Cleve Moler: Mathematical Software Pioneer and Creator of Matlab

Thomas Haigh

➡ For additional information about this article
https://muse.jhu.edu/article/235247

Cleve Moler: Mathematical Software Pioneer and Creator of Matlab

Thomas Haigh
University of Wisconsin

Cleve Moler was born in 1939 to a family of journalists, and grew up in Salt Lake City, Utah. A distinguished student, he developed an early interest in science and mathematics, pursuing technical hobbies such as ham radio. In 1957, just prior to the launch of Sputnik, he entered the California Institute of Technology as an undergraduate. Despite his scientific acumen, Moler retained his family's interest in writing, serving as editor of the college newspaper. Though he eventually majored in mathematics, ostensibly because this gave "the most options and fewer required courses," Moler was already demonstrating the interdisciplinary interests and commitment to the broad dissemination of technical knowledge that were to characterize his entire career.

A Caltech class with John Todd, then recently arrived from the Institute for Numerical Analysis, provided Moler an introduction to electronic computing as well as the fundamentals of numerical analysis: Todd's students were able to sign up for time on the institute's Datatron 205. Moler excelled in this class, discovering a love for computers and for programming, yet he realized that he was not, by Caltech standards, an outstanding theoretical mathematician. During his senior year, he worked on an independent project with Todd, including a machine language program to invert Hilbert matrices.[1]

On Todd's recommendation, Moler followed graduation from Caltech with a move up the coast to Stanford University as a graduate student in mathematics. He was drawn by George Forsythe, then in the process of establishing computing as a distinct area of graduate study. Although this was initially done as a specialization within the mathematics program, by the end of Moler's time at Stanford the effort had produced one of the nation's first (and most distinguished) departments of computer science. Like many computation specialists of the 1950s and early 1960s, Moler was forced to prove himself within a discipline to which "numerical analysis has always been the black sheep." He was "as applied as you could get and still be in the math department," and narrowly satisfied its broader requirements despite what he recalls as a "lack of ability in many of the standard topics of a classical mathematical education."[2] His talents in computing and applied mathematics, however, made Moler a valued member of the emerging computing department. He taught a course on numerical analysis designed by Forsythe, refining and expanding it. Their collaboration formed the basis for the textbook *Computer Solutions of Linear Algebraic Systems*, widely recognized as a classic work in its field.[3] The book included a selection of example subroutines for the solution of systems of simultaneous linear equations, and introduced Moler to the publication of software.

For his thesis, Moler extended earlier work by Forsythe on the use of finite difference methods to calculate upper and lower bounds for the eigenvalues of Laplace equations in

---

Background of Cleve Moler

Born: 17 August 1939, Salt Lake City, Utah.

Education: California Inst. of Technology, BS (mathematics), 1961; Stanford Univ., MS, (mathematics), 1963; PhD (mathematics), 1965. **Professional experience:** *Stanford Univ.*, Computer Science Dept., instructor, 1965; visiting associate professor, 1970–1971; visiting professor, 1978–1979; visiting professor, 2002. *Eidg. Technische Hochschule*, Zurich, Switzerland, ONR research associate, 1965–1966. *Univ. of Michigan*, Mathematics Dept., assistant professor, 1966–1970; associate professor, 1970–1972. *Univ. of New Mexico*, Mathematics Dept., professor, 1974–1982; associate professor, 1972–1974; Computer Science Dept., professor and chairman, 1980–1984. *Intel Scientific Computers*, Beaverton, Oregon, manager, applications research, 1985–1987. *Ardent Computers*, Sunnyvale, California, vice president, scientific software, 1987–1989. *The MathWorks Inc.*, chairman and chief scientist, circa 1989–present. **PhD students:** Norman Schryer, Michigan, 1969; Alan K. Cline, Michigan, 1970; Charles Crawford, Michigan, 1970; David Kammler, Michigan, 1971; Stanley Eisenstat, Stanford, 1972; Linda Kaufman, Stanford, 1973; Charles Van Loan, Michigan, 1973; Charles Burris, New Mexico, 1974; James Sanderson, New Mexico, 1976; John Starner, New Mexico, 1976; George Davis, New Mexico, 1979; Jack Dongarra, New Mexico, 1980; Ronal Jones, New Mexico, 1985; Augustin Dubrulle, New Mexico, 1986; Humberto Madrid, New Mexico, 1986.
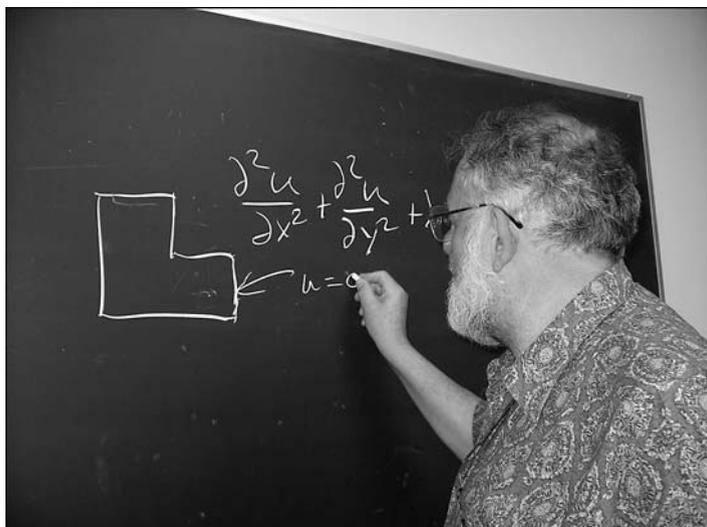

Figure 1. Moler draws the L-shaped membrane and its defining equation. During his PhD thesis under the supervision of George Forsythe, he worked to analyze its vibrations. The vibrating membrane became a visual motif for further work by Moler and his colleagues during the 1960s and 1970s. Since the 1980s, it has provided the basis for the MathWorks' logo, and a favorite example of the effectiveness of Matlab's programming and visualization tools.

two-dimensional domains.[4] His main example was an L-shaped membrane (see Figure 1), a shape for which he developed a considerable fondness and which had earlier been used as a test case by celebrated British numerical analyst Jim Wilkinson. Moler later chose it as the MathWorks corporate logo, and celebrated it in such unlikely forms as a Christmas tree ornament and, at a party for his 60th birthday, an ice sculpture.

Academic career

After obtaining his PhD, Moler spent a year at ETH, the Swiss Federal Institute of Technology in Zurich. ETH housed an important computing group, with close ties to the Institute for Numerical Analysis in California, and Moler's postdoctoral fellowship was a specialized program administered by the US Office of Naval Research. Here he pursued his work on the L-shaped membrane, devising a new mathematical method in conjunction with Leslie Fox and P. Henrici.[5] On his return the US, he was recruited by Bernie Galler to a numerical analysis faculty position in the mathematics department at the University of Michigan, Ann Arbor.

Moler rapidly became a prominent figure within the numerical analysis community. One of his responsibilities at Michigan was the organization of an annual series of summer courses in numerical analysis. While the students were mostly from research groups within aerospace and automotive firms, the instructors included most of the leading figures in numerical analysis, including Alston Householder and Jim Wilkinson. The series formed a social hub for the community, where established instructors swapped ideas with each other and with younger colleagues. Moler also played an active role in the ACM numerical analysis interest group, SIGNUM, where he served as newsletter editor and on the board of directors during the early 1970s. However, he was frustrated by the lack of autonomy the group enjoyed with the ACM, instead devoting himself to the Society for Industrial and Applied Mathematics. From 1973 to 1976 he was editor in chief of the *SIAM Journal of Numerical Analysis*.

Moler remained at Michigan until 1972, before moving on to a post at the University of New Mexico. During his time on the Michigan and New Mexico faculties, Moler's research interests gradually shifted from the more theoretical aspects of numerical analysis toward mathematical software, then viewed by funding bodies such as the Office of Naval Research as a less respectable area for academic investigation. However, his involvement in the numerical analysis community had brought him into contact with the staff of the Argonne National Laboratory, and in particular with the team working on the EISPACK project.[6] EISPACK was designed to produce a high-quality, well-documented, and portable Fortran library of routines implementing the matrix eigenvalue algorithms created by Jim Wilkinson and presented in high-quality Algol implementation by Wilkinson and Reinsch.[7] Wilkinson himself was a regular summer visitor to Argonne. Moler worked to test and evaluate preliminary versions of this software, and started spending some or all of his summers at Argonne. He was one of the authors of the commercially published EISPACK user guide.[8] He contributed a new routine to the third version of EISPACK, working with Burt Garbow to incorporate the QZ algorithm for generalized eigenvalue problems he had developed with G.W. (Pete) Stewart.[9] Moler then played a central role as one of four primary contributors to Argonne's LINPACK project, a follow-up to EISPACK designed to create a complementary library covering other kinds of matrix computation functions.[10]

Moler stayed at New Mexico for 12 years, first in the Mathematics Department and then as chair of the Computer Science Department.

As chair, he helped to build the department during a period of rapid growth. His graduate students there included several prominent figures, among them Jack Dongarra, who assisted with the EISPACK and LINPACK projects, later emerging as major figure in supercomputing. While continuing his close involvement with Argonne, Moler also developed a relationship with the computer center at Los Alamos National Laboratory, providing advice on its mathematical software library. Retaining his interest in pedagogy, he co-authored a second textbook, based on a course developed by George Forsythe, to show science and engineering students how to solve problems with the aid of standard software packages and libraries.[11] The book was accompanied by a set of usable, easy-to-understand subroutines, distributed on punched card or tape. This blending of education, mathematical theory, and high-quality software was unusual and effective.

Around 1976, he began work on another educational software package. Matlab was a simple matrix calculator, offering a small subset of the capabilities of EISPACK and LINPACK. Moler recalls that "you could invert matrices and compute determinants, and compute their eigenvalues, and that was about all." However, the nonspecialist students in this introductory course found the need to write, and compile, Fortran programs in order to call LINPACK functions a major hurdle. They were too distracted by the rituals of programming to focus on the mathematics. In contrast, Matlab worked interactively using terminals connected to a minicomputer. No programming was required (or even possible), and users could experiment with matrix operations while receiving instant feedback. This required Moler to write a simple parser based on Wirth's demonstration PL0 language (quite a challenging thing to do in the Fortran of that era, given its lack of support for recursion), and to create portable input and output code able to work on a variety of different systems. He was also influenced by APL, an exotic, interactive mathematical language, and a program called Speakeasy developed at Argonne. The first public presentation of Matlab appears to have been in February 1979, when Moler and Stewart co-taught a SIAM Short Course on Matrix Methods in Scientific Computing at the Naval Postgraduate School in Monterey, California. But during this period Moler treated Matlab as a hobby, a welcome diversion from his duties as department chair. Moler made only two formal presentations of this system at academic conferences.[12] The first was to the 1980 National Computer Conference. He recalls that many in the audience left disappointed, having come to hear about matrix management, a then-fashionable technique wherein employees reported to more than one boss. By the early 1980s, several hundred copies of Matlab had been distributed, for use by the instructors of introductory mathematics courses.

Work in industry

During his years in New Mexico, Moler worked as a consultant on several projects to produce mathematical libraries for the new generations of vector architecture and parallel processing computers. One was for IBM and another for Convex (then creating a "minisupercomputer" with a similar architecture to the Cray 1 but at a much lower cost). In 1985, he left academia to perform a similar role full time at Intel, which was then creating a supercomputer subsidiary to build powerful machines out of large numbers of microprocessors working in parallel. Moler recalls the main discovery of his team as being how much harder it was to write software for such a machine than anyone had expected. He spent a lot of time promoting the idea of massively parallel computing within the scientific community, and ran a group that was more like an academic research department than an industrial software development team.

After two and a half years, he moved to Silicon Valley start-up Ardent, then one of the hottest and best-funded new computer companies in industry history. Ardent's aim was to produce an extremely high performance personal workstation, using supercomputer technology and a parallel vector architecture. Although the firm boasted the efforts of a star team of computer scientists, exceptional graphical capabilities, and a high-performance version of LINPACK, by 1989 Ardent had entered a terminal spiral.

By this point, however, Matlab had been reborn as a commercially successful product. Moler traces the roots of this to a sabbatical year he spent at Stanford in 1979 and 1980, in which he used Matlab as a pedagogical tool. Although computer scientists in the group were unimpressed, their classmates studying electrical engineering found Matlab extremely useful for control theory and signal processing applications. Some brought the program with them to the firms they worked for. The first commercial Matlab derivatives were produced by spin-off firms from Stanford University's

electrical engineering department. One of these products, Matrix-X, enjoyed considerable success. The other, Control C, was produced by a systems engineer named John Little, working for a company called SCT.

The arrival of the personal computer opened new opportunities for ordinary scientists and engineers to use interactive computers as a part of their daily work. In 1983, Little approached Moler at a conference to suggest that they collaborate on a commercial version of Matlab, for the new and extremely popular IBM PC. Moler received equity in the new venture, MathWorks, but was not involved in its day-to-day operations. Little rented a cabin, purchased an early Compaq portable, and worked with a colleague, Steve Bangert, to reimplement Matlab in C (including the portions of code from LINPACK and EISPACK). They added a programming language and, in a later release, a "toolbox" capability to allow the creation of new packages of functions for specific application areas. Matlab retained its focus on matrices, which were initially its only data type. The commercial version of Matlab was introduced to the world in Las Vegas, during the Automatic Control Conference held in December 1984. The first sale, early the next year, was to Nick Trefethen of the Massachusetts Institute of Technology. Versions for the DEC VAX and for Sun Microsystems workstations soon followed. Matlab 3, released in 1987, included graphical capabilities.

At first, Little and Bangert worked from their homes, and early copies of Matlab were duplicated manually on Little's computer. Sales started slowly, but grew steadily. Little started to draw a salary, and Bangert came to work for the firm full time. Additional programmers, a business manager, office staff, and a marketing specialist followed. Until 1988, distribution and sales work was handled by a small firm called MGA Inc. Moler recalls a visit during 1988 when he walked into and, stunned by the sight of a conference room and well-stocked office supply room, thought "My God, this is getting to be a real company."[13] It was real enough that in 1989, as Ardent failed, Moler became a MathWorks employee, adding the specially created role of chief scientist to his ceremonial position as chairman.[14] He joined a staff of around 30.

Moler has continued to avoid involvement in the firm's routine operations, claiming that he does not enjoy this work and is not good at it. He does, however, enjoy writing code and adding new capabilities to the system and describes himself as a "real hacker" who can still find himself staying up all night to tackle an interesting programming job.[15] He has written new code to support sparse matrices, created an interface with Maple to use its symbolic capabilities, and extended Matlab with special functions and ordinary differential equation solvers. He has also written much of the product documentation. Over time, however, he gradually moved out of developing core parts of the system, leaving this work to a newer generation of mathematical software specialists.

Today, the MathWorks employs more than 1,000 people but offers only two main product families: Matlab, and a related system for embedded application generation called Simulink. Matlab dominates the market for mathematical software. Although it is very widely used in universities, its real edge over competitors such as Mathematica and Maple comes in its acceptance by industrial groups. Most of its users are practicing scientists and engineers in industrial organizations, particularly automotive and aerospace firms. Its early strength in systems control has been followed with specialized toolboxes for signal processing, image processing, bioinformatics, and financial modeling applications. The firm has tried to retain an academically inspired culture and strong ties with universities as it has grown. It remains privately held, Moler and his partners having held out against the lure of the stock market during the boom years of the 1990s.

Current interests

In late 2005, Moler was elected as SIAM president, to serve a two-year term that started in January 2007. This followed his receipt, in July 2005, of the SIAM Prize for Distinguished Service to the Profession.

Moler retains a role in directing the technological evolution of Matlab. Work is currently under way to switch from interpretation to a just-in-time compiler to boost performance of Matlab programs. He is also enlisting his colleagues in the numerical analysis community to help replace the core Matlab algorithms with versions able to exploit the multiprocessor configurations now common on high-end personal computers. Beyond this, Moler writes that

The traditional features that I've worked on, like basic matrix computation, are mature. But there is a great deal of mathematical development in progress and planned for the future. As a company, we are broadening our mathematical interests and expertise. Current work includes:

- New optimization techniques
- Genetic algorithms
- Mixed integer/continuous linear and nonlinear programming
- More powerful statistical computations
- Bioinformatics
- Biosimulation
- Control in the presence of uncertainty (robust control)
- Image morphing
- Filter design

We may not be inventing new mathematics or new mathematical algorithms, but we are making mathematical thinking and mathematical techniques accessible to scientists and engineers who might not encounter them otherwise.[16]

Moler now has the freedom and security to work on almost anything he wants to. On the policy level, this has included service on the advisory boards of various interdisciplinary centers for applied computing research and education. On a more personal level, he has been teaching an undergraduate course on numerical methods to computer science majors at the University of California, Santa Barbara, something he describes as "a real challenge, because they don't know very much mathematics."[17] This course uses a new textbook, *Numerical Computing*, with Matlab which he has made available online,[18] together with sample code to provide interactive examples of the use of mathematics, such as Fourier analysis of touch-tone phone dialing. Moler says he is prouder of this code than of any other program he has written, because it instructs people as well as controls machines. He admits to the hope that one or two of the students he teaches will perhaps have awakened in them an enduring love for the subject akin to that he experienced more than 40 years ago, in John Todd's classroom at Caltech.[19]

Acknowledgments

References and notes

1. C. Moler, oral history interview by T. Haigh, 8-9 Mar. 2004, Santa Barbara, Calif., Soc. for Industrial and Applied Mathematics, Philadelphia, 2004; available from http://history.siam.org/oralhistories/moler.htm, p. 4. Throughout this biography, details not otherwise attributed are taken from this source.
2. Ibid., p. 9.
3. G.E. Forsythe and C.B. Moler, *Computer Solution of Linear Algebraic Systems*, Prentice Hall, 1967.
4. C.B. Moler, "Finite Difference Methods for the Eigenvalues of Laplace's Operator," doctoral dissertation, Dept. Mathematics, Stanford Univ. 1965.
5. C. Moler, L. Fox, and P. Henrici, "Approximations and Bounds for Eigenvalues of Elliptic Operators," *SIAM J. Numerical Analysis*, vol. 4, 1967, pp. 89-102.
6. J.M. Boyle et al., "NATS: A Collaborative Effort to Certify and Disseminate Mathematical Software," *Proc. ACM Ann. Conf.*, vol. 2, 1972, pp. 630-635. C.B. Moler and J.J. Dongarra, "EISPACK—A Package for Solving Matrix Eigenvalue Problems," *Sources and Development of Mathematical Software*, W.R. Cowell, ed., Prentice Hall, 1984, pp. 68-87.
7. J.H. Wilkinson, *The Algebraic Eigenvalue Problem*, Clarendon Press, 1965; J.H. Wilkinson and C. Reinsch, *Handbook for Automatic Computation, Vol. 2: Linear Algebra*, Springer-Verlag, 1971.
8. B.T. Smith et al., *Matrix Eigensystem Routines—EISPACK User's Guide*, Springer-Verlag, 1974.
9. C.B. Moler and G.W. Stewart, "An Algorithm for Generalized Matrix Eigenvalue Problems," *SIAM J. Numerical Analysis*, vol. 10, no. 2, 1973, pp. 241-256.
10. J.J. Dongarra et al., *LINPACK Users' Guide*, SIAM Publications, 1979.
11. G.E. Forsythe, M.A. Malcolm, and C.B. Moler, *Computer Methods for Mathematical Computations*, Prentice Hall, 1977.
12. The presentations were published as C.B. Moler, "Design of an Interactive Matrix Calculator," *Proc. 1980 Nat'l Computer Conf.*, AFIPS Press, 1980, and C.B. Moler, "Demonstration of a Matrix Laboratory," *Proc. Cocoyoc Conf. Numerical Analysis*, LNM 909, J.-P. Hennart, ed., Springer-Verlag, 1983, pp. 84-89.
13. C. Moler, oral history interview, p. 79.
14. See http://www.mathworks.com/company/aboutus/founders/clevemoler.html.
15. C. Moler, oral history interview, p. 84.
16. C.B. Moler, personal communication to author, 8 Mar. 2006.
17. C. Moler, oral history interview, p. 64.
18. See http://www.mathworks.com/moler.
19. C. Moler, oral history interview, p. 62.

Readers may contact department editor Thomas Haigh at annals-biographies@computer.org.