Bulletin of the Seismological Society of America, Vol. 78, No. 6, pp. 2105–2109, December 1988

PRESIDENTIAL ADDRESS

THE EARLY YEARS OF COMPUTATIONAL SEISMOLOGY AT CALTECH

DAVID G. HARKRIDER

Early this year I was asked to make some "remarks" at the 100th anniversary of the Berkeley station as President of the SSA. Not knowing exactly what was expected, I decided to do some background research on the history of the Berkeley station as found in papers, letters, and reports to the *Bulletin*. I found these very informative and some very entertaining. Even though a lot of the information was obviously hearsay and written well after the events, I felt that there was a place for this type of anecdotal "history" in the *Bulletin* and probably no better place for it than in the *Presidential Address*.

With this in mind I felt that there might be some interest in the early days of computational seismology. Since I was there when it all started, at least at the Caltech Seismological Laboratory, I felt that I had better put down what I remember of the events and accomplishments before I am too old to recount them. It would be interesting to hear also about similar developments from Jarosch in Israel; Dorman, Landisman, and Alsop at Lamont; and M. Saito in Japan.

It all began, for me at least, in the spring of 1959. As was usual in those of days of yore, it was the news that Lamont had bought a computer that spurred our young director Frank Press to buy one for the lab. It was announced at coffee by Frank that he had arranged for the purchase of a Bendix G-15D, which was to arrive in a few months and would be working by July.

Press was particularly motivated by the news that Jim Dorman at Lamont was developing the Haskell-Thompson algorithm for calculating Rayleigh and Love dispersion curves for realistic earth models. At that time I was using a Frieden desk calculator and column tab data paper trying to evaluate Love-wave dispersion approximately with Jeffrey's variational principles (and an extension of them developed by Takeuchi a couple of years earlier while at Caltech for a year with a new young post-doc, Kei Aki). Since this was slow, tedious work and of unknown accuracy, Frank wanted to know if I would check over his Rayleigh-wave program in the Bendix interpretive language and write one for Love-wave dispersion. Being a young graduate student and scientifically naive, I thought this would be an ideal way to check the accuracy of my variational calculations. I hate to admit that it didn't even occur to me then that, with these new dispersion codes, there was no need for variational calculations no matter how theoretically elegant they were.

One of the reasons that I chose Caltech over MIT for graduate study was the emphasis that Caltech put in their brochure on the importance of computers in the future of science. It wasn't until I got there that I found out that it was public relations hype and that their so-called digital computer, the Burrough's 220, was not only awkward to use but that it was an open shop where each user scheduled

Presented on May 25, 1988, at the joint meeting of the Seismological Society of America and the Seismological Society of Japan, held in Honolulu, Hawaii.

the total use of the machine. And if there was a bug in your program you might have to wait a day or more for your next scheduled use to correct it. Not only was the machine difficult but it was located four miles away at Caltech's main campus and was the object of desire of the 500 or so graduate students there. Only four of my fellow colleagues from the lab had the desire and motivation to take the special course needed to memorize the machine language instruction set and sign up for their turn on the Burroughs: Bob Phinney, Don Anderson, Shelton Alexander, and Laszlo Lenches, a newly arrived Hungarian freedom fighter who was later to set the standards and high quality for geophysical illustrations in our technical journals. Meanwhile I was putting off the inevitable necessity of learning to use this machine by playing with equations and approximation techniques.

Even though our new Bendix computer would be an open shop operation also, it was going to be located at the Lab and be available to us 24 hours a day. It also had an interpretive machine language, which was one step up from primary machine language programming. I differentiate between machine language, which is a stored number, combining command and operand location, and assembly language, which incorporates alphanumeric mnemonics and is a great step forward. For all of the above reasons, I jumped at the chance to work on the new computer with Frank Press.

The minute the computer arrived and was installed, Frank and I entered into an undeclared race to be the first to have a program that worked on the Bendix. I even worked over the Fourth of July to win. Frank used to relate that the first program I ever wrote worked without debugging. That was sort of true, in that it gave the right answer on the console typewriter, but until now I never confessed that the output format was not what I thought I had programed.

On the Bendix G-15D you either entered your program command by command on the console typewriter (no CRT) or punched a paper tape on the Friedan Flexwriter and then ran the tape through an optical reader mounted on the face of the computer, which looked like a large refrigerator. The first computer I ever saw was similar except that it looked like a deep-freeze (refrigerator lying on its side) with only a typewriter on top of it. Our only output was typed by the console on the fan-folded, perforated, continuous sheet of typewriter paper, which was fed into the console. Each sheet was sequentially numbered. The number on the bottom of the page with the results or elastic input model was sometimes used as the model name. It was rumored that was how Jim Dorman's ocean model name 8099 came about.

There were faster computers than the Bendix available to our group, including the Burroughs. One of these was the IBM 604 at UCLA. Bob Phinney seemed to be content with Caltech's Burroughs and continued to use it on his leaky mode calculation, but Shelton Alexander felt the need for a larger and faster computer such as the IBM. The problem was that we heard that it used a new and completely different language. This was confirmed by the visit of Norm Ness, an old MIT friend of Stewart Smith. He not only told us of the joys of using FORTRAN but left a manual with us. It was completely different from anything I had ever seen and I found it almost impossible to read.

Somehow Frank Press arranged for Shelton to obtain an account at UCLA and off he went, usually in the evening, and he stayed rather later. Being suspicious that Shelton was using this as an excuse to get out at night for some extra-curricular activities, and not wanting to miss out on a good thing, I asked if I could go too.

Unfortunately, he really was working on the IBM. Since I had nothing else to do while waiting for Shelton to submit his job at UCLA's closed shop operation, I picked up a discarded program, with a programers' reference manual deciphered the program, and in this way learned FORTRAN. I hate to admit it but, 30 years later, I still cannot read a programing teaching manual. The only way I can learn a language such as "c" is to take a c-program and decipher it line by line with a reference manual. Before I deciphered the original program, I had Alexander verify that it was a debugged and working program. This ability of Shelton's to pick up a program and almost immediately detect any errors was exploited by all of us at the Lab. The less Alexander knew about the program the better.

This technique for learning new programing language is slow but effective and can lead to some interesting programing habits. For instance, in the early days of FORTRAN the I/O was handled by FAP, the FORTRAN Assembly Program. The IBM word at that time was 36 bits long and each character or Binary Coded Decimal was 6 bits or 6 characters per word. Since they wanted the FORTRAN line to be an even multiple of words, they chose 12 words or a 72 character FORTRAN line. FAP handled the input data the same way, i.e., only 72 characters per line even though there was space for 80 characters on each punched card. (I bet you always wondered why the standard FORTRAN line is only 72 characters long and why the early magnetic tapes were 7 tracks, i.e., 6 bits + a parity bit.) Although IBM long ago gave up this restriction when they went to bytes, or 8 bit characters, I still use data input fields that are multiples of six and never input more than 72 characters per line. That was the way it was done on the first FORTRAN program I ever picked up and that's the way I do it now.

One of the reasons that Shelton Alexander changed to the IBM from the Bendix was a need for a Fourier transform program. Press was especially eager for one at the Lab, since he wanted to analyze the 1952 Kamchitka earthquake recordings at Caltech for free oscillation peaks. He hired a professional programer, Saul Shragowitz, to develop a fast machine language algorithm on the Bendix. I was in the computer room the day Frank plotted his results and remember his excitement. To this day he claims that this was the first experimental determination of the Earth's free oscillation periods.

Kei Aki, who was now back as a post-doc at the Lab, had other uses for a Fourier transform and he was in too much of a hurry to wait for the machine language version. So he developed an approximate method that made use of an explicit knowledge of the phase and its spectral derivative. Since he had independent estimates of these in the form of phase and group velocity, Kei was able to perform phase and amplitude equalization of surface waves in order to estimate the source-time-function or to phase-correct the time-domain signals back to a common range in order to determine the radiation pattern. The technique gave good time-domain estimates at selected times of interest and was not affected by wrap-around caused by large frequency intervals. Then as now, fast is relative and if we came in at night to work and found Aki's program executing on the Bendix, we knew that we had time to go to a night baseball game before it would finish. At least that was what Bob Kovatch, Shelton Alexander, and I used as our excuse to go to the L.A. Dodgers game.

Sometimes it is hard for me to believe the number of geophysical advances that occurred between the summers of 1959 and 1960. To name a few: the publication by Jim Dorman et al. of the first realistic model of the shear-velocity structure of the oceanic mantle with a low-velocity zone and the "race to Helsinki," the 1960 IASPIE meeting. Following the May 1960 Chilean earthquake, Stewart Smith as Benioff's student and I as Press's student went to Sandia Labs, Albuquerque, New Mexico to analyze Caltech recordings of this event with their large IBM 7090 computer. It was early July and, after registering at the motel as Mr. Harkrider and Mrs. Smith, we went out to the facility, which was the most secure national laboratory I have ever visited. Since Stewart and I did not have security clearances, we were never out of sight of our escort. The paths between buildings were well-marked with armed guard towers at intervals to make sure you didn't wander. Our movements were so restricted that all we were allowed to do was sit at a table and plot the results. I quickly tired of the project and on our return told Frank that I would rather work on something else. Steward carried on and gave Press enough results so that he was able, according to Bullen in the 1963 edition of his book to start off one of the most dramatic scientific sessions ever witnessed by Bullen: The observation of the Earth's free oscillations.

In these early days, the computer was used not only in the reduction and interpretation of data but also as a means of testing theoretical assumptions. Anderson was able to numerically show that the often-stated assumption that group velocity perturbations were second-order with velocity and density changes was incorrect.

In the meantime, the UCLA IBM 604 had grown into a 704 and then into a 709. At about that time Anton Hales arrived from South Africa as a visiting scientist. He was interested in the possibility of very-low-velocity layers in the mantle. Earlier Press and I had worked in collaboration with a JPL (Jet Propulsion Lab) programer on a Rayleigh-wave dispersion program that was developed on an IBM 704, then on an IBM 709, and finally on an IBM 7090 at JPL. My main contributions were the equations and algorithm for a liquid layer at depth. Press therefore thought I was the ideal student to work on this problem with him and Hales. I remember Anton asking me if I was basically interested in the theory or in the Earth? Of course I lied and said I was interested in the Earth's composition. It wasn't till much later that I found out that an interest in the physical rather than chemical processes in the Earth was acceptable geophysics.

My ability in program development always made me a popular collaborator with Press when some new geophysical wave phenomenon was observed, especially if the wave was dispersive. This was how I became involved, in the spring of 1962, in the analysis of the dispersion of atmospheric acoustic gravity waves from atmospheric nuclear explosions. This required working on JPL's IBM 7090, which was only available to Caltech during the hours from midnight until morning, since Caltech didn't get its own IBM 7090 until the summer of 1962. For awhile I thought I would never get to work on my thesis, which was on the excitation and synthesis of surface waves—but my turn finally came in 1962–1963, when Caltech finally replaced its Burroughs with an IBM 7090 as the main campus computer. The convenience and reasonable turn-around, especially when this was modernized to an IBM 7094, gave me the time to finish my thesis and collaborate with Ben-Menahem and Toksöz on many interesting problems.

Because of time, I had decided to restrict my tale to the years when computers were electronic tubes and memory was made of tiny magnetic cores. This physical restriction on memory or core size kept the maximum memory available on an IBM 7094 to around 16 kilowords for the individual user, depending, of course, on the operating system. Actually the main memory on the Seismo Lab's Bendix G-15D was a magnetic drum. The advent of distributed or disk memory and the transistor-based central processing units, CPUs, made possible the analysis of large data sets without the necessity of tape drives being used for tempory memory during execution. This revolution, spearheaded by the IBM 360 systems, introduced the byte and virtual memory. This, along with the development of the FFFT, happened just after I left Caltech for Brown University as a new assistant professor.

But that's another story.