Optimum Cost Design of Partially Composite Steel Beams Using LRFD

M. ASGHAR BHATTI

1. INTRODUCTION

Design of composite steel beams is a trial and error process. The procedure recommended by the *Manual of Steel Construction, Load and Resistance Factor Design* (LRFD)[1] is to first assume a depth for the steel beam and then compute a trial beam size from an equation on page 5-11.[1] For a full composite action the number of shear studs is chosen with the assumption that the plastic neutral axis (PNA) is in the concrete slab. For partially composite design the number of shear studs is chosen rather arbitrarily by the designer. The flexural capacity of the composite section is then computed based on the ultimate stress distribution over the cross-section. If the capacity is less than the required, the design is revised in one of several ways: by increasing the number of shear studs, by increasing the beam depth, or by choosing a heavier section. These decisions must be based on the designer's judgment without any available guidelines. Shear and deflection considerations may force additional changes in the design. The resulting design is usually not the minimum cost design and at best is a feasible design. Several iterations of this entire process may be necessary if the goal is to minimize the cost.

The optimum design can directly be obtained by formulating the problem as a nonlinear optimization problem. However conventional methods for solution of these nonlinear programming problems are iterative and require large computer resources. Itoh[2] presented a formulation for minimum weight design of continuous composite girders based on the AASHTO specifications.[3] The economies of using LRFD in composite floor beams were discussed by Zahn.[4] A cost based optimization model for design of composite beams was presented by Lorenz.[5] However they did not present a methodology for obtaining optimum designs. Their main goal was to develop a better understanding of the partially composite design using the LRFD specifications.

This paper starts with the basic minimum cost model for composite beams presented by Lorenz[5] and extends it into a standard optimal design form. In this form the minimum cost design problem is reduced to the solution of a nonlinear programming problem. The optimum solution of this problem is obtained from the necessary optimality conditions. The required algebra can easily be carried out using symbolic algebra programs such as Mathematica[6] which is widely available at a reasonable cost on a variety of computers including UNIX workstations, Apple Macintosh and PC-Windows personal computers. The problem is entered into Mathematica in a customary algebraic form. The optimum solution is obtained quickly without many iterations. The procedure does not require detailed knowledge of the optimization theory. Also very few Mathematica commands are necessary to implement the solution algorithm. Besides being simple to implement, the biggest advantage of this approach is its generality. The objective function and the constraints can be changed at any time to reflect designer and client preferences and a new optimum design obtained quickly. A designer can thus concentrate on the design problem formulation and explore more alternatives.

The basic problem formulation is presented in Section 2. A brief review of the solution procedure for solving a nonlinear optimization problem using the Lagrangian function is presented in Section 3. A number of numerical examples are presented in Section 4.

M. Asghar Bhatti is associate professor, department of civil and environmental engineering, University of Iowa, Iowa City, IA.

2. PROBLEM FORMULATION

For design of partially composite beams there is a trade-off between steel beam weight and number of shear studs for a given set of design requirements. Thus the primary design variables are, area of cross-section (A_s), depth of steel beam (d), and the number of shear studs (N_s).

The total cost of a composite beam is the sum of steel beam cost and the stud cost. To minimize cost, the objective function therefore can be expressed as follows.

$$\text{cost} = W_s L C_m + N_s C_s$$

where

W_s = weight of steel beam (lbs/ft).
L = beam span (ft)
C_m = fabricator's cost of mill steel ($/lb)
N_s = total number of studs per beam
C_s = cost of a field installed stud ($/stud)

From optimization point of view it is not important to know the exact cost coefficients C_m and C_s. It is the relative cost of the two that determines an optimum design. Therefore the objective function is expressed as follows.

$$f = A_s \rho_s L + N_s C_s / C_m \equiv A_s \rho_s L + N_s C_{sm}$$

where

A_s = area of cross-section (in.^{2})
ρ_s = unit weight of steel, usually taken as 490 lbs/ft^3
C_{sm} = relative cost of a stud to the cost per lb of mill steel

The relative cost coefficient C_{sm} varies based on a job-size and geographical region and could range between 6 and 12.[5]

LRFD Flexural Strength Constraints

Using the LRFD specifications the flexural strength of a partially composite beam can be computed from a fully plastic stress distribution. The shear studs control the compression carried by the concrete. The plastic neutral axis (PNA) can either be in the beam flange or in the beam web.

PNA in Beam Flange

In this case the ultimate stress distribution is as shown in Figure 1. The compression taken by concrete is limited by the number of shear studs provided. Using the notation shown in Figure 1, the forces taken by concrete and steel can be expressed as follows.

$$C_c + \Sigma Q_n \le 0.85 f_c' b t_c$$

$$C_s = F_y b_f \bar{t}$$

$$T = A_s F_y - F_y b_f \bar{t}$$

where

f_c' = specified compressive strength of concrete, ksi
ΣQ_n = strength of shear connectors between the point of maximum positive moment and the point of zero moment to either side, kips. For a simply supported beam under uniform loading $\Sigma Q_n = Q_n N_s / 2$ where Q_n is the strength of a single shear connector.

The neutral axis is located by equating total compression to total tension as follows.

$$C_c + C_s = T \rightarrow \bar{t} = \frac{1}{2F_y b_f}[A_s F_y - C_c]$$

The nominal moment capacity of the beam can then be computed as follows.

$$M_n = C_s(\bar{y} - \bar{t}/2) + C_c(\bar{y} + Y_{con} - a/2)$$

where

$$\bar{y} = \frac{A_s d/2 - b_f \bar{t}^2/2}{A_s - b_f \bar{t}}$$

$$a = \frac{\Sigma Q_n}{0.85 f_c' b}$$

Note that for a fully composite section with $C_c = A_s F_y$ the above equations give

$$\bar{t} = 0 \text{ and } \bar{y} = d/2$$

Thus the full composite situation is covered by these equations as a special case.

PNA in Beam Web

In this case the ultimate stress distribution is as shown in Figure 2. Using the notation shown in Figure 2, the forces taken by concrete and steel can be expressed as follows.

$$C_c = \Sigma Q_n \le 0.85 f_c' b t_c$$

$$C_{sf} = F_y b_f t_f$$

$$C_{sw} = F_y t_w \bar{t}$$

$$T = A_s F_y - C_{sf} - C_{sw}$$

The neutral axis is located by equating total compression to total tension as follows.

$$C_c + C_{sf} + C_{sw} = T \rightarrow \bar{t} = \frac{1}{2F_y t_w}[A_s F_y - C_c - 2C_{sf}]$$

The nominal moment capacity of the beam can then be computed as follows.

$$M_n = C_{sw}(\bar{y} - t_f - \bar{t}/2) + C_{sf}(\bar{y} - t_f/2) + C_c(\bar{y} + Y_{con} - a/2)$$

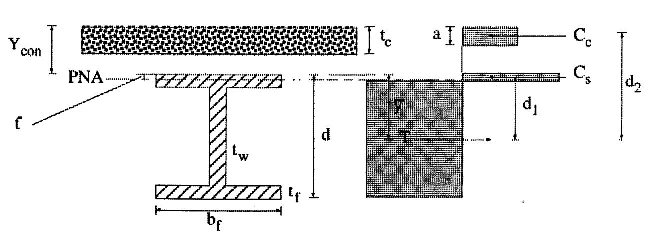

Fig. 1. Stress distribution at ultimate with PNA in beam flange.

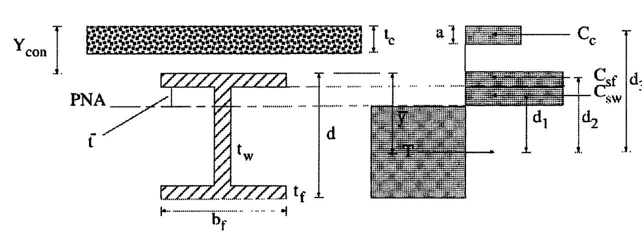

Fig. 2. Stress distribution at ultimate with PNA in beam web.

$$\bar{y}=\frac{A_s\,d/2-b_f t_f^2/2-t_w\bar{t}(t_f+\bar{t}/2)}{A_s-b_f t_f-t_w\bar{t}}$$

$$a=\frac{\Sigma Q_n}{0.85f_c'b}$$

Note that these equations reduce to those for a non-composite design when there are no shear studs.

Constraint expressions

The LRFD specifications require that the maximum bending moment due to factored loads must be smaller than 0.85 × nominal moment capacity. For unshored construction the steel beam itself must be strong enough to resist weight of wet concrete and other temporary load before concrete cures. Thus the flexural strength constraints can be expressed as follows.

$$M_u[\text{composite}] \le 0.85M_n$$

or

$$g_1 \equiv M_u[\text{composite}] - 0.85M_n \le 0$$

$$M_u[\text{construction}] \le 0.9F_y Z_s$$

or

$$g_2 \equiv M_u[\text{construction}] - 0.9F_y Z_s \le 0$$

where

M_u[composite] = maximum bending moment due to factored loads acting on the composite section (k-in.)
M_u[construction] = maximum bending moment due to factored loads acting on the steel section alone during construction before concrete cures (k-in.)
F_y = yield strength of concrete (ksi)
Z_s = Plastic section modulus of the steel section alone (in.^{3})

Two additional constraints are needed to make sure that the assumptions on which the nominal moment capacity is computed are not violated. These constraints are expressed as follows.

$$a \le t_c \text{ or } g_3 \equiv a - t_c \le 0$$

$$\bar{t} \ge 0 \text{ or } g_4 \equiv -\bar{t} \le 0$$

Normalized form of these constraints is sometimes more convenient and is obtained by dividing the constraints by their upper limit. The constraints in the normalized form can be written as

$$\tilde{g}_1 \equiv \frac{M_u[\text{composite}]}{0.85M_n} - 1 \le 0$$

$$\tilde{g}_2 \equiv \frac{M_u[\text{construction}]}{0.9F_y Z_s} - 1 \le 0$$

$$\tilde{g}_3 \equiv \frac{a}{t_c} - 1 \le 0$$

$$\tilde{g}_4 \equiv -\bar{t} \le 0$$

An advantage of the normalized form is that numerical values of the constraint function are between 0 and 1 for most feasible designs. These normalized forms will be used for obtaining graphical solutions later in Section 4.

Deflection Constraint

For most composite beam designs, it is desirable to limit live load deflection to *L*/360. Assuming a uniformly distributed load the constraint can be written as follows.

$$\frac{5w_L L^4}{384E_s I_{eff}} \le \frac{L}{360}$$

or

$$g_5 \equiv \frac{5w_L L^4}{384E_s I_{eff}} - \frac{L}{360} \le 0$$

or

$$\tilde{g}_5 \equiv \frac{5w_L L^4/384E_s I_{eff}}{L/360} - 1 \le 0$$

where

w_L = uniform service live load (lbs/in.)
L = span length (in.)
E_s = modulus of elasticity for steel (ksi)
I_{eff} = effective moment of inertia of a partially composite beam (in.^{4})

The following equation is suggested for computing I_{eff} (LRFD *Manual*[1]).

$$I_{eff} = I_s + \sqrt{\frac{\Sigma Q_n}{C_f}}(I_{tr} - I_s)$$

where

I_s = moment of inertia of steel section (in.^{4})
I_{tr} = moment of inertia for the fully composite uncracked transformed section (in.^{4})
C_f = compression in concrete for fully composite beam, kips.

Since the constraint g_3 makes sure that sufficient concrete slab exists, the fully composite design is governed by the steel beam. Thus $C_f = A_s F_y$ is used in this formulation.

With reference to Figure 3, the following expression for I_{tr} can easily be derived.

$$I_{tr} = I_s + A_s(d/2 + Y_{con} - \bar{y}_{tr})^2 + \frac{b_e t_c^3}{12} + b_e t_c(\bar{y}_{tr} - t_c/2)^2$$

where

$$\bar{y}_{tr} = \frac{b_e t_c^2/2 + A_s(d/2 + Y_{con})}{b_e t_c + A_s}$$

The equivalent width of the transformed concrete slab, $b_e = b/n$ where n the ratio of the modulus of elasticity of steel and concrete. As suggested by Segui[7] the modular ratio n can be adjusted to take into account long term concrete creep effects, if desired.

Vibration Constraint

Floor vibrations may be a concern since the optimized composite floor systems are usually lighter than conventional non-composite designs.[4] Thus a limit on the natural frequency of the beam may be desirable. Following Murray[8] the motion of the floor system caused by normal human activity will not be objectionable if the following inequality is satisfied.

$$D \geq 35A_o f_n + 2.5$$

or

$$g_6 \equiv 35A_o f_n + 2.5 - D \leq 0$$

or

$$\tilde{g}_6 \equiv \frac{35A_o f_n + 2.5}{D} - 1 \leq 0$$

where

D = damping in percent of critical
A_o = maximum initial amplitude of the floor system due to heel-drop excitation (in.)
f_n = fundamental natural frequency of the floor system (Hz)

For simply supported slab-deck systems, the parameters f_n and A_o are estimated as follows.

$$f_n = 1.57\left[\frac{gE_s I_{eff}}{WL^3}\right]^{1/2}$$

$$A_o = DLF_{\max}\frac{600L^3}{48E_s I_{eff}} / N_{eff}$$

where

W = Total dead load + 10 to 25 percent of live load (lbs)
g = acceleration due to gravity = 386 in/sec^2
$DLF_{\max}$ = maximum dynamic load factor. Typical values are 0.75 for secondary beams and 1 for main girders.
N_{eff} = 1 for main girders and given by the following equation for secondary beams

$$N_{eff} = 2.97 - \frac{S}{17.3d_e} + \frac{L^4}{1.35E_s I_{eff}}$$

where

S = beam spacing (in.) and
d_e = effective slab depth (in.)

Typical office floor systems provide a damping of 3.5–4 percent. Any additional damping requires special treatment at considerable cost.[8]

Expressions for I_s and Z_s

The calculation of constraints g_2, g_5, and g_6 poses a problem because they need I_s and Z_s. As mentioned earlier, the primary steel section design variables are A_s and d. Thus for common W shapes relationships for I_s and Z_s are needed in terms of A_s and d. The following equations were developed by Bhatti[9] for section properties using least squares data fitting.

$$I_s \approx 41.45 + 0.18A_s d^2 - 0.02d^3$$

$$Z_s \approx 6.27 + 0.41A_s d - 0.045d^2$$

For most sections from W30 to W12 the values computed from these equations are within 5 percent of those given in the LRFD *Manual* and thus are suitable for optimization purposes. Once a design is established, a final analysis using the exact section properties must be performed to make sure that the design meets all LRFD requirements.

Practical Design Constraints

Besides the strength and serviceability constraints, any number of practical constraints can be incorporated into the formulation. For example d can be restricted to be between a specified upper and lower limit. In the numerical examples presented the following constraint on the overall depth of steel beam is imposed.

$$d \leq d_{\max}$$

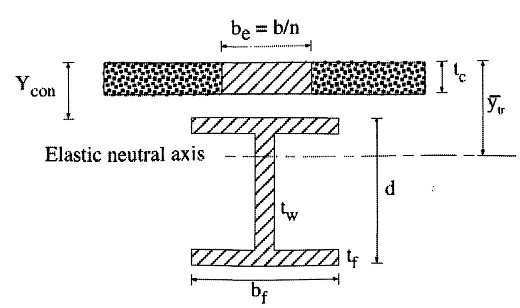

Fig. 3. Transformed steel section.

or

$$g_7 \equiv d - d_{\max} \le 0$$

or

$$\tilde{g}_7 \equiv \frac{d}{d_{\max}} - 1 \le 0$$

3. SOLUTION USING THE LAGRANGIAN FUNCTION

The general form of optimal design problem is expressed as follows.

> Find n design variables $x = x_1, x_2, \ldots, x_n$ which minimize $f(x)$ subject to m constraints $g_i(\boldsymbol{x}) \le 0$, $i = 1, \ldots, m$.

Note that the general problem is expressed as a minimization problem with all constraints being of less than or equal to type ($\le$). A maximization problem can be converted to a minimization problem by multiplying the objective function by a negative sign. Constraints of greater than or equal to type ($\ge$) can be converted to the standard form of less than or equal to type ($\le$) type by multiplying their expressions by a negative sign. Thus the general problem as stated covers all practical cases.

A solution of the general constrained problem can be obtained by minimizing the following function known as the Lagrangian function.[10]

$$L(\boldsymbol{x},\boldsymbol{u}) = f(\boldsymbol{x}) + \sum_{i \in \text{active}} u_i g_i(\boldsymbol{x})$$

where the sum is over active constraints and u_i are additional unknown variables called Lagrange multipliers. The active constraints are those which at the optimum point are exactly equal to 0. The necessary conditions for minimum of the Lagrangian function are

$$\frac{\partial L}{\partial x_i} \quad i = 1, \ldots, n$$

$$\frac{\partial L}{\partial u_i} \equiv g_i = 0 \quad i \in \text{active}$$

A solution of this system of equations (nonlinear in general) gives the optimum for the general optimization problem.

The main difficulty in the use of the Lagrangian function is the determination of active constraints. For problems with two design variables, constraint functions and objective function contours can be drawn to visually determine the active constraints at optimum. The procedure is explained in the Example 1 in the next section. For problems with more than two variables no simple procedure is available to determine the active constraints at optimum. Using intuition, based on anticipated design, one must assume one or more constraints to be active and solve the resulting system of equations. The solution is correct if the following two conditions are met.

- All Lagrange multipliers u_i are positive
- All constraints are satisfied, i.e., $g_i(\boldsymbol{x}) \le 0$, $i = 1, \ldots, m$.

If any one of these conditions is not satisfied, a different set of constraints is assumed to be active and the procedure is repeated.

For the beam formulation presented in Section 2, several interesting problems can be formulated in terms of two design variables. Thus the graphical solution is possible. The numerical examples presented in the next section all involve two design variables and therefore their solutions did not require any iterations.

4. NUMERICAL EXAMPLES

Several numerical examples are presented in this section to illustrate the solution procedure. Only a few of the Mathematica functions are needed for the solution. These functions and general Mathematica conventions are described in detail in the first example. The remaining examples present the solutions without additional comments related to Mathematica syntax. The first example should be studied carefully for readers not already familiar with Mathematica. For all examples the problem formulation is entered into Mathematica as follows.

Assumptions:

Unshored construction, Simply supported beams, Compact W shapes. Uniformly distributed loads. Use k-in. units. For partial composite design, the neutral axis in the steel beam flange. Sufficient concrete compressive strength exists

Initialization:

```
Clear[wd, wdSuperimposed, constructionLive,
  wl, L, Csm, Qn, Ns, As, d, tw, bf, Fy,
  Es, fc, b, tc, deckDepth];

rhoS = 490/(12^3);

Is := 41.45 + 0.18 As d^2 - 0.02 d^3;
Zs := 6.27 + 0.41 d As - 0.045 d^2;
```

The first line clears any previously defined values for the problem variables. `Clear` is a standard Mathematica function. The notation for the variables is same as that used in Section 2 with the following additional definitions.

wd	= dead load (k/in.)
wdSuperimposed	= additional dead load applied after concrete cures (k/in.)
constructionLive	= live load during construction applied before concrete cures (k/in.)
wl	= live load (k/in.)
L	= span length (in.)
fc	= f_c' concrete compressive strength

Note all variables defined in Mathematica are case sensitive. Thus *Fy* is not the same as *fy* and so on. Later the variables must be referred to exactly the same way as they were first defined. The semicolon (;) at the end of a line tells Mathematica to suppress any output resulting from execution of this command. This is a useful feature to avoid screen clutter when simply entering definitions or when the results are of intermediate nature and are not worth looking at.

The second line defines *rhoS* = unit weight of steel (lbs/in.^{3}). The next two lines define approximate expressions for moment of inertia and plastic section modulus in terms of *As* and *d*. Note the multiplication symbol (*) between different variables is optional. The notation := used in these definitions makes them *delayed execution* statements. Expressions defined with just an equal sign (=) are evaluated immediately using the current values of the variables on the right hand side. However the delayed execution statements are evaluated when they are actually referenced in the right-hand side of another expression using the parameter values at that time. Since we are interested in evaluating these functions for a variety of different parameter values, the delayed execution mechanism of Mathematica is used here.

Objective function:

```
f := As rhoS L + Ns Csm;
```

Flexural strength of composite beam (PNA in beam flange)

```
Ycon := tc + deckDepth;
sQn := (Ns/2) Qn;
a := sQn / (0.85 fc b);
tbar := (As Fy - sQn)/(2 Fy bf);
ybar := (As d/2 - bf tbar tbar/2) / (As -
  bf tbar);
Mn := Fy bf tbar (ybar - tbar/2) + sQn
  (ybar + Ycon - a/2);
```

Beam moments:

```
wuComposite := 1.2 (wd + wdSuperimposed +
  As rhoS/1000) + 1.6 wl;
wuConstruction := 1.2 (wd + As rhoS/1000) +
  1.6 constructionLive;
Mu[w_] := w L^2 /8;
```

wuComposite defines factored load acting on the composite section and *wuConstruction* defines the load during construction. The *Mu* `[w_]` line defines a function in Mathematica that computes maximum bending moment for a simply supported beam. *w_* in the function definition tells Mathematica to replace *w* in the right hand side by the argument used in this function call. For example *Mu[wuComposite]* gives maximum bending moment due to *wuComposite* load.

Flexural strength constraints (both in standard and normalized form as defined in Section 2)

```
g1 := Mu[wuComposite] - 0.85 Mn;
g2 := Mu[wuConstruction] - 0.9 Fy Zs;
g3 := a - tc;
g4 := -tbar;

ng1 := Mu[wuComposite]/(0.85 Mn) - 1;
ng2 := Mu[wuConstruction]/(0.9 Fy Zs) - 1;
ng3 := a/tc - 1;
ng4 := -tbar;
```

Deflection and vibration constraints:

```
be := b/n;
ybarTr := (be tc tc/2 + As(d/2 + Ycon))/(be
  tc + AS);
Itr := Is + As (d/2 + Ycon - ybarTr)^2 +
  1/12 be tc^3 + be tc (ybarTr - tc/2)^2;
Ieff := Is + Sqrt[sQn/(As Fy)] (Itr - Is);
g5 := (5 wl L^4)/(384 Es Ieff) - L/360;

W := (wd + wdSuperimposed + As rhoS/1000 +
  0.2 wl) L;
Ao := (DLFmax 0.600 L^3 /(48 Es Ieff))/Neff;
fn := 1.57 Sqrt[386. Es Ieff/(W L^3)];
g6 := 35 fn Ao + 2.5 - maxDamping;

ng5 := ((5 wl L^4)/(384 Es Ieff))/(L/360) -
  1;
ng6 := (35 fn Ao + 2.5)/maxDamping - 1;
```

Practical constraints:

```
g7 := d - dmax;
ng7 := d/dmax - 1;
```

Flexural strength when PNA is in the web:

```
CsfB := bf tf Fy;

tbarB := (As Fy - 2 CsfB - sQn)/(2 Fy tw);

ybarB := (As d/2 - bf tf tf/2 - tw tbarB
  (tbarB/2 + tf)) / (As - bf tf - tbarB tw);

MnB := Fy tw tbarB (ybarB - tf - tbarB/2) +
  CsfB (ybarB - tf/2) + sQn (ybarB + Ycon -
  tc/2);
```

Example 1—Full Composite Design: *As* & *d* Variables

The numerical data for this example is same as that of Example 5-2, p. 5-14 LRFD *Manual*.[1] For this example *As* and *d* are the only two optimization design variables. The other variables are assigned the following values.

```
wd = 0.9/12;
wdSuperimposed = 0;
constructionLive = 0.2/12; wl = 2.5/12; L =
  40*12;
Csm = 12; Fy = 50; fc = 4; Es = 29000; n =
  9;
b = 120; tc = 4.5; deckDepth = 3;
DLFmax = 1; Neff = 1; maxDamping = 4;
bf = 7; Qn = 26.1; dmax = 24;
Ns := 2 As Fy/Qn;
```

All quantities are expressed in k-in. units. A relative cost value of $C_{sm} = 12$ is used. DLF_{max} is set to 1 for vibration constraint. After an initial design a better value may be obtained from the tables given in the LRFD *Manual.* The flange width *bf* is needed in partial composite design. A reasonable value should be used here. After a design is established the final check is performed using the exact value obtained from the steel tables.

If desired, the `Simplify` function from Mathematica can be used to get objective and constraint functions in a simplified form. Mathematica automatically substitutes the above values for the variables and produces concise expressions in terms of the remaining variables. For example

```
Simplify[{f, g1, g2}]
```

```
{182.088 As, 12192. - 308.95 As + 2.60417
  As^2 - 21.25 As d, 3077.85 + 9.8 As -
  18.45 As d + 2.025 d^2}
```

The Mathematica output is shown in boldface. The expressions clearly show that there are only two remaining variables and therefore a graphical solution is possible to determine the status of constraints. The next line defines the lower and upper limits for the plots. A_s is plotted on the x-axis and d is plotted on the y-axis. A certain trial and error may be necessary to get the suitable plot limits.

```
x1min=10; x1max=40; x2min=10; x2max=30;
```

For plotting purposes it is better to use the normalized form. Two closely spaced contours are drawn for each constraint. One contour is for constraint value equal to 0 that defines the constraint boundary. The other contour is for a slightly violated constraint (with a value greater than 0) to define the infeasible side of the constraint. With normalized constraints

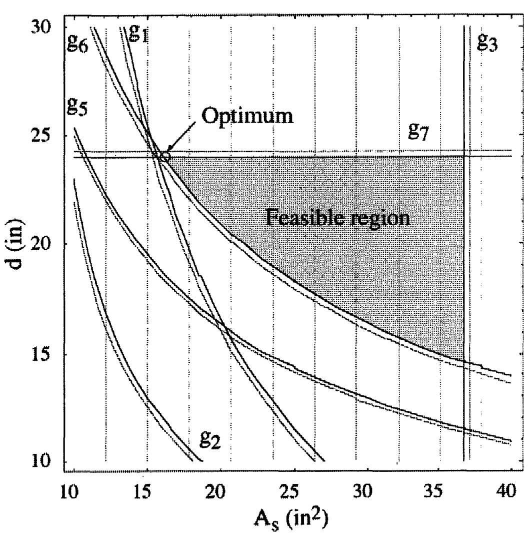

Fig. 4. Graphical solution of Example 1.

a value of 0.01–0.05 for this second contour works well. If the constraints are not normalized, it is difficult to determine a suitable second contour value that is close enough to 0 and still shows up on the graph as a distinct contour.

The following line produces a contour plot of constraint *ng1* using the `ContourPlot` function from Mathematica. The second and third arguments define the variables to be plotted with their lower and upper limits. The fourth argument turns the shading option off. By default Mathematica produces shaded contour plots which are not very meaningful for our purposes here. The fifth argument defines that we want to plot two contours with values of 0 and 0.02. The next argument defines styles for the two contours. The actual boundary of the constraint is drawn as a solid line. The second contour denoting the infeasible side is drawn using 50 percent gray level. The last argument suppresses the actual screen display of this contour. After this command is executed, the resulting plot will not show up on the screen but will be stored internally in Mathematica as variable *p1*.

```
p1 = ContourPlot [ng1,
{As, x1min, x1max}, {d,x2min, x2max},
ContourShading->False, contours->{0, .02},
ContourStyle->{{GrayLevel [0]},
  {GrayLevel [0.5]}},
DisplayFunction->Identity];
```

Using the same command the contour plots of functions *g2, g3, g4, g5, g6* and *g7* are generated and stored as variables *p2* through *p7.*

The contours for the objective function are generated next using essentially the same procedure as that for constraint functions. The only difference is that instead of plotting contour for a specific value of *f,* 10 contours are generated. Mathematica automatically decides at suitable values for plotting these contours. The contour style defines a different gray level to distinguish the objective function contours from the constraint function contours. A variable called *obj* is used to store the objective function contours.

```
obj = ContourPlot [f,
{As, x1min, x1max}, {d,x2min, x2max},
Contourshading->False, Contours->10,
ContourStyle->{{GrayLevel [0.8]}},
DisplayFunction->Identity];
```

The last step is to show all these contours superimposed on a single graph. This is accomplished using the `Show` command and setting the *DisplayFunction* to the default display device (which is screen on most computer systems).

```
Show [{obj, p1, p2, p3, p4, p5, p6, p7},
DisplayFunction->$DisplayFunction]
```

The resulting plot is shown in Figure 4. The only information added manually to the plot are the axes labels, the identification of the feasible region and the labels for constraints. The objective function contours are vertical lines because with N_s fixed the cost is a function of A_s alone. The cost increases as A_s increases. The lowest cost function contour that does not violate any constraint corresponds to the optimum solution. In this example the optimum point is reached when the objective function contour passes through the intersection of contours g_6 and g_7. Thus for this design if the depth is limited to 24 inches then the limiting factor in choosing A_s is the vibration constraint. A lower cost design can be obtained if larger depths are allowed. For depths greater than 25 inches the flexural strength will govern the design.

The exact optimum solution is obtained by writing the Lagrangian function and solving the equations resulting from setting its partial derivatives to zero. The following Mathematica statements accomplish this task.

The Lagrangian function is defined by the following line. The `Expand` function is used to get the result in a simple form.

```
LC = Expand [f + u6 g6 + u7 g7];
```

The system of equations is obtained by evaluating partial derivatives of the Lagrangian. Mathematica function `D` computes partial derivative of the first argument with respect to the second argument. The resulting system of equations is defined as a list called *eqns*.

```
eqns = {D[LC,As]==0, D[LC,d]==0, g6==0,
  g7==0};
```

Note the subtle but important difference between = and == operators in Mathematica. If we had used `D[L,As] = 0` in the above expression, then Mathematica will define the partial derivative of *LC* with respect to *As* as 0 which is obviously not what we want. With the symbol == the left hand side is evaluated independently and is set logically equivalent to the right hand side and thus defines an equation.

The solution of these equations is obtained using `Solve` function. The first argument refers to a list of equations and the second argument gives a list of variables for which the solution is desired. The `Solve` function tries to find all possible solutions. Another function `NSolve`, which uses a numerical method, can also be used to find solutions if the `Solve` fails for some reason.

```
soln = Solve[eqns,{As, d, u6, u7}]
{{As = 16.0721556, d = 24.,
  u7 = 195.2521008, u6 = 4275.859697}}
```

Since this is a two variable problem and the two active constraints are known from the graph, the solution can be obtained even more easily by simply solving for the intersection of the two constraint functions as follows.

```
soln = NSolve[{g6==0, g7==0},{As, d}]
{{d -> 24., As -> -171.913},
  {d -> 24., As -> 16.0722}}
```

Mathematica has found two solutions. The first solution is not acceptable on physical grounds because *As* is negative. The second solution is acceptable which is same as that obtained by solving four nonlinear equations.

This optimum solution is substituted into the objective and the constraint functions to get their values at the optimum. The symbol /. (slash-dot) is used by Mathematica to designate substitution operation. The right hand side of this operator must be of the form {*var* -> *value*}. The quantity on the left hand side is evaluated by substituting given value for each occurrence of *var*. The `Solve` command returns the solution in the form suitable for the substitution operator. The `soln[[2]]` symbols refers to the second solution. The `Chop` function rounds-off small numbers to 0.

```
Chop[{Ns, As, As*rhoS*12, d, f, g1, g2, g3,
  g4, g5, g6, g7} /.
soln[[2]]]
{61.5758, 16.0722, 54.69, 24., 2926.55,
  -297.599, -2714.99, -2.53037, 0,
  -0.397262, 0, 0}
```

The resulting constraint values must be inspected to see if they are all negative or zero (and thus there is no constraint violation). This condition is clearly satisfied with the given solution. Thus the theoretical optimum design is as follows.

N_s = 61.58
A_s = 16.07 in.^{2}
Weight = 54.69 lbs/ft
d = 24 in.
Cost = 2926.55

Final design and constraint check

Based on the optimum design choose a W24×55 section with 62 shear connectors. The following lines define actual section properties and evaluate corresponding constraints functions values. The Mathematica function `N` is used to get numerical values.

```
(*Choose W24x55*)

{Ns, As, d, tw, bf, tf, Is, Zs} =
  {62, 16.2, 23.57, 0.396, 7.005, 0.505,
  1350, 134};

N[{fn, Ns, f, ng1, ng2, ng3, ng4, ng5,
  ng6, ng7}]
{4.6988, 62., 2949., -0.0190074, -0.416458,
  -0.559314, -0.0012848, -0.276977,
  0.00550224, -0.0179167}
```

The natural frequency is computed to check the DLF_{max} value. The vibration constraint is slightly violated (0.00550224) but this violation is with $DLF_{max} = 1$. For $f_n = 4.7$ the actual value of $DLF_{max} = 0.6835$. The vibration constraint is evaluated for this DLF as follows.

```
DLFmax = .6835; N[ng6]
-0.114927
```

The constraint is satisfied. Thus the design is satisfactory. Use W24×55 with 62 shear connectors. Incidentally this is the same design as that given in LRFD *Manual*.[1]

In addition to the optimum design, Figure 4 gives a great deal of insight into the problem. For example it shows that if the vibration constraint is relaxed then the optimum is controlled by the flexural strength for beam depths greater than 16 inches. For shallower beams the deflection constraint becomes the controlling factor. The constraint g_2 is clearly not a factor in the design. This tells a designer that there is really no need to consider a shored construction for the given loading and span.

Example 2—Partial Composite Design: *As* & *d* Variables

This example is the same as Example 1 except that the number of shear studs is limited to 40 which is less than that required for a full composite action. A partial composite design is therefore desired. The following Mathematica statements define parameters for this problem.

```
wd = 0.9/12; wdSuperimposed = 0;
constructionLive = 0.2/12; wl = 2.5/12;
  L = 40*12;
Csm = 12; Fy = 50; fc = 4; Es = 29000;
  n = 9;
b = 120; tc = 4.5; deckDepth = 3;
DLFmax = 0.7; Neff = 1; maxDamping = 4;
bf = 7; Ns = 40; Qn = 26.1; dmax = 24;
```

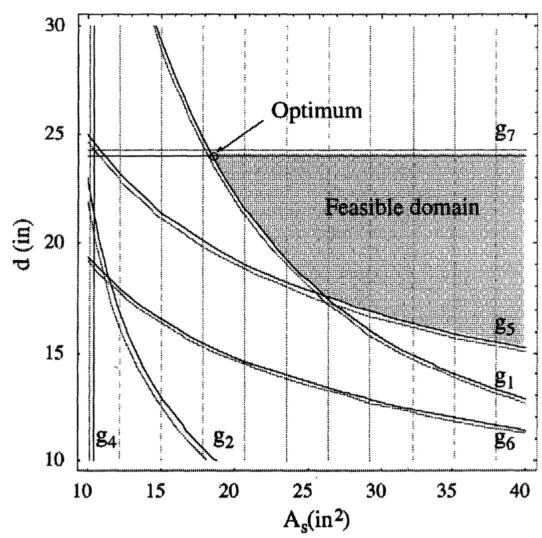

Fig. 5. Graphical solution of Example 2.

Graphical solution is obtained using exactly the same commands as those for Example 1. The resulting plot is shown in Figure 5. From the figure it is clear that the optimum is at the intersection of constraints g_1 and g_7.

It is interesting to compare Figure 5 with Figure 4. The only difference in the two cases is in the number of shear studs. Figure 5 shows that the vibration constraint is not a factor for this partially composite design unless the depth is less than approximately 18 inches. With fewer studs the natural frequency is lowered and hence the demand on the required damping is lowered.

The actual solution is obtained by writing the Lagrangian function and solving the equations resulting from setting its partial derivatives to zero.

```
LC = f + u1 g1 + u7 g7;

eqns = {D[LC,As]==0, D[LC,d]==0, g1==0,
  g7==0};

soln = NSolve[{g1==0, g7==0},{As, d}]

{{d -> 24., As -> 18.4853}, {d -> 24.,
  As -> 331.938}}
```

The solution is obtained by simply solving for the intersection of the two constraint functions as before. Mathematica has found two solutions. The second solution is not acceptable because *As* is too large. The first solution is acceptable. This optimum solution is substituted into the objective and the constraint functions to get their values at the optimum.

```
Chop[{Ns, As, As*rhoS*12, d, f, g1, g2, g3,
  g4, g5, g6, g7} /.
soln[[1]]]
{40, 18.4853, 62.9013, 24., 2996.05, 0,
  -3759.87, -3.22059, -0.574662, -0.323618,
  -0.412543, 0}
```

All constraints are satisfied since their values are less than zero. Thus the theoretical optimum design is as follows.

N_s = 40
A_s = 18.49 in.^{2}
Weight = 62.9 lbs/ft
d = 24 in.
Cost = 2996.05

Final design and constraint check

Based on the optimum design choose a W24×68 section. Since area of a W24×68 is slightly larger area than the optimum, it is possible to reduce the number of shear connectors. The following line defines its section properties.

```
(*Choose W24x68*)

{Ns, As, d, tw, bf, tf, Is, Zs} =
```

```
{38, 20.1, 23.73, 0.415, 8.965, 0.585,
  1830. 177};
```

The constraint values corresponding to actual design are evaluated by the following line.

```
N[{fn, Ns, f, ng1, ng2, ng3, ng4, ng5, ng6,
  ng7}]
{4.60815, 38., 3191.83, -0.0409526,
  -0.553424, -0.729902, -0.567875,
  -0.255043, -0.112207, -0.01125}
```

All constraints are satisfied and the PNA is in the flange. The design is acceptable. As expected the cost of this design is little more than the cost of design in Example 1.

Example 3—Partial Composite Design: *As* & *Ns* as Variables

This example is same as Example 1 except that beam depth is fixed at 27 inches and lower strength materials, $Fy = 36$ ksi and $fc = 3$ ksi, are used. The relative cost coefficient C_{sm} is set to 10. The two design variables are A_s and N_s. The following Mathematica statements define this problem.

```
wd = 0.9/12; wdSuperimposed = 0;
constructionLive = 0.2/12; wl = 2.5/12:
  L = 40*12;
Csm = 10; Fy = 36; fc = 3; Es = 29000;
  n = 9;
b = 120; tc = 4.5; deckDepth = 3;
DLFmax = 0.7; Neff = 1; maxDamping = 4;
bf = 7; Qn = 26.1; dmax = 27; d = 27;
```

Graphical solution is obtained using exactly the same commands as those for Example 1. The resulting plot is shown in Figure 6. From the figure it is clear that the optimum is at the intersection of constraints g_1 and g_4. Recall that the constraint g_4 is simply to keep the PNA in beam flange. The optimum is achieved when $g_4 = 0$ meaning that it is a full composite design. The deflection and vibration constraints are not a factor in this example.

The solution is obtained by simply solving for the intersection of the two constraint functions g_1 and g_4.

```
soln = NSolve[{g1==0, g4==0},{As, Ns}]
{{NR -> 56.4335, As -> 20.4572},
  {Ns -> 913.375, As -> 331.098}}
```

Mathematica has found two solutions. The first solution is acceptable. This optimum solution is substituted into the objective and the constraint functions to get their values at the optimum.

```
Chop[{Ns, As, As*rhoS*12, d, f, g1, g2, g3,
  g4, g5, g6, g7} /. soln[[1]]]
{56.4335, 20.4572, 69.6112, 27, 3348.78, 0,
  -2917.12, -2.09328, 0, -0.696464,
  -0.638324, 0}
```

All constraints are satisfied since their values are less than zero. Thus the theoretical optimum design is as follows.

N_s = 56.43
A_s = 20.46 in.^{2}
Weight = 69.61 lbs/ft
d = 27 in.
Cost = 3348.78

Final design and constraint check

The smallest W27 section available is W27×84 with $A_s = 24.8$ in.^{2} which is considerably larger than the theoretical optimum. If this section is chosen anyway then the number of shear studs can be reduced. From Figure 5, with $A_s = 24.8$ the optimum number of studs is around 35. Thus define the following design.

```
(*Choose W27x84*)

{Ns, As, d, tw, bf, tf, Is, Zs} = {35,
  24.8, 26.71, .46, 9.96, .64, 2850, 244};
```

The constraint values corresponding to actual design are evaluated by the following line.

```
N[{fn, Ns, f, ng1, ng2, ng3, ng4, ng5, ng6,
  ng7}]
{5.52584, 35., 3725.55, -0.016541,
  -0.544242, -0.668301, -0.608057,
  -0.487514, -0.108764, -0.0107407}
```

For $f_n = 5.5$ the actual value of $DLF_{max} = 0.7819$. The vibration constraint is evaluated for this DLF as follows.

```
DLFmax = 0.7819; N[ng6]
```

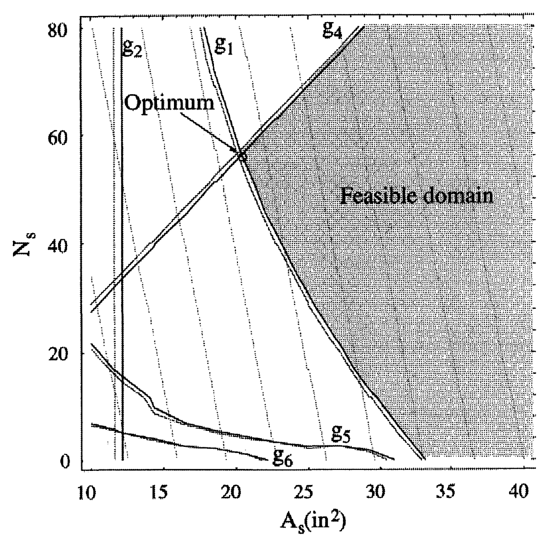

Fig. 6. Graphical solution of Example 3.

```
-0.127001
```

The design is acceptable. Use W27×84 beam with 35 shear studs.

Example 4—An Example with Smaller Loads

As a different example a 25-ft long lightly loaded beam is considered in this example. The two design variables are A_s and N_s. For the vibration constraint *Neff* is computed from the equation suggested for secondary beams. The following Mathematica statements define this problem.

```
wd = 0.356/12; wdSuperimposed = 0.13/12;

constructionLive = 0/12; wl = 0.52/12;
  L = 25*12;
Csm = 10; Fy = 36; fc = 3; Es = 29000;
  n = 9;
b = 90; tc = 3.25; deckDepth = 1.5;
DLFmax = 0.9; maxDamping = 4;
bf = 7; Qn = 21; dmax =20; d=18;
S = 120;
Neff := 2.97 - S/(17.3 (tc + deckDepth/2))
  + L^4 /(1.35 Es 1000 Itr);
```

Graphical solution is obtained by using exactly the same commands as those for Example 1. The resulting plot is shown in Figure 7. From the figure it is clear that the optimum is at the intersection of constraints g_4 and g_6.

The fully composite design with vibration constraint controlling the design is optimum.

```
Ns := 2 As Fy/Qn

soln = NSolve[g6==0, As]
```

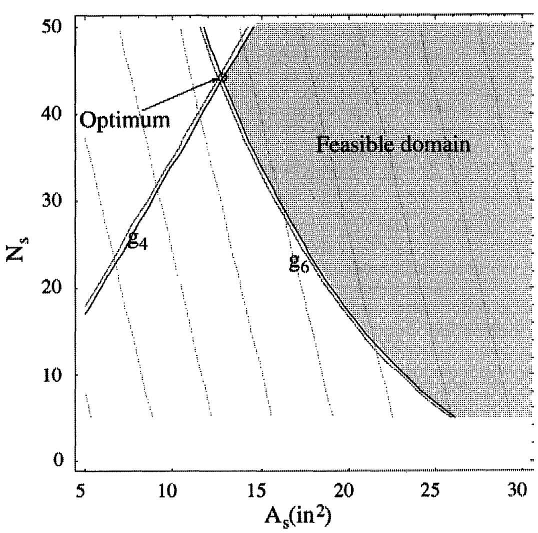

Fig. 7. Graphical solution of Example 4.

```
{{As -> -113.81}, {As -> -72.85},
  {As -> 0.2865}, {As -> 13.66}}
```

This optimum solution is substituted into the objective and the constraint functions to get their values at the optimum.

```
Chop[{Ns, As, As*rhoS*12, d, f, g1, g2, g3,
  g4, g5, g6, g7} /. soln[[4]]]
{46.8261, 13.6576, 46.4738, 18, 1630.11,
  -3919.74, -2543.67, -1.10763, 0,
  -0.760496, 0, -2}
```

All constraints are satisfied since their values are less than zero. Thus the theoretical optimum design is as follows.

N_s	= 46.83
A_s	= 13.66 in.^{2}
Weight	= 46.47 lbs/ft
d	= 18 in.
Cost	= 1630.11

Final Design And Constraint Check

Select W18×50.

```
{Ns, As, d, tw, bf, tf, Is, Zs} = {50,
  14.7, 17.99, 0.355, 7.495, 0.57, 800,
  101};
```

The constraint values corresponding to actual design are evaluated by the following line.

```
N[{fn, Ns, f, ng1, ng2, ng3, ng4, ng5, ng6,
  ng7}]
{12.1449, 50., 1750.52, -0.75545,
  -0.860416, -0.296129, -0.00778297,
  -0.918111, -0.0113194, -0.1005}
```

For f_n = 12.15 the actual value of DLF_{max} = 1.2879. The vibration constraint is evaluated for this *DLF* as follows.

```
DLFmax = 1.2879; N[ng6]
0.145427
```

The vibration constraint is violated. Therefore we need a larger section. A revised optimum design with d = 21 inches that satisfies vibration constraint is as follows.

```
(*Choose W21x122*)

DLFmax = 1.3;

{Ns, As, d, tw, bf, tf, Is, Zs} = {25,
  35.9, 21.68, 0.6, 12.39, 0.96, 2960, 307};

N[{fn, Ns, f, ng1, ng2, ng3, ng4, ng5, ng6,
  ng7}]
{16.0217, 25., 3353.99, -0.8813, -0.945919,
  -0.648064, -1.15449, -0.957712,
  -0.00116719, -0.1328}
```

This design may be accepted or damping may be increased. Following design is obtained with *maxDamping* = 4.5.

```
(*Choose W21x44*)

maxDamping = 4.5;

DLFmax = 1.3:

{Ns, As, d, tw, bf, tf, Is, Zs} = {44, 13,
  20.66, 0.35, 6.5, 0.45, 843, 95.4};

N[{fn, Ns, f, ng1, ng2, ng3, ng4, ng5, ng6,
  ng7}]
{12.799, 44., 1405.9, -0.753282, -0.854328,
  -0.380593, -0.0128205, -0.925594,
  0.00550286, -0.1736}

DLFmax = 1.3185; N[ng6]
0.0119
```

The constraint is very slightly violated but the design may be acceptable especially considering uncertainty in the damping values. When comparing the cost of this design with others, the cost of providing additional damping should be considered.

CONCLUDING REMARKS

A formulation for optimum cost design of composite steel beams is presented. Mathematica is used to formulate specific constraint expressions and to obtain optimum solution. No detailed knowledge of optimization techniques is necessary to use the procedure developed. A designer is free to modify cost coefficients, add or delete constraints, treat any variable as a specified constant or an optimization variable. The procedures are explained through several numerical examples.

For two variable problems the graphical solutions are readily obtained. In addition to giving the optimum design these graphs contain a wealth of useful information that a designer can use to make practical decisions.

REFERENCES

1. American Institute of Steel Construction, *Manual of Steel Construction, Load and Resistance Factor Design, Vol. I*, 2nd Ed., 1994.
2. Ito, Mitsuru, "Minimum-weight design of continuous composite girders," *Journal of Structural Engineering*, Vol. 119, April 1993, p. 1297–311.
3. American Association of State Highway and Transportation Officials, *Standard Specifications for Highway Bridges*, 1989.
4. Zahn, M. C., "The Economies of LRFD in Composite Floor Beams," AISC *Engineering Journal*, 2nd Qtr., Vol. 24, 1987.
5. Lorenz, R. F., "Understanding Composite Beam Design Methods Using LRFD," AISC *Engineering Journal*, 1st Qtr., Vol. 25, 1988.
6. Wolfram, S., *Mathematica: A System for Doing Mathematics by Computer*, 2nd Ed., Addison-Wesley, Redwood City, CA, 1988.
7. Segui, W. T., *LRFD Steel Design*, PWS Publishing Company, Boston, MA, 1994.
8. Murray, T. M., "Building Floor Vibrations," AISC *Engineering Journal*, 3rd Qtr., Vol. 28, 1991.
9. Bhatti, M. Asghar, "Approximate Equations for Cross Section Properties of W Shapes," submitted for publication in *Engineering Journal*, 1995.
10. Arora, J. S., *Introduction to Optimum Design*, McGraw-Hill, New York, NY, 1989.