

## ABSTRACT

Computer Simulation as a Decision Support Tool. (August 1986)

Michael G. Ketcham, B.A., University of Washington;

M.A., The Johns Hopkins University;

Ph.D., The Johns Hopkins University

Co-Chairmen of Advisory Committee: Dr. Gary L. Hogg  
Dr. Robert E. Shannon

This dissertation describes an experimental simulation environment called MBS, for "Model Based Simulator." The dissertation presents research results in three stages: (1) MBS design goals, (2) the status of the MBS prototype, and (3) future developments that will lead to a complete MBS design. The discussion of the prototype itself has four parts: a discussion of the MBS modeling logic; the representation of experiments in MBS; the software implementation; and an example of simulation development in MBS.

MBS views a simulation model as comparable to a production database in which the schema specifies the information needed to characterize objects in the system, the movement of objects from one location to another, and simulation control information. The relationships in the schema represent the information needed to model a large class of manufacturing systems.

The database, user interface, and MBS simulation routines are implemented in a prototype, which is entirely original code written

in C. Commands for developing models, specifying experiments, and examining simulation results are all coordinated in the prototype through the central database. Further flexibility in developing models is achieved through data management capabilities that allow users to specify defaults, move to different levels of a modeling hierarchy, and add new object attributes to match particular modeling requirements. Executing simulations use these same information relationships. When an experiment is run, the model specifications needed to represent a system are selectively extracted from the database and mirrored in memory using dynamic data structures. Models, therefore, do not exist as program code but exist as information relationships.

Because the prototype schema incorporates a degree of knowledge about manufacturing systems, an interactive interface uses this knowledge to automatically generate prompts and defaults that lead a user through a complete representation of the system. Also, because all modeling information is held in the central database, modeling data and logical relationships can be accessed by future software modules intended for model verification and experiment formulation.