MARKOV: A GENERAL PURPOSE SIMULATION PROGRAM

Maurice F. Aburdene and Ralph J. Kochenburger

The University of Connecticut

ABSTRACT

This paper describes in language free flow diagram form, a program for simulating Markov processes (MARKOV) intended for use by someone with programming experience. The program establishes the initial state of the process (if it is unknown), the present state, the previous state, and the number of transitions from each state to all states. These parameters can be used to determine various characteristics of Markov processes of interest to the systems analyst.

INTRODUCTION

The theory of stochastic (random) processes plays an important role in the study of social systems. The manner of transition from one state to another is usually of interest to the researcher.

A special type of a stochastic process which often occurs in the models of social and physical systems is the "Markov process", where the probability that the system will be in a given state at a given time, t_2, may be obtained from knowledge of its state at any earlier time, t_1, but does not depend on how the process reached the state at time t_1. Space does not permit a more formal definition.

A finite Markov Chain is a Markov process having a discrete and finite number of states. Markov Chains occur frequently in sociology, psychology, physical systems and other fields of study.

The model for the "naive subject" experiment first performed by Asch is a discrete parameter Markov Chain with four states [1]. The "social mobility" model developed by Prais is a seven state discrete parameter chain [2]. The "survival after treatment of cancer" model developed by Zahl is a four state continuous parameter Markov Chain [3]. In order to simulate processes such as these, a general purpose algorithm (MARKOV) was developed, and is presented here in language free flow diagram form.

ACKNOWLEDGEMENT

This research was supported by the University of Connecticut Research Foundation and the University of Connecticut Computer Center.

DESCRIPTION OF THE PROGRAM

MARKOV is a program used to determine the starting state (if it is unknown, NEW=1) and the transition from one state (LASTATE) to any state (NEWSTA). The program calculates the number of transitions from every state to every other state (NOT, a number of transition matrix), and will determine when the process reaches an absorbing state, a state which cannot be departed from (SUBFIN=1). The program's algorithm is capable of handling a system with N states. The program must be supplemented by a probability transition matrix, the initial probability vector (if the starting state is unknown), and a random number generator. Figure 1 is a general flow diagram describing this program. The algorithm, MARKOV, is best explained by Figure 2.

DEFINITION OF PARAMETERS AND SUMMARY

- K = Starting number for random number generator.
- LASTATE = The previous state of the System.
- N = The number of states of the process.
- NDT = Increment between time points.
- NEW = Logical variable (NEW=0 indicates previous state of system is known).
- NEWSTA = The new state of the system.
- NOT = An (NxN) array, the number of transitions matrix from each state to every state.
- NT = Time, or parameter of the process.
- NUMBER = Random number generated by random number generator.
- PO = (Nx1) array, the initial probability vector.
- PT = (NxN) array, probability transition matrix.
- SUBFIN = Logical variable, SUBFIN=1 indicates process reached an absorbing state.

A program (MARKOV) has been presented for the simulation of discrete and continuous parameter Markov Chains. The three models mentioned earlier were simulated using MARKOV and PL/1 computer programs are available upon request from the authors.

---

1. Cohen, B.P., A Probability Model for Conformity, Sociometry, 21, 1, 1958, pp. 69-81.
2. Prais, S.J., Measuring Social Mobility, J.R. Statist. Soc., A118, pp. 56-66, 1955.
3. Zahl, S., A Markov Process Model for Follow-Up Studies, Human Biology, 27, pp. 90-120, 1955.

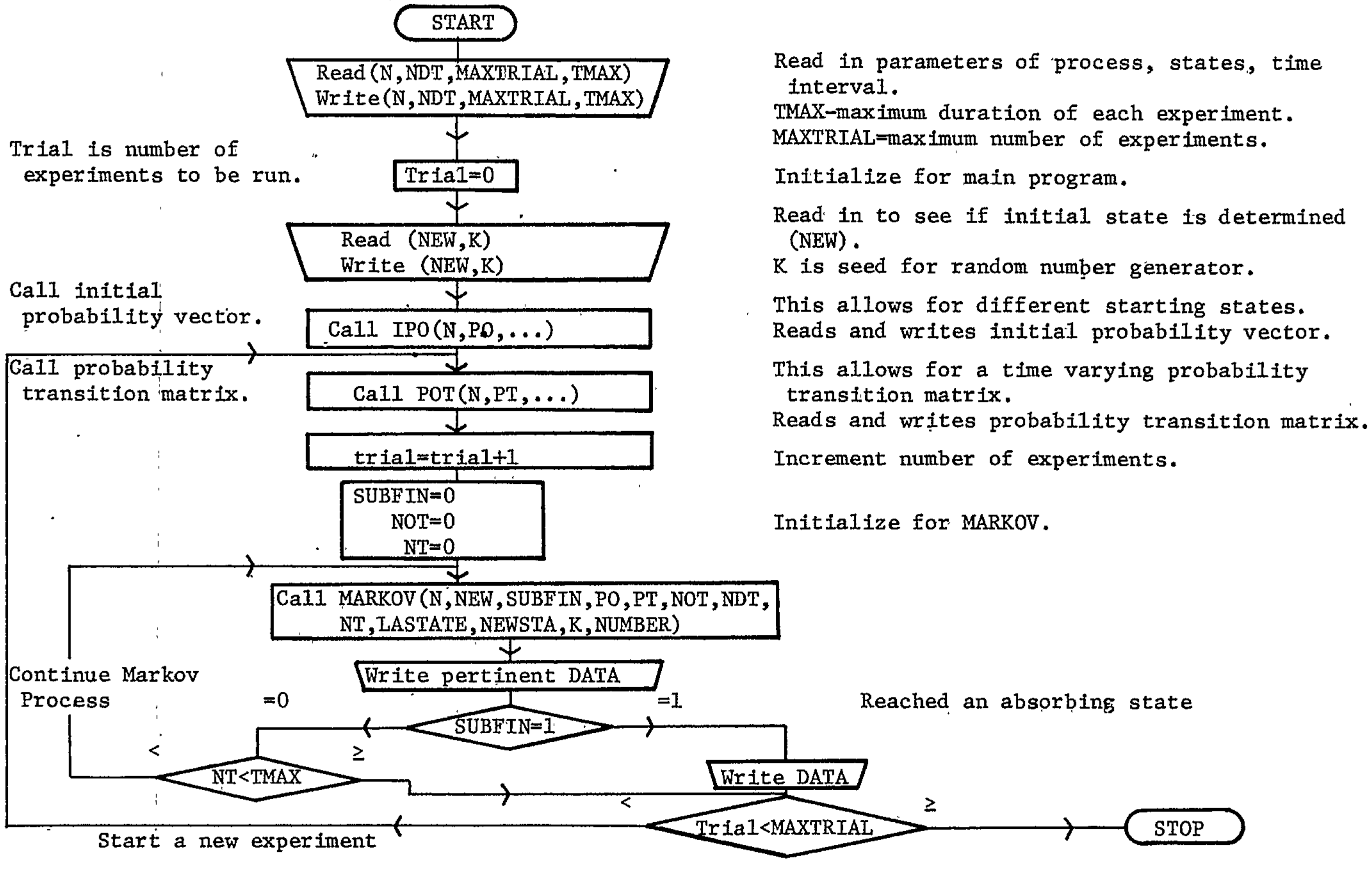

Fig. 1
General Flow Diagram Using MARKOV

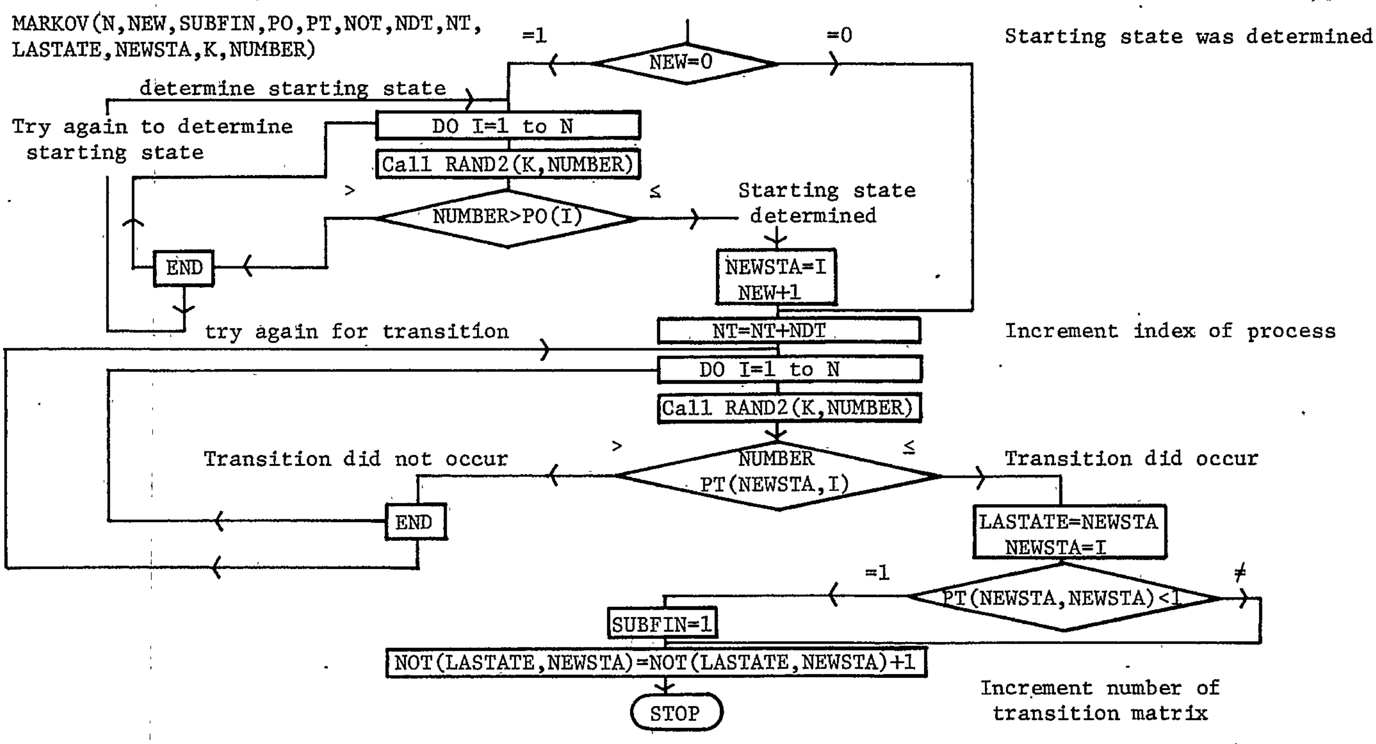

Fig. 2
Flow Diagram of MARKOV