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

Band: 11 (1980)

Artikel: Ein Programmier-System anstelle von Programmpacketen?

Autor: Hartmann, Bernd / Beyer, Rolf

DOI: https://doi.org/10.5169/seals-11328

Nutzungsbedingungen

Die ETH-Bibliothek ist die Anbieterin der digitalisierten Zeitschriften auf E-Periodica. Sie besitzt keine Urheberrechte an den Zeitschriften und ist nicht verantwortlich für deren Inhalte. Die Rechte liegen in der Regel bei den Herausgebern beziehungsweise den externen Rechteinhabern. Das Veröffentlichen von Bildern in Print- und Online-Publikationen sowie auf Social Media-Kanälen oder Webseiten ist nur mit vorheriger Genehmigung der Rechteinhaber erlaubt. Mehr erfahren

Conditions d'utilisation

L'ETH Library est le fournisseur des revues numérisées. Elle ne détient aucun droit d'auteur sur les revues et n'est pas responsable de leur contenu. En règle générale, les droits sont détenus par les éditeurs ou les détenteurs de droits externes. La reproduction d'images dans des publications imprimées ou en ligne ainsi que sur des canaux de médias sociaux ou des sites web n'est autorisée qu'avec l'accord préalable des détenteurs des droits. En savoir plus

Terms of use

The ETH Library is the provider of the digitised journals. It does not own any copyrights to the journals and is not responsible for their content. The rights usually lie with the publishers or the external rights holders. Publishing images in print and online publications, as well as on social media channels or websites, is only permitted with the prior consent of the rights holders. Find out more

Download PDF: 02.07.2025

ETH-Bibliothek Zürich, E-Periodica, https://www.e-periodica.ch

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 language 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 beigefügten 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 paralelle 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 Minicomputer-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 addressierbaren 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 jeoch 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.

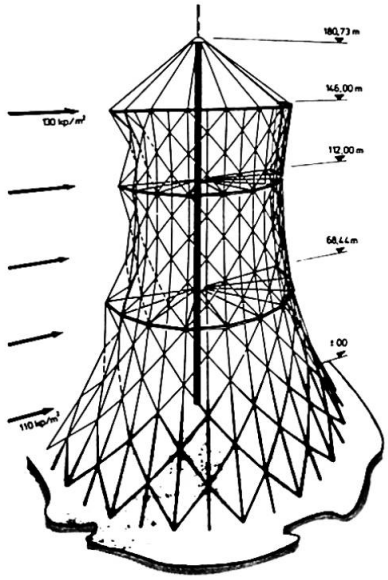

Bild 1:
Seilnetzkühlturm unter Windlast

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 unbekannten 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.

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.

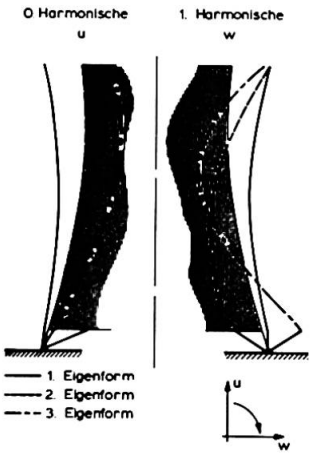

Bild 2:
Stahlbetonkühlturm, unterste Eigenform

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

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 Koppeln fertiger Programmbausteine auf den Leib zu schneidern.

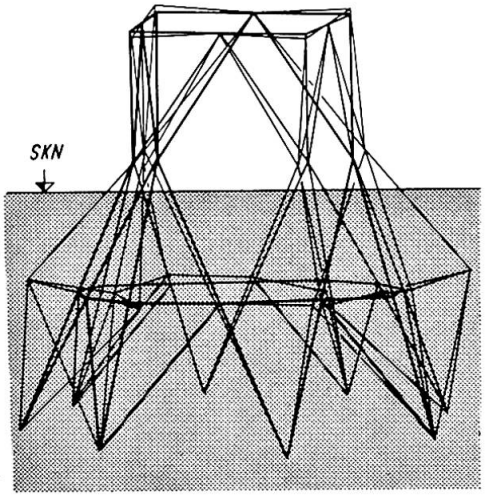

Bild 3:
Forschungsplattform Nordsee
5. Eigenform, f = 2,85 Hz

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. 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

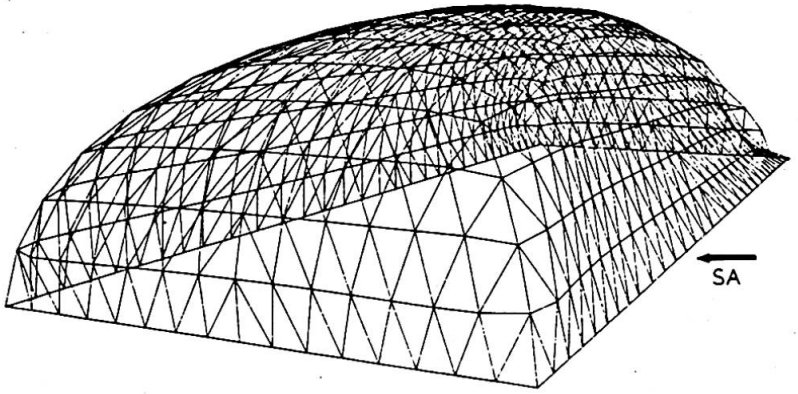

Bild 4: Traglufthalle unter Windlast

oder auch nicht periodischer Erregung.

Stellvertretend für die vielen Tragwerksuntersuchungen seien hier einige angeführt. Bild 1 zeigt in 10-facher Vergrößerung die Deformation eines Seilnetzkühlers unter Windbelastung, während Bild 2 die untersten Eigenformen einer longitudinal und transversal schwingenden Kühlturmschale mit Unterstützungsfachwerk 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 dreistöckiger 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.

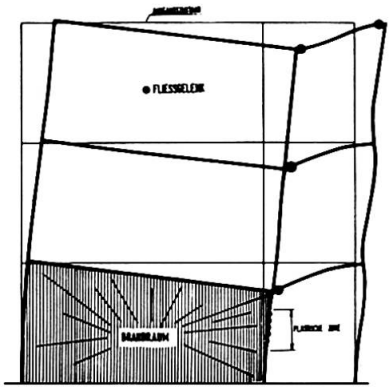

Bild 5:
Stockwerkrahmen, Versagensmechanismus unter Brandbelastung

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 problemorientierten Sprache. Bibliographisches Institut, Hochschulskript 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.