

VII. Computer analysis and structural engineering

Objektyp: **Group**

Zeitschrift: **IABSE congress report = Rapport du congrès AIPC = IVBH
Kongressbericht**

Band (Jahr): **11 (1980)**

PDF erstellt am: **13.07.2024**

Nutzungsbedingungen

Die ETH-Bibliothek ist Anbieterin der digitalisierten Zeitschriften. Sie besitzt keine Urheberrechte an den Inhalten der Zeitschriften. Die Rechte liegen in der Regel bei den Herausgebern.

Die auf der Plattform e-periodica veröffentlichten Dokumente stehen für nicht-kommerzielle Zwecke in Lehre und Forschung sowie für die private Nutzung frei zur Verfügung. Einzelne Dateien oder Ausdrucke aus diesem Angebot können zusammen mit diesen Nutzungsbedingungen und den korrekten Herkunftsbezeichnungen weitergegeben werden.

Das Veröffentlichen von Bildern in Print- und Online-Publikationen ist nur mit vorheriger Genehmigung der Rechteinhaber erlaubt. Die systematische Speicherung von Teilen des elektronischen Angebots auf anderen Servern bedarf ebenfalls des schriftlichen Einverständnisses der Rechteinhaber.

Haftungsausschluss

Alle Angaben erfolgen ohne Gewähr für Vollständigkeit oder Richtigkeit. Es wird keine Haftung übernommen für Schäden durch die Verwendung von Informationen aus diesem Online-Angebot oder durch das Fehlen von Informationen. Dies gilt auch für Inhalte Dritter, die über dieses Angebot zugänglich sind.



SEMINAR

VII

Computer Analysis and Structural Engineering

Calcul électronique et constructions de génie civil

Elektronische Berechnung im konstruktiven Ingenieurbau

Co-chairmen:

(morning session)

A.R. Cusens, Great Britain

I. Holand, Norway

(afternoon session)

H. Mang, Austria

Introductory Papers:

“Computers in Structural Design: Some General Thoughts”

D. Alcock, Great Britain

“Evolving Design Practice in the Computer Era”

J. Blaauwendraad, Netherlands

Coordinator:

M. Fanelli, Italy

Leere Seite
Blank page
Page vide

VII**Optimum Design of Metal Structures by Backtrack Programming**

Dimensionnement optimal de constructions métalliques au moyen de la méthode de programmation „backtrack“

Optimalbemessung von Metallkonstruktionen mittels der Backtrack-Programmierungsmethode

JOZSEF FARKAS
Prof., Dr. techn.
Technical University
Miskolc, Hungary

SUMMARY

The backtrack method solves the nonlinear constrained function minimization problems by a systematic search procedure. This combinatorial discrete programming method can be successfully applied to optimization problems if the number of unknowns is not too large. In the optimum design of welded beams a substantial search reduction may be achieved. A simple numerical example shows that this method is also advantageous for educational purposes.

RESUME

La méthode „backtrack“ donne la solution des problèmes de minimisation de fonctions nonlinéaires conditionnelles par procédé de recherche systématique. Cette méthode combinatoire de la programmation discrète est bien applicable aux problèmes d'optimisation, si le nombre des inconnues n'est pas trop grand. Dans le cas de poutres soudées, il est possible de réduire considérablement les pas de la recherche. Un simple exemple numérique illustre les avantages que cette méthode offre pour l'enseignement.

ZUSAMMENFASSUNG

Die Backtrack-Methode löst die nichtlinearen bedingten Funktionsminimierungsprobleme mit Hilfe eines systematischen Suchverfahrens. Diese kombinatorische diskrete Programmierungsmethode ist auf Optimierungsprobleme gut anwendbar, wenn die Anzahl der Unbekannten nicht sehr gross ist. Im Falle von Schweißträgern kann man die Suchschritte wesentlich reduzieren. Ein einfaches Zahlenbeispiel zeigt, dass diese Methode auch für Unterrichtszwecke vorteilhaft ist.



1. THE BACKTRACK PROGRAMMING METHOD

The general exposition of backtrack was given among others by GOLOMB and BAUMERT [4]. This method was applied to welded girder design e.g. by ANNAMALAI, LEWIS and GOLDBERG [1]. SZABÓ [5] has used it for the minimum cost design of hybrid I-beams. The backtrack method solves the constrained function minimization problems by a systematic search procedure. A partial search is carried out for each variable and, if the possibilities are exhausted, then a backtrack and a new partial search is performed. In the optimum design of welded beams a substantial search reduction may be achieved by utilizing the fact that, in most cases, the cost function becomes maximum if the variables take their maximum values. Thus, the optimum solution can be found by decreasing the variables.

We should search for a vector of variables $\underline{x} (x_1, x_2, \dots, x_n)$ for which the cost function will be minimal: $C(\underline{x}) \rightarrow \min$ and which satisfies the design constraints: $g_j(\underline{x}) \geq 0 (j=1 \dots p)$. The series of discrete values of variables are determined by $x_{i\min}$, $x_{i\max}$ and by the constant steps Δx_i between them. The flow diagram of backtrack for three variables is given in Fig.1. The main phases of the calculation are described as follows.

1/ With constant values of $x_{2\max}$ and $x_{3\max}$ the minimum x_{1m} value is searched which still satisfies the design constraints. For the sake of simplicity we use further on the notation $x_{1m} = x_1$. The search may be more efficient by using the interval halving procedure. First, $x_{1\min}$ value is proved. If $x_{1\min}$ dissatisfies the requirements, the interval $x_{1\max} - x_{1\min}$ is halved. For the halving method it should be $x_{i\max} - x_{i\min} = \Delta x_i \cdot 2^q$ where q is an integer.

2/ Similarly to the first phase, the minimum $x_{2m} = x_2$ value is determined by means of halving process, which satisfies the design constraints.

3/ In order to obtain the next x_3 -value, we do not use the interval halving method. In general, x_3 may be easily expressed from the equation $C(x_1, x_2, x_3) = C_0$, so this will be the next x_3 -value. Thus, for the series of x_3 -values it is not necessary to prescribe the condition $x_{3\max} - x_{3\min} = \Delta x_3 \cdot 2^q$. Regarding the next x_3 -value three cases may occur as follows.

3a/ If $x_3 > x_{3\max}$, we take $x_3 = x_{3\max}$ and decrease it step-by-step till x_{3m} which satisfies the constraints or till $x_{3\min}$. Then the first partial search region is exhausted and we must backtrack to x_2 . If $x_2 < x_{2\max}$, we continue the calculation with $x_2 + \Delta x_2$; if $x_2 = x_{2\max}$, we backtrack to x_1 .

3b/ If $x_3 < x_{3\min}$, we backtrack to x_1 .

3c/ If $x_{3\min} < x_3 < x_{3\max}$ and x_3 dissatisfies the constraints, we backtrack to x_2 . If the constraints are satisfied, we continue the calculation according to 3a.

The simplicity of the method enables us to apply it in the education as well. The following numerical example shows that the steps of the calculation can be easily followed by a pocket calculator.

2. NUMERICAL EXAMPLE

The cross section area of a welded I-section subjected to bending and compression should be minimized. The objective function is

$$C = A = ht_w + 2A_f \quad (1)$$

$h = x_1$ web height, $t_w = x_2$ web thickness, $A_f = x_3$ area of a flange. The constraint of maximum stress (g_1) is

$$\sigma_M + \sigma_N \leq R_u \quad (2)$$

where $\sigma_M = M/W_x$; the section modulus $W_x \approx h(A_f + ht_w/6)$ (3)

and $\sigma_N = N/A$ (4)

M and N are the factored bending moment and compressive force, respectively, R_u is the limit stress (ultimate resistance).

The constraint of web buckling (g_2) for steels of tensile strength 370 MPa may be expressed as [3]

$$\frac{h}{t_w} \leq 145 \sqrt{\frac{(1 + \sigma_N/\sigma_M)^2}{1 + 17(\sigma_N/\sigma_M)^2}} \quad (5)$$

Take $M = 320$ kNm, $N = 128$ kN, $R_u = 200$ MPa. The lists of discrete values are as follows (cm, cm²): $h_{\min} = 66$; $h_{\max} = 74$; $\Delta h = 2$; $t_{w\min} = 0,5$; $t_{w\max} = 0,9$; $\Delta t_w = 0,1$; $A_{f\min} = 14$; $A_{f\max} = 22$; $\Delta A_f = 1$. Thus, the total number of combinations is $5 \cdot 5 \cdot 9 = 225$. The steps of the calculations are shown in Table 1. It can be seen that the backtrack method requires only 37 tests to obtain the following optimal values: $h = 70$ cm; $t_w = 0,6$ cm; $A_f = 18$ cm².

Note that a significant search reduction can be achieved by using



first a coarse scale of discrete values (larger Δx_i) and then continuing with smaller Δx_i values in a smaller region near the optimum found in first phase.

3. APPLICATIONS

We have carried out calculations with a Fortran program of backtrack method for the following optimization problems.

1/ Optimum design of compressed columns of constant welded square box cross section by using the author's calculation method published in [2] (2 unknowns).

2/ Minimization of cross section area of welded I-sections subjected to bending and compression (4 unknowns). The application of these suboptimized I-sections to the elastic minimum weight design of frames was demonstrated by a numerical example in [3]. This numerical example was calculated as a problem with 8 unknowns as well.

3/ Minimum cost design of welded homogeneous and hybrid I-beams, simply supported and uniformly loaded (4 unknowns). In the objective function the costs of materials, welding and painting were taken into account. The lowering of flange thickness with a welded splice was also considered [3], [5].

REFERENCES

1. ANNAMALAI, N., LEWIS, A. D. M., GOLDBERG, J. E.: Cost optimization of welded plate girders.- J. Struct. Div. Proc. ASCE 98 (1972) No. ST10, p. 2235-2246
2. FARKAS, J.: Optimum design of compressed columns of constant welded square box cross section... Acta Techn. Hung. 84 (1977) No. 3-4. p. 335-348
3. FARKAS, J., SZABÓ, L.: Optimum design of beams and frames of welded I-section by means of backtrack programming.- Acta Techn. Hung. 91 (1981) (to be published)
4. GOLOMB, S. W., BAUMERT, L. D.: Backtrack programming.- J. Assoc. Comput. Machinery 12 (1965) p. 516-524
5. SZABÓ, L.: Optimum design of hybrid I-beams. Ph.D. Dissertation (in Hungarian). Techn. Univ. Miskolc, 1978

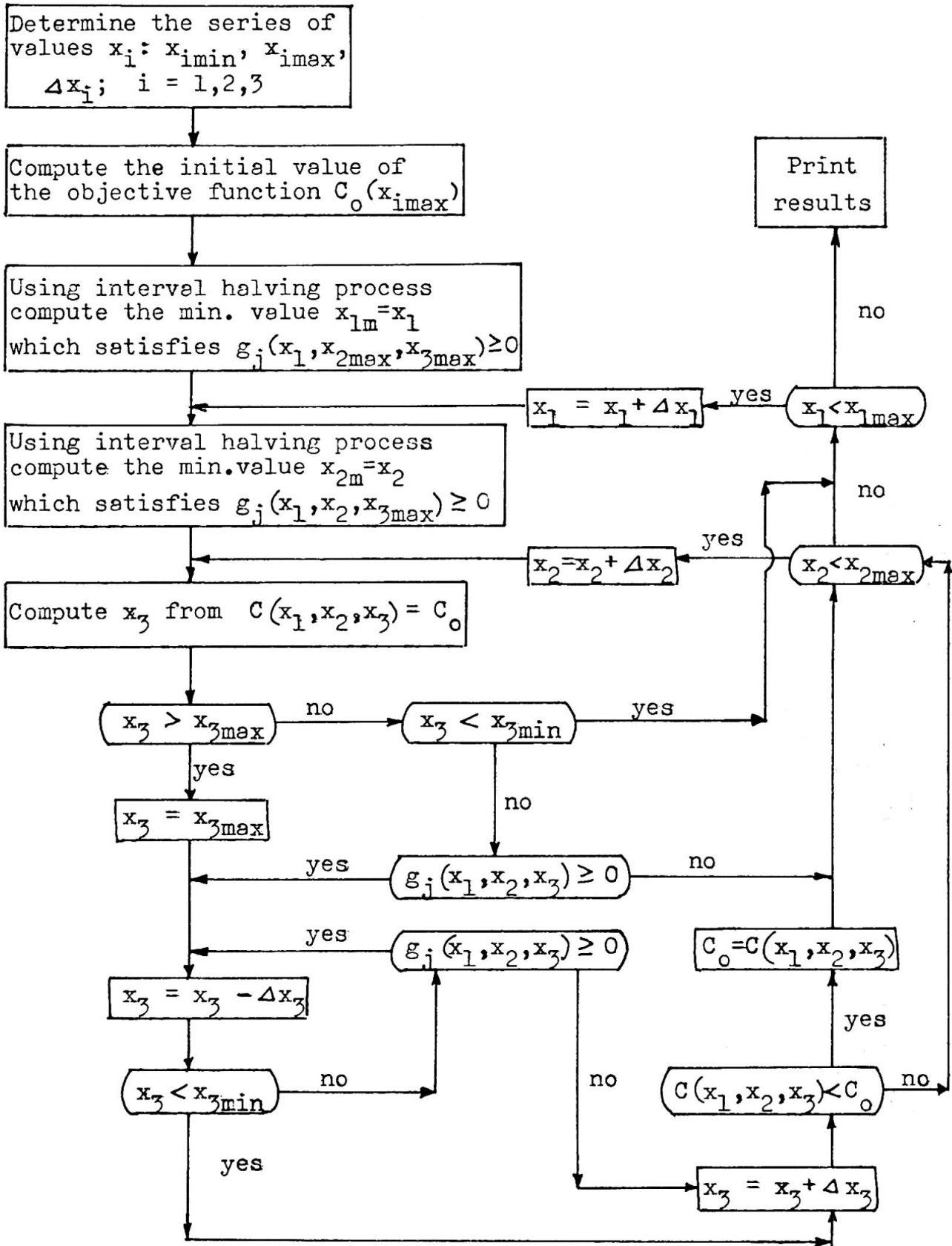


Fig.1 Flow chart for the backtrack programming

Table 1. Numerical example illustrating the steps of backtrack method in the case of three unknowns. Notation: + satisfies, - dissatisfies

x_1 (cm)	x_2 (cm)	x_3 (cm ²)	C (cm ²)	ξ_1 Eq. (2)	ξ_2 Eq. (5)	Comments
74	0,9	22	110,6	+	+	$C_0 = 110,6 \text{ cm}^2$
66	0,9	22	103,4	+	+	$x_{1\min}$ satisfies, it is not necessary to use the halving process for x_1 , only for x_2
66	0,5	22	77,0	+	-	
66	0,7	22	90,2	+	+	
66	0,6	22	83,6	+	+	
66	0,6	22	83,6	+	+	$x_3 = (110,6 - 66 \cdot 0,6) / 2 = 35,5 > x_{3\max}$; $C_0 = 79,6$
66	0,6	21	81,6	+	+	
66	0,6	20	79,6	+	+	
66	0,6	19	77,6	-	+	
66	0,7	16	78,2	-	+	$x_3 = (79,6 - 66 \cdot 0,7) / 2 = 16,7$
66	0,8	13				$x_3 < x_{3\min}$, backtrack with x_1
68	0,9	22	105,2	+	+	halving process for x_2
68	0,5	22	78,0	+	-	
68	0,7	22	91,6	+	+	
68	0,6	22	84,8	+	+	
68	0,6	19	78,8	+	+	$x_3 = (79,6 - 68 \cdot 0,6) / 2 = 19,4$
68	0,6	18	76,8	-	+	
68	0,7	15	77,6	-	+	$C_0 = 78,8$; backtrack with x_2
68	0,8	12				$x_3 = (78,8 - 68 \cdot 0,7) / 2 = 15,6$
68	0,8	12				$x_3 < x_{3\min}$, backtrack with x_1
70	0,9	22	107,0	+	+	halving process for x_2
70	0,5	22	79,0	+	-	
70	0,7	22	93,0	+	+	
70	0,6	22	86,0	+	+	
70	0,6	18	78,0	+	+	$x_3 = (78,8 - 70 \cdot 0,6) / 2 = 18,4$
70	0,6	17	76,0	-	+	
70	0,7	14	77,0	-	+	$C_0 = 78,0$; backtrack with x_2
70	0,8	11				$x_3 = (78,0 - 70 \cdot 0,7) / 2 = 14,5$
70	0,8	11				$x_3 < x_{3\min}$, backtrack with x_1
72	0,9	22	108,8	+	+	halving process for x_2
72	0,5	22	80,0	+	-	
72	0,7	22	94,4	+	+	
72	0,6	22	87,2	+	-	
72	0,7	13				$x_3 = (78,0 - 72 \cdot 0,7) / 2 = 13,8 < x_{3\min}$, backtrack with x_1
74	0,9	22	110,6	+	+	halving process for x_2
74	0,5	22	81,0	+	-	
74	0,7	22	95,8	+	+	
74	0,6	22	88,4	+	-	
74	0,7	13				$x_3 = (78,0 - 74 \cdot 0,7) / 2 = 13,1 < x_{3\min}$
						$x_1 = x_{1\max}$; results: $C_{\min} = 78,0$
						$x_1 = 70$; $x_2 = 0,6$; $x_3 = 18$

VII**Computing Intrinsic Values of Flexural Vibrations of Cable-Stayed Bridges**

Calcul des valeurs propres des oscillations verticales des ponts à haubans

Berechnung der Eigenwerte vertikaler Schwingungen von Schrägseilbrücken

MEHMED S. CAUSEVIC

Assist. Professor of Civil Eng.

University of Banja Luka

Banja Luka, Yugoslavia

SUMMARY

Discrete mass systems (or mathematical models) are described rather than actual bridge cables. The mathematical modelling of a cable represents the essential features of the cable's behaviour and is at the same time sufficiently simple to enable analysis of a complex assembly of such cables in a dynamic analysis of cable-stayed bridges.

RESUME

Les modèles mathématiques représentant le comportement dynamique de câbles réels de ponts sont décrits. La valeur de ces modèles mathématiques est importante car les masses discrètes des éléments du pont et des câbles peuvent être utilisées lors de l'analyse des caractéristiques dynamiques.

ZUSAMMENFASSUNG

Es werden die mathematischen Modelle dargestellt, die das dynamische Verhalten realer Kabel einer Brücke beschreiben. Diese Modelle erlauben, für eine dynamische Analyse das Zusammenwirken der Kabel mit der restlichen Brückenkonstruktion auf einfache Weise zu erfassen.

ONE NEW PROPOSAL FOR COMPUTING EIGENVALUES OF FLEXURAL VIBRATIONS OF
CABLE-STAYED BRIDGES
DISCRETE MASS SYSTEMS WHICH SUBSTITUTE THE CABLES OF THE BRIDGE

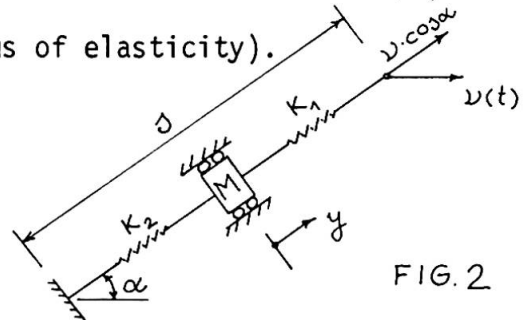
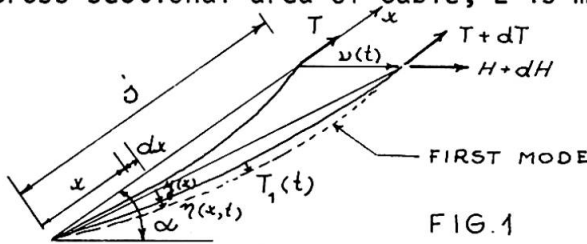
Taking into consideration that the second and all the other even oscillating cable forms do not call out the change of force in cable, the first odd cable form has predominant influence upon dynamic behaviour of the bridge. This is so because the first odd cable form, on the occasion of oscillation, causes the greatest change of static force in the cable. To this purpose such mathematical models will be made which completely substitute the behaviour and the characteristics of the cables oscillating by their first form. On this occasion in the springs of mathematical models will be formed the same force which was produced in the real cable on the occasion of its forced oscillations during the oscillation of the bridge, while masses of mathematical models oscillate by the same frequencies by which the real cables oscillate. The contribution and significance of these mathematical models reflects in the fact that the whole assembly consisting of discrete masses of the bridge structure and the discrete masses of mathematical models (instead of cables) is analysed together. Now it is possible to include the increased influence of the cable, which has the natural frequency close to the frequency of the oscillating assembly, upon the dynamic behaviour of the bridge. Differential equation of motion for cables with one and fixed and the other one connected to the oscillating structure (fig.1) has the following form [1,2] :

$$\frac{w}{g} [\ddot{\eta}(x,t) + \frac{x}{s} \ddot{y}(t) \sin \alpha] = T \cdot \eta''(x,t) + dT(t) \cdot y''(x) \tag{1}$$

where w is the weight per unit length of cable, T is static force in cable before the beginning of oscillations, g is acceleration of graviti, dT is the value for which the force in cable (due to its oscillations) has changed and has the following form [1,2] :

$$dT(t) = \frac{AE}{s} \cdot \cos \alpha \left[y(t) + \frac{w}{T} \int_0^s \eta(x,t) \cdot dx \right] \tag{2}$$

(A is cross sectional area of cable, E is modulus of elasticity).



The form of forced vibrations of the problem can be expressed by natural forms of free vibrations of the cable X_i and can be presented by the following equation [3] :

$$\eta(x,t) = \sum_{i=1}^{\infty} X_i(x) \cdot T_i(t) \quad ; \quad X_i = \sin \frac{i \alpha x}{s} \quad i=1,2,3... \tag{3}$$

where T_i is a generalized coordinate which represents the displacements in the middle of the cables span during its forced vibrations. The function T_i will be found by substitution of the eq. (3) into the eq. (1). In this case the value of the function T_1 will be required and be the resultant solution of the following differential equation:

$$\ddot{T}_1 + \omega_0^2 T_1 = - \frac{4gk}{\alpha T} \cdot y(t) - \frac{2 \sin \alpha}{\alpha} \cdot \ddot{y}(t) \tag{4}$$

$$k = \frac{AE}{s} \cdot \cos^2 \alpha \quad ; \quad \omega_0^2 = \omega_1^2 + \frac{8sgkw}{T^2 \alpha^2} \tag{5}$$

where k is a horizontal stiffness of the cable conceived as a weightless wire.

If the eq. (3) is substituted into the eq. (2) we shall obtain the value for which the static force in the cable has changed:

$$\Delta T = \frac{AE}{\delta} \cdot \nu(t) \cdot \cos \alpha - \frac{2wAE \cos \alpha}{\alpha T} \cdot T_1(t) \tag{6}$$

During the oscillation $\nu(t)$ of one end of the mathematical model its discrete mass is oscillating by the same frequency by which the real cable oscillates, and in the spring K_1 the same force (eq.6) is produced as in the real cable (while the real cable is oscillating by its first form of oscillation). For the mathematical model from fig.2 the following equations are related

$$\ddot{y} + \omega_0^2 y = \frac{K_1}{M} \nu(t) \cdot \cos \alpha \quad ; \quad \omega_0^2 = \frac{K_1 + K_2}{M} \quad ; \quad \Delta T = \nu(t) \cdot K_1 \cdot \cos \alpha - K_1 y(t) \tag{8}$$

where M is mass and K_1 and K_2 are spring stiffnesses of the mathematical model. In order to make the eq. (7) and (9) identical to the eq. (4) and (6) the following transformations of coordinates will be introduced [3]:

$$T_1(t) = E_1 y(t) - \frac{2 \sin \alpha}{\alpha} \nu(t) \tag{10}$$

where E_1 is the unknown constant of transformation. That is the method how the elements of mathematical model with single degree of freedom and constant of transformation are obtained [3,4]:

$$K_1 = \frac{AE}{\delta} - \frac{4wAE \sin \alpha}{\alpha^2 T} \tag{11} \quad M = \frac{(K_1 \alpha T \cos \alpha)^2}{8wsgk^2} \tag{13}$$

$$E_1 = -K_1 \frac{\alpha T \cos \alpha}{2wsgk} \tag{12} \quad K_2 = M \omega_0^2 - K_1 \tag{14}$$

The values ν_r and y_r are produced for each r -th form of the bridge oscillation. From the eq. (10) and (12) is obtained

$$T_{1,r} = -\frac{K_1 \alpha T \cos \alpha}{2wsgk} y_r - \frac{2 \sin \alpha}{\alpha} \nu_r \tag{15}$$

and the real values of the cables displacement can be found from eq. (3). Let now introduce the discrete mass system (mathematical model) substituting the cable with both ends in oscillation (fig.3). The lower end of the cable connected to the oscillating structure has vertical direction of oscillation ν_1 while the upper end of the cable has horizontal direction of oscillation ν_2 . According to the eq. (11) - (14) the mechanism may be established, which represents two mathematical models from fig.2, whose elements can be presented by the following expressions:

$$K_{12} = \frac{AE}{\delta} \cos \alpha - \frac{2wAE \cos \alpha}{\alpha T} \cdot \frac{2 \sin \alpha}{\alpha} \quad ; \quad E_{12} = -\frac{K_{12} \alpha T \cos \alpha}{2wsgk} \quad ; \quad M_2 = \frac{(K_{12} \alpha T)^2 \cos \alpha}{8wsgk^2} \tag{16}$$

$$K_{11} = \frac{AE}{\delta} \sin \alpha - \frac{2wAE \cos \alpha}{\alpha T} \cdot \frac{2 \cos \alpha}{\alpha} \quad ; \quad E_{11} = -\frac{K_{11} \alpha T \cos \alpha}{2wsgk} \quad ; \quad M_1 = \frac{(K_{11} \alpha T \cos \alpha)^2}{8wsgk^2 \sin \alpha}$$

$$K_{22} = M_2 \cdot \omega_0^2 - K_{12} \quad ; \quad K_{21} = M_1 \cdot \omega_0^2 - K_{11} \quad ;$$

Let us pass to the possibility when both ends are oscillating at the same time. In this case the principle of superposition can be applied to the previously mentioned mechanism and the next mechanism (fig.3) as a system with two degrees of freedom can be established. Υ is an angle under which the resultant displacement of mass M given by coordinate y is performed. This angle is obtained by one of the following equations:

$$\Upsilon = \arctan \left(\frac{K_{22}}{K_{21}} \right) \quad ; \quad \Upsilon = \arctan \left(\frac{K_{12}}{K_{11}} \right) \tag{17}$$

The link is made between the displacements y_1 and y_2 of the previously mentioned mechanism and the displacement y of the discrete mass of the required mathematical model

$$y(t) = y_2(t) \cdot \sin \Upsilon + y_1(t) \cdot \cos \Upsilon \tag{18}$$

Accordingly, the rest of the unknown elements of mathematical model, which completely substitutes the oscillations of the cable with both ends connected to the oscillating structure, are:

$$K_2 = \sqrt{(K_{22})^2 + (K_{21})^2} \quad ; \quad M = \sqrt{(M_2)^2 + (M_1)^2}$$

$$E_1 = \sqrt{(E_{12})^2 + (E_{11})^2} \quad ; \quad \Upsilon_r = \arctan \left(\frac{K_{12} \cdot \nu_{2,r}}{K_{11} \cdot \nu_{1,r}} \right)$$

$$T_{1,r}(t) = E_{1,r} y_r(t) - \frac{2 \sin \alpha}{\alpha} \nu_{2,r}(t) - \frac{2 \cos \alpha}{\alpha} \nu_{1,r}(t)$$

eq. (19)

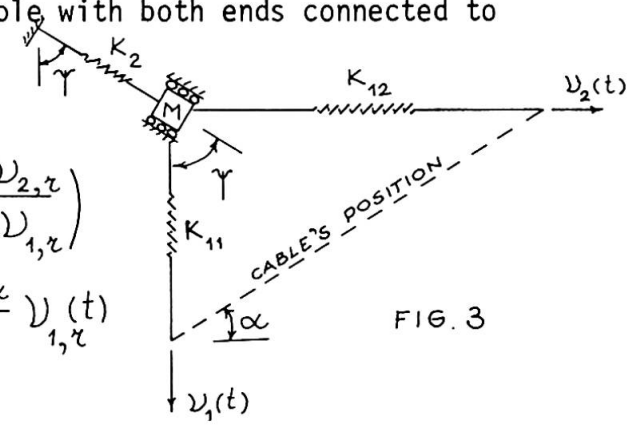


FIG. 3

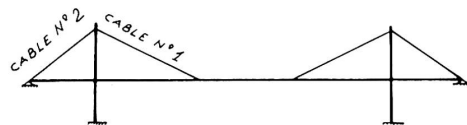


FIG. 4a

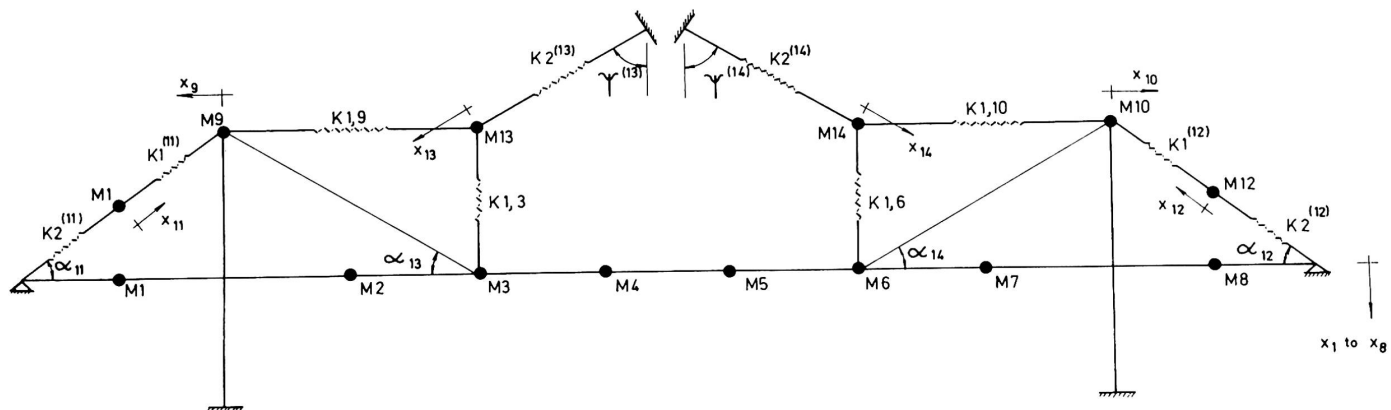


FIG. 4b

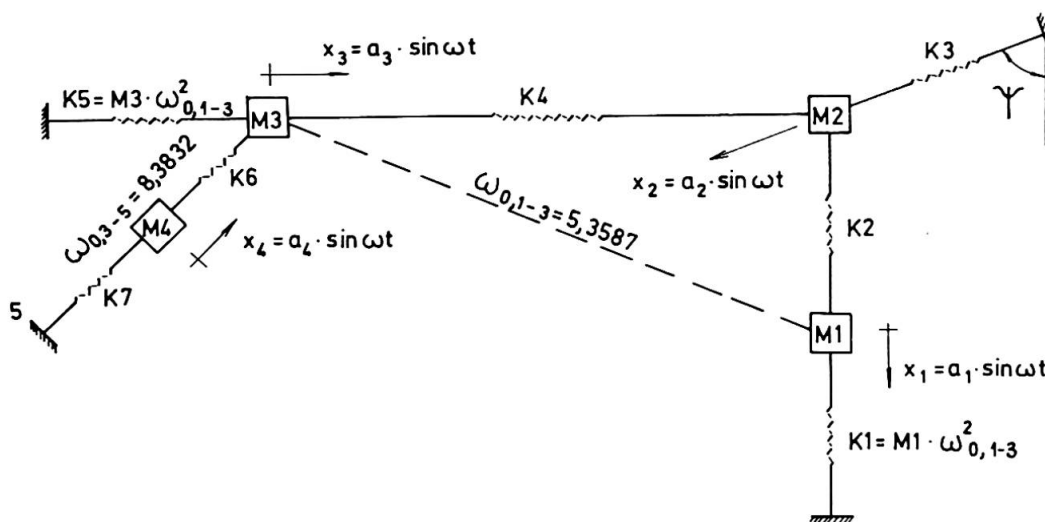


FIG. 5

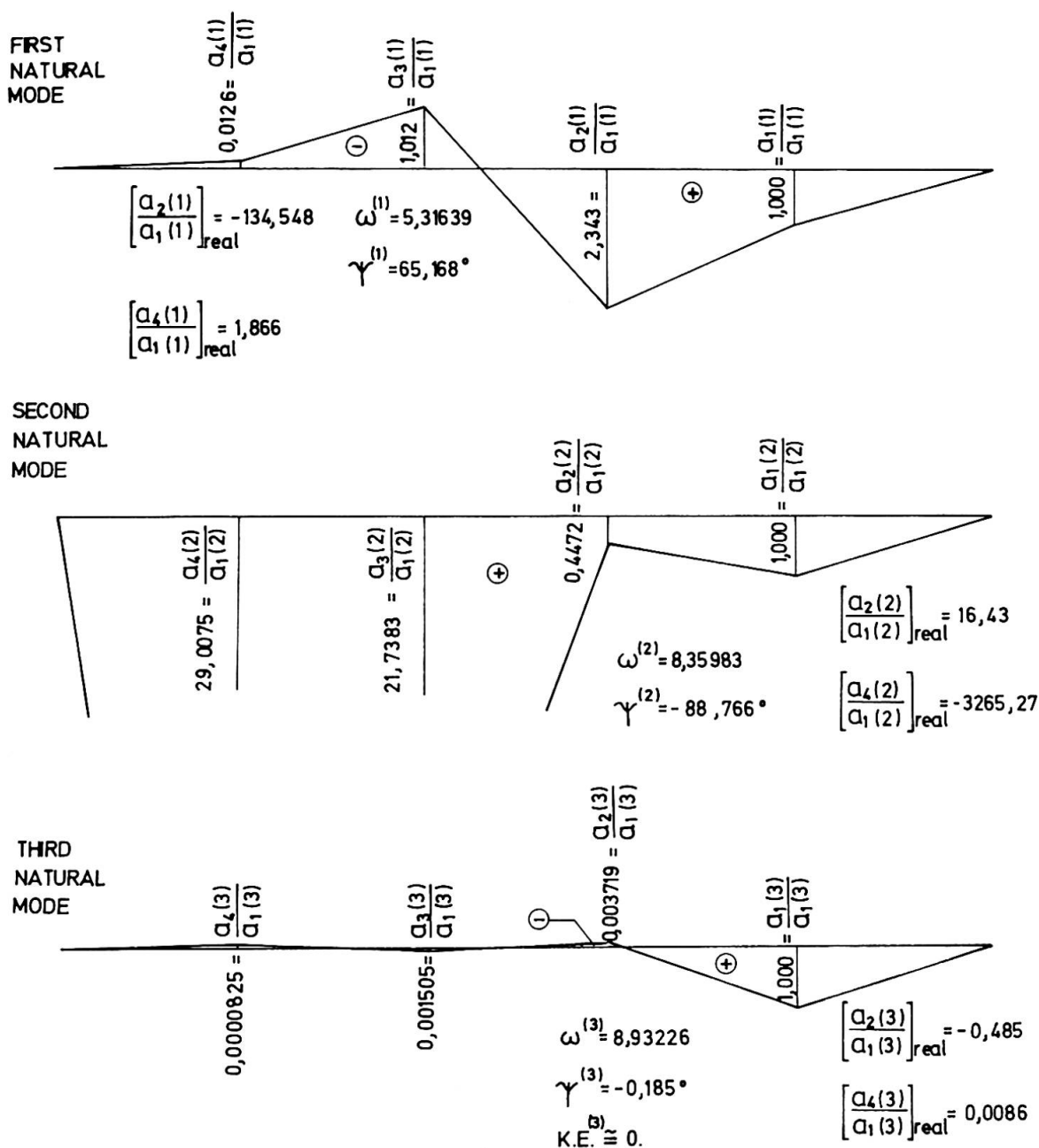


FIG. 6



THE NUMERICAL EXAMPLE AND CONCLUSIONS

The natural modes and frequencies of bridges are the result of the following matrix differential equation $[M]\{\ddot{x}\} + [k]\{x\} = \{0\}$ (20)

By introduction the mathematical models instead of cables, the matrix of masses and the matrix of stiffnesses in eq. (20) contain the elements of mathematical models as well. The results of analysis reported herein are concerned with the part of the structure in fig. 4 comprising the cable with both ends in oscillation (cable no.1), the cable with only one end in oscillation while the other one is fixed (cable no.2) and the parts of the bridge structure to which the referring cables are connected (fig.5).

The stiffnesses of the springs $K1$ and $K5$ of the referring discrete system will be chosen to make possible the analysis of the most interesting case when the discrete system in fig. 5 has the first natural frequency near to the natural frequency $\omega_{0,1-3}$ of the cable no.1 and the second natural frequency near to the natural frequency $\omega_{0,3-5}$ of the cable no.2.

$$K1 = M1 \cdot \omega_{0,1-3}^2 ; \quad K5 = M3 \cdot \omega_{0,1-3}^2 \quad (21)$$

Discrete masses $M1$ and $M3$ are the masses at the connection points of the cables and the bridge structure. The numerical values of these discrete masses as well as natural frequencies of the cables are referred to the new designed cable-stayed bridge over the river Sava in Belgrade, which is now in stage of construction [5]. By the analysis of such a system (fig.6) the following is concluded:

- 1.) The results obtained in case when the analysed structure oscillates with frequency near to natural frequency of some of the cable in the system demonstrate that this cable produces kinetic energy of a great value. This energy can now be taken into account when natural modes and frequencies of the bridge structure are required.
- 2.) In the discrete system analysed here is easy to notice the great values of the real natural amplitudes of oscillations of the cables (1-3) and (3-5) when the analysed discrete system in its first and second form oscillates with frequencies which are near to the natural frequencies of the cables (1-3) and (3-5). When the analysed discrete system oscillates in its third and fourth natural form with frequencies different from the natural frequency of any cable in the system, the real natural amplitudes of oscillation of cables became very small and may be concerned as negligible.
- 3.) The designers of such kind of bridges can find out eigenvalues of flexural vibrations using here presented method. Forced or natural frequencies of the bridge should not approach the vicinity of natural frequencies of any of the cables on the bridge. The natural frequency of the cable can be found using reference [2].

REFERENCES

- [1] Davenport A. "Dynamic behaviour of massive guy cables", ASCE, Journal of Steels G. Structural Division, April 1965.
- [2] Čaušević M. "The Oscillations of the Cables as the Elements of Cable-Stayed Bridge or TV Antena Mast" (in Yugoslav), TEHNIKA, March 1978.
- [3] Hartman A. "Comparison of the predicted and measured dynamic response of structure to wind", Engineering Science Research Report ST-4-66, The University of West. Ontario, Canada 1966.
- [4] Čaušević M. "Natural Modes and Frequencies of Flexural Vibrations of Cable-Stayed Bridges Using the Concept of Mathematical Modelling" (in Yugoslav), TEHNIKA 1978/4
- [5] Hajdin N. "Eisenbahnschrägseilbrücke über die Sava in Belgrad" Jevtović Lj. Der Stahlbau, April 1978.

VII

Kriech- und Schwindberechnung mit PRAKSI 4

Creep and Shrinkage Analysis with PRAKSI 4

Calcul des effets du fluage et du retrait avec PRAKSI 4

W. HAAS

Dr. -Ing.

Rechen- und Entwicklungsinstitut für EDV im Bauwesen
Stuttgart, Bundesrepublik Deutschland

ZUSAMMENFASSUNG

Die Anforderungen an ein Programm zur Berechnung der Auswirkungen des Kriechens und Schwindens werden an einem Beispiel aus dem Brückenbau erörtert. Insbesondere die Forderung, die abschnittsweise Herstellung und Belastung des Bauwerks im Programm berücksichtigen zu können, hat weitreichende Auswirkungen auf das Leistungsspektrum eines derartigen Programms. Berechnungsergebnisse werden erläutert und mit gemessenen Werten verglichen.

SUMMARY

A typical case study is used to demonstrate the basic requirements which must be met by structural programs for creep and shrinkage analysis. Specifically, the condition that the sequence of construction phases and the corresponding load history are to be modelled properly plays a dominant role in the design of the program architecture. In addition, the results of the numerical analysis are compared with those obtained by measurements.

RESUME

Les exigences posées à un programme pour le calcul des effets du fluage et du retrait du béton sont illustrées dans le cas d'un pont. L'influence de l'exécution et du chargement de la construction en plusieurs étapes est déterminante pour l'organisation et la capacité du programme. Des résultats de calcul sont présentés et comparés avec des valeurs mesurées.



1. EINLEITUNG

Bei der Konzipierung eines Programmes für die Ermittlung der Auswirkungen des zeitabhängigen Verformungsverhaltens (Kriechen und Schwinden) von Beton auf die Schnittkräfte und Verschiebungsgrößen konzentriert sich das Interesse häufig auf die mechanischen und numerischen Probleme, also auf die Formulierung geeigneter Stoffgesetze für das Spannungs-Dehnungs-Verhalten des Betons und auf numerisch stabile Algorithmen für die Integration der Änderungen der Schnittkräfte und Verschiebungsgrößen über die Zeit (Tab.1).

Über diesen sicherlich wichtigen Problemen werden leicht die organisatorischen Aspekte der Aufgabenstellung vernachlässigt, die sich z. B. daraus ergeben, daß Betontragwerke stets in einer Folge von Bauzuständen hergestellt werden, die unterschiedliche statische Systeme aufweisen können, daß die Querschnitte nachträglich z. B. durch Beton oder Spannstahl ergänzt werden können, und daß das Eigengewicht und die Vorspannung entsprechend den Bauzuständen abschnittsweise aufgebracht wird (Tab.1). Diese Belastungsgeschichte kann sich über mehrere Jahre erstrecken.

math.-mech. Aspekte	organisatorische Aspekte
<ul style="list-style-type: none"> . geeignete Stoffgesetze . numerisch stabile Algorithmen für die Zeitintegration der Schnittkräfte und Verschiebungsgrößen 	<ul style="list-style-type: none"> . Bauzustände . nachträgliche Querschnittsergänzungen . Belastungsgeschichte

Tab.1 Problemfelder der Kriech- und Schwindberechnung

Mit dem vorliegenden Beitrag sollen anhand einer durchgeführten elektronischen Berechnung aus dem Brückenbau die sich aus dem Bauablauf ergebenden organisatorischen Aspekte einer derartigen Berechnung exemplarisch erörtert, sowie einige Berechnungsergebnisse geschildert und mit Messwerten verglichen werden.

2. DIE KRIECH- UND SCHWINDBERECHNUNG DER KOCHERTALBRÜCKE

2.1 Konstruktion und Belastungsgeschichte

Die Kochertalbrücke ist eine 1128 m lange Autobahnbrücke aus Spannbeton. Ihre größte Höhe über dem Talgrund beträgt 185 m.

In Längsrichtung ist das statische System der Brücke ein Rahmentragwerk, wobei die vier Mittelpfeiler in den Überbau eingespannt sind. Die Regelfeldweite des Überbaus beträgt 138 m (s. Abb.1).

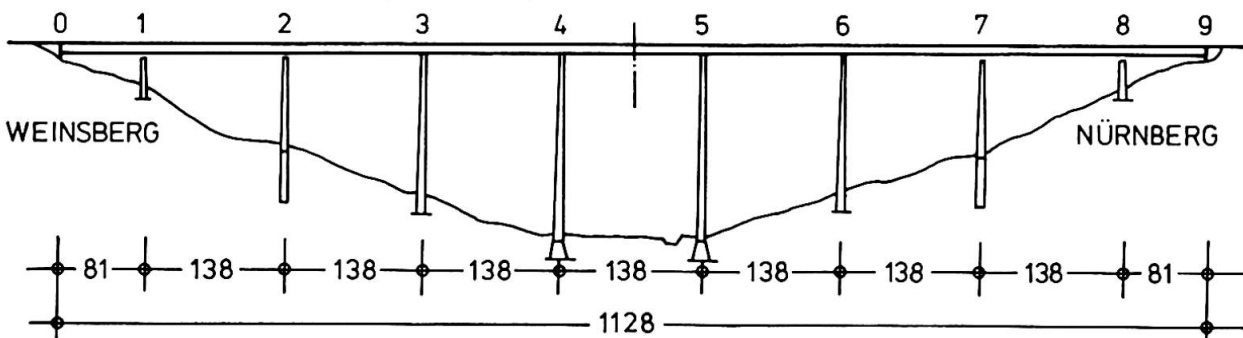


Abb. 1 Statisches System in Längsrichtung

Der Brückenquerschnitt ist ein 6.50 m hoher einzelliger Hohlkasten mit weit auskragenden Platten (Abb.2).

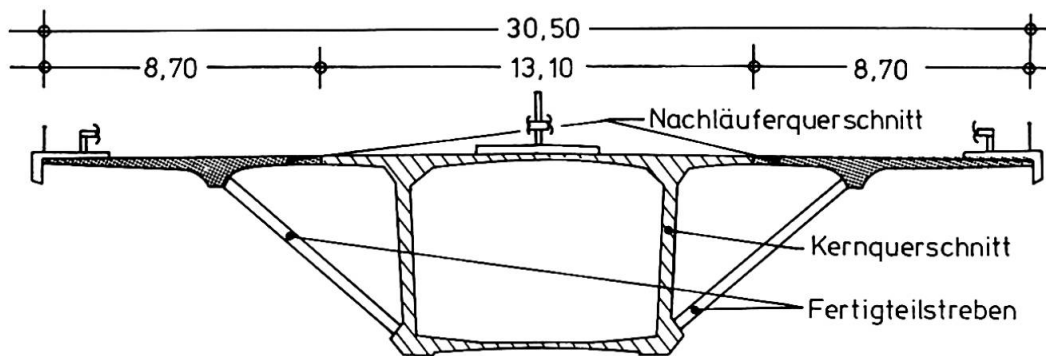


Abb. 2 Brückenquerschnitt

Der Oberbau wird abschnittsweise im Freivorbau unter Verwendung eines Vorfahrträgers hergestellt. Dabei wird zunächst nur der einzellige Hohlkastenquerschnitt, der sogenannte Kernquerschnitt (s. Abb. 2), im Freivorbau vom Pfeilerkopf aus in beiden Richtungen hergestellt. Zu einem späteren Zeitpunkt wird der Kernquerschnitt durch das beidseitige Anbetonieren der Plattenstreifen, des sogenannten Nachläufers, zum Gesamtquerschnitt ergänzt. Diese Plattenstreifen sind durch schräge Fertigteilstreben auf dem Kernquerschnitt abgestützt. Wegen der kurzen zur Verfügung stehenden Ausführungszeit wurde der Oberbau von beiden Widerlagern aus vorangetrieben. Dabei ergeben sich 14 Bauzustände mit unterschiedlichen statischen Systemen.

2.2 Ziele der Berechnung

Durch die Herstellung des Bauwerks in 14 Bauzuständen mit unterschiedlichen statischen Systemen ergeben sich Umlagerungen im Gesamtsystem. Um diese Umlagerungen möglichst gering zu halten, wurden in den Koppelfugen der Freivorbauabschnitte, bevor sie geschlossen wurden, mit Pressen sogenannte Fugenexpansionsschnittkräfte aufgebracht. Mit diesen Fugenexpansionsschnittkräften werden dort diejenigen Schnittkräfte eingepreßt, die vorhanden wären, wenn alle Lasten von Anfang an auf das nach dem Schließen der Fugen vorhandene statische System wirken würden. Da das Brückenbauwerk jedoch Beton unterschiedlichen Alters enthält, läßt sich das Systemkriechen durch diese Maßnahme nicht vollständig unterdrücken.

Die nachträgliche Querschnittsergänzung durch den Nachläuferquerschnitt bewirkt Umlagerungen im Querschnitt ("Querschnittskriechen") und Verformungen, die ihrerseits Umlagerungen im Gesamtsystem ("Systemkriechen") hervorrufen. Beide Umlagerungen können hier nicht unabhängig voneinander betrachtet werden [1].

Neben dem zeitlichen Verlauf der Gesamt- und Teilschnittkräfte sollte auch noch der zeitliche Verlauf der Durchbiegung berechnet werden, um die erforderliche Überhöhung der Brücke festlegen zu können.

2.3 Annahmen für die elektronische Berechnung

2.3.1 Statisches System

Da das statische System und der Bauablauf symmetrisch sind, genügt es, das halbe Brückenbauwerk zu untersuchen. Es wird als ein ebenes Stabwerk mit 13 Stäben idealisiert (Abb. 3).

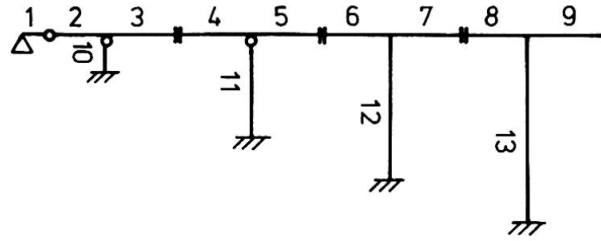


Abb. 3 Statisches System für die elektronische Berechnung

In den einzelnen Bauzuständen ergaben sich unterschiedliche Lager- und Kopplungsbedingungen für die Stäbe, entsprechend dem jeweiligen Baufortschritt. Jeder Stab kann einen beliebigen Verlauf der Querschnittswerte längs der Stabachse aufweisen und jeder Querschnitt kann ein Verbundquerschnitt mit beliebig vielen Teilquerschnitten sein. Dieser außerordentlich leistungsfähige Stab wird benötigt, um z.B. das durch das allmähliche Anbetonieren des Nachläuferquerschnitts bedingte Wandern des Querschnittsprunges des Gesamtquerschnitts sowie das sukzessive Anspannen und Verpressen der Spannglieder berücksichtigen zu können.

2.3.2 Belastungsgeschichte

Die Belastungsgeschichte bestand aus folgenden Lastfällen

- 32 Lastfälle Eigengewicht des Kerns
- 32 Lastfälle Eigengewicht des Nachläufers
- 39 Lastfälle für verschiedene Stellungen des Vorbauwagens und des Vorfahrträgers
- 9 Lastfälle Fugenexpansion
- 9 Lastfälle Belag und Gesimse
- 41 Lastfälle Vorspannung mit 175 Spanngliedlagen im Kern und 90 Spanngliedlagen im Nachläufer.

Diese Lastfälle wurden zu insgesamt 42 verschiedenen Zeitpunkten aufgebracht.

2.3.3 Materialannahmen

Die Kriech- und Schwindverläufe wurden Spannbetonrichtlinien [2] entnommen. Aus diesen Werten wurden die Materialkennwerte für das alternde Drei-Elemente-Modell ermittelt, mit dem die Berechnung durchgeführt wurde. In [3] ist genauer beschrieben, wie man mit dem alternden Drei-Elemente-Modell das in den Spannbetonrichtlinien [2] festgelegte Materialverhalten darstellen kann.

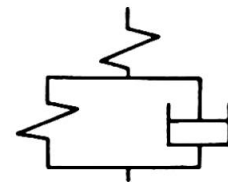


Abb.4 alterndes Drei-Elemente-Modell

2.3.4 Zeiteinteilung für die Integration über die Zeit

Die Integration der Änderungen der Schnittkräfte und Verschiebungsgrößen wurde in 90 Zeitintervallen durchgeführt. Dabei wurden jedoch 83 Zeitintervalle für eine genügend genaue Berücksichtigung des Bauablaufs und der Belastungsgeschichte benötigt und lediglich sieben Zeitintervalle um die Änderungen der Schnittkräfte und Verschiebungsgrößen über die Zeitspanne von der Verkehrsübergabe ($t=1120$ Tage) bis zum angenommenen Endzeitpunkt ($t=12000$ Tage) zu integrieren.

2.4 Ergebnisse der Berechnung

2.4.1 Gesamtschnittkräfte

In Abb. 5 ist die Umlagerung des Gesamtbiegemoments in der Mitte des dritten Brückenfeldes dargestellt. Die Wendepunkte zu Beginn der Kurve sind durch die unterschiedlichen Bauzustände bedingt. Die zum Zeitpunkt $t \rightarrow \infty$ vorhandene Gesamtumlagerung $M=12.1\text{MNm}$ beträgt ca. 5% des Biegemoments infolge des Gesamteigengewichts des Überbaus am Endsystem. Durch die Fugenexpansion gelang es also, das Systemkriechen stark zu reduzieren.

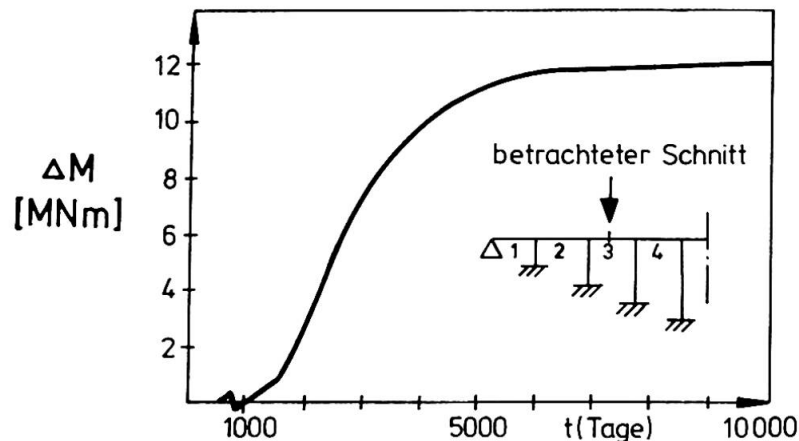


Abb. 5 Umlagerung des Gesamtbiegemoments

2.4.2 Teilschnittkräfte

In Abb. 6 sind die Änderungen der Teilnormalkräfte im Kern- und Nachläuferquerschnitt, ebenfalls in der Mitte des dritten Brückenfeldes, dargestellt.

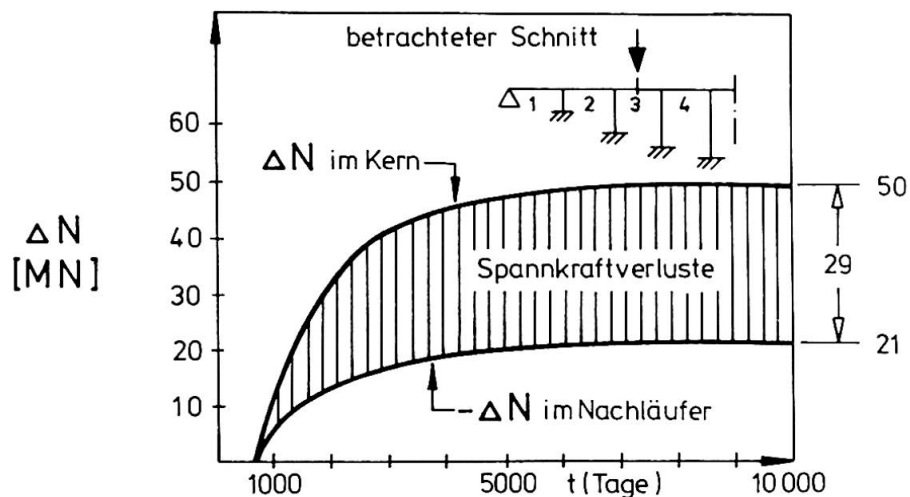


Abb. 6 Umlagerung der Teilnormalkräfte

Im Kernquerschnitt ergibt sich infolge Kriechen und Schwinden ein Abbau der Druckkraft um 50 MN. Davon sind 29 MN Spannkraftverluste und 21 MN wandern in den Nachläuferquerschnitt. Für $t \rightarrow \infty$ ist im Nachläuferquerschnitt eine Normalkraft von -31 MN vorhanden. Davon werden -21 MN (das sind 68%) durch Kriechen und Schwinden aufgebaut. Es zeigt sich also, daß der Nachläuferquerschnitt in erheblichem Umfang nachträglich zur Lastabtragung herangezogen wird.



2.4.3 Überhöhungen

In Abb.7 sind für drei verschiedene Zeitpunkte die berechneten und gemessenen Überhöhungen vergleichend dargestellt. Die Darstellung umfaßt den zu Pfeiler 7 (s.Abb.1) gehörigen Freivorbaubereich. Um den Einfluß der Ungenauigkeiten bei der Ermittlung der Baugrundsetzungen und der Temperaturdehnungen des Pfeilers in Abb.7 zu eliminieren, wurden die gemessenen Überhöhungskurven (um maximal 4 cm) so parallel verschoben, daß sie über dem Pfeilerkopf mit den berechneten Kurven übereinstimmen.

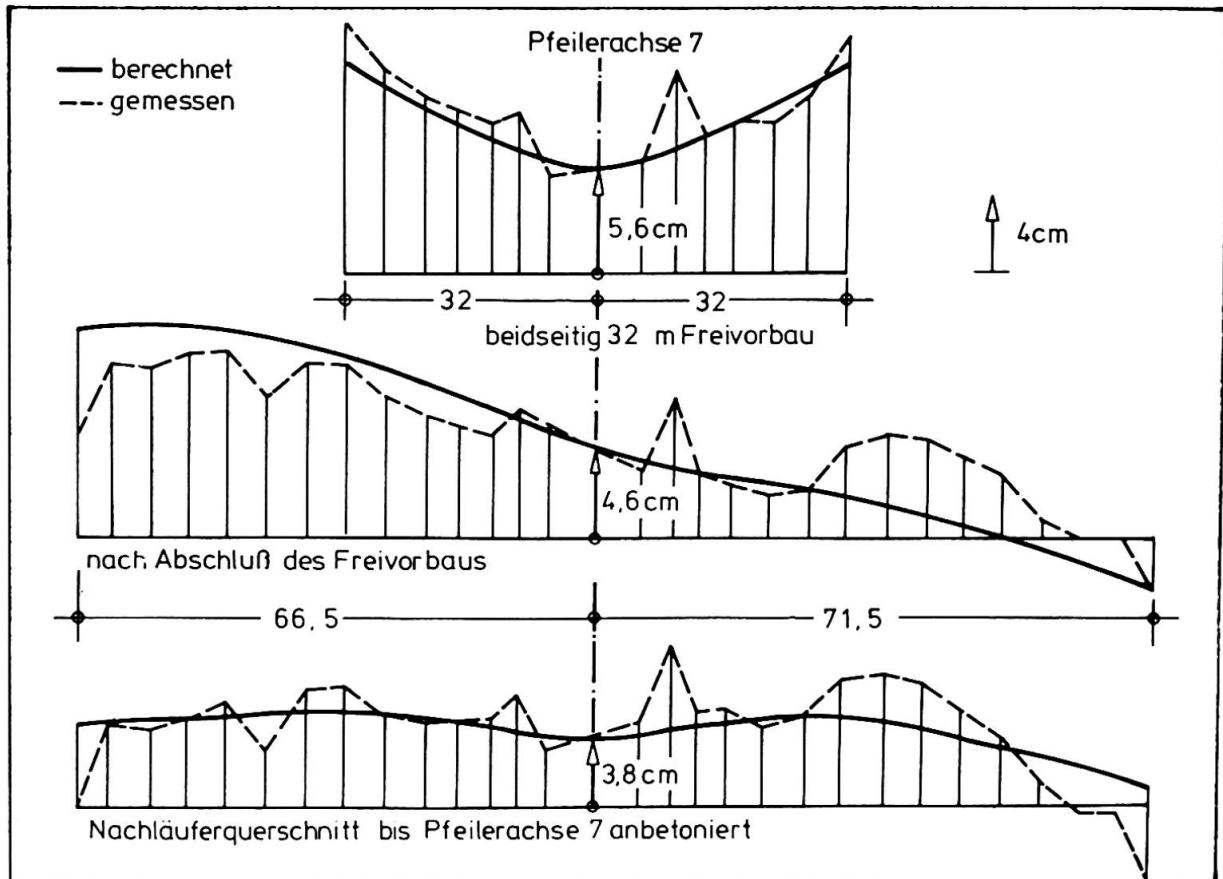


Abb. 7 Überhöhungen des zum Pfeiler 7 gehörigen Freivorbaubereichs

In Abb.7 beträgt die größte Abweichung zwischen gemessener und berechneter Überhöhung 5,5 cm. In Anbetracht der bei der Bauausführung zwangsläufig auftretenden Ungenauigkeiten, den Unsicherheiten bei der Annahme der Kriech- und Schwindbeiwerte und der Regelfeldweite von 138 m, ist diese Genauigkeit als gut zu bezeichnen.

LITERATURVERZEICHNIS

- [1] Schade, D., Haas, W.: Elektronische Berechnung der Auswirkungen von Kriechen und Schwinden bei abschnittsweise hergestellten Verbundstabwerken. Heft 244 des DAFStb, Berlin 1975.
- [2] Richtlinien für die Bemessung und Ausführung von Spannbetonbauten, Fassung Juni 1973, abgedruckt im Betonkalender 1975.
- [3] Haas, W.: Normengerechte Berücksichtigung des zeitabhängigen Verformungsverhaltens von Beton, dargestellt an zwei Beispielen aus dem Hoch- und Brückenbau, abgedruckt in Finite Elemente in der Baupraxis, Verlag Wilhelm Ernst und Sohn, 1978.



VII

Interactive and Automated Design of Cable-Stayed Bridges

Projet de ponts à haubans à l'aide de l'ordinateur

Interaktive und automatisierte Vorbemessung von Schrägseilbrücken

YOSHIKAZU YAMADA
Professor of Civil Engineering
Kyoto University
Kyoto, Japan

DAIZO OKUI
Vice President
Harumoto Iron Works Co., Ltd
Osaka, Japan

HISASHI DAIGUJI
Manager, Engineering Division
Harumoto Iron Works Co., Ltd
Osaka, Japan

SUMMARY

This paper presents the outline of the concept and application of the computer-oriented design system, TOPCAB. This system is primarily used for the preliminary design of cable-stayed bridges. Through the use of TOPCAB an approximate steel weight based on the optimum prestresses in cables and stiffness of girder-cable is promptly obtained.

RESUME

L'article présente un aperçu général de la conception et de l'application du système TOPCAB. Ce système sert essentiellement au projet préliminaire de ponts à haubans. Le poids d'acier approximatif peut être obtenu sans difficulté au moyen du système TOPCAB, sur la base de la précontrainte optimale des câbles et de la rigidité des câbles de poutres.

ZUSAMMENFASSUNG

Das Konzept und die Anwendung des Bemessungssystems TOPCAB werden gezeigt. Dieses System wird hauptsächlich für die Vorbemessung von Schrägseilbrücken, die im modernen Stahlbrückenbau eine wichtige Rolle spielen, verwendet. Mit Hilfe von TOPCAB wird das ungefähre Stahlgewicht unter Berücksichtigung der optimal vorgespannten Kabel und der Steifigkeit der Trägerkabel schnell ermittelt.



1. INTRODUCTION

The present paper deals with the computer use in the structural design considering the main title of Thema(VII) "Computer Analysis and Structural Engineering" in the introductory report of the 11th congress of IABSE.(1)

The authors have engaged in the development of the large scale computer-oriented system in the design and fabrication of steel bridges since 1973(5). The present TOPCAB system(4) is the finally developed computer-oriented design system in the total project including the formerly developed structural optimization techniques(2)(3) and automated drawings, etc.

The total process for the design of cable-stayed bridges is a multistage procedure which ranges from consideration of overall system requirements down to the detailed design of individual components. In the analysis and design of cable-stayed bridges the use of computers is always essential. The development of matrix structural analysis programs has enabled detailed structural analysis of any type of cable-stayed structure appearing in modern bridges.

In the planning design of the cable-stayed bridges the arrangement of the cables in the longitudinal direction of the bridge may vary according to the designer's sense of proportion of clear spans and tower heights. In this stage the designer needs the many design-oriented information to make the best decision among the given choices.

In this paper outline and results of interactive and integrated large scale system named TOPCAB are presented. TOPCAB is conveniently used for the preliminary design, but the modules of input generator and structural analysis are also applicable to the final design.

2. ORGANIZATION OF THE ENTIRE SYSTEM

Since the completely automated design of large system is suffering from inflexibility when faced with the various designer's requirements, interactive and conversational systems are employed for the data generation and structural optimization.

TOPCAB is mainly divided into four basic subsystems and each module of the subsystem and database are controlled by the TOPCAB supervisor. The organization of TOPCAB is illustrated in Fig. 1. The input and output data are generated in the TOPCAB database under the control of the supervisor. This database structure is convenient as input data source and also a media through which many modules exchange data and make a consistent design. All TOPCAB commands and available program names are stored in a relational data structure for easy information restriction and efficient data management. The entire system is designed to be interactive by using the database with relational data structure and the graphic terminal.

3. BASIC FUNCTIONS OF TOPCAB

An integrated system for the design of cable-stayed bridges is organized by making full use of four subsystems provided by TOPCAB. The primary function of each subsystem is outlined as follow.

3.1 Data Generation

For a given bridge span, many variations and combinations of the cable arrangements in the longitudinal direction and the tower elevations can be considered as shown in Fig. 2. If you want to get the design information for all combinations of the types, it is tedious and time consuming to make input card decks in the usual structural analysis.

The structural member data in the longitudinal direction and the transverse direction of any cable-stayed bridges can be generated by using 16 commands with 39 parameters and 5 commands without parameter. Any structural data can be generated by the translation and computation of these commands. TOPCAB has another facility to prepare the database from graphic terminal in demand mode

as shown in Fig. 3. Input data generator is developed to minimize the input data for structural analysis.

3.2 Structural Synthesis

The optimization concept and technique can be used to obtain the initial design-oriented information. Since the designer must make a best choice based on the economic evaluation, he wants to obtain the economic information of considerable type of cable-stayed bridges before beginning the final design. The optimality parameter method (2) is used for the optimization of cable-girder structural system. The newly developed visualized optimization technique using graphic terminal is also applicable to the stress arrangements of cables. Structural optimization technique is also applied to determine the basic dimension and the material combination of steel tower section shown in Fig. 4. Thus the approximate steel weight is promptly computed by full use of the available capabilities of TOPCAB.

3.3 Structural Analysis and Design

The available structural analysis programs in TOPCAB are mainly based on the matrix displacement method. Nonlinearity behavior of the structure can be also considerable by the utility of TOPCAB. The design standard used in the system is "Specifications for Steel Highway Bridges (Japan Road Association)".

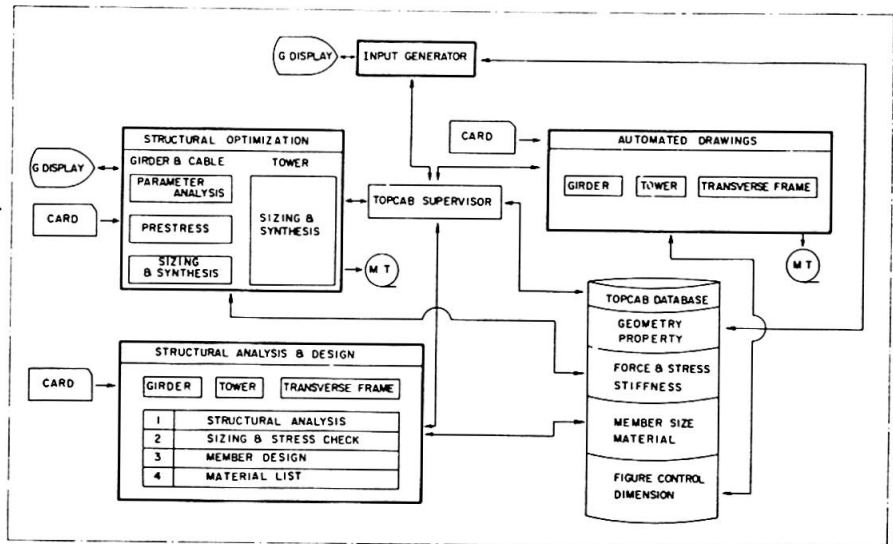


Fig. 1 Organization of the Entire System

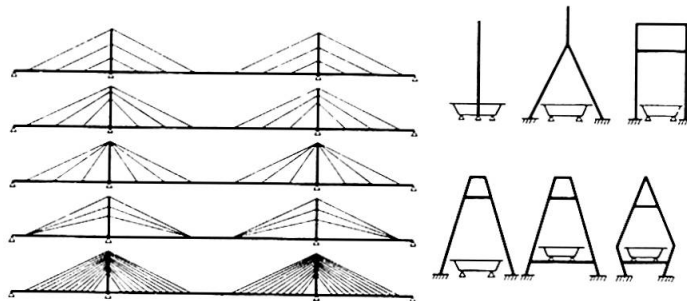


Fig. 2 Variety of the Elevation

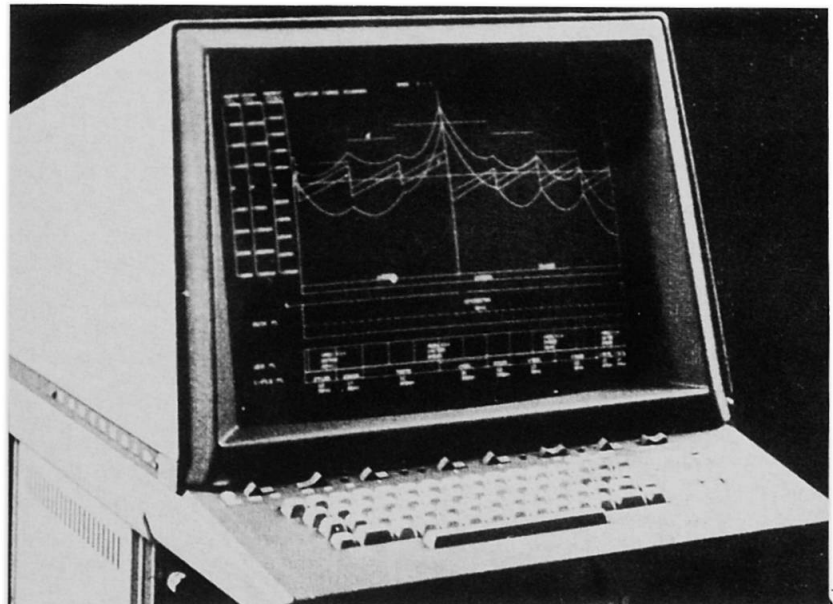


Fig. 3 Graphic Terminal



3.4 Automated Drawing Facilities

Automated drawing facilities can be applied to the drawing of trapezoidal shaped main girder, transverse frame, and tower. The automated drawing facilities are used to make the more accurate economic evaluation in the early design stage by accounting the steel weight of detail member (splice plates, stiffening plates etc.).

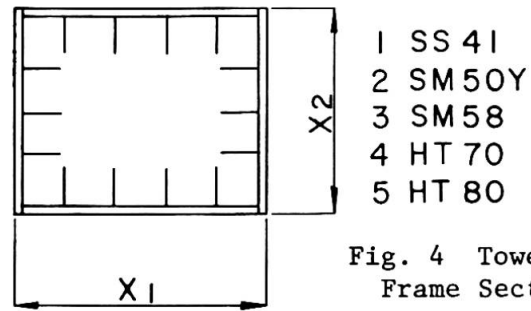


Fig. 4 Tower Frame Section

4. APPLICATION OF TOPCAB

The primary purpose of developed TOPCAB is to provide the designers with practical tools in the preliminary design stage of cable-stayed bridges. Two resulting examples of the application to the actual models are presented herein.

4.1 Optimization of the Prototype Model

Basic bridge section and tower elevation of the analytical model are shown in Fig. 5. Basic side elevation and five different cable arrangements in the longitudinal direction are shown in Fig. 6. The basic dimension of the bridge is very similar to the Yamato River Bridge now under construction in Osaka Prefecture Japan.

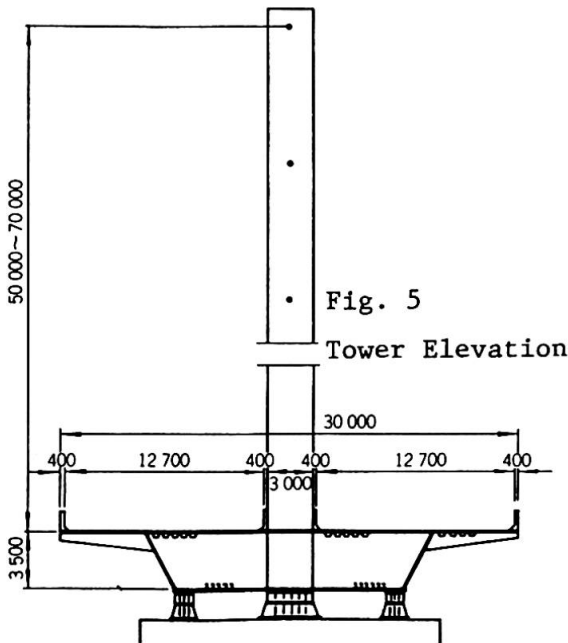


Fig. 5 Tower Elevation

Results of the optimization are given in Table 1. In this study the optimum structural system based on the minimum weight design is the harp type with triple cable arrangements. The section force due to the dead load and live load is shown in Fig. 7.

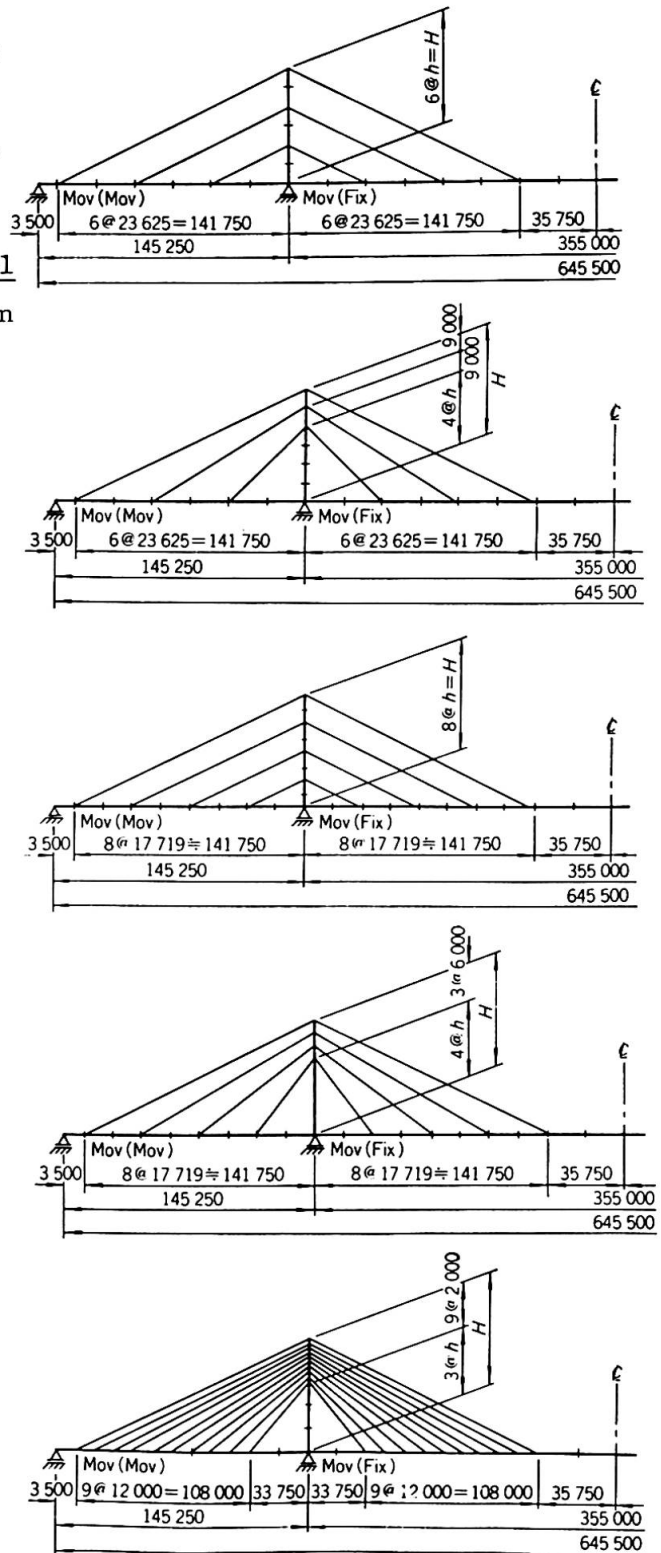


Fig. 6 Cable Arrangements

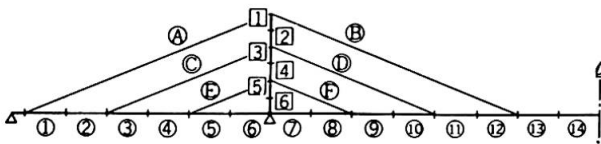
The optimum prestresses in this structural system are given in Fig. 8, and the section force arranged by the optimum prestresses is also drawn in Fig. 7. These section forces and girder sections can be drawn by either the graphic display terminal or high speed NC drafter.

4.2 A Drawing Example by NC Drafter

After completion of the member sizing, all digital data are stored in the TOPCAB database as the input source for the sequential automated drawings, and then the automated drawings are carried out. Fig. 9 and Fig. 10 are drawing examples of the actually completed Toyosato Bridge Osaka Japan.

Table 1 Results of Optimization

CABLE TYPE	TOWER HEIGHT	GIRDER (ton)		CABLE (ton)		TOWER (ton)		SUMMATION (ton)	
		MAIN	ETC.	CABLE	ETC.	SM58	HT70	SM58	SM70
HARP 3-CABLE	50 m	4 860	2 040	780	340	670	590	8 690	8 610
	60	4 830	2 040	710	340	930	820	8 850	8 740
	70	4 840	2 040	640	340	1 230	1 150	9 090	9 010
	50	4 900	2 040	830	340	670	590	8 780	8 700
	60	4 840	2 040	760	340	900	810	8 880	8 790
	70	4 830	2 040	690	340	1 200	1 120	9 100	9 020
FAN 3-CABLE	50	4 830	2 040	770	340	770	680	8 750	8 660
	60	4 820	2 040	690	340	1 080	980	8 970	8 870
	70	4 830	2 040	670	340	1 390	1 330	9 270	9 210
	50	4 800	2 040	830	340	730	650	8 740	8 660
	60	4 820	2 040	760	340	1 080	980	9 040	8 940
	70	4 830	2 040	730	340	1 380	1 320	9 320	9 260
MULTI 4-CABLE	50	4 840	2 040	970	340	740	660	8 930	8 850
	60	4 850	2 040	840	340	1 090	990	8 160	8 060
	70	4 830	2 040	830	340	1 380	1 330	9 420	9 370



Section Force Diagram Stiffening Girder

★ : SM50Y

1	2	3	4	5	6	7	8	9	10	11	12	13	14
12	12★						13	12					
14	14★						14	14★	14	14★			
12	12★	13★	12★	13★	12★	13	13★	12★	12	12★			
2.301	2.389	2.301	2.389	2.301	2.427	2.301							

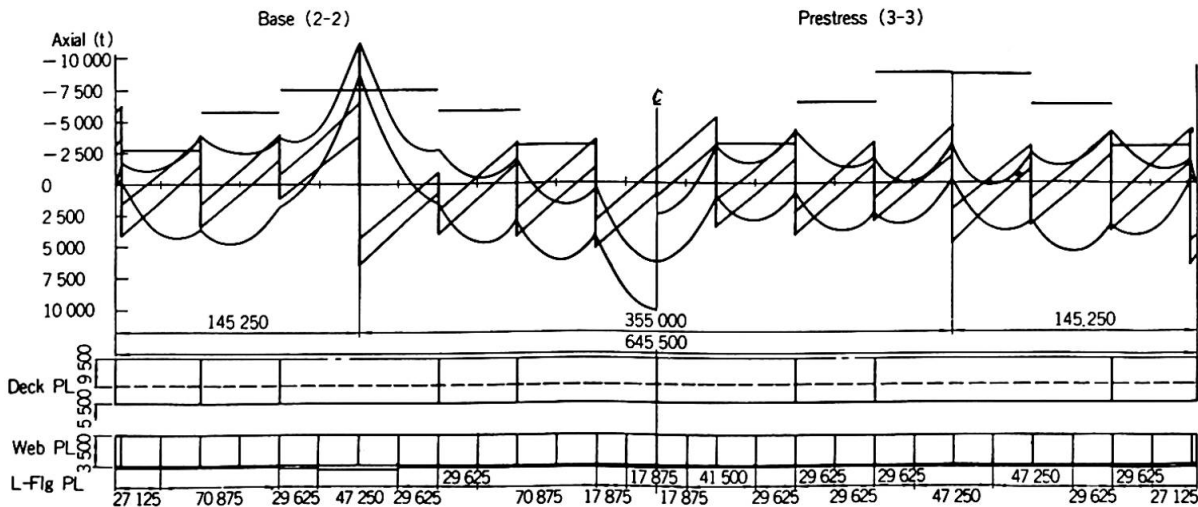
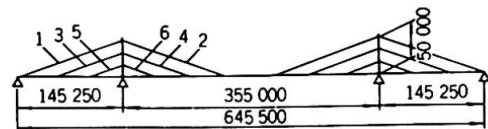


Fig. 7 Section Force Diagram and Basic Girder Section

5. CONCLUDING REMARKS

The authors formerly developed the large scale integrated design system for steel I-girder bridges and fabrication system using NC equipment. Those systems are now successfully used in the design offices and fabricating shops. TOPCAB was developed by the structural engineers and system engineers engaged in the former project. TOPCAB system can be conveniently used for the preliminary design of cable-stayed bridges since the cable-stayed bridge is one of the most sophisticated bridge structure appearing in modern bridges and it require the multi process of complicated computer using structural analysis.



NO.	PST	D + L + PS (t)	PWS-217
1	170	3 655	23
2	170	3 551	23
3	400	3 224	19
4	400	3 354	19
5	700	2 453	15
6	700	2 503	15

Fig. 8 Optimum Prestresses

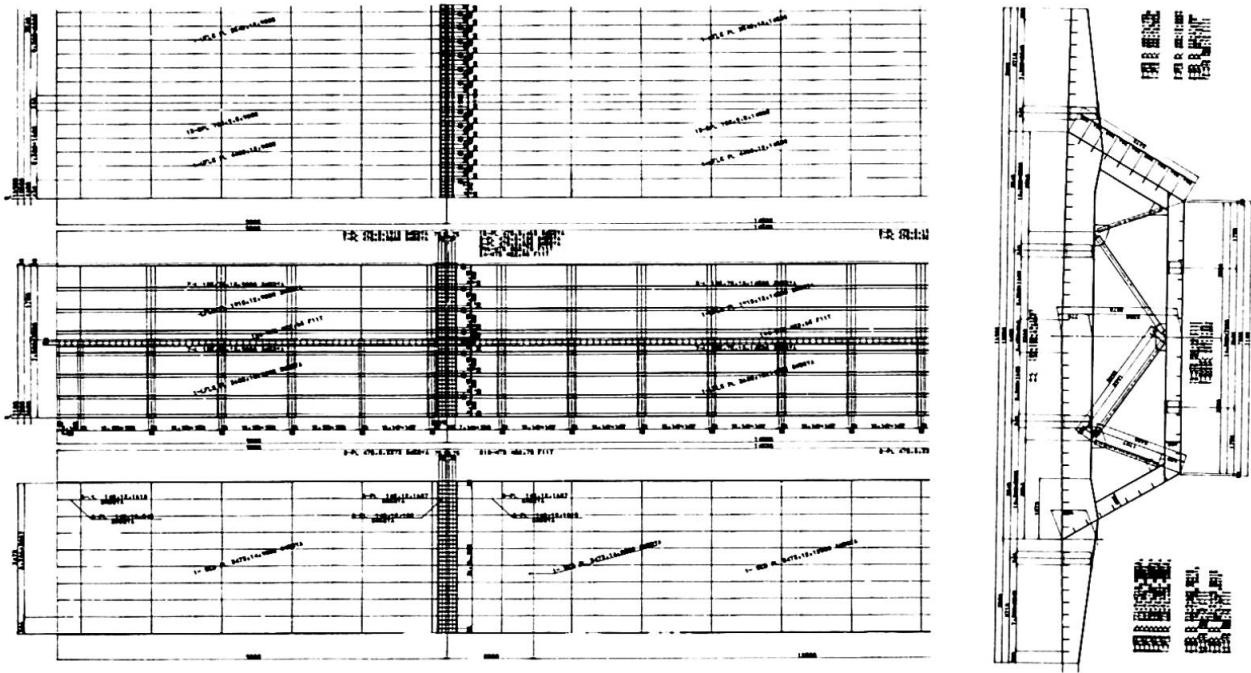


Fig. 9 Automated Drawing Example of Stiffening Girder

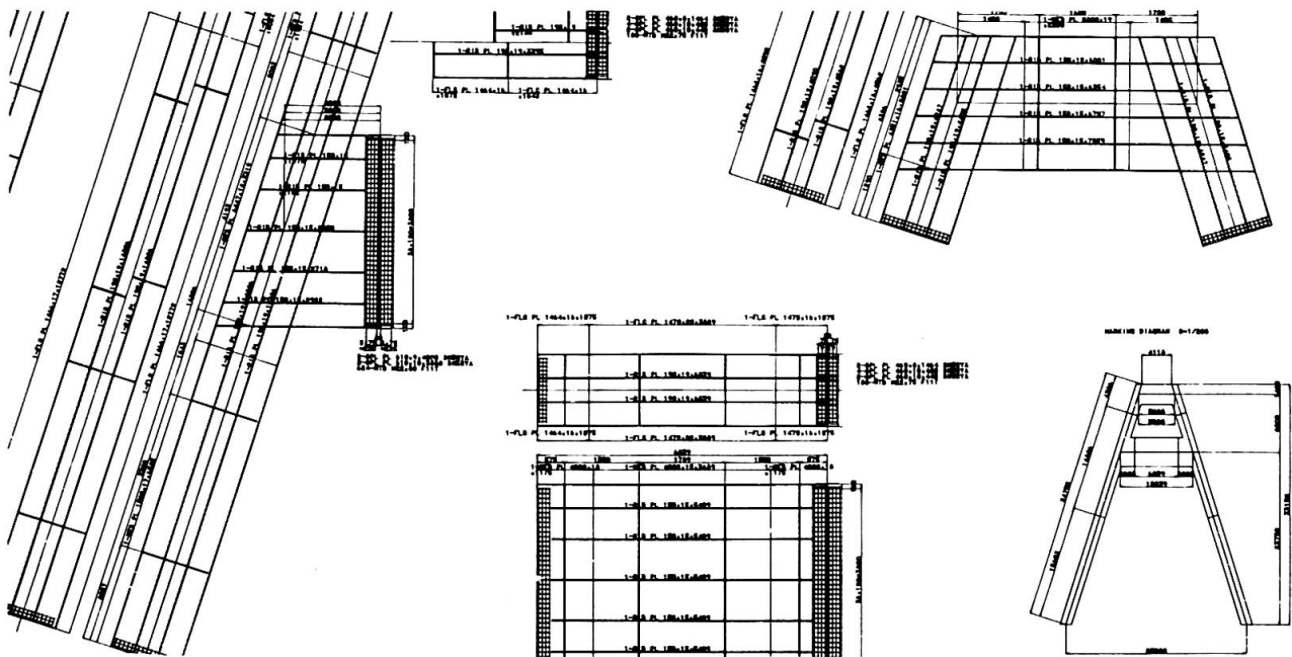


Fig. 10 Automated Drawing Example of Tower

REFERENCES

1. Alcock, D.: Computers in Structural Design: Some General Thoughts, Introductory Report of Eleventh Congress of IABSE, Vienna, Sep. 1980.
2. Daiguji, H., Yamada, Y.: Optimum Design of Cable-Stayed Bridges using an Optimality Parameter, Final Report of Tenth Congress of IABSE, Tokyo, Sep. 1976.
3. Yamada, Y., Daiguji, H., Imamura, K.: Extended Study of the Optimality Parameter Design for Cable-Stayed Bridges, Bridge and Foundation Engineering Jan. ~ Feb. 1978.
4. Daiguji, H., Mizutani, K.: Total System for the Preliminary Design of Cable-Stayed Bridges, Bridge Engineering, Jan. 1979.
5. Daiguji, H., Imamura, K., Kimoto, E., Hamamura, M.: Application of the Computer Oriented Total System to the Design and Fabrication of Steel Bridges, 23th National Symposium on Bridge and Structural Engineering, Tokyo, 1977.



VII

Computer Design of Complex Building Structures

Projet de structures complexes de bâtiments à l'aide de l'ordinateur

Planung komplexer Baukonstruktionen mit Hilfe des Computers

SVEND E. PETERSEN

Cowiconsult,
Consulting Engineers and Planners AS
Copenhagen, Denmark

TOMMY BUNCH-NIELSEN

Cowiconsult,
Consulting Engineers and Planners AS
Copenhagen, Denmark

ARON ARKIN

Cowiconsult,
Consulting Engineers and Planners AS
Copenhagen, Denmark

SUMMARY

This paper deals with the use of computers in the design of complex building structures such as structures in sport centres and industrial buildings. Rather than use large integrated computer systems it is preferred to develop chains of programmes for each of the different building components: beams, columns, frames, etc. Such a chain of programmes for reinforced concrete columns is described with special emphasis on the analytical aspects.

RESUME

Ce rapport décrit l'utilisation de l'ordinateur dans l'étude de structures complexes telles que les structures de centres sportifs et de bâtiments industriels. Au lieu d'utiliser des systèmes de calculs très intégrés il est préférable de développer une chaîne de programmes pour chaque élément de construction: poutres, poteaux, portiques, etc. Une telle chaîne de programmes pour poteaux en béton armé est décrite en mettant l'accent sur les aspects analytiques.

ZUSAMMENFASSUNG

Der Artikel behandelt den Einsatz von Computern bei der Projektierung komplexer Baukonstruktionen wie Sportzentren oder Industriebauten. Es wird vorgeschlagen, anstelle komplizierter Computerprogramme eine Reihe von Teilprogrammen für die einzelnen Bauelemente wie Stützen, Balken, Rahmen etc. zu benutzen. Ein solches Teilprogramm für Stahlbetonstützen wird beschrieben.



1. INTRODUCTION

Large integrated computer systems for calculation and drawing of building structures become extensive, elaborate and overwhelming if they shall cover the complexity and the variety of structures facing the designer of buildings.

Buildings for sports and industrial production have to be designed to meet individual needs. The variation from project to project is so great that systematic computerized design of the total structural system is meaningless.

However these individual projects often contain problems which can only be solved properly with computer aid. In our building department we have therefore put emphasis on the development of advanced programmes for calculation of individual structural members as columns, beams, frames, foundations and slabs.

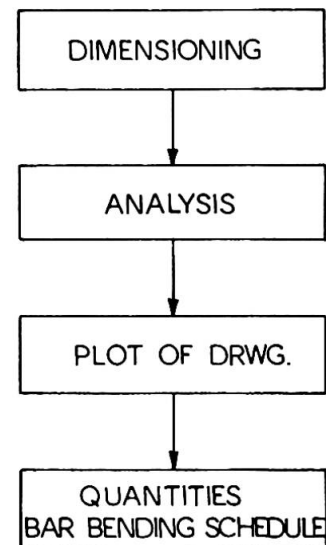
2. BASIC CONCEPT

As the programme library grows in width - covering more and more structural elements - and in depth - covering all phases of design from dimensioning to prepairing of bill of quantities - the need for a framework, a common basic concept, is strongly felt.

The system under development consists of chains of programmes for the individual structural members consisting of programmes for dimensioning, analysis, plotting of drawings, measuring and storing of quantities and, for reinforced concrete, extracting bar bending schedules.

From these chains of programmes information may be collected and used in programmes to produce the synthesis in the form of layout drawings, joint details, bill of quantities etc.

To illustrate the structure of these chains the programmes for calculation and drafting of reinforced concrete columns are described in the following.



3. PROGRAMME CHAIN FOR REINFORCED CONCRETE COLUMNS

Traditional simplified analysis of reinforced concrete columns disregarding such effect as the variation of stiffness with the extent of cracking, the non-linear stress/strain diagramme, the true shape of the moment curve etc. are in many cases not satisfactory.

One of our first programmes was therefore a column programme which took these effects into account (2), (3). This first generation programme, which was limited to statical determinate columns with rectangular or circular cross section and subject to mono-axial bending, has now been replaced by a second generation programme. This programme analysis statically indeterminate reinforced concrete columns with arbitrary polygonal, solid or hollow, section and subject to variable normal forces and biaxial bending.

3.1 Dimensioning

The possibility of developing a programme for dimensioning has been considered and given up for two reasons: It would be complicated and the geometrical properties depend to a great extent on irrational considerations e.g. aesthetics which is not well suited for computers.

3.2 Analysis

The programme uses a finite element method to calculate stresses and strains in a reinforced concrete column subject to biaxial bending and normal force with due regard to the physical and geometrical non-linearities.

A space beam element with an inner joint is used in the finite element formulation. This element has 11 degrees of freedom as torsion is neglected. The solution to the non-linear problem is arrived at by an iterative process divided into a physical and a geometrical iteration.

The purpose of the physical iteration is to determine the stiffness constants corresponding to the actual stress/strain distribution. The stiffness constants are calculated as the area integral over the cross section of the product of the geometrical properties and the secant modulus of elasticity based on the elastoplastic stress/strain diagrams for concrete and steel. It can be shown (1) that this procedure leads to the exact correlation between generalized strains and stresses.

In the integration the reinforcement is considered placed in points and the surface integral over the compressed concrete area is transformed to a line integral using Gauss' integral law i.e. the integration is performed along the periphery of the compressed concrete area.

The purpose of the geometrical iteration is to determine the deflection curve corresponding to the stresses and strains in the sections in a state where the column is in equilibrium subject to moment and normal force and taking into account second order effects i.e. the moments from the normal force due to the deflection. It is the second order phenomena which necessitate the iterative process to solve the system equations.

In the development of the element matrices, it is taken into account that the centerline of the system not necessarily coincide with a line through the gravity centers of the elements. The use of elements with an inner joint is necessitated by this fact. The system matrix is composed of the traditional stiffness matrix, \bar{K}_O , the geometric matrix, \bar{K}_G , the geometric matrix for large deformations, \bar{K}_L , and a new matrix called the load dependent geometric matrix, \bar{K}_F . Thus the system matrix has the following form:

$$\bar{K}_O = \bar{K}_O + \bar{K}_G + \bar{K}_L + \bar{K}_F$$

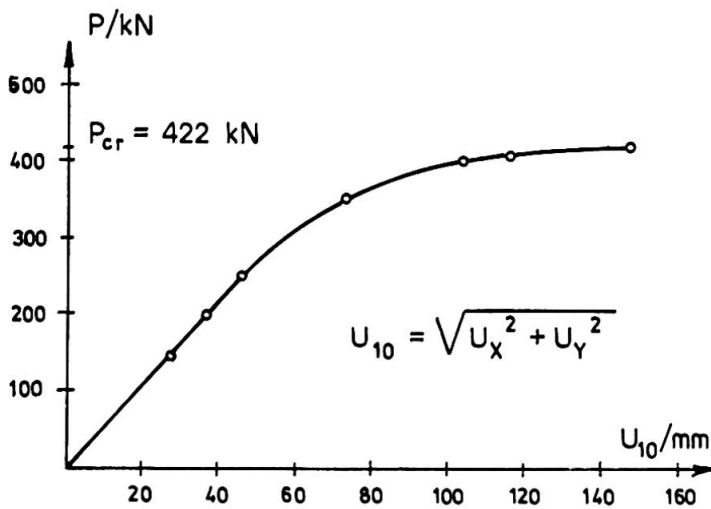
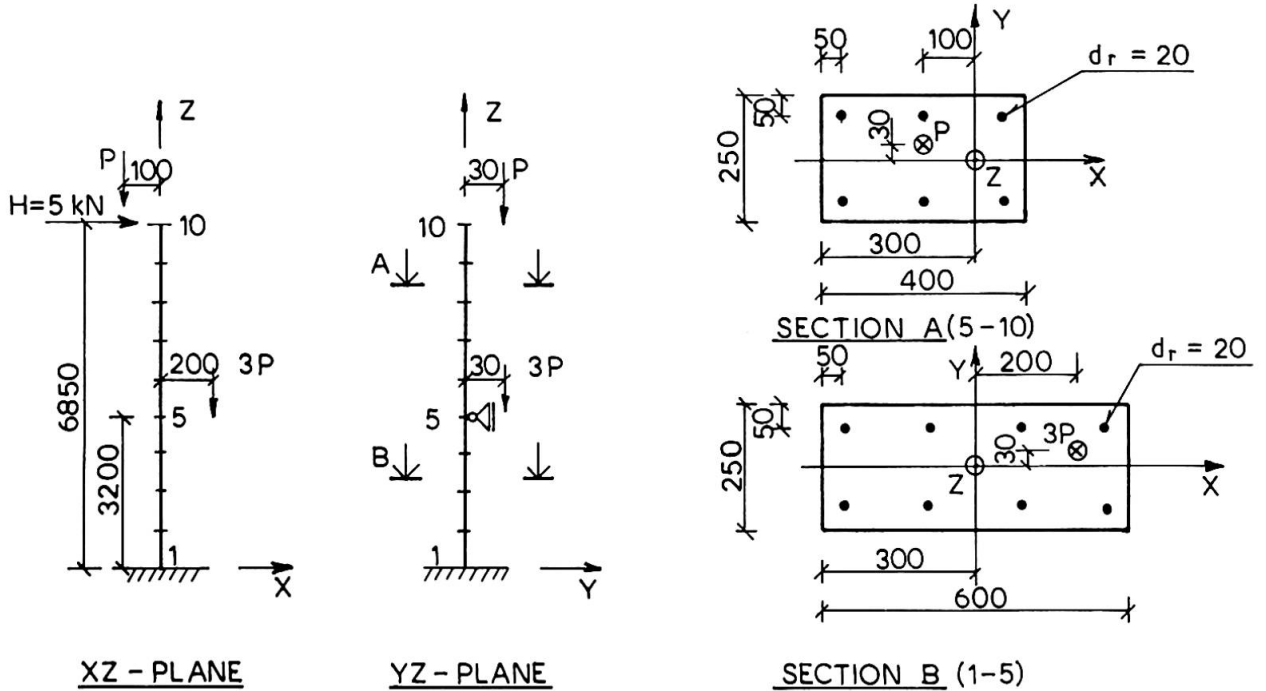
The geometrical iteration process is accelerated by means of a Newton-Raphson method.



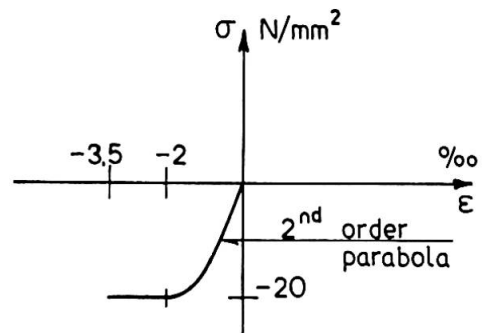
The core of the programme is the general finite element part, which provides the means for dealing with the support conditions in a general way. One of the advantages is the possibility of analysing multispans columns.

The programme is an analysis programme that calculates stresses and strains in a columns with given geometry, reinforcement and load.

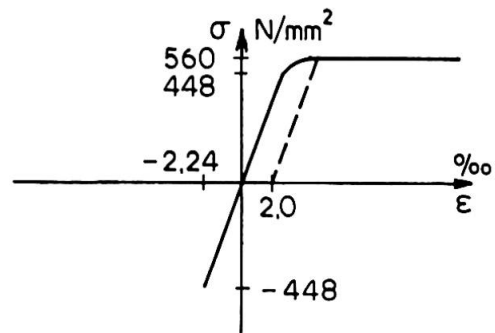
By stepwise increase of the load on the columns it is possible to determine the critical load. The positive definiteness of the system matrix is used as criterion for stopping. This procedure is illustrated below.



LOAD/DEFLECTION CURVE
(Deflektion in point 10)



STRESS-STRAIN CURVE: CONCRETE



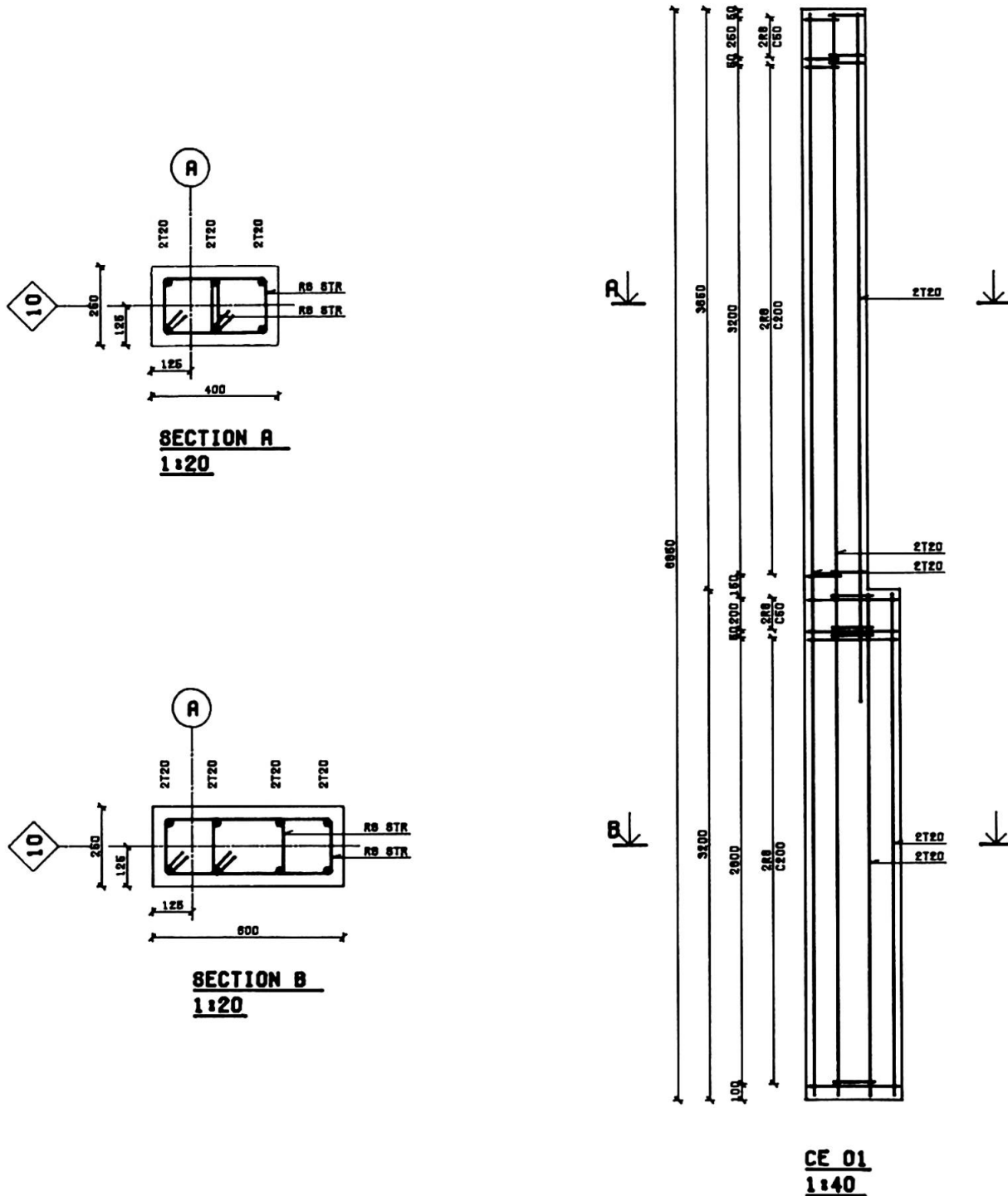
STRESS STRAIN CURVE: REINFORCEM.



3.3 Drawing programme

The next link in the chain is the plotter programme which draws up the complete reinforcement drawing based on the data used in the analysis programme and a supplement of drafting instructions. The drawing is produced on a computer controlled plotter.

An example is shown on the figure.





3.4 Calculation of quantities and working out of bending schedules

Using the same data as used in the plotter programme the quantities of concrete, formwork, and reinforcement can be measured and stored for use in the bill of quantities.

Furthermore data may be drawn from the plotter programme and used to produce the corresponding bar bending schedule.

4. CONCLUSION

The programmes, the analysis programme as well as the plotter programme, have build in check of the structural rules given in the Danish Code of Practice for Reinforced Concrete. This facility can naturally be extended to any kind of Code of Practice.

The output of the analysis programme can be displayed on a printer or on a graphic computer terminal, where the results are shown as curves for moment, stresses and deflections.

The latter facility provides a simple and efficient check of the calculation. Commonplace errors and misunderstandings will normally be revealed.

It has furthermore proved to be of utmost importance for the safe use of the programme that it has been developed by engineers in the same department where it shall be used and in close collaboration with the users. This means that the daily use of the programmes takes place in close contact with engineers who have a thorough knowledge of the way the programmes operate. This close collaboration between programmers and designers also secures that the programmes are adjusted, updated and supplemented according to the needs of the users.

5. REFERENCES

- (1) M.Menegotto and P.E.Pinto
Slender RC Compressed Members in Biaxial Bending
ASCE, St 3, March 1977.
- (2) A.Hougaard Nielsen and S.E.Petersen
EDB-beregning af jernbetonsøjler
Nordisk Betong 2, 1973
- (3) S.E.Petersen
Some Remarks on the Preliminary Report
Final Report, IABSE Symposium
Quebec 1974.

VII

Eine effektive Methode zur Spannungsanalyse

Efficient Method for Stress Analysis

Méthode effective pour l'analyse des tensions

JAN KMITA

Prof. Dr. -Ing.
Institut für Hoch- und Tiefbau,
Technische Universität Wroclaw
Wroclaw, Polen

JAN BIEN

Dr. -Ing.
Institut für Hoch- und Tiefbau,
Technische Universität Wroclaw
Wroclaw, Polen

CZESLAW MACHELSKI

Dr. -Ing.
Institut für Hoch- und Tiefbau,
Technische Universität Wroclaw
Wroclaw, Polen

ZUSAMMENFASSUNG

Der Beitrag stellt eine aus zwei Etappen bestehende Methode der statischen Analyse von prismatischen Konstruktionen vor. Die Lösung basiert auf der Deformationsmethode. Bei der Aufteilung der Konstruktion in Elemente werden zwei- und dreidimensionale finite Elemente angewandt. Im Vergleich zu anderen Methoden unterscheidet sich die hier vorgeschlagene durch bedeutend kleineren Rechenaufwand, wobei sie eine gute Übereinstimmung des Rechenmodells mit der wirklichen Konstruktion sichert.

SUMMARY

The article presents a two-stage method for statical analysis of prismatic structures. The solution is based on the stiffness method. In the discretization procedure the two- and threedimensional finite elements are used. The method proposed is considerably less labour consuming when compared with other methods and assures a good agreement between the idealised and real structures.

RESUME

L'article présente une méthode en deux étapes pour l'analyse d'une construction prismatique. La résolution est basée sur la méthode des déplacements; des éléments bi- et tri-dimensionnels ont été utilisés pour la discrétisation de la construction. La méthode proposée présente une diminution de calcul par rapport aux autres méthodes et présente une bonne concordance du modèle de calcul et de la construction réelle.

1. EINLEITUNG

Grosse Bedeutung beim Entwurf hat die genaue Bestimmung der Verteilung von Spannungen von bestimmten Teilen der Tragwerke wie z.B. Verbindungen der Elemente, Angriffspunkte der Einzellasten. Die Wahl eines geeigneten Rechenmodells wird einerseits von der geforderten Genauigkeit und andererseits von der Grösse des Gleichungssystems bestimmt. Das Problem liegt also im Suchen nach einem rationellen Kompromiss zwischen der Abbildungstreue der geometrischen und physischen Objektkennzeichen und den Rechenkosten. In der vorliegenden Arbeit werden die Anwendungsmöglichkeiten von zwei- und dreidimensionalen Elementen in der 2-Etappen-Lösung dargestellt. Dies erlaubt eine genaue statische Analyse mit Anwendung von kleineren Digitalrechnern wie auch bedeutende Senkung der Rechenkosten.

2. GEOMETRISCHE AUFTeilUNG DER KONSTRUKTION IN ELEMENTE

Als Vorbild für den Grundgedanken der vorgeschlagenen Methode dient hier die Analyse von Plattenbalkenkonstruktionen (Bild 1a). In der ersten Berechnungsetappe benutzt man bei der Systemlösung spezielle zweidimensionale Plattenscheibenelemente [1, 2] mit einer geringen Anzahl von Knotenparametern (Bild 1b). In der zweiten Etappe werden die gewählten Konstruktionsbereiche unter Benutzung dreidimensionaler Elemente aufgeteilt (Bild 1c) [3, 4]. Die äussere Belastung besteht aus den in der ersten Etappe bestimmten Schnittkräften und den unmittelbar angreifenden Kräften.

Die erarbeitete Programme ermöglichen die Analyse von prismatischen Konstruktionen, die ein beliebiges Querschnittsprofil besitzen, gerade oder gekrümmt im Grundriss sind, an den Enden frei gelagert werden und Mittelaufleger besitzen.

3. EIN RECHENBEISPIEL

Praktische Anwendung der dargestellten Konzeption wird am Beispiel des freiaufliegenden, unsymmetrisch belasteten Brückenfeldes dargestellt (Bild 1). Die ermittelten Spannungsverteilungen auf der oberen und unteren Fläche der Fahrbahnplatte und in den Trägerstegen werden auf den Bildern 2 und 3 dargestellt. Die punktierte Linie zeigt die in der ersten Etappe (für die belastete Konstruktionshälfte) bestimmten, die kontinuierliche Linie die in der zweiten Etappe genauer berechneten Spannungen. Der Unterschied zwischen den in der ersten und den in der zweiten Etappe berechneten Verschiebungen ist nicht grösser als 4,5 %. Diese Tatsache zeugt von der Richtigkeit der vorgeschlagenen Lösungskonzeption.

4. SCHLUSSFOLGERUNGEN

Die dargestellte Methode kann sowie beim Entwurf, als auch zur Kenntnisanalyse der statischen Konstruktionsarbeit und im Falle von Schadensfällen zur Erklärung deren Ursache angewandt werden.

Die Hauptmerkmale der oben genannten Methode sind :

- kurze Berechnungszeit,

- Anwendungsmöglichkeit von Digitalrechnern mit kleinen Arbeitsspeichern,
- beliebige praktische Genauigkeit der Beurteilung des Spannungszustands ausgewählter Konstruktionsteile.

LITERATURANGABEN

1. KMITA J., MACHELSKI CZ.: Analiza pracy mostowych płyt pomostowych. *Archiwum Inżynierii Lądowej*, H.4, 1978.
2. MACHELSKI CZ.: Wytężenie płyt w układach płytowo-żebrowych. Diss., Bericht PRE Nr. 35/79 des Instituts für Hoch- und Tiefbau der TU Wrocław, 1979.
3. ZIENKIEWICZ O.C., TOO J.J.M.: The finite prism in analysis of thick simple supported bridge boxes. *The Institution of Civil Engineers, Proceedings, Part 2, Vol.53, 1972.*
4. BIEN J.: Analiza statyczna pręseł mostowych metodą pryzmatycznych elementów skończonych. Diss., Bericht PRE Nr. 3/79 des Instituts für Hoch- und Tiefbau der TU Wrocław, 1978.

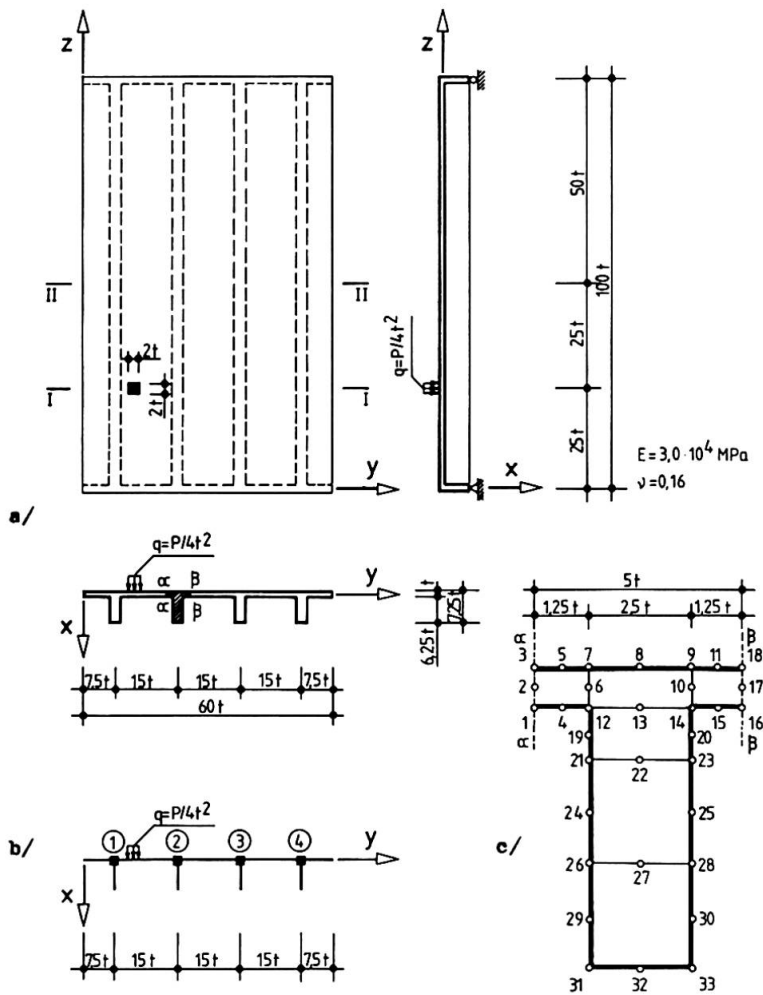


Bild 1. Konstruktionbezeichnung und Aufteilung der Konstruktion in Elemente.

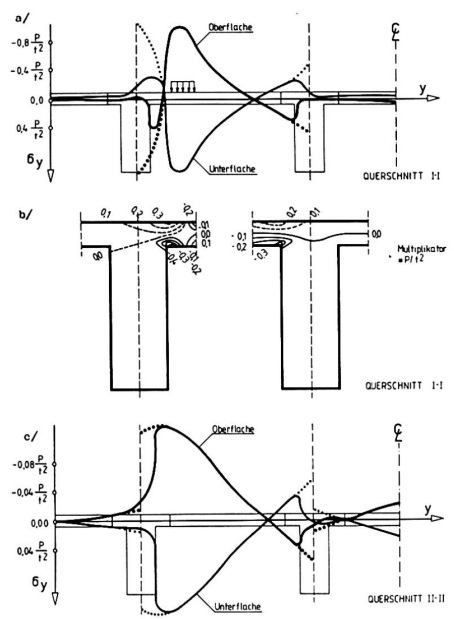


Bild 3. Normalspannungen σ_y in Fahrbahnplatte und Trägerstegen

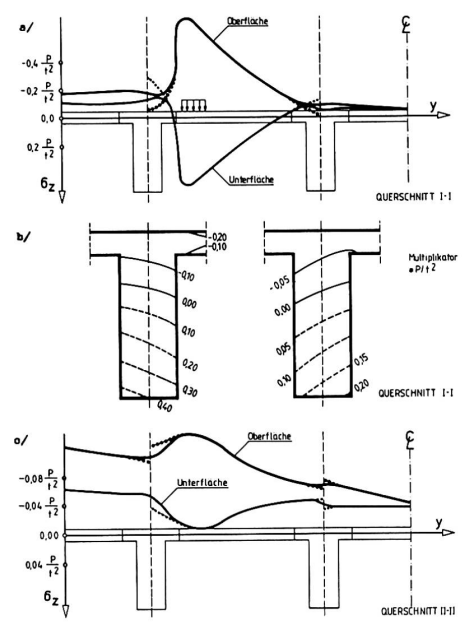


Bild 2. Normalspannungen σ_z in Fahrbahnplatte und Trägerstegen.



Interactive Computer Graphics in Structural Engineering

Représentation graphique interactive dans les projets de génie civil

Interaktive graphische Darstellungsmethoden im Bauingenieurwesen und beim Entwurf

D.P. GREENBERG

Professor, Director, Program of Computer Graphics
Cornell University
Ithaca, NY, USA

J.F. ABEL

Associate Professor of Structural Engineering
Cornell University
Ithaca, NY, USA

W. McGUIRE

Professor of Structural Engineering
Cornell University
Ithaca, NY, USA

SUMMARY

The presentation graphically depicts the advantages of the uses of interactive graphics in structural engineering and design. Full color images of numerous projects conducted at Cornell University's Program of Computer Graphics were shown. These included two and three dimensional frames, two dimensional plates, shell surfaces, and cable and membrane structures. Results of nonlinear geometric and material behaviour, as well as dynamic analyses were illustrated. Color was used to represent stress levels and also provide realistic simulations of three-dimensional structures.

RESUME

L'article représente les avantages de l'emploi de la représentation graphique interactive dans les projets de génie civil. Quelques projets réalisés à l'Université Cornell sont illustrés. Il s'agit de cadres bi- et tridimensionnels, de plaques, coques, structures en câbles et membranes. Le comportement non linéaire du matériau et les études dynamiques sont illustrés. La couleur est utilisée pour représenter des états de contraintes et permettre la simulation réaliste de structures tridimensionnelles.

ZUSAMMENFASSUNG

Der Artikel enthält eine Übersicht über die Vorteile der Verwendung interaktiver graphischer Darstellungsmethoden im konstruktiven Ingenieurbau und beim Entwurf. Farbbilder von zahlreichen an der Cornell Universität ausgeführten Objekten werden gezeigt. Diese schliessen zwei- und dreidimensionale Rahmen, Platten, Schalen, Kabel- und Membran-Tragwerke ein. Resultate von nonlinearem geometrischem und materiellem Verhalten sowie dynamische Analysen werden beschrieben. Farbe wurde verwendet, um Spannungsintensität anzuzeigen und um realistische Nachahmungen von dreidimensionalen Tragwerken zu verschaffen.



1. INTRODUCTION

During the past two decades, a number of important technological advances have occurred in both the computer industry and structural engineering. These advances have created an environment conducive to the marriage of computer graphics and structural design.

- A. The cost of computing is decreasing exponentially.
- B. Analytical techniques, such as matrix analysis or finite element analysis, have been refined.
- C. The complexity and detail of problems we are now able to analyze and examine are rapidly increasing. Large scale problems with many thousand unknowns are now commonly investigated.

Unfortunately, the time required to accurately define the information necessary for a computer analysis is excessive. It may take many man-weeks to properly prepare this input information. Furthermore, the cost of this labor, particularly skilled engineering labor, is expensive. Perhaps 85% of the typical cost of a finite element analysis may be in the input task.

If this situation is unacceptable, an even worse problem exists in the interpretation of the results. One sometimes spends many hours searching through pages of computer output trying to understand and interpret the analysis.

All of this leads to the conclusion that another method for communication between man and the MACHINE is necessary. Furthermore, it is obvious that this method should be graphical.

We live in a visual world. Our ability to comprehend graphical information far exceeds our ability to understand verbal or numerical information. The largest percent of our informational intake is through our eyes. In short, "a picture is worth a 1,024 words!"

Fortunately, advances in the computer graphics industry now make it possible to communicate graphically, both for problem descriptions and the display of results. Digitizing tablets with a stylus for user interaction are common. Hardware is available for dynamic black and white displays or high resolution color displays. Through software, we can create realistic images of things which do not exist.

It is our belief that the future of design engineering will rely on this technology and, specifically, on the uses of INTERACTIVE COMPUTER GRAPHICS.

It is important to emphasize the meaning of the word "interactive" in computer graphics. When graphical operations and commands are specified, it is necessary to have response times fast enough to provide a continuous communication dialogue with the user. For three-dimensional investigations or dynamic problems, continuous motion displays are necessary. In order to fully understand complex geometries, one would like to simulate walking around the structure, like taking a model in your hands and turning it around to examine it. This requires the rapid generation of perspective images, maybe 30 or 40 times per second, in order to imply motion. Color is also useful, not only to provide a realistic three-dimensional image, but to display results.

These concepts will be illustrated by showing you the results of several research projects which have been conducted at Cornell's Program of Computer



Graphics.

In 1974, Cornell University established an interdisciplinary Center of Computer Graphics dedicated to the development of graphics techniques and the use of these techniques in research applications. The center now is one of the most advanced computer graphics laboratories in the world and has obtained a substantial amount of research funding during the past six years. In addition to the basic computer graphics research, investigations have been conducted in such application areas as water resources planning, medicine, architecture, and cartoon animation. We are currently creating a new instructional facility in Computer-Aided Design. The ultimate mission of this center will be to guarantee all 2,000 undergraduate engineering students at least one course on this interactive graphics system. A diagram of the laboratory is shown in Figure 1.

In the remaining time I wish to briefly show the results of some of our specific efforts in structural engineering. The work is not my own, but represents the contribution of my co-authors, Professor John Abel and Professor Bill McGuire, as well as the efforts of a large number of graduate students over the past five years.

I apologize for the medium of presentation as static images are obviously not the best way to illustrate interactive graphics. However, without the use of video projection equipment or constrained by the inability to transport my laboratory to the conference, these slides will have to suffice.

2. VISUAL PRESENTATION OF SLIDES (Examples shown in Figures 2-9)

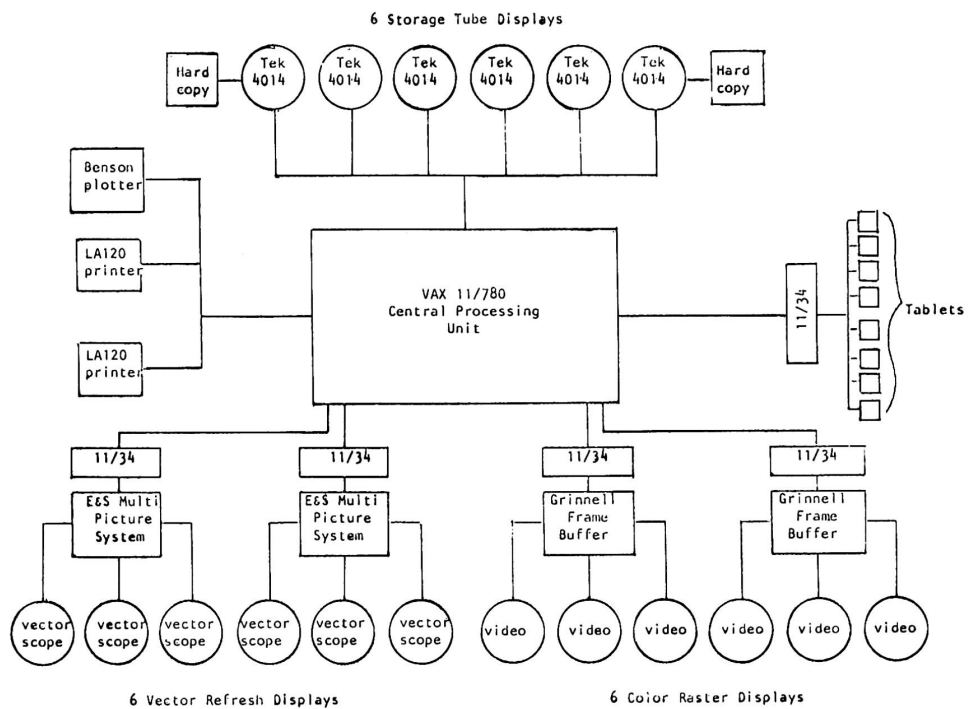
3. CONCLUSION

It should be obvious that INTERACTIVE Computer Graphics is an important tool demanding further refinement. It may not be obvious that it is much more than a tool. The advent of the electronic computer did not result in additions to the laws of mechanics, but it has revolutionized structural and geotechnical engineering. The engineer who truly understands the methods developed in response to the computer's capabilities has a broader insight into system behavior than his forebearer whose perceptions were restricted by the limited power of older methods. Interactive computer graphics should have a similar impact on the profession.

In conclusion, let me state that INTERACTIVE Computer Graphics offers the best promise for inserting the designers ability into the iterative design and analysis feedback loop. More importantly, it will offer the opportunities for new insights, new understandings, and new design concepts simply because it reduces the barriers of communication and thus will allow man to more effectively use the machine.

I hope this brief talk has shown that the technology is here, and is waiting to be accepted. To quote Ralph Waldo Emerson, "An idea whose time has come is the most potent of all natural forces".

Thank you.



SCHMATIC EQUIPMENT CONFIGURATION
Figure 1



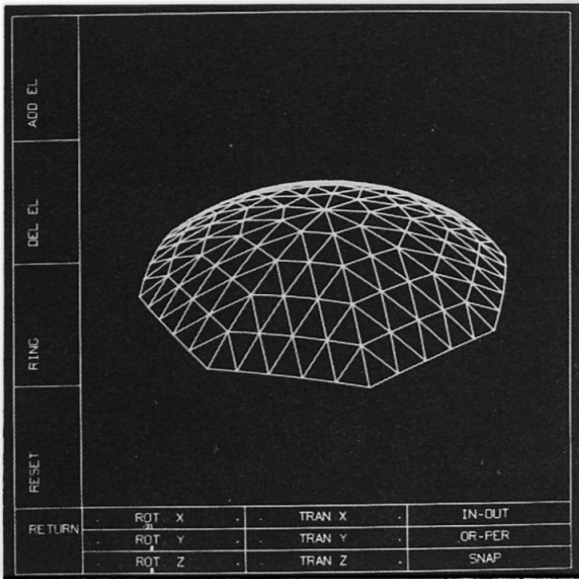


Fig. 2

Example of a varax dome structures interactively modeled on a refresh vector display.

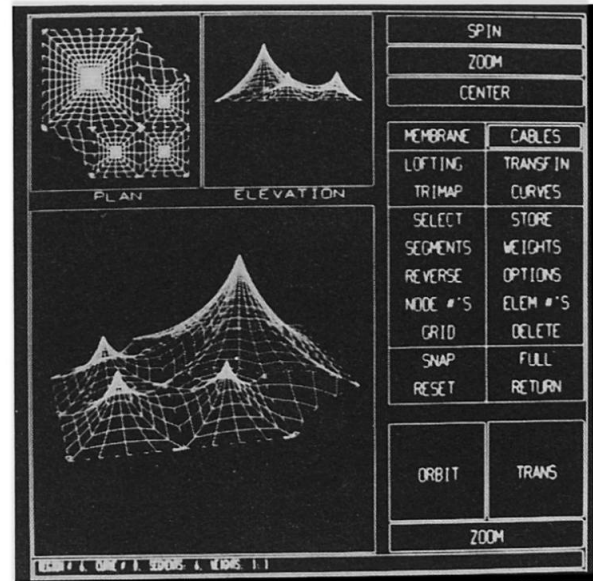


Fig. 3

Example of a cable or membrane structure.

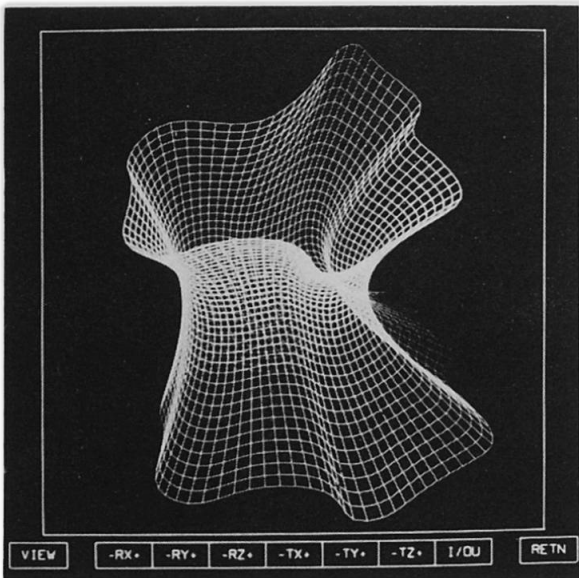


Fig. 4

Example of an arbitrary surface interactively modeled using B-splines for the sectional contours and lofted with cardinal splines.

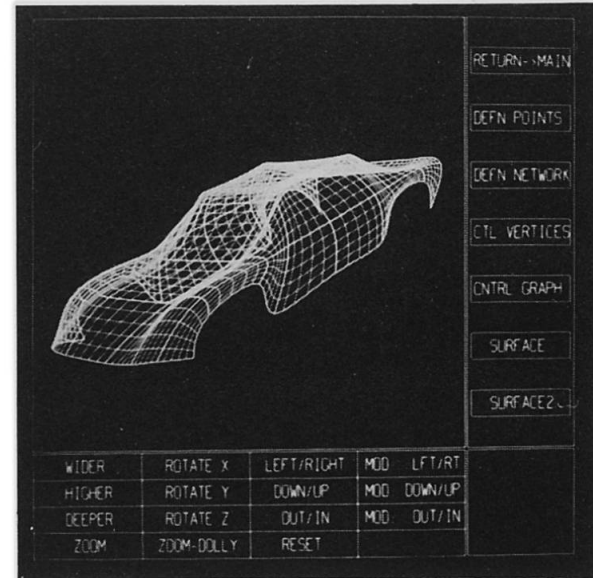


Fig. 5

A bivariate B-spline surface interactively modeled by manipulating the control graph.

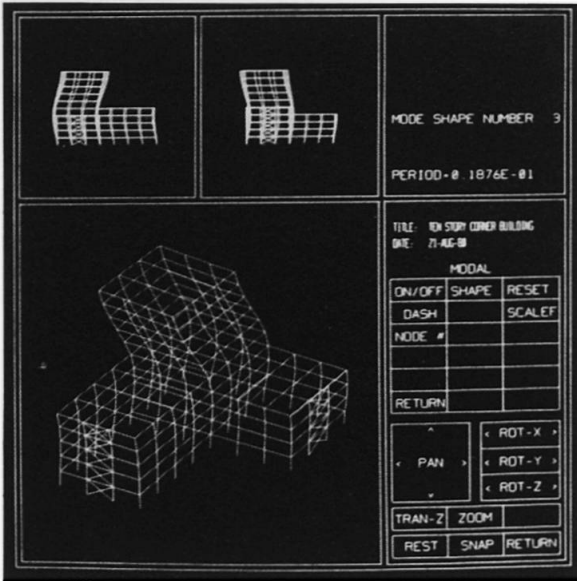


Fig. 6.

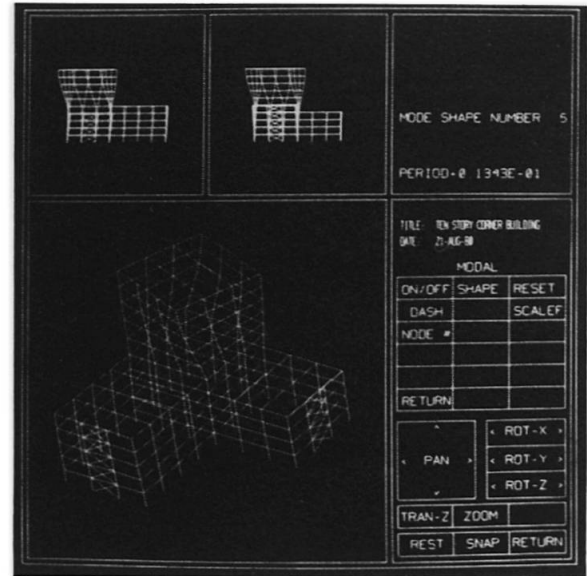


Fig. 7

Interactively modeled ten-story building frame subjected to dynamic loads. Figures 6 & 7 indicate two of the modal shapes displayed on a refresh vector display.

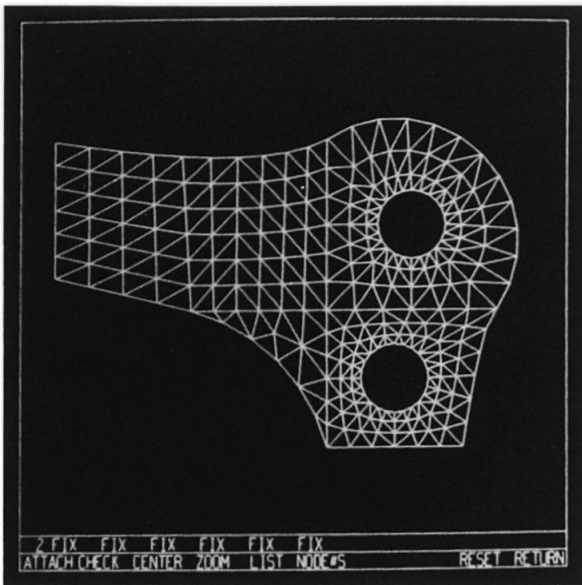


Fig. 8

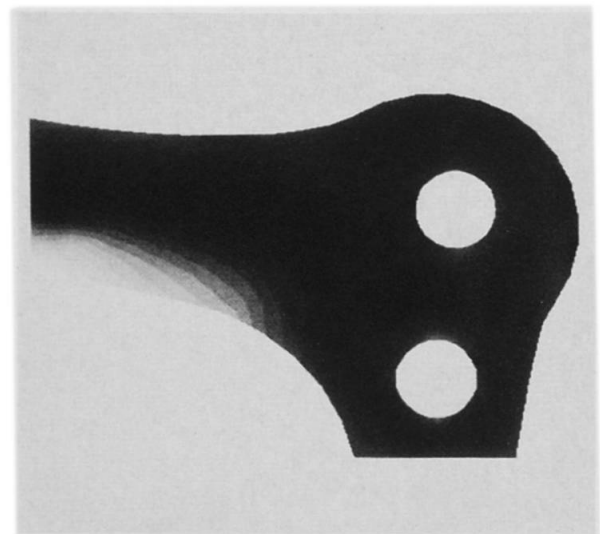


Fig. 9

Display of a completed finite element mesh of a bracket. The object was interactively meshed with a general two dimensional graphic finite element preprocessor utilizing discrete transfinite mapping. The shaded picture is a grey scale rendition of a color raster display of the stress levels using the stress analysis postprocessor.

VII

Mathematical Analysis of Structures: Usefulness and Risks

Modèles mathématiques pour l'analyse des structures: utilité et danger

Mathematische Analyse von Tragwerken: Gebrauchstüchtigkeit und Risiko

M. FANELLI

Professor
ENEL
Milano, Italy

G. GIUSEPPETTI

Ing.
ENEL
Milano, Italy

SUMMARY

The implication of choice of „mathematical models“ of different degrees of complexity is analyzed in an effort to elucidate the risks incurred through inevitable departures from reality. This analysis is necessary in view of the ever-growing use of very complex models, more and more removed from the intuition and sensitivity of structural engineers.

RESUME

Les conséquences du choix de „modèles mathématiques“ au degré croissant de complexité sont étudiées. Le but de cette analyse est de définir les risques introduits par suite de différences pratiquement inévitables entre le modèle et la réalité. Une telle analyse s'avère toujours plus nécessaire car les ingénieurs utilisent des modèles toujours plus compliqués et toujours plus éloignés de l'intuition et de la sensibilité personnelle.

ZUSAMMENFASSUNG

Es werden die Auswirkungen analysiert, die die Auswahl von mathematischen Modellen steigender Komplexität nach sich zieht. Zweck dieser Analyse ist die Untersuchung und die Bestimmung der auftretenden Risiken, bedingt durch die praktisch unvermeidbaren Unterschiede zwischen Modell und Realität. Diese Analyse wird immer nötiger, wenn man die Tatsache berücksichtigt, dass die Ingenieure zunehmend kompliziertere Modelle benutzen, die sich immer mehr von der Intuition und der persönlichen Sensibilität entfernen.



FOREWORD

Electronic computers and computational codes are but the intermediaries linking mathematical models, built by us, of reality, with the numerical results on the basis of which we intend to take practical, technical decisions.

In fact, before setting up a computational numerical code we have to define, with precision, the "mathematical model", chosen as an acceptable representation (simulation) of the real situation in question. Obviously, this model will represent the fundamental variables of this situation, as well as its actions and reactions, both internal and in relation to the outside world.

Moreover, even an efficient computational code based on such a mathematical model will not by itself suffice: it will also be necessary to extract from the real world credible input data, without which the results will have no meaning or value whatever.

These concepts may be expressed in the following block diagram (fig. 1).

It follows that critical analysis of the use of computers and computational codes, however important and necessary, is not by itself sufficient to ensure correct interaction between quantitative analysis of an engineering problem and the operative technical decisions.

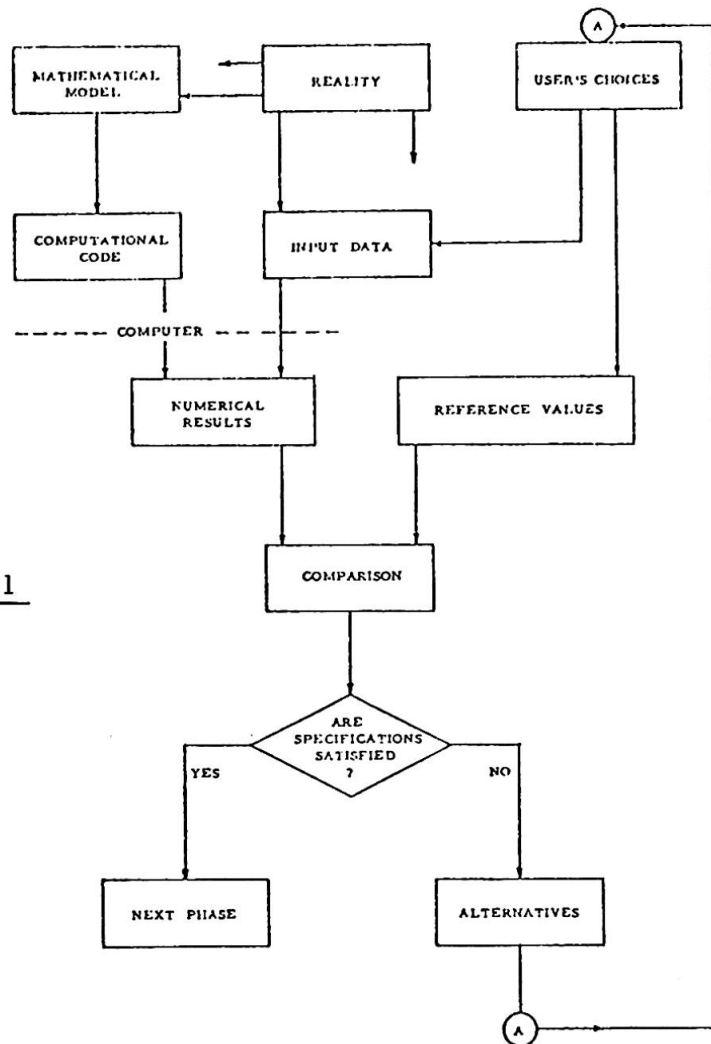


Fig. 1



It will also be necessary to examine critically the often unstated criteria underlying the choice and formulation of the basic mathematical model. The latter, it must be admitted, is not determined in isolation - at least, not for those complex problems that require the use of electronic computers; on the contrary, it is partly the result of subjective choices from among various possible alternatives. And where there is a choice, there is also a probability or risk of error, or, at any rate, of a less than-optimal result.

This paper will develop some considerations that we deem relevant to this type of analysis. These preliminary thoughts obviously cannot aim to exhaust the subject, but will, it is hoped, open up a debate and stimulate awareness of often-neglected, or taken-for-granted implications that are, nonetheless, important for the consequences, whether positive or negative, stemming from them.

1. More and more often, recourse is had to so-called "mathematical forecasting models", and this includes the engineering field.

For greater clarity, we should specify what we mean by this expression. We refer to numerical simulation processes - nearly always implemented by means of computational codes on computers - where-by it is claimed that the evolution (in time and/or in space) of certain physical quantities typical of the system being examined, or checked, can be simulated.

Without excluding others, let us therefore focus our attention on two applications typical of such models:

A) - The design - or rather, the parametric optimization of the design.

B) - On-line check of an installation or process.

In case A), the designer has at his disposal certain degrees of freedom as regards the design - some dimensions, some physical constants, some characteristic times - and, by varying parametrically the corresponding model variables, he can claim to be able to investigate the effects of such variations on the behaviour of the system designed.

In this case, use of the model (by the designer) is interactive - in principle, even though, often, this is not strictly speaking so.

In case B), the model has to be implemented on a mini-or-micro-computer, connected to the physical system by adequate sensors that will feed into the model the correct information on the state of the system and on the settings given by the operators, or by the automatisms, to the control devices. A continuous comparison is then set up on some of the fundamental quantities simulated in the model and monitored in the physical system; and if the difference between forecasts and observations of the value of one or more of these variables exceed a pre-established threshold, then a warning is emitted, or else appropriate, pre-set action is taken.

2. Undoubtedly, the techniques referred to above have met with considerable success in application, and are showing themselves to be indispensable in the design (A) or checking (B) of complex systems, in which the conventional methods or the intuition of the engineer no longer succeed in mastering the whole situation represented by the many possible interactions. But these same successes may create a dangerous tendency to underestimate - or even not to take into consideration - the ever-present uncertainties, the causes of error, and,



lastly, the risks inherent in the two types of model utilization.

That there are indeed risks is shown clearly, among other things, by a qualitative analysis such as that which follows. But such considerations are not usually brought explicitly to light, or else the various parties involved - i. e., the designer and author of the model, for example, when they are two different people - tend to have a more or less complete and inevitably different awareness and evaluation of them. From this fact there follows that the results may be badly or erroneously utilized or interpreted, and it would therefore seem worth-while attempting to clarify, as far as possible, the logical implications of the decisional processes stemming from the results of the model itself.

3. It is obvious, in the first place, that any mathematical model is, of its very nature, imperfect. The operation involved in representing a physical reality, which is always extremely complex and open to every kind of interaction, with a finite series of numerical variables and mathematical relationships between the latter, already involves a considerable degree of abstraction, and therefore inevitable departures from reality.

At best, we shall find that the model is, if we look at it clearly incomplete, because it will ignore variables or "secondary" phenomena in relation to the quantities on which the user wishes to concentrate his attention, but that also interact to some extent with the phenomenon modelled.

Moreover, very often the laws and relationships applied as a basis for the model will only be valid as an initial approximation. To this must be added the fact that the inputs of the model themselves (initial conditions and/or boundary conditions of the system) will be incomplete and affected by inevitable errors.

There may also be errors in implementing the model (programming of the computational codes), or in its use.

The consequence will obviously be that the outputs of the model will also be affected by errors and approximations.

The forecasts of the model, on which the user's decisions are based, may therefore sometimes lead to mistaken, or at any rate less-than-optimal choices. Reducing the matter to its simplest terms, we may say that very often, both in case A) and in case B), what the user wants is to be able to recognize, based on the responses of the model, whether the situation considered on each separate occasion is safe (S) or unsafe (U). In case A), each indication U will be followed by a different choice of free parameters - within the framework bounded by the limitations of the problem - until an S verdict is obtained.

In case B), each U verdict will have to be followed by appropriate operative decisions calculated to restore the system to an S situation.

However, with regard to what we have said above, the verdict issued on the basis of the model's responses will not always correspond to the real situation.

In particular, a real situation U may sometimes correspond to an S verdict, while, on other occasions, the opposite will occur.

If we assume that the model is used repeatedly in homogenous circumstances, in which the inputs are made to vary in all possible ways, these mistaken verdicts will be issued with a certain frequency (or probability), which we may consider to be represented in the following possible "table of decisions" by the user:

real situa tion verdict issued on basis of model	S	U	
S	$(1-\eta)(1-\beta)$	$\alpha\eta$	$(1-\eta)(1-\beta) + \alpha\eta$
U	$(1-\eta)\beta$	$(1-\alpha)\eta$	$(1-\eta)\beta + (1-\alpha)\eta$
	$1-\eta$	η	1

(erroneous decisions in the thick frames)

$\alpha \ll 1$
risk of error S-U

$\beta \ll 1$
risk of error U-S

N. B. In the case of the design, it should be assumed that $0 < \eta = 1 - \epsilon$ with $\epsilon \ll 1$.

In the case of on-line checking, usually $0 < \eta \ll 1$.

4. At this point, it is clear that using a mathematical model involves two distinct types of risk:

- 1 - Type a. risk (probability α): the user considers U to be a S situation (and therefore neglects to take corrective decisions); correspondingly, he will have to bear the costs of any consequences of not having recognized the lack of safety.
- 2 - Type b. risk (probability β): the user considers S to be U situation, and therefore takes corrective action that is not necessary; in this case, he will have to bear the additional cost of the corrective action, and the cost of any harmful consequences of such action, which might, in an extreme case, bring the system, which was initially in S conditions, towards a U condition.

While the cost connected with the correct decisions (SS and UU) can be evaluated and introduced into financial optimization processes, the costs associated with incorrect decisions (SU or US) are not usually explicitly recognized, still less evaluated. The only evaluation that the user makes is generally an intuitive evaluation of the "degree of trust" to be placed in the model used, and therefore of the more or less wide "safety margin" that will have to be taken in relation to the indications given by the model.

In his turn, the author of the model instinctively tries to protect himself by adopting "conservative assumptions" which may lead him to accentuate the probability of a U-S error in order to diminish the probability of an S-U error. But this choice of pessimistic assumptions is made on a qualitative-intuitive basis, and in the case of complex systems may not lead to the desired results, as we shall be pointing out later on.

It is fairly obvious, and is definitely borne out by the results, that this way of proceeding is often, in practice, acceptable; nor, incidentally, would it be easy to minimize the overall risk run by the user, given the fact that the frequencies (or probabilities) listed in the "table of decision" are not known in advance.

But the mere putting of the question in logical, rational terms helps those concerned to sensitize the problem and to evaluate it in as homogenous a way as



possible. Moreover, both the attitude of anyone who, whether consciously or unconsciously, places blind faith in the model, and the opposite attitude, are fairly widespread but both clearly mistaken on the basis of what has so far been acknowledged.

It is therefore worth-while trying to analyze to the full these interactions between the model, reality, and the user, for this may certainly lead to models being used with more awareness and sense of responsibility than hitherto.

5. Let us now look at the advantages and disadvantages that may be the result of a growing degree of sophistication of the model, from the simplest to the most complex, which, notionally, in a sort of "passage to the limit", we might think of as being "as near as one likes" to an unattainable "perfect model" that accords completely with reality.

In particular, we are interested in making a critical appraisal of whether or not it is always justified to rely on models of an intermediate degree of sophistication, as we may consider many of the models at present in existence or being developed, with greater trust than on more schematic models (see fig. 2 on pag. 7).

Generally speaking, the procedure whereby we pass by degrees from the simplest to the most sophisticated model can be presented in simple terms as above. We begin with a very schematic formulation, in which only the fundamental relationships between the more important quantities are taken into consideration; and, among any different alternative formulations there may be, the choice falls, provided it is the same in other respects, either on the simplest from the formal and mathematical point of view, or on the most "conservative" in relation to the desired "safety margin" of the responses supplied in relation to reality: that is, the formulation that is presumed to make $\beta > \alpha$ (see figure and degree of sophistication 0).

Then, if we wish to improve the model - usually because we instinctively judge to be excessive the risk of β introduced by a simple model (and the costs incurred, therefore, likewise to be excessive), either because of too conservative a design or of the introduction of superfluous gadgets and safety devices - we gradually add to the basic mathematical formulation other variables and other "secondary" relationships: that is, relating to effects that are known to be present in the system, but that are regarded as bearers of consequences "of the second order" for the behaviour of the "main" variables. This verdict, too, and the decision as to where we should stop in considering subsequent secondary effects - since it will usually be impossible to consider them all - are taken on a qualitative and intuitive basis, at best with the help of experience of analogous cases. In this way it is possible to commit involuntary errors of strategy, since, while it may be justifiable to ignore all secondary effects when they introduce consequences that, quantitatively, are not too unbalanced and whose sign is such that they mostly cancel each other out - in this case, to introduce just one, or only a few of the secondary effects may produce a model that is further from reality, in its responses, even than the simplest model.

This is shown in simplified form in the figure;

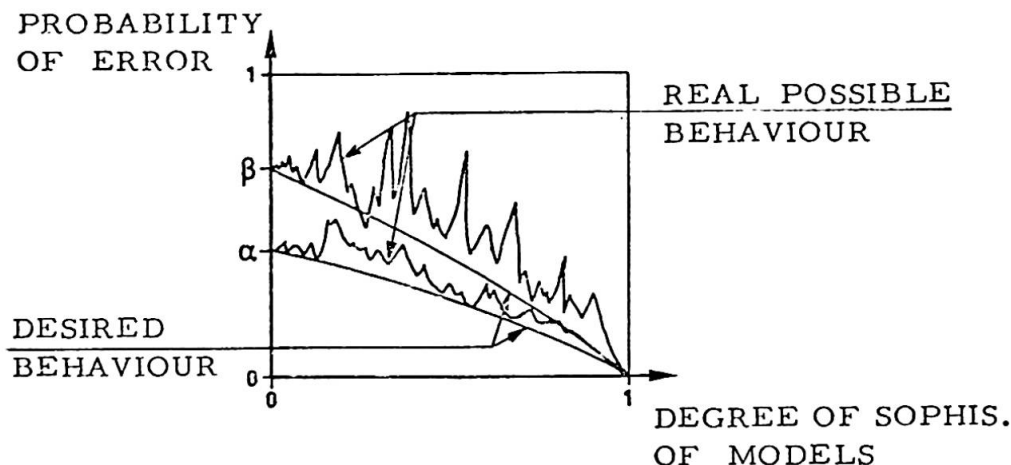


Fig. 2

while it would be desirable for both the probabilities of error α and β to decrease monotonically down to zero, with the indefinite increase in the degree of sophistication of the model, it may happen that one or the other, or both these probabilities of error show a rising (or alternately rising and falling) trend, up or down to intermediate degrees of sophistication, to tend towards zero only with the use of very complete models.

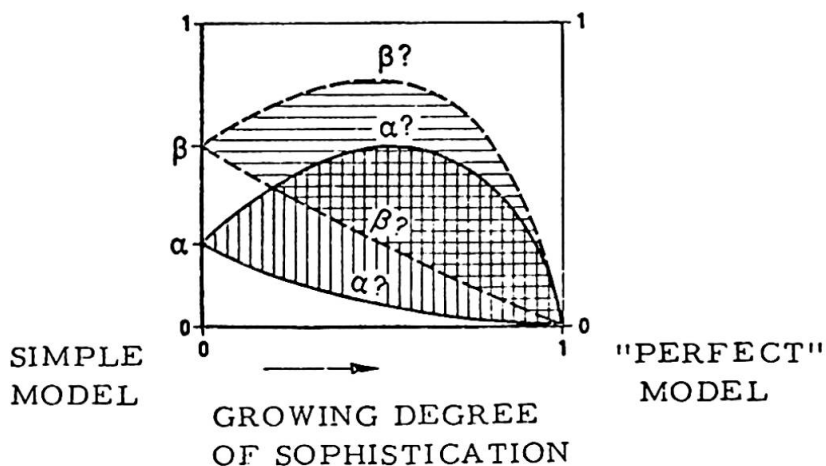


Fig. 3

Now, an increase in probability β of a type b error (U-S) usually leads to additional costs, but only rarely to an actual decrease in safety. Vice-versa, an increase in the probability α of a type a error (S-U) leads directly to a decrease in safety, and therefore to a risk of accidents, possibly serious ones, whose additional costs are usually orders of magnitudes higher than those involved in evaluations that are too cautious.

What has been said is certainly not intended as an a priori criticism of the more sophisticated models, which it is often impossible to do without; it is only desired to show that it is necessary - and, in a sense, urgently so - to find the means to evaluate whether the complications gradually introduced in a model actually fulfil the desired purpose (which, in the terms introduced above, may



be summed up as the wish to diminish the probabilities β and, especially, α) or otherwise, and to decide at what point of sophistication it is advisable to stop.

6. Some devices that can be adopted to test the reliability of the model are suggested by mere common sense. For example, it is important to check that the model is stable, in the sense of making sure that, if we feed into it one, or rather, if successively, several secondary effects that were previously ignored, the responses given will not show variations such as to change the operative decisions based on them. Obviously, an unstable model in the sense defined here is not reliable; but unfortunately it is not certain that a model whose "stability" has been ascertained within a limited range, will be reliable throughout the range - and therefore, ultimately, also for the maximum degree of sophistication, in practice unattainable except as an extrapolation, that accurately coincides with reality.

The same stability requirements must obviously apply with regard to possible variations in the input parameters, which are to be expected in view of the uncertainties and degree of indetermination of our knowledge of the real conditions reflected by the value of these parameters.

At the same time, however, the model must be sufficiently sensitive - that is, it must show sufficient variation of the outputs - and especially of the safety indices - in relation to those variations in input or model structure that are known for certain to lead to significant variations in the degree of safety of the real result. (°)

This implies, among other things, the need not to be content with a single application of the model, which may meet only immediate requirements, but to explore, with the model, several situations close to the one of interest; also, whenever at all possible, it implies the need to vary the structure itself of the model by repeating the application each time with the same input data.

These variations, both in the data and in the structure of the model may usefully be made by a "straddling" approach: by attempting, that is, to see to it that the "real" situation, as far as we may know it or guess it, is sure to be within the range of variations investigated.

Another step dictated by common sense is that of "validating" the model, as far as possible, by applying it to actual situations that have already been produced and documented.

Clearly, this operation involves both the suitability of the mathematical model and the correctness of the computational numerical code. In principle, therefore, the possibility cannot be entirely ruled out that a conceptual error in setting up the model and a programming error will, quite by chance, cancel each other out. This eventuality, however, which is in itself improbable, may be virtually ruled out, if the aforementioned "validation" gives positive results, not just for a single real situation, but for a series of different ones - and, if possible, in

(°) It is superfluous to add that the sensitivity must have the correct sign: that is, when unfavourable variations of the model or the inputs occur, it must show a variation in the safety indices that, in addition to being of significant extent, disclose an increase in the degree of risk, and not vice versa.



relation to different jobs or installations.

In the absence of sufficiently well-documented actual situations, the validation may be thought through other comparisons. For example, use may be made of the results of tests on small-scale physical models, to which the mathematical model in question may reasonably be applied; or even, for want of better, of the results of numerical structure analyses, conducted with completely different methods from the method proposed and, if possible, already independently validated.

Other fairly obvious qualitative criteria can be stated.

For example, it is highly advisable, in the process of subsequent enrichment and progressive sophistication of the mathematical model, to be able to establish and follow a sequence, in descending order of importance, of the various factors successively introduced into the simulation. Indeed, there will obviously be no point in introducing the weaker interactions until the stronger one have been satisfactorily represented. But it may very often be difficult to judge in advance which of the two influences leads to the more significant effects. In such a case, the only course remaining will be to implement both on the model, in any order, and to ascertain the extent of the effects afterwards.

It is likewise evident that, in constructing the model, it will not be enough to trust in intuition or experience in order to decide which factors to introduce into the model at each phase of development; this approach will have to give way to a detailed and exhaustive analysis of the possibilities of interaction, both internal and external, of the system. These possibilities will have to be catalogued, and if possible represented in their interrelationships and their order of importance in a kind of genealogical-tree-like or block structure (see the "fault tree" for analysis of plant reliability).

7. From what has so far been said, it may be concluded that the construction of a reliable mathematical model (in relation to the purpose for which it is required) is an "art" to a great extent associated, in the present state of affairs, with the experience and with the intuition of the individual.

It has also been shown that, for every model that can be conceived, risks of error will be incurred between which it would be necessary to strike an optimal balance that it would be difficult to state.

It should not be ruled out that scientific support for better, more rational model construction might come from specialized studies on operating strategies based on incomplete or insufficiently known data (e.g., the theory of "fuzzy sets and systems" etc.).

Until these studies reach the stage of being more utilizable and become more widely known among technicians, we can advance a few criteria that are certainly useful, but not decisive for checking, on each separate occasion, whether the model it is intended to use is adequate or not.

7.1 For each mathematical model implemented in a computational code, there should be available an exhaustive list of basic assumptions for the model, and an analytical examination should be made of the suitability or otherwise of each assumption for the case in question. Should such suitability be lacking, even only in the case of one assumption, then the applicability of the model is impaired.



7.2 A list should be drawn up of the possible causes of departure in the behaviour of the job or installation under examination, or of its environment, from the aforementioned assumptions, and at least a qualitative degree of probability should be assigned to such departures, as well as the degree of gravity of the consequences. This will be a help in seeing whether one can stop there - that is, whether the work or installation can be examined with the model chosen, assuming, obviously, that the condition stated in 7.1 above has been checked - or whether it is advisable to give consideration to other models, that will include the effects ignored by the first model.

By way of an example, for a structure situated in a non-seismic area, it is physically possible that, at a future date, that structure may be subjected to the effects of an earthquake. If the structure is particularly vital, it will be advisable to make a seismic analysis, despite the low probability of such an occurrence in the area (see structures of nuclear plants).

7.3 For each input variable, it will not be sufficient to assign a "design" or "check" value, but will be necessary to define a variation interval that can reasonably be assumed, and the domain defined by the ranges of variation of the different quantities will have to be appropriately investigated with repeated applications of the model.

7.4 For each model, it will be necessary to document a case-test that is sufficiently significant and will show the reliability of the results, by way of comparison with the real behaviour of installations already constructed, of small-scale models, or at least of theoretical situations sufficiently similar to the case in question, for which the "exact solution" would be known.

7.5 For each important installation, it will be necessary to set up a file documenting the successive phases of the study on that installation, starting with the design and including both the observations made in service (and their interpretation with adequate models) and any subsequent checks made with models other than the one designed.

Appropriate international organizations (including IABSE) might usefully make themselves responsible for standardizing - as far as possible - criteria for drawing up such files and keeping them up to date, as well as promoting the build-up of case history records, access to which will be extremely valuable to the progress of the profession and for the work of model rationalization that is the aim of this memorandum.

VII

Man-Computer Communication in Structural Analysis

Communication homme-ordinateur dans le domaine de l'analyse des structures

Mensch-Computer Kommunikation bei der Berechnung von Tragwerken

DIETER D. PFAFFINGER
FIDES Trust Company
Zurich, Switzerland

SUMMARY

Present interactive facilities already permit close man-computer communication. The amount of communication necessary for a specific problem can be kept to a minimum by the use of data base techniques, highly expressive problem-oriented language, and by strictly leaving all automatable tasks to the computer. As an example a new approach to automated FE-mesh generation is presented. Upcoming means of man-computer communication are discussed.

RESUME

Les possibilités actuelles de communication offrent déjà un dialogue étroit entre l'homme et l'ordinateur. En faisant appel aux banques de données, aux langages de programmation adaptés et expressifs et en exploitant au mieux les possibilités de l'ordinateur pour les tâches automatisables, l'ampleur de la communication qu'exige un problème donné peut être maintenue à un bas niveau. Une nouvelle approche pour la génération automatique d'une subdivision en éléments finis est présentée à l'aide d'un exemple. Les possibilités futures de la communication homme-ordinateur sont mentionnées.

ZUSAMMENFASSUNG

Bereits die heutigen interaktiven Möglichkeiten gestatten enge Kommunikation zwischen Mensch und Computer. Die Menge der für ein bestimmtes Problem benötigten Kommunikation kann klein gehalten werden durch Verwendung von Datenbanken, durch Einsatz ausdrucksstarker problemorientierter Sprachen und durch konsequente Erledigung aller automatisierbarer Aufgaben durch den Rechner. Als Beispiel wird ein neuer Ansatz zur automatisierten Erzeugung von FE-Netzeinteilungen vorgestellt. Zukünftige Möglichkeiten der Kommunikation zwischen Mensch und Computer werden besprochen.



1. INTRODUCTION

In February 1979 in a televised event, international chess champion David Levy played at a chess board in Hamburg against a Cyber 176 computer in Minneapolis. The program on the machine was CHESS 4.8. Levy's moves were automatically transmitted to the computer via satellite. The moves of his artificial opponent were executed on the board by a robot arm under remote control of the computer. The game ended with a draw after 89 moves.

In chess, the communication between the players consists of their moves on the board only. Due to this very limited amount of information, human-like communication with a computer can be simulated rather easily. In structural analysis, however, the man-computer communication is much more complex. In order to use the computer in design and analysis, communication is necessary to describe the overall situation for preliminary studies, to define structures, to select analysis options, to display and evaluate results, to make modifications, to generate reports, to produce construction drawings and so on. Powerful means of communication exist already today to facilitate man-computer interaction in these areas.

2. PRESENT MEANS OF COMMUNICATION

Fig. 1 shows the typical phases of a structural design and analysis task. A high degree of interaction between the engineer and the computer is desirable

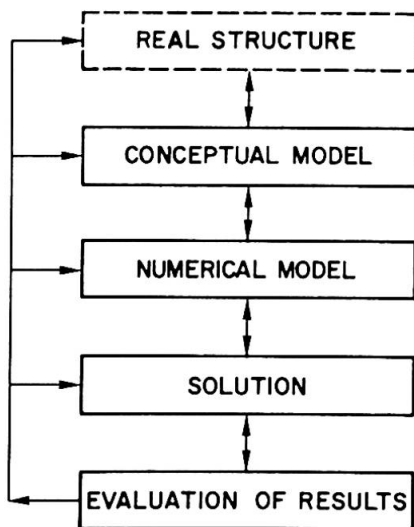


Fig. 1 Phases of design and analysis

when developing the design as well as the conceptual analysis model of the structure, to set up the numerical model and to retrieve, display and eventually postprocess the results of the analysis. On the other hand, the actual solution phase is usually highly automated and low interactivity but high numerical processing capability is required.

Present hardware and software satisfy some of these needs. The interactive processing capabilities of today's minicomputers as well as of some large machines are geared towards a dialogue between man and computer. Interactive graphic systems with digitizers and display equipment allow the design of a structure on the screen and are gradually replacing conventional drawing-boards. With computer networks it is possible to transfer data between interactive decentralized work stations and large machines with high processing capability. Thus interactivity and processing power are available where they are specifically needed.

Technically the present means of communication already permit a high degree of man-computer interaction. However, the amount of communication necessary to accomplish a particular task may be high. It can be reduced by using data base techniques to store information only once and to retrieve it for different purposes, by developing problem-oriented and highly expressive communication languages and by strictly leaving all automatable tasks to the computer. These considerations are illustrated in the following by discussing a current research project [2] for the automated mesh generation in finite element structural analysis.

3. AUTOMATED FE-MESH GENERATION

Conventional mesh generators easily permit the definition of nodes and elements by using interactive graphics. This approach usually requires a large amount of man-computer interaction. The objective of the mentioned research project is the development of a preprocessor program, in which the mesh generation is automated once the geometry of the structure has been defined. The essential role of the engineer is then to influence the generation process rather than to define the mesh in detail. Options exist that permit any degree of influence between fully automated generation and the description of fully predefined meshes.

The geometrical and mechanical properties of the structure are described in a highly expressive problem-oriented language [3]. To define the geometry any scalar and vector operations can be used. Surfaces and bodies are generated by special functions. For example the function

$$\text{FLAECHE (P1,P2,...PN)}$$

describes a plane surface with polygonal boundaries through points P1 to PN. By similar functions and statements parts are transformed and put together. Materials, boundary conditions and loads are defined in the same manner. The language permits the description of parts in parametric form. When calling the part the actual values of the parameters are substituted. Using this feature and taking advantage of all kinds of repetitiveness in the geometry of the structure further reduces the amount of communication. Comprehensive plotting options allow easy verification of the generated data.

Fig. 2 shows schematically the steps required to generate a finite element model. It is seen that the different tasks can be accomplished in an iterative way.

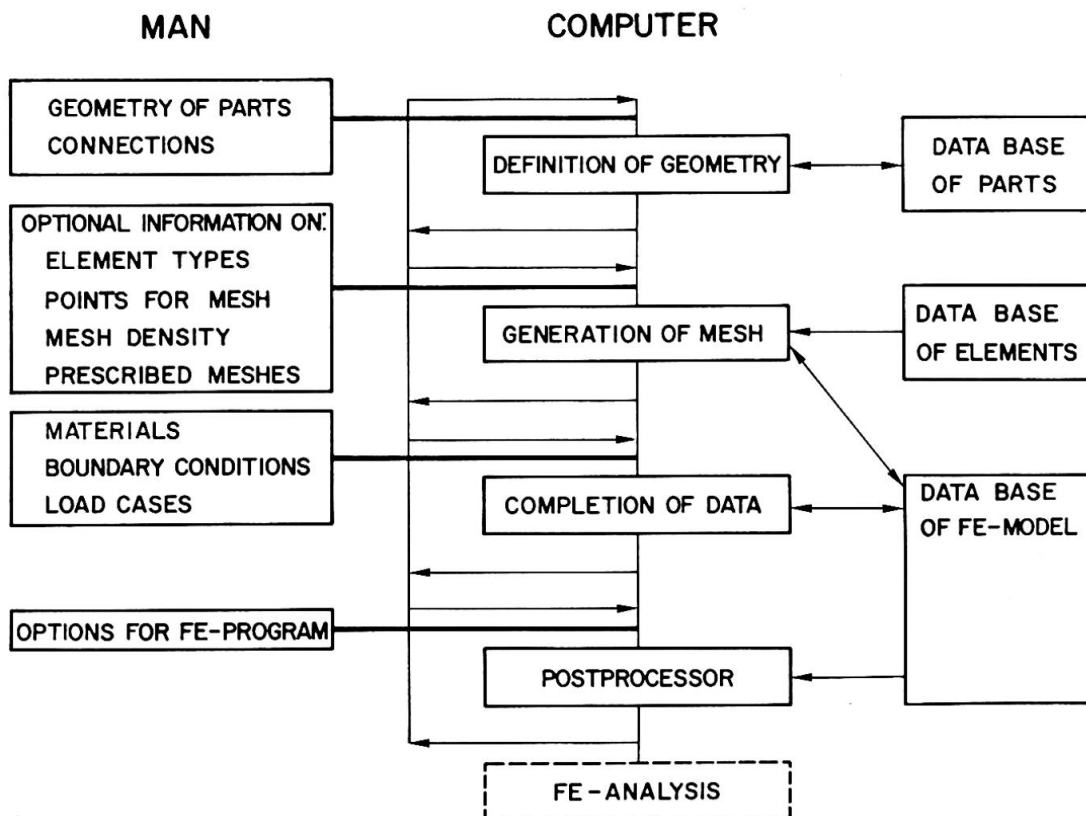


Fig. 2 Automated mesh generation



Modifications can be easily made. The sequence of the steps is flexible. Geometrically defined parts are stored in a data base and are retrieved from there. An element data base contains the properties of the finite elements used for the mesh generation. The numerical data of the FE-model are stored in general form in a third data base. They are then postprocessed into input data of a particular finite element program at the user's option.

Once the finite element analysis is completed, the results have to be displayed and evaluated. The same principles as used for the input generation can be applied to the development of postprocessor programs.

4. EVOLVING NEW TOOLS FOR MAN-COMPUTER COMMUNICATION

The trend for decreasing prices and for increasing power of computer hardware will continue. It can thus be expected, that decentralized interactive systems will be even wider distributed in the future. On the other hand, rapidly developing computer networks will permit the easy exchange of data between decentralized work stations and central large-scale processors. The communication costs are expected to drop due to increasing volume of data communication, by making full use of transmission lines by compression of data and statistical mix, and by using new transmission technologies. Among these high capacity data channels between computers as well as between work stations and computers using fiber optics etc. have to be mentioned. Over long distances packet switching networks based on satellites will offer low cost data transmission facilities.

High capacity mass storage will encourage the development of large central data bases. It is hence expected, that geometrical data bases will be available to design a structure, to generate the analysis model and to produce the construction drawings. Furthermore, the data to control a NC-machine can also be derived from the geometrical data base. Automated digitizers, colour displays together with software for colour graphics, and interactive report generators are some of the upcoming facilities to be used in structural engineering. Modern structural analysis programs permit the solution of rather complex problems. Thus frequently support from an expert is needed. Interactive programs are under development which may in the future guide the engineer to set up the conceptual and the numerical analysis model. In a broader context, computer-based educational systems will be available for many subjects. Much progress has recently been made with synthesized speech. It can be expected, that in the future the means of communication between man and computer for some tasks will be a human language.

5. CONCLUDING REMARKS

The computer revolution has not yet come to an end. Decentralized work stations with network connections to large computers will be widely distributed. New means of communication will permit easy man-computer interaction. The effective use of these modern tools, however, will require new skills and attitudes of the engineer. In structural engineering, the computer can significantly contribute to the design of better structures. In almost all domains the increasing penetration with computerized facilities will in general lead to more and better products and services at decreasing working hours. The task thus remains not only to use the computing facilities in a knowledgeable and prudent way but also to master the sociological changes caused by them.



REFERENCES

1. Bolc, L. Editor, Speech Communication with Computers. Carl Hanser Verlag, Munich and Vienna, 1978.
2. Engeli, M. and Pfaffinger, D., Automatisierte Generierung der FE-Einteilung von Getriebegehäusen. Informationstagung des wissenschaftlichen Beirats der Forschungsvereinigung für Antriebstechnik. Böblingen, 1979.
3. Engeli, M. and Hrdliczka, V., EUKLID-Eine Einführung. FIDES Trust Company. Zurich, 1974.
4. Interactive Techniques in Computer Aided Design. Proc. Int. Conference, Bologna, Sept. 21-23, 1978. IEEE, New York, 1978.
5. Pfaffinger, D., Effective Use of Structural Computer Programs. IABSE Colloquium on Interface between Computing and Design in Structural Engineering. Bergamo, 1978.
6. Wilson, E.L., The Use of Minicomputers in Structural Analysis. Proc. ADINA Conference, Massachusetts Institute of Technology. Cambridge, Mass., 1979.
7. Zweites Deutsches Fernsehen, Programmdirektion, Comments on the game Levy-CHES 4.8/CYBER 176. Mainz, 1979.

Leere Seite
Blank page
Page vide



VII

Ein Programmier-System anstelle von Programmpaketen?

A Programming System instead of Programme Packages?

Un système de programmation au lieu d'une agglomération de programmes?

BERND HARTMANN

Prof. Dr. -Ing.

FH Bochum

Bochum, BRD

ROLF BEYER

Dipl. -Ing.

Beyer-Hartmann-Ingenieursozietät

Erkrath, BRD

ZUSAMMENFASSUNG

Das Konzept eines Programmier-Systems wird nahegebracht, das bei Tragwerkanalysen eine optimale Anpassung erlaubt und schon auf kleinen Rechnern eingesetzt werden kann. Es kann im Gebrauch gegenwärtige Programmpakete ablösen, die schwerfällig und an grössere Rechner gebunden sind. Im Hinblick auf eine breite, praktische Anwendung wird das Programmier-System in PASCAL kodiert und auf Mikroprozessor-Systemen installiert. Anwendungsbeispiele des Programmier-Systems MESY weisen die grosse Leistungsfähigkeit des Konzepts nach.

SUMMARY

The concept of a programming-system is suggested allowing an optimal flexibility in structural analysis even with small computers. Thus it may replace present programme-packages. For practical purposes the programming-system will be coded in PASCAL and implemented on microprocessor systems. Some applications of the programming-system MESY prove the great capability of the concept.

RESUME

L'article présente la conception d'un système de programmation qui peut être utilisé de façon optimale pour l'analyse des constructions avec des ordinateurs moyens. Ce système peut remplacer les programmes actuels compliqués nécessitant un grand ordinateur. Le langage de programmation PASCAL est utilisé dans ce système pour assurer une installation facile sur les microprocesseurs. La capacité remarquable de cette conception a été démontrée par le système de programmation MESY.

1. DAS KONZEPT PROGRAMMIER-SYSTEM

1.1 Programmpakete

Heute werden zahlreiche Programmsysteme angeboten, die es gestatten, die gängigen Baukonstruktionen mit dem Rechner beliebig genau zu untersuchen. In der Regel erwirbt man die Programmsysteme als geschlossene Software-Pakete. Die Benutzungsanleitung, die Aufbereitung der Eingabedaten sowie die Interpretation der Ausgabe sind in einem beigelegten Manual erläutert. Eine Benutzung als "black box" wird so auch für den Nichtfachmann möglich.

Programmsysteme haben aber Nachteile, von denen zwei besonders ins Auge fallen. Zum ersten sind sie starr auf bestimmte, klar abgegrenzte Anwendungsgebiete ausgerichtet, was sich schon in der Aufbereitung der notwendigen Eingabedaten äußert, die mehr langweiligen Zahlenchiffren gleichen als der Beschreibung eines Tragwerks. Doch häufiger als erwünscht tauchen Probleme auf, die nicht direkt in die Domäne des Programmpakets fallen, sich aber nur mit dem Rechner lösen lassen. Da sich das parallele Betreiben von mehreren Programmsystemen aus ökonomischen Gründen in der Regel verbietet, wird man versuchen, das Programm durch einen Eingriff in das Software-Paket der aktuellen Fragestellung anzupassen. Diese Änderung können aber nur Ingenieure vornehmen, die die Interna des Programmpaketes detailliert kennen. Da es solche nur in begrenzter Anzahl gibt, kommt dieser Eingriff, wenn überhaupt, teuer zu stehen und kostet viel Zeit, was oft noch viel unangenehmer ist. So ist es nicht verwunderlich, daß "black box"-Systeme wegen ihrer fehlenden Anpassungsfähigkeit heute nicht mehr als der Weisheit letzter Schluß angesehen werden.

Der zweite Nachteil von Programmsystemen ist vielleicht noch gravierender. Sie sollen nämlich einen möglichst großen Anwendungsbereich abdecken und werden deshalb sehr umfangreich. Und da sie stets als Gesamtsysteme gewartet und gepflegt werden, benötigen sie einen größeren Rechner. Veröffentlichungen, wie z. B. /1/, die sich mit der Implementation von Programmsystemen auf kleinen Rechnern befassen, wählen als kleinstmögliche Anlage immer noch eine Mini-computer-Konfiguration, die wenigstens eine halbe Million DM kostet. Aus diesem Grund ist der direkte Einsatz von Programmsystemen auch nur an Rechenzentren möglich, wie sie Hochschulen und größere Firmen installiert haben. Kleinere und mittlere Ingenieurbüros, die sich eine solche Anlage nicht leisten können, aber den Mammutanteil bei der rechnerischen Untersuchung von Tragwerken leisten, bleiben so von der Benutzung der Programmsysteme in der Regel ausgeschlossen. Größere Büros, die über einen externen Anschluß an Rechenzentren verfügen, stehen unter dem Druck, gleich beim ersten Mal die numerische Aufbereitung ihrer mechanischen Systeme fehlerfrei vorzunehmen und sich exakt an die Formatierung der Eingabedaten zu halten, um so teure Fehlrechnungen zu vermeiden, die die Rentabilität von Aufträgen gefährden. Also auch unter diesem Gesichtspunkt der – man möchte sagen – Exklusivität der Benutzung von Programmsystemen ist es sinnvoll, sich nach geeigneteren Möglichkeiten umzuschauen.

1.2 Die Alternative

Ein alternatives Konzept zu den Programmsystemen bietet das Programmier-System. Bekanntlich können alle denkbaren Untersuchungen an mit finiten Elementen diskretisierten Systemen besonders elegant mit den Methoden der analytischen Mechanik verfolgt werden, siehe z. B. /2/ oder /3/. Infolge dieser Analyse lassen sich leicht autonome Teilprozeduren gewinnen und programmieren. Die so entstehenden Routinen gehen als Moduln in das Programmier-System ein, wie Bausteine in einen Baukasten, und bilden sein Gerippe. Zwar lassen sich für Standardprobleme diese Bausteine auch so kombinieren, daß sie als "black box" in der Industrie von Nichtfachleuten eingesetzt werden können. Zur vollen Entfal-

tung kommen die Möglichkeiten eines Programmier-Systems jedoch erst, wenn ein Ingenieur mit Vorbildung in Mechanik und ausreichenden Programmierkenntnissen die Module in einem jeweils sehr kurzen Individualprogramm so verknüpft, daß sie dem anstehenden Problem optimal angepaßt sind. Da ständig neue Kinematik- und Elementprozeduren eingefügt werden können, kann ein Programmier-System allen denkbaren Problemkonstellationen gerecht werden.

Aus dem Gesagten ergibt sich noch ein weiterer, wesentlicher Vorteil eines Programmier-Systems. Durch seine hohe Auflösung in autonome Teiloperationen brauchen nämlich in dem jeweiligen Rechner bei der Untersuchung eines Tragwerks aktuell nur die Routinen verfügbar sein, die für die Problemlösung benötigt werden. Ein Programmier-System kann so schon auf sehr viel kleineren Rechnern implementiert werden als gängige Programmpakete.

2. ZUR PRAKTISCHEN UMSETZUNG

2.1 Vorbedingungen

Um der Idee eines modularen Programmier-Systems zum wünschenswerten Durchbruch in der Praxis zu verhelfen, mehren sich in letzter Zeit die Chancen. Die wesentlichste Vorbedingung hierzu beginnt sich langsam zu erfüllen. Denn es kommen jetzt vermehrt Studenten auf die Hochschulen, die von ihrer Schulbildung her bereits mit der Mentalität des numerischen Aufbereitens auch der Programmierung vertraut sind. Existierende Programme sind für sie nicht anonyme Vorgaben, die man unter Eingabe richtiger Daten abspulen muß. Sie scheuen sich nicht, auch korrigierend einzugreifen. Sie nehmen das Angebot, selbständig Programme zu erstellen, gern an. Mit einer entsprechenden Ausbildung in Mechanik sind sie schnell in der Lage, in der Baupraxis auf dem Gebiet der rechnerischen Untersuchung von Tragwerken neue, kreative Möglichkeiten zu erschließen.

Als weitere wichtige Voraussetzung für eine weite Anwendung eines Programmier-Systems müssen in der Praxis genügend geeignete Rechner zur Verfügung stehen. In ausreichender Anzahl werden aber nur solche verfügbar sein, die auch für kleine Ingenieurbüros erschwinglich sind. Vom Preis her kann man da nur an solche Rechner denken, die sich im Zeitraum von ein oder zwei Jahren für diese Büros bei der Abwicklung von Aufträgen rentieren. Schließlich ist als letzte wichtige Vorbedingung für den Aufbau eines Programmier-Systems die Verfügbarkeit einer geeigneten Programmiersprache zu nennen. Sie muß modular konzipiert sein und das Erstellen von selbständigen Unterprogrammen unterstützen.

2.2 Das Mikroprozessorsystem

Vor allem die Notwendigkeit eines relativ geringen Preises legt die Wahl einer Mikroprozessor-Konfiguration als den vorteilhaftesten Rechnertyp für die Implementation eines praxisnahen Programmier-Systems nahe. Die für das Programmier-System charakteristische hohe Auflösung des Berechnungsgangs in viele autonome Teiloperationen kommt dieser Wahl sehr entgegen.

Sind aber Mikroprozessorsysteme für ein solches Vorhaben überhaupt geeignet? Die Antwort hierauf ist: Zur Zeit beschränkt, in Zukunft ja. Fast alle gängigen Mikroprozessoren basieren auf Z80- oder 6502 Zentraleinheiten oder aus diesen abgeleiteten Bauvarianten. Diese 8-bit-Prozessoren leiden im Hinblick auf die Implementation eines Programmier-Systems an dem Manko einer sehr kurzen Befehlswort-Länge. Hierdurch werden die mittleren Zeiten zur Ausführung von Befehlen trotz Taktfrequenzen bis hin zu 4 MHz noch relativ lang, die technisch vernünftig realisierbare Wortlänge zur Darstellung von Zahlen kann kaum größer

als 4 byte gewählt werden und die direkt adressierbaren Speicherstellen bleiben auf maximal 64 k byte beschränkt. Auf Grund der langsamen Rechengeschwindigkeit, des knappen Arbeitsspeichers und der begrenzten Rechengenauigkeit erscheinen Mikroprozessoren daher im Augenblick nur für die numerische Untersuchung einfacher Tragwerke geeignet. Auf diesem Gebiet leisten sie jedoch schon Beachtenswertes. So wird ein von den Autoren vor kurzem in BASIC erstelltes Programm zur Untersuchung von allgemeinen, ebenen Stabwerken schon Tag für Tag auf Apple- und Commodore-Mikrorechnern eingesetzt.

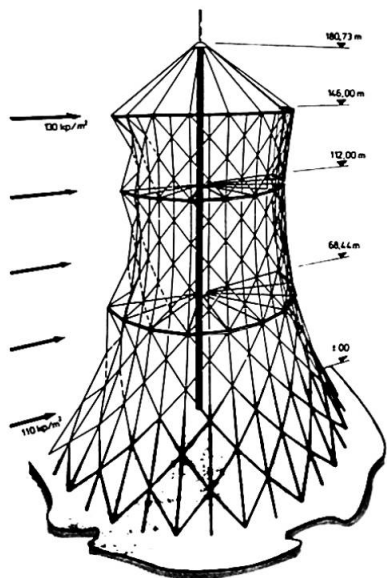


Bild 1:
Seilnetzkühlturm unter Windlast

2.3 Die Sprache PASCAL

Die Entwicklung der Mikroprozessortechnik seit etwa 1975 ist erfreulicherweise ohne den Einfluß der etablierten Firmen geblieben, die elektronische Rechenanlagen entwickeln und anbieten. Diese Firmen unterstützen nämlich heute noch Sprachen, die, wie FORTRAN, direkt zu Beginn der Computeraera vor ca. 30 Jahren konzipiert wurden. Diese, man möchte sagen, fossilen Sprachen werden sicherlich deshalb gepflegt, um das gesamte im Verlauf der Zeit entstandene Software-Gut möglichst kompatibel zu erhalten. Dieser legitime Gesichtspunkt ist allerdings ein großes Hindernis auf dem Wege zu einer bequemen Benutzung der Rechenanlagen. Und diesem Grunde ist es wahrscheinlich auch zuzuschreiben, daß wesentlich benutzerfreundlichere Sprachkonzepte, wie das von ALGOL-60, den weltweiten Durchbruch eigentlich nie geschafft haben.

Die Hersteller von Mikroprozessorsystemen hatten sich bei der Entwicklung ihrer Rechner nicht um bestehende Software-Pakete zu kümmern. Als ihre Technologie vor gut zwei Jahren so weit gereift war, kleinere programmierbare Anlagen auf dem Markt anzubieten, hatten sie die freie Wahl der Programmiersprachen. Anfänglich implementierten alle BASIC als erste, fundamentale Sprache. In letzter Zeit stand die Wahl der Sprache

Die Entwicklung der Mikroprozessortechnik schreitet aber rasant fort. So werden schon seit längerem 16-bit CPU's, z. B. bei Texas Instruments, in Mikroprozessorsystemen verwendet, von denen ein für das Programmier-System vielleicht schon heute geeignetes Modell, das von Alpha Micro, auf dem deutschen Markt aus unbekanntem Gründen leider nicht richtig vertrieben wird. Bei einer weiteren Firma, nämlich Cromemco, ist im Gespräch, daß noch im Verlaufe dieses Jahres mit dem Einsatz einer Konfiguration mit einer 32-bit-CPU zu rechnen sei. Aber auch ohne die weitere Entwicklung vorhersagen zu wollen, kann mit Sicherheit erwartet werden, daß für die Installation von Programmier-Systemen voll geeignete Mikroprozessoranlagen in weniger als zwei Jahren zur Verfügung stehen. Dieser Rechnertyp ist daher unbedingt als das Gerät anzusprechen, das für die Verbreitung und Anwendung eines allgemeinen Programmier-Systems am besten geeignet ist.

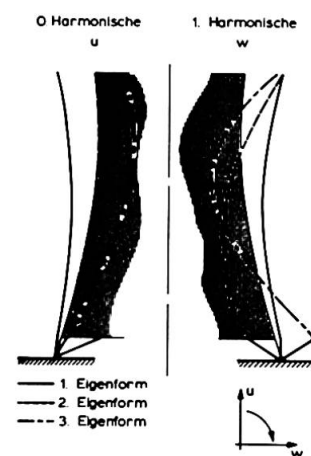


Bild 2:
Stahlbetonkühlturm,
unterste Eigenform

an, die auf den Mikroprozessorsystemen auf höherer, benutzerfreundlicherer Ebene zum Einsatz kommen sollte. Nachzu alle Hersteller – das Angebot an Compilern und Interpretern weist das aus – haben sich für PASCAL entschieden, und zwar für die Version, wie sie von der University of California in San Diego vertrieben wird.

Die Programmiersprache PASCAL, die etwa 1965 konzipiert wurde, verdient diese Wahl. Sie ist ähnlich wie ALGOL-60 in hierarchische Blöcke strukturiert, also sehr modular aufgebaut, und bedient sich des wohl modernsten Datenkonzepts. Sie ist für die erstrebenswerte Anwendung der Unterprogrammiertechnik ausgelegt, wodurch von der erstellten Software schon kleinste Abschnitte kompatibel werden. Durch diese Eigenschaft ist PASCAL in gleicher Weise für die Anwendung auf Mikroprozessoren geeignet, wie auch für die Realisation des Konzeptes eines modernen Programmier-Systems. Diese Sprache wird sicher bald eine weltweite, praktische Anwendung erfahren.

3. ZUR REALISATION

3.1 MESY

Eine Realisation des Konzeptes Programmier-System ist MESY /4/, das in ALGOL-60 kodiert ist und vor allem am Telefunkenrechner TR 440 der Ruhr-Universität Bochum gepflegt wird. Es erfüllt schon weitgehend die Forderungen, die an ein modulares nahezu allen Problemen gerecht werdendes Programmier-System zu stellen sind. Einer größeren Verbreitung steht leider die Tatsache im Wege, daß es in ALGOL-60 programmiert ist. Denn für diese Sprache stehen nur in wenigen Rechenzentren sauber und ökonomisch arbeitende Compiler zur Verfügung. Ein weiterer Grund für die mangelnde Verbreitung liegt aber sicher auch darin, daß es bis heute für einen Großteil der Anwender unvorstellbar ist, sich selbst optimale Programme durch geschicktes Kopieren fertiger Programmbausteine auf den Leib zu schneiden.

Mit dem Programmier-System MESY sind schon zahlreiche, recht unterschiedliche Tragwerke statisch+dynamisch untersucht worden. Der modale Aufbau verleiht MESY hierfür eine hohe Anpassungsfähigkeit, die es gestattet, die verschiedensten Untersuchungen mit einer nahezu gleichartigen Formulierung des sogenannten Individualprogramms vorzunehmen.

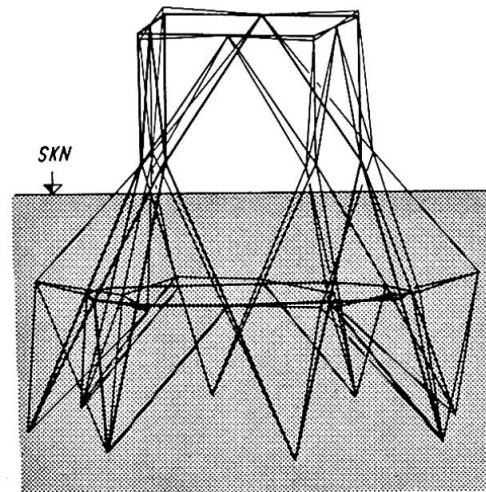


Bild 3:
Forschungsplattform Nordsee
5. Eigenform, $f = 2,85$ Hz

Das gleiche Programm mit einzelnen ergänzenden oder weggelassenen Anweisungen dient der Lösung so unterschiedlicher Fragestellungen wie dem Aufsuchen von Gleichgewichtslagen mit beliebig großen Verschiebungen bis hin zur Traglast, dem Ermitteln von Eigenlösungen bei Stabilitäts- und Schwingungsproblemen oder dem Verfolgen von Schwingungen infolge stationär harmonischer

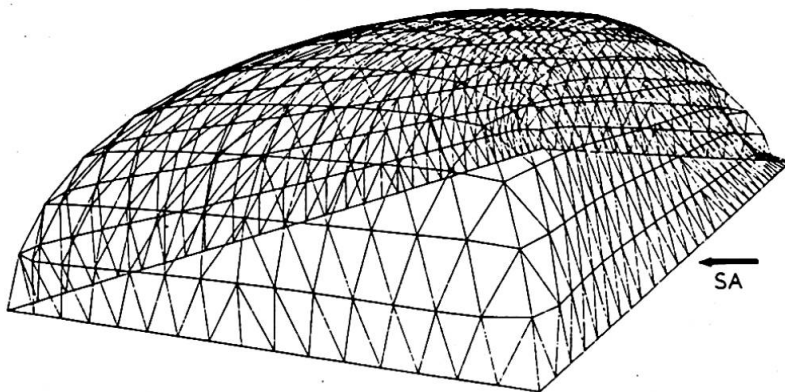


Bild 4: Traglufthalle unter Windlast



oder auch nicht periodischer Erregung.

Stellvertretend für die vielen Tragwerksuntersuchungen seien hier einige angeführt. Bild 1 zeigt in 10-facher Vergrößerung die Deformation eines Seilnetz-kühlers unter Windbelastung, während Bild 2 die untersten Eigenformen einer longitudinal und transversal schwingenden Kühlturmschale mit Unterstützungs-fachwerk darstellt. Bild 3 gibt die 5. Eigenform der Forschungsplattform in der Nordsee wieder, deren unter Wasser liegende Stäbe mit der 5-fachen Masse beaufschlagt sind, um das mitschwingende Wasser zu simulieren. Bild 4 zeigt die Verformung einer Traglufthalle, die bei schräger Windausströmung eine größere Falte wirft. Bild 5 schließlich deckt auf, auf welche Weise ein drei-stöckiger Rahmen nach etwa 2,4 Stunden Brandbelastung versagt.

3.2 Aktueller Stand

Zur Zeit sind die Autoren damit beschäftigt, ihr Konzept auf einem Apple-Microcomputersystem in PASCAL zu verwirklichen. Das entstehende Programmier-System ist äußerlich MESY sehr ähnlich, was wegen der großen Verwandtschaft von ALGOL-60 und PASCAL nicht verwundern kann. Die beschränkte Kapazität dieses Mikrorechners erlaubt vorerst nur die Untersuchung von Stabwerken, dieses aber schon nicht-linear.

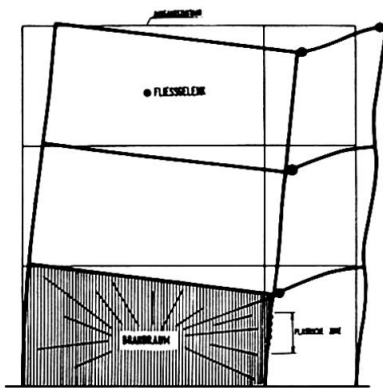


Bild 5:
Stockwerkrahmen, Versagensmechanismus
unter Brandbelastung

SCHRIFTTUM

- /1/ Schrem, E.: ASKA auf Minicomputern. In: Finite Elemente in der Baupraxis. Herausgeber Pahl, J.P., Stein, E. und Wunderlich, W., Verlag von Wilhelm Ernst & Sohn, Berlin, München, Düsseldorf 1978.
- /2/ Schrader, K.-H.: Die Deformationsmethode als Grundlage einer problem-orientierten Sprache. Bibliographisches Institut, Hochschulschrift 830, Mannheim, Zürich 1969.
- /3/ Hartmann, B.: Ein Koordinateninvariantes Finite-Element-Verfahren. Technische Wissenschaftliche Mitteilung 75-1, Institut für Konstruktiven Ingenieurbau, Ruhr-Universität Bochum 1975.
- /4/ Hartmann, B., Schrader, K.-H. und Winkel, G.: MESY - Ein Programmsystem zur Untersuchung von Tragwerken. Konstruktiver Ingenieurbau, Berichte, Heft 22, Vulkan Verlag, Essen 1975.

VII

Datenorganisation für elektronische Entwurfsberechnungen im konstruktiven Ingenieurbau

Data Organization for Electronic Design Computations in Structural Engineering

Organisation de données pour le projet de constructions à l'aide de l'ordinateur

HEINRICH WERNER

Professor, Dr. -Ing.

Technische Universität

München, Bundesrepublik Deutschland

ZUSAMMENFASSUNG

Entwurfsunterstützende Programmketten verbinden Teilaufgaben des Entwurfsprozesses. Der Datenfluss zwischen den Gliedern erfolgt über eine Datenbasis. Es wird die Struktur einer Datenbasis für den konstruktiven Ingenieurbau vorgestellt. Zwei Beispiele zeigen den Einfluss des Baufortschrittes auf den Ablauf der Entwurfsberechnungen.

SUMMARY

Design-supporting program chains integrate several tasks of the design process. Data flow between members runs through a database. The structure of a database for structural engineering is presented. Two examples show the influence of the construction progress on the design computations.

RESUME

Les chaînes de programmes de calcul recouvrent plusieurs activités particulières du projet. L'échange de données entre les membres est assuré par une base de données. La structure d'une base de données pour les constructions de génie civil est exposée. Deux exemples montrent l'influence des progrès génie civil sur le déroulement des calculs.



1. EINFÜHRUNG

Den Forderungen des rechnerunterstützten Entwerfens nach

- Steuerung der Programm- und Datenfolge durch den Entwurfsbearbeiter
- Verfügbarkeit und Veränderbarkeit der Daten während des gesamten Entwurfszeitraumes
- Erweiterbarkeit des Programmsystems auf neue Vorschriften, Verfahren und Anwendungen

werden am besten Programmsysteme mit Kettenstruktur [1] gerecht:

Aus einer Programmbibliothek werden vom Entwurfsbearbeiter autonome Module ausgewählt. Dateneingabe, -speicherung und -transport erfolgen über eine externe Datenbasis.

2. DIE PROGRAMMKETTE SET FÜR ENTWURFSBERECHNUNGEN IM KONSTRUKTIVEN INGENIEURBAU

2.1 Aufbau der Kette

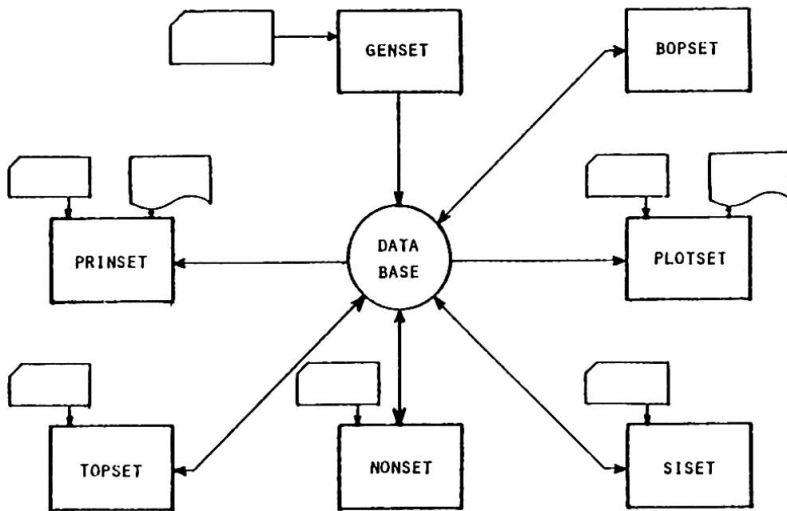


Bild 1: Programmkette SET

Entwurfsberechnungen im konstruktiven ingenieurbaue haben zu berücksichtigen, daß der Ablauf des Bauvorganges oft einen wesentlichen Einfluß auf die Beanspruchungen und das Tragvermögen der Bauwerke hat. Bauzustände sind häufig kritischer als der Endzustand.

Die Programmkette SET geht auf die genannten Erfordernisse ein. Aus einem durch GENSET erzeugten Katalog von Tragwerkselementen (Querschnitte, Tragglieder, Lasten, Material) werden

gemäß dem Baufortschritt sich ändernde statische Systeme gebildet und in NONSET berechnet. Verformungen und Beanspruchungen werden in der Datenbasis als "erworbene Eigenschaften" festgehalten, um ihren Einfluß auf den weiteren Baufortschritt berücksichtigen zu können.

Die SET Programme haben im einzelnen folgende Aufgaben [2]:

- Generierung von Tragwerkselementen (GENSET),
- Ausdruck von Daten in definierbaren Tabellen (PRINSET),
- Graphische Darstellungen (PLOTSET),
- lineare und nichtlineare Statik an Rahmen-, Flächen- und Kontinuumstragwerken (NONSET),
- Statik nach der Theorie II. Ordnung und Bemessung an räumlichen Stahlbeton-, Spannbeton- und Stahlrahmen (TOPSET),
- ebene, rotationssymmetrische und räumliche Sicker- und Grundwasserprobleme (Siset)
- Bandweitenoptimierung bei Systemmatrizen (BOPSET).



Der Datenverkehr zwischen den Programmen und der Datenbasis wird durch eine in FORTRAN geschriebene, dynamische Verwaltung von Kern- und Massenspeicher (DYNCO) unterstützt.

2.2 Die Datenbasis

Eine Datenbasis für Entwurfsberechnungen hat folgende Anforderungen zu erfüllen:

- Einsetzbarkeit auf unterschiedlichen Maschinenkonfigurationen. Im konstruktiven Ingenieurbau findet man das gesamte Hardwarespektrum vom Großrechner bis zum Tischrechner vor.
- Transparenz der Datenstruktur. Mit einfachen Datenstrukturen sollten für Anwendungsprogrammierer die Aufwendungen für Datenspeicherung und -zugriff zu einem Minimum werden.
- Anschlußmöglichkeit für fremde Programme. Die Gewährleistung der Datenaustausches zwischen Programmen unterschiedlicher Autoren gehört zu den wichtigsten künftigen Aufgaben der Softwareentwicklung.

Keine der bekannten Datenbanksysteme erfüllt die genannten Forderungen ausreichend. Deshalb wurde das folgende Datenbasiskonzept ausgewählt und in der SET-Kette zum Einsatz gebracht |3|:

- Die Datenbasis besteht aus einer einzigen sequentiellen Datei pro Bauwerk oder Bauteil.
- Sie kann in einfacher Weise mit Standard-FORTRAN bearbeitet werden; man benötigt keine spezielle Datenbanksoftware.
- Alle Sätze haben die gleiche Länge. Keine Information ist auf die Stellung des Satzes in der Datei bezogen.
- Jeder Satz wird mit einer Kennung eingeleitet.

Neben der Erfüllung der oben genannten Forderungen bietet das beschriebene Konzept noch weitere Vorteile:

- Da keine Information von der Reihenfolge der Sätze abhängig ist, können diese leicht nach Begriffen, die den Daten entnommen sind, sortiert werden.
- Bezugsdaten werden explizit gespeichert (die Elementinformationen enthalten z. B. eine Liste der zugehörigen Knoten).
- Die Art von Zugriff und Sortierung ist dem Programmierer überlassen. Damit bleibt er unabhängig von vordefinierten Datenstrukturen.

3. ANWENDUNGSFÄLLE

3.1 Vorentwurf für einen Tunnel

Zum Vorentwurf eines doppelstöckigen U-Bahnhoftunnels waren statische Untersuchungen anzustellen |4|.

Die lichte Weite der beiden übereinander liegenden Röhren betrug 10,00 m, die lichten Höhen oben 6,30 m und unten 7,00 m. Die Tunnel sollten mit einer 30 cm starken Spritzbeton-Außenschale verbaut und mit einer 50 cm starken Sperrbeton-Innenschale ausgekleidet werden. Das 19,5 m breite und 32,95 m hohe, auf Federn gelagerte Finite-Element-Modell (Bild 2) enthielt elastoplastische und gerissene Boden-Scheibenelemente sowie biegesteife Stabelemente für die Tunnelwände. Unter der Wirkung der gestörten Primärlasten waren folgende Bauzustände zu simulieren:

1. Ausbruch der unteren Tunnelröhre und gleichzeitiger Verbau durch die Spritzbetonschale.
2. Auskleidung der verformten, unteren Röhre mit Sperrbeton und Bodenauflockerung im oberen Bereich (= reduzierte Bodensteifigkeit). Das Zusammenwirken zwischen Außen- und Innenschale konnte durch elastische Kopplungen erfaßt werden, die bei Zug ausfallen und Schubkräfte durch Reibung übertragen (Bild 3).
3. Ausbruch der oberen Tunnelröhre und Anbringen der Spritzbetonschale (Bild 4).
4. Auskleidung der oberen Röhre mit Sperrbeton und Belastung des Tunnels durch den Druck des wieder ansteigenden Grundwassers.

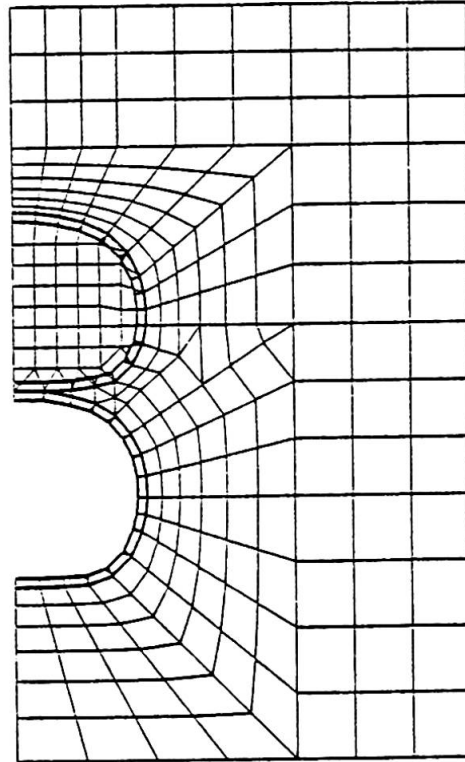
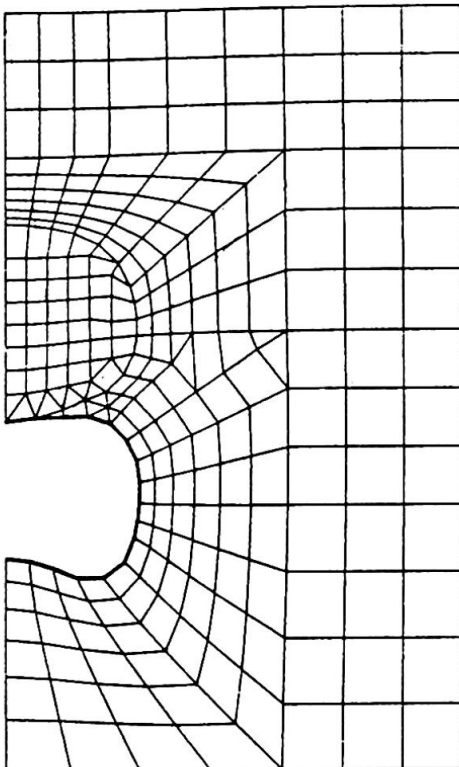
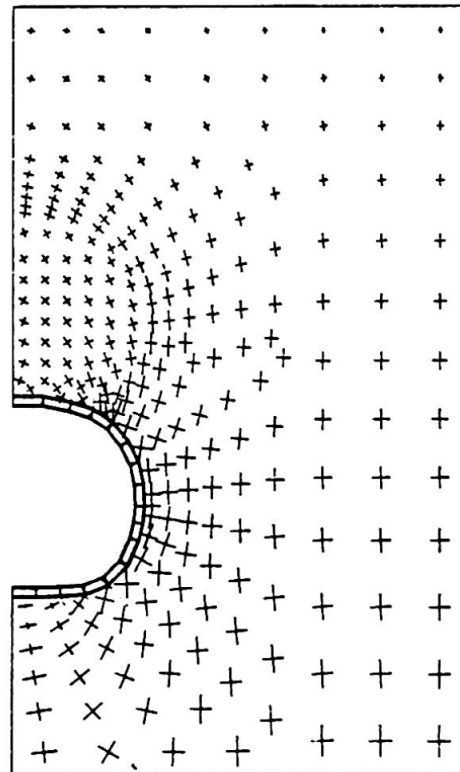


Bild 2: Elemente des Tunneltragwerkes

Bild 3: Verformungen im
2. BauzustandBild 4: Bodenspannungen im
3. Bauzustand

Das Programm NONSET berechnet jeden Bauzustand unter Hinzunahme der vorher ermittelten und in der Datenbasis gespeicherten Verformungen und Beanspruchungen.

3.2 Hohe Talbrücke im Taktschiebverfahren

Im Rahmen des Entwurfs für die Taktschiebverfahren herzustellende Talbrücke Morschler Grund war die Stabilität des Tragwerks zu untersuchen. Es wurde das Programm TOP [5], [6], ein Vorläufer von TOPSET, eingesetzt.

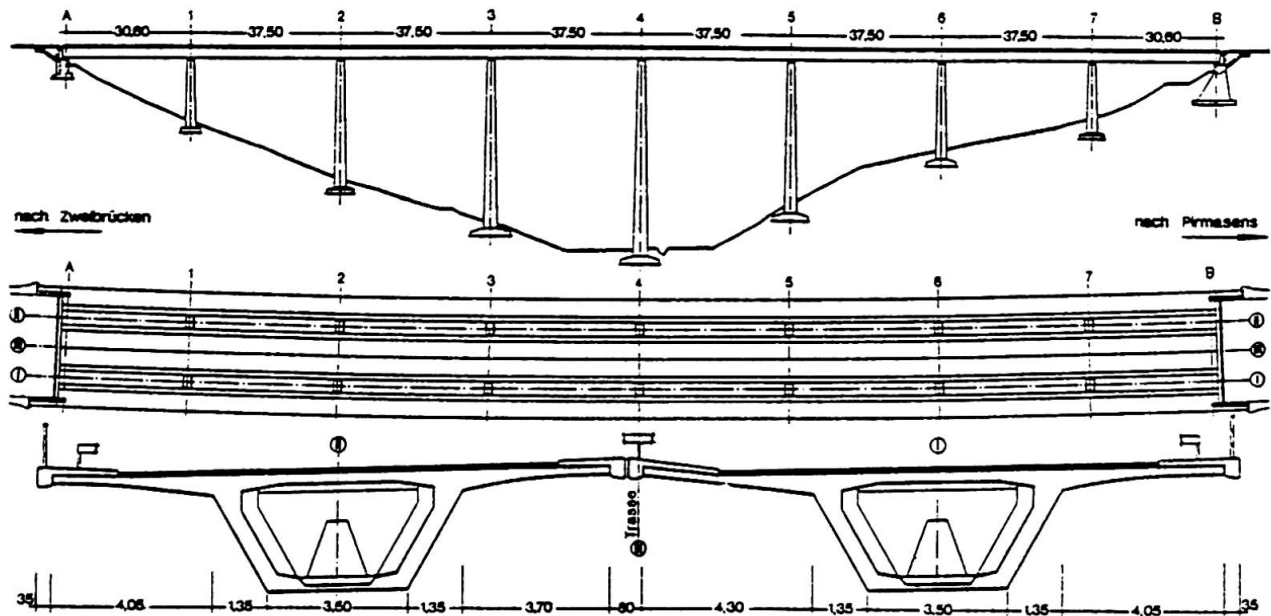


Bild 5: Talbrücke Morschler Grund/Dyckerhoff & Widmann AG, Wiesbaden

Bei einer Taltiefe von etwa 50 m war nicht von vornherein klar, welcher Vorbaustand der kritische sein würde, d. h. welcher unter erhöhten Lasten nach Theorie II. Ordnung bei Berücksichtigung der effektiven Biegesteifigkeiten die höchste Bewehrung erforderte.

Es wurden fünf Bauzustände und ein Endzustand untersucht. Das statische System war jedes Mal ein anderer räumlicher Rahmen, dessen Stiele mit veränderlichem Querschnittsverlauf am Fuß in Längs- und Querrichtung gefedert eingespannt wurden. Die Kopplung zwischen Überbau und Unterbauten erfolgte in Querrichtung durch feste Lager, in Längsrichtung durch feste Lager auf den Mittelstützen und verschiebliche Lager auf den übrigen Pfeilern. Um den Reibungseffekt einzugrenzen wurde eine Variante mit vorhandener und eine mit verschwindender Reibung gerechnet.

Zur Erfassung von Bauungenauigkeiten während des Taktschiebevorganges wurde zwischen Überbau und Pfeiler eine Zwangsklaffung in Höhe von $\pm 0,15$ ‰ berücksichtigt.

LITERATURVERZEICHNIS

- [1] ENCARNACAO, J.: Systemtechnologische Aspekte von CAD-Systemen. In Gnatz, R., Samelson, K.: Methoden der Informatik für rechnerunterstütztes Entwerfen und Konstruieren, Springer Berlin, 1977, S. 20-51



- | 2 | AXHAUSEN, K., FINK, TH., KATZ, C., RANK, E., V.VERSCHUER, TH., WERNER, H.: SET-Benutzerhandbuch, Technische Universität München, 1980
- | 3 | KATZ, C., V.VERSCHUER, TH., WERNER, H.: Data Handling in a Design Supporting Program Chain. Proceed. 5th Intern. Seminar on Computational Aspects of the Finite Element Method (CAFEM - 5), Berlin 1979
- | 4 | KATZ, C., WERNER, H.: FEM-Anwendungen im Tunnelbau, KfK-CAD 151: Anwendung der Finite-Element-Methode im Bauwesen, Kernforschungszentrum Karlsruhe, 1980
- | 5 | WERNER, H.: Rechnerorientierte Nachweise an schlanken Massivbauwerken. Beton- und Stahlbetonbau 73, 1978, S. 263-268
- | 6 | WERNER, H.: Programme für den Brückenentwurf. KfK-CAD 143: CAD im Brückenbau, Kernforschungszentrum Karlsruhe, 1979



VII

Organisation of Software

Organisation du logiciel

Organisation von Software

K. SRISKANDAN

Deputy Chief Highway Engineer

Department of Transport

London, England

SUMMARY

Computer hardware costs are decreasing all the time. On the other hand the production and maintenance of applications software is still a very expensive item and needs to be organised in such a way as to maximise the benefits from the investment in it. This paper discusses how to get the best out of applications software.

RESUME

Les coûts d'ordinateur sont en baisse constante. Cependant la création et le maintien du logiciel d'application sont toujours très chers; il faut donc organiser celui-ci de telle manière qu'on en tire au maximum les avantages de son investissement. Cet exposé discute les possibilités de mieux utiliser le logiciel d'application.

ZUSAMMENFASSUNG

EDV-Kosten sinken ständig. Andererseits sind die Erzeugung und der Unterhalt von Anwendungs-Software noch sehr teuer und sollen so organisiert werden, dass die Investitionen einen grösstmöglichen Nutzen bringen. Es wird erörtert, wie man solche Anwendungs-Software am besten benützen kann.



1. INTRODUCTION

Computer costs have reduced drastically in the last few years and this trend seems likely to continue. The production and maintenance of applications software is nevertheless a very expensive item and needs to be organised in such a way as to maximise the benefits from the investment in them. The Highways Engineering Computer Branch was set up in the Department of Transport in 1969 as a focal point for all Highways and Bridges Computing matters, including the provision of programs. In the light of experience in the Department of Transport, this paper discusses how to get the best out of applications software.

2. Structural programs and systems, whether they be for analysis or for optimised design, tend to be large (4,000 statements and upwards). The determination of the engineering requirements, parameters, the development and testing, and the subsequent documentation, is a very time-consuming process. At roughly 200 statements per man-month, these programs do represent a considerable investment and it is essential therefore that they are

- User orientated.
- Used by a large number of users without the massive duplication of effort that will take place if each organisation had to write its own programs.
- Maintained adequately and can be used during their working life without the need for significant alterations to match changes in hardware - even with advent of micro-computers.

3. USER REQUIREMENTS

3.1 Computer programs are written for the benefit of users who should therefore be consulted. The larger general purpose programs require considerable amount of data preparation for input and also unravelling of the output before it can be eventually used by the designer. Significant improvements can be made in the usefulness of programs by writing special input and output routines. The first of the input routines can be specific to a loading code, reducing even further the work of the designer. This will be of particular benefit with the new generation of codes which require the consideration of a variety of load combinations.

3.2 As for any development project, the cost-benefit from a program should be evaluated before the decision is taken to write it. In most civil engineering applications the designer has been able to use methods which would have been impractical without a computer. The benefits come partly from savings in design time but more from savings in material in the final design.

4. PORTABILITY

4.1 Clearly, the benefits from any one program will increase with the number of users. However, user organisations will not all have the same type of computer. Therefore programs which are written for a wide audience must be capable of running on any machine with

the minimum of change. In my Department programs have been written in a special sub set of ANSIFORTRAN which has enabled the programs to be run on any computer which offered FORTRAN, ie all makes of main frame and mini computers that have so far been encountered. A total of 8,124 copies of 166 different programs, some of which are grouped into systems, have been issued.

4.2 The writing of programs is so expensive, that the need for this kind of portability will apply not only to a central organisation which acts as a focal point, but also to individual organisations writing their own programs. The larger general purpose programs have a useful life of about 10 years before they become obsolete. Within this time the company computer is likely to have changed at least once - maybe even twice. Re-writing a program is a costly exercise. All of this assumes, of course, that the same high level language will be available in successive generations of computers. Even as I am writing this, FORTRAN is becoming available on some MICROS. Computer manufacturers should be made aware of the tremendous investment in the existing stock of software and unless there are over-riding reasons, should be "encouraged" to provide the capabilities to continue using these programs.

5. THE USER AND THE PROGRAM WRITER

5.1 Apart from the very small programs which a designer may write himself, the others are written by those who are employed full-time on programming. However, it is the designer who uses the program and he must have complete documentation, in the form of a user manual, on how to use the program. Any self respecting designer will also want to know the basic theory underlying the program, the suitability of it to his particular problem, and advice on idealisation and selection of parameters. In my Department we also prepare a program manual which describes both the theory and the program itself. In addition to helping the designer it also serves as a record which can be referred to if the program has to be amended or translated for some reason. Following discussions with designers on what further advice they would like, we have started work on the preparation of a series of guides giving advice on how to analyse given types of structures. This consists of the selection of type of computer program, idealisation of the structure and determination of the properties of its constituent parts, all to give a sufficient and sensible degree of accuracy at a reasonable cost. Some parts of the first of these guides dealing with slabs and pseudo-slab bridge decks [1] were completed in 1978.

5.2 A designer who uses a program (whether it be from the computer section of his own firm, or one from a central organisation, or one at a commercial bureau) will want to have confidence in its correctness. The user can check the overall goodness of a program by carrying out his own sensitivity checks and comparisons with exact results or he can rely on such checks done by a central organisation. However much a programmer may check his own program or check someone else's on behalf of the designer, it is the latter who prepares the data, unscrambles the output and finally uses it in the design of the structure. It is therefore the designer who must take full professional and legal responsibility for his design. There should never be any question of divided responsibility between the designer and the program writer.



6. MICROS

With the advent of the micro-computer and its direct response to the user, it should be possible to provide a library of short micro functions, ie individual processes of about 100 statements which can be assembled, preferably by the engineer, to carry out a number of different functions. Such ~~micro~~-functions will be easy to write, test and document and easy to understand. This will make designers more conversant with the computer and narrow the gap between programmer and designer. However the one great problem with this is that it requires open documentation and the user could corrupt the coding. In the interest of progress it seems to be a risk worth taking and my Department is looking into the feasibility of this proposal.

7. FOCAL POINT

Duplication in program writing is a waste of highly skilled manpower. Apart from program writing and associated documentation, there is also the need for program maintenance, specialist advice, and even having research carried out in special cases for program verification. It is very desirable that all these activities are centralised in one focal point which provides a service to the users. In UK, the Highways Engineering Computer Branch of the Department of Transport acts as the focal point for all computing related to road and bridge design. On the building side, there are cases of some Consulting Engineers forming a group of their own for this very purpose. Organisations vary in each country - but it should be possible to have focal points in software. At this stage such focal points can only be national - not international.

8. CONCLUSION

- Programs must reflect user requirements and therefore have special input and output facilities.
- Programs must be written in such a way that they can be used on a variety of computers - ie Portable.
- Manufacturers must be encouraged to provide capabilities in their new machines to continue using existing stock of programs.
- Users must have adequate documentation, both on the program and on how it is to be used.
- Designers must take full responsibility for the results as used in a design.
- The advent of Micros could result in a new approach to computing with the Engineer getting closer to the Computer
- A Focal Point for Computing will help to achieve some of the aims set out above.

REFERENCE

1. HECB/B1/7: User Guide for Slab and Pseudo Slab Bridge Decks, Department of Transport, London.

VII

Software Engineering — eine Notwendigkeit im Bauwesen

Software Engineering — A Necessity in Civil Engineering

Ingénierie du logiciel — une nécessité en génie civil

DIETRICH HARTMANN

Priv.-Doz. Dr. -Ing.

Universität Dortmund

Dortmund, Bundesrepublik Deutschland

ZUSAMMENFASSUNG

Die Erstellung von SOFTWARE hat — verursacht durch den Preisrückgang bei der HARDWARE — industrielle Ausmasse angenommen. Nur durch den gezielten Einsatz der neu entstandenen Ingenieurdisziplin SOFTWARE ENGINEERING wird es möglich sein, die SOFTWARE Entwicklung und -Wartung in den Griff zu bekommen. Es wird definiert, was SOFTWARE ENGINEERING bedeutet und welche Konsequenzen sich für die Ausbildung, Praxis und Forschung im Bauwesen ergeben.

SUMMARY

Computer HARDWARE is becoming ever cheaper. On the other hand SOFTWARE costs have risen enormously. This trend can only be stopped if the principles, methods and tools of the so-called SOFTWARE ENGINEERING are applied. The article shows what is meant by SOFTWARE ENGINEERING and describes its consequences for program development in civil engineering.

RESUME

Le matériel devient toujours meilleur bon marché, le logiciel au contraire toujours plus cher. L'augmentation des frais dans le domaine du logiciel ne peut être contrôlée que par l'emploi de la science du logiciel (Software Engineering). Il s'agit dans cet exposé de montrer la signification de l'ingénierie du logiciel et les conséquences de cette nouvelle discipline.



1. EINLEITUNG

Wer die neueste Entwicklung auf dem Computer-Markt studiert, stößt immer wieder auf Meldungen, welche besagen, daß die EDV im Bauwesen im Vergleich zu anderen Industriezweigen nach wie vor "Entwicklungsgebiet" ist [1]. Diese Feststellung wurde auch auf der im September 1979 in München stattgefundenen SYSTEMS 79 bestätigt, einer EDV-Messe, die sich immer mehr zum "EDV-Mekka" fürs Bauwesen entpuppt. Belegt werden derartige Behauptungen mit Hilfe statistischer Zahlen aus dem Jahr 1975 (vgl. [2]): der Anteil des Bauwesens am Bruttosozialprodukt beträgt danach rund 15%; der Anteil an den eingesetzten EDV-Anlagen dagegen nur 5%.

Es ist jedoch nicht überzeugend, den Anteil der eingesetzten Rechenanlagen im Bauwesen dazu zu benutzen, um das Bauwesen als "Entwicklungsgebiet" abzuqualifizieren. Man muß berücksichtigen, daß gerade im Bauwesen eine sehr große Zahl von kleineren Ingenieur- und Architekturbüros zu den 15% Anteil am Bruttosozialprodukt beiträgt und es für diese Büros sicherlich gar nicht so sinnvoll war, vor 1975 (als Bezugszeitpunkt der o.g. Erhebung) Rechner oder EDV zu betreiben, da aufgrund hoher Anschaffungs- oder Mietkosten die Wirtschaftlichkeit der Büros in Frage gestellt worden wäre. Der wichtigste Grund für den geringen EDV-Einsatz waren also hauptsächlich die hohen Kosten für die HARDWARE. Ein zweiter Grund war sicherlich auch der geringe Komfort im Hinblick auf die zu erfüllenden, in ihrer Art unterschiedlichsten Aufgaben, wie Planung, Statik, Dimensionierung, Zeichnungserstellung und Abrechnung. Nun haben sich in den letzten 5 Jahren die Verhältnisse enorm geändert: die HARDWARE wird billiger und billiger, der Komfort steigt. Im folgenden sollen insbesondere die durch den Preisverfall der HARDWARE eingetretene Situation dargestellt, zukünftige Trends aufgezeigt und Konsequenzen für Aus- und Fortbildung, Forschung und Entwicklung im EDV-Bereich des Bauwesens gezogen werden.

2. HARDWARE UND SOFTWARE SITUATION HEUTE

Es ist allgemein bekannt, wodurch der Preisverfall der HARDWARE bewirkt wurde: Die neue LSI-Technologie¹⁾, durch die Tausende elektronischer Bauelemente auf nur wenigen mm² großen Plättchen - den sogenannten chips - untergebracht werden können, erlaubt es, unsere heutigen Computer industriell und dadurch billig herzustellen. Die HARDWARE wurde aber nicht nur "spottbillig", sondern gleichzeitig auch noch schneller, betriebssicherer, robuster und in den Abmessungen erheblich kleiner. Gerade die Miniaturisierung der Elektronik hat den Markt in den letzten Jahren total verändert und Rechnertypen herausgebracht, die mit Sicherheit das Arbeiten im Bauwesen erheblich beeinflussen werden.

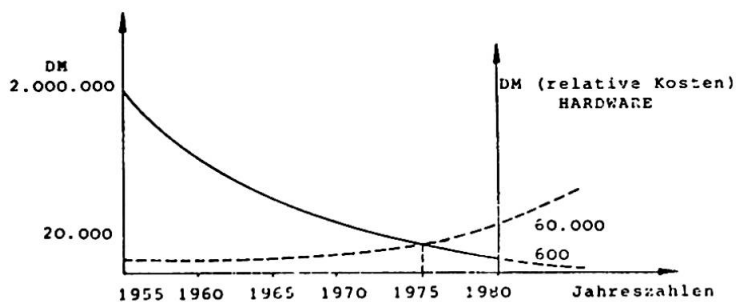
Um einen Überblick über den derzeitigen Stand der Technik zu erhalten, werden die verschiedenen Rechnertypen tabellarisch zusammengestellt:

1) LSI = Large Scale Integrated

Eigenschaften Rechnertyp	Kosten	Output	Speicher	Sprache
programmierbarer Taschenrechner	100 DM bis 1000 DM	Papierrollen Printer	Magnet Karteien	system- abhängig mnemotech.
Personalcomputer	1000 DM bis 10000 DM	Printer, Plotter, Display	Kassette Floppy Disk	BASIC
Microcomputer	10000 DM bis 40000 DM	Printer, Plotter, Display	Floppy Disk	BASIC FORTRAN
Minicomputer	80000 DM bis 600000 DM	Printer Plotter Display	Disk magnetband	BASIC FORTRAN PASCAL COBOL
Mainframe Computer	500000 DM bis 10000000 DM	Printer Plotter Display	Disk Magnetband	keine Einschränkung

Aus der obigen Tabelle ist zu entnehmen, daß selbst für kleinere Büros die Möglichkeit besteht, für erschwingliches Geld funktionstüchtige - wenn auch relativ langsame - "Datenverarbeitungssysteme" betreiben zu können. Es ist zu erwarten, daß gerade im Mini- und Mikrocomputerbereich in den nächsten 5 bis 10 Jahren erhebliche Anstrengungen unternommen werden, um die HARDWARE noch leistungsfähiger, d.h. noch schneller zu machen und mit mehr Kapazität und Komfort auszustatten.

Wie stark sich der Preisverfall der HARDWARE ausgewirkt hat, läßt sich am besten aus der folgenden Graphik ablesen, in der die relativen Kosten für die HARDWARE der letzten 25 Jahre aufgetragen sind (vgl. [3]).



In der Weise wie die Kosten für die "schlaue Kiste" Computer geschrumpft sind, ist jedoch das Bedürfnis gestiegen, den Computer zu nutzen - eben weil er so billig geworden ist! Selbst der Laie weiß jedoch, daß die HARDWARE allein nicht ausreicht, um EDV einzusetzen. Man braucht "Programme" also eine zweite "WARE-Form", die sogenannte

SOFTWARE. Die Erstellung von SOFTWARE ist aber arbeits- und lohnintensiv - und demnach teuer. Um zu zeigen, wie sich die SOFTWARE-Kosten entwickelt haben, sind diese in die obige Graphik (gestrichelt) eingetragen worden (vgl. [3]).

Die Erstellung von SOFTWARE hat - wie hieraus zu ersehen ist - ein erhebliches wirtschaftliches Gewicht bekommen: betrug 1955 das Kostenverhältnis HARDWARE : SOFTWARE noch 100:1, so hat sich dieses Verhältnis heute genau ins Gegenteil verkehrt, nämlich 1:100! Verursacht durch den Preissturz der HARDWARE und ihre große Verbreitung, hat die Erstellung von SOFTWARE industrielle Außmaße angenommen. War früher die HARDWARE ein Engpaß der Daten-



verarbeitung, so jetzt zunehmend die SOFTWARE. Dieser Entwicklung muß mehr entgegengewirkt werden als bisher, will man die Kostenschere nicht noch weiter aufklaffen lassen. In den letzten Jahren hat sich deswegen folgerichtig so etwas wie eine neue Disziplin, das SOFTWARE ENGINEERING herausgebildet, mit dem man die Schwierigkeiten bei der Entwicklung, Wartung und Anwendung von SOFTWARE in den Griff bekommen möchte. Was hierunter im einzelnen zu verstehen ist und welche Folgerungen sich für die EDV im Bauwesen ergeben, soll in den folgenden Kapiteln abgehandelt werden.

3. SOFTWARE ENGINEERING - RETTUNG IN DER NOT

Jeder Bauingenieur, der eine Brücke oder ein Hochhaus bauen will, weiß genau, wie er sein "Ingenieurprodukt" zu erstellen hat. Er verwendet Methoden, die auf wissenschaftlich fundierten Grundlagen - wie der Elastizitäts- oder Plastizitätstheorie - und durch einschlägige Vorschriften abgesichert sind. Bei der Herstellung seines Produktes greift er außerdem noch auf Methoden des Managements und der Kostenkontrolle zurück, um möglichst wirtschaftliche Lösungen zu erzielen.

Ganz anders ist die Situation bei der Erstellung von SOFTWARE. Im Gegensatz zu den traditionellen technischen Disziplinen, wie etwa dem Bauingenieurwesen oder Maschinenbau - im anglo-amerikanischen mit CIVIL ENGINEERING bzw. MECHANICAL ENGINEERING bezeichnet - gibt es keine eigenständige Ingenieurdisziplin SOFTWARE ENGINEERING. Wenn demnach der Begriff SOFTWARE ENGINEERING verwendet wird, so ist dies eigentlich mehr als Provokation zu verstehen, da das "SOFTWARE ENGINEERING" zur Zeit noch längst nicht das Niveau der klassischen Ingenieurfächer erreicht hat. Der Begriff SOFTWARE ENGINEERING charakterisiert vielmehr eine veränderte Einstellung und Arbeitshaltung zur SOFTWARE - hervorgerufen durch die enormen Kostensteigerungen im SOFTWARE-Bereich.

Obwohl das SOFTWARE ENGINEERING erst in der Entwicklungsphase ist, soll dennoch kurz so etwas wie eine Definition gegeben werden. Wir wollen unter SOFTWARE ENGINEERING im folgenden analog zu [4] die Bereitstellung von Prinzipien, Methoden und Werkzeugen für die SOFTWARE-Entwicklung und -Wartung auf der Basis wissenschaftlicher Erkenntnisse und praktischer Erfahrungen verstehen. Ziel ist es dabei, die SOFTWARE-Qualität entscheidend zu verbessern und somit der unheilvollen Kostenexplosion im SOFTWARE-Bereich entgegenzuarbeiten.

Prinzipie sind allgemeingültige Grundsätze des Denkens oder Handelns im Sinne einer Norm. (Beispiele: Modularisierung, begleitende Dokumentation).

Methoden sind planmäßige Vorgehensweisen im Hinblick auf ein Ziel, meistens auf der Grundlage eines allgemeinen Prinzips. (Beispiele: Strukturierte Programmierung, Programmierstandards, Netzplantechnik und Terminplanung).

Werkzeuge sind Arbeitsmittel, die für die Erstellung von SOFTWARE verfügbar sind. (Beispiele: Programmiersprachen, Programm-

bibliotheken, Makroprozessoren oder Generatoren, Pre- und Postprocessing).

Ziel des SOFTWARE ENGINEERING ist die bewußte Qualitätsverbesserung der SOFTWARE, wobei folgende Qualitätskriterien maßgebend sind:

- Effizienz
- Benutzerkomfort
- Zuverlässigkeit
- Portabilität
- Änderbarkeit

Es liegt auf der Hand, daß die einzelnen Qualitätskriterien nicht immer gleichzeitig durchsetzbar sind und somit Zielkonflikte auftreten. Beispielsweise ist es nicht möglich ein Programm zugleich hinsichtlich seiner Effizienz optimal zu machen und mit höchstem Benutzerkomfort auszustatten, da diese beiden Eigenschaften konträr zueinander sind. Es kommt auf den jeweiligen Einzelfall an, ob einem bestimmten Qualitätskriterium der Vorzug zu geben ist oder ein ausgewogenes Mittel verschiedener Qualitätsmerkmale sinnvoller ist. Allen Beteiligten, d.h. dem Hersteller von SOFTWARE und dem Anwender muß jedoch klar sein, daß zwischen Qualität, Kosten und Zeitaufwand zahlreiche Abhängigkeiten und Rückkopplungen bestehen.

Welche Konsequenzen ergeben sich aus dem SOFTWARE ENGINEERING für Ausbildung, Praxis und Forschung des Bauingenieurwesens? Im folgenden soll hierauf eine kurze Antwort gegeben werden.

4. FOLGERUNGEN FÜR AUSBILDUNG UND FORSCHUNG

Fragen wir zunächst einmal, inwieweit die heutige Ausbildung unserer Bauingenieure den veränderten Verhältnissen durch die Einwirkung der modernen EDV Rechnung trägt? Die Antwort lautet: Wir sind noch weit von einer qualifizierten EDV-Ausbildung entfernt. Im wesentlichen beschränkt man sich derzeit lediglich darauf, den Studenten eine Programmiersprache (etwa FORTRAN) beizubringen. Dies reicht aber angesichts des heute ablesbaren Trends, daß die HARDWARE immer billiger und die "SOFTWARE" ständig teurer wird, nicht mehr aus. Wer verantwortungsbewußt denkt, sollte deswegen folgende Forderungen an die Ausbildung der Bauingenieure stellen:

- a) es muß für eine schon zu Beginn des Studiums einsetzende, fachspezifische und praxisorientierte EDV-Grundausbildung der Bauingenieurstudenten gesorgt werden.
- b) die EDV-Grundausbildung muß das gleiche Gewicht erhalten wie auch andere Grundlagenfächer.
- c) im Hauptstudium müssen die Grundlagen sinnvoll durch eine kooperative Zusammenarbeit mit den klassischen Bauingenieurfächern vertieft werden. Insbesondere müssen Kenntnisse des SOFTWARE ENGINEERING, das Arbeiten mit großen Programmsystemen, Probleme des CAD und der graphischen Datenverarbeitung vermittelt werden.

Für die Praxis müssen insbesondere die SOFTWARE-Probleme klei-



ner und mittlerer Anwender abgebaut werden. Da dem Anwender die Übersicht fehlt, brauchte man quasi ein "Testlabor" für Programme, das mit Mitteln des SOFTWARE ENGINEERING objektiv die angebotenen Programme auf ihre Qualität hin überprüft. Da ein solches Labor bei realistischer Einschätzung der Gegebenheiten Utopie ist, müssen von den Entwicklern wenigstens Mindestanforderungen erfüllt werden, die dem Anwender Möglichkeiten des Vergleichs an die Hand geben:

- a) die Programme müssen einheitlich dokumentiert und strukturiert programmiert werden,
- b) es muß eine genaue Angabe darüber erfolgen, wie Änderungen vorgenommen werden können.

Für die computergestützte Forschung sind allgemein gültige EDV-Richtlinien auszuarbeiten, die - wie andere Normen - anerkannt und eingehalten werden. Wichtigste Forderungen sind:

- a) die Verpflichtung, eine ausreichende Portabilität der SOFTWARE zu gewährleisten, um den Aufwand für die Erstellung von neuer SOFTWARE zu reduzieren.
- b) die Erstellung einer einheitlichen Dokumentation für strukturiert programmierte SOFTWARE,
- c) die Einrichtung einer koordinierenden Stelle, die bereits existierende SOFTWARE hinsichtlich ihrer Qualität zu überprüfen und gegebenenfalls Verbesserungen vorzunehmen hat.

5. ABSCHLIESSENDE BEMERKUNG

Der Verfasser dieses Aufsatzes ist der Meinung, daß vor allem durch eine verbesserte fachspezifische Ausbildung und durch den gezielten Einsatz der Prinzipie, Methoden und Werkzeuge des SOFTWARE ENGINEERING in Praxis und Forschung in wenigen Jahren jeglicher Verdacht, das Bauwesen sei ein "Entwicklungsgebiet" der EDV, ad absurdum geführt wird.

REFERENZEN

- [1] COMPUTERWOCHE: Deutsches Bauwesen ist noch DV-Entwicklungsgebiet, 5.10.1979, S. 32.
- [2] SYSTEMS 77: Computersysteme und ihre Anwendung, Branchen-Seminar Bauwesen, München, 1977, S. 13.
- [3] WILSON, E.L.: Role of small computer systems in structural engineering, Electronic Computation 7th Conference, American Society of Civil Engineers, New York, 1979.
- [4] GEWALD, K.; HAAKE, G.; PFADLER, W.: Software Engineering, Reihe Datenverarbeitung, R. Oldenbourg Verlag, München, Wien, 1979.

VII

„**FLOWERS: Ein neues Finite-Element-Programm für Lehre, Praxis und Forschung**“

„FLOWERS: A New Finite Element Programme for Teaching, Practice and Research“

„FLOWERS: Un nouveau programme d'éléments finis pour l'enseignement, la pratique et la recherche“

E. ANDERHEGGEN

Prof. Dr.

Institut für Informatik ETHZ

Zürich, Schweiz

U. WALDER

Dr.

Institut für Informatik ETHZ

Zürich, Schweiz

3

ZUSAMMENFASSUNG

Ein neues Programmsystem für verschiedenartige Anwendungen der Methode der finiten Elemente wird zur Zeit am Institut für Informatik der Eidgenössischen Technischen Hochschule in Zürich entwickelt. Die Ziele, die man dabei verfolgt, werden hier erläutert.

SUMMARY

A new general purpose finite-element programme is being written at the Institute for Computer Science of the Swiss Federal Institute of Technology in Zurich. The main criteria followed for this software development are discussed in the present paper.

RESUME

Un nouveau programme d'éléments finis est développé à l'Institut d'Informatique de l'Ecole Polytechnique Fédérale de Zurich. On discute ici les principaux critères suivis pour sa réalisation.



1. EINLEITUNG

Der vorliegende Artikel soll die Ziele erläutern, die man am Institut für Informatik der Eidgenössischen Technischen Hochschule in Zürich mit der Entwicklung des neuen Finite-Element-Programms FLOWERS verfolgt. Dabei geht es nicht um theoretische Herleitungen die z.B. FE-Modelle oder Lösungsalgorithmen betreffen, sondern in erster Linie um die Anforderungen, die heute an ein modernes, den Bedürfnissen der Lehre, Forschung und Praxis Rechnung tragendes FE-Programm zu stellen sind. Bei der Planung des Programmsystems FLOWERS stützen sich die beiden Autoren auf die Erfahrungen, die sie bei der Entwicklung, Anwendung und Betreuung der in der Schweiz seit Jahren viel verwendeten Programme STRESS, FLASH und STATIK gesammelt haben. Auf einige Hauptmerkmale dieser drei Vorgänger des Programmsystems FLOWERS soll deswegen im nächsten Abschnitt kurz eingegangen werden.

2. DIE PROGRAMME STRESS, FLASH UND STATIK

Gegen Mitte der sechziger Jahre wurde am Institut für Baustatik und Konstruktion der Eidgenössischen Technischen Hochschule in Zürich das ursprünglich am Massachusetts Institute of Technology entwickelte Programm STRESS [1] übernommen, verschiedentlich verbessert und den Bauingenieurstudenten als didaktisches Hilfsmittel zur Verfügung gestellt. Das Programm wurde dann in zwei von über 500 Ingenieuren besuchten Fortbildungskursen der Praxis vorgestellt und wurde bald das während mehreren Jahren in der Schweiz am meisten verwendete baustatische Computerprogramm. Historisch gesehen spielte STRESS in der Schweiz und auch anderswo eine wichtige Rolle, da es für viele Bauingenieure den ersten Kontakt mit technischen Computeranwendungen darstellte. STRESS kann nur linear-elastische, statisch-belastete ebene und räumliche Rahmen- und Fachwerke berechnen. Darin liegt aber einer der Hauptgründe seines Erfolges. Von den vielen möglichen Anwendungen allgemeiner FE-Programme ist nämlich die lineare Stabstatik bei vielen Bauingenieuren das einzige, das sie zur Lösung praktischer Bemessungsaufgaben brauchen und das sie folglich auch klar überblicken. Ein weiterer Grund des Erfolges von STRESS lag in seiner leicht erlernbaren, problemorientierten, formatfreien Eingabesprache. Diese erlaubte den Ingenieuren aus der Praxis, selbst wenn sie nur sporadische STRESS-Benutzer waren, die Eingabeanweisungen ohne Hilfe von Spezialisten selber vorzubereiten.

Nach dem Erfolg von STRESS wandte man sich einem weiteren, für Lehre und Praxis wichtigen Anwendungsgebiet von FE-Programmen, nämlich der statischen Analyse von Flächentragwerken zu. Dafür wurde das Programm FLASH [2], [3] zur Berechnung linear-elastischer, statisch belasteter Scheiben, Platten und Schalen entwickelt. FLASH ist das erste

Grossprogramm, dessen Eingabesprache mit Hilfe von Syntaxdiagrammen beschrieben wird. Diese erlauben auf äusserst übersichtliche Art und Weise die vom Programm erwartete Reihenfolge der problembezogenen Eingabedaten, d.h. die Syntax der Eingabesprache festzulegen. Die vollständige Eingabesprache von FLASH mit all den vielen Möglichkeiten, die im Programm eingebaut sind, benötigt nur ein dreiseitiges, leicht überblickbares Syntaxdiagramm, das in der Benutzeranleitung im Detail beschrieben wird. Selbst der mässig geübte Programmbenützer kann nach einer einmaligen Studie der Benutzeranleitung allein mit dem Syntaxdiagramm seine Eingabe jeweils vorbereiten. Syntaxdiagramme sollen hier nicht näher erläutert werden (siehe auch Abb. 1). Es soll jedoch betont werden, dass sie auch im Hinblick auf die positiven Erfahrungen, die man damit mit dem Programm STATIK gemacht hat, in den Augen der beiden Autoren als unerlässliches Attribut von lehr- und praxisorientierten Computerprogrammen zu betrachten sind. Neben der besonders bequemen Eingabe hat FLASH, nach mehreren Jahren Entwicklungszeit, heute einen Reifegrad erreicht, der die meisten Bedürfnisse der Praxis, selbst für anspruchsvolle Benutzer erfüllen dürfte: numerische Effizienz, ausgedehnte graphische Ausgabe, interne Variablennummeroptimierung, Restart-Möglichkeiten usw.

Als weiteres Grossprogramm wurde gegen Mitte der siebziger Jahre, das Programm STATIK [4], [5] entwickelt, dessen Hauptziel in der Ersetzung des längst veralteten und in mancher Hinsicht recht primitiven STRESS-Programms bestand. STRESS und STATIK behandeln die gleiche Art von Problemen und wenden sich an den gleichen Benutzerkreis. Beim Programm STATIK hat man sich von vornherein bewusst auf Probleme beschränkt, bei denen eine theoretisch exakte Lösung und nicht wie üblich bei der FE-Methode, eine Näherungslösung gefunden werden kann. Damit stellt sich die für wenig sachkundige Benutzer oft schwierige Frage nach der Wahl des geeigneten mathematischen Modells (Elementmasche, Elementtypen, usw.) bei gegebenem statischen Modell nicht. Man gewann nämlich den Eindruck, dass viele bekannte FE-Programme für durchschnittliche Bauingenieuranwendungen zu viel können, was ihren Einsatz erschwert und oft die Inanspruchnahme von Spezialisten notwendig macht. Das Programm STATIK wurde mit einem Aufwand von sieben bis acht Mannjahren entwickelt. Es deckt ein relativ enges Gebiet ab (lineare Stabstatik, inklusiv Querschnittanalyse, Vorspannberechnungen, Einflusslinien, Berücksichtigung von Bauzuständen, usw.), bietet jedoch, dank Syntaxdiagrammen, graphischer Ausgabe, Restart-Möglichkeiten, usw. einen von keinem der beiden Autoren bekannten Stabwerkprogrammen erreichten Benutzercomfort. STATIK ist heute an der ETH Zürich und an anderen Hochschulen ein viel verwendetes didaktisches Instrument geworden. Es wird in der Schweiz, sowie mehr und mehr auch in Nachbarländern, von vielen Ingenieuren aus der Praxis regelmässig verwendet.

Sowohl FLASH wie STATIK wurden ursprünglich für Grosscomputeranlagen, die im Batch-Betrieb arbeiten geschrieben.



Die heutige Tendenz, technische Anwendungen auf kleineren, dezentral gelegenen Kleincomputern zu implementieren hat es jedoch notwendig gemacht, völlig portable Programmversionen zu entwickeln. Der potentielle Benutzerkreis erweitert sich damit auf die steil wachsende Anzahl von Bauingenieurbüros sowie von Ingenieurberufsschulen, die einen eigenen Kleincomputer besitzen.

3. ZIELE DES NEUEN PROGRAMMSYSTEMS FLOWERS

Anwendungsbereich und angesprochene Benutzerkreise sind bei den erwähnten Programmsystemen begrenzt. Beim neuen Programmsystem FLOWERS sollen diese beiden Beschränkungen weitgehend fallen gelassen werden.

Bezüglich Anwendungsbereich soll sich das Programmsystem FLOWERS bei möglichst vielen Problemarten einsetzen lassen, die sich nach der FE-Methode behandeln lassen. Dazu gehört eine Reihe kontinuumsmechanischer und nichtkontinuumsmechanischer, statischer bzw. stationärer und dynamischer bzw. instationärer, linearer und nichtlinearer Probleme des Bau-, des Maschinen- und des Elektroingenieurwesens. Das Programmsystem FLOWERS ist von vornherein so geplant, dass neue Anwendungen der FE-Methode mit einem Minimum an Neuprogrammierung, d.h. unter Verwendung möglichst vieler schon vorhandener Programmteile, implementierbar sind. FLOWERS wird damit, im Gegensatz zu den im letzten Abschnitt besprochenen Programmen, auch noch ein Instrument der Hochschulforschung werden, indem es ein Rahmenprogramm bilden wird, das die Implementation neuer Anwendungen mit verhältnismässig wenig Aufwand ermöglicht.

Bezüglich Anwenderkreis richtet sich das FLOWERS System:

- an Bau-, Maschinen- und Elektroingenieurstudenten die FLOWERS als didaktisches Instrument verwenden. Für diese Benutzer werden verschiedene Standardprogrammversionen vorhanden sein, die klar definierte Probleme lösen und einfach zu gebrauchen sind. Wie bei STATIK und FLASH soll diese letzte Forderung mit Hilfe einer möglichst leicht erlernbaren, mittels Syntaxdiagrammen beschriebenen Eingabesprache, einer umfassenden graphischen Ausgabe und einer klaren Dokumentation erreicht werden.
- an Ingenieure aus der Praxis, die das Programm selbstständig, jedoch nicht notwendigerweise als Hauptbeschäftigung anwenden wollen. Die Bedürfnisse dieses Benutzerkreises entsprechen weitgehend denen der Ingenieurstudenten. Allerdings kommen in der Praxis oft Probleme der Datenaufbereitung und der Resultatweiterverarbeitung vor, die man gerne mit Hilfe von Vor- und Nachlaufprogrammen löst. Besondere Vorkehrungen wurden deswegen getroffen, um die problembezogenen

Eingabedaten wahlweise aus einer manuell oder aus einer maschinell erzeugten Eingabedatei lesen zu können. Dementsprechend können die Ausgabedaten sowohl auf einem von Menschen als auch auf einem von Nachlaufprogrammen leicht lesbaren Ausgabedatei geschrieben werden. Dies erlaubt u.a. die Eingabedaten, z.B. mit Hilfe graphischer Eingabegeräte, auf einem eigenen Kleincomputer interaktiv vorzubereiten und die Ausgaberesultate, vielleicht zu Bemessungszwecken, ebenfalls interaktiv weiterzuverarbeiten. Das FLOWERS-System, das im Batch-Betrieb arbeitet und das möglicherweise auf einem entfernten Grosscomputer installiert sein wird ("Distributed Processing") wird dann nur für die zentralen, rechenintensiven, numerischen Berechnungen eingesetzt.

- an Hochschulforscher die auf der Stufe der lokalen Elementmatrizen eigene Entwicklungen austesten und implementieren wollen. Weil es bei der FE-Methode möglich ist, die globale Systembehandlung und die numerische Erfassung der elementeigenen Eigenschaften mittels lokaler Elementmatrizen klar voneinander zu trennen, konnte das FLOWERS-System so ausgelegt werden, dass es relativ leicht sein wird, selbstgeschriebene Elementunterprogramme anzuhängen. Dateneingabe, Zusammensetzung und Lösung der globalen Systemgleichungen, interne Datenorganisation, Iterationsverfahren bei nichtlinearen Anwendungen, graphische Ausgabe, usw. können somit vom FLOWERS-System unverändert übernommen werden.
- an besonders sachkundigen Hochschulforscher, die auf der Stufe des globalen Systems, z.B. bei speziellen nichtlinearen Anwendungen, Eigenentwicklungen implementieren wollen. Solche Benutzer werden selbstverständlich, um überhaupt gewisse Programmteile übernehmen zu können, eine tiefgehende Kenntnis des Quellenprogramms und der verwendeten Datenorganisation vom FLOWERS-Programm benötigen. Dies soll erleichtert werden durch eine klare Programmstrukturierung und eine umfassende Dokumentation vor allem in Form von im Programm eingebetteten Kommentaren.
- an Informatik-Studenten, deren Hauptinteresse bei Fragen der Programmierung liegt. Dafür soll das FLOWERS-System ein beispielhaft sauber geschriebenes und optimal strukturiertes FORTRAN-Programm werden.

Zu den Zielen, die beim FLOWERS-System nicht allzu stark in Vordergrund stehen, sind Kapazität und z.T. sogar numerische Effizienz zu nennen. Die Möglichkeit einer Unterteilung des Systems in Substrukturen, was bei gewissen Grossproblemen des Flugzeug- oder des Schiffbaus heute als unerlässlich erscheint, ist z.B. nicht geplant. Kapazität und Effizienz sind nämlich Fragen, die in höherem Masse mit der verwendeten Hardware, bei der z.T. noch unabsehbare Fortschritte zu erwarten sind, verknüpft sind (billige Speicher und Prozessoren, "Virtual-Storage", usw.). Benutzercomfort und Flexibilität sind hingegen Fragen, die



die Software allein betreffen und die heute sicherlich die grösste Aufmerksamkeit verdienen.

4. SYSTEMAUFBAU

Das Programmsystem FLOWERS besteht aus mehreren sogenannten Programmmodulen, aus einer Elementbibliothek und aus einer Reihe von Standardprogrammversionen. Die Programmmodule setzen sich aus Unterprogrammen zusammen, die auf der Stufe des globalen Systems und damit weitgehend unabhängig vom behandelten physikalischen Problem arbeiten. Die gleichen Programmmodule können folglich für völlig verschiedene Problemarten, wie z.B. Schalenstatik und Wärmeleitungsanalyse eingesetzt werden. Die Elementbibliothek besteht aus einer Anzahl von Unterprogrammen, die für spezifische physikalische Probleme die lokalen Elementmatrizen numerisch bestimmen. Bei linearen kontinuumsmechanischen Problemen sind diese die lokalen Steifigkeits-, Last-, geometrischen Steifigkeits-, Massen- und Spannungsmatrizen der einzelnen Elemente. Die Standardprogrammversionen bestehen aus mehreren Programmmodulen an denen die Unterprogramme bestimmter Elementarten angeschlossen sind, so dass spezifische Problemtypen gelöst werden können.

Folgende Programmmodule sind z.Z. fertiggestellt oder befinden sich in einer fortgeschrittenen Entwicklungsphase:

- SYSIN zur Eingabe und Kontrolle der system- und lastbezogenen Daten sowie zur Berechnung der lokalen Elementmatrizen (siehe Abb. 1).
- SOLVE zur Zusammensetzung und Lösung der globalen Systemgleichungen sei es in Form eines linearen Gleichungssystems oder eines Eigenwertproblems.
- GRAPH zur graphischen Elementmasche- und Resultatausgabe.
- STOUT zur numerischen Resultatausgabe bei statischen bzw. stationären Problemen.
- DYNOUT für die Zeitintegration und zur numerischen Resultatausgabe bei dynamischen bzw. instationären Problemen.

Mit diesen Programmmodulen dürfte ein grosser Teil der in der Praxis vorkommenden linearen Probleme lösbar sein: Statische Analyse erster und zweiter Ordnung, Stabilitätsanalyse, lineare Dynamik entkoppelter Eigenschwingungen sowie verschiedene stationäre und instationäre Feldprobleme. Zwei weitere noch unbenannte Programmmodule für die nicht-lineare statische Analyse und die nicht-lineare dynamische Analyse von Tragwerken befinden sich z.Z. in einer früheren Entwicklungsphase.

Die Elementbibliothek umfasst heute Unterprogramme für Stabelemente in zwei oder drei Dimensionen, für Scheibenelemente, für rotationsymmetrische Ringelemente sowie dreidimensionale Elemente zur Berechnung elektrischer und magnetischer Felder. Weitere Elementunterprogramme für Platten, Schalen, dreidimensionale Kontinua sowie für verschiedene Arten von zwei- und dreidimensionalen Feldproblemen werden demnächst zur Verfügung stehen.

Eine Standardprogrammversion für Scheibenprobleme wurde im Sommersemester 1980 zum ersten Mal von Bau- und Maschinenbauingenieurstudenten der ETH Zürich zu Übungszwecken verwendet.

Die Programmdokumentation ist in Vorbereitung. Sie wird aus einer Benutzeranleitung und einer Programmieranleitung bestehen. Die Benutzeranleitung richtet sich an Programmanwender, die nur Standardversionen, ev. mit angeschlossenen Vor- und Nachlaufprogrammen, verwenden wollen. Sie enthält, neben allgemeinen Informationen, die Syntaxdiagramme, welche die Eingabesprache sowohl der Programmmodule als auch der verschiedenen Elementtypen festlegen, sowie die Detailbeschreibung der einzelnen Eingabeweisungen. Die Programmieranleitung richtet sich an Programmierer die das FLOWERS-System erweitern oder modifizieren wollen und soll in erster Linie als Ergänzung zum Quellenprogramm dienen, wo die notwendige Detailinformationen zu finden sind.

5. SCHLUSSWORT

Die FE-Methode hat in vielen Gebieten der Technik eine derartige Verbreitung gefunden, dass entsprechende einführende Kurse im Angebot der Lehrveranstaltungen zumindest für Bau- und Maschinenbauingenieurstudenten nicht länger fehlen dürfen. Dazu ist es unerlässlich, dass die Studenten selbständige Erfahrungen mit der Anwendung vorhandener FE-Programme sammeln können. Diese Programme müssen ein breites Anwendungsspektrum abdecken, gleichzeitig aber transparent und leicht zu gebrauchen sein. Ähnlich Anforderungen gelten für FE-Programme die für die Praxis bestimmt sind, und zwar vor allem dann, wenn diese von Ingenieuren verwendet werden sollen, die die Verantwortung nicht nur für die Analyse sondern auch für den Entwurf und die Bemessung von Bauwerken oder Maschinen tragen und die somit keine ausgesprochene FE-Spezialisten sein können. Schliesslich ist die FE-Methode ein wichtiges Instrument der Hochschulforschung geworden. Dabei wäre viel geholfen, wenn Neuentwicklungen sich auf einem möglichst flexiblen Rahmensystem stützen können, sodass gewisse Probleme, die sich bei ganz verschiedenartigen Anwendungen immer wieder stellen, nicht jedes mal von vorne neu gelöst werden müssen. Mit dem neuen Programmsystem FLOWERS versucht man allen diesen Anforderungen gerecht zu werden.



LITERATUR

- [1] E. ANDERHEGGEN, G. ALBERTI, A. LAESSER: "Handbuch der STRESS-Sprache", Institut für Baustatik und Konstruktion, ETH - Zürich, April 1969.
- [2] U. WALDER, D. GREEN: "FLASH Benutzeranleitung", Institut für Informatik ETH Zürich, FIDES Treuhandgesellschaft, CDC Data Services, 5. Auflage, 1980.
- [3] U. Walder: "FLASH: A Simple Tool for complicated Problems", Proc. First Conference on Engineering Software, Southampton, September 1979.
- [4] E. ANDERHEGGEN, G. BAZZI, P. MUELLER, J. THEILER, U. WALDER: "STATIK Benutzeranleitung", Institut für Informatik ETH Zürich, FIDES Treuhandgesellschaft, CDC Data Services, 2. Auflage, 1979.
- [5] E. ANDERHEGGEN: "STATIK: A Computer Programm for Everyday's Struktural Engineering Applications", Int. Colloquium on Interface between Computing and Design in Structural Engineering, IVBH, Bergamo, August 1978.
- [6] E. ANDERHEGGEN: "FLOWERS User's Manual" (in Vorbereitung)

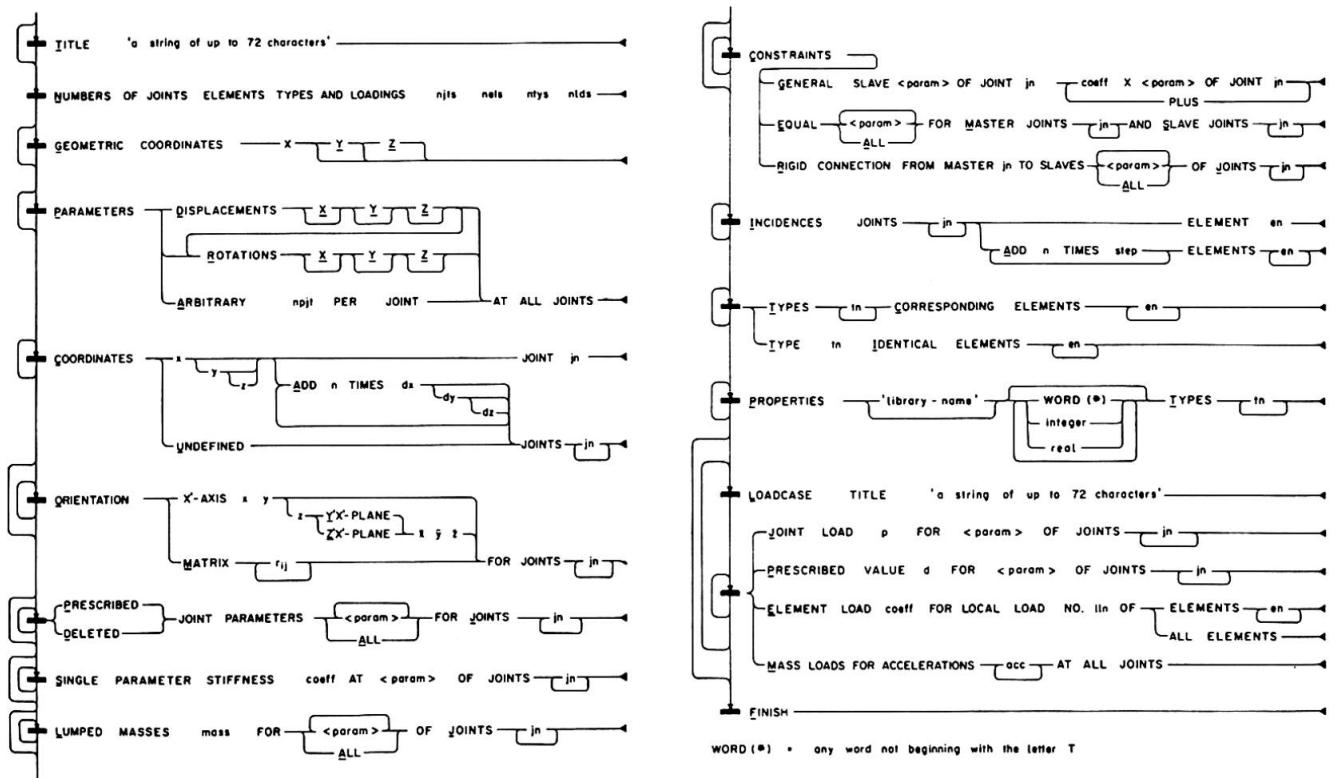


Abb. 1: FLOWERS: Syntaxdiagramm des Moduls SYSIN zur Eingabe von System- und Lastdaten

VII

The Role of C.A.D. in the Design of Major Structures

Le rôle du C.A.D. dans le projet de structures importantes

Die Rolle von C.A.D. bei der Projektierung wichtiger Bauwerke

P. BONALDI

Istituto Sperimentale Modelli e Strutture
ISMES
Bergamo, Italy

M. DI GAETANO

Istituto Sperimentale Modelli e Strutture
ISMES
Bergamo, Italy

A. PEANO

Istituto Sperimentale Modelli e Strutture
ISMES
Bergamo, Italy

R. RICCIONI

Istituto Sperimentale Modelli e Strutture
ISMES
Bergamo, Italy

SUMMARY

The design of major civil engineering structures, such as arch dams, can be drastically simplified and improved by the use of special purpose graphic packages. In fact in the very preliminary stage there is a need for preparing at a low cost and in a short time a good design in order to be able to closely estimate the cost of the construction (volume to be excavated, quantity of concrete required etc.). Since major civil engineering structures are not built routinely, the experience of the designer is obviously limited and a system for parametric analysis and optimization may be extremely useful.

RESUME

Le projet de structures importantes en génie civil, tels les barrages-voûte, peut être simplifié et perfectionné au moyen de programmes graphiques pour des applications spéciales. Dans une phase préliminaire, il est nécessaire de préparer, avec une bonne approximation, le coût de la construction (le volume des excavations, la quantité nécessaire de béton, etc.). D'autre part, les structures importantes en génie civil ne sont pas des constructions de routine, de sorte que l'expérience de l'ingénieur projeteur est naturellement limitée. Un système d'analyse et d'optimisation à l'aide de paramètres peut être extrêmement utile.

ZUSAMMENFASSUNG

Die Projektierung wichtiger Bauwerke wie Bogenstaumauern kann bei der Anwendung zweckorientierter graphischer Programmsysteme beträchtlich vereinfacht und verbessert werden. Es ist schon wichtig in den frühesten Projektphasen gute Entwürfe billig und kurzfristig realisieren zu können, um die Baukosten (Aushubmassen, erforderliche Betonmenge u.s.w.) genügend angenähert schätzen zu können. Da grosse Bauwerke oft nicht Routinearbeit des Ingenieurs und somit die Erfahrungen beschränkt sind, ist ein geeignetes System für Parameterstudien und Optimierungen sehr nützlich.



1. ROLE OF C.A.D. IN CIVIL ENGINEERING

Computers and programmes for automated analysis and design of structures are a powerful tool for the civil engineering profession. A recent IABSE Colloquium has discussed its role, critical aspects, potential and limitations [1].

It is however obvious that it is very difficult, if at all possible, to speak in general terms about the role of computer aided design and computer graphics in civil engineering: the field of application in mind has to be specified more strictly. In fact, at variance with general purpose packages developed in the sixties and early seventies, interactive design systems are highly application oriented. Moreover programs developed for very repetitive applications have very different aims than those oriented to one-of-a-kind structures. There is clearly a continuous spectrum which presents on one extreme the building industry where computer is often applied to bridge the gap between pilot project and final design. In this case the aim is to reduce time and manpower during, say, the design of reinforcement or the production of construction drawings. On the other extreme of the spectrum there are more impressive but less frequent civil engineering applications, like dams or containment structures of nuclear power stations. In this case interactive graphic system may be particularly useful in the very preliminary phase ranging from the conceptual study to the preliminary design where the aim is to establish the fundamental characteristics of the structure (say arch dam rather than earth dam) and to choose the most appropriate shape and dimensions.

At this stage very limited information is available about important design data, like rock foundation characteristics, and it is important to compare several alternative designs taking into account economic factors as well: the data used for comparison need not to be very accurate ones. Consequently, very elaborate and costly analyses can be ruled out because the expenses involved would be hardly justifiable. On the contrary there is scope for approximate, quick methods of analysis which can readily accommodate shape and size modifications, as well as parametrization of the ill-known properties. Still the method of analysis should be sensitive enough to detect significant changes in stress distribution induced by modifications of the structural shape or of the foundation deformability.

A critical assessment of the role and potential of this class of graphic system is made more clearly by discussing the particular case of the package GISFADD for analysis of arch dams [2]. The graphic system is based on a very efficient analysis method, a modification of the classical Ritter crown adjustment for arch dams. The modifications suggested by Fanelli take into account torsional and shear effects [3, 4].

Using GISFADD the designer may analyse and compare in a few hours a large number of design configurations (see Figure 1). The simplified method of analysis leads to a compact description of the structural behaviour and this also helps the designer intuition to grasp the

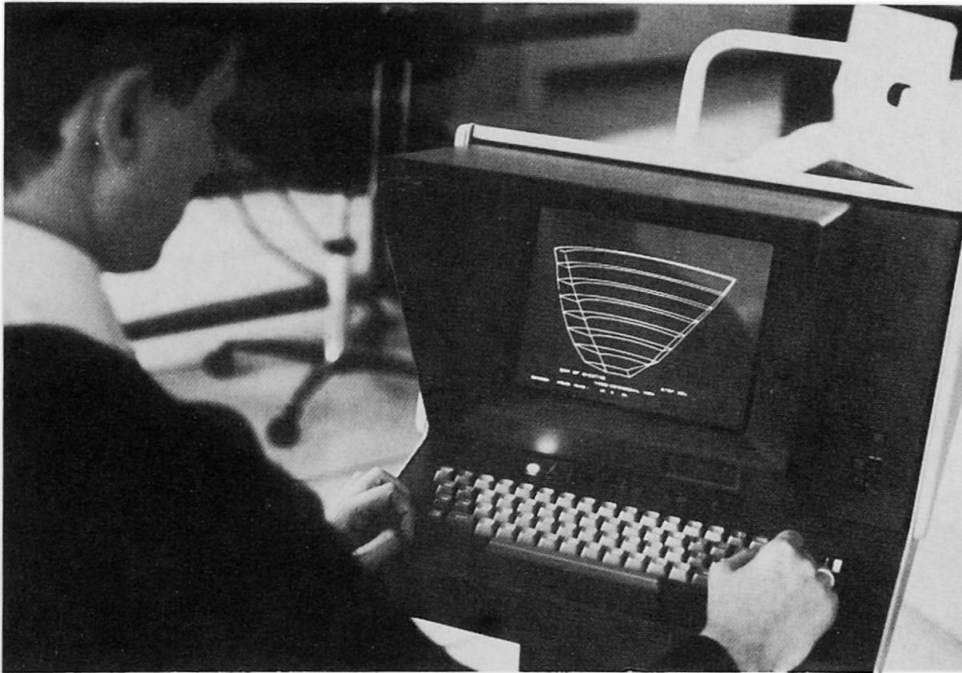


Fig.1 Graphic Interactive System for arch dam design.

influence of the different design parameters

The importance of this fact should not be underestimated. At variance with simpler usual structures, very limited learning experience is available to the designer of an arch dam as well as of others structures of the same class. In fact the number of structures of the same kind designed during a lifetime is very limited and moreover little can be learned by failures because they are fortunately very rare due to the high safety coefficient used for any important structure. For instance the total number of arch, arch gravity and double curvature dams built in Italy is 93 out of 485 large dams. However very few have been built during the last fifteen years. Since many of the old designers have now more administrative positions, fresh and available design experience is mainly based on dams designed for foreign countries. This scenario, which is probably common to other European countries, provides a motivation for the development of a tool like GISFADD that enables the designer to educate his intuition to a level accessible to very few individual in the past.

Potential users of GISFADD include regulatory bodies, which may need a quick tool for a first screening of dams designed a long time ago and universities for students attending, say, a Shell Design course. The importance of educating the engineering intuition has grown further in the computer era in view of the need of assessing the accuracy of automated stress analysis [5].

2. ASPECTS OF ARCH DAM DESIGN

Some salient points of the design procedure for arch dams are summarized here for reference in the following paragraph.

The topographical features of the valley are the point of departure for establishing a tentative design of the dam. Usually the designer bases his choice on a comparative analysis of previous designs of dams built in similar valleys and with analogous foundation conditions. Once a frame of reference is established, shape and dimensions of the geometrical elements of the dam, arches and cantilevers, must be defined. This is a long and difficult task because various parameters (radii, angles, thickness, etc.)

must be assigned compatible values so as to result in the design of a smooth shell of acceptable shape. The number of reciprocal geometrical constraints imposed by the various elements is very high and additional geometrical requirements arise from the shape of the rock foundation because the thrust of the arches may cause inordinate stresses in the abutments in the presence of an ill designed interface (Figure 2).

Moreover several parameters of a merely geometric kind, but with deep implications on the construction process and the cost of the structure, must be kept under control. The most important are the excavation volume and the concrete volume.

Static analysis of the dam used to be a formidable task before the computer era and many more or less simplified computing schemes were developed [3, 6, 7]. Still the difficulty of the computation was so high that an unsatisfactory design was more likely to be modified by the designer intuition rather than by a new complete cycle of the design process. Final judgment on the safety of the structure was made by means of a reduced scale model. Nowadays finite element models have come into play but still data

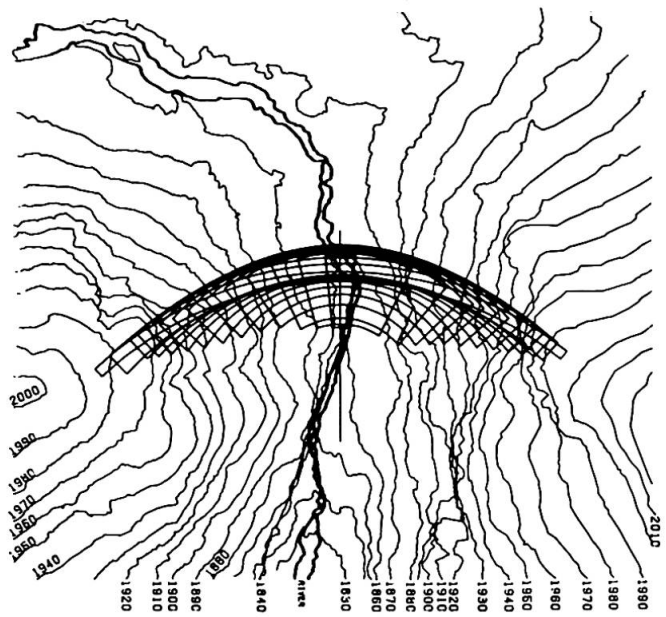


Fig.2 Valley contours and superposed arches of Chiota's dam.

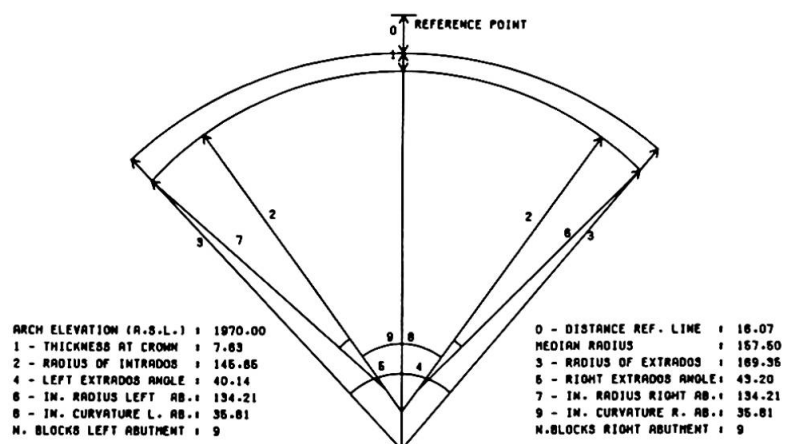


Fig.3 Geometrical input data of an arch.

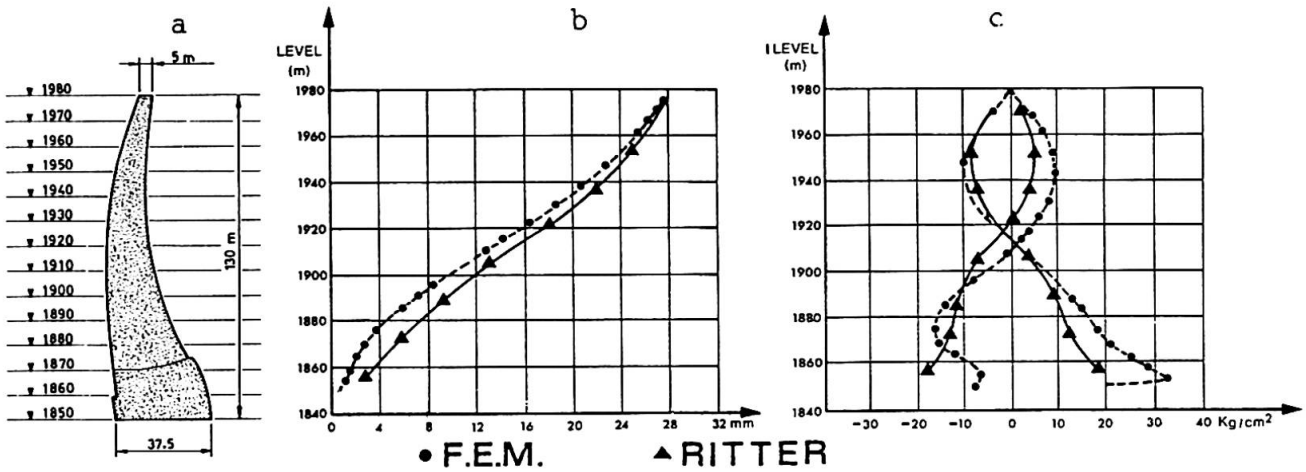


Fig.4 Comparison between results of FEM analysis and Modified Ritter's method for crown cantilever (a): deflections (b) and vertical stresses (c) under hydrostatic load.

preparation is so expensive that they are not used in preliminary design. On the other hand the old simplified analysis methods can be implemented in the computer and be used for a real optimization process. In addition to be fast and inexpensive, they provide the results in the format expected by the designer and this is a strong advantage over the majority of finite element programs. In fact a simplified method of analysis leads to the computation of stress resultants over the same elements of geometric idealization (arches and cantilevers) used by the designer to set up the model and provides a more direct indication of the remedies to be taken.

Again the graphic post processing capabilities of digital hardware are enormous. Previous considerations do not intend to underestimate the importance of more accurate methods of analysis, on the contrary a good package for preliminary design should have a capability for data output of a finite element discretization of the selected shape so that validation of the results may require little time and manpower. An outline of the role, potential and limitations of numerical analysis of concrete dams is given in reference [8].

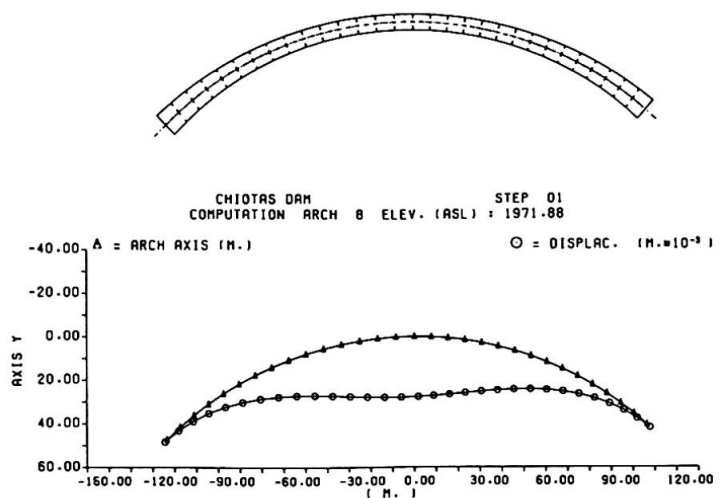


Fig.5 Deformation of central fiber of an arch.



3. INTERACTIVE DESIGN OF ARCH DAMS

In order to automate the design procedure the interactive system GISFADD has been developed at ISMES under support of ENEL/CRIS. The package may analyse arch, arch-gravity and double curvature dams. The main phases of the analysis are the following:

- Input of valley contours for visual checking of the static and geometric compatibility between abutments and rock foundation.
- Input of tentative dam geometry through standard design parameters such as upstream and downstream radii and thickness of arches at different elevations (Fig.3).
- Visual check of the geometry: cantilever sections, upstream and downstream developed surfaces, full perspective view of dam, etc.
- Input of loading data and material properties for concrete and rock.
- Analysis for all the basic design load conditions: hydrostatic load, dead weight, concrete shrinkage, seismic and thermal loads under

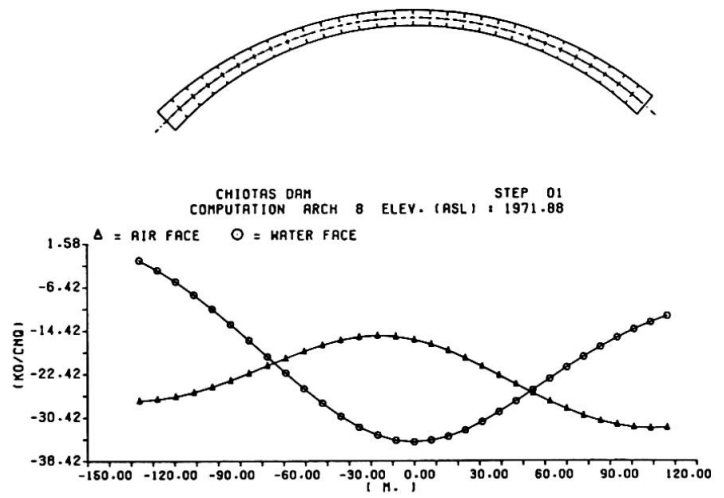


Fig.6 Upstream and downstream hoop stresses of an arch.

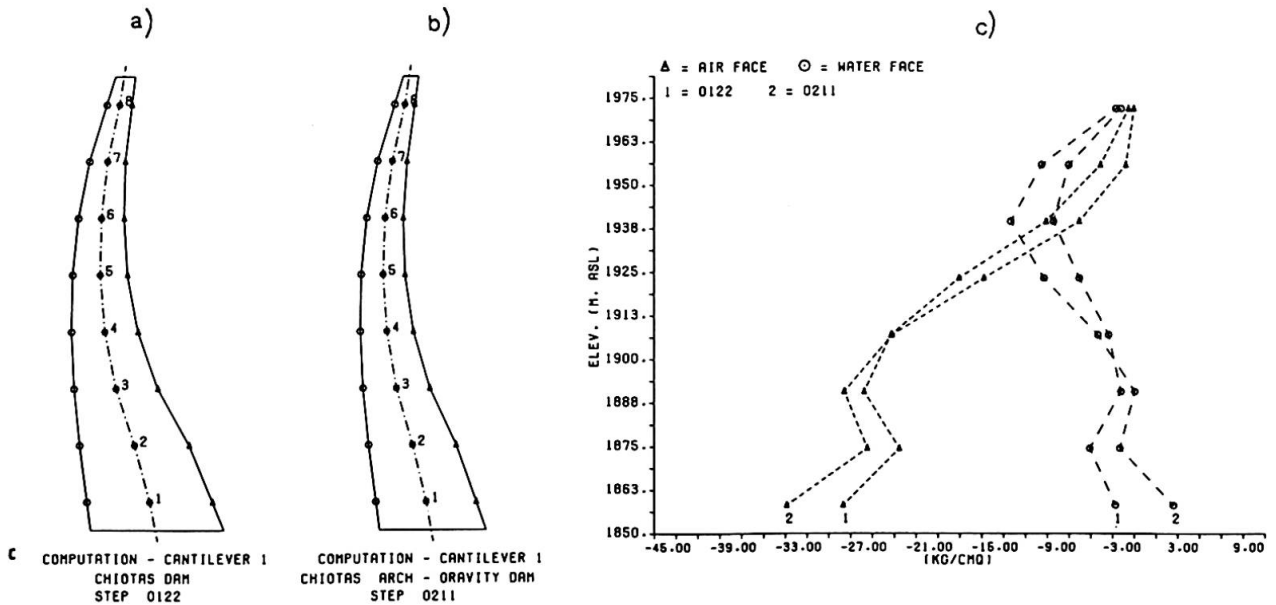


Fig.7 Comparison of vertical stresses (c) due to dead weight and hydrostatic load for two different designs (a) and (b).



both full and empty reservoir conditions. The reliability of the modified Ritter's method has been documented by a systematic comparison of the results obtained for a number of dams by means of the modified Ritter's method, of the finite element method and by physical models [4]. Some results for Chiotas dam are given in Fig.4.

- Deformations: radial and tangential displacements of arches and crown cantilever section (see Figure 5).
- Stresses: upstream and downstream hoop and vertical stresses for arches and cantilever section (see Figure 6).
- Comparison of results from current and previous designs for any load and load combination (see Figure 7).

4. CONCLUSION

Good design usually evolves through the experience gained by the performance of structures. This is typical of, say, building design but applies much less to the design of important one-of-a-kind structures. The reason is that the designer cannot rely much on past experience because of the limited number of structures of that kind available or because of the lack of performance data.

This leads to two conclusions. First important structures should be monitored and the data made accessible. Second once the basic phenomena governing structural behaviour are identified, it may be very useful to apply a graphic system for sensitivity analysis and design optimization. This conclusion has been supported by the case of arch dams, however the conclusions are likely to apply to other classes of structures as well.

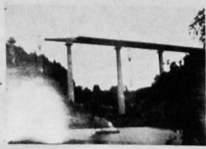
REFERENCE

- 1 IABSE Colloquium "Interface between Computing and Design in Structural Engineering" Bergamo, August 1978.
- 2 "GISFADD-2 Graphic Interactive System For Arch Dam Design: User's Manual" ISMES Technical Report 1980.
- 3 FANELLI M. "Il calcolo delle dighe a volta: Una modifica del Metodo di Ritter". L'Energia Elettrica, 1964, n.4.
- 4 FANELLI M., GIUSEPPETTI G., RABAGLIATI U.: "Calculation of Arch Dams: The Modified Ritter's Method" ISMES Publication n.108, Ottobre 1978.
- 5 MAIER G., PEANO A., "Educational and Professional Implication of Reliability Assesment in Computerized Structural Analysis" IABSE Colloquium, Bergamo, August 1978.
- 6 TOLKE F., LUDIN: "Wasser Kraftanlagen-Talsperren und Stau-mauern" Berlin, 1938.
- 7 U.S.B.R "Trial Load Method" Denver - Colorado, 1938.
- 8 BONALDI P., DI MONACO A., FANELLI M., GIUSEPPETTI G., RICCIONI R., "Concrete Dam Problems: an Outline of the Role, Potential and Limitations of Numerical Analysis" Proc. of 'International Symposium held at Swansea, U.K. 8-11 September 1975.

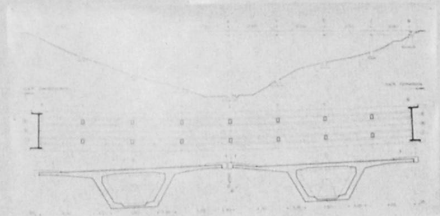
Leere Seite
Blank page
Page vide

Axhausen
Fink
Katz
Rank
Stieda
Verschuer
Werner

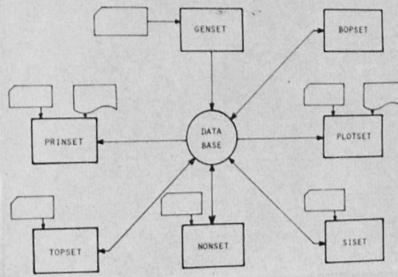
Programmkette S E T : Entwurfsberechnungen im konstruktiven Ingenieurbau
Prof. Dr.-Ing. Heinrich Werner - Technische Universität München



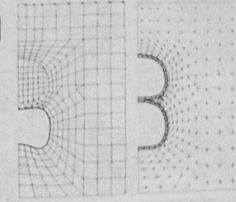
Brückenbau



Aufbau der Programmkette



Tunnelbau



GENSET: Generierung von Strukturen

- Knoten, Auflager, Elemente
- Querschnitte, Materialgesetze
- Lasten, Primärspannungszustände
- Generierungen, Fehlerprüfungen

NONSET: Lineare und nichtlineare statische Berechnungen, FE-Methode

- Nichtlineares Materialverhalten
- Große Verformungen
- Verfolgung der Bauzustände

TOPSET: Statik, Bemessung und Stabilität räumlicher Stahlbetonrahmentragwerke

- Statik nach der Theorie 2. Ordnung
- Bemessung, effektive Steifigkeiten
- Verfolgung des Baufortschrittes

Siset: Sicker- und Grundwasserströmungen

- Ebene, axialsymm., räumliche Probleme
- Lineare und nichtlineare Fließgesetze
- Freie Oberfläche mit Niederschlag
- Verknüpfung zu GENSET



Programmkette SET: Entwurfsberechnungen im konstruktiven Ingenieurbau

Heinrich Werner
Professor, Dr.-Ing.
Technische Universität
München, Deutschland

1. EINSATZMÖGLICHKEITEN DER PROGRAMMKETTE SET

Die Programme GENSET, PRINSET, PLOTSET und BOPSET sind problemunabhängig.

NONSET untersucht alle statischen Systeme, die folgende Elemente enthalten:

- Stäbe mit abschnittsweise veränderlichen Steifigkeiten (Rahmen, Fachwerke)
- Anisotrope ebene oder rotationsymmetrische Elemente (Scheiben)
- Federn und elastische Kopplungen
- Isotrope, isoparametrische Platten- und Faltwerkselemente (Platten, Schalen)
- Isotrope, isoparametrische räumliche Elemente mit 4 - 21 Knoten.

Folgende Nichtlinearitäten sind vorhanden:

- Große Verformungen für Fachwerkstäbe und ebene Elemente (z.B. Stabilitätsuntersuchungen an Gerüsten und räumlichen Fachwerken)
- Elastisch-plastisches oder nichtlinear-elastisches Materialverhalten bei Fachwerkstäben und ebenen Elementen (z.B. Plastifizierungen von Fachwerkteilen oder in geschichteten Böden)
- Schlupf, Vorspannung oder Abreißen bei Lagern und Kopplungen (Verbindungsmitel).

Das Programm TOPSET ist für Traglastuntersuchungen an Rahmentragwerken aus Stahl, Stahlbeton oder Spannbeton geeignet. Es berücksichtigt plastifizierte Bereiche mit vorgegebenen Arbeitslinien der Materialien.

2. VERFÜGBARKEIT DER PROGRAMMKETTE SET

Die Programmkette SET wurde bisher auf folgenden Maschinentypen installiert:

- CDC CYBER 175, Betriebssystem NOS 1.3
- IBM 370/125, Betriebssystem DOS/VS
- IBM 370/145, Betriebssystem OS/VS
- Siemens 7600 Betriebssystem Bs 2000
- Siemens 7.738 Betriebssystem Bs 2000

Das Programm Siset ist in Kürze einsatzbereit.

Der modulare Programmaufbau erfordert einen Kernspeicherbedarf von 128 KB. Der gesamte Datenverkehr zwischen den Teilprogrammen oder innerhalb eines Programmes erfolgt über die SET-Datenbasis. Ein in FORTRAN geschriebenes Datenverwaltungsprogramm DYNCO speichert die Daten dynamisch in freien Kern- und Massenspeicherbereichen.

Damit werden folgende Vorteile erzielt:

- Die Größe der zu berechnenden Probleme ist praktisch nicht begrenzt
- Optimale Ausnutzung der Kern- und Peripheriespeicherkapazitäten
- Einsetzbarkeit auch auf Minirechnern.

PROBLEM FORMULATION

$$Q = \begin{matrix} Q_1 \\ Q_2 \\ Q_3 \end{matrix} \quad \begin{matrix} S_1 \\ S_2 \\ S_3 \end{matrix} \quad \begin{matrix} Q_1 \\ Q_2 \\ Q_3 \end{matrix} \quad \begin{matrix} S_1 \\ S_2 \\ S_3 \end{matrix}$$

$s = \max \omega$
 $\varphi = A\lambda + B\omega + k$
 $\varphi^T \lambda = 0 : \varphi, \lambda, \omega \geq 0$

PLCP: Parametric Linear Complementarity Problem solvable by Linear Programming algorithm (Restricted Basis entry rule).

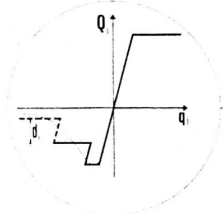
STEP-WISE SOFTENING
 Incremental formulation from POST-CRITICAL UNLOADING ($d =$ buckled bars):

$$\varphi_i = \varphi_i^* + d = 0 : \varphi_i^*, d \geq 0$$

$s = \max \delta \omega$

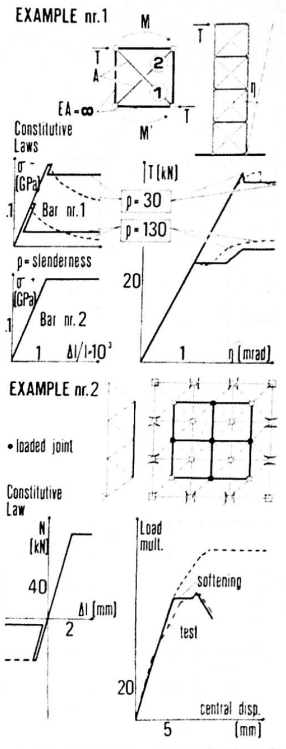
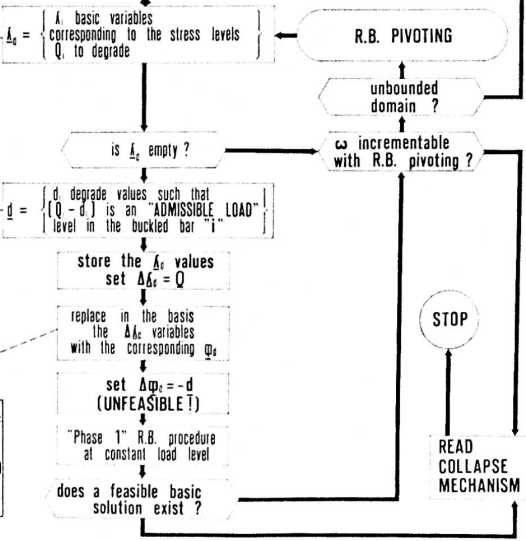
$$\begin{cases} |\varphi_i^*| - A \Delta \lambda - B \delta \omega - |d| \\ |\varphi_i^*| \\ |\varphi_i^*| = 0 : \Delta \lambda = 0 : |\varphi_i^*| \Delta \lambda = 0 \\ |\varphi_i^*| = 0 : \Delta \omega \text{ unfr.} : |\varphi_i^*| \Delta \lambda = 0 \end{cases}$$

BASIS	ω	φ	$\lambda (\Delta \lambda)$	R.H.S.
φ_i		T_i	T_i	φ_i
$\lambda_i (\Delta \lambda_i)$			T_i	$\lambda_i (0)$
ω				ω
z		c		z



BASIS	ω	$\varphi (\varphi^*)$	$\lambda (\Delta \lambda)$	R.H.S.
φ_i		T_i	T_i	φ_i
$\lambda_i (\Delta \lambda_i)$			T_i	$0(-d)$
ω				ω
z		c		z

FORMULATION OF TRUSS ELASTO-PLASTIC PROBLEM WITH STEP-WISE SOFTENING





COMPUTERIZED ELASTO-PLASTIC ANALYSIS OF STEEL RETICULATED PLATES

Mauro MEZZINA - Giuseppe PRETE - Antonio TOSTO
 Istituto di Scienza delle Costruzioni, Facoltà di Ingegneria
 Università di Bari
 Bari, Italy

In the structural collapse analysis special emphasis must be given to the knowledge of all the parameters that characterize the reached limit state. In fact this is the only instrument for evaluating the agreement in truth of such a limit state, that the specific mathematical model allows to simulate. In the special case of trusslike structures such a problem is of primary importance, due to the presence of "physically unstable" elements in the collapse mechanism. This event can invalidate the quality of the results obtainable with the "classical" limit analysis methods. In this context we propose an approach to the problem that, still remaining in the Mathematical Programming method field, arrange the solution of the evolutive elasto-plastic analysis of this kind of structures as a particular Linear Programming Problem.

The formulation starts, as indicated in [1], from the equilibrium, compatibility and stress-strain relationships for the whole structure in elastic-perfectly plastic behaviour: after some algebra it is possible to see that the collapse load factor is obtainable by the solution of the L.P. Problem shown in the Poster, where:

\underline{q} = bar elongations	\underline{Q} = bar forces
s = collapse load factor	ω = load multiplier
$\underline{\phi}$ = yield functions	$\underline{\lambda}$ = plastic multipliers
\underline{k} = yield limits	\underline{N} = diag {1 -1} = activable yield modes
$\underline{A} = - \underline{N}^T \underline{Z} \underline{N}$	$\underline{B} = - \underline{N}^T \underline{Q}^e$
\underline{Q}^e = elastic stress response to loads \underline{F}	\underline{Z} = influence matrix of self-stresses due to imposed strains

Such a formulation must be completed taking into account the post-critical behaviour, simulated in the $Q_i - q_i$ law with a series of constant-load steps, decreasing more and more up to the asymptotic level. The logical operations that must be performed from a "degrading" point are shown in detail in the depicted "flow-chart".

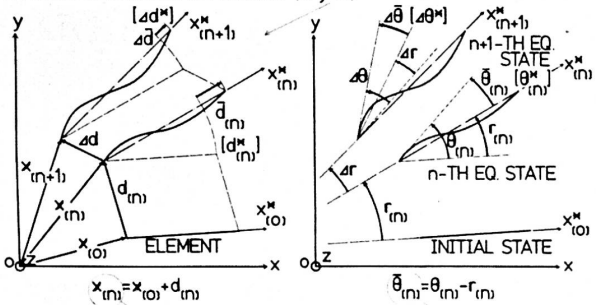
The first results of the numeric-experimental comparison, presently in effect, are very comfortable. From these applications we report here the most significant ones, according to us. In particular the Example n. 2 concerns the numerical simulation of a collapse test, performed on a prototype of reticulated grid. As it is possible to see, the calculated values are very close to the experimental ones both from a static and deformative point of view.

REFERENCES.

1. M. MEZZINA and A. TOSTO: Analisi elastoplastica di strutture reticolari in acciaio. Proc. of VII Congresso C.T.A., Torino, Italy, Oct. 1979.

DERIVATION OF ELEMENT EQUATION

1. COORDINATES x AND DISPLACEMENTS $u^T = (d^T \theta^T)$ IN GLOBAL COORDINATE SYSTEM (x, y, z)



2. DISPLACEMENTS $u^*T = (d^*T \theta^*T)$ IN LOCAL SYSTEM (x^*, y^*, z^*)
 $d_{(n)}^* = \Lambda_{(n)} d_{(n)}$ $\theta_{(n)}^* = \Lambda_{(n)} \theta_{(n)}$
 $\Lambda_{(n)}$: COORDINATE TRANSFORMATION MATRIX

3. INCREMENTAL NODAL DISPLACEMENTS IN LOCAL COORDINATE SYST. (FROM N-TH TO N+1-TH EQUILIBRIUM STATE)

$$\Delta u^* = u_{(n+1)}^* - u_{(n)}^* = \Delta T \begin{pmatrix} x \\ \theta \end{pmatrix}_{(n)} + T_{(n+1)} \begin{pmatrix} d \\ \theta \end{pmatrix} \quad ; T_{(n)} = \begin{pmatrix} \Lambda \\ \Lambda \end{pmatrix}_{(n)}$$

$$= \Delta T \begin{pmatrix} x \\ \theta \end{pmatrix}_{(n)} + T_{(n+1)} \begin{pmatrix} 0 \\ -\Delta r \end{pmatrix} + T_{(n+1)} \Delta u$$

4. FORCE-DISPLACEMENT RELATION IN LOCAL COORDINATE SYSTEM
 ASSUMPTION OF SMALL STRAIN
 $\Delta f^* = K^* \Delta u^*$
 K^* : CONVENTIONAL LINEAR STIFFNESS MATRIX

5. RELATION BETWEEN LOCAL AND GLOBAL NODAL FORCES

$$\Delta f = f_{(n+1)} - f_{(n)} = \Delta T^T f_{(n)}^* + T_{(n+1)}^T \Delta f^* \quad ; f_{(n)} = T_{(n)}^T f_{(n)}^*$$

6. FORCE-DISPLACEMENT RELATION IN GLOBAL SYSTEM [EQ. A]

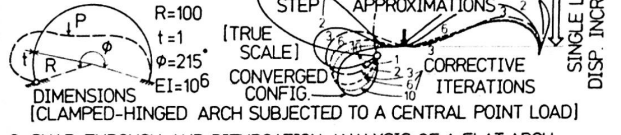
$$\Delta f = \Delta T^T \begin{pmatrix} f_{(n)}^* + T_{(n+1)}^T K^* \Delta T \begin{pmatrix} x \\ \theta \end{pmatrix}_{(n)} + T_{(n+1)}^T \begin{pmatrix} 0 \\ -\Delta r \end{pmatrix} \end{pmatrix} + T_{(n+1)}^T K^* T_{(n+1)} \Delta u \quad \text{KNOWN}$$

FUNCTIONS OF UNKNOWN INCREMENTAL DISPLACEMENTS Δu
 DEVELOP. OF A TWO STEP APPROX. & CORRECT. ITERATION SOL. PROC.

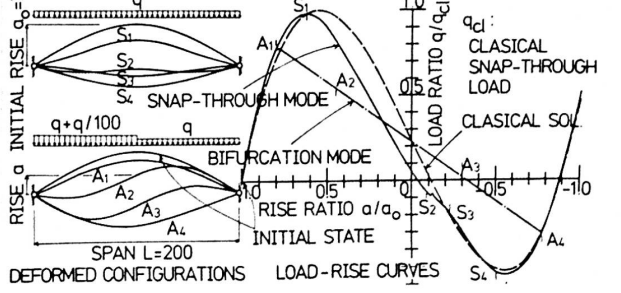
In the first two steps, EQ.A is linearized with respect to Δu , before the assemblage of element equations. Then in iterations, only Δu_{k+1} are treated as unknowns, others are estimated by using k-th approximation of Δu .

NUMERICAL EXAMPLES

1. CONVERGING PROCESS ILLUSTRATED BY DEFORMED CONFIGURATIONS



2. SNAP-THROUGH AND BIFURCATION ANALYSIS OF A FLAT ARCH (RISE/SPAN RATIO : 1/100) ONLY BY INCREMENTAL CALCULATION





A COMPUTER ORIENTED FORMULATION FOR GEOMETRICALLY NONLINEAR PROBLEMS

Yutaka YOSHIDA
Associate Professor
Tokyo Institute of Technology

Takeo NISHIWAKI
Professor
Musashi Institute of Technology
Tokyo/Japan

Ncbutoshi MASUDA
Lecturer

FORMULATION OF THE ELEMENT EQUATION

An incremental formulation is developed here, under the assumption of large displacement but small strain. The key point of the formulation is the description of nodal locations by the coordinates themselves instead of by mere displacements. A local coordinate system ($x^* y^* z^*$) is introduced which is at any time on the element itself, and the element force-displacement relation is assumed to be linear in this local system. The geometrical nonlinearity is considered only through the nonlinear transforming relations of displacement components between the space-fixed global coordinate system ($x y z$) and the element-fixed local coordinate system. The transformations are evaluated rigorously without neglecting small terms, so that incremental relations satisfy equilibrium conditions after deformation as accurately as possible.

SOLUTION PROCEDURE

To construct and solve structural equilibrium equations, an effective corrective iteration solution procedure is also originally developed. The process, outlined in the following, makes use of physical properties of each term in the derived element equation. At first, the changes of transformation matrix and rigid body rotation in the element equation, ΔT and Δr respectively, are linearized with respect to incremental displacement Δu at the n -th equilibrium state A. And the transformation matrix after the increment, $T(n+1)$, is estimated at the same state A. Hence the element incremental equation is written in the following quasitangential form.

$$\Delta f = (K_f + K_u + K)_A \Delta u$$

Then summing up the element incremental equations thus approximated for overall structure and solving them, the first approximation of the $n+1$ -th equilibrium state C is obtained. Secondly, ΔT and Δr are linearized at the midpoint B between A and C, meanwhile $T(n+1)$ is approximated at the above obtained first approximating state C. Thus the element incremental equation is written as

$$\Delta f = (K_f + K_u + K)_{B,C} \Delta u$$

Thus the second approximating solution, denoted by D, can be obtained. Then afterwards in iterations, ΔT and Δr as well as $T(n+1)$ are all evaluated by using the just preceding approximating solution, and only the incremental displacement Δu is treated as unknown variable. Namely, for the $k+1$ -th approximation

$$\Delta f - h_k = K_k \Delta u$$

is used, where

$$K_k = T_k^T K^* T_k, \quad h_k = T_k^T f^*(n) + T_k^T K^* (\Delta T \begin{pmatrix} x \\ \theta - r \end{pmatrix} (n) + T_k \begin{pmatrix} 0 \\ -\Delta r \end{pmatrix})$$

and

$$\Delta T = T_k - T(n), \quad \Delta r = r_k - r(n)$$

The iteration is continued until satisfactory convergence is obtained.

FEATURES OF THE PROPOSED METHOD

- With mere incremental calculations and without special techniques such as eigenvalue analysis, not only snap-through and limit point phenomena but also bifurcation can be pursued.
- Even for an extraordinary large increment, numerical stability exists and sufficient accuracy is obtained.
- As the consequence of the above characteristics, calculating time can be reduced considerably.

Buckling Analysis of Reticulated Cylindrical Shell Roofs

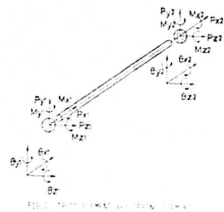
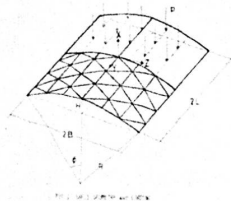
TOSHIYUKI OGAWA and TOSHIRO SUZUKI, Dr. of Eng., Tokyo Institute of Technology, Japan

SYNOPSIS

It is noted that instability failures such as snap-through are likely to occur in single layer reticulated shells as well as in true shells. Computer analysis approach in this field is usually done by the matrix method but there arise many difficulties due to large-scale degrees of freedom. In this study, we develop a simplified analytical method and apply it to the buckling analysis of single layer reticulated shells.

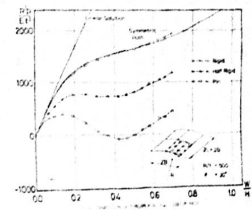
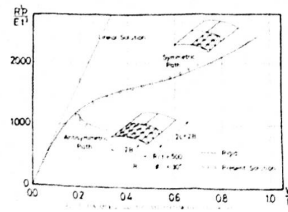
EVALUATION OF MEMBER STIFFNESS

As shown in Fig. 2, a reticulated shell is considered to be composed of a truss element with rotational springs at its edges. A truss element has only axial stiffness. Bending stiffness and joint rigidity is represented by these springs. Total member stiffness is obtained by the equilibrium method.



NONLINEAR BUCKLING BEHAVIORS

Fig. 3 shows fundamental equilibrium and post-bifurcation path of a rigidly jointed shell. Loading condition is shown in Fig. 1. Post-bifurcation path is pursued by giving small asymmetrical perturbation. Present solutions give good agreement with the solid lines obtained by the conventional FEM method. Fig. 4 shows the transition of fundamental path as joint rigidity decreasing. It is shown that snap-through occurs as decreasing of joint rigidity even when no snap-through exists on the fundamental path of perfectly jointed one. Although this is one example, this fact indicates the importance of joint rigidity in single layer reticulated shells. Required CPU time is much deduced by the simplified evaluation of bending stiffness.



Buckling Analysis of Reticulated Cylindrical Shell Roofs

Toshiro SUZUKI¹ and Toshiyuki OGAWA²¹Assoc. Prof., Dr.-Eng., Dep. of Architecture, Tokyo Inst. of Tech.²Research Assoc., Dr.-Eng., Dep. of Architecture, Tokyo Inst. of Tech.

Tokyo, Japan

SUMMARY OF THE DISCUSSIONS AND QUESTIONS

We received several questions at the "Poster Session". These questions and discussions are classified into the following

- 1) Iterative procedure in the incremental step, 2) Assembly of the element stiffness, 3) Accuracy of this method, 4) CPU time.

In this analysis, the unbalanced forces appear at each incremental loading step. As mentioned before, these forces are caused by the second-order terms in the axial strain expression. These forces are extinguished by the Newton-Raphson method. In this iterative procedure, the stiffness matrix is always recomposed. Accuracy of this analysis is depend on the validity of ignoring the terms related to initial inner moments. A uniformly loaded reticulated shell is considered to be near the membrane state. In other words, the inner moments are secondary compared with the axial force, and like a 3-hinged truss structure, there may be an equilibrium state by axial force. By these reason, we consider this analytical method effective when applied to the buckling analysis of uniformly loaded reticulated shells. The computing time is much deduced by the reason that solutions converge well and the required time of forming a stiffness matrix is deduced. In comparison with the required CPU time of the conventional FEM, this simplified method need 1/5 ~ 1/10 of that time. In addition to it, these calculations are done by Hitac M-180 which seems to have an equal ability with IBM's large computer. The assembly of the element stiffness is shown in the next section with the formulation process of a stiffness matrix [K].

FORMULATION OF A STIFFNESS MATRIX

A stiffness matrix of a truss element is lead by the virtual work principle. The finite element formulation of the equilibrium is shown in quadratic form of displacement vector $\{\Delta u_e\}$.

$$\begin{aligned} & \iiint [\Delta \sigma_{xx} \delta \Delta e_{xx}^* + \sigma_{xx} \cdot \frac{1}{2} \delta \left[\left(\frac{d\Delta v}{dx} \right)^2 + \left(\frac{d\Delta w}{dx} \right)^2 \right] - \Delta p_i \delta \Delta u_i] dv - \iint \Delta f_i \delta \Delta u_i ds - \delta \Delta W_r \\ & = \delta \{\Delta u_e\}^t \cdot [Ke] \cdot \{\Delta u_e\} - \delta \{\Delta u_e\}^t \cdot \Delta F_{ex} - \delta \{\Delta u_e\}^t \cdot R \end{aligned} \quad (1)$$

where, σ_{xx} = fiber stress, p_i = body force increment, and f_i = surface force increment. According to the arbitrariness of the virtual displacement $\{\Delta u_e\}$, a stiffness matrix $[K_e]$ of a truss element and equilibrium equation is obtained. Total flexibility matrix $[F]$ is obtained using flexibility matrix of spring elements $[F_1]$, $[F_2]$ and equilibrium matrix.

$$[F] = [H]^t \cdot [F_1] \cdot [H] + [F_m] + [F_2] \quad (2) \quad [F_m] = [K_e]_{jj}^{-1} \quad (3)$$

Total member stiffness matrix is obtained in inversed form of equation (2).

COMPUTER APPLICATION IN DESIGN AND CONSTRUCTION OF SPACE FRAME

AKIRA WADA: Senior Adviser, Nikken Sekkei Ltd.
 MAMORU IWATA: Dr. Eng., Nippon Steel Corporation
 DAJI NAKANO: Manager, Nippon Steel Corporation
 MORIHISA FUJIMOTO: Prof. Dr. Eng., Tokyo Institute of Technology

For the construction of large span structure, architects often use space frames. When an architect designs a space frame having a curved surface, the sorts of members and joints vary so greatly. Computers can be effectively used to construct complex space frames through accurate information management.

I DESIGN OF SPACE FRAME SHAPE AND GRID PATTERN

Space frames can be constructed in any shape desired.
 Definition of the space frame shape

Expanding

Space frame A

II STRUCTURAL CALCULATIONS

Computers can be used to present diagrams and figures, showing stress distribution and so on.

Examples of stress distribution

Space frame A

Space frame B

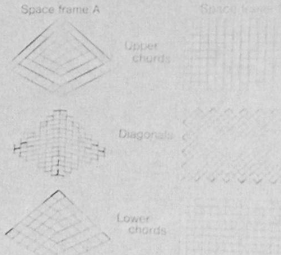
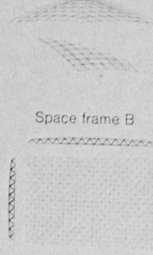
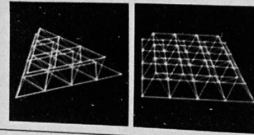
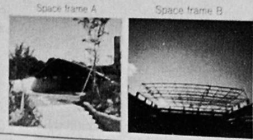
Upper chords

Diagonals

Lower chords

Examples of grid patterns

Space frame B



III DESIGN OF MEMBERS AND JOINTS

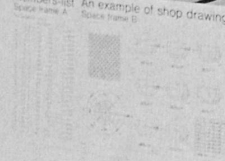
The preparation of lists and drawings of members and joints is easy if computers are applied.

An example of joint



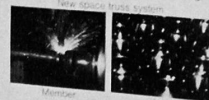
An example of members list

An example of shop drawing



IV MANUFACTURE OF MEMBERS AND JOINTS

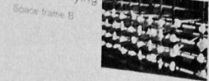
Members and joints are manufactured with a high level of quality control, according to the lists and the drawings.



V MARKING

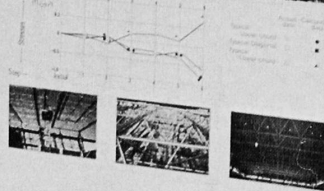
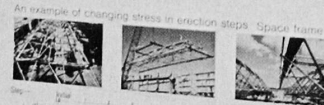
Members and joints are marked with identification numbers.

An example of classifying



VI ON-SITE ASSEMBLY AND ERECTION

Members and joints are assembled and erected to the designed shape according to the assembling lists. Computers can be also used for the safety analysis at the time of erection.



The flow from the design to the construction of space frames can be regarded as a process where a space frame designed as software at the initial stage is gradually converted into hardware through information management by computer, and eventually into a space frame as a physical entity.



COMPUTER APPLICATION IN DESIGN AND CONSTRUCTION OF SPACE FRAME

AKIRA WADA Senior Adviser Nikken Sekkei Ltd.	MAMORU IWATA Dr.Eng. Nippon Steel Corp.	DAIJI NAKANO Manager Nippon Steel Corp.	MORIHISA FUJIMOTO Prof.Dr.Eng. Tokyo Inst. of Tech.
	TOKYO	JAPAN	

BACKGROUND

In Japan, the construction of huge space structures, such as sports halls and exhibition halls, has recently become more widespread. Space frames are an important method for constructing steel structures having extensive column-free space. In space frames technology, computers have been used for structural analysis, however, computer can also be used for design and construction, from the standpoint of information management.

FURTHER COMMENTS

I. Design of Space Frame Shape and Grid Pattern.

When the plane on the nondimensional coordinates is divided by a grid, such as "Example of grid patterns", the space frame on the three dimensional coordinates is also divided by a certain rule based on the isoparametric shape function of the finite element method. The member lengths, the joint coordinates and the joint angles are automatically determined.

II. Structural Calculations.

"Examples of stress distribution" shows the magnitudes of stress by the thickness of the line of the members, the qualitative tendency of stress distribution is visually portrayed. The line in black indicates compression stress, while the line in white indicates tension stress.

III. Design of Members and Joints.

"An example of member list" shows the sequential number marked on a member, the number of joints at both ends, and the length of a member. "An example of shop drawing" shows the drilling positions of a joint.

IV. Manufacture of Members and Joints.

Manufactures make the members and joints, accurately observing each specification, with no idea as to where they are used in the space frame.

V. Marking.

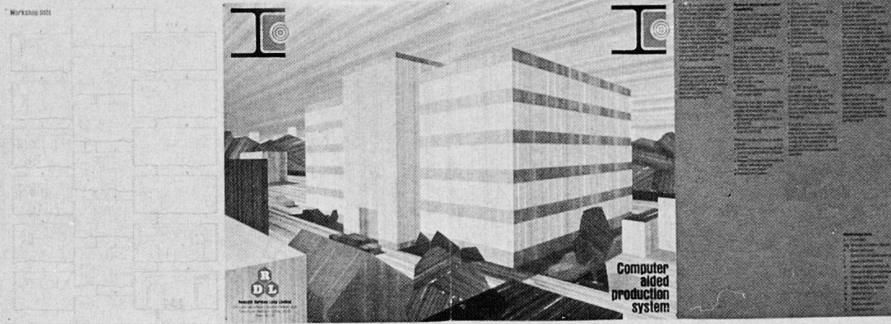
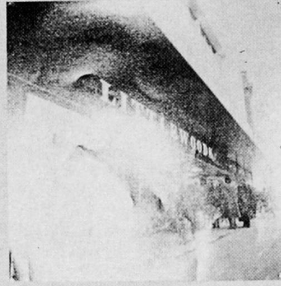
VI. On-Site Assembly and Erection.

Computers can be also used for the safety analysis at the time of erection. The figure shows the changing stress at each of erection and the calculation results.

FUTURE POSSIBILITIES

The concept of computer controlled information management is not limited to such examples and will become increasingly important to general building structures.

Littlewoods Store, Stoke on Trent, England—involved CAPS (Computer Aided Production System) detailing of 973 tonnes of structural steelwork.





CAPS - BASIC FEATURES

Some 18 man-years of research and development by RDL have gone into making the computer-aided detailing of beam and column structures a functioning reality. The benefits of this work are now available to the steelwork fabrication industry, consultants and other major users of structural steelwork.

CAPS is set up to handle beam and column structures of any geometry using a rationalised basis of design for all connections in accordance with British or National Standards.

All connections are designed automatically except where they exceed the design criteria and these are reported to the user.

The system works to a set of preferences, for instance bolt pitch and sizes, but it will adjust where necessary.

Users can easily specify their own preferences by simple commands and can select, for example:

- Bolts from 4.6, 8.8 or HSFG grades.
- Rolled sections, made-up section plate girders and also AUTOFAB.

Bolted and welded connection types available include:

- Bolted moment connections using top and bottom angles or tees.
- Welded moment connections with or without haunch.
- Double web cleats.
- Welded end plates.
- Splices, base plates.

CAPS will handle automatically:

- Eccentricity and skew, both horizontal and vertical.
- Extra fittings, etc., extra holes, shear studs, ledger angles, unusual stanchion bases, stiffeners.

CAPS will automatically check for:

- Bolt fouling.
- Notching.
- Adequate connection strength.

Users may nominate the erection sequence by giving the boundaries of 'blocks' in terms of grid lines and floor levels and the order of the blocks determining the erection sequence.

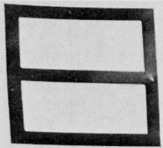
The output produced by CAPS is extremely comprehensive and includes the following schedules automatically:

- Cost estimates - materials and workmanship.
- Scheme drawings - marked and cross-referenced.
- Detail drawings - with piece marks, standard symbols and an index.
- Material lists.
- Cutting and drilling lists.
- Shop bolt lists.
- Site bolt lists.
- Erection sequence lists.
- Numerically controlled tool tapes.

All CAPS operations are carried out in a fraction of the time they would take by traditional methods.

For further information please contact:

BUROTEC, 53 Goldington Road, Bedford, England.



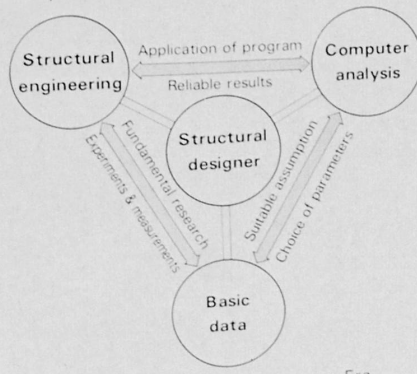
NIKKEN SEKKEI
planners architects engineers
JAPAN

Harmonious Development of Computer Analysis and Structural Design

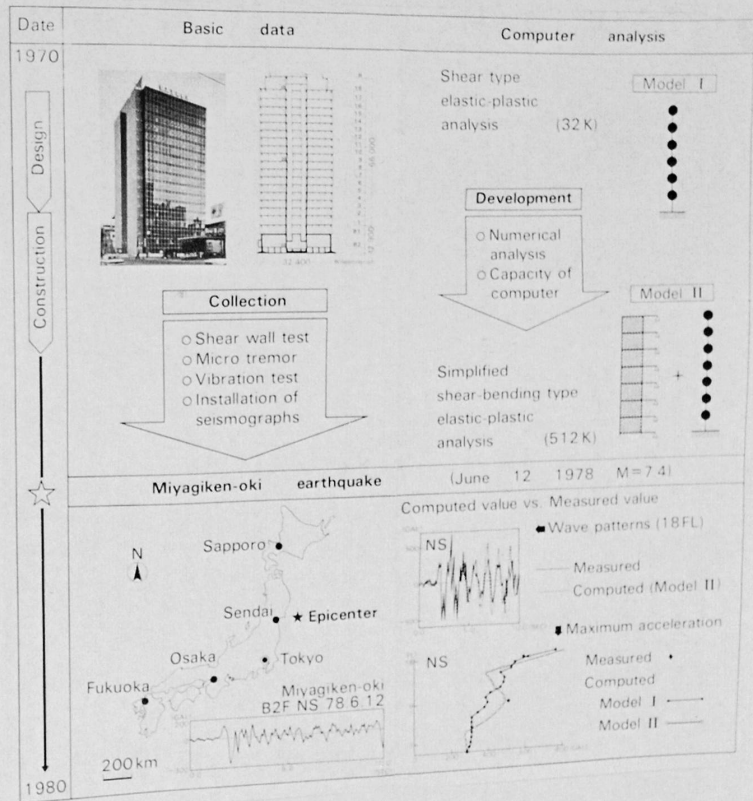
Dynamic Analysis of a Tall building under Strong Earthquakes

T. AOYAGI N. UCHIDA H. KIRIHARA
K. MATSUNAGA M. KAWAMURA

A computer plays the following two major roles in the field of structural design. One is the fast and accurate processing of data - a task which involves enormous volume of work but requires only simple thinking. The other is the computer's aid in theoretical analysis of structure, which has tremendously expanded the domain of numerical analysis. Future development of structural engineering depends, to a great extent, on the pursuit of theoretical analysis through understanding of the computer's roles, and at the same time, on the storage of basic data, both requirements having to be advanced in a well-balanced state. From this viewpoint, the authors introduce here their ten year experience obtained from the design through simulation analysis for a tall building (in Sendai city) which encountered strong earthquakes.



Structural design in Computer Era





" HARMONIOUS DEVELOPMENT OF COMPUTER ANALYSIS AND STRUCTURAL DESIGN "

Tsukasa AOYAGI
Chief, Struc. Dept.
 Koichi MATSUNAGA
Struc. Engr.

Naoki UCHIDA
Senior Struc. Engr.
 Masayoshi KAWAMURA
Eng. D., Chief Advisory Struc. Engr.

Hideaki KIRIHARA
Struc. Engr.

Nikken Sekkei Ltd, Osaka, Japan

Basically, the authors' poster is separable into two segments: a theoretical vision reflecting a structural engineer's roles in the computer era; and a typical example of such correlative event and study as envisaged in that vision. Thus, the following two became the main subjects of discussion with the audience.

Discussion about "Structural Design in Computer Era"

Most of audience favorably agreed to this vision, with the related discussion covering the following.

- 1) Development of Program: At the authors' firm, the typical way of developing computer programs is that related engineers, say, those at the computer department and the structural department associate themselves with each other by establishing an adhoc task force on a specific project. When this was explained to the audience, most of them admitted that this approach was the most preferable way because the structural engineers' participation in programming would certainly make the program easier to operate and more effective, but some audience commented that this approach was difficult to follow in their country because of professional separation of analysis from design in practice.
- 2) Operation of Program: The authors explained that they used their own computers installed for in-house use, viz. two machines of IBM 370-M138 and associated terminals. Besides these machines, small computers of about 100 K are provided at the structural departments for simple calculation and analysis. Though all these programs being developed for in-house use, two major programs for integrated structural analyses as authorized by the Minister of Construction are made available to outside structural engineers under licensing agreements. Such positive attitude of the authors toward computer use was favorably accepted by the audience.

Discussion about "Typical Example (Dynamic Analysis of a Tall Building under Strong Earthquakes)"

The practitioners for whom seismic force is an indispensable consideration made wide-ranging inquiries, from aseismic design methods in Japan to assumption of coefficients for the analyses, eventually being followed by rewarding discussion. They regarded this example as very rare and invaluable study which certainly helped to provide a basis for the harmonious development of computer analysis and structural design.

Thus, the authors acknowledge that the poster session afforded the authors with an otherwise unobtainable chance for better communication among participants to have them thoroughly understand the authors' philosophy. With their attitude so successfully supported, the authors wish to develop further study.



Conclusions

M. FANELLI

Professor

ENEL

Milan, Italy

In this seminar we heard many interesting papers and had good discussions. In this respect the seminar was successful.

I would also like to mention the Poster Session on the use of computers. It was a new format for IABSE and it worked out quite well. In this format of presentation, people are probably more encouraged to contact the authors (and to ask for more details on their work) than they would be in a normal session, in front of a large audience. I think that the idea showed its worth and that it should be adopted again in future Congresses.

It was extremely interesting to hear the lecture on Computer Graphics by Professor Greenberg and to see the illustrative material.

In conclusion, I would like to point out that we originally asked for contributions not so much of a technical nature (on new computer programs, new methods of analysis etc.) as of a critical/organizational nature, in order to ease the transition of structural engineering into the computer era. We were only partially successful in obtaining the right kind of contributions.

However, I made quite clear the main problems that a structural engineer should have in mind in his use of computers as everyday tools:

- 1) Organizational problems: choice of hardware and software adequate to the technical tasks to be performed.
- 2) Information problems:
 - Input data certification, in relation to aims and stage of design;
 - Mathematical model certification, and
 - Software certification, as distinct from each other;
 - Result certification;
 - Structure behaviour follow-up and interpretation in the light of the correspondence of the design to the built structure.
- 3) Integrated management problems: how the structural engineer can tie up in the most efficient and painless way into a management system that is growing increasingly computer-orientated in all stages of the carrying out of a project, and not only as far as technical matters are concerned, but also in planning, accounting, logistics, coordination, economics etc.



These problems are already here to stay. They will not be solved in a short time, but the next generation will have a solution and will have to live with it. It is in the interest of our profession to think out a solution that is safe and sound rather than have it imposed upon us from external agencies. I would even go further and say that it is vital to the civil engineer to take the lead now, if the evolution of the computer is to serve his tasks in the most beneficial way.

Thank you for your attention.