

Woody Bledsoe

His Life and Legacy

Michael Ballantyne, Robert S. Boyer, and Larry Hines

■ Woodrow Wilson (Woody) Bledsoe died on 4 October 1995 of ALS, more commonly known as Lou Gehrig's disease. Woody was one of the founders of AI, making early contributions in pattern recognition and automated reasoning. He continued to make significant contributions to AI throughout his long career. His legacy consists not only of his scientific work but also of several generations of scientists who learned from Woody the joy of scientific research and the way to go about it. Woody's enthusiasm, his perpetual sense of optimism, his can-do attitude, and his deep sense of duty to humanity offered those who knew him the hope and comfort that truly good and great men do exist.

Woodrow Wilson (Woody) Bledsoe died on 4 October 1995 of ALS, more commonly known as Lou Gehrig's disease. Woody was one of the founders of AI, making early contributions in pattern recognition and automated reasoning. He continued to make significant contributions to AI throughout his long career. His legacy consists not only of his scientific work but also of several generations of scientists who learned from Woody the joy of scientific research and the way to go about it. Woody's enthusiasm, his perpetual sense of optimism, his can-do attitude, and his deep sense of duty to humanity offered those who knew him the hope and comfort that truly good and great men do exist.

Early Years

Woody was born on 12 November 1921 on a little farm near Maysville, Oklahoma. His father, Thomas, had run a turpentine plantation in Alabama, but in 1917, a fire burned him out. He moved to Oklahoma to try his luck at farming. Woody was the fourth child born to Thomas's second wife Eva, his first

wife having died after raising six children. Two more siblings followed. There were many mouths and precious little money. However, because they lived on a farm, there was plenty of good food.

Although Thomas had only a couple years of formal schooling, both he and Eva, who had been a school teacher, "always emphasized education and encouraged learning" (Bledsoe 1976). Woody's schools consisted of one- or two-room buildings:

We didn't know we were being deprived in those one and two room schools. We spent a lot of time reading by ourselves, because most of the time the other grades were having their classes. But we DID learn, and had some pretty good teachers (Bledsoe 1976).

Early on, Woody was enthralled by mathematics and recalls spending "hours just roaming around, sometimes working mathematics problems mentally" (Bledsoe 1976).

When Woody was 12, his father died. It was a devastating blow both emotionally and financially. As Woody recalled, "We were poor before, but after papa died in January 1934, things got worse" (Bledsoe 1976). He and the rest of his brothers and sisters worked dreary 10-hour days to make ends meet.

Woody ran away from home at 16. He found work in north Texas driving a tractor all night. After a month, he hopped a freight train to Colorado to visit his brother, who was working in a Civilian Conservation Corps camp. After a few weeks, he made his way to the south Texas town of Calliham, where he lived with some friends for a year. He graduated from high school during that year (1939). Woody then returned to live with his mother, who had moved to Norman, Oklahoma. He took a job as a dishwasher, working 12-hour days 7 days a week. He enrolled at the University of Oklahoma in the spring of 1940, cut-

ting his work load back to 8 hours a day. In April, the restaurant owner forced him back into working 12-hour days, which was too much even for Woody's fortitude. He left the university without saying good-bye and joined the United States Army.

The army suited Woody. By the time he went to Officer's Candidate School (OCS) in 1942, he had been promoted to second lieutenant. While at OCS, Woody had an experience that had a profound effect on him:

Another experience at OCS at Fort Belvoir left a lasting impression on me. One day we were to do Map Reading, but it was raining pretty hard as we traveled to the training place in the back of a covered army truck. We thought, "Well this rain is a blessing, no one can do map reading on a day like this."

Shortly after we stopped, the sergeant opened the flap and said, "Get out here. Let's do the map reading." One trainee said, "How can we do it if it is raining?" The sergeant replied, "If we have to do it, then we DO IT. GET OUT OF THERE (*caps ours!*)" Well we did it. It took a lot of struggling to try to keep our maps dry, to get on with the work, to finish the training. It taught me that "if we have to do it, then we DO IT." Nothing seems to have delayed me much since that day, even some very challenging times when we crossed the Rhine River (Bledsoe 1976).

While on leave in the summer of 1943, Woody went to Ogden, Utah, to visit his mother, who had moved there several years previously. On this trip, he went to Salt Lake City to visit an army buddy, Richard Norgaard. Woody met Richard's sister Virginia and canceled a date with someone else so that he could take Virginia out instead. Woody later recalled, "I did not see much of my mother during the rest of that leave" (Bledsoe 1976). In November 1943, Woody was sent for training to Fort Belvoir, Virginia. On the way, he stopped in Salt Lake City and proposed to Virginia. On his return, Virginia accepted. They were married the following January.

Woody was sent to Europe in August 1944 as a member of Patton's Third Army, in the Corps of Engineers. The Boyers, in their biographical sketch of Woody (Boyer and Boyer 1991), give the following account of Woody's wartime efforts:

Bledsoe received the Bronze Star medal

for his heroic activities in arranging the transportation of troops across the Rhine in March, 1945. At that time, all the Rhine bridges except the one at Remagen had been destroyed by the retreating German army. Patton's Third Army decided to cross the Rhine by boats near Frankfurt rather than suffer the delay of waiting for bridge construction. Therefore the Army Corps of Engineers hauled naval landing craft (designed for beach landings) by truck across Europe to ferry troops across the Rhine. Bledsoe, by then an Army captain, recalls that there was only light enemy fire during the crossing; his main contribution was figuring out how to get the very large landing craft down narrow roads and actually into the water. His first "research" was experimenting with techniques for launching the craft from trucks into the Mosel River; initial experiments were disastrous, but ingenuity prevailed. The simple idea of backing the trucks into the water, floating off the boats, and hauling the trucks back out with tractors turned out to be key (p. 4).

While Woody was in Europe, his wife gave birth to a daughter, Margaret, who died less than a month later. He would later father a son, Greg, born in March 1947. He eventually had two more children, Pam and Lance.

In October 1945, Woody returned to his wife in Salt Lake City. He could not enter the University of Utah until December, so he took a couple of correspondence courses. In December, he enrolled in the university. His first major was electrical engineering, but he switched to physics. After taking a course from mathematician Ferdinand Bieseke, he changed again, this time to mathematics. After 2-1/2 years, Woody earned his B.S. in June 1948.

Woody planned on attending the California Institute of Technology for graduate school, but when the University of California at Berkeley offered a full fellowship, he went there instead. Among the many well-known mathematicians that Woody took classes from were Alfred Tarski, J. Kelley, and A. P. Morse. Morse was his Ph.D. adviser. While a graduate student, Woody published several papers on his mathematics research. Woody continued this research long after beginning his work in AI. Among his mathematical publications were Bing, Bledsoe, and Mauldin 1974; Bledsoe and Wilks 1972; Bledsoe and Morse 1963, 1955; Bledsoe, Norris, and Rose 1954; and Bledsoe 1952a, 1952b.

It took a lot of struggling to try to keep our maps dry, to get on with the work, to finish the training. It taught me that "if we have to do it, then we DO IT."

Sandia

Woody received his Ph.D. in 1953. Turning down teaching offers at the universities of Utah, Virginia, and Michigan, he decided to “try his hand at atomic research” (Bledsoe 1976); so, he and his family moved to Albuquerque, where Woody went to work for the Sandia Corporation. Sandia was supported entirely by the Atomic Energy Commission for the purpose of defense research. Woody worked in the mathematics department on systems analysis. Boyer describes his work as follows (Boyer and Boyer 1991, p. 8):

Systems analysis is a term used to express engineering concerned with the interactions of components in a “system” that is itself composed of many other systems, e.g., a telephone system or a modern battlefield.

Almost all Woody’s research during this period remains classified. In 1956, he witnessed a hydrogen bomb test on Eniwetock Island in the northwest Marshall Islands.

In 1959, Woody teamed with polymath and fellow Sandia employee Iben Browning to work on pattern recognition. The result of their work was the Bledsoe-Browning or N -tuple method (Bledsoe and Browning 1959). The primary impetus of this work was to recognize letters, both printed and handwritten. Some of the features of this system were that “(1) it handles all kinds of patterns with equal facility; (2) because it does not depend on absolute pattern matching, it can identify a pattern that is not exactly like, but only similar to, a pattern it has previously learned; (3) it does not depend significantly on the location of a pattern on the photomosaic for identification; and (4) it is only partially dependent on the orientation and magnitude of a pattern for identification” (Bledsoe and Browning 1959, p. 225).

The original N -tuple experiment used a 10×15 photocell mosaic partitioned into 75 randomly chosen, exclusive pairs (elements). When an image is projected onto the grid, each cell can assume a value of either 0 or 1. Thus, each element can be in one of four states. The device is “trained” by presenting characters from multiple alphabets. Each character is presented in slightly different positions and orientations. For each of the 300 (75×4) possible states, there is associated a string of n bits, where n is the size of the set of characters to be distinguished. (This set of strings was referred to as the *memory matrix*).

When training the device to recognize the i th character, the i th bit of each of the 75-bit strings associated with the 75 states is set to 1. When presenting a character for recognition, a score for each possible character j is computed by summing the j th-bit values of the strings associated with the 75 states represented by the pixel values on the photocell. The character corresponding to the highest score is the winner.

Bledsoe experimented with many variants of this scheme (Bisson 1962; Bledsoe and Bisson 1962; Bledsoe and Browning 1959). Increasing the size of the tuple improves readability, although it increases the training time. Storing averages, rather than binary values, also improves readability. Careful layout of the tuples on the grid can also improve readability. After discussing some of these improvements in Bledsoe and Bisson (1962), Bledsoe asks,

How much more readability can be expected with further improvements in the matrix? Clearly an optimum set of stored matrix values exists for any given pattern set. Indeed, learning can be described precisely as the attempt to obtain the optimum matrix for specified sets, some methods being superior to others (p. 414).

Panoramic Research, Inc.

In 1960, Woody, along with Browning and Lloyd Lockingen, left Sandia to start Panoramic Research, Inc. (PRI), in Palo Alto, California. The majority of the work involved AI-related contracts from the U.S. Department of Defense and various intelligence agencies.

We were constantly, repeatedly, searching for, proposing and trying out ideas which we hoped would “move the world.” That was one of the first AI groups before the term “Artificial Intelligence” came into use (Bledsoe 1976).

One interesting nongovernment contract was with *McCalls* magazine. At that time, each address on a magazine mailing jacket was printed from a metal plate onto which the subscriber’s name and address were punched. *McCalls* wanted to switch to a computer and commissioned Woody to use his pattern-recognition programs to read all these addresses into the computer. If his program achieved recognition within a small fixed-error tolerance, then his company was to receive a small fee for each address processed.

After some improvements to the recogni-

We were constantly, repeatedly, searching for, proposing and trying out ideas which we hoped would “move the world.”

The studies of Bledsoe and Bremermann came closest to the modern notion of a genetic algorithm. Both suggested binary string codings. Bledsoe presented results of a scheme that combined individual-by-individual generation, mutation, and save-the-better selection.

tion algorithms, he eventually satisfied the contractual restrictions on the error rate using the sample data provided. When the program was tried on the real production data, however, it failed. The problem resulted from reusing old address plates left from canceled subscriptions: New addresses were punched on top of old addresses. The extra noise resulting from the remnants of previous impressions confused the algorithm so that the error rate was exceeded in production.

Many other early AI pioneers were frequent guests at PRI. Among them were John McCarthy, Marvin Minsky, Saul Amarel, and Hans Bremermann. Many of Woody's early publications at PRI were inspired by conversations with Bremermann.

Much of Woody's early efforts at PRI concentrated on improving the N -tuple algorithm:

If "number of images correctly read" is accepted as the definition of "readability, readability for a given pattern set can be regarded as a function of many variables, the variables being values recorded in the matrix. In this light, optimization techniques can be employed to seek the optimum matrix for a given set. The actual matrix which proved optimum will depend naturally on the particular n -tuples selected, as well as on other system parameters, including n itself. Optimization, therefore, must at least include: finding the "best" n , finding the "best" set of n -tuples, and finding the "best" corresponding memory matrix (Bledsoe and Bisson 1962, p. 415).

In Bisson (1962), a gradient search technique was successfully used to improve the N -tuple algorithm. A more interesting effort at optimization involved the use of evolutionary techniques. Bremermann and Anderson (1991) discuss the early attempts to optimize pattern-recognition algorithms:

Around 1960–61, however, we both experimented with the idea of applying "genetic algorithms" to optimize the performance of perceptrons: Treat the synaptic weights like nucleotides of DNA; mutate, recombine cross over and select, as in Darwinian evolution. HJB spent several weeks during the summer with Frank (Rosenblatt) in Ithaca. Subsequently Woody and HJB tried it in earnest, but on simpler and well-defined objective functions (Bremermann 1962; Bledsoe 1961b; Bledsoe and Browning 1959). The method works *in principle* (and is currently a popular method in Artificial Intelli-

gence [Edelman 1987; Holland 1975]) but for neural nets it turned out to be an optimization problem of extraordinary computational requirements, due to the very large numbers of synaptic weights that are involved (p. 121).

Woody (Bledsoe 1962b, 1962c, 1961a) analyzed how mating, mutations, and gene interaction would affect the convergence rate to an optimum. Goldberg (1989) assesses Woody's role in the creation of the genetic algorithm:

The studies of Bledsoe and Bremermann came closest to the modern notion of a genetic algorithm. Both suggested binary string codings. Bledsoe presented results of a scheme that combined individual-by-individual generation, mutation, and save-the-better selection (p. 104).

Another interesting result originating from the collaboration with Bremermann is found in Bledsoe (1962b). Using quantum theoretic methods, Bremermann had proposed a lower bound on the number of bits of information that could be contained in any computer of a given volume. Using the Heisenberg uncertainty principle, Bledsoe obtained a lower bound on the memory-access time of a computer of a fixed density and memory size. He shows, for example, that any computer that has a memory capacity of 10^9 bits of information and a density of 20 grams/cm³ must have an access time in excess of 10^{-18} sec. To overcome this limitation, Woody proposed that computers of the future might contain a great many local processing units distributed uniformly throughout the memory storage area and under the general control of a central processing unit.

During 1964 and 1965, Bledsoe, along with Helen Chan and Charles Bisson, worked on using the computer to recognize human faces (Bledsoe 1966a, 1966b; Bledsoe and Chan 1965). He was proud of this work, but because the funding was provided by an unnamed intelligence agency that did not allow much publicity, little of the work was published. Given a large database of images (in effect, a book of mug shots) and a photograph, the problem was to select from the database a small set of records such that one of the image records matched the photograph. The success of the method could be measured in terms of the ratio of the answer list to the number of records in the database. Bledsoe (1966a) describes the following difficulties:

This recognition problem is made difficult by the great variability in head

rotation and tilt, lighting intensity and angle, facial expression, aging, etc. Some other attempts at facial recognition by machine have allowed for little or no variability in these quantities. Yet the method of correlation (or pattern matching) of unprocessed optical data, which is often used by some researchers, is certain to fail in cases where the variability is great. In particular, the correlation is very low between two pictures of the same person with two different head rotations.

This project was labeled man-machine because the human extracted the coordinates of a set of features from the photographs, which were then used by the computer for recognition. Using a GRAFACON, OF RAND TABLET, the operator would extract the coordinates of features such as the center of pupils, the inside corner of eyes, the outside corner of eyes, point of widows peak, and so on. From these coordinates, a list of 20 distances, such as width of mouth and width of eyes—pupil to pupil, were computed. These operators could process about 40 pictures an hour. When building the database, the name of the person in the photograph was associated with the list of computed distances and stored in the computer. In the recognition phase, the set of distances was compared with the corresponding distance for each photograph, yielding a distance between the photograph and the database record. The closest records are returned.

This brief description is an oversimplification that fails in general because it is unlikely that any two pictures would match in head rotation, lean, tilt, and scale (distance from the camera). Thus, each set of distances is normalized to represent the face in a frontal orientation. To accomplish this normalization, the program first tries to determine the tilt, the lean, and the rotation. Then, using these angles, the computer undoes the effect of these transformations on the computed distances. To compute these angles, the computer must know the three-dimensional geometry of the head. Because the actual heads were unavailable, Bledsoe (1964) used a standard head derived from measurements on seven heads.

After Bledsoe left PRI in 1966, this work was continued at the Stanford Research Institute, primarily by Peter Hart. In experiments performed on a database of over 2000 photographs, the computer consistently outper-

formed humans when presented with the same recognition tasks (Bledsoe 1968). Peter Hart (1996) enthusiastically recalls the project with the exclamation, "It really worked!"

The Mormon Church

Woody was a devout member of the Church of Jesus Christ of Latter Day Saints, more commonly known as the Mormon Church. In his youth, he joined several churches but became disillusioned with them. In his autobiographical sketch, he declared, "I was determined to have nothing to do with any of them ever again" (Bledsoe 1976). After marrying Virginia, who was a Mormon, Woody began attending church with her. Woody did not join immediately, however. He described his hesitation: "I really got involved in the Mormon Church in Richmond. I expected to join then but just did not have the testimony, so I delayed" (Bledsoe 1976). It took Woody another 15 years to join. When he did, the church became a focus in his life. The Mormon Church does not have paid clergy; its leadership positions are occupied by laymen who have other jobs. The church expects its members to take an active role. Woody twice served two-year appointments as bishop of the congregation. The role of bishop is similar to that of a pastor in other Protestant religions or a priest in the Catholic Church. The bishop is responsible for the administration of the congregation as well as the welfare of each member of the congregation. The Mormons generally do not accept government aid, and it is the bishop's role to determine what church assistance the congregation members need. During his first term as bishop, his administrative talents were also being challenged as chair of the Mathematics Department.

Woody also served as a counselor to the stake president, the administrative head of a stake, which is a local collection of congregations. While serving these appointments, Woody was also a scout leader. His last church appointment was as stake patriarch. The patriarch gives patriarchal blessings to church members. These blessings provide inspired guidance and direction. Although a bishop and a stake president must be deemed worthy individuals, they are chosen in large part because of their dedication and administrative skills. The patriarch is selected because he lives life according to the highest moral and spiritual principles—he is always a revered individual.

In experiments performed on a database of over 2000 photographs, the computer consistently outperformed humans when presented with the same recognition tasks (Bledsoe 1968). Peter Hart (1996) enthusiastically recalls the project with the exclamation, "It really worked!"



University of Texas

In 1965, Bledsoe decided to return to university life:

By 1965 it looked like PRI was doing pretty well, and I saw no reason why we couldn't make some money with it. But as president, I was spending more and more of my time administering, going to Washington, Albuquerque, or wherever, and less time with the thing I loved, namely science. Also, I had a latent desire to get back to university life, so I accepted an offer as Professor of Mathematics at the University of Texas at Austin (Bledsoe 1976).

Lockingen, one of Bledsoe's partners at PRI, had left PRI for the University of Texas (UT) in 1963. Wilson Stone, a prominent geneticist at UT and an expert on speciation, had met Bledsoe several years previously and had been trying to woo Bledsoe to Texas. Stone, vice-chancellor of research, was in the process of forming a quantitative genetics faculty and wanted Bledsoe as a principal member. Bledsoe finally accepted Stone's invitation and joined the Mathematics Department in January 1966.

During the early 1970s, Bledsoe collaborated with R. H. Richardson, a geneticist who

was also brought to UT by Stone. They worked on a project to provide the geneticist with automated tools for discovering near replications of large chunks of genetic material on the large chromosomes of *Drosophila*. One of the primary goals of this work was to analyze the hypothesis that speciation occurred by modifying and reusing large functional assemblies of genes. Thus, within the chromosomes of an organism should be many large groups of genes that occur several times in slightly modified form. Some of these groups should also be found across species.

At the time, genetic material on these chromosomes was identified primarily by the banding patterns. The bands at a particular locus, however, would vary from cell to cell and from one developmental stage to another as genes were turned on and turned off. The computer was enlisted to process the microscopic images of the chromosomes; straighten them out; and, with the aid of the geneticist, identify the bands. The computer could then try to find a match for specified regions in the same or other chromosomes from the sample or from chromosomes in a database of samples. In this project, as in the earlier faces project and as we see in later theorem-proving projects, Bledsoe has the human in the loop.

Bledsoe was twice appointed to two-year stints as chair of the Mathematics Department. Woody recalled that "things were buzzing in the Math Department at UT when I came and I was soon in the middle of it" (Bledsoe 1976). When Woody first became chair in 1967, the department was divided into two camps: One group followed the teaching methods of R. L. Moore, and the other used more conventional methods. Moore, who in 1967 was in his late 80s, used the Socratic method, where teaching was accomplished exclusively by means of question and answer. There were no lectures, and the use of books was forbidden. According to Leonard Gillman (1996), who succeeded Woody as chair, "The Moore group tried to prevent the library from buying any mathematics books." Although Moore had his share of detractors, he also had his share of supporters. He could count among his former students three members of the National Academy of Sciences. Elaborating on the buzzing metaphor, Gillman (1996) said that "Woody really walked into the hornet's nest but somehow managed to hold the whole thing together." Woody was entirely sympathetic to the Moore method. In fact, as an undergraduate at UT, he decided to become a mathematician, during a Moore-style course, because of

the thrill he derived by proving himself the theorem that the Riemann integral exists for continuous functions. Woody's second appointment began in 1973. By 1975, the math faculty had doubled in size.

Students

Woody was an active teacher and a thesis adviser. The following individuals received their Ph.D. under him; many of them went on to become first-rate scientists: John Wade Ulrich, 1968, computer sciences; Stephen Charles Darden, 1969, computer sciences; Charles Edward Wilks, 1969, mathematics; James Bertram Morris, 1969, computer sciences; Robert Brockett Anderson, 1970, mathematics; Robert Stephen Boyer, 1971, mathematics; Dallas Sylvester Lankford III, 1972, mathematics; Vesko Genov Marinov, 1973, computer sciences; Mark Steven Moriconi, 1977, computer sciences; John Threecivelous Minor, 1979, computer sciences; Peter Leonard Bruell, 1979, computer sciences; William Mabry Tyson, 1981, computer sciences; Tie Cheng Wang, 1986, computer sciences; Larry Marvin Hines, 1988, computer sciences; Don Simon, 1990, computer sciences; and Guo Wei Feng, 1994, computer sciences.

MCC

In 1983, a group of 20 computer companies formed a consortium for the purpose of doing advanced research in computer science. This consortium was, in part, a response to the perceived threat of the Japanese fifth-Generation Project, the goal of which was to build high-performance inference machines for running AI applications. Austin, Texas, was selected as the site for the consortium, and Bobby Ray Inman, former deputy director of the Central Intelligence Agency and former director of the National Security Agency, was chosen to lead the effort. Among the major research areas were packaging, software technology, database, natural language, and AI.

In response to Woody's offer of assistance, Inman proposed that he become a vice president in charge of the AI group. Woody accepted the job in the spring of 1984 and quickly got to work building the AI group. One of the most conspicuous projects was *CYC*, headed by Doug Lenat, with whom Woody had collaborated while on sabbatical at Carnegie Mellon University. Lenat (1996) describes how *CYC* came about as well as what its goals were:

Woody and Virginia spent a semester's sabbatical visiting me at CMU in

1977–78, where I was an assistant professor, and we kept in touch during my ensuing years as a Stanford professor. In the early 80s, I published a series of four "Nature of Heuristics" articles in the *AI Journal*, which gradually came to the conclusion that machine learning, theorem proving, natural language and speech understanding, etc., were all hitting a brick wall, for lack of a large commonsense knowledge base. When he came to work for Bob Inman, as head of AI at MCC, in early 1984, Woody realized that this was exactly the sort of grand enterprise that the MCC consortium was set up to pursue. He gave me a call, and I came and talked with him about my dream, and met Inman. The combination of the two of them was irresistible, not to mention the long-term high-risk high-payoff charter of MCC itself, and I moved to Austin and started the *CYC* Project.

CYC, which lasted as an MCC project for more than 10 years, involved dozens of staff members (ontological engineers) explaining the everyday world in a formal language that computers could manipulate. In January 1995, it spun off from MCC as an independent company, Cycorp, whose charter is to commercialize and foster the widespread use of the technology, both for the expected applications (such as English front ends) and for others that we didn't expect (such as database integration).

Woody's former student and fellow UT professor Robert Boyer took a leave from UT and also joined Woody at MCC.

IJCAII, AAI, and I Had a Dream

Woody served the AI professional organizations in various administrative roles. He was chairman of the International Joint Conference on Artificial Intelligence, Inc. (IJCAII), in 1977 and a member of the IJCAII board of trustees from 1978 to 1983.

Bledsoe was named president-elect of the American Association for Artificial Intelligence (AAAI) in 1983. The following year, he helped organize the annual AAAI meeting in Austin, Texas. In 1985, he gave the presidential address at the annual meeting held at the University of California at Los Angeles (Bledsoe 1986). He began his address with a description of his conversion to AI:

Twenty-five years ago I had a dream, a daydream if you will. A dream shared with many of you. I dreamed of a spe-

cial kind of computer, which had eyes and ears and arms and legs, in addition to its “brain.”

When I awoke from this daydream, I found that we didn’t have these things but we did have some remarkable computers, even then, so I decided then and there to quit my job and set about spending the rest of my life helping bring this dream to reality (p. 57).

He then described how many of the grand goals were still unaccomplished: “SHAKEY liked shaking more than running” (Bledsoe 1986, p. 58). He emphasized that progress had been made and that progress will be made, albeit slowly, if researchers persist in their efforts.

First, let me express my annoyance with some of the distracted individuals who criticize AI researchers for not “jumping to infinity” in one leap. Somehow, to them it is OK to work step by step on the dream of obtaining controlled thermonuclear energy or a cure for cancer or a cure for the common cold, but no such step-by-step process is allowed for those trying to (partially) duplicate the intelligent behavior of human beings. To these cynics, a natural language system that converses with us in a restricted form of English is somehow not a legitimate step toward passing the Turing test. I know of no case in the history of science where such “naysayers” actually helped with a new discovery (Bledsoe 1986, p. 58).

Bledsoe then enumerated several areas he felt should be researched. Among these were large knowledge bases, machine learning, and analogical and commonsense reasoning. He closed his address with some advice to young researchers (Bledsoe 1986, p. 61):

The principle I want to make is this: when you have what looks like a good idea, give it your best shot, waste a little money to get some early feedback. Don’t take forever to study the problem, because that is even more expensive (and less exciting). Of course, this strategy (this scientific method) requires character on the part of the researcher. He/she must be willing to analyze those experiments, reformulate theories, and press on. Otherwise, that person does not qualify for the work and should not be entrusted with research funds.

So, again, I would say to young people, set a dream. Set a goal (your part of bringing about the dream). Tool up: edu-

cation, employment, facilities. Pursue it with vigor—and impatience. Want it today. I’ve never seen a content researcher who was worth his salary. Don’t be easily deterred by those who don’t have your insight and training. Work hard, provide momentum, don’t give up easily. Don’t spend too much time extolling the work of others; you will never be properly recognized or satisfied until you make your own personal contribution. Compare and compete. These are rules for a researcher in any field. My conviction is that the field of AI is worth your finest efforts.

Return to the University

Woody returned to UT in 1987, where he remained until his retirement in 1994. Woody held the Ashbel Smith Professorship of Mathematics and Computer Science for several years and the Peter O’Donnell Jr. Chair in Computing Systems from 1987 to his retirement. In 1991, he received the IJCAI Distinguished Service Award and the third “milestone” award from the American Mathematical Society. These awards are presented to individuals who perform outstanding research in the area of automatic theorem proving. The citation on the award reads

The Milestone Award of 1991 goes to Woodrow W. Bledsoe, who has been a central figure in Automatic Theorem Proving, inspiring and guiding this field for over twenty years. His broad view of the subject, using resolution and nonresolution techniques, his deep study of theorem proving in analysis with inequalities, and his work on interactive theorem provers distinguish him as a major innovator in the field.

In 1991, a conference celebrating Woody’s seventieth birthday was held at the university, with researchers coming from around the world. The papers presented at this conference, as well as another biographical account of Bledsoe’s life, are found in Boyer (1991).

Automated Reasoning

Bledsoe spent part of the summer of 1966 back at Sandia, where he began to implement a proof checker for A. P. Morse’s (1965) set theory with his friend E. J. Gilbert. In discussing this work (Bledsoe and Gilbert 1967), Bledsoe makes the following observation, which was to provide the underlying impetus for almost all his work in automated reasoning:

Much of Bledsoe’s efforts were spent in getting his mechanical provers to make larger and larger inference steps.

Logic systems usually state a certain minimal set of rules of inference, which might include detachment (modus ponens), substitution, and universalization (generalization), and from which all theorems are derivable. However, if just this minimal set of rules of inference is used, proofs become very long and tedious, and consequently, provers usually begin making larger steps in their proofs. This tendency to larger steps is increased as one proceeds further away from elementary logic into other areas of mathematics, until a sort of “steady-state” proof step size is reached (p. 6).

Much of Bledsoe’s efforts were spent in getting his mechanical provers to make larger and larger inference steps.

When Bledsoe began his work in automatic theorem proving or automated reasoning, nearly all research in this area was based on the resolution procedure (Robinson 1965). To determine whether a statement follows from a set of axioms with resolution, one places the axioms and the negation of the statement into a normal form, called *clause form*, and, using a simple rule of inference (resolution) that combines modus ponens with matching, generates new clauses from old clauses. These new clauses are called *resolvents*. After all possible resolvents are generated from the original set, they are thrown into the pot, and the algorithm proceeds by resolving these new resolvents against the original clauses and one another. This process can repeat round after round indefinitely. If the empty clause is eventually generated (by resolving S and $\text{not } S$), then the original statement is a theorem. Much of resolution’s appeal stems from its simplicity and the fact that it is a *complete proof procedure*, meaning that if a statement S logically follows from a set of axioms, then resolution applied to the clause form of the axioms, along with the negation of S , will eventually generate the empty clause and pronounce S to be a theorem. This simplicity came with a price; a naive implementation of this basic procedure soon swamps the computer with generated clauses.

Researchers applying the resolution method tried to limit the explosion of generated clauses. One of the first of these restrictions, or *refinements*, was the set of support strategy (Wos, Robinson, and Carson 1965) that forbid the generation of new clauses by resolving between clauses derived solely from the axioms. When limiting the generation of new clauses, it is important to ensure that the restrictions are not so great that the algorithm

fails to prove statements that are theorems, thus the emphasis on completeness proofs.

Along with his student Robert Anderson, Bledsoe invented a powerful restriction (Anderson and Bledsoe 1970). They introduced (Anderson and Bledsoe 1970) the *excess literal method*, a simple but general approach for establishing completeness of resolution restrictions. Boyer (1971) discovered a resolution restriction called *locking* and proved its completeness also using the excess literal method. Although these restricted resolution algorithms tend to generate fewer clauses on each round, it often takes more rounds to find a proof if one exists. Thus, what is saved on breadth is often conceded to depth.

Because the semantics of the equality relation introduces axioms that generate so many resolvents, the resolution method, as originally described, could not deal effectively with the equality relation. Woody’s student James B. Morris (1969) overcame this limitation by defining an extension of resolution called *E-Resolution*. The basic idea behind E-Resolution is that when trying to unify two terms, if at some point they don’t match, try to use occurrences of the equality predicate to rewrite them into terms that do match. Notice that E-Resolution essentially builds equality handling into the unification algorithm.

During the late 1960s and 1970s, the majority of theorem-proving researchers focused on improving the performance of the resolution method. The primary approach was to improve the efficiency of the new clause-generation process itself. Much improvement was made along this line by cleverly indexing clauses or compiling them. A direct result of clause compilation was the Prolog programming language based on David H. Warren’s (1987) beautiful abstract machine. Some impressive theorems were produced by these fast inference machines, but typically, they were of a restricted form. The initial clause set was always small, meaning that inference proceeded from a small hand-picked set of axioms or other theorems. Defined terms such as continuous function were usually expanded away in advance, and the theorems typically did not require algebraic manipulations. In recent years, improvements in resolution search strategies and implementation techniques have led to provers that have settled open questions in mathematics (Wos and McCune 1992; also see URL http://www.mcs.anl.gov/home/mccune/ar/new_results).

Although Bledsoe continued to work at improving resolution, most of his time was spent studying nonresolution methods, as

described in his survey article (Bledsoe 1977b). In this article, he describes how he became disenchanted with his earlier approach:

The author was one of the researchers working on resolution type systems who “made the switch.” It was in trying to prove a rather simple theorem in set theory by paramodulation and resolution, where the program was experiencing a great deal of difficulty, that we became convinced that we were on the wrong track. The addition of a few semantically oriented rewrite rules and subgoaling procedures (Bledsoe 1971) made the proof of this theorem, as well as similar theorems in elementary set theory, very easy for the computer. Put simply, the computer was not doing what a human would do in proving this theorem. When we instructed it to proceed in a “human-like” way, it easily succeeded (p. 2).

Bledsoe used the term *nonresolution* in a general sense. Rather than implying a particular logical proof method such as natural deduction, Bledsoe used nonresolution to mean any technique whatsoever that could be considered mathematical reasoning. Bledsoe was a working mathematician who wanted to build mechanical tools for assisting mathematicians. Proving well-formulated theorems is only a small part of a mathematician’s activities. Someone once remarked, “To say that a mathematician’s job is proving theorems is like saying that a writer’s job is writing sentences.” Among a mathematician’s activities are formulating theorems to be proved and defining terms of mathematical interest. A mathematician looks at examples. A mathematician computes when necessary. A mathematician constantly refers to a large corpus of mathematical and commonsense knowledge. Bledsoe considered all these activities worthy of study under the automatic theorem proving or automated reasoning umbrella and indeed investigated a number of these areas.

The UT PROVER

The primary research vehicle for this exploration of nonresolution theorem proving was a program called the UT PROVER (Bledsoe and Tyson 1975; Bledsoe and Bruell 1974). This prover was based on a natural deductionlike logic. PROVER worked by splitting a problem into subgoals, backtracking whenever it looked to be on a wrong path. The program was designed to have the human in the loop. Among the many methods the human could employ to guide the proof were to specify

lemmas to add to the hypothesis, specify a term to define away, provide instantiations for existential variables, and abort the current goal. This prover and its variants were used over the course of nearly a decade. Some interesting applications were proofs of the standard-limit theorems from analysis (Bledsoe, Boyer, and Henneman 1972) using the limit heuristic and the proofs of many difficult theorems in analysis (Ballantyne and Bledsoe 1977) using nonstandard analysis. PROVER, augmented with special rules to handle integer linear inequalities (Bledsoe 1975), was the proof engine for a very large program-verification project (Good 1985; Good, London, and Bledsoe 1975).

In Ballantyne and Bledsoe (1982), PROVER was used to study how mathematicians create examples and counterexamples to conjectures and how they are used to guide proofs. The program tried to mimic a mathematician at a board drawing pictures of a problem. It was successful at discovering minimal finite counterexamples to a number of conjectures in topology.

We list a few of the areas in automated reasoning where Bledsoe and his students made contributions. Many of these methods were used either in PROVER or in resolution proof systems.

Simplification

Although an equation such as $x + 0 = x$ can in theory be used in either direction, in practice, it is almost always used left to right. An intuitive justification is that the term resulting from such an application is simpler. Similarly, the equation $x * (y + z) = x * y + x * z$ is also almost always used from left to right, although the criteria by which one can say the resulting term is simpler are not as obvious as in the previous case.

A group of equations that are used only in one direction for the purpose of simplifying expressions is called a *set of simplifiers* or *reducers*. When simplifying an expression, all the simplifiers are applied to each term in the expression until no more simplifications can be made. Notice that the act of reducing a formula is a large inference step. The use of simplifiers in automatic theorem proving was pioneered by Wos (Wos et al. 1967), Bledsoe (1971), and Slagle (1974). Simplification was used as one of the primary computational tools by Bledsoe’s student Boyer in the BOYER-MOORE PROVER (Boyer and Moore 1988, 1979).

To be truly useful, a set of simplifiers E should have the property that if the equality $s = t$ follows from E viewed as a set of equa-

tions, then the fully reduced form of s should be identical to the fully reduced form of t . Given an arbitrary system of equations, the reduced are not identical. For example, the axioms of group theory

1. $(x + y) + z = x + (y + z)$
2. $x + (-x) = 0$
3. $x + 0 = x$

do not have this property. It is an elementary theorem of group theory that $0 + x = x$. However, both sides of this equation are fully reduced. Thus, if one uses these equations solely as simplifiers, then valid group-theoretic equalities will not be identified as such. In a landmark paper (Knuth and Bendix 1970), Donald Knuth formalized the notion of simplifier and described a means for testing a set of simplifiers to see whether it has the property described. Moreover, in the case where it fails, Knuth's test will often transform the original set into an equivalent set that does. Such sets are called *complete sets of reducers* (or *complete term-rewriting systems*). In a series of papers, Bledsoe's student Dallas Lankford (1975a, 1975b) and Michael Ballantyne (Lankford and Ballantyne 1977) generalized Knuth's work and incorporated it into general proof procedures such as resolution.

Chaining

Although early researchers in automated reasoning concentrated their efforts on proving theorems in obscure formal systems such as ternary Boolean algebras, Bledsoe early on focused on proving hard theorems in analysis (Bledsoe 1990, 1984; Bledsoe, Boyer, and Heneman 1972). We have already seen how he used simplifiers to handle the arithmetic axioms of the real numbers. The order axioms, such as transitivity of the less-than predicate, presented an equally difficult challenge for resolution systems. The transitivity axiom states that for real numbers a , b , and c , if a is less than b and b is less than c , then a is less than c . One difficulty with this axiom when presented to resolution is that it resolves with itself, producing longer and longer clauses. Another problem is that virtually any other clause containing the less-than predicate will resolve with it. Another axiom of the real numbers that is often required in proofs states that between any two real numbers, there is another real number. If these axioms are allowed to participate unchecked in an attempt at a resolution proof of some theorem in analysis, they will almost immediately swamp the computer.

Bledsoe (1989) devoted an entire paper to illustrating the pathology of these axioms when used in a resolution setting. To combat this proof devastation, Bledsoe proposed the variable-elimination and chaining rules (Bledsoe and Hines 1980), which, when incorporated into resolution, eliminate the need for explicitly adding the axioms mentioned. (This statement is only true if the less-than relation is total, as it is for real numbers.) Although this new super inference rule is simple in form, proving its completeness was difficult. A partial result along these lines was obtained in Bledsoe, Kunen, and Shostak (1985), and the proof was completed by Bledsoe's student Larry Hines (1992). We list some of the theorems proved by this method (Hines and Bledsoe 1990): First, the limit of sums is the sum of limits. Second, the composition of continuous functions is a continuous function. Third, nonempty, closed, and bounded sets have a maximum element. Fourth is the intermediate-value theorem. A similar approach to dealing with transitivity in elementary set theory and the theory of the integers also showed considerable promise (Hines 1994, 1990; Bledsoe 1975).

Set-Variable Instantiation

When proving theorems in many branches of mathematics, one must often have to find a set that satisfies certain properties. For example, when using the least-upper-bound axiom of the real numbers, one must instantiate the axiom with some bounded set. Huet (1973) described a complete proof procedure for a formulation of higher-order logic. Unfortunately, this procedure suffered from all the inefficiencies of resolution and then some. For example, two terms could possibly have an infinite number of most general unifiers. Bledsoe (1977a) discovered a method that lacks the generality of Huet's method but that works well enough on a subset of second-order logic to enable his provers to prove many theorems in topology and analysis that required the instantiation of a set variable. The method was called a *maximal method* because it always instantiates the set variable with the largest possible solution. Bledsoe (Bledsoe and Feng 1993) extended the method and proved a form of completeness.

Analogy

Bledsoe's last years focused primarily on the task of building analogical reasoning into his provers. He considered this task one of the most important tasks confronting the field. Analogy is the heart and soul of intelligent

Bledsoe's last years focused primarily on the task of building analogical reasoning into his provers.

behavior. We do very little that is absolutely new (Bledsoe and Hodges 1988).

Bledsoe (1995) described a program that proves theorems by modifying analogous proofs of other theorems:

The basic idea is as follows: when a person is proving a theorem, they [sic] often make a number of decisions ("let this," "let that," etc.), and sometimes makes a mistake on one of these early decisions. They do not discover that there is a problem until "deep" in the proof, and then has to decide which of these early decisions has to be altered and how. A similar situation occurs in analogy proofs when trying to use a "guiding" proof to construct a new proof. To cope with this problem, detected deep in a proof, the PC prover sends back up through the proof a message on what is needed to complete the proof, namely a "precondition," and attempts to make the appropriate changes (p. 226).

Thus, the program selects a guiding theorem from a database of previously proved theorems, maps the symbols to be proved onto the symbols in the guiding theorem, attempts to follow the guiding proof using the hypotheses of the target theorem, and returns weakest preconditions for certain of the proof steps. A program implementing these ideas succeeded in proving the ground completeness of lock resolution using the excess literal proof of the completeness of general resolution as the guide. The first pass through the proof returned the weakest precondition, which stated a restriction on which literals to select when splitting a clause. Armed with this new hypothesis, the program succeeded on the second pass. When Bledsoe refers to a database of theorems, he does not mean a database of resolution proofs. The proofs are presented at a high level, as an outline of the essential steps of the theorem. To be useful as a guide, a theorem must provide the essential ideas without constraining every step.

Final Days

In the spring of 1993, Woody noticed a slurring of his speech. A neurologist diagnosed the cause as nerve degeneration as a result of Lou Gehrig's disease. In spite of his diminishing capabilities, he never complained, and he seemed never to despair. Always the scientist, Woody made tapes of his speech so that he could chronicle the progress of the disease. Even though his strength and coordination continued to deteriorate, he played tennis

until he felt he could no longer do so without hurting himself. He taught until he thought his poor speech impaired his ability to lecture, and he pursued his research on analogy until he could no longer hold a pen.

He was perhaps the most wonderful person we have ever known. He was the sort of perfect embodiment of faith, hope, and charity. It's almost as though he was not subject to the common human frailties. We never saw him discouraged, we never saw him depressed, and we never saw him unkind.

Leonard Gillman (1996) described Woody thusly:

He was universally calm and fair—a model citizen of the world.

Acknowledgments

We want to thank Pam Bledsoe for organizing Woody's autobiographical notes and Virginia Bledsoe for several discussions about Woody. We finally want to thank Woody himself for providing constant inspiration, encouragement when needed, criticism when deserved, and many years of love and friendship.

References

- Anderson, R., and Bledsoe, W. W. 1970. A Linear Format for Resolution with Merging and a New Technique for Establishing Completeness. *Journal of the Association for Computing Machinery* 17:525–534.
- Ballantyne, A. M., and Bledsoe, W. W. 1982. On Generating and Using Examples in Proof Discovery. In *Machine Intelligence* 10, eds. J. Hayes and D. Michie, 3–39. Chichester, U.K.: Harwood.
- Ballantyne, A. M., and Bledsoe, W. W. 1977. Automatic Proofs of Theorems in Analysis Using Non-standard Techniques. *Journal of the Association for Computing Machinery* 24(3): 353–374.
- Bing, R. H.; Bledsoe, W. W.; and Mauldin, R. D. 1974. Sets Generated by Rectangles. *Pacific Journal of Mathematics* 51(1): 27–36.
- Bisson, C. 1962. Optimization of n -Tuple Memory Matrices Using a Gradient Technique, PRI Internal Memo, Panoramic Research Inc., Palo Alto, California.
- Bledsoe, W. W. 1995. A Precondition Prover for Analogy. *Biosystems* 34:225–247.
- Bledsoe, W. W. 1990. Challenge Problems in Elementary Analysis. *Journal of Automated Reasoning* 6:341–359.
- Bledsoe, W. W. 1989. Using Hyper-Resolution on Problem One of LIM+, Technical Report, ATP-91, Automatic Theorem Proving Project, University of Texas at Austin.
- Bledsoe, W. W. 1986. I Had a Dream: AAAI Presidential Address, 19 August 1985. *AI Magazine* 7(1): 57–61.
- Bledsoe, W. W. 1984. Some Automatic Proofs in Analysis. *Contemporary Mathematics* 29:89–118.

- Bledsoe, W. W. 1977a. A Maximal Method for Set Variables in Automatic Theorem Proving. In Proceedings of the Fifth International Joint Conference on Artificial Intelligence, 501–510. Menlo Park, Calif.: International Joint Conferences on Artificial Intelligence.
- Bledsoe, W. W. 1977b. Non-Resolution Theorem Proving. *Artificial Intelligence* 9:1–35.
- Bledsoe, W. W. 1976. My Personal History. Unpublished manuscript. Manuscript obtained from daughter Pam Bledsoe.
- Bledsoe, W. W. 1975. A New Method for Proving Certain Presburger Formulas. In Proceedings of the Fourth International Joint Conference on Artificial Intelligence, 15–21. Menlo Park, Calif.: International Joint Conferences on Artificial Intelligence.
- Bledsoe, W. W. 1971. Splitting and Reduction Heuristics in Automatic Theorem Proving. *Artificial Intelligence* 2:55–77.
- Bledsoe, W. W. 1968. Semiautomatic Facial Recognition, Technical Report SRI Project 6693, Stanford Research Institute, Menlo Park, California.
- Bledsoe, W. W. 1966a. Man-Machine Facial Recognition: Report on a Large-Scale Experiment, Technical Report PRI 22, Panoramic Research, Inc., Palo Alto, California.
- Bledsoe, W. W. 1966b. Some Results on Multicategory Pattern Recognition. *Journal of the Association for Computing Machinery* 13(2): 304–316.
- Bledsoe, W. W. 1964. The Model Method in Facial Recognition, Technical Report PRI 15, Panoramic Research, Inc., Palo Alto, California.
- Bledsoe, W. W. 1962a. A Basic Limitation on the Speed of Digital Computers. *IRE Transactions on Electronic Computers* EC-10(3): 530.
- Bledsoe, W. W. 1962b. An Analysis of Genetic Populations, Technical Report, Panoramic Research, Inc., Palo Alto, California.
- Bledsoe, W. W. 1962c. The Evolutionary Method in Hill Climbing: Convergence Rates, Technical Report, Panoramic Research, Inc., Palo Alto, California.
- Bledsoe, W. W. 1961a. Lethally Dependent Genes Using Instant Selection, Technical Report PRI 1, Panoramic Research, Inc., Palo Alto, California.
- Bledsoe, W. W. 1961b. The Use of Biological Concepts in the Analytical Study of Systems, Technical Report PRI 2, Panoramic Research, Inc., Palo Alto, California. Also 1961. Presented at ORSA-TIMS National Meeting, San Francisco, California, November 10.
- Bledsoe, W. W. 1952a. Neighborly Functions. *Proceedings of the American Mathematical Society* 3(1): 144–145.
- Bledsoe, W. W. 1952b. Some Aspects of Covering Theory. *Proceedings of the American Mathematical Society* 3(5): 804–812.
- Bledsoe, W. W., and Bisson, C. 1962. Improved Memory-Matrices for the N-Tuple Pattern Recognition Method. *IRE Transactions on Electronic Computers* EC-11:414–415.
- Bledsoe, W. W., and Browning, I. 1959. Pattern Recognition and Reading by Machine. In Proceedings of the Eastern Joint Computer Conference, 225–232.
- Bledsoe, W. W., and Bruell, P. 1974. A Man-Machine Theorem-Proving System. *Artificial Intelligence* 5:51–72.
- Bledsoe, W. W., and Chan, H. 1965. A Man-Machine Facial Recognition System—Some Preliminary Results, Technical Report PRI 19A, Panoramic Research, Inc., Palo Alto, California.
- Bledsoe, W. W., and Feng, G. 1993. SET-VAR. *Journal of Automated Reasoning* 11:293–314.
- Bledsoe, W. W., and Gilbert, E. H. 1967. Automatic Theorem Proof-Checking in Set Theory—A Preliminary Report, Research Report SC-RR-67-525, Sandia Laboratories, Sandia, New Mexico.
- Bledsoe, W. W., and Hines, L. M. 1980. Variable Elimination and Chaining in a Resolution-Based Prover for Inequalities. In *Proceedings of the Fifth Conference on Automated Deduction*, 70–87. New York: Springer-Verlag.
- Bledsoe, W. W., and Hodges, R. 1988. A Survey of Automated Deduction. In *Exploring Artificial Intelligence, Survey of Talks from the National Conference on Artificial Intelligence, Seattle, 1987*, ed. H. Shrobe, 483–543. San Francisco, Calif.: Morgan Kaufmann.
- Bledsoe, W. W., and Morse, A. P. 1963. A Topological Measure Construction. *Pacific Journal of Mathematics* 13(4): 1067–1084.
- Bledsoe, W. W., and Morse, A. P. 1955. Product Measures. *Transactions of the American Mathematical Society* 79(1): 173–215.
- Bledsoe, W. W., and Tyson, M. 1975. The UT Interactive Prover, Technical Report, APT-15, Automatic Theorem Proving Project, University of Texas at Austin.
- Bledsoe, W. W., and Wilks, C. E. 1972. On Borel Product Measures. *Pacific Journal of Mathematics* 42(3): 569–579.
- Bledsoe, W. W.; Boyer, R. S.; and Henneman, W. H. 1971. Computer Proofs of Limit Theorems. *Artificial Intelligence* 3:27–60.
- Bledsoe, W. W.; Kunen, K.; and Shostak, R. 1985. Completeness Results for Inequality Provers. *AI Journal* 27:255–288.
- Bledsoe, W. W.; Norris, M. J.; and Rose, G. F. 1954. On a Differential Inequality. *Proceedings of the American Mathematical Society* 5(6): 934–939.
- Boyer, R. S. 1991. *Automated Reasoning: Essays in Honor of Woody Bledsoe*. Boston: Kluwer.
- Boyer, R. S. 1971. Locking: A Restriction of Resolution. Ph.D. diss., Department of Mathematics, University of Texas at Austin.
- Boyer, A. O., and Boyer, R. S. 1991. A Biographical Sketch of W. W. Bledsoe. In *Automated Reasoning: Essays in Honor of Woody Bledsoe*, R. S. Boyer, 1–31. Boston: Kluwer.
- Boyer, R. S., and Moore, J. S. 1988. *A Computational Logic Handbook*. New York: Academic.
- Boyer, R. S., and Moore, J. S. 1979. *A Computational Logic*. New York: Academic.
- Bremermann, H. J. 1962. Optimization through

- Evolution and Recombination. In *Self-Organizing Systems*, eds. M. T. Yovits, Jacobi, and Goldstein. Washington, D.C.: Spartan.
- Bremermann, H. J., and Anderson, R. W. 1991. How the Brain Adjusts Synapses—Maybe. In *Automated Reasoning: Essays in Honor of Woody Bledsoe*, R. S. Boyer, 119–147. Boston: Kluwer.
- Edelman, G. M. 1987. *Neural Darwinism*. New York: Basic.
- Gillman, L. 1996. Personal Communication with Authors, January 1996, Austin, Texas.
- Goldberg, D. E. 1989. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Reading, Mass.: Addison-Wesley.
- Good, D. I. 1985. Mechanical Proofs about Computer Programs. In *Mathematical Logic and Programming Languages*, eds. C. A. R. Hoare and J. C. Shepherdson, 55–75. New York: Prentice-Hall.
- Good, D. I.; London, R. L.; and Bledsoe, W. W. 1975. An Interactive Verification System. *IEEE Transactions on Software Engineering* 1:59–67.
- Hart, P. 1996. Personal Communication with Authors, January 1996, Palo Alto, California.
- Hines, L. M. 1994. STR+VE and Integers. In *Proceedings of the Twelfth International Conference on Automated Deduction*, 416–430. New York: Springer-Verlag.
- Hines, L. M. 1992. Completeness of a Prover for Dense Linear Order. *Journal of Automated Reasoning* 8:45–75.
- Hines, L. M. 1990. STR+VE-Subset: The STR+VE-Based Subset Prover. In *Proceedings of the Tenth International Conference on Automated Deduction*, 193–206. New York: Springer-Verlag.
- Hines, L. M., and Bledsoe, W. W. 1990. The Str+ve Prover, Technical Report, ATP-94, Automatic Theorem Proving Project, University of Texas at Austin.
- Holland, J. H. 1975. *Adaptation in Natural and Artificial Systems*. Ann Arbor, Mich.: University of Michigan Press.
- Huet, G. 1973. A Mechanization of Type Theory. In *Proceedings of the Third International Joint Conference on Artificial Intelligence*, 139–146. Menlo Park, Calif.: International Joint Conferences on Artificial Intelligence.
- Knuth, D. E., and Bendix, P. B. 1970. Simple Word Problems in Universal Algebras. In *Computational Problems in Abstract Algebras*, ed. J. Leech, 263–297. Oxford, U.K.: Pergamon.
- Lankford, D. S. 1975a. Canonical Inference, Technical Report, ATP-32, Automatic Theorem Proving Project, University of Texas at Austin.
- Lankford, D. S. 1975b. Complete Sets of Reductions for Computational Logic, Technical Report, ATP-21, Automatic Theorem Proving Project, University of Texas at Austin.
- Lankford, D. S., and Ballantyne, A. M. 1977. Decision Procedures for Simple Equational Theories with Commutative-Associative Axioms: Complete Sets of Commutative-Associative Reductions, Technical Report, ATP-39, Automatic Theorem Proving Project, University of Texas at Austin.
- Lenat, D. 1996. Personal Communication with Authors, January 1996, Austin, Texas.
- Morris, J. B. 1969. E-Resolution: Extension of Resolution to Include the Equality Relation. In *Proceedings of the First International Joint Conference on Artificial Intelligence*, 287–294. Menlo Park, Calif.: International Joint Conferences on Artificial Intelligence.
- Morse, A. P. 1965. *A Theory of Sets*. New York: Academic.
- Robinson, J. A. 1965. A Machine-Oriented Logic Based on the Resolution Principle. *Journal of the Association for Computing Machinery* 12:23–41.
- Slagle, J. 1974. Automated Theorem Proving with Simplifiers, Commutativity, Associativity. *Journal of the Association for Computing Machinery* 21:622–642.
- Warren, D. H. D. 1987. Implementing PROLOG—Compiling Predicate Logic Programs, Technical Report, 39(40), Department of Artificial Intelligence, University of Edinburgh.
- Wos, L., and McCune, W. 1992. The Application of Automated Reasoning to Questions in Mathematics and Logic. *Annals of Mathematics and Artificial Intelligence* 5:321–370.
- Wos, L.; Robinson, G.; and Carson, D. F. 1965. Efficiency and Completeness of the Set of Support Strategy in Theorem Proving. *Journal of the Association for Computing Machinery* 12:276–310.
- Wos, L.; Robinson, G.; Carson, D.; and Shalla, L. 1967. The Concept of Demodulation in Theorem Proving. *Journal of the Association for Computing Machinery* 14:698–709.

Michael Ballantyne is currently working on software reengineering for EDS. He worked as a research scientist with Woody in the Mathematics Department at the University of Texas at Austin and at MCC.

Robert S. Boyer is a professor in the computer sciences, mathematics, and philosophy departments at the University of Texas at Austin. See <http://www.cs.utexas.edu/users/boyer/>. The most recent version of the Boyer-Moore theorem prover, together with thousands of example theorems and definitions, can be obtained at <ftp://ftp.cli.com/pub/nqthm/>.

Larry Hines is currently working for EDS Research and Development. His research principally involves automating proof discovery in transitive orders. He has developed first-order theorem provers for set theory, integral inequalities, and dense inequalities. He also worked for Woody at MCC.