foam is problematic at best and a manufacturing nightmare at worst. Balsa wood is likewise of low density, but is fairly expensive and has very low strength (even though the bending load carried by the core is low, it still must support high shear loads). Bamboo is a natural composite

with low density and high strength. It can be readily bonded with epoxies such as those used in carbon fiber pre-preg materials.

2.2 Geometry Optimization

Finite element modeling was done using commercial software (ANSYS). SHELL91 elements (nonlinear layered structural solids) were used to model the composite layup. The model was parameterized to allow a variable layup, limb length, limb curvature, riser (handle) length, limb width at the root, limb width at the tip, and string length. The assumed topology was a simple reflex (curvature away from the archer while unstrung). This topology was chosen because it has the ability to store large amounts of energy and is relatively simple to analyze. Additionally, historical examples of this topology include Korean and Turkish bows, with reflex so extreme that the ends of the bow almost touch when unstrung. Such bows are purported to have shot special flight arrows in excess of 500 yards in antiquity, though modern sources report a record of 434 meters in 1910. See Figure 3 for an example of this.

The geometry and layup are generated by executing an ANSYS script. Only a single limb is modeled, and symmetry is



Figure 3: An example of a Persian bow exhibiting very extreme limb curvature. Such bows were typical of several ancient cultures, including among the Persians, Koreans, and Turks. Image courtesy of Worldantiques Antiques (Pinterest).

exploited to reduce computation time. The limb is split into four sections along its length to better tailor both the limb thickness and layup at different sections. All nodes at the root of the limb are fixed in all degrees of freedom. All nodes at the tip of the limb are linked together with a constraint element, MPC184. This ensures that the string load is distributed among several elements to avoid high local strains. A node that represents the center of the string is then meshed with the nodes of the limb tip using a LINK180 tension-only element. This LINK180 element represents the string. This configuration is shown in Figure 4a.

To “string” the bow, x-displacement constraints are progressively applied to the bow tip and the string center until it has reached the brace height (its strung position before additional displacement is applied). This is shown in Figure 4b. At this point the y-position of the string center node is fixed to be the centerline of the bow and the x-displacement constraint on the limb tip is removed. Additional x-displacements are applied to the string center until the bow has reached full draw, shown in Figure 4c. The reaction force on the string center and its displacement are recorded and integrated to give an estimate of the bow’s stored energy.

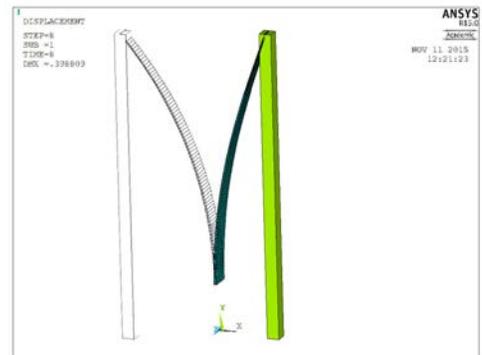
It was found that this method of solution was the most robust and reliable of those tried. Once the stored energy was found, a storage efficiency was calculated that compared the total energy stored to the theoretical maximum that could be stored. That is:

$$\eta = \frac{E}{F_{\max}D} \quad (2)$$

Thus, the finite element model output a single number representing the efficiency with which the design stores energy. This number automatically scales with the draw force, so only the shape of the force-draw curve is evaluated. This is important because the magnitude of the curve can be adjusted through other



(a) Initial, unstrained geometry



(b) Bow strained to brace height



(c) Bow at full draw

Figure 4: The finite element model at various load steps. Note that the long lime-green element represents the string but is not true shape or size.

means, such as scaling the limb’s width, without changing its energy storage efficiency.

The finite element model was paired with a Matlab optimization routine to find the optimal values for the geometric parameters of riser length, limb length, limb curvature, and string length. It was found that the limb should be as long as possible to reduce

geometric nonlinearities in the force-draw curve, and an optimal value for the curvature was found. It was found that the string length had a large effect on the model's ability to solve without crashing, so string length was removed from the optimization. Finally, an acceptable value for the riser length was found.

The ratio of tip width to root width and the thickness of the bamboo core at each limb section were determined using optimization by hand (trial and error). The goal was to find the appropriate values that would give a smooth circular arc shape to the limb when it reached full draw. This deflected shape is ideal because it indicates the absence of stress concentrations. Final geometric parameter values are listed in Table 1.

2.3 Layup Optimization

The optimal layup was determined to be: $[45_w/0^\circ/0^\circ/45_w/\text{Bamboo Core}/45_w/0^\circ/0^\circ/45_w]$. This ply sequence was found to provide the requisite draw weight (50 lbs) in a mass-efficient manner and be thin enough to effectively debulk and have good consolidation. Because large stiffness mismatch is known to be a cause for delamination, 45_w layers were applied directly to the bamboo core to provide a stiffness "bridge." 45_w layers make up the outermost layers to provide some degree of impact resistance; their lower stiffness means that if they are experience an impact they are less likely to break than the highly stressed unidirectional plies. Additionally, the outer 45_w layers provide increased torsional stability, which becomes important as reflex becomes more pronounced.

3 Manufacturing

The basic steps in the manufacturing process are as follows:

Table 1: Final values of geometric parameters for the BamBow.

Parameter	Value	Units
Limb Length	0.70	m
Limb Curvature	1/0.95	1/m
Riser Length	0.2567	m
Root Width	0.0508	m
Tip Width	0.0305	m
Section 1 Thk	4.8	mm
Section 2 Thk	4.2	mm
Section 3 Thk	3.6	mm
Section 4 Thk	2.4	mm



Figure 5: My manufacturing efforts steadily contributed to a pile of broken limbs.

1. Build appropriate tooling
2. Form bamboo core
3. Layup the pre-preg layers
4. Apply the peel-ply, breather, and vacuum bag
5. Debulk and cure in the oven
6. Remove bagging and trim part
7. Assembly

Each step is discussed in more detail in the sections below. Quality manufacturing proved to be a significant challenge on this project. While in the end I was able to make two high quality limbs, along the way I produced a pile of broken limbs, shown in Figure 5. I determined that many of the failures were due to inadequate strength in the bamboo (leading to the change to bamboo laminate flooring from bamboo planks) and manufacturing flaws (such as dust inclusions or inadequate consolidation).

3.1 Tooling

To aid in manufacturing, two tools were made using the ShopBot. A bow form was built consisting simply of a 2x6 board that had one side cut to the prescribed curvature of the limb. This was used as a template when heating and bending the bamboo core to ensure it had achieved the proper curvature. Additionally, a template was cut from 1/4" plywood to ensure the pre-preg plies were cut to a uniform size.

3.2 Forming the Bamboo Core

The bamboo core was made from laminated bamboo flooring. Early prototypes used cured bamboo planks, but these experienced failure due to excessive shear forces. Bamboo laminate was chosen for its higher strength and more uniform dimensions. The flooring was first trimmed to the outer dimensions of the core. Then its width was reduced using a sanding operation, and its thickness was reduced using another sanding operation. Fi-



(a) Trimming the flooring material (b) Bamboo blanks with width reduced



(c) Tapering thickness with a belt sander (d) After applying heat, the bamboo core is bent to its final shape.

Figure 6: Steps in shaping the bamboo core

nally, heat was applied to soften the bamboo and bend it to the proper curvature. A heat gun was used to apply heat to short sections, which were bent by hand and held until cool. This method was much faster than soaking or steaming and required far less tooling. These steps are illustrated in Figure 6. Finally, any sawdust was removed by wiping with a damp towel and drying using compressed air.

3.3 Layup

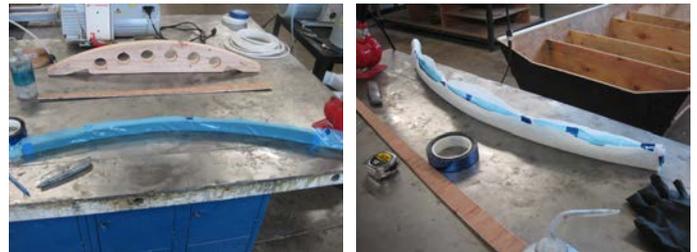
Before the layup can begin, each ply must be cut from the pre-preg material at the proper orientation. This was done using my plywood template and a rotary cutting tool. The woven plies are over ten years old, and their age has contributed to a good deal of "crustiness" which eliminates any tackiness and makes it difficult to stick the plies together. Accordingly, a heat gun was used to soften the resin in the woven pre-preg and improve its tackiness so it would stick to the bamboo core. The unidirectional plies were applied on top of the woven ply. Working from one end to the other helped eliminate bubbles and ensure good consolidation. Finally the outer layer of woven pre-preg was applied, again using a heat gun to soften the resin. This was repeated on both sides of the bamboo core to complete the layup. Figure 7 shows the layup in progress.

3.4 Breather, Peel Ply, and Vacuum Bag

The carbon pre-preg requires a cure under vacuum at elevated temperatures. Therefore, the completed layup must be wrapped with peel ply, breather, and vacuum bag before being



Figure 7: A layup in progress. One woven ply has been applied to the core; the others wait in the background.



(a) Single limb with perforated peel ply (b) Single limb with breather applied over the peel ply



(c) Single limb with complete vacuum bag (d) Single limb in oven during cure

Figure 8: Steps in vacuum bagging and cure

placed in the oven and subjected to vacuum. Kapton tape was used to secure these different layers in place. These steps are shown for a single limb in Figure 8 for a single limb. It should be noted that the final product was produced by building both limbs at once. Additionally, bricks were placed in the oven to support the core and preserve its curvature.

3.5 Debulking and Curing

The limbs were placed in the oven and underwent a debulk cycle at 160°F for one hour, then cured at 355°F for two hours.



Figure 9: Steel handle with wooden soft jaws and bamboo arrow rest.

The overall cycle time was near four hours due to preheating of the oven. It was found that limbs performed better if they were allowed to cool in the oven overnight. It is believed that this minimized any thermal stresses from rapid cooling, and also helped ensure a complete cure.

3.6 Removing Bag and Trimming

After removal from the oven, the vacuum bag, breather, and peel-ply materials were stripped off. The root and tip of each limb were trimmed with a rotary tool. Notches were cut in the tip to accept a bow string.

4 Assembly

The completed limbs were clamped in a specially designed and built handle, shown in Figure 9. This handle was designed to be easily clamped into the launch mechanism used at the bow competition. It uses wooden blocks epoxied to the clamp jaws to prevent crushing of the bow limbs. A bamboo arrow rest holds the arrow away from anything that might interfere with its fletching or flight.

To string the bow, it was clamped in a vise and deflected until the string loops could slip over the ends and seat in the notches. The completed bow is shown in Figure 10.

5 The Competition

The bow competition was held on November 5th 2015. The launch mechanism had been newly redesigned to use a pneumatic cylinder instead of a hand-cranked winch. Due to a miscalibration in the new system, the bows were not drawn to a full 50 pounds of draw force. Rather, they were pulled to 32 pounds. Because of variations in manufacturing, I brought two sets of limbs to the competition. One set was slightly stiffer and more



Figure 10: The completed bow

likely to pull the full fifty pounds, but was already beginning to delaminate. The other set was unbalanced (one limb was slightly stiffer than the other), but was structurally sound. I decided to start with the first set, then use the second set for my second shot if the first set broke (your distance is recorded as the best of two shots).

Figure 11 shows me setting up the bow in the launching mechanism. Figure 12 shows the bow after this first shot. As predicted, the limb experienced catastrophic delamination. Interestingly, the failure occurred during rebound, not during the drawing of the bow. Finally, Figure 13 shows my trophy collection including the Grand Prize trophy won using the BamBow. After the competition I used the launch apparatus to bring the bow to full draw; it launched the arrow a full 100 yards.

6 Summary

I designed and built a bamboo-cored carbon-fiber reflex bow. Its performance was adequate to win the Grand Prize in 2015 ME Department's Nephi's Steel Bow Competition.

Manufacturing a matched pair of high-quality bow limbs proved to be a significant challenge. The material is under extremely high stress, and any slight imperfection in the material could initiate failure. Eliminating manufacturing imperfections required diligent attention to details and assiduous manufacturing techniques. Many design changes were implemented to mitigate the effects of material and manufacturing imperfections. The design accomplished all of its goals, and provided me one final win in my long career participating in the Nephi's Steel Bow Competition.



Figure 11: Setting up the bow in the launcher.



Figure 12: Limb delamination experienced after my first shot.



Figure 13: My trophy collection. During the first year of competition trophies were not awarded, but note that I took second place that year.

7 Appendix

This appendix includes all of the code referenced in the body of the report. Some modifications may be required, especially if these are to be used with an optimization routine. For example, the values used in the optimization should be commented out of the parameters file or the optimization values will be overwritten.

7.1 ANSYS Scripts

The main ANSYS file builds and executes the finite element model. It references a parameter file that sets all geometric and layout parameters.

```
/CWD,'C:\ANSYS'  
!target draw weight is 222 N (50 lbs) at specified draw length  
!because this is only half a bow, we want 111 N  
!also I want to minimize weight  
fini  
/clear  
/PNUM,KP,1  
/PNUM,LINE,1  
pi = acos(-1)  
!first we call a file containing the model parameters  
/INPUT,'design_params_bam','txt','J:\Bow\Bambow\',, 0  
  
/prep7  
!build keypoints  
k,1,-r,riser/2,0  
k,3,0,riser/2,-wid1/2  
k,2,0,riser/2,wid1/2  
*do,n,1,num_sec,1  
k,,-r*(1-cos(theta*(n/num_sec))),riser/2+  
    r*sin(theta*(n/num_sec)),(wid1+((wid2-wid1)*n/num_sec))/2  
k,,-r*(1-cos(theta*(n/num_sec))),riser/2+  
    r*sin(theta*(n/num_sec)),-(wid1+((wid2-wid1)*n/num_sec))/2  
*get,kp%%,kp,,num,maxd  
*enddo  
  
!connect keypoints with lines  
*do,n,1,num_sec,1  
lstr,kp%%-3,kp%%-2  
larc,kp%%-3,kp%%-1,1,r  
lstr,kp%%-1,kp%%  
larc,kp%%,kp%%-2,1,r  
*enddo  
  
!build areas from the lines to represent limb geometry  
al,1,2,3,4  
*do,n,3,(num_sec-1)*3,3  
al,n,n+2,n+3,n+4  
*enddo  
  
!define a coordinate system for the elements  
local,11,0,0,0,90.,.,,  
esys,11
```

```

!assign element types to each section. each section gets its own element.
*do,n,1,num_sec,1
et,n,shell91
keyopt,n,1,numlay(n)
keyopt,n,2,1 !must set keyopt(2)=1 before we can input section data
keyopt,n,11,0 !center element nodes in the thickness
keyopt,n,8,1 !store layer data for failure analysis
*enddo

```

```

!the string is a link element
et,20,mpc184
keyopt,20,1,0

```

```

!use this element type to help hold the elements at the tip
et,21,mpc184
keyopt,21,1,0

```

```

et,22,link180 !string is a link, tension-only element
keyopt,22,3,1

```

```

!material properties for string - not reflective of actual string
mp,ex,20,207e9
mp,prxy,20,.36

```

```

r,20,0.001,0,1

```

```

!several choices for material
!input material properties
!uni-directional pre-preg
MPTEMP,,,,,,,,
MPTEMP,1,0
MPDATA,EX,1,,90.e9 !181e9
MPDATA,EY,1,,5.7e9 !10.3e9
MPDATA,EZ,1,,5.7e9 !10.3e9
MPDATA,PRXY,1,,.28
MPDATA,PRYZ,1,,.59
MPDATA,PRXZ,1,,.59
MPDATA,GXY,1,,7.17e9/2
MPDATA,GYZ,1,,10.3e9/(2*(1+.59))/2
MPDATA,GXZ,1,,10.3e9/(2*(1+.59))/2

```

```

!S-Glass/Epoxy
MPTEMP,,,,,,,,
MPTEMP,1,0
MPDATA,EX,2,0,43e9
MPDATA,EY,2,0,8.9e9
MPDATA,EZ,2,0,8.9e9
MPDATA,PRXY,2,0,.27
MPDATA,PRYZ,2,0,.4
MPDATA,PRXZ,2,0,.4
MPDATA,GXY,2,0,4.5e9
MPDATA,GYZ,2,0,8.9e9/(2*(1+.4))

```

```
MPDATA,GXZ,2,0,8.9e9/(2*(1+.4))
```

```
!Carbon weave
```

```
MPTEMP,,,,,,,,
```

```
MPTEMP,1,0
```

```
MPDATA,EX,3,0,28e9! 68.9e9
```

```
MPDATA,EY,3,0,28e9! 68.9e9
```

```
MPDATA,EZ,3,0,5.31e9/2
```

```
MPDATA,PRXY,3,0,.05
```

```
MPDATA,PRYZ,3,0,.4
```

```
MPDATA,PRXZ,3,0,.4
```

```
MPDATA,GXY,3,0,5.31e9/2
```

```
MPDATA,GYZ,3,0,5.31e9/(2*(1+.4))/2
```

```
MPDATA,GXZ,3,0,5.31e9/(2*(1+.4))/2
```

```
!bamboo
```

```
MPTEMP,,,,,,,,
```

```
MPTEMP,1,0
```

```
MPDATA,EX,4,0,9.0e9
```

```
MPDATA,EY,4,0,1.80e9
```

```
MPDATA,EZ,4,0,1.80e9
```

```
MPDATA,PRXY,4,0,.2
```

```
MPDATA,PRYZ,4,0,.2
```

```
MPDATA,PRXZ,4,0,.2
```

```
MPDATA,GXY,4,0,1.8e9
```

```
MPDATA,GYZ,4,0,1.8e9/(2*(1+.2))
```

```
MPDATA,GXZ,4,0,1.8e9/(2*(1+.2))
```

```
!input real constants for each element
```

```
!these include how many layers, their material, orientation, and thickness
```

```
*do,n,1,num_sec,1
```

```
R,n,numlay(n),sym,,,,dens
```

```
RMORE,,,,,,,,
```

```
*do,i,1,numlay(n)/2,1
```

```
RMORE,mat_nums(i),orient%%(i),numply%%(i)*thk%mat_nums(i),,,
```

```
*enddo
```

```
*enddo
```

```
!set mesh attributes
```

```
esize,esz
```

```
*do,n,1,num_sec,1
```

```
asel,s,area,,n
```

```
aatt,,n,n,,,,
```

```
*enddo
```

```
allsel
```

```
!mesh areas one by one
```

```
AMAP,1,2,3,4,5
```

```
*do,n,2,num_sec,1
```

```
amap,n,kp%%,kp%%-1,kp%%-3,kp%%-2
```

```
*enddo
```

```

allsel,all

csys,0
/eshape,1
epplot
nlgeom,on

!get node numbers from which to extract displacements and rotations.
nsel,s,loc,x,-r*(1-cos(theta))
nsel,r,loc,y,riser/2+r*sin(theta)
nsel,r,loc,z,0
*get,tip,node,0,num,max
allsel,all

type,21
nsel,s,loc,x,-r*(1-cos(theta))
nsel,r,loc,y,riser/2+r*sin(theta)
*get,numtip,node,0,count !find the number of nodes selected
*get,anchor,node,0,num,min
*do,i,1,numtip-1,1
e,anchor,ndnext(anchor)
nsel,u,node,,ndnext(anchor)
*enddo
allsel,all

nsel,s,loc,x,-r*(1-cos(theta/4))
nsel,r,loc,y,riser/2+r*sin(theta/4)
nsel,r,loc,z,0
*get,sec1,node,0,num,max
allsel,all

nsel,s,loc,x,-r*(1-cos(theta/2))
nsel,r,loc,y,riser/2+r*sin(theta/2)
nsel,r,loc,z,0
*get,sec2,node,0,num,max
allsel,all

nsel,s,loc,x,-r*(1-cos(3*theta/4))
nsel,r,loc,y,riser/2+r*sin(3*theta/4)
nsel,r,loc,z,0
*get,sec3,node,0,num,max
allsel,all

!constraints - string
n,-r*(1-cos(theta)),str,0 !node where we'll apply displacements
type,22 !build constraint elements. use 20 for mpc184 or 22 for link180
mat,20
real,20
*get,nock,node,0,num,maxd !get the node number of the nock
e,nock,tip

!fix base end

```

```

lssel,s,line,,1
nsl1,s,1
*get,num_grip,node,0,count
*dim,grip_nodes,array,num_grip,1
d,all,all,0

*do,i,1,num_grip,1
*get,maxnode,node,0,num,max
grip_nodes(i)=maxnode
nset,u,node,,maxnode
*enddo
allsel,all

!stringing the bow is tricky because we need to displace the string and tip together
!then constrain the string displacement in y but not x

d,nock,uz,0
d,tip,uz,0,,,,rotx,roty

*do,i,1,8,1
d,nock,ux,(0.15+r*(1-cos(theta)))*i/8
d,tip,ux,(0.15+r*(1-cos(theta)))*i/8
lswrite,i
*enddo

brace=(0.15+r*(1-cos(theta)))

ddelete,tip,ux,,,off
d,nock,uy,-str
lswrite,9

steps = 15
*do,i,1,steps,1
d,nock,ux,brace+(draw-.15)*i/steps
lswrite,i+9
*enddo

fini
/sol
lssolve,1,24,1 !lssolve,1,9+steps,1
fini

!now we extract the force and moment loads.
!these will be used in mathCAD to check material failure, and
!in Excel to generate a force-draw curve.
/post26
numvar,200
rforce,2,nock,f,y,yfor
rforce,3,nock,f,x,draw
nsol,4,nock,u,x,xdisp
filldata,7,,,,r*(1-cos(theta)),0

```

```

add,8,4,7,,ddisp,,1,-1
prvar,draw,yfor,ddisp

nsol,5,tip,u,x,tipx
nsol,6,tip,u,y,tipy

int1,9,3,8,,energy
prvar,energy
*get,en_tot,vari,9,rtime,24 !grab the value of total stored energy
*get,draw_for,vari,3,rtime,24 !grab the max value of draw weight

stor_eff=en_tot/(draw_for*draw)

!since we want to maximize the storage efficiency, we invert it to get fitness
fit = 1/stor_eff

*CFOPEN,shape_targ,txt,,
*VWRITE,fit
(F12.6)
*CFCLOSE

```

The parameter file contains all the information about the geometry and layup. To change the design, the parameter file would be modified and saved, then the above script re-run.

```

thk1 = 0.000203 !meters - the thickness of an individual ply
thk3 = 0.000331 !meters - thickness of woven ply
thk4 = 0.0003 !thickness of bamboo 'ply'
targ_draw = 40/(2*.2248) !target draw weight
dens = 1600.*thk1 !density (kg/m^2)
sym = 1 !1 for a symmetric layup, 0 for asymmetric
numstp = 20 !default number of substeps in each loadstep

num_sec = 4 !number of different lay-ups

len = 0.700 !0.7 !meters
r = 0.95!*1000 !0.9 !radius of curvature
riser = 0.2567 !8*.0254 !inches->meters Height of riser.
str = -0.045 !-.5*.0254
curv = 0.064 !curvature of limb cross section

wid1 = 0.0381*4/3 !meters
wid2 = 0.02286*4/3 !meters
brace = 5*.0254 !meters. The brace height.
draw = 33*.0254 !28.5*.0254 !inches -> meters. The draw distance 28.5

esz = wid1/6. !element edge length
mat_num = 3 !1 for carbon, 2 for S-glass, 3 for carbon weave, 4 for bamboo

theta = len/r

*dim,numlay,array,num_sec,1 !number of layers
numlay(1) = 8.
numlay(2) = 8.

```

```

numlay(3) = 8.
numlay(4) = 8.

*dim,mat_nums,array,numlay(1)/2,1 !material number of each ply
mat_nums(1) = 3. !this is laminated with carbon weave
mat_nums(2) = 1. !and unidirectional carbon fiber
mat_nums(3) = 3.
mat_nums(4) = 4. !the bow has a bamboo inner core

*dim,orient1,array,numlay(1)/2,1
orient1(1) = 45.0
orient1(2) = 0.
orient1(3) =45.0
orient1(4) = 0. !the bamboo core is oriented parallel to bow limb

*dim,orient2,array,numlay(2)/2,1
orient2(1) = 45.0
orient2(2) = 0.0
orient2(3) = 45.0
orient2(4) = 0.0 !the bamboo core is oriented parallel to bow limb

*dim,orient3,array,numlay(3)/2,1
orient3(1) = 45.
orient3(2) = 0.
orient3(3) = 45.
orient3(4) = 0. !the bamboo core is oriented parallel to bow limb

*dim,orient4,array,numlay(4)/2,1
orient4(1) = 45.
orient4(2) = 0.0
orient4(3) = 45.
orient4(4) = 0. !the bamboo core is oriented parallel to bow limb

*dim,numply1,array,numlay(1)/2,1
numply1(1) = 1. !
numply1(2) = 2.
numply1(3) = 1. !-
numply1(4) = 8. !bamboo core

*dim,numply2,array,numlay(2)/2,1
numply2(1) = 1.
numply2(2) = 2.
numply2(3) = 1. !-
numply2(4) = 7. ! bamboo core

*dim,numply3,array,numlay(3)/2,1
numply3(1) = 1.
numply3(2) = 2.
numply3(3) = 1. !-
numply3(4) = 6. !bamboo core

*dim,numply4,array,numlay(4)/2,1

```

```

numply4(1) = 1.
numply4(2) = 2.
numply4(3) = 1. !-
numply4(4) = 4. !bamboo core

```

7.2 Matlab Scripts

These scripts were used in conjunction with the ANSYS scripts given above to optimize the bow geometry. The first contains the optimization routine. The second executes the ANSYS batch files and returns a value for the optimization to use in finding gradients and optimal values.

```

%This script runs an optimization routine to find the bow shape that stores
%the most energy for a given draw weight and length

```

```

%define the starting point for the optimization
len = 0.7;    %length of bow limb in meters
r = 0.9;     %radius of limb curvature in meters
riser = 8.*0.0254;    %length of riser (inches -> meters)
str = -0.5*0.0254;    %length adjustment of string

```

```

X0 = [len, r, riser, str];

```

```

%define the max and min bounds on variables

```

```

lenmin = 0.5;
lenmax = 1.0;

```

```

rmin = 0.6;
rmax = 3.0;

```

```

risermin = 4.*0.0254;
risermax = 16.*0.0254;

```

```

strmin = -2*0.0254;
strmax = 0;

```

```

LB = [lenmin, rmin, risermin, strmin];
UB = [lenmax, rmax, risermax, strmax];

```

```

options = optimoptions('fmincon','DiffMaxChange',0.2,'DiffMinChange',0.005);

```

```

X = fmincon(@my_shape_opt,X0,[],[],[],[],LB,UB,[],options);

```

This script executes the ANSYS scripts and reads the ANSYS output file to get a value for the design fitness.

```

function target = my_shape_opt(X)
my_filename = 'C:\ANSYS\shape_targ.txt';
len = X(1)
r = X(2)
riser = X(3)
str = X(4)

```

```

command = ['"C:\Program Files\ANSYS Inc\v150\ANSYS\bin\winx64\ansys150.exe" ...
          -p aa_t_a -np 2 -dir "C:\ANSYS" -j "bambow" -s read -l en-us -len1 ' num2str(len) ' ...

```

```
-rad ' num2str(r) ' -riser ' num2str(riser) ' -str ' num2str(str) ' -b -i ...  
"J:\Bow\Bambow\shope_optim.mac" -o "C:\ANSYS\shape_out.txt"]];  
  
[status, result] = dos(command)  
%ANSYS returning anything but a 0 means an error occurred. Check the output  
%file for details.  
    if status == 1  
        target = 1e6;  
    else  
        fid = fopen(my_filename);  
        target = fscanf(fid, '%f12.6');  
        fclose(fid);  
    end  
end  
end
```