SOCIETAL AND TECHNOLOGICAL PROBLEMS OF COMPUTERS

Qi JIANG
Danny KOPEC

Department of Computer Science, University of Maine, Orono, Maine 04469, U.S.A.

Abstract. In this paper we are going to examine several factors, both societal and technological, that account for the slowdown of computer production and applications. Our basic premise is that there are serious problems associated with the use of computers in industry. It is the responsibility of both businessmen and engineers alike to solve these problems in order to create a better society for us to live in. Computers, like other machines, have their own limitations and are not the panacea, as they may appear to be to some people, for all problems.

Keywords: software, hardware.

1 INTRODUCTION

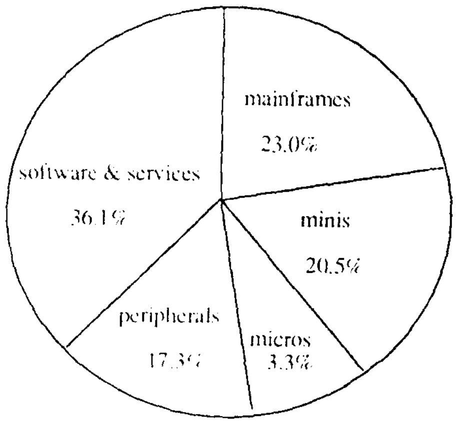
In the 1970s and 80s the computer industry became a cornerstone of the American economy. Every major indicator in this sector — sales, production, and employment — has shown strong and consistent growth. On an international level, U.S. firms have occupied a position of overwhelming superiority, controlling some 75-80% of the world computer market during the 1970s. Fig. 1. illustrates the huge U.S. computer sales.

One traditional application area for computers is database systems where a large amount of information is stored and retrieved at a relatively fast speed. Another major application is realtime systems where programs are written to control specific devices such as automobiles, aircraft, and nuclear power stations. Recently, artificial intelligence, as a branch of computer technology, has gained wide application. Expert systems are used to diagnose diseases, locate oil wells, and so on.

In the last two decades there has been a widespread "computer fever". Some people thought of them as magic machines, and sometimes labelled computers as "electronic brains" because they were supposed to be as smart as human beings. The term "artificial intelligence" became very fashionable. There was fear that the robots controlled by computers would take over control of the whole society. The debate has been ongoing as to whether or not computer scientists should develop such super machines. In other words, the computer and its pervasive manifestation were a "hot" topic at that time.

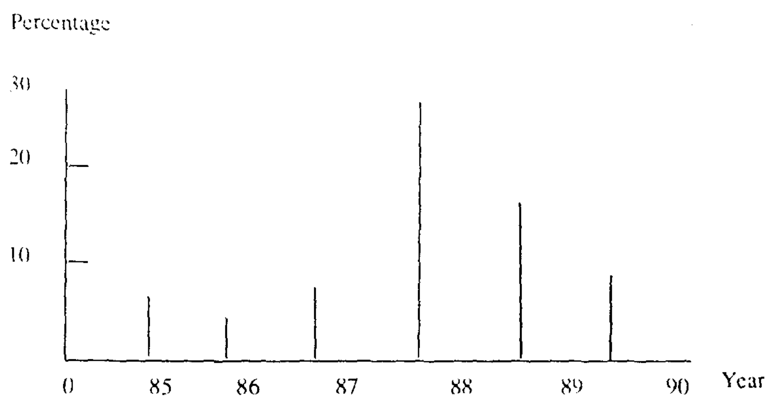
Since the late 1980s, however, the computer industry has been experiencing a relatively sluggish growth. Contrary to popular belief, there has been steady slackening of demand for personal computers, turmoil in mid-range machines, and relatively flat sales for large mainframes. In 1990, IBM made plans to shed 10,000 jobs in the U.S. The goal: cut expenses by $1 billion and boost sagging profits by $1 per share. Fig. 2 is from *Businessweek* [2].

1986 U.S. Computer Sales (worldwide)



Total values: $108.3 billion

Fig. 1



slower growth for computers
annual growth in U.S. system shipment

Fig. 2

Computer sales are down because people do not find them as useful as expected, and also due to experience difficulties in learning how to use them in business. Many personal computers, especially machines bought during the personal computer boom of 1983 and 1984, sit idle much of the time. In fact, according to US Bureau of Labor Statistics figures, overall office productivity in the US is no higher than it was in the late 1960's. William Bowen, the technology writer wrote: "Recent research by the Brookings Institution think-tank and by investment bankers Morgan Stanley has also found it hard to discern any improvements in office productivity, yet capital expenditure on computers has zoomed. While it is possible to find many individual success stories and while it is plainly evident that many commercial sectors are now wholly dependent on computers to service their customers, it seems that demonstrating the economic benefits in macro terms is much more difficult" [1].

2 THE SOCIETAL PROBLEMS OF COMPUTERS

2.1 VDT can be hazardous to health

Within the past decade a great change has swept the workplace, with nearly universal adoption of computers and word processors. An estimated 10 million Americans — travel agents, clerks, secretaries, students, and professors — now spend their days with keyboards and video display terminals (VDTs). There have been complaints of various health problems and fatigue associated with the use of VDTs.

As time passes there is mounting evidence that VDTs can cause eyestrain, headaches, miscarriages and birth defects. In management, trade union and feminist circles, the reported health and safety hazards of VDTs have become a big issue.

On July 23, 1980 the *Toronto Globe and Mail* carried an article under the headline "Work Conditions Probed at [*Toronto*] *Star* as Defects Found in 4 Employees' Babies" [4]. Four women in the classified advertising department of the *Star* had given birth the previous autumn to children with birth defects. All four mothers had worked with VDTs during the early stages of their pregnancies.

A 1987 Japanese study by Professor Satoshi Ishikawa of Kitazato University found that 90 percent of VDT users reported eyesight problems, 17 percent suffered from eyeball degeneration, 27 percent reported stiff shoulders and necks, and 13 percent complained of insomnia [5].

In 1986, Berhard Tribukait's group at the Karolinska Institute in Stockholm reported a significant increase in malformed fetuses in mice exposed to pulsed magnetic fields like those produced by the transformers in VDTs [13]. Gunnar Walinder's group at the Swedish University of Agricultural Sciences in Uppsala in 1987 also reported significantly more fetal deaths in pregnant mice exposed to magnetic fields. A US Office of Naval Research group reported a significant increase in abnormalities among chicken embryos exposed to a low frequency magnetic field [5].

Probably the most important report on VDT hazards to emerge in recent years came in June 1987 from the Kaiser-Permanente Medical Care Program in Oakland, California [5]. Kaiser-Permanente studied 1,600 women clerical workers who had become pregnant since 1984 and found that expectant mothers who had spent more than 20 hours per week at terminals were more than twice as likely to suffer a miscarriage as other clerical employees.

People are beginning to realize the serious problems associated with the use of VDTs. Legislation regulating the use of VDTs has been introduced in 22 states [4]. More recently, San Francisco passed a law which requires employers to provide proper lighting, non-glare screens, and work-time breaks for their VDT operators. But the consensus of a 1981 National Academy of Science (NAS) Coference was that the application of ergonomics — in the form of better lighting, improved seating and more appropriate screen technology — would only go part way toward solving both the physical an psychological problems of VDT users and that job stress was major causal factor.

This legislation has already met and will meet resistance from the business people, who argue that the bill would cost them too much. We, however, believe that in no case should individual health be sacrificed for an increase in productivity.

2.2 Affect of computers on the quality of work

A major lesson of the IT (Information Technology) story is that the productivity pay-off from computerization has been somewhat disappointing. In manufacturing, commerce, government and elsewhere, productivity gains have often been hard to discern, despite massive spending on IT equipment.

Researchers who have studied the problem of white-collar productivity offer many explanations for the poor pay-off from computers. One suggestion is that stress in computerized offices is costing millions of dollars in lost working hours and reduced productivity. In the US, for example, the Office of Technology Assessment (OTA) has estimated that stress-related illnesses cost business between $ 50 billion and $ 75 billion per year. In her book, *The Electronic Sweatshop* (1988), Barbara Garson argues that computers are transforming the "office of the future" into a kind of stressed-out "factory of the past" [6].

Another issue causing cosiderable controversy at present is that of the computerized monitoring of employees. According to a recent Office of Technology Assessment report to the US Congress, between 25 and 30 percent of US clerical employees are now under surveillance in this way [3]. In the past, employees were monitored directly by progress-chasers, foremen and supervisors. These days such monitoring can be done by computers.

Computerized monitoring is constant, reliable and cheap. But critics charge that it is actually counter-productive because employee morale declines and with it productivity.

Canadian researchers Rebecca Grant, Christopher Higgins and Richard Irving found that monitoring had a dramatic effect on employee attitudes towards productivity and customer service [7]. Generally, they report that the effect of computerized monitoring was to degrade the quality of the product offered to the customer and the work environment in the process. Whereas no less than 85 percent of unmonitored employees rated work quality (customer service and teamwork) as the most important factor in their jobs, as many as 80 percent of monitored employees said production quantity was most important.

Peter G. Neumann suggests that computers can increase fatigue and boredom, because so many tasks associated with computers are highly repetitive: data entry, tape library maintenance and audit-trail watching [9]. It is quite evident that once people get bored it creates oportunities for accidental misuse or intentional abuse. Such errors in high stress socio-critical domains, i.e. air traffic control, nuclear power stations, robot controlled assembly lines, etc. can lead to the most hazardous accidents. The problems for operators in these industries (and others with similar characteristics) is that much of their time is spent performing routine tasks such as monitoring and data entry. These are followed by sudden bursts of activity where they must make critical decisions possibly affecting the entire plant/system and many lives.

Computers also increase the sense of depersonalization. People using computers tend to spend long hours in front of a terminal trying to figure out a programming solution or enter a huge amount of data into a database. This reduces the amount of time spent with other people. Further, Neumann states: "Interactions with computers tend to depersonalize both the user community and application itself. The resulting sense of anonymity can inspire a lack of respect for the system and its resources, and a diminished sense of ethics, values, and morals on the part of the affected people" [9]. If this is indeed the case then there can be more serious consequences.

3 THE TECHNOLOGICAL PROBLEMS CREATED BY COMPUTERS

3.1 We cannot trust our computer software

Since computers have been used in military, industrial plants, and business operations, there are numerous reports of computer failures which have caused either loss of life or financial losses. Most of the problems have been caused by software bugs that hide in long line of codes. Software failures are ubiquitous. In the following sections we examine their effect from the prospective of economic, health, and complexity issues.

3.1.1 Economic factors

A study in 1979 by US Government's General Accounting Office showed that of nine federal software projects which cost a total $ 6.8 million, projects worth $ 3.2 million (47 percent of the total value) were delivered but not used, $ 2 million worth were paid for but not delivered, $ 1.3 million were abandoned or reworked and just $ 200,000 worth were used after substantial modification [5]. Incredibly, just one project worth less than $ 100,000 was used as delivered by the developer. Recent reports from Logica and Price Waterhouse have also indicated that poor software quality costs the United Kingdom about $ 900 million per year.

3.1.2 Health hazards: case studies

In 1980, a man undergoing microwave arthritis therapy died when the therapy reprogrammed his pacemaker. It has also been reported that another man was severely affected when his pacemaker was reprogrammed by interference from an anti-theft device in a store [5]. He subsequently died as a result of the trauma.

Malfunctions occurred in 1985 and 1986 at the East Texas Cancer Center in Tyler, Texas, and at the Kennestone Regional Oncology Center in Marietta, Georgia. A subtle **software error** caused some body areas to receive between 17,000 and 25,000 rads, where research has shown that doses as low as 1,000 rads when delivered to the whole body can be fatal. Typical therapeutic doses to small areas of the body range from between 4,000 and 6,000 rads delivered in 20 or 30 treatments over a month or more.

Software inadequacies can have a detrimental effect on the health of individuals, but eventually it may be demonstrated that one particular software problem can affect the future health and lifestyle of every person on the earth — because of limitations in the programs onboard NASA observation satellites used during the 1970s and 1980s. These programs actually rejected the ozone readings they were registering at the time — because they were so low, they were regarded as spurious. It was only when British scientists, using ground-based instruments, reported that a decline in ozone levels was occurring, that NASA scientists reprocessed data going back to 1979 and confirmed the British findings [10]. It is quite possible that our treatment and handling of the current ozone layer problem, and its ultimate effects on the planet, could have been substantially improved had the NASA systems been able to provide us with warnings a decade earlier.

3.1.3 Software complexity

We have read many stories about accidents caused by software failures. The bad news is that at present time our computer scientists or engineers are unable to produce software that is exempt from error. In other words large complex software systems are inherently prone to errors.

Barbara Liskov, Professor of Computer Science at the Massachusetts Institute of Technology, gives the following example to illustrate the difficulties with software testing [8]. Suppose there is a program with three integer inputs, each of which ranges over the values 1 to 1,000. One way to make sure that the program is free of bugs is exhaustive testing, which would require more than 1 billion times of execution through the entire data set. If each run took 1 second, this would take slightly more than 31 years. This is akin to Perrow's [11] notion of complex interaction in a system (i.e. Complex system failures often occur as a result of multiple interactions of components which could not have been tested or anticipated).

With advances in software engineering, software developers are trying to use object-oriented design methodologies to cope with large complex software. However, that approach depends on formal specification, which is precise but hard to understand. Another method of eliminating software errors includes program verification, but computer scientists have only developed techniques for verifying small programs. Recently some high level programming languages such as ADA have introduced a new language feature called Exception Handling, which is meant to create a **robust program**. When errors occur, such a program should provide some approximation to its behavior in the absence of an error. Computer scientists have termed this "graceful degradation". Again such a technique still cannot guarantee that a program will always work in its expected way.

3.2 AT & T breakdown and some other major software disasters

On January 15, 1990 millions of Americans dialed long distance and got nowhere. The phone was silent, or even worse, kept spewing out the annoying message that "All circuits are busy." They were not busy, they had stopped working. Airline and hotel reservation numbers went dead. Catalog shoppers couldn't order what they wanted. For nine hours, AT & T's telephone network virtually collapsed nationwide, delaying or aborting more than half its telephone traffic. The telephonic debacle was not due to a burned out computer or a balky transistor. It was caused by the mysterious failure of a complicated computer software program.

In recent years the competition in the long distance telephone service has becme very intense. Various companies are selling their services at cheaper prices to win customers. AT & T chose to offer sound reliability as a way to attract more customers. That is why AT & T's glitch on January 22, 1990 — which left 65 million calls unconnected — hurt so badly [12]. Most business analysts predicted that AT & T would lose some market share because when the AT & T network crashed many companies which had a second carrier switched to MCI or US Sprint.

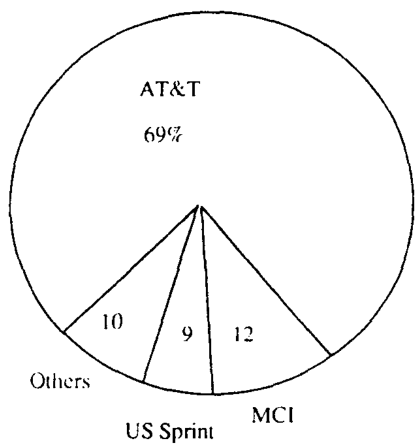
The AT & T glitch was caused by a computer software error. It took nine hours for Bell Laboratory engineers to locate and temporarily fix a problem in the new advanced signaling network, called Signaling System 7, that AT & T is in the process of installing nationwide.

The engineers ruled out a computer virus, and traced the problem to a faulty software program in one of the new signaling systems in lower Manhattan [15]. The program was used to determine the most efficient path for routing long-distance calls and had worked flawlessly for months, until some unforeseen and still unknown combination of telephone calls caused an error to occur. The software glitch should have caused the system to shut down automatically so that a backup system should take over. Instead it caused the system to keep operating and inform the other systems in the network that it was experiencing an overload. That message set off a chain reaction across the country and caused the whole system to crash.

Please note that some unforeseen and still unknown combination of telephone calls caused the error to show up. That just corresponds to our program testing example with three integer inputs (see Section 3.1.3), which means that it would have been impossible for programmers to test all possible inputs. A large software program can only be tested by actually trying ever conceivable combination of instructions to see whether it fails. The AT & T breakdown illustrates precisely the inherent problems that exist with large complex computer software. Our conjecture is that similar problems will occur in the future with perhaps more disastrous consequences to human life unless we adopt a new approach to our computer system solutions.

Market Share

ESTIMATED, 1989



SOURCE: NEWSWEEK, JANUARY 1990

Fig. 3

The AT & T breakdown is just one example of huge financial losses from software errors. The complex computerized financial networks that transfer trillions of dollars from coast to coast daily are consistently vulnerable to error. In 1985 a single software malfunction forced the Bank of New York to borrow $ 24 billion to cover its accounts temporarily; the cost was $ 5 million in extra interest. One programming error in Washington allowed ATM cardholders to make unlimited cash withdrawal regardless of their account balance [5].

4 WHAT SHOULD WE DO ABOUT OUR COMPUTER SYSTEM? SOME RECOMMENDATIONS

In light of all the problems discussed above (both societal and technological), should we just simply abandon the use of computers? The answer is NO. Our society depends heavily on computer technology; computers now control not only the nation's telephone communications, but everything from traffic lights and microwave ovens to the aircraft over Manhattan, therefore the impact of suddenly stopping the use of computers would be too great. Our conjecture is that computers should be used on a limited basis with great caution and awareness of the problems they pose. Computer research should also be more concentrated and goal oriented. Since VDTs can be very harmful to human health, computer scientists and engineers should try to create better screen technology or, alternatively, focus on the use of voice communications instead of the traditional keyboard/screen computer interface.

As engineers have shrunk computer hardware — chips, screens, keyboards —, the software, the coded instructions that tell computers what to do has ballooned in size and become less dependable. It is generally recognized that computer hardware is more reliable but the programs that run it are fraught with peril [12]. Since many functions performed by software can be hard-coded into microchips, it might be a good idea to have hardware as a substitute for software wherever possible. Computer hardware has the additional advantage of being fast and hard to copy in contrast to the increasing threat of software piracy.

REFERENCES

[1] Bowen, W.: The puny payoff from office computers. In: Forester, T. (Ed.): Computers in human context, Cambridge, Massachusetts, The MIT Press 1989.

[2] Businessweek, No. 3140, Jan. 1990, McGraw-Hill Inc.

[3] Federal Government Information Technology: Electronic surveillance and civil liberties. Office and Technology Assessment, US Congress, Washington, DC 1985.

[4] Foster, K.: The VDT debate. In: Forester, T. (Ed.): Computers in human context, Cambridge, Massachusetts, The MIT Press 1989.

[5] Forester, T.—Morrison, P.: Computer ethics. Cautional tales and ethical dilemmas in computing. Cambridge, Massachusetts, The MIT Press 1989.

[6] Garson, B.: The electronic sweatshop: how computers are transforming the office of the future into the factory of the past. New York, Simon and Schuster 1988.

[7] Grant, R. A.—Higgins, C. A.—Irving, R. H.: Computerized performance monitors: are they costing you customers? Sloan Management Review, Spring 1988.

[8] Liskov, B.—Guttag, J.: Abstraction and specification in program development. McGraw-Hill, The MIT Press 1986.

[9] Neumann, P.: Are risks in computer systems different from those in other technologies? Software Engineering Notes, Vol. 13, April 1988, No. 2.

[10] New York Times Science Section, 29 July 1986, p. C1.

[11] Perrow, Ch.: Normal accidents, living with high-risk technology. Basic Books Inc., New York 1984.