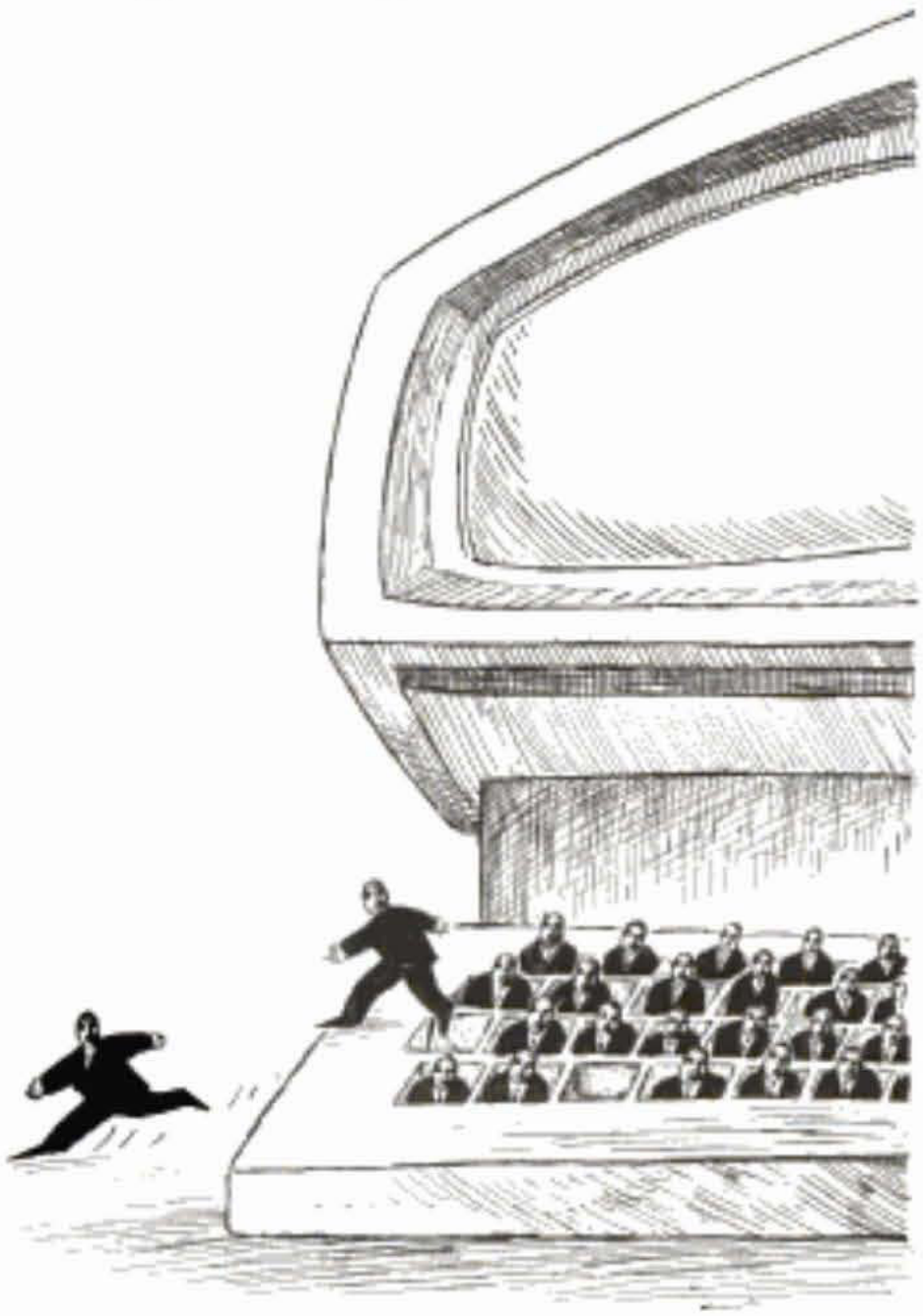
Bulletin of the Atomic Scientists, December 1984; 40 (10): pp. 11–15

Strategic computing

The Department of Defense Advanced Research Projects Agency (DARPA) was formed in the late 1950s to promote basic research. Indeed, DARPA's Information Processing Techniques Office, headed by distinguished computer scientists, has established itself as the principle government sponsor of computer research at universities and industrial laboratories. Much of this research has been generic in nature—applicable to a large variety of military and nonmilitary problems. But in October 1983, DARPA launched a new "Strategic Computing Plan" with the express purpose of focusing research on specific military applications. Because of the broad influence that DARPA exercises on the direction of computer research in this country, such a pronounced shift of purpose deserves public scrutiny. The authors contend that the Strategic Computing Plan is dangerously misleading, because it blurs the distinction between straightforward progress in computer science and mere wishful thinking. The plan's suggestion that "artificial intelligence" will enable strategic nuclear weapons to be handled almost entirely by computer illustrates the serious consequences that could result if policy makers begin to depend upon technological fantasy.

by Severo M. Ornstein, Brian C. Smith and Lucy A. Suchman



Paul Valerry, West Germany

IN THE 1940s, atomic physics was about 25 years old. Building on the discoveries of the new field, scientists were able to produce a weapon more powerful than had ever before been conceived. In the 1980s computer science—which also happens to be about 25 years old—has become the critical field underlying modern weapon systems. This is not yet widely recognized. When we think of nuclear weapons, we tend to envision the warheads and the explosions, forgetting about the complex computer technology that supports the decision to fire the missiles and directs them to their targets. Computer systems are by now used throughout the military, for early warning, communications, weapons guidance and in the simulations with which targets are selected and battles planned.

DARPA's Strategic Computing Plan aims to develop a new generation of computing technology for military applications. The plan initiates a five-year, $600 million program, and there is good reason to believe that this is just the beginning. The proposal contains plans for developing an underlying technology base of new hardware and software. The hardware emphasis will be on microelectronics and multiprocessor architectures, from which the Agency hopes to obtain at least a thousand-fold increase in net computing power. The software component focuses on artificial intelligence—particularly on what is known as expert systems—to provide machines with "human-like, intelligent capabilities" including natural language understanding, vision, speech and various kinds of automated reasoning.[1]

On top of this technology base, three specific military applications are to be developed. For the Army, the plan proposes a class of "autonomous vehicles," able not only to move around independently, but also to "sense and interpret their environment, plan and reason using sensed and other data, initiate actions to be taken, and communicate with humans or other systems." For the Air Force, the suggestion is a "pilot's associate" to aid aircraft operators who are "regularly overwhelmed by the quantity of incoming data and communications on which they must base life or death decisions," in tasks ranging from the routine to those that are "difficult or impossible for the operator altogether" and require the "ability to accept high-level goal statements or task descriptions." Finally, the Navy is offered a "battle management system," "capable of comprehending uncertain

Severo M. Ornstein is a computer scientist and chairman of Computer Professionals for Social Responsibility, based in Palo Alto, California. Brian C. Smith, who teaches computer science and philosophy at Stanford University, and Lucy A. Suchman, an anthropologist, are members of the research staff at the Xerox Palo Alto Research Center.

data to produce forecasts of likely events, drawing on previous human and machine experience to generate potential courses of action, evaluating these options, and explaining the supporting rationale." These three applications are intended to illustrate the power of the technology; we are also asked to imagine "completely autonomous land, sea, and air vehicles capable of complex, far-ranging reconnaissance and attack missions."

Two facts stand out:

- The Strategic Computing Plan proposes the use of artificial intelligence technology in military systems in order to provide a radically new kind of flexibility and adaptiveness. Referring repeatedly to the increasing speed and unpredictability of modern warfare, the plan promises that computing technology can be developed capable of adapting to "unanticipated enemy behavior in the field."[2] This will require "a new generation of military systems" that could "fundamentally change the nature of future conflicts." The change involves both increasing the amount of computation and enlarging its role to include automation of military decision-making.
- There are specific proposals about how to direct computer science research. Rather than letting researchers follow their own course, the plan aims to focus them on military objectives. Various mechanisms are suggested to do this, such as a close coupling of fundable research goals and military needs, adherence to strict development timetables and the selection of specific development projects intended to "pull the technology-generation process." (The Army, Navy and Air Force projects cited above are the first examples.)

In assessing the Strategic Computing Plan, our concern is not with the underlying technology base or with military projects as such. Nor do we question the power of artificial intelligence as a new and important technology. Our concern is that increased reliance on artificial intelligence and automated decision-making in critical military situations, rather than bringing greater security, leads in an extremely dangerous direction. Specifically, the plan creates a false sense of security in the minds of both policy-makers and the public. Like all computer systems artificial intelligence systems may act inappropriately in unanticipated situations. Because of this fundamental limit on their reliability, we argue against using them for decision-making in situations of potentially devastating consequence.

Automation and uncertainty

Modern warfare is marked by three interacting trends: increasingly powerful weapons; more separation, in both time and space, between planning and execution; and a faster and faster pace. The first means that the consequences of our actions, intended or unintended, can be greater than ever before. The second means that we rely on increasingly large, complex and indirect systems for command, control and communication. The third means that any miscalculation can quickly lead to massive ramifications which are difficult, perhaps impossible, to control. It is easy to see the dangerous potential of the three in combination. They are all the direct product of technological developments in offensive and defensive weapons systems. And they have brought us to the situation that we live with now: two nations confronting each other with forces that, if unleashed, would destroy both in less than an hour.

This danger is recognized on all sides; people differ only in what they think we can or should do about it. But if anything is universally accepted, it is that the current state is precarious. And into this situation the Strategic Computing Plan proposes to introduce artificial intelligence as a new ingredient:

> Improvements in the speed and range of weapons have increased the rate at which battles unfold, resulting in a proliferation of computers to aid in information flow and decision making at all levels of military organization. . . . A countervailing effect on this trend is the rapidly decreasing predictability of military situations, which makes computers with inflexible logic of limited value. . . . Confronted with such situations, leaders and planners will . . . be forced to rely solely on their people to respond in unpredictable situations. Revolutionary improvements in computing technology are required to provide more capable machine assistance in such unanticipated combat situations. . . . Improvements can result only if future computers can provide a new quantum level of functional capabilities.

What this means in plain English is: Faster battles push us to rely more on computers, but current computers cannot handle the increased uncertainty and complexity. This means that we have to rely on people. But without computer assistance, people can't cope with the complexity and unpredictability, either. So we need new, more powerful computer systems.

In observing that increased uncertainty and confusion are critical problems of modern warfare, the Strategic Computing Plan accepts the situation as inevitable, embracing artificial intelligence and automatic decision-making as a means of coping with it. The decisions to be automated, furthermore, are not minor; the Plan makes clear that reliance on automatic systems is meant to include the control of strategic weapons. For example:

> Commanders remain particularly concerned about the role that autonomous systems would play during the transition from peace to hostilities when rules of engagement may be altered quickly. An extremely stressing example of such a case is the projected defense against strategic nuclear missiles, where systems must react so rapidly that it is likely that almost complete reliance will have to be placed on automated systems. At the same time, the complexity and unpredictability of factors affecting decisions will be very great.

The Plan offers no argument to warrant this reliance on automatic decision-making. Although computers have contributed to more effective weapon systems and will continue to do so, it doesn't follow that we can automate the complex

processes of assessment and judgment. There is a long and still unresolved debate within the computer profession about what we should expect of artificial intelligence. But there is agreement that it is still in its infancy. The first systems based on the technology are just beginning to be used, in highly controlled and delimited circumstances. But the problem isn't just one of immaturity. Rather, it is that the Plan expects reliable decision-making in circumstances where there may simply be no way to achieve it, with computers or with people.

The limits of reliability

Any computer system, however complex, and whether or not it incorporates artificial intelligence, is limited in the scope of its actions and in the range of situations to which it can respond appropriately. This limitation is fundamental and leads to a very important kind of failure in reliability—beyond the obvious troubles of transistors shorting out or systems breaking down. Those failures are serious enough in and of themselves, but there is a much more intractable kind of failure, having to do with limitations of design. Computers are maddeningly literal-minded; they do exactly what we program them to do. Unfortunately, except in trivial cases, we cannot anticipate all the circumstances they will encounter. The result is that, in unexpected situations, computers will carry out our original instructions, but may utterly fail to do what we intended them to do.

The ballistic missile warning systems of the United States (and presumably those of the Soviet Union) regularly give false alarms of incoming attacks.[3] Although most of these alerts are handled routinely, on a number of occasions they have triggered the early stages of a full-scale reaction. These false alerts stem from causes as varied as natural events, in one case a moonrise, in another a flock of geese; failures in the underlying hardware, such as a faulty integrated circuit chip that started sputtering numbers into a message about how many missiles were coming over the horizon; and human errors, such as when an operator mounted a training tape onto the wrong tape drive, thereby causing the system to react seriously to what was intended to be a simulation. The primary insurance against accidents resulting from this kind of failure has been the involvement of people with judgment and common sense. So far, there has always been enough time for them to intervene and prevent an irretrievable, and perfectly real, "counterattack."

Despite these lessons, the Strategic Computing Plan promotes the view that the human element in critical decision-making could be largely, if not totally, replaced by machines. This would require that computers embody not only "expert knowledge" but also common sense and practical reasoning. Such capabilities, however, are beyond the state of the art. Expert systems are so called because they capture some of the specialized knowledge that an expert has acquired—not because they surpass the abilities of the rest of us generally. Despite much work, there hasn't been much progress in automating plain old common sense.

What distinguishes common-sense reasoning is the ability to draw on an enormous background of experience in the most unpredictable ways. In directing a friend to your house, for example, you don't have to give instructions about all the possible things that might happen along the way: fallen trees, accidents, flat tires. Similarly, if you were to say "The city council didn't give the demonstrators a permit because they feared violence," you would expect your audience to know "they" refers to the councillors, not to the demonstrators. The point is that a vast range of knowledge and experience may be relevant; we never know what we'll need, or when we'll need it. Nor do we usually even notice that we are using this background knowledge. These facts undermine any attempt to codify common sense and practical reasoning. Current expert systems don't have the common sense of even a small child.

In terms of their fundamental limitations, artificial intelligence systems are no different from other computer systems. Computers carry out, with lightning speed and unparalleled accuracy, rules that a human programmer has coded in advance. It is the job of programmers and system designers to try to anticipate the range of situations that a computer system will encounter, and to provide recipes for all the possible actions that it should take in those situations. This planning is designed so that the computer can recognize the particular situation that does in fact arise and select an appropriate response. Because of its great speed, the computer will typically be able to select a response very rapidly.

This all sounds very promising. Designers plan carefully so that the computer can respond instantly when it matters most. And it often works very well, as in the case of the computers that control the phone system, help to land aircraft and provide missile guidance. But the behavior of the system depends entirely on the structure of the program—on how it is put together. Classical computer systems not only have rigidly pre-specified rules, but put them together in brittle and inflexible ways. What distinguishes artificial intelligence and expert systems, and gives them the "flexibility" so touted by the Strategic Computing Plan, is that they facilitate more productive interaction of the rules. But they continue to rely on the programmer's ability to state the rules in advance. And to do so, the programmer must first develop a conceptual structure appropriate to a given problem area.

The rules on which all computer systems are based, in other words, treat the world as if it were built from a stock of pre-defined building blocks, assembled in carefully prescribed ways. Artificial intelligence systems are particularly good at dealing with very complex configurations of these building blocks, often better than more traditional computer programs. But they are ill-equipped to respond appropriately to new kinds of blocks. They work best in areas that are well understood, highly constrained, predictable and easily controlled.

In more complex environments, unanticipated events are liable to trigger anomalous reactions. That is why the radar reflections off the rising moon fooled the North American

Air Defense system; moons were not among the pre-defined building blocks. The system had no way to say "Oh, yes, I forgot about the moon," because it had no common sense to underlie its set of domain-specific rules. Even worse, computer systems don't "know" that they are encountering an event outside the scope of the assumptions on which they were built; they merely sort every event into the pre-specified set of categories. Not only was the moonrise not recognized as such; it was mistaken for something quite different.

All complex systems, including artificial intelligence systems, have to evolve for a substantial period before they are reliable enough to be used. Any first version will invariably contain flaws, some of which will be obvious as soon as the system is installed. Other more subtle problems will surface only after it has been used for some time in a wide variety of situations. During this evolution, the system makes many, often serious, errors, some of which require substantial modifications to correct. These errors, furthermore, may interact; the "fix" to one problem will often introduce another, more subtle problem. In this process, perfection is never achieved; the best one can hope for is to reduce to an acceptable level the rate at which new flaws reveal themselves. The system will then be described as "reliable" and may lead us to a sense of security. Even in the most reliable systems, however, residual flaws, although improbable, may still surface with dramatic effects.

The 1965 Northeast power failure demonstrates how a large system containing hidden design flaws can run trouble-free for years and suddenly collapse under unexpected circumstances. In that case the problem stemmed from simultaneous lightning strikes affecting separate parts of the system. By design, the system tried in each case to absorb the load elsewhere, causing a series of further overloads that eventually interacted to bring down the whole Northeast power grid. On October 27, 1980, a similar problem in the nationwide computer communications network known as the ARPANET brought all communication to an abrupt halt.[4] While they usually have less dramatic consequences, such problems arise in all computer systems.

Computer systems that achieve a sufficient level of reliability to be used in real applications do so because they have been heavily tested beforehand in the laboratory. After being installed in their particular domain, they are observed, extended and corrected to meet real-world conditions. No amount of simulation can replace the testing that comes from embedding the system in the actual environment for which it was designed. The reason is straightforward: simulated tests exercise exactly those circumstances that the designers expect the system to encounter. It is the designers, after all, who build the simulators, based on the same understanding of the problem area used to build the system in the first place. But all experience with complex systems indicates that it is the circumstances we fail to anticipate that cause the serious problems.

One obvious solution is to provide ways for human operators to intervene and override the default system behavior. But this too is a problem; we just don't know yet how to build large systems with enough human interactions to make the combination reliable. Given a person capable of perfectly adequate performance in a domain without machine assistance, and a supporting machine capable of adequate performance on its own, the performance of the combined "system" is often quite poor because of problems in the interaction. Three Mile Island is perhaps the best known example.

Finally, when a computer system is intended for use under crisis conditions, all of the standard problems are likely to be highly aggravated. The behavior of any system is only as predictable as the behavior of the people and technology that make it up. Yet human behavior in situations of fear and confusion—such as war—is notoriously unpredictable. Systems designed for use in a crisis should be thoroughly tested before one begins to rely on them. Yet there is no way that military systems—especially nuclear systems—can be fully tested in advance; nor can crisis conditions ever be fully simulated. As the Strategic Computing Program points out, it is the unpredictability of war that poses the gravest threat.

The myth of technological solutions

If the uncertainty of battle is so serious, and if computer systems are so unreliable, why should the Computing Plan propose computer technology as a solution? The easiest explanation seems to be a version of "If we *can* do it, we *should* do it." If there is some possibility that we can build new military systems, especially powerful new computing systems, we must try to do so.

There are also more subtle answers. Sophisticated artificial intelligence systems are scientifically intriguing; they enable us to explore areas of human capability in which we have enormous interest, including those areas that are relevant to coping with uncertainty. The hope that these systems could cope with uncertainty is understandable, since there is no doubt that they are more flexible than traditional computer systems. Understandable, but wrong, because in the end the increased flexibility is limited by the same inexorable facts that limit all computer systems.

Over the years, the lure of artificial intelligence has led to a growing appetite for research funding. The appetite, in turn, has led the professional community to make promises, many of which have turned out to be more difficult to fulfill than was anticipated. For example, it was widely believed in the 1950s that we would soon have fully automatic machine translation, an accomplishment that still eludes us. These unfulfilled promises are frequently a combination of ordinary naivete, unwarranted optimism and a common if regrettable tendency to exaggerate in scientific proposals. Shortcomings are often masked by subtle semantic shifts. When we fail to instill "reasoning" or "understanding" in our machines, we tend to adjust the meaning of these terms to describe what we have in fact accomplished. In the process, we obscure the real meaning of our claims for artificial intelligence.

When these claims are taken literally, without appropriate qualification, they give rise to unrealistic confidence in the power of the technology. Policy-makers, even those close to the profession, are not immune to such misconceptions. Witness the following discussion of Defense Department research on space-based weapon systems, as reported in the *Los Angeles Times* on April 26, 1984:

> The fireworks began when a panel that included Robert S. Cooper, director of the Defense Advanced Research Projects Agency, George Keyworth, Reagan's science adviser, and Lt. Gen. James A. Abrahamson, director of the Strategic Defense Initiative, acknowledged that a space-based laser system designed to cripple Soviet long-range missiles in their 'boost' phase would have to be triggered on extraordinarily short notice.
>
> To strike the boosters before they deployed their warheads in space would require action so fast that it might preclude a decision being made in the White House—and might even necessitate a decision by computer, the panel said.
>
> At that, Sen. Paul E. Tsongas (D-Mass.) exploded: 'Perhaps we should run R2-D2 for President in the 1990s. At least he'd be on line all the time.'
>
> 'Has anyone told the President that he's out of the decision-making process?' Tsongas demanded.
>
> 'I certainly haven't,' Keyworth said.
>
> Sen. Joseph R. Biden, Jr. (D-Del.) pressed the issue over whether an error might provoke the Soviets to launch a real attack. 'Let's assume the President himself were to make a mistake. . . .' he said.
>
> 'Why?' interrupted Cooper. 'We might have the technology so he couldn't make a mistake.'
>
> 'OK,' said Biden. 'You've convinced me. You've convinced me that I don't want you running this program.'

Cooper's final comment betrays a belief that computers are competent to take over critical decisions and might correct deficiencies in human judgment as well. As the discussion shows, common sense suggests that these claims are implausible. It might have been that common sense was wrong—that the underlying science had advanced beyond the layperson's expectations. But we believe that the skepticism is in fact well founded.

To cope with problems of complexity and speed in modern warfare, the Strategic Computing Plan proposes a quantum leap in computer technology, comparable to the advent of nuclear weapons technology in the 1940s. Ironically, the problems arise in part from the very technology that is proposed as a solution. Past attempts to achieve military superiority by developing new technology, rather than increasing our security, have brought us to the present untenable situation. The push to develop so-called "intelligent" weapons as a way out of that situation is another futile attempt to find a technological solution for what is, and will remain, a profoundly human political problem. ☐

1. Unless otherwise noted, quotations are from *Strategic Computing.* "New Generation Computing Technology: A Strategic Plan for its Development and Application to Critical Problems in Defense," Defense Advanced Research Projects Agency (Oct. 28, 1983).
2. *Electronic News* (March 19, 1984), p. 18.
3. See, for example, the Hart-Goldwater report to the Committee on Armed Services of the U.S. Senate: "Recent False Alerts from the Nation's Missile Attack Warning System" (Washington, D.C.: U.S. Government Printing Office, Oct. 9, 1980); Physicians for Social Responsibility, *Newsletter*, "Accidental Nuclear War," (Winter 1982), p. 1.
4. Eric Rosen describes this event in ACM SIGSOFT, "Software Engineering Notes," 6, no. 1 (Jan. 1981).