NOVEL APPLICATIONS OF COMPUTERS

Encouraged by the availability of computers and challenged by particular problems, physicists have discovered new ways to use these powerful machines

Barbara G. Levi

BECAUSE most physicists now have access to a computer, they are able to tackle problems that might formerly have been unsolvable; at the very least they are able to solve their problems far more quickly and accurately. In a recent sample poll of its readers, PHYSICS TODAY learned that about 90% now use computers in their work. Most of these (85%) perform their calculations at the university or company computer center. Some 34% have computing facilities available to them on a time-sharing basis, and about 7% have their own computer. In approaching their problems, how many of these physicists are devising novel techniques?

In one sense, no applications of these computers are "novel": Computers are such a general tool that the same techniques apply to a wide range of problems. At the same time, all applications are novel: Each experiment has its own idiosyncracies that require some innovation and adaptation. Here are some programs that may be new to you. We have tried to select a variety of applications and hope that at least one stimulates further new ideas.

Barbara G. Levi is an associate editor on the staff of PHYSICS TODAY.

Symbolic algebra

Any graduate student who has struggled through the calculation of Compton scattering, even to lowest order, will appreciate the value of the REDUCE system[1,2] developed by Anthony C. Hearn of the University of Utah. This system of computer programs for nonnumeric algebraic manipulation was designed to perform the tedious repetitive calculations often involved in elementary-particle physics. As it evolved, REDUCE acquired properties general enough to be applied to many other fields, and its users now include even chemical and mechanical engineers.

As a general program, REDUCE can handle expressions involving rational functions of polynomials with symbolic variables and functions introduced by the user. It has facilities for

- simplifying expressions
- performing symbolic differentiation of expressions
- cancelling the greatest common divisors in the numerator and denominator of expressions
- manipulating general matrix and tensor expressions, and computing traces of Dirac gamma-matrix expressions.

As a specific program, Hearn uses REDUCE to perform a symbolic analysis of Feynman diagrams, primarily for quantum electrodynamical processes. These diagrams are schematic representations of terms in a perturbation expansion of the S-matrix. The number of vertices in each diagram is proportional to the order of the term describing the contribution of that process to the total expansion. As the energy of the process increases, so does the order of terms that must be kept in the perturbation expansion. Thus, to compare theory with experiment at the high energies available with today's big machines requires a formidable calculation on the part of the theorist. Hearn's system enables one to perform in a few minutes of computer time a calculation that might take six months if done "by hand."

REDUCE itself is used to simplify the Dirac gamma-matrix trace expressions that arise in these computations. An additional program,[3] developed in collaboration with John A. Campbell of the University of Texas

```
COMMENT SYMBOLIC COMPUTATION OF LEGENDRE POLYNOMIALS
ARRAY P(7)
P(Ø):=1$
P(1):=X$
FOR M:=2:7 DO P(M):=(2-1/M)*X*P(M-1) - (1-1/M)*P(M-2);
P(2):=3/2 X - 1/2
P(3):=5/2 X^3 - 3/2 X
P(4):=35/8 X^4 - 15/4 X^2 + 3/8
P(5):=63/8 X^5 - 35/4 X^3 + 15/8 X
P(6):=231/16 X^6 - 315/16 X^4 + 1Ø5/16 X^2 - 5/16
P(7):=429/16 X^7 - 693/16 X^5 + 315/16 X^3 - 35/16 X
END;
```

GENERATION of legendre polynomials to any order is trivial with the REDUCE program. The programmer merely specifies the initial conditions, desired order and the recursion relation (statements in color).

at Austin, also tackles the often difficult problem of first deriving these trace expressions. Once you specify the particles involved, their interaction and the order of perturbation desired, the program will generate all possible Feynman graphs for that order, derive and simplify the matrix elements and then compute the trace expressions that REDUCE can simplify.

REDUCE was written in the symbol-manipulating language LISP. However, the user is not required to know this language. The commands that he uses are similar to the program language ALGOL. REDUCE is presently available in versions that

Generalized spectrometry

Solid-state physicists and chemists at the IBM Research Laboratory in San Jose use a variety of spectrometers in their research. If any one of these researchers wishes to connect his instrument on line to the IBM 1800 computer, he finds that the road has already been paved for him with a general experiment monitoring (GEM) system developed by Henry M. Gladney.[1,2] This system provides much of the software interface, consisting of input/output (I/O) control and data-gathering routines, required by the general user. The experimenter need only supply additional routines peculiar to his own apparatus.

The GEM system provides resource sharing; many experimenters share the computer response time. Together with the time-sharing executive (TSX) monitor written by IBM for the 1800, it allows up to 15 instruments to be connected to the computer, with six of these simultaneously recording data. The 12 instruments currently on line include a far-infrared Michelson interferometer, a nuclear magnetic resonance (NMR) spectrometer and electron paramagnetic resonance (EPR) spectrometer, a gas-liquid chromatograph and a high-resolution Stark spectrometer.

Gladney observed that, from the point of view of experimental control, a large class of spectrometers shares many common requirements. One or several independent variables must be set and controlled. Constant variables, such as temperature, are set once at the beginning of the experiment; variables that are swept in time require an interface and digital control by the computer. The analog signal from the response must be sensed and amplified, and usually requires analog integration to improve the signal-to-noise ratio. The computer must control the gain, time constant and modulation amplitude of the detector. Because it is normally desirable to display the output on an x-ray plotter or on a storage oscilloscope, the computer must control a pair of analog outputs.

The GEM system supports independent operation modules that perform these general tasks. The modules can be concatenated (linked) and specified in any order. This design allows maximal flexibility and adaptability. The experimenter may construct his hardware interface with little reference to the programming system.

The spectrometer experiments also share data manipulation requirements. All measure stochastic data, and hence require routines for filtering and smoothing, curve fitting, performing integral transformations, averaging and plotting. The group at San Jose is accumulating a library of such routines.

The on-line experiments are run in a "background" mode so that the central processor and memory of the computer are available for processing data from completed experiments or making other calculations. This arrangement is especially feasible with most spectrometers because the rate of gathering data is slow compared to the rate of digital computation. Furthermore, most spectrometers do not require accurately set time intervals as long as the actual duration of the interval is recorded; the timing can be controlled by the computer. Provisions have been made for manual intervention, should the experimenter wish to monitor the results and make adjustments.

One research project concerns the study of transition-metal impurities in single crystals at low temperatures. In this research they use an EPR spectrometer. Because of the nature of these studies, computer control is almost a necessity. It is difficult to prepare single crystals with a satisfactory level of doping. Among run on the IBM 360, PDP-10 and CDC 6400. A debugging service is offered and the system is updated as improvements are made.

Among the future improvements planned for REDUCE are the addition of routines for symbolic factorization and for integration. Hearn and coworkers are also developing routines to plot mathematical expressions and to display graphical representations of the Feynman diagrams generated by the programs.

References

1. A. C. Hearn, Commun. ACM **9**, 573 (1966).
2. A. C. Hearn, "REDUCE—A User Oriented Interactive System for Algebraic Simplification," in *Interactive Systems for Experimental Applied Mathematics* (M. Klerer, J. Reinfelds, eds.), Academic, New York (1968), page 79.
3. J. A. Campbell, A. C. Hearn, J. Computational Phys. **5**, 280 (1970).

many samples grown, perhaps only a few will be acceptable. The computer can conduct the tedious and routine search of all crystals for an acceptable sample, freeing the researchers from a time-consuming chore.

In the EPR experiment, the independent variable is the magnetic field, controlled by a modified version of a commercially available Hall-effect device. In addition, Gladney and his colleagues observe the spectrum as a function of the magnetic-field orientation. Thus the computer controls the angular position of a 2-ton rotating electromagnet. They have the usual interface between detector and computer. Finally, they connect the computer to an x-y recorder to monitor input to the computer and to display the output.

As a final testimony to the flexibility of the system, Gladney shares a teletypewriter with another experiment in the same room; the output from the two experiments appears on two halves of the paper, and the paper is split down the middle at the conclusion of the experiments!

References

1. H. M. Gladney, J. Computational Phys. **2**, 255 (1968).
2. B. Johnson, T. Kuga, H. M. Gladney, IBM J. Res. Develop. **13**, 36 (1969).

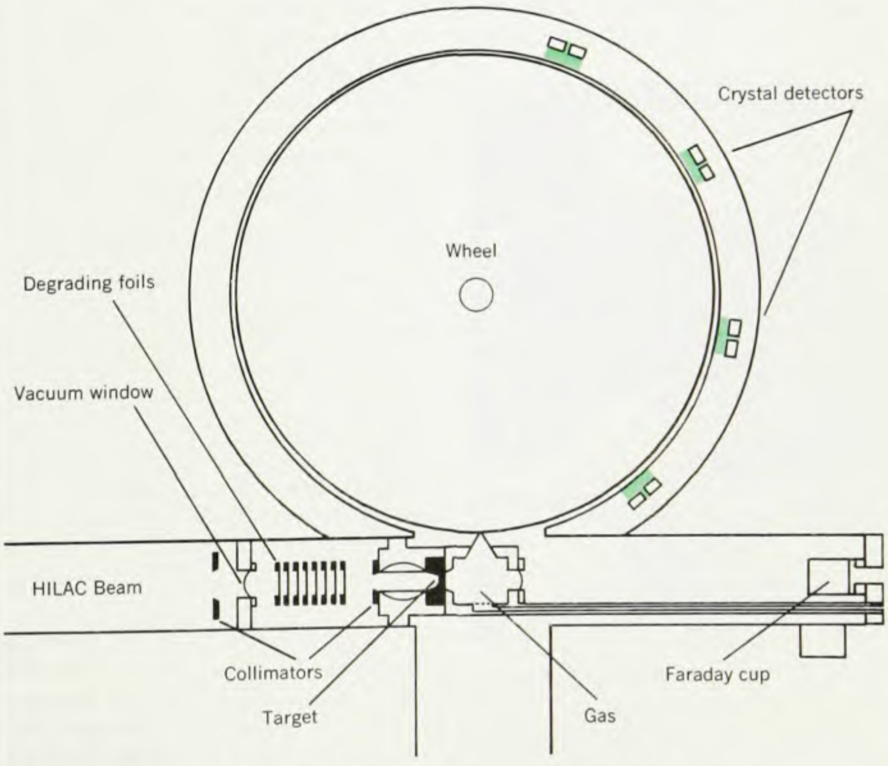
Multichannel analyzer

In the past year, several alpha-emitting isotopes of elements 104 and 105 have been identified[1,2] by a group led by Albert Ghiorso at the Heavy-Ion Linear Accelerator (HILAC) located at the University of California's Lawrence Radiation Laboratory. It is critical in this work to be able to measure the complex alpha-particle spectrum with high resolution: For example, the element 104^{261} emits an alpha particle with energy very close to that of its daughter, the 8.3-MeV No^{257}. Furthermore, it is usually necessary to measure precisely the lifetime of the daughter and even the granddaughter to ensure definite identification of the parent element. In the HILAC experiments, an on-line computer handles the extensive bookkeeping tasks involved in these measurements and acts as a huge multichannel analyzer for the measurement of the spectra.

The detection system at the HILAC is sketched in the accompanying figure. When the heavy-ion beam hits the target, it ejects any atoms of heavy elements that are formed. The recoil energy of such atoms is absorbed in a chamber of helium gas. The atoms then stream along with the gas into a vacuum chamber, where many of the atoms are adsorbed onto the surface of a large (50 cm) wheel. Solid-state crystal detectors facing the rim of the wheel register the activity from decays of these atoms. The wheel rotates periodically so that each group of atoms is measured a number of times by each of the seven detectors.

The computer stores the spectrum from each of the detectors and for each new position of the wheel. These spectra are analyzed in 512-channel segments covering the range from 6 to 12 MeV. Ghiorso's group uses 112 000 channels and plans to extend this figure eventually to half a million channels. They use a PDP-9 computer, with an 18-bit, 24 000 word storage.

The activity of the daughter atoms can be measured simultaneously with that of the mother atoms. Because alpha particles are emitted in all directions, some alpha particles from the decay of the parent will be emitted into the wheel itself while the recoiling daughters are deposited on the crystal detector opposite the wheel. Periodically, this detector is shuttled to a new position facing a second detector that measures the activity of the daughter atoms.



DETECTION SYSTEM for alpha-emitting isotopes. When these heavy isotopes are formed, they are ejected from the target, stopped in the gas chamber, then deposited on the wheel. Signals from each crystal detector (color) are analyzed by computer.

This "milking" of the daughter atoms may be done at regular intervals, or, especially if one wishes to detect the granddaughter, it may be controlled by computer logic. Once the computer identifies the daughter by its spectrum or decay time, it can shut off the HILAC beam and shuttle the crystals into position to detect the granddaughter.

In addition to storing the spectrum, the PDP-9 must keep track of a large amount of data. It must record the position of the wheel, the shuttle configuration, the crystal identification, the time corresponding to the detection of mother atoms and the same for daughter atoms, and so forth. Often the computer decides, on the basis of the data, what information is relevant, and is able to store the raw data in the most compact form. It may also sort the data as it comes in according to preprogrammed specifications.

At present the computer performs only limited calculations on line. Data processing, such as spectrum fitting or normalizing the gain on the detectors, is done at some later time. Such work is usually performed on a CDC 6600 machine.

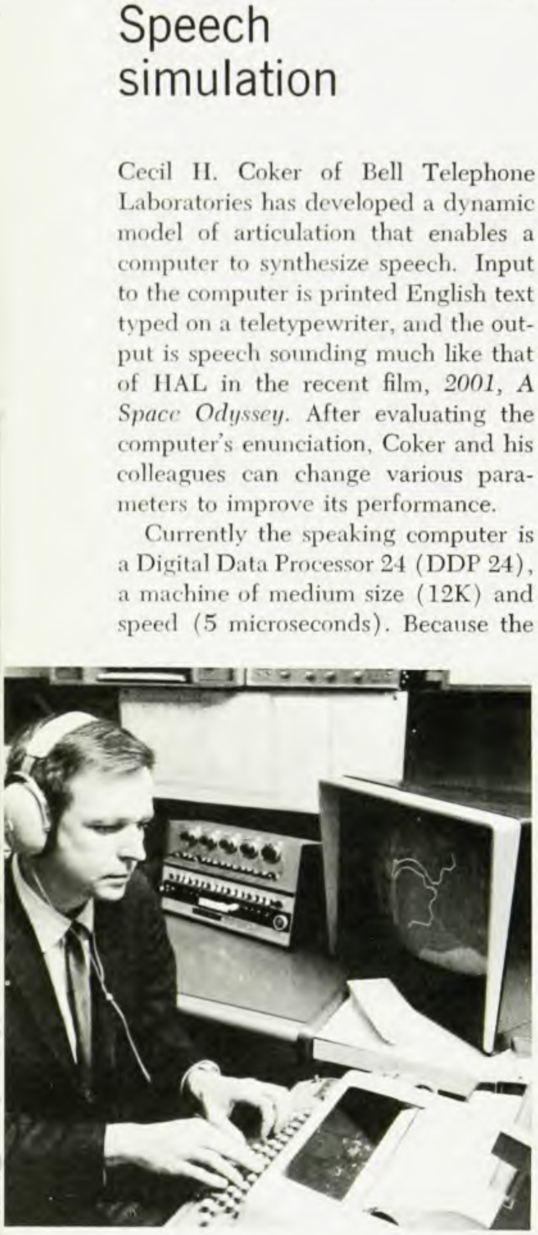
References

1. A. Ghiorso, M. Nurmia, J. Harris, K. Eskola, P. Eskola, Phys. Rev. Lett. **22**, 1317 (1969).
2. A. Ghiorso, M. Nurmia, K. Eskola, J. Harris, P. Eskola, to be published in Phys. Rev. Lett.

Speech simulation

Cecil H. Coker of Bell Telephone Laboratories has developed a dynamic model of articulation that enables a computer to synthesize speech. Input to the computer is printed English text typed on a teletypewriter, and the output is speech sounding much like that of HAL in the recent film, *2001, A Space Odyssey*. After evaluating the computer's enunciation, Coker and his colleagues can change various parameters to improve its performance.

Currently the speaking computer is a Digital Data Processor 24 (DDP 24), a machine of medium size (12K) and speed (5 microseconds). Because the model involves many recursive calculations, this computer can speak only on a time scale ten to fifteen times slower than real time including time to type the input. However, this program is now being put on a faster DDP 516 machine, where it will run two to three times slower than real time. The availability of faster computers and medium-scale-integration components guarantees the possibility of real-time speech. To extend the memory capacity of the DDP, Coker's group has connected it by phone lines to a General Electric (GE) 635 machine, available to them through in-house time sharing.



AS THE COMPUTER SPEAKS, an outline of the mouth appears on an oscilloscope, showing the positions of the tongue and lips in the model. Coker dictates sentences to the computer on the teletypewriter and evaluates the spoken output.

Originally, the speech-synthesizer program could accept as input only phrases that had already been reduced to basic phonetic units (phonemes), together with notation indicating the pitch and stress. Recently, Coker, Noriko Umeda and other members of the group have developed a program that enables the computer itself to make this phonetic translation, given written English text.

The dynamic model of speech developed by Coker and Osamu Fujimura[1] uses seven coordinates to describe the position of the tongue body, tongue tip, lips and velum (soft palate). These seven coordinates were selected empirically as those that gave the set of most nearly independent response modes. Corresponding to each phoneme are specific values for these coordinates, giving the target position for pronouncing that phoneme. The mouth never actually reaches this target position before it must begin to form the next sound, so the computer must control the transition between phonemes. For this reason each position coordinate is assigned a priority corresponding to its importance in forming a particular phoneme. For a higher priority, the coordinate makes the transition to the phoneme earlier and stays there longer. The vocal chords are represented in this model by two parameters, but the representation of actual events is still very approximate.

Given the set of position coordinates, the computer proceeds to characterize the spoken output as that from an acoustic tube with a time-varying, nonuniform cross-sectional area. It solves for the resonant frequencies of such a tube, and reproduces these resonances with an on-line analog synthesizer.[2]

In addition to the audio output, Coker can study visual output on an oscilloscope screen. It displays an outline of the mouth, monitoring the changing positions of the tongue, lips and so forth as the computer speaks. It also displays plots of frequency versus time for the first three resonances.

At present, Coker and his colleagues interact with the computer on line, adjusting the various parameters to make the output more closely resemble the human voice. When developing the phonetic-translation program, they matched spectrograms (3-dimensional plots of frequency, amplitude and time) with real speech. However they find that the human ear is more sensitive than these quantitative comparisons.

The speech simulator is a valuable tool for the study of linguistics and the physiology of speech. The main possibilities for practical applications are the oral relay of stored information and results from information-retrieval programs, aids for the blind and programmed instruction.

References

1. C. H. Coker, O. Fujimura, J. Acoust. Soc. Am. **40**, 1271 (1966).
2. C. H. Coker, P. Cummiskey, J. Acoust. Soc. Am. **38**, 940 (1965). □