CHAPTER 1

INTRODUCTION

The increasing applicability and use of computer technology to promote social and economic goals is a welcome trend. From its beginning as a mechanism for performing accounting procedures and automating scientific calculations, the automatic digital computer's use has spread into almost every major area of human activity. Wisely and appropriately used, it can make a significant contribution to the study of many economic and social problems, assist in the formulation of policies in these areas, and be instrumental in their application and administration.

The developments reported in this volume and in its companion volume [06] provide one of a growing number of examples of the use of computer technology to assist the process of social science research and development. While the problems studied generally do not involve computer technology *per se*, the methodology used for the studies often is feasible only if supporting computer technology is available. Such a state of affairs is not accidental, but results from an interplay between the development of methodology and technology. The availability of computer technology has allowed methodologies once only of theoretical interest to become operationally feasible. This provides a spur to the development of such methodology, which both exploits the technology and becomes dependent upon it for much of its effectiveness. The work reported here and in [06] provide a strong example of the interdependence between simulation methodology and computer technology in the social sciences.

Background

The system for microanalytic simulation described in this volume has been created as an integral part of a larger project to study, *inter alia,* the determinants of personal income and wealth in the U.S. The level and distribution of personal income have long been a subject of interest for economists and formulators of economic policy. The income received by an individual contributes directly to his claim upon resources, his health and well-being, and his ability to obtain education, develop skills, participate directly in the labor force, and share many of the benefits of the society in which he lives. Recent concern regarding socially acceptable distributions of income has centered upon the problems of members of racial minority groups, aged persons, and other persons whose income and assets are not capable of supporting a moderate standard of living.

The primary motivation for this research activity is that current social and economic concern for public policy is being focused increasingly upon the distribution of income shares among population subgroups as well as the level or rate of growth of national income. Urban problems, migration to cities, and educational inequalities have focused the attention of policy makers on problems of income generation for racial minorities. The problems of loss of earnings, irregular pension coverage and inflation are of particular importance to elderly persons. Differential welfare standards, concern for alleviating poverty, and a weakening of the ideological relationship between work performed and income have made discussion of and experimentation with negative income tax plans possible and increasingly acceptable.

Public policy formation concerned with the rate or stability of economic growth has often been based upon the use of aggregate economic models to study the implications of exogenous government policy alternatives. Aggregate models generally represent an economic system as a system of equations in which economic variables such as gross national product, national unemployment rate, and others are linked in a system of partial difference equations. The success of such models depends in large measure upon the accurate specification and stability of aggregate relationships over time, in the same way as the profitability of life insurance companies depends upon the stability of mortality rates. However, if public policy formation is to focus upon the implications of policy for the individual, including behavioral responses of persons and families, then aggregate data are not relevant for the examination of many relationships within the family unit; and the use of behavioral and economic relationships that aggregate behavioral response does not allow testing of most hypotheses relating to the micro unit.

Microeconomic models offer an alternative to aggregate models when the policies to be simulated affect individuals differently and when distributional outputs are of significant importance. Microeconomic models operate upon a specific representation of the individual decision making units in the population to be modelled. A necessary condition for the development of such models of the household sector is that there exist at least one sufficiently detailed social and economic description of the population of households to specify an initial state, and that there also exists sufficient substantive knowledge of household behavior to formulate the processes, or operating characteristics, of the model.

Although the concepts of microeconomic theory and analysis are not new, the implementation and detailed study of the dynamic characteristics of such models requires the use of flexible, reliable, and inexpensive computing tools. In a path-breaking study in 1959, Orcutt and his colleagues developed a conceptualization of socioeconomic microsimulation and used it to construct a demographic microsimulation model of the household sector [O4]. The simulation system design and implementation supporting this research was performed by Greenberger [G3]. The system for simulation described in this volume supports the continuation of this research effort by Orcutt and his colleagues.

The application of simulation methodology is absolutely essential for the solution of general microanalytic models of the class generated by Orcutt et al. [O4]. Classical analytical methods are inapplicable. Such a model can, however, be formulated as a Markov process, but the size of such a process for even a simple model overwhelms all existing resources for its solution. In [O4], Greenberger derives the Markov process resulting from a model containing a population, each of whom has 7 attributes -- race, age, sex, marital status, number of children, sex of children, and ages of children. He concludes that:

> "It must be clear by now that a procedure such as this, although of considerable theoretical interest, is totally out of the question from a practical point of view. The transition matrix contains $10^{100,000}$ elements of which at least $10^{50,000}$ are non-zero. The amount of storage required for this many numbers, as well as the processing time necessary to calculate and use them, is well beyond the economic bounds of computers under present day technology."[1]

One might ask whether such an approach is now feasible after 20 years of improvement in computer processing and storage technologies. The answer is that it is not feasible now and shows no likelihood of ever being feasible. While a set of $10^{50,000}$ numbers is mathematically interpretable, the implications for storage are overwhelming. Current rotating storage capacity is now approaching a density of 1 billion (10^{9}) numbers per cubic foot. In order to hold $10^{50,000}$ numbers, a sphere of approximately $10^{16,000}$ light years in radius would be required to be filled with such devices. Such a space is considerably larger than the size of the known universe. It should be remembered that these implications are derived from a very simplified model rather than from the considerably more complex models that are of interest for research and policy.

The microanalytic simulation techniques developed for [O4] surmount this obstacle. The reasons have been succinctly stated:

> "The simulation method used in our study gives good approximation to the theoretical solution, yet has relatively modest cost and time requirements. The key to the success of this method is that it works with the *n* (approximately 5000) households rather than with the M (approximately $10^{50,000}$ population states). Likewise, it works with the decision unit probabilities (the probabilities of household and individual behavior) rather than with the transition probabilities. By comparison with the quantity of transition probabilities, the decision unit probabilities are extremely few in number."[2]

The general methodological framework developed in [O4] has been useful for a wide variety of subsequent activity.

---

1 G.H. Orcutt, et al., *Microanalysis of Socioeconomic Systems: A Simulation Study*. New York: Harper and Brothers, 1961, pp. 292-293.

2 Orcutt, et al., *ibid.*, p. 293.

The analysis of the effects of the U.S. and Canadian Federal Individual Income Tax structures provided a parallel impetus for the construction and use of microanalytic simulation models. In 1963, a static microsimulation model of the individual taxation process was constructed by the author and Benjamin A. Okner of the U.S. Treasury Department; it is described in [Pl] by Pechman. This model led to the construction of several families of more complex models by Okner and Pechman in the U.S. [O1], [P2], and by Bossons in Canada [B5]. The use of static microsimulation models of individual taxation for the measurement of aggregate tax collections and the redistributive effects upon income of tax structures is now routine.

In a related study [S7] focusing upon the effect of private pensions upon the distribution of income, Schulz used the l-in-l000 microdata sample of the 1960 Census of Population and Housing to estimate both the distribution and level of income of the future aged and their dependence upon changes in the status of private pension plans in the U.S. More recently, substantial interest in various forms of negative taxation plans have resulted in the construction of reasonably detailed static microsimulation systems such as [O2] and [O3] for more detailed analysis of their potential effects.

With the exception of the Schulz study, the other microanalytic models have been largely static, with little or no feedback mechanisms included. The Schulz study, while incorporating a number of dynamic features, uses a very select initial population consisting of all married couples with husband between 45 and 60 years old and all unmarried individuals between 45 and 60. Death was the only demographic process included in the model. Furthermore, the model was applied independently to each family and individual in the initial micropopulation.

The results reported in [06] which were implemented with the computer system described herein are based upon a model which is a significant extension of Orcutt's earlier microanalytic model of the demographic sector. This extension has several significant dimensions: (1) inclusion of aspects of household behavior other than demographic behavior; (2) use of a larger and more detailed set of individual and family attributes and more complex submodels to provide a more realistic representation of behavior; and (3) incorporation of an aggregate economic model that both interacts with the micro submodels and allows the introduction of aggregate economic policy measures into simulations. These extensions were added both to improve the correspondence between model characteristics and the complexity of individual and household behavior in the U.S. and to provide a more comprehensive framework within which a large set of policy alternatives could be studied using simulation techniques.

While the model described in [06] represents a significant extension of research in the construction of microeconomic models, it should be distinguished from general equilibrium models which form a quite different subset of microeconomic models. In the present work, no attention is given to the computation of equilibrium values for any time period, although some attention is given to consistency of values across sectors. In general equilibrium models, it is assumed that preference functions exist for each commodity, and the derived demand schedules enter into determination of the equilibrium position. In the present work, preferences are expressed in a rudimentary and stochastic manner with respect to a few decisions, such as the demand for marriage, education, and employment. Supply schedules, on the other hand, are treated in various ways. The supply of mates for marriage is defined by the demand for marriage by the opposite sex; the supply of education is perfectly elastic at zero cost; and the supply of employment is determined by government policy implemented in a macroeconomic submodel.

The most important distinction between the two types of models, however, is the distinction between equilibrium and adjustment. General equilibrium models achieve equilibrium solutions for all endogenous variables in the model for specific instances in time. Such a solution is consistent and global. In contrast, Orcutt's extended model is concerned with adjustment processes, both within a given time period and from one time period to another. The variables in Orcutt's model undergo continuous adjustment as a result both of changes in exogenous variables and state values in previous time periods. It is of course assumed that this adjustment process tracks a moving equilibrium position, but that position may never be reached.

System Description

The task of constructing a computer based system to support the activities of a project in socioeconomic modelling such as [O6] is an exacting one. Such a system must be capable both of supporting implementation and solution of the model structure initially specified and of adapting to the inevitable evolution of the structure over time. Of equal importance is the requirement that the system be "user-oriented," i.e. it should be easy to use and sufficiently flexible to permit the wide variety of operations required to support effective economic research.

Such a computer based system is only one part of a complex process including research in, development of, and use of microanalytic models. The computer based system described in this volume is part of a considerably larger process, and the success of such a system must be judged not only by its effectiveness and efficiency as a system *per se*, but also by its contribution to the entire process of creating and exploiting microanalytic models. Thus, while the system structure is described here in a way that places the system at the center of the processes surrounding it, such a framework is chosen largely for focusing upon the characteristics of the system and its relation to the overall research work. In particular, there is not meant to be any implication that the computer based system is the central element in the work.

In a more general framework, such a computer based system is an essential tool in support of economic and social policy research and exploration.

The relationship of the computer based system to other activities involved in microanalytic model construction may be viewed as a series of concentric sets of activities, as illustrated in Figure 1-1. Such a framework places the specific functions of the computer system in the center, surrounded first by activities directly related to the model and next by the more general environment of economic research and policy evaluation. Within the innermost area in Figure 1-1, the functions of the system are shaded, while the entities which are manipulated by the system are not shaded.

The outermost ring consists of the economic research and policy environment in which modelling activity occurs. In general, this environment must be taken as given by the model builder. Important exogenous factors at this level include the existing stock of social science knowledge and literature, the stock of aggregate and micro data available, and the degree of importance and urgency attached to the study of various issues and to the evaluation of alternative policy proposals.

Within this general environment, microanalytic modelling activity takes place. The middle set of activities in Figure 1-1 contains such activities. From the existing stock of household surveys and censuses, microdata are chosen (1) for constructing an initial micropopulation for simulation, and a machine readable codebook is prepared (2) describing the microdata file chosen. The stock of social science knowledge is used both to initiate (3) special research activity and to formulate (4) components of the microanalytic model. The policy issues to be studied and the alternatives to be evaluated also effect (5), (6) the additional research to be performed and the formulation of the microanalytic model. The model formulation in turn will generally require additional micro entity attributes not present in the initial microdata source, and these attributes are introduced (7) in the attribute library. Some published aggregate data are moved (8) into a machine readable aggregate time series data bank. If new series are required, they are generally the product (9) of special research using the existing stock of time series data. Such new series are also stored (10) in the machine readable data bank.

The outputs of the computer-based model implementation also appear at this conceptual level. Simulation logs (11), time series generated by solving the model (12), surveys and censuses of the simulated micropopulation (13), and user interaction directly with the model (14) all combine to assist in the analysis of results and the preparation of research findings. These findings, in turn, are translated both into published research outputs (15) and into appropriate inputs for policy evaluation and the decision making process (16).

The micromodel formulated using the existing stock of social science knowledge with specific augmentation consists of a precise statement of understanding about reality. This level of the model is called the "source model." These statements are then translated (17) into computer programs which are functionally equivalent to the source model and which can be executed by a computer based simulation system. This set of computer programs is called the "object model" corresponding to the source model. This object model is then embedded (18) in a system for executing simulations, which executes the object model properly and allows users to control its operation.

Figure 1-1. Schematic Diagram of Microanalytic Modelling Activity with Emphasis upon Computer Based Processes

The innermost area of Figure 1-1 contains the various functions of the MASH (*M*icro*A*nalytic *S*imulation of *H*ouseholds) system. MASH is an integrated system of modules for performing the variety of microsimulation functions. Initial micropopulations are created from survey or census data (1) and from machine readable descriptions of these files (19). Attributes from the attribute library may also be included (20). A user dictionary may be used to create and modify certain entities (21) that are useful for population creation (22), simulation execution (23), and other functions. The initial micropopulation is used for simulation (24), and the object model is applied to it successively (25) to produce future states predicted by the model. Within the dictionary is an aggregate time series data bank that interacts with both the microanalytic model (26) and the aggregate models (27). The aggregate models generate time series that assist in the analysis of the simulation experiment (28).

The experimenter may, at any time, browse through the data at the micropopulation level (29) and modify microdata values. Such browsing activity has been considered useful in the past to allow the experimenter to obtain a "feel" of the data (30). Censuses and surveys may be extracted from the micropopulation (31); each is documented by a codebook that is generated at the same time. These derived microdata sets may be used for analysis of the simulation experiment (13), and they may also be used as a source for new micropopulations for the initial simulation state (32).

Finally, the system contains a variety of "housekeeping" functions for micropopulation manipulation and maintenance (33). Using the facilities of the host computer's file system, micropopulations may be stored in various states (34) and may be retrieved later for further simulation work or for analysis.

While Figure 1-1 does not stress the feedback or closure (33), (34) between substantive results obtained from the modelling activity and the environment in which it takes place and the knowledge upon which it is based, there are significant and strong links between the two. The results of modelling activity add to the stock of social science knowledge and provide information regarding implications of the adoption of various policies. Such results often lead to the formulation of still other policies and to further analysis and modelling tasks. Likewise, the results of modelling activity can identify weaknesses in existing microdata coverage which can lead to additional data collection activity. These feedback mechanisms are not stressed here because the presentation is oriented to the supporting computer system. However, it should be clear that these relationships are of greatest importance and provide the basis for the socioeconomic modelling and computer system development effort.

System Implementation

The MASH system was developed and is being used on a Digital Equipment Corporation PDP-10 (DECsystem-10) computer located at The Brookings Institution in Washington, D.C. The configuration of the Brookings computer includes a "time-sharing" central processor, 128K (K=1024) 36-bit words of high speed immediate access memory, 5 disk drives with demountable packs, several magnetic tape drives, a card reader and line printer, and communications interfaces for remote interactive low speed terminals. The computer is accessed by a number of social science research organizations in the Washington area. Access to the computer is generally obtained using typewriter-like devices which are connected at the user's end to standard voice public telephone circuits.

The decision to implement MASH on an interactive system was conscious and deliberate. This decision, and its implications, form a major difference between this system and its predecessor. It is the author's conviction that an interactive environment offers extremely important benefits both for system development and for effective use by system users. This section discusses the mechanics and use of interactive systems and describes the hypotheses leading to the choice of such a system for MASH.

The concept of "time-sharing" embodies a variety of interpretations and has no precise technical meaning. What is often meant by "time- sharing" is a computer environment that combines two ingredients: (1) concurrent sharing of central computer resources such as the central processor and memory by generally similar computer-based tasks; and (2) interaction with computer programs through on-line, low speed terminal devices, with potential response times of several seconds for simple tasks. The first ingredient without the second is generally called multiprogramming and is the most common form of operation of current batch environments. The second ingredient without the first provides single user interaction, which is a common working environment for many small scale computers and minicomputers. Both ingredients combined provide the desired interaction with the power of larger size and the economies of scale typical of larger machines.

In the most common mode of operation of a "time-sharing" system, a programmer or user of the system uses a terminal or console device consisting of a keyboard similar to that of a typewriter and either a video display unit or a '"hard copy" output device again similar to that of a typewriter. This console is connected to the system, either directly or through public communications facilities, and the user and the system engage in a structured dialogue through which the system is instructed to perform a desired sequence of operations. The dialogue between the user of a computer program and the program may be frequent, or it may occur only once; it may be highly stylized or it may be free form; and it may be in imperative or command style or it may be in interrogative or question-answer style. A major advantage of interactive computing is that it allows the design and construction of computer programs that utilize the computational ability and speed of the computer and the judgment and decision making capability of the user within an effective structure of dialogue that either resource alone would not support. One of the design goals for MASH is to achieve that mix for users who construct and perform simulation experiments.

A major difference between this method of using computing machinery and its historical alternative, batch computing, is that an interactive user can make decisions concerning his research or programming strategy in a sequential manner with assistance from the feedback supplied by the computer, whereas the user of a batch computing system must either prespecify a longer sequence of operations or use more system resources and more of his own skills to obtain such flexibility. Multi-access interactive computing systems are feasible because of the great disparity between the speeds at which computers and people function and because of the wide spectrum of requirements placed upon the resources of most computer systems by members of its user population. A time-sharing system is analogous to a chess master who can play "simultaneous" games of chess with many opponents because of his ability to remember more and think more quickly than his opponents. In the same manner, a multi-access interactive computer system circulates among its users and allocates its resources to their requirements according to a predetermined system of priorities.

In other respects, however, interactive and batch computing systems may be regarded as complementary computing techniques rather than competitive ones. Interactive and batch computing often use similar types and quantities of hardware resources, and generally support the same programming languages and applications functions. In many respects it is not an oversimplification to suggest that interactive computing may be regarded as batch computing, plus access to one additional type of input-output device -- an interactive terminal coupled to a human operator. This additional device is addressable by an interactive program just as other input-output devices such as tape and disk drives are addressable both by batch and interactive programs. If the computer's operating system includes an input and output service routine for this new device, then the operating system supports interactive computing. Regarded in this framework, interactive and batch operating systems are conceptually quite similar. Programs using the terminal (human) device may, however, be considerably more efficient, powerful, and flexible than those restricted to automatic devices. Some multi-access interactive systems, such as the PDP-10, are foreground-background compatible, so that tasks prepared in either mode can be executed in either mode. Such a unified system design allows the user to make decisions concerning the type of use of computer resources almost wholly on the basis of the relative comparative advantage of the techniques available.

There exist classical arguments for time-sharing both at the global and at the local level. At the global level, time-sharing allows the physical resources of a large computer system to be exploited by a number of concurrent users. If these users are distributed geographically in different time zones, or if there are incentives or administrative mechanisms that encourage a distribution of use over time, than such systems tend to exhibit an efficient use of resources over a daily cycle. These arguments also apply to environments consisting of multiprogramming without any general interactive facility.

At the level of the individual user, the potential benefits of time sharing environments become considerably stronger. First, for the community of users of such a system, there exists the possibility of a substantial and meaningful sharing of the intellectual resources within the community, as partially realized in the stock of programs and data made available to users of the system. Second, it is now generally acknowledged that program development is more efficiently performed in an interactive environment. Of at least equal importance is the ability in such environments to develop programs that are directed interactively at execution time by their users. A very important explanation for the effectiveness of interactive systems for both programming and use is their "permissiveness" and other supporting characteristics, as opposed to the generally "punitive" behavior of current batch systems. In interactive environments, strong positive reinforcement is generally generated by repeated trials in short intervals of time when successful results are finally achieved. In batch environments, "turnaround time" is generally sufficiently long and the overhead mechanics of job submission are sufficiently large to dampen any positive reinforcement mechanism that is generated by achieving correct results. In other words, the cost of an error is generally low in interactive systems since it can often be corrected almost immediately, but the cost can be very high in batch systems because of the often substantially delayed feedback that characterize such environments.

Several additional considerations entered into the choice of a general purpose interactive computing environment for the development and execution of MASH. Four of the most important appeared to be: (1) the possibility of exporting such a system through making it directly accessible from remote locations; (2) the impact of a high level language for microanalytic simulation in an interactive environment; (3) extension of the principle of program and data sharing to the sharing of research progress and results; and (4) establishment both of a central operation and a tight feedback loop between system developers and system users.

Substantial attention has been devoted in recent years to the problem of transferability of software, both in social science computing and in other fields. The kernel of this problem is how to maximize the usefulness of software, once developed and debugged, in a variety of computing environments. The problem is caused by the existence of different hardware and different software characteristics in the computing market. Approaches to solving the transferability problem have included higher level languages, source program generators, source program filters, and use of minimum language subsets for writing programs.

In the present context, the transferability problem or exportability problem is regarded as one possible problem derived from the more basic problem, which is access to useful software. In many environments such as academic computing centers, accessibility implies local accessibility and therefore the successful importation of externally developed software is generally necessary. In a widely dispersed research community of moderate size, however, it was felt that the costs of creating an exportable package and the exporting process might be considerably higher than the costs of accessing a central machine on which specialized software is implemented. Whereas the costs of transportability are measured in additional development effort and execution time inefficiency, the costs of central access may be measured in terms of the technical problems and costs of communication with the central facility, the administrative problems of allowing widespread remote access, and the problems of providing service to remote users. Because of the limited number of people who were expected to use MASH, it was felt that the most effective alternative was to develop the system for execution at a central site but to facilitate access to it by any authorized remote user.

High level, "user-oriented," languages have been available for both general and special purpose computing activities ever since the mid-1950s. While at first their use was quite expensive, they are now generally accepted as being effective for a wide variety of tasks. The main arguments for their use include increased programmer productivity, potential ease of transferability, self-documenting features, and ease of understanding. As the ratio of programmer costs to machine costs continues to rise, such arguments become stronger.

A principal hypothesis underlying the development of the MASH system was that a major gain in effectiveness and usefulness would result if the system control language was constructed at a high level and corresponded closely to the vocabulary used to discuss microanalytic source models. In particular, it was hoped that such a high level program interface would encourage direct use of the system by social science research staff rather than only through intermediate technical personnel. Some support was given to this hypothesis by the ready acceptance of mathematically oriented high level level languages by scientific and engineering workers who had mostly used programmers as intermediaries prior to the availability of such languages.

While the use of high level languages is independent of whether a computing environment offers interactive or batch services, the effectiveness of their use is related. It was hypothesized that social science research staff would be more likely to use such a language if the barriers to its use were minimized. This was judged to imply the necessity of a completely interactive environment during execution of MASH, for two major reasons. First, the permissiveness of such a system in providing rapid error detection and immediate possibilities for correction were judged to be even more important for non-technical social scientists than for professional programmers, since it was felt that non-technical persons would become more easily discouraged in a non-supportive environment and therefore not use the system directly.

Second, it was hypothesized that the research process itself was highly interactive, with research persons interacting with models, data, and each other, either directly or through common literature. The modelling process especially was thought to be quite interactive, with the implications of a version of a model determining subsequent changes. To support such activity, the command language of MASH was structured to include commands that allowed interactive examination of the results of applying a model to its associated data base. It was hoped that such commands would permit the user to maintain a "feel" of the implications of his or her work that would allow the system to be used as an effective aid in model construction and research activity.

Another consideration that prompted the choice of a time sharing environment for MASH was the possibility of using it to create an environment for "research sharing." In early time sharing environments such as CTSS [Cl], behavioral patterns of program, data and task sharing were soon observed, and a major reason was the ease of accessibility that such a multi-access interactive system provided to both one's own work and that of others and the ease of use of this work. It was hoped that the development of MASH on a central, multi-access interactive computer system would lead not only to program and data sharing but also to both cooperative model development and the sharing of research activities and results. While an interactive computing environment is not technically necessary for such behavior to be feasible, it appears to have been instrumental in encouraging such behavior. It seems reasonable to believe that interactive capability and services based upon it would also be a necessary element in encouraging active sharing and cooperation in a discipline-specific task such as microanalytic modelling, and it was hoped that the choice of a time sharing environment would increase the extent to which this would occur.

Finally, a multi-access interactive facility provided the facilities upon which to base a variety of user assistance and feedback mechanisms. Such mechanisms were considered of benefit to system users since they would result in more effective use of the system, but they were considered of at least equal importance to a process of continuously assessing user experience with the system and modifying the system on the basis of those assessments. While in theory such a mechanism could have been established using a non-interactive environment, it was felt that the result would have been far weaker, less immediate and generally much less effective.

In summary, the hypothesis that important benefits would result from establishing MASH in an interactive multi-access computing environment rests upon several basic assumptions. It is assumed that persons active in research and policy explorations involving microanalytic models of the economy do form an "invisible college" within which there exists active communication and interaction. Computer based systems that support such activities should be oriented to the structure and requirements of this user community. The benefits of multiple access interactive systems for the computing community, if generalizable, could provide significant substantive and technical benefits to this community of users. Finally, the commercial availability of such systems coupled with increasingly effective use of the public telephone network for remote access make it possible to allow access to and participation in the system to a geographically decentralized group of users, thereby providing a solution to the transferability problem, economies in centralized development, and more effective user feedback mechanisms.