Discussion over the design assumption of buckling length “2a” for studs braced by sheathing
1. Introduction

Since AISI (1962) to the newest version of AISI-COFS (2007) stud buckling should be checked in the weak-axis over a length of two times the fastener spacing, \( a \). Figure 1 shows the configuration assumed. The idea behind the assumption is that would be safe in the case that the fastener is missed or ineffective consider that the column buckles over a length \( 2a \). The arbitrary number \( 2a \) has been carried over the codes for 46 years without any reliability study found.

The project makes an effort to understand the model and provide an overview of the problem, providing information to be considered in further studies.

2. Statistics study of the tests for spring stiffness

The literature review about spring stiffness conducted to the research developed by Fiorino et al. (2006), the tests are illustrated on Figure 2. Basically, the idea for the test is pull or push two C sections connected to a strip and study the stiffness provided for the connections between C section and the strip. The material used for the strips were gypsum and plywood. For this project only the data for plywood was used.
Different kinds of failure modes were found, Figure 3, and coupling of the modes illustrated were reported too. The test varied the strand orientation, the load direction, load rate and the loaded edge distance. From a reliability study point of view, too many variables were admitted for the number of test, but knowing the lack of study in this area the results encountered provide reasonable data to base this project.

To a reliability study the data has to be approximated by a good fitting curve. In the case studied Figure 4a shows the comparison between lognormal and normal CDF curve generation. It is clear that the lognormal curve provide better results than the normal curve. Figure 4b shows the results acquired generating lognormal distributed random numbers using the Matlab’s function.

In fact, lognormal distribution is the best fitting but it can’t be assumed to be an excellent solution. Figure 5 shows the relative frequency histogram and the PDF fitting, as can be seen, it is a rough approximation. Definitely, it would be worth conduct more
test decreasing the number of variables and increasing the number of tests. For this project was assumed a lognormal distribution based on the data available.

3. Example studied

The example chosen is slightly different as the one presented in the design manual AISI – COFS (2001), Figure 6. Cross-section, material properties and fastener spacing were assumed deterministic variables. The spring stiffness associated to the lateral restriction provided by the wall is the variable studied, as soon as, the probability of failure for the same.
Two results have to be highlighted: a) Buckling over the length 2a as proposed by the code is \( P_{cr} = \pi^2 E I / (2a)^2 = \pi^2 \times 29500 \times 0.1133 / (2 \times 8)^2 = 128.86 \text{kips} \) and b) Buckling over length a is \( P_{cr} = 515.43 \text{kips} \). Those numbers are reference for further discussions.

The examples analyzed differentiate from the example in the design manual, because the total length of the column is considered 100in. In this case are admitted 2in between the wall edge and the first and last fastener, then the 96in. between the first and last fastener is multiple of 16in (2a) and 8in (a).

4. Model to analyze beam on discrete spring

In order to develop a solution for the problem of a beam on multiples discrete springs the energy method was assumed. Due to the difficulty to find a closed solution for the problem Rayleigh-Ritz method was used.

Rayleigh-Ritz method is based on assume a displacement function that satisfy the boundary conditions, then the total potential energy function reduces from a functional to a function and can be used ordinary calculus to solve the problem, Chen and Lui (1987).

The deflection shape is represented in equation 1.

\[
\bar{v} = \sum_{i=1}^{n} a_i \phi_i = \sum_{i=1}^{n} a_i \sin \left( \frac{i \pi x}{L} \right) \tag{1}
\]

The strain energy, equation 2, can be simplified to equation 3, and the differentiation in relation to \( a_1, a_2, \ldots, a_n \) is represented for equation 4.

\[
\bar{U} = \frac{1}{2} \int_0^L EI \left( \frac{d^2 \bar{v}}{dx^2} \right)^2 dx \tag{2}
\]

\[
\bar{U} = \sum_{i=1}^{n} EI \pi^4 a_i \frac{h^4}{4 L^3} \tag{3}
\]

\[
\frac{\partial \bar{U}}{\partial a_n} = EI \pi^4 a_n \frac{h^4}{2 L^3} \tag{4}
\]

The potential energy due to the force \( P \), equation 5 can be simplified to equation 6, and the differentiation is represented in equation 7.

\[
\bar{V}_p = -\frac{P}{2} \int_0^L \left( \frac{d\bar{v}}{dx} \right)^2 dx \tag{5}
\]
\[ \bar{V}_p = -\sum_{i=1}^{n} \frac{P \pi^2 a_i^2 i^2}{4L} \]  \hspace{1cm} (6)

\[ \frac{\partial \bar{V}_p}{\partial a_n} = -\frac{P \pi^2 a_n n^2}{2L} \]  \hspace{1cm} (7)

The variables \( \frac{\partial U}{\partial a_n} \) and \( \frac{\partial \bar{V}_p}{\partial a_n} \) will turn to be a matrix that fills only the main diagonal due to the independence of the crossed terms, but in the case of the potential energy due to the springs the crossed terms has to be considered. The potential energy due to the springs is represented in the equation 8, the differentiation follows a rule which is presented in equation 9. In equation 9, \( n \) represents the line in the matrix and \( j \) the column that will form the matrix. The final matrix will always be of dimension \( n \times n \).

\[ \bar{V}_s = \sum_{i=1}^{\infty} \frac{1}{2} k \left( \sum_{i=1}^{n} a_i \sin \left( \frac{i\pi x}{L} \right) \right)^2 \] \hspace{1cm} (8)

\[ \frac{\partial \bar{V}_s(n,j)}{\partial a_n} = \sum_{i=1}^{\infty} k \sin \left( \frac{n\pi x}{L} \right) \cdot \sin \left( \frac{j\pi x}{L} \right) \] \hspace{1cm} (9)

The sum of the equations 4, 7 and 9 by energy equilibrium is equal to zero, then it is formed the eigenvalue problem.

A Matlab routine was developed to analyze the problem and vary the spring stiffness based on the lognormal distribution. The program is in the appendix 10.2. The program also permits the variation of the probability of failure for each spring using a linear function.

5. Comparison with ABAQUS model

To confirm the results using the Matlab program one of the examples was simulated in ABAQUS, the input is in the Appendix 10.1. The example was defined in chapter 3. Figure 7 shows a sketch of the problem as soon as the restrictions adopted to force the column to buckle in the direction that the springs are effective and the eigenvector for the first mode.
The eigenvector for the first mode is not composed for the same number of half-waves than the spaces between springs, what means that the springs does not provide enough restriction to the column to buckle between connections. By another way, the lengths of half-waves are not equal to 2a, although the number is very close for this example.

The eigenvalue identified using ABAQUS was 146.23kips and using the Matlab program the value is 148.11kips. The small difference between FEM and the energy method (Rayleigh-Ritz) can be explained by the fact that both assumptions are approximated solutions for the problem. Hence, the Matlab routine was considered a good approximation.

Another comment is how close the results are to the value calculated in chapter 3, \( P_{cr}(2a) = 128.86 \text{kips} \), and has to be said that the difference of 13% is acceptable considering the lack of a reliability knowledge to analyze the problem. The results are close because of the fact that the half-wave length formed for this spring stiffness is close to 2a, if it was not true the difference can be bigger. Remember that the spring stiffness are based on the material plywood but it could be gypsum which present low values for the spring stiffness and consequently the number of waves would be less and the assumption 2a would be unsafe. The consideration of a buckling length of 2a is reasonable but can be better.
6. Model studied

Two considerations can be done to analyze the problem: a) associate a probability of failure to each point connected in the column, Figure 1, or b) associate a probability of failure to each fastener that connects to the stud, Figure 8. The difference will be if happens a failure in a, the point in the column is complete ineffective, by another way considering b the point in the column can still have the contribution of the another spring. In fact, what happens is case b, where the probability of failure of each spring is uncorrelated, but case a is the case that the codes are based.

Figure 8 – Explanation of each point connected.

6.1 Probability of failure in each point connected, case a

Figure 9 shows the study done for case a, the probability of failure vary from 0 to 10%, for each probability of failure were analyzed 1000 models. All the values are plotted for the fastener spacing of 8in. (circles), as soon as the mean and the bounds (mean plus and minus standard deviation), in order to compare are plotted the results for fastener spacing of 16in. and probability of failure 0%.

The comparison between fastener spacing of 8 and 16in. shows that it is very conservative consider the distance 16in. based on the assumption that the fastener can be missed or over-drifted. As was expected the mean decrease and the standard deviation increase with the probability of failure, more discussion about the results are in the next section.
6.2. Probability of failure associate to each spring, case b

As the same way were done the simulations for the case b. Figure 10 shows the results, comparing the results to the previous simulations definitely case b conduct to results more uniform and with smaller values for standard deviation.

For a design engineer point of view what really matters is the partial safety factor, $\phi$, assuming that the coefficient of variation and mean of the professional factor are equal to 1, $\phi$ can be defined as equation 10.
\[ \phi = e^{-0.55\beta \sqrt{\mu_p}} \]  \(10\)

Assuming the reliability index, \(\beta\), equal 3.5, Table 1 presents the \(\phi\) values found.

<table>
<thead>
<tr>
<th>Case</th>
<th>(P_f=0%)</th>
<th>(P_f=1%)</th>
<th>(P_f=2%)</th>
<th>(P_f=5%)</th>
<th>(P_f=8%)</th>
<th>(P_f=10%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Case a</td>
<td>0.34</td>
<td>0.30</td>
<td>0.27</td>
<td>0.20</td>
<td>0.16</td>
<td>0.13</td>
</tr>
<tr>
<td>Case b</td>
<td>0.36</td>
<td>0.35</td>
<td>0.34</td>
<td>0.31</td>
<td>0.28</td>
<td>0.27</td>
</tr>
</tbody>
</table>

Consider case a or b makes difference when calculate the partial safety factor, and in a real system case b is what really happens. The partial safety factor apparently assume values very small, what contributes to this is the fact that the relation between the mean critical load and the critical load assuming mean values for the spring stiffness is very small.

7. Conclusions

As soon as, the engineers are able to model the column on discrete springs, how was explained in this project, the partial safety factors are defined in Table 1. Considering the buckling over the length \(2a\) is definitely too conservative, if it is a tentative to compute the probability to the fastener be over-drifted or missed.

By another way, seems that consider buckling over \(2a\) is a tentative to adapt the Euler column to the column on discrete spring. Anyway, doesn’t make sense hide this valuable information knowing the innumerous tools that are available nowadays.

8. Future research

The problem of lateral buckling on a column can be extended to a more complicated system where is considered the actual section. The simulation, Figure 11, lead to local and distortional buckling what is certainly a huge field to be explored.
9. References


10. Appendix

10.1 Abaqus inp file

*HEADING
*PREPRINT, MODEL=YES
FLEXURAL BUCKLING
*RESTART,WRITE,FREQUENCY=999
*NODE
  1,0.0,0.0,0.0
  101,100,0.0,0.0
*GEN, NSET=ALL
  1,101
*ELEMENT,TYPE=B31OS
  1,1,2
*ELGEN, ELSET=BEAM
1, 100, 1, 1
*BEAM_GENERAL SECTION, SECTION=GENERAL, ELSET=BEAM
0.327, 0.1133, 0, 0.1133, 100
0.0, 0.0, -1.0
29500
*Element, type=spring1
101, 3
102, 11
103, 19
104, 27
105, 35
106, 43
107, 51
108, 59
109, 67
110, 75
111, 83
112, 91
113, 99
*Elset, elset=spring1, generate
101, 113, 1
*Spring, elset=spring1
2, 2
13.0868
*BOUNDARY
101, 1, 3
1, 2, 3
ALL, 3
*STEP
*BUCKLE
5,
*CLOAD
1, 1, 1.
*END STEP

10.2 Matlab program

close all
clear all
cic

%Generate k OSB
%Test data
osb_k=[1.30 1.42 1.22 1.73 1.07 1.27 0.85 1.13 1.07 1.10 0.82 0.77 1.11
0.92 1.39 2.05 1.05 1.10 1.11 1.82 1.28 0.92 0.77 0.98 0.73 0.86 1.10];
%Trasformation of units kN/mm to kip/in (google='1kN/mm in pound/in' divide by 1000)
  osb_k=osb_k/0.175126835;
mean_osb_k=mean(osb_k);
std_osb_k=std(osb_k);

%Change for lognormal
sigma_ln_osb_k=(log((std_osb_k/mean_osb_k)^2+1))^0.5;
mean_ln_osb_k=log(mean_osb_k)-0.5*(std_osb_k)^2;

%Datas
lt=100;
x=[2:8:98];
E=29500;
I=0.1133;
n=length(x);
pf=10 %probability of failure in %

%Loop to change number of simulation
for sim=1:1000
%Spring's Matrix generation - Us
for i=1:n
    for j=1:n
        % Random generation of ksl
        ksl=lognrnd(mean_ln_osb_k,sigma_ln_osb_k,1,1);
        if ksl<0
            ksl=0;
        else
            end
        % Active or not
        act=rand(1,1);
        act=act>pf/100;
        ksl=ksl*act;

        % Random generation of ks2
        ks2=lognrnd(mean_ln_osb_k,sigma_ln_osb_k,1,1);
        if ks2<0
            ks2=0;
        else
            end
        % Active or not
        act=rand(1,1);
        act=act>pf/100;
        ks2=ks2*act;

        %Final spring stiffness
        ks=ks1+ks2;
        plus=0;
        for k=1:length(x)
            Us(i,j)=plus+ks*sin(i*pi*x(k)/lt)*sin(j*pi*x(k)/lt);
            plus=Us(i,j);
        end
    end
end

%Strain Energy's Matrix - Ub
for m=1:n
    Ub(m,m)=(E*I*(pi)^4*(m)^4)/(2*(lt)^3);
end

%Potential Energy's Matrix without contribution of spring
for m2=1:n
    V(m2,m2)=((pi^2*m2^2)/(2*lt));
end

%Build Matrix not dependent of P
U=Ub+Us;

%Find eigenvalue
P(:,sim)=eig(U,V);
minimum(sim)=min(P(:,sim));
end

%Find statistic of the minimum P found in each simulation
Pmean=mean(minimum)
Pstd=std(minimum)
exp1=exp(-0.55*3.5*std(minimum)/mean(minimum))

%Save variable for more studies
save('Pcr_8_pf10_2xk', 'minimum')