Computing Then & Now

Bruce L. Mann
Memorial University

The Morning Watch: Social and Educational Analysis

9/14/94, Memorial University Press
Abstract

This paper is a brief review of the history of educational computing and some of the issues surrounding the use of computers in education.

Computing Before Personal Computers

Two centuries ago, Leibnitz envisioned a calculating machine which embodied most of the essential features of recent keyboard devices. Financial considerations, however, prevented its production. In the nineteenth century, Charles Babbage developed a Difference Engine and his steam-driven Analytical Engine for creating mathematical tables. Maintenance costs, however, prevented Babbage from mass producing his arithmetical machines.

In 1936, Alan Turing described the nature and theoretical limitations of computers. Vannevar Bush speculated in 1945 that, "the world has arrived at an age of cheap, complex devices of great reliability and something is bound to come of it" (p. 25). And something did come of it. The first electrical computer was the ABC (Atanasoff-Berry Computer) developed in the early 1940's for solving linear algebra equations. Soon after the ABC, the ENIAC computer was developed with vacuum tubes for military and census purposes. The vacuum tubes were replaced in 1959 with transistors, and replaced again in 1964 with integrated circuits.

During the 1960's an increasing number of computers were used in education. Computers began appearing in American schools and universities. Patrick Suppes initiated a large math education project at Stanford University in 1963. By 1967, computer-assisted instruction (CAI) programs were developed across the curriculum for grades 1 through 6. Thousands of students were involved in the formative testing of these programs and research results were maintained on the students' achievement. Educational software such as PLATO (Programmed Operation for Automatic Teaching Operation) was developed at the University of Illinois and elsewhere, while SCHOLAR was an early "intelligent tutor" which attempted to simulate the behaviour of a human tutor. In 1977, the Apple II microcomputer was developed and marketed by Wozniak and Jobs.

Coincidentally, Cognitive Science also became established in the 1960's with the initial goal of using computers to imitate human intelligence. Cognitive Science emerged as one of the most dynamic new sciences of the twentieth century. The interdisciplinary nature of Cognitive Science extended to education with a mission to improve education by extending the frontiers of scientific understanding of the human mind. Work in Applied Cognitive Science impinged on education at several levels. It offered down-to-earth guidance for teaching, but at higher levels, it addressed questions such as what needed teaching and what was best acquired by other means. Applied Cognitive Science pointed to new possibilities for the design of educational approaches and technologies and for promoting new kinds of learning cultures. It helped researchers to identify the challenges education must face in the next century and in devising imaginative ways to meet these challenges.

Educational computing and Cognitive Science have had their share of critics, however. Cognitive Science critics tended to focus on the continuing quest for artificial intelligence. Educational computing critics represent a diverse set of disciplines and opinions. In the mid-1950's,
Jacques Ellul (1964) advanced a detailed argument to the effect that technology was now operating according to its own inner logic and had become the dominant force shaping the direction of cultural development around the world. Heidegger's existentialist perspective added a potent realization that our personal affect about technology could not be reduced by our personal affirmations or denials about its effects or use. Heidegger's technological determinism gradually gained more prominence throughout the field in works by Sherry Turkle (1984), C. A. Bowers (1988) and Ron Ragsdale (1988), discussed later. Neil Postman at New York University argues, "Americans have come to believe that technological progress is the same thing as human progress and that through technological ingenuity and development we can all reach paradise" (Cordes, 1994, p.10). Cordes (1994) reported five of the most common criticisms:

- A lack of democratic participation in the design and use of technologies that can profoundly alter ordinary citizens lives
- The threat of global and local ecological crises fueled by technological advances with unforeseen consequences
- The increasing isolation of humans from the rest of the living world and a technological vision that is making of the natural living world, including most of the human body, obsolete
- Family and friends become strangers, and strangers friends, with technology that permits users long-distance relationships- and a false sense of power
- The tendency for new technologies and the global economy they promote to centralize political and economic power and to homogenize and impoverish cultures

Arguably the most influential critic of computing in education, however, has been Richard Clark (1985), who has stated that computers no more affect student achievement than the truck that delivers their groceries can affect their nutrition' . We will also revisit Dr. Clark's argument later.

_Educational Computing Before 1988: Tutor, Tool, Tutee_

Before 1988, advocates of educational computing tended to adopt one or more of Patrick Suppes' roles for the computer (in Taylor, 1980); as a "tutor", as a "tool" or as a "tutee". Briefly, the "tutor" mode of computer operation required content design in a specific content area and substantial coding by expert computer programmers. This mode was thought to accommodate a wide-range of individual differences. Psychologists and educators who adopted the tutor mode of computer operation proposed that computer programs be designed to teach all manner of knowledge and skills to a wide range of audiences. Instructional designers diligently applied behavioural principles of operant conditioning (Skinner, 1960) and data transmission (Shannon & Weaver, 1949; Weiner, 1950) to a variety of educational situations. The advice to educators was to adopt computers as "integrated learning systems which permit students to work at their own pace" (Becker, 1993, p. 129). Proponents of the "tutor" approach criticized the "tutee" camp for delaying the acquisition of skills which would be needed to make "real" discoveries.
The "tool" mode of computer operation required only pre-programmed software like a statistical package or word processor. This mode was thought to save time and preserve intellectual energy by transferring necessary but routine clerical tasks to the computer. The advice to educators from researchers was, "Don't treat students as objects to be stuffed with information. Students should use computers as tools like adults; they should learn how to use spreadsheets, databases and word processors" (Becker, 1993, p. 129).

Trainers, teachers and students who adopted the "tool" mode of computer operation (to this day) sensed intuitively that this was somehow better than doing things the old way, despite empirical research to the contrary (see Lockard et al, 1994). Only recently have teaching and assessing methods been improved. When the computer was given a "tutee" role, the students would learn a programming language to program the computer (Becker, 1993). The "tutee" mode of computer operation required students or the teacher to learn a programming language to communicate with the computer. The computer was thought to be a good tutee because of its dumbness, its patience, its rigidity and its capacity for being re-initialized. Papert (1980) at MIT championed this use of the computer in education with his children's programming language called LOGO. Papert and others claimed that LOGO programming forced students to think like the computer thereby concretizing the learning process for them. In this way, student computer programming (the "tutee" role) was thought to be more beneficial than when students were being programmed by the computer (the "tutor" role).

Critics of educational computing, however, were gaining a firm hold. They came down hard on claims about knowledge transfer and cognitive benefits from tutor, tool and tutee computing. Richard Clark (1983; 1985, 1991), for example, maintained that, in educational settings, any technology was as good as any other as long the technology was matched with the appropriate instructional strategy. Supposedly, this was always true because there was nothing unique about any educational technology and that many effects could be attributed to instructional method. As a consequence of this statement, several generic, print-based models of instructional design were applied to computing contexts. These generic models, however, were found to be inappropriate: in intelligent tutoring systems (Dede & Swigger, 1988); in computer-supported intentional learning environments to improve knowledge-building discourse (Scardamalia et al, 1989); and when the computer was a zone of proximal development (reading partner) for internalizing reading-related metacognitions (Salomon et al, 1989), to name a few.

Nevertheless, Clark continued to apply the delivery truck analogy to illustrate his point. In 1991, Clark admitted, with some regret perhaps, that many good jobs in the mid-1980's in educational technology had been dissolved as a direct result of the publication of his delivery truck argument'. The delivery truck analogy stated that computers no more affect student achievement than the truck that delivers their groceries affects their nutrition'.

However, several false premises made Clark's argument presented by analogy invalid. First, there was what Salomon & Gardner (1986) called Clark's "average effects of a technology" premise. This premise incorrectly stated that the multitude of typical computing attributes were all similar to the solitary typical purpose of a delivery truck. Second, the delivery truck-computer analogy was false as well because it was imprecise on the attribute level, as noted by Mann (1992a). Cognitive activity had become more important than the physical evidence of the learner's
interactions with software. It was no longer adequate for researchers in the 1980's to describe interactions simply in terms of either the input technology employed or the physical characteristics of the responses made, since these would change over time.

Despite these inaccuracies in the educational computing literature, skepticism about the actual effects of tutor, tool and tutee uses of computers on student achievement and problem solving continued to have an impact on computing educators. Sociological and axiological issues began to play a greater role in educational computing research. Three other researchers deserve mention here: Dreyfus and Dreyfus, Pea and Kurland, and Winograd and Flores. Dreyfus and Dreyfus (1984) advocated that the optimism over intelligent tutoring systems was unwarranted. Pea and Kurland (1984) took issue with Papert's Bank Street studies on LOGO programming with children. Winograd and Flores (1987) introduced what might be described as a hermeneutical approach to Educational Computing in which observers and settings were scrutinized as intensely as learners' posttests and verbal protocols.

**Educational Computing After 1988: Constructing a Rationale**

Bowers and Ragsdale brought fresh insight to this fledgling field in 1988. C. A. Bowers (1988) published an important little book entitled, Cultural Dimensions of Educational Computing: Understanding the Non-Neutrality of Technology. In this work, Bowers criticized Taylor's tutor, tool and tutee categories of educational computing for assuming conditions of technological neutrality, the autonomous nature of people and a conduit view of language. An elaboration of these arguments is beyond the purpose of this paper.

Ragsdale extended Bowers' sociological investigations with practical suggestions for the student, teacher and parents of children doing educational computing. In his book, Permissible Computing in Education: Values, Assumptions and Needs, Ron Ragsdale (1988) proposed the idea that the benefits of educational computing were always situated and relative. He presented several frameworks for computing educators to consider including: four kinds of student expectations about using computers; four different roles for teachers; and a four-part role for parents. Ragsdale also offered direction for pre-service training and certification, for graduate education and for appropriate research directions in educational computing.

The field was beginning to develop a scholarly tradition of its own. Educational research with the personal computer was shifting away from the three laboratory modes (tutor, tool, tutee) toward new combinations of attributes that computers could be made to have (Salomon & Gardner, 1986). Raster graphics, for example, substantially increased a computer's attributes for simulating expensive chemical experiments (Mann et al, 1989) or dangerous events in a school setting such as detecting radioactive isotopes on a workbench (Mann, 1988). Another, more widespread use of attributes was the computer's capability of managing text.

The debates about educational computing, with the exception of the Clark argument, remained at the heart of educational computing research and development. The physical presence of computers was a necessary albeit insufficient cause of widespread usage (Becker, 1993). Between 1983-84 and 1991-92, the number of personal computers in American schools subsequently increased to 600 per cent, with a reduction in the number of students-per-computer
from 125 to 19 (Kinnamen, 1992). Computer use was restricted to either a few teachers who could use computers intensively or many teachers who could use them only incidentally (Pelgrum & Plomp, 1993). Most teachers, however, had little opportunity to learn about attributes and teaching methods until the mid-1980's when personal computers became affordable and available. Many educators simply adapted their old teaching methods and learned some new ones to accommodate the computing attributes (Healy et al, 1994). In sum, this meant that the majority of teachers had less than a decade of experience using technology in the classroom (Kearsley & Lynch, 1992).

Computing in Education Today

Today, educational computing tends to adopt one of three approaches: Transformativism, Collaborationism or Incrementalism. Transformationalism proposes that an overhaul of the education system is required before teacher development involving computers is possible (Becker, 1993; Lockard et al, 1994; Mann, 1994; Pelgrum & Plomp, 1993; Zorfass, 1993). Undoubtedly, this classical western approach to improvement has led to dramatic improvements, but typically has not been standardized and maintained. Transformativism, therefore, is often found to be an exercise in planned obsolescence.

Collaborationism advocates that educational reform should be spear-headed by collaborative interactions over computer networks such as computer workgroups or computer conferences (Harrington, 1993; Flanders, 1991; Hunter, 1993; Mann & Weir, 1993). The main rationale for adopting a collaborationist policy is that learning activities and environments can include and take advantage of interactions of learners and teachers with people, information and machines that are geographically and institutionally distributed (Hunter, 1993). The assumption that distributed collaborations computer networks between learners or teachers improves achievement in curricular areas, however, is untested. Users of computer networks, like users of any other hypermedium often feel subjected to what is sometimes described as "a cohesion deficit from hyperspace wandering" (Duchastel, 1990, p. 230). Collaborationism involving computer networks often relies too heavily on untested distributed resources. Quality control, therefore, must become an important issue in collaboration over educational computer networks (Mann, 1992b, 1993; Mann & Weir, 1993).

Incrementalism is the best of the three approaches because of its reliance on knowledgeable people instead of massive expenditure on innovation. This preference is consistent with the Japanese management practice of kaizen meaning "slow, never-ending improvement in all aspects of life" that focuses on quality control (Mann, 1992). Incremental changes can be made to occur at different levels of an educational system. Incrementalism accounts for differences between general and special-purpose computing factors for graduate and undergraduate students and can be implemented by administrators or individuals with a commitment to respond to business and government concerns about computing policy for education. Policy makers in Newfoundland and Labrador, for example, have taken an incremental, bottom-up approach policy development to improve the chances of policy implementation. Phase 2-3 of the "Technology in Learning Environments" (T.I.L.E) document (Eaton, 1994), written by and for educators of Newfoundland and Labrador, proposes substantive and cost-effective improvements in the policies and practices of educational computing.
Summary

Educators aspire to implement substantive and cost-effective improvements in their policies and practices on educational computing. The relatively short history of this field, however, offers few prescriptions toward this end. This paper briefly reviewed computing history and examined three contemporary approaches to implementing change. The preferred position advanced in this paper might be called incrementalism. Incrementalists propose that inservice courses in educational computing be provided to assist teachers in how to implement computers in the instructional process. This approach is focused beyond the computer "how to's"; to understanding and applying concepts, and to developing thinking, reasoning, living-with-change and problem-solving skills.

References