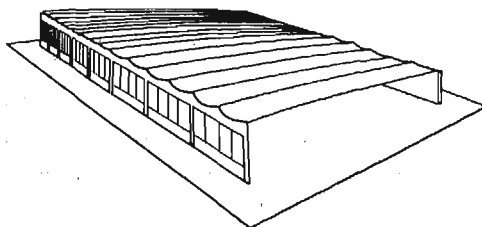


OPTYMALIZACJA KSZTAŁTU ŁUKU JAKO PRZEKROJU DŹWIGARÓW SKLEPIENIOWYCH

ZDZISŁAW K. LEŚNIAK (WARSZAWA)

1. Wstęp

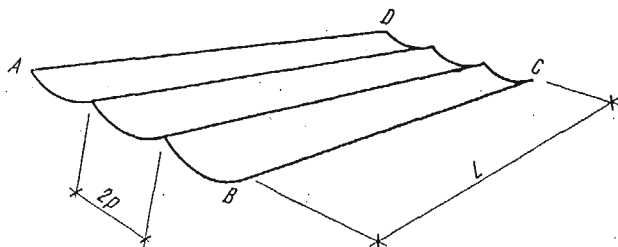
W wielu konstrukcjach budowlanych stosuje się przekrycia dachowe w postaci sklepień o zarysie łukowym. I tak np. dachy obiektów przemysłowych (hal fabrycznych, magazynów itd.) miewają postać dźwigarów sklepieniowych o przekroju poprzecznym złożonym



Rys. 1. Dach budynku przemysłowego w postaci dźwigarów sklepieniowych

z szeregu łuków (rys. 1). Dźwigary te pracują przede wszystkim na zginanie jako belki o rozpiętości L oparte na podporach w płaszczyznach AB oraz CD (rys. 2).

Względy ekonomiczne nakazują optymalizację tych konstrukcji. Optymalne dźwigary sklepieniowe powinny zapewniać spełnienie wymagań funkcjonalnych, wytrzymałościowych itp. w stopniu zadowalającym, a przy tym powinny być najlepsze z punktu widzenia

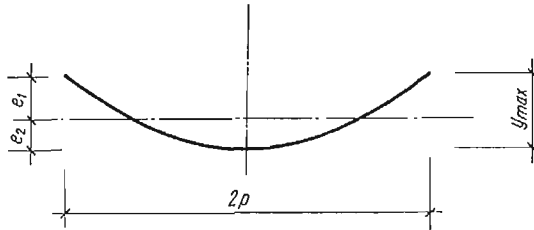


Rys. 2. Dźwigar sklepieniowy

pewnej ich cechy przyjętej za kryterium optymalności. Jako kryterium optymalności przyjmuje się najczęściej ogólną ekonomiczność budowli.

W zastosowaniu do dźwigara sklepieniowego kryterium to da się sprowadzić do wymagania użycia minimum materiału w dźwigarze przy zachowaniu jego nośności nie mniejszej od wymaganej (wariant 1) bądź też do żądania uzyskania maksimum wytrzymałości dźwigara przy użyciu nie większej niż określonej ilości materiału (wariant 2). Zagadnienie to ujmijemy teraz matematycznie.

Rozważmy jeden element przekroju poprzecznego dźwigara sklepieniowego, mianowicie łuk o szerokości 2ρ i wysokości y_{\max} (rys. 3).



Rys. 3. Przekrój poprzeczny elementu łukowego dźwigara sklepieniowego

Pod wpływem zginania wywołanego obciążeniem naprężenia w przekroju dźwigara nie powinny przekroczyć wartości dopuszczalnych, czyli

$$(1.1) \quad M \geq \frac{Be}{k},$$

gdzie B oznacza moment zginający, przypadający na jeden element dźwigara (rys. 3), e odpowiednio e_1 i e_2 , tj. odległości skrajnych punktów łuku (przekroju dźwigara) od osi obojętnej, M moment bezwładności łuku (przekroju dźwigara) względem jego osi ciężkości, k naprężenie dopuszczalne w materiale łuku.

Kryterium optymalizacji

wariant 1

$$(1.2) \quad l = \text{minimum}, \quad (l \text{ jest długością łuku})$$

przy spełnieniu warunków $\sigma \leq k$, co sprowadza się do warunku

$$(1.3) \quad M \geq M_d$$

ponadto

$$(1.4) \quad y \leq y_{\max},$$

gdzie M_d jest daną wartością momentu bezwładności łuku, y oznacza rzędne osi łuku y_{\max} dane ograniczenie wysokości łuku, reszta oznaczeń jak wyżej.

wariant 2

$$(1.5) \quad M = \text{maximum} \quad (M = \text{j.w.})$$

przy spełnieniu warunków

$$(1.6) \quad l \leq l_d, \quad y \leq y_{\max},$$

gdzie l oznacza długość łuku, l_d daną wartość, pozostałe oznaczenia — jak wyżej.

Zadanie optymalizacji polega na znalezieniu takiego kształtu łuku czyli takiego jego równania, aby były spełnione warunki (1.3) i (1.4) przy wyborze kryterium optymalizacji

wg wariantu 1 lub warunki (1.6) w przypadku wariantu 2, a ponadto aby funkcja kryterium optymalizacji osiągnęła ekstremum. W wariantcie 1 funkcja-kryterium (1.2) powinna osiągnąć minimum, a w wariantcie 2 funkcja-kryterium (1.5) powinna uzyskać wartość maksymalną.

Wielkość M (moment bezwładności łuku) występująca we wzorze (1.5) jako funkcja-kryterium optymalizacji oraz we wzorze (1.3) jako warunek zależy od funkcji y (równania łuku), jest więc funkcjonalem.

Przy zastąpieniu znaków nierówności we wzorach (1.3), (1.4) i (1.6), znakami równości obydwa warianty optymalizacji są równoważne tworząc zagadnienie dualne programowania matematycznego [1].

Wariant 2 optymalizacji po zastąpieniu we wzorach (1.6) znaków nierówności znakami równości stanowi izoperymetryczne zagadnienie rachunku wariacyjnego, które nie jest dotychczas rozwiązane¹⁾.

Praca niniejsza podaje sposób rozwiązania tego problemu przez potraktowanie go jako zagadnienia programowania matematycznego i zastosowanie metody Monte Carlo do jego rozwiązania. Ponadto podano numeryczne rozwiązanie pewnego przypadku tego zagadnienia.

2. Sformułowanie matematyczne zagadnienia programowania

Zadanie optymalizacji kształtu łuku przedstawimy teraz jako zagadnienie programowania matematycznego.

Założmy równanie kształtu łuku w postaci szeregu potęgowego

$$(2.1) \quad y = a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0$$

i potraktujmy współczynniki a_n ($n = 0, 1, \dots, n$) jako zmienne decyzyjne, których wartości mamy określić w postępowaniu optymalizacyjnym.

Przyjmując wariant 2 optymalizacji łuku możemy sformułować zadanie optymalizacji łuku następująco:

Należy znaleźć takie wartości zmiennych a_n ($n = 0, 1, \dots, n$), aby moment bezwładności M łuku względem jego osi ciężkości był maksymalny przy równoczesnym spełnieniu warunków, że długość łuku l nie przekroczy danej wartości l_d , a wysokość łuku nie będzie większa od danego wymiaru y_{\max} .

Z uwagi na symetrię łuku względem osi pionowej wystarczy rozpatrywać tylko jego połowę (rys. 4) przechodzącą przez początek układu współrzędnych. Dzięki temu we wzorze (2.1) zniknie wyraz wolny a_0 i wyraz $a_1 x$. Równanie łuku będzie więc miało postać

$$(2.2) \quad y = a_n x^n + a_{n-1} x^{n-1} + \dots + a_2 x^2.$$

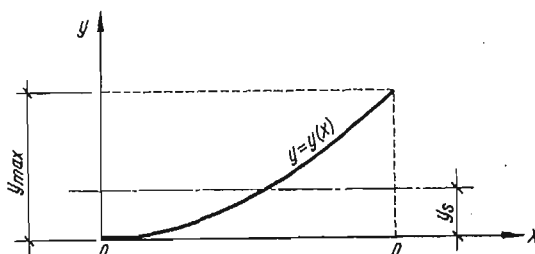
Grubość dźwigarów sklepieniowych jest nieznaczną w porównaniu z szerokością łuku. W dalszym ciągu będziemy więc rozpatrywać samą oś łuku jako przekrój dźwigara sklepieniowego (grubość 1).

¹⁾ Autor dziękuje doc. d-rowsi inż. Zbigniewowi Mazurkiewiczowi za zwrócenie uwagi na nierozwiązany dotychczas problem optymalizacji łuku.

Położenie środka ciężkości y_s łuku znajdziemy z warunku, że moment statyczny łuku względem swego środka ciężkości równa się zeru:

$$(2.3) \quad y_s = \frac{S}{l} = \frac{\int_0^p y \sqrt{1+y'^2} dx}{\int_0^p \sqrt{1+y'^2} dx}$$

gdzie $y = y(x)$, p oznacza połowę szerokości łuku (granica całkowania), S moment statyczny łuku względem osi x , l długość łuku.



Rys. 4. Łuk optymalizowany

Moment bezwładności M łuku względem osi ciężkości wynosi

$$(2.4) \quad M = \int_0^p (y - y_s)^2 \sqrt{1+y'^2} dx,$$

gdzie $y = y(x)$.

Zadanie optymalizacji łuku polega na znalezieniu maksimum warunkowego funkcji M . Warunkami są

$$(2.5) \quad l = \int_0^p \sqrt{1+y'^2} dx \leq l_d$$

oraz

$$(2.6) \quad y \leq y_{\max},$$

gdzie l_d i y_{\max} są z góry danymi liczbami. Krótko mówiąc należy znaleźć $M =$ maksimum przy $l \leq l_d$, $y \leq y_{\max}$. Funkcje M , l i y są nieliniowe względem x .

Zagadnienie to nie daje się rozwiązać za pomocą rachunku różniczkowego. Zastosujemy więc do jego rozwiązania metody programowania matematycznego.

Jak zobaczymy niżej, w przypadku kształtu łuku wyrażonego wielomianem (2.2) całki występujące we wzorach (2.3)–(2.5) nie dają się ogólnie obliczyć w sposób elementarny i przedstawić w postaci zamkniętych wzorów. Muszą one być określane w sposób przybliżony metodami numerycznymi. Ten fakt wyklucza zastosowanie niektórych metod programowania matematycznego. Nie jest np. możliwe zastosowanie metod gradientowych do znalezienia maksimum warunkowego funkcji M .

Do rozwiązania zagadnienia optymalizacji łuku zastosujemy metodę Monte Carlo [2] stosując komputer do obliczeń numerycznych.

3. Przykład optymalizacji łuku (obliczony numerycznie)

Równanie łuku wyraża wielomian czwartego stopnia (por. rys. 4)

$$(3.1) \quad y = ax^4 + bx^3 + cx^2.$$

Położenie środka ciężkości

$$(3.2) \quad y_s = \frac{\int_0^p (ax^4 + bx^3 + cx^2) \sqrt{1 + (4ax^3 + 3bx^2 + 2cx)^2} dx}{\int_0^p \sqrt{1 + (4ax^3 + 3bx^2 + 2cx)^2} dx}.$$

Moment bezwładności łuku względem osi ciężkości

$$(3.3) \quad M = \int_0^p (ax^4 + bx^3 + cx^2 - y_s)^2 \sqrt{1 + (4ax^3 + 3bx^2 + 2cx)^2} dx.$$

Zadanie optymalizacji łuku polega na znalezieniu takich wartości zmiennych decyzyjnych a, b, c , dla których wystąpi maksimum funkcji M przy równoczesnym spełnieniu następujących warunków:

warunek 1

$$(3.4) \quad l = \int_0^p \sqrt{1 + (4ax^3 + 3bx^2 + 2cx)^2} dx \leq l_d;$$

za l_d przyjmujemy ćwiartkę długości koła o promieniu p : $l_d = \frac{\pi p}{2}$;

warunek 2

$$(3.5) \quad y = ax^4 + bx^3 + cx^2 \leq y_{\max}.$$

Za y_{\max} będziemy przyjmować kolejno $y_{\max} = p$ i $y_{\max} = p/2$.

Jako wymiar połowy szerokości łuku (por. rys. 4) przyjmujemy $p = 1$.

Do rozwiązania zagadnienia zastosowano metodę Monte Carlo. Szczegółowy opis metody Monte Carlo wraz z opisem generatora liczb losowych podany jest w pracy [2].

W omawianym tu przypadku optymalizacji łuku całki nie dały się obliczyć metodami elementarnymi i były liczone numerycznie wg wzoru

$$(3.6) \quad \int_0^p f(x) dx = \frac{h}{2} (y_0 + 2y_1 + 2y_2 + \dots + 2y_{n-1} + y_n),$$

gdzie $y_i = y(x_i)$, $i = 0, 1, \dots, n$

$$x_0 = 0, \quad x_i = x_{i-1} + h, \quad h = p/n,$$

czyli

$$y_0 = f(x_0) = f(0), \quad y_1 = f(h), \quad y_2 = f(2h), \dots$$

Dobór odpowiedniej liczby n podziału całkowanego odcinka nastąpił doświadczalnie przy użyciu komputera przez dobieranie różnych wartości n i porównanie otrzymanych wyników. Okazało się, że stosowanie mniejszych odcinków niż $h = p/40$ nie daje już zwiększenia dokładności drukowanych wyników. Dla zapewnienia jak największej dokładności zastosowano ostatecznie jednak $h = p/100$.

Losowanie wartości a , b , c następowało najpierw w kostce $\langle 0; 2, 5 \rangle$, a następnie po otrzymaniu obrazu co do wartości a , b , c dających wysokie wartości M zmniejszono ją do przedziału $\langle 0; 1 \rangle$. Przyczyniło się to do poprawienia przybliżenia optimum.

Obliczenie przeprowadzono na komputerze GIER. Program optymalizacji łuku napisany w języku GIER ALGOL III ma postać następującą:

```

PROGRAM ARCH(X $\sqrt{4}$ ):
begin comment: program oblicza metoda Monte Carlo współczynniki łuku
                a $\times$ x $\sqrt{4}$ +b $\times$ x $\sqrt{3}$ +c $\times$ x $\sqrt{2}$ , zapewniające maksimum momentu bezwładności
                łuku względem jego osi ciężkości;
integer i,n,N,1;
real h,C1,C2,C3,p,ys,M,y $\max$ ,aa,ad,ag,bb,cc;
boolean random,d,c,b,a; comment: procedura generowania liczb losowych o rozkładzie
                                równomiernym w przedziale [0,1], wywoływana
                                przez instrukcje gier(random);

pack(a,0,20,0,20,39,1);
pack(b,0,9,142,10,19,111,20,29,155,30,39,961);
pack(c,0,19,1,20,25,15,26,35,17,36,39,1,40,41,3);
pack(d,0,9,3,10,25,26,26,30,5,31,39,0,40,41,0);
pack(random,0,9,4,10,19,3,20,25,27,26,29,2,30,36,25,37,39,2,40,41,2);
input(ad,ag,n,p,N,y $\max$ );
1:= 0;
M:= 0;
begin
array y,y1[0:n];
h:= p/n;
L:
1:= 1+1;
C1:= C2:= C3:= 0;
if 1>N then
go to A;
aa:= (ag-ad) $\times$ gier(random)+ad;
bb:= (ag-ad) $\times$ gier(random)+ad;
cc:= (ag-ad) $\times$ gier(random)+ad;
if aa $\times$ p $\sqrt{4}$ +bb $\times$ p $\sqrt{3}$ +cc $\times$ p $\sqrt{2}$  > y $\max$  then go to L;
for i:= 0 step 1 until n do
begin
y1[i]:= sqrt(1+(4 $\times$ aa $\times$ (hx1) $\sqrt{3}$ +3 $\times$ bb $\times$ (hx1) $\sqrt{2}$ +2 $\times$ cc $\times$ (hx1) $\sqrt{2}$ );
y[i]:= aa $\times$ (hx1) $\sqrt{4}$ +bb $\times$ (hx1) $\sqrt{3}$ +cc $\times$ (hx1) $\sqrt{2}$ ;
C2:= C2+y1[i];
C1:= C1+y[i] $\times$ y1[i];
end;
C1:= (C1-(y[0] $\times$ y1[0]+y[n] $\times$ y1[n])/2) $\times$ h;
C2:= (C2-(y1[0]+y1[n])/2) $\times$ h;
if C2 > 1.5708 $\times$ p then
go to L;
ys:= C1/C2;
for i:= 0 step 1 until n do
begin
y[i]:= (aa $\times$ (hx1) $\sqrt{4}$ +bb $\times$ (hx1) $\sqrt{3}$ +cc $\times$ (hx1) $\sqrt{2}$ -ys) $\sqrt{2}$  $\times$ sqrt(1+(4 $\times$ aa $\times$ (hx1) $\sqrt{3}$ +3 $\times$ bb $\times$ (hx1) $\sqrt{2}$ +
+2 $\times$ cc $\times$ (hx1) $\sqrt{2}$ );
C3:= C3+y[i];
end;
C3:= (C3-(y[0]+y[n])/2) $\times$ h;
if C3>M then
begin
M:= C3;
output({ndddd,ddd},M); outsp(5);
output({nd,ddd},aa); outsp(5);
output({nd,ddd},bb,outsp(5),cc); outsp(5);
output({ndddd,1},out);
end;
go to L;
end;
end program arch(x $\sqrt{4}$ );

```

Wyniki obliczeń podane przykładowo dla przypadku

$$y_{\max} = p = 1, \quad n = 100$$

są następujące:

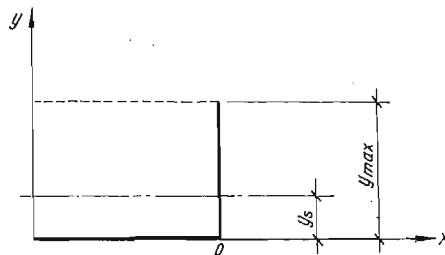
Wyniki (ARCH x^4)

M	a	b	c	kolejna liczba losowania
0,138	0,5671	0,2008	0,1820	5
0,143	0,3789	0,3637	0,2291	104
0,148	0,5991	0,2650	0,1070	330
0,150	0,0297	0,9267	0,0264	1170
0,151	0,5549	0,4035	0,0163	1391
0,152	0,4557	0,2105	0,3319	1406
0,153	0,1490	0,6471	0,2038	1907
0,156	0,3573	0,5105	0,1299	3400
0,158	0,5919	0,3279	0,0767	3927
0,159	0,6826	0,2141	0,1009	9078

Łączna liczba losowań zespołów wartości dla zmiennych a, b, c wynosiła 15 563. Najlepszy wynik uzyskano w 9078 losowaniu. Następne losowania nie dały już poprawy wyniku. Czas liczenia wynosił ok. 3,5 godziny.

Optymalny kształt łuku dany jest więc wyrażeniami (dla przypadku b) nie podano tu wyników obliczeń):

$$(3.7) \quad \begin{aligned} \text{a) dla } y_{\max} = p: & \quad y = 0,6826x^4 + 0,2141x^3 + 0,1009x, \\ \text{b) dla } y_{\max} = p/2: & \quad y = 3,447x^4 + 0,1187x^3 + 0,365x. \end{aligned}$$



Rys. 5. Łuk o kształcie linii łamanej

Dla porównania przeprowadzono obliczenia dla łuków:

1. O kształcie linii łamanej (rys. 5) wg poniższych wzorów napisanych dla $y_{\max} = p$. W przypadku $y_{\max} = p/2$ wzory są podobne.

$$l = 2p,$$

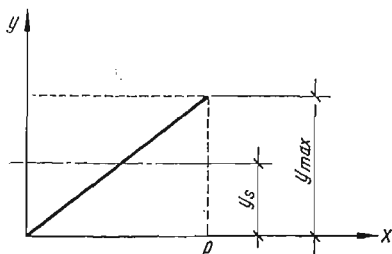
$$y_s = \frac{p^2}{2 \cdot 2p} = \frac{p}{4},$$

$$M = p \left(\frac{p}{4} \right)^2 + \frac{p^3}{12} + p \left(\frac{p}{2} \right)^2 = 0,20833 p^3 \quad (2).$$

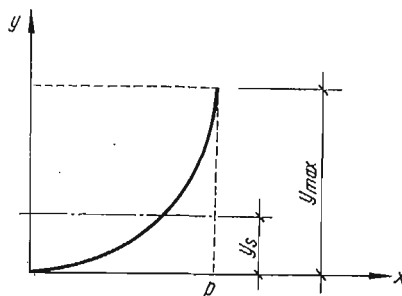
2. O kształcie odcinka prostej (rys. 6). Dla $y_{\max} = p$ otrzymuje się wzory

$$l = p\sqrt{2}, \quad y_s = \frac{p^2\sqrt{2}}{2p\sqrt{2}} = \frac{p}{2},$$

$$M = \int_0^p (x-p/2)^2 \sqrt{2} dx = 0,1178 p^3.$$



Rys. 6. Łuk o kształcie odcinka prostej



Rys. 7. Łuk kołowy

3. O kształcie łuku koła (rys. 7). Odpowiednie wzory mają postać:

$$y = p - \sqrt{p^2 - x^2}, \quad y' = \frac{x}{\sqrt{p^2 - x^2}}, \quad l = \frac{\pi p}{2},$$

$$y_s = \frac{\int_0^p \left(p - \sqrt{p^2 - x^2} \right) \frac{p}{\sqrt{p^2 - x^2}} dx}{\pi p/2} = \frac{p^2 \left[\arcsin \frac{x}{p} \right]_0^p - p x \left|_0^p \right.}{\pi p/2} = \frac{\frac{p^2}{2} (\pi - 2)}{\pi p/2} = \frac{p(\pi - 2)}{\pi}$$

$$\begin{aligned} M &= \int_0^p \frac{(p^2 - 2p\sqrt{p^2 - x^2} + p^2 - x^2 - 2py + 2y_s\sqrt{p^2 - x^2} + y_s^2)p}{\sqrt{p^2 - x^2}} dx = \\ &= p \left\{ (2p^2 - 2py_s + y_s^2) \left[\arcsin \frac{x}{p} \right]_0^p + (2y_s - 2p)x \left|_0^p \right. - \left[\frac{p^2}{2} \arcsin \frac{x}{p} - \frac{x}{2} \sqrt{p^2 - x^2} \right]_0^p \right\} = \\ &= p \left[(2p^2 - 2py_s + y_s^2) \frac{\pi}{2} + 2p(y_s - p) - \frac{p^2\pi}{4} \right] = 0,1488 p^3. \end{aligned}$$

Ponadto przeprowadzono optymalizację kształtu łuku określonego równaniem

$$(3.8) \quad y = ax^2.$$

²⁾ Przypadek łuku w kształcie linii łamanej wymaga uwzględnienia ponadto innych wzorów wytrzymałościowych, które tu pominięto, gdyż ten przypadek traktowany jest tylko porównawczo jako graniczny kształt łuku.

Optymalizację przeprowadzono metodą Monte Carlo przy użyciu komputera GIER analogicznie jak w przypadku łuku wyrażonego wielomianem czwartego stopnia. Wszystkie całki były liczone numerycznie.

W przypadku $y_{\max} \leq p = 1$ uzyskano dla kostki $\langle 0; 2, 5 \rangle$ oraz 3000 losowań w 959 losowaniu jako najlepszy wynik:

$$a = 0,9993, \quad M = 0,141.$$

Czas liczenia wynosił około 15 minut. Przy zmniejszeniu kostki oraz powiększeniu liczby losowań można się spodziewać poprawienia wyniku.

W celu otrzymania dokładnej optymalnej wartości współczynnika a oraz dla określenia dokładności wyników optymalizacji metodą Monte Carlo przeprowadzono tablicowanie wartości funkcji M (moment bezwładności łuku) w zależności od współczynnika a paraboli ax^2 .

```

PROGRAM M(a):
begin comment: Program oblicza momenty bezwładności paraboli
                y=ax^2 względem jej środka ciężkości jako
                funkcje od a, na przedziale [0,p];
integer i,n;
real a,h,p,r,ymax,C1,C2,M;
A:
writecr;
writetext(⟨⟨START dane: n,p,ymax⟩⟩);
n:=typein;
p:=typein;
ymax:=typein;
begin
array R1,R2,y[0:n];
h:=p/n;
for a:=0.0001, a+0.05 while a*p^2<ymax+0.01 do
begin
C2:=M:=0;
r:=sqrt(p^2 + 1/(4*a^2));
C1:=a*p*r + ln(p+r)/(4*a) - ln(1/(2*a))/(4*a);
if C1 > 1.5707813*p then
go to EA;
for i:=0 step 1 until n do
begin
R1[i]:=a * (h*i)^2;
R2[i]:=sqrt(1+(2*a*h*i)^2);
y[i]:=R1[i] * R2[i];
C2:=C2+ y[i];
end;
C2:=(C2-(y[0]+y[n])/2)*h;
r:=C2/C1;
for i:=0 step 1 until n do
begin
y[i]:=(R1[i]-r)^2 * R2[i];
M:=M+y[i];
end;
M:=(M-(y[0]+ y[n])/2)*h;
outcr;
output(⟨nd. dddd⟩,a,outsp(4));
output(⟨nd. dddd⟩,M);
EA:
end a;
end;
goto A
end program M(a);

```

Aby zapewnić wysoką dokładność wyników obliczono ściśle długość łuku paraboli, tzn. całkę występującą w mianowniku wzoru na y_s , mianowicie

$$(3.9) \quad y_s = \frac{a \int_0^p x^2 \sqrt{1+4a^2x^2} dx}{ap \sqrt{p^2 + \frac{1}{4a^2}} + \frac{1}{4a} \ln \left(p + \sqrt{p^2 + \frac{1}{4a^2}} \right) - \frac{1}{4a} \ln \frac{1}{2a}}$$

Pozostałe występujące w zadaniu całki obliczono numerycznie.

Obliczenia wykonano przy użyciu komputera GIER wg programu M (a), s (87).
Otrzymano następujące wyniki:

a	M
0,0001	0,00000
0,0501	0,00022
0,1001	0,00090
0,1501	0,00205
0,2001	0,00370
0,2501	0,00590
0,3001	0,00870
0,3501	0,01214
0,4001	0,01628
0,4501	0,02118
0,5001	0,02690
0,5501	0,03350
0,6001	0,04102
0,6501	0,04954
0,7001	0,05910
0,7501	0,06977
0,8001	0,08159
0,8501	0,09462
0,9001	0,10893
0,9501	0,12455
1,0001	0,14155

Czas liczenia wynosił około 15 minut.

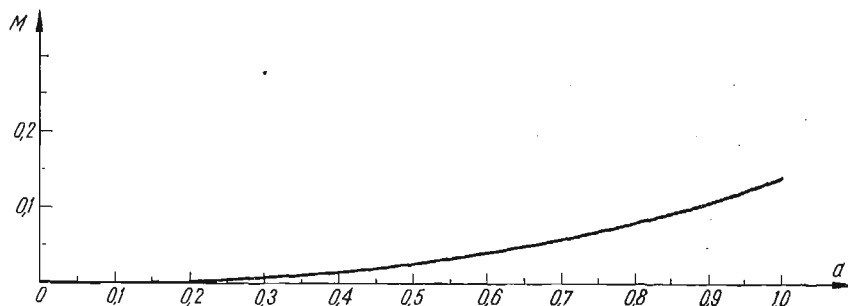
Graficzne przedstawienie pierwszego przypadku tych wyników, tj. przebiegu zależności momentu bezwładności M od współczynnika paraboli a dla $p = 1$, $y_{\max} = 1$ ($n = 100$) podaje rys. 8.

Optymalna wartość współczynnika a we wzorze (3.8) przy $y_{\max} = p = 1$ wynosi więc 1. Za pomocą metody Monte Carlo otrzymano bardzo bliski wynik ($a = 0,9993$). Przy $y_{\max} = p/2 = 1/2$, $a = 0,5$.

Optymalny kształt łuku w postaci paraboli drugiego stopnia określony jest równaniem

(3.10) a) przy $y_{\max} = p = 1$: $y = x^2$,

(3.11) b) przy $y_{\max} = p/2 = 1/2$: $y = \frac{1}{2}x^2$.



Rys. 8. Zależność momentu bezwładności łuku od współczynnika paraboli ax^2

4. Zestawienie wyników optymalizacji i ich omówienie

W tabelcy 1 podano zestawienie wyników optymalizacji kształtu łuku dla wszystkich przypadków przedstawionych poprzednio.

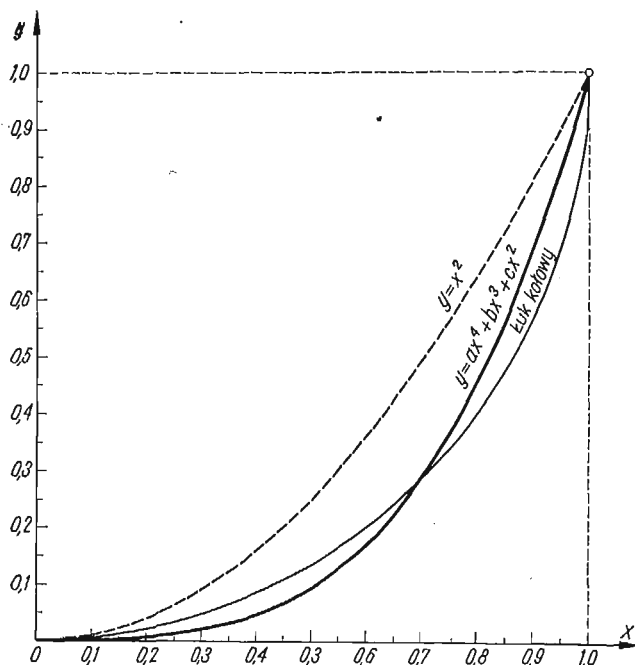
Tablica 1 wartości momentu bezwładności M łuku (połowa szerokości łuku $p = 1$)

				ax^2 	$ax^4 + bx^3 + cx^2$
	1	2	3	4	5
$y_{\max} = p = 1$	0,1178 (79)	$\frac{0,2083}{40}$	0,1488 (100)	0,1415 (95)	0,159 (107)
$y_{\max} = \frac{p}{2} = \frac{1}{2}$	0,0233 (87)	$\frac{0,03125}{(116)}$	—	$\frac{0,0269}{(100)}$	$\frac{0,028}{(104)}$

W pierwszym przypadku, gdy wysokość łuku jest równa połowie jego szerokości, tj. gdy $y_{\max} = p = 1$, porównanych jest pięć rodzajów kształtu połowy łuku, a to: odcinek prostej (1), linia łamana (2), łuk koła (3), parabola drugiego stopnia (4) oraz parabola czwartego stopnia (5). W przypadku linii łamanej (2) nie był spełniony warunek ograniczonej długości połowy łuku $l \leq l_d = \pi p/2$, obowiązujący w pozostałych przypadkach.

W drugim przypadku, gdy $y_{\max} = p/2 = 1/2$, porównano cztery przypadki bez łuku koła. Ponieważ zachowano tu taką samą jak poprzednio wartość $l_d = \pi p/2$, wszystkie rozpatrywane kształty łuku spełniają warunek $l \leq l_d$. Nawet w przypadku łuku w kształcie linii łamanej połowa jego długości wynosi tylko 1,5.

Największy moment bezwładności daje łuk w kształcie linii łamanej (2). Pokazuje to, do jakiego najkorzystniejszego ukształtowania dąży oś łuku. Najmniejszy moment bezwładności daje odcinek prostej (1). Spośród właściwych łuków najmniej korzystna jest parabola drugiego stopnia. Dobry wynik zapewnia parabola czwartego stopnia, która przyjmuje korzystniejszy kształt i daje lepszy wynik niż łuk kołowy o tej samej długości. Na rys. 9 podano porównanie trzech kształtów łuku. Charakterystyczny jest kształt przyjęty przez parabolę czwartego stopnia.



Rys. 9. Porównanie różnych kształtów łuku

W celu wzajemnego porównania podano w tablicy 1 w nawiasach procentowe wartości momentów bezwładności M przyjmując za 100 w pierwszym przypadku łuk kołowy, a w drugim parabolę drugiego stopnia. *Najlepszy wynik momentu bezwładności M wzięto w ramkę a drugi co do wartości wynik podkreślono.*

5. Zakończenie

Zagadnienie optymalizacji kształtu łuku (ze względu na maksimum momentu bezwładności łuku) przy istnieniu warunków ubocznych nie daje się rozwiązać rachunkiem różniczkowym ani rachunkiem wariacyjnym. W niniejszej pracy rozwiązano powyższy problem przy użyciu metody programowania matematycznego.

Przyjmując równanie kształtu łuku w postaci wielomianu potraktowano współczynniki wielomianu jako zmienne decyzyjne, poszukując takich ich wartości, aby moment bezwładności łuku osiągnął maksimum przy równoczesnym spełnieniu warunków ubocznych

(nieprzekroczenie pewnej długości łuku oraz określonej wysokości łuku). Do rozwiązania zagadnienia zastosowano metodę Monte Carlo. Obliczenia numeryczne wykonano na komputerze GIER.

Rozwiązano liczbowo przypadek łuku o kształcie wyrażonym wielomianem czwartego stopnia znajdując optymalne współczynniki wielomianu. Podano programy i wyniki obliczeń komputera. Dla porównania obliczono przypadki łuków o kształcie linii prostej, linii łamanej, łuku koła i paraboli drugiego stopnia.

6. Literatura cytowana w tekście

1. O. LANGE, *Optymalne decyzje — zasady programowania*, PWN, Warszawa 1964.
2. J. GOLIŃSKI i Z. K. LEŚNIAK, *Wybór optymalnych wymiarów konstrukcji za pomocą metody Monte Carlo*, Arch. Inż. Łąd., **10** (1964), 341–356.

Р е з ю м е

ОПТИМАЛИЗАЦИЯ ФОРМЫ АРОЧНОГО СЕЧЕНИЯ ПЕРЕКРЫТИЙ

Задача об оптимализации формы арки (по условию максимума момента инерции) при наличии побочных условий не может быть решена в рамках дифференциального или вариационного исчисления. В настоящей работе выше указанная задача решается при использовании математического программирования.

Уравнение арки берется в виде полинома и затем разыскиваются такие значения коэффициентов, чтобы момент инерции арки достиг максимума, при одновременном удовлетворении побочных условий (не превышение некоторой длины арки и определенной ее высоты). Для решения задачи переменялся метод Монте Карло. Численные расчеты проводились на электронной счетной машине GIER.

Дан числовой пример для случая полинома четвертой степени. Приведены программы и результаты расчета. Для сравнения проводится расчет для случая арок очерченных по прямой, по ламаной линии, по дуге окружности и параболы второй степени.

S u m m a r y

OPTIMUM DESIGN OF THE ARCH SECTION OF SHELL BEAMS

The problem of the optimization of the arch shape (to obtain a maximum of the moment of inertia) with the presence of constraints can be solved neither by using the differential calculus nor by the calculus of variation. In this paper the problem has been solved by using the mathematical programming method.

Assuming the arch shape equation in the form of a polynomial, its coefficients were treated as decisive variables. Such values of the said variables have been searched for to obtain the maximum of the moment of inertia, observing at the same time some side conditions, such as the constraints, i.e. the length of the arch and its height should not be longer than the given values. To solve the problem, the Monte-Carlo-method was applied. The computations were made using the GIER computer.

The case of the arch by the polynomial of the fourth order was numerically solved. The optimum values of its coefficients were found. The programs and the results of the computations are given. For comparison sake, other arch shapes were also computed, i.e. straight line, broken line, the segment of the circle and the parabola of the second order.

INSTYTUT TECHNIKI BUDOWLANEJ

Praca została złożona w Redakcji dnia 10 lipca 1967 r.