U.S. Department of Justice
Bureau of Justice Statistics

# Models of the Criminal Justice System: A Review of Existing Impact Models

by

William Rhodes
Abt Associates Inc.
Cambridge, Massachusetts



June 1990
NCJ-124011

This report was prepared by Abt Associates, Inc., under the supervision of Carol Kaplan, chief, Bureau of Justice Statistics Federal statistical and information policy branch. The project is supported by BJS grant number 87-BJ-CX-K063. The support and helpful comments of many members of BJS are reflected throughout this document.



The Assistant Attorney General, Office of Justice Programs, coordinates the activities of the following program Offices and Bureaus: the Bureau of Justice Statistics, National Institute of Justice, Bureau of Justice Assistance, Office of Juvenile Justice and Delinquency Prevention, and the Office of Victims of Crime.

## Preface

This study reviewed criminal justice models relevant to the President's National Drug Control Strategy, in preparation for later design and development of a Federal model at the Bureau of Justice Statistics. This report covers the following topics:

- Detailed descriptions of existing computer models of local, state, and Federal criminal justice systems which are currently used to estimate the impacts of policy changes on resource needs and flows through the systems

- A summary of the applicability of these existing models to the requirements of modeling the Federal justice system

- A general description of the modifications that would be required in existing computer models, and new models that would have to be developed, in order for BJS to obtain a model or integrated collection of models that meet the requirements of the National Drug Control Strategy.

## 1. Models--an overview

The models discussed in this paper are abstract representations of selected aspects of the criminal justice system (CJS). Such models can be used to predict how the CJS will adjust to changes such as increases in arrest rates, additional resources for police, introduction of sentencing guidelines, and so on. The changes that are estimated are called "impacts."

To be a useful policy tool for decision makers, an impact model must represent the CJS with sufficient fidelity that the model's estimates of system behavior under simulated conditions reflect actual CJS behavior under similar conditions.[1] In addition, the model's developmental and operational costs must be justifiable given its applications and policy environments. Well-funded research organizations staffed by analysts skilled in computer science, programming, and statistical techniques can develop and manipulate complex models that might be less useful to others who are more comfortable with qualitative methods. Policy makers can benefit from sophisticated models if technicians are available to facilitate access; otherwise, less sophisticated models may be more suitable for them.

Consequently, it is impractical to judge CJS models by an absolute standard, and we do not attempt to do so in this paper. Instead, we provide an overview of extant models, emphasizing their potential utility in policy settings and the cost of their development and use. We also speculate about how model development might proceed in the future, especially on the Federal

---

[1] The term "sufficient fidelity" implies that the model is accurate when used to make projections. Although the model's accuracy would seem to be an absolute standard, experienced modelers know that the act of modeling a complex system can be valuable regardless of how accurate the model proves to be in practice. Indeed, accuracy may be unknown, especially when a policy innovation anticipated by the simulation is not implemented. Much could be written about how the act of developing a system simulation improves understanding of complex systems, but the topic receives no further attention in this paper.

level, and about what data would be necessary to develop and operate Federal models.

As a framework for illustrating the capabilities and levels of sophistication of CJS models, we introduce figure 1, a very simple flow model. In this model, 1,000 burglars are arrested per year. Of these 1,000, 900 are convicted; of the 900 convicted, 450 receive prison terms; of the 450 receiving prison terms, the average time spent in prison equals 2 years. If 1,000 burglars were arrested every year, prison populations would always include 900 burglars.

In this model, 90% of all arrested burglars are convicted, 50% of the convicted burglars receive prison terms, and prison terms average 2 years in length. These numbers are "parameters." If these parameters remained constant, the model could be used to predict imprisoned populations of burglars if 1,500 burglars were arrested per year (1,350 burglars in prison). The model could also predict the prison impact of increasing the average prison term from 2 to 3 years while leaving the number of arrests constant (again, 1,350 burglars would be in prison).

## 1.1 A comparison of impact models

The model in Figure 1 is so simple that a mental calculation might have sufficed to produce the same estimates. But we can draw on it in the sections that follow to illustrate some major characteristics which distinguish among different CJS models:

- focus
- flows, stocks, and system resources represented in the model
- dynamics of system adjustments
- level of detail of the units of analysis
- level of detail of CJS processes

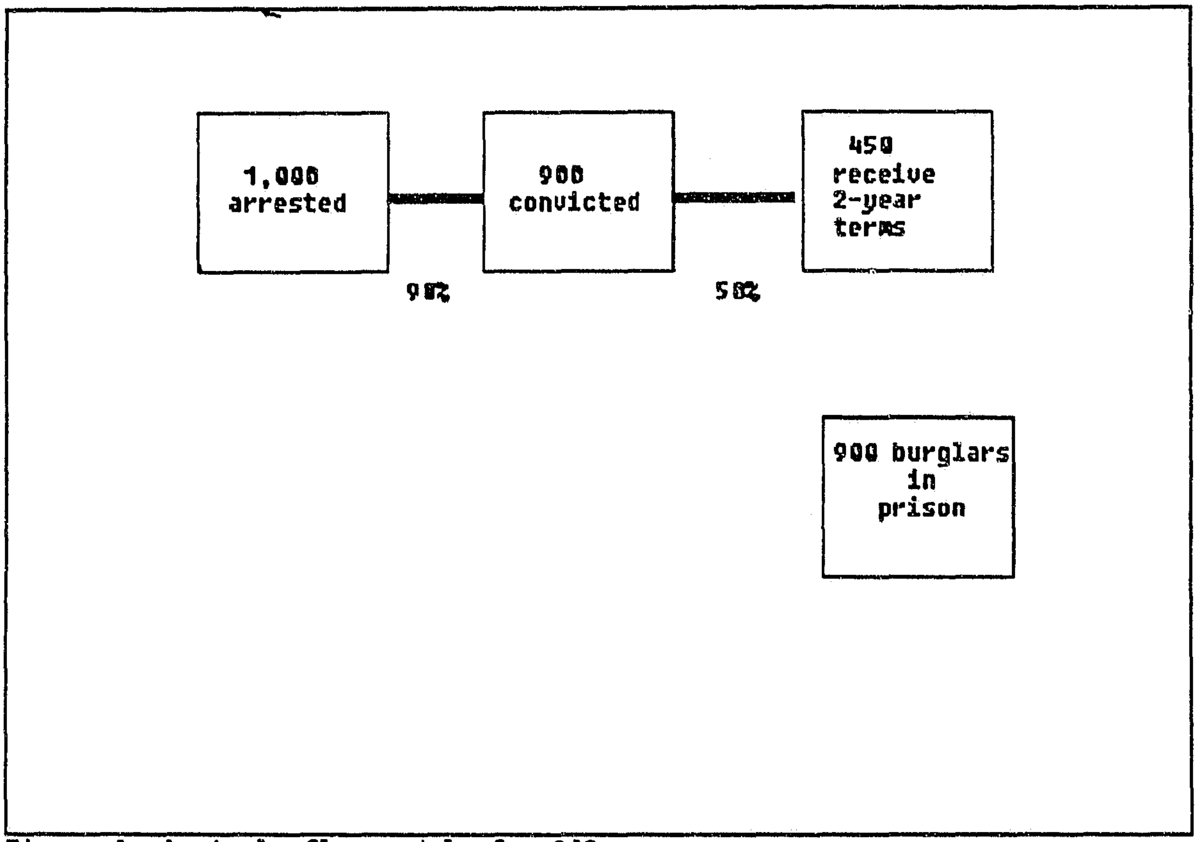


Figure 1--A simple flow model of a CJS

- methods for predicting growth
- methods for estimating impact
- methods for handling existing stocks or backlogs
- implementation effort
- cost of using.

### 1.1.1 Focus

Some CJS models focus on narrow subject areas such as plea bargaining or the impact of sentencing guidelines on prison populations. These models may represent only portions of the CJS, or they may permit only limited types of policy changes to be simulated. Other models have a wide reach, encompassing the entire criminal justice system from arrest through corrections.

A model's reach has little to do with its technical sophistication. Models that do nothing more than project the number of arrests can potentially be very sophisticated; by contrast, the model in figure 1 encompasses the entire criminal justice system but is unsophisticated.

### 1.1.2 Flows, stocks, and system resources

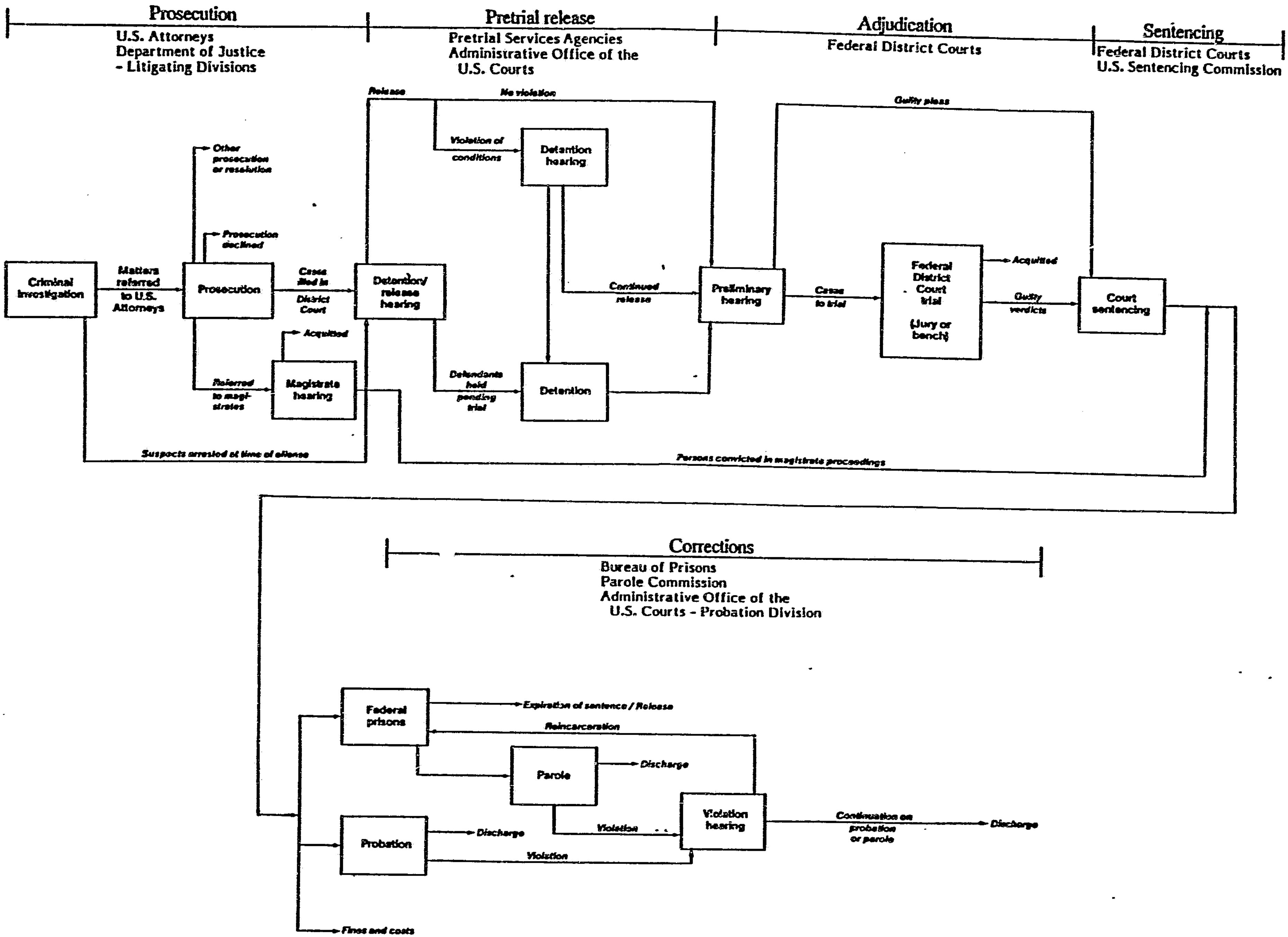
Most simulation models represent "flows." For example figure 1 shows that 900 offenders flow from an arrest to a conviction, and 450 flow from a conviction to a prison term. Usually the flows consist of offenders, defendants or cases, although other units are sometimes used, such as the rate of utilization of prosecutors, judges, and defense attorneys. A flow model of the criminal justice system can be detailed, as in the flow model of the Federal CJS in figure 2. Were this figure to depict a model disaggregated by offense type, the model's complexity would be even greater than is suggested by the diagram.

In Figure 1, the number of offenders in prison is not a flow; it is called a "stock." It is modeled analogously to water in a bathtub, which increases and decreases with the flows of water coming in through the intake and going out the drain. The total volume of water in the tub at a given time is the stock of water; likewise, the number of people in prison at a point in time is the stock of prisoners.

Of what use are simulations of stocks and flows? Some models report the stocks and flows but leave policy implications to the model's users. Other models go beyond this to provide additional information. The model in figure 3 (discussed subsequently), for example, projects not only the number of people who will be in prison at different times, but also the costs of caring for these inmates (assuming an annual cost of $15,000 per inmate). Many CJS simulations are sufficiently sophisticated to assign costs to each separate component of the justice process--the costs of attorneys to prosecute and defend cases, the costs of judges to hear cases, and so on.

# Federal Criminal Case Processing

Figure 2



1.1.3 Dynamics of system adjustments

While most CJS models simulate flows, not all of them simulate stocks. A few simulate only stocks. The ways that stocks and flows are simulated differ across models. Some models provide "steady state" descriptions only; others provide information about the dynamics of the flows and stocks.

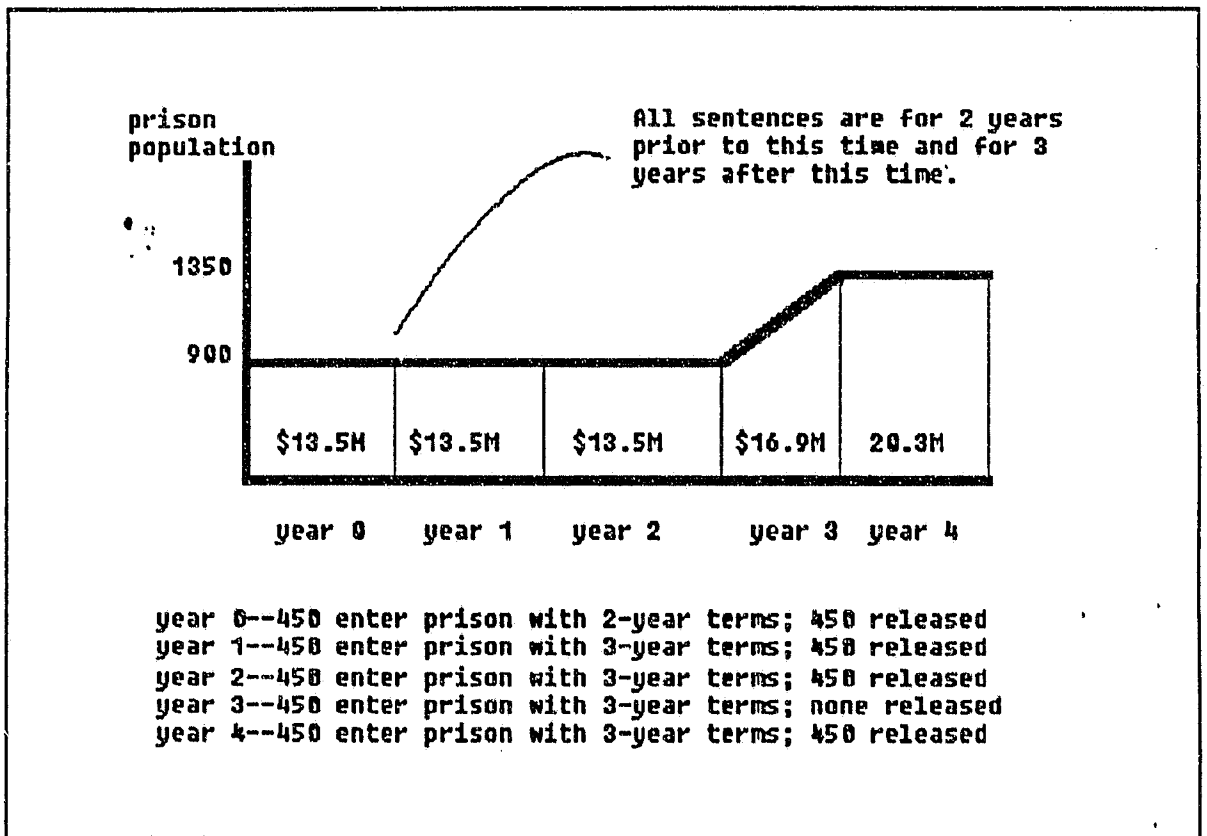



Figure 3--Illustration of a dynamic adjustment process

In our discussion of changing the average prison term from 2 years to 3 years in figure 1, we noted that the stock of prisoners would increase from 900 to 1,350 prisoners. However, it is apparent that this change would occur over time, as illustrated in figure 3. Because in the model the same numbers of offenders are entering prison each year, the impact of an increase in the length of a prison term would not be manifest for two years, when the one-year extension to time served would become operational. After two years, the prison population would grow, because inmates are experiencing the longer sentence instead of being released. The stock of prisoners would stabilize at the end of year three, when the flow of offenders into prison would equal the flow of offenders out of prison. This is the steady state situation: the constant level of inmates in prison after the dynamic process of growth is completed.

While the process modeled in figure 3 is not realistic, it illustrates that some models can describe both stocks and flows, and can handle both dynamic and steady state situations.

Adequate modeling of the dynamics of the adjustment process is crucial to some applications of simulation models. Steady state solutions are obviously pertinent if they will occur in 10 minutes or 10 weeks, but their implications are different if they will not occur until 10 months, or especially 10 years, in the future. CJS simulations have varying abilities (and needs) to model the dynamics of the adjustment process.

System dynamics are expressed in flows and stocks. In subsequent sections of this paper, we discuss flows, stocks and system dynamic concurrently.

### 1.1.4 Methods for handling existing stocks or backlogs

Referring back to figure 1, this simple simulation translates 1,000 arrestees per year into 900 years of prison time. If 1,000 people were arrested every year, the stock of prisoners would always be 900. If arrestees varied from year to year, however, the future stock of prisoners could not be determined for any specific point in time without additional information about the beginning stock of prisoners.

Impact models differ regarding how they acquire information about beginning stocks. Some models require the user to provide data about level of stocks at the time that the simulation begins, and data about how these existing stocks will dissipate over time. Providing such data can be demanding. Consequently, some modelers have devised alternative ways to infer the beginning level and rate of dissipation of existing stocks based on current flows and assumptions about how those flows have changed over time.

As was true of system dynamics, discussions of how one models existing stocks cannot be divorced from the general discussion of flows and stocks. In subsequent sections, we discuss these subjects together.

### 1.1.5 Level of detail about the units of analysis

Models vary in their attention to details about the units of analysis (persons, cases, defendants, etc.). For example, the model represented in figure 1 pertains to burglars. It is more detailed than a model that pertains to all felons without distinguishing the type of felony, but less detailed than a model that distinguishes between burglars with prior criminal records and those without, or between burglars carrying weapons and unarmed burglars.

Level of detail in a model affects its utility. An aggregate model of the flows and stocks of all felony crimes may not be useful if one is interested in simulating a policy change that applies just to burglars. Similarly, a model that pertains to undifferentiated burglars alone may be inadequate to simulate a sentence enhancement regarding possession of a gun during a burglary.

Models that take into account some details of the units of analysis that flow through the model are said to be "disaggregated." Disaggregation may be by offense type or offense characteristics, as mentioned above, or by offender characteristics such as age, sex, race, criminal record, or security level of supervision needed when incarcerated.

Attention to detail is not itself a standard by which a model should be judged; more detail is not always better and is almost always more expensive to simulate.

### 1.1.6 Level of detail of CJS processes

Simulation models of the CJS typically have "branches" along which entities flow. The model in figure 1 provides minimal CJS process detail; it has only two different branches. Arrestees are either convicted or they are not (the first branch); those who are convicted either go to prison or they do not (the second branch.)

In contrast to the model implicit in figure 1, other simulations provide great detail about the CJS. For example, a simulation of the Federal justice process that was developed by Mathematica has 200 unique nodes where branches enter and exit; many of these branches have more than two forks. A profusion of branches makes a model more realistic but not necessarily more useful than a model with fewer branches.

When entities spend time in a state that is on a branch between two stages of the system, we say that a "queue" has developed. A queue is a stock, as we used that term earlier, but the term queue is usually reserved for stocks that develop while waiting for a system resource (such as an attorney) to process a simulated unit (such as a case). A model may or may not permit estimating the time delays involved when flows move along its branches.

### 1.1.7 Methods for projecting growth

The model in figure 1 projects prison populations given the number of arrests that will occur during the future. However, the model does not itself project the number of arrests.

All the simulation models discussed in this paper require the user to provide input about growths in inputs. Some of the models assist the user in carrying out this task, while others do not.

Statistical techniques exist for projecting system inputs, especially for projecting arrests. These techniques presume that historical correlations--say between arrests and population statistics--are sufficiently stable that they can be used to project future trends. Unfortunately, stability cannot always be assumed; for example, past forecasts that did not anticipate the Nation's enhanced commitment to drug-law enforcement have reportedly underestimated current arrests.

### 1.1.8 Methods for incorporating impacts

Some models simulate policy changes by having the user modify input into the system. For example, had the model in figure 1 been developed for a computer, the prison impact of an increase in arrests could have been simulated by imputing first 1,000 arrests and then 1,500 arrests. The computer program would respond with the expected impact on the prisons.

Other models simulate policy changes by having the user change the parameters that are used by the model. Again using the model in figure 1 as an illustration, a policy that reduced the conviction rate from 90% to 80% could be simulated by telling the computer that the branching ratio should be changed from 0.90 to 0.80.

In some models, the user can change parameters while the program is being run, either by responding to prompts from the computer to make changes, or by "feeding" numbers to the computer through a data file. In other models, however, a programmer is required to modify the computer program in order to incorporate changes in parameters such as branching ratios.

### 1.1.9 Implementation effort

Most of the simulation models reviewed in this paper were developed expressly to be transferred to environments that differ from the development setting. The original development costs were likely extensive, as developers incorporated features into their models that made those models adaptable to diverse environments. We do not discuss such development costs, which are generally unknown and, anyway, of little interest to anyone but the developer and the developer's funding source.

Instead, we focus on the costs of adopting and adapting a simulation model once the initial development has been completed. That is, we focus on the costs to a user after he or she has purchased a simulation from one of the vendors who have developed and who market such products. Implementation costs vary markedly across extant simulation models.

### 1.1.10 Cost of using

Some simulation models run on mainframe computers. Their use requires the costs of the mainframe as well as a staff experienced with mainframe applications. Many such models require considerable computer time, so that the cost of even a single simulation "run" can be hundreds of dollars or more. In contrast, some other models have been developed on microcomputers or ported from the mainframe to a microcomputer environment. Because microcomputers typically require less sophisticated users, and because execution time on a microcomputer is virtually free, microcomputer models can be less expensive to operate.

In addition to the costs associated with computer usage and professional staff, computer models differ regarding the demands that they make on the policy analysts who is using the simulation. Some simulation models provide an interface query that facilitates the policy analyst's ability to simulate different events. That is, the user literally sits in front of a computer screen and is guided by an expert system through a series of questions that allow different events to be simulated.

Other simulation programs are less user-friendly. They may require the user to provide assumptions embodied in a special data file that is read by the simulation. Or they may require the user to enter these assumptions directly into the computer model. When the latter technique is used, the user must "reprogram" the computer simulation. This reprogramming may be as simple as changing the numbers in a spreadsheet, or as complicated as altering, recompiling, and running the computer code.

Another cost of use is "waiting time." Some simulation models are designed to provide instant feedback; that is, the simulation takes only seconds or minutes so that the user can see and presumably act on the result from the model immediately. Other simulation models are designed to provide answers hours or days after a policy question is posed.

Cheaper is only better when the less expensive model will satisfy the user's needs. This truism aside, among simulation models that are equivalent on other dimensions, cost of use may be a determinant criterion.

## 1.2 Focus of the review

Many varied sources can produce changes that have an impact on the criminal justice system. Legislatures' actions can create impacts; for example, the Federal Anti-Drug Abuse Act of 1986 instituted new mandatory minimum prison terms for drug-law violators and increased mandatory minimum terms when they already existed in law. Administrative decisions create impacts; to illustrate, mandatory sentencing guidelines can shift both the incidence and duration of prison terms. Executive branch changes can create impacts; as an example, increasing the number of undercover agents devoted to drug investigations can increase the number of arrests for drug-law violations.

The ability of computer models to simulate the impact of these and other changes in legislation, administrative policy decisions, and changes in executive branch resource allocations are the subject of this paper. Also considered are models that project demands on the system in the face of growing populations of offenders. We do not discuss computer-based information systems, although there is often a close relationship between the utility of a computer simulation and the availability of requisite data about the CJS.

We are interested in models that project future impacts, meaning impacts that have not yet occurred. This paper does not discuss methodologies used to evaluate and measure impacts that have already taken place.

## 2. Types of models--a review

We have reviewed most computer models that have been used to project policy impacts on the criminal justice systems. We have classified these models into three groups. The first group consists of statistical models. The second consists of disaggregated simulation models. And the third consists of microsimulations. Such models are also known by other terms, and the distinctions are somewhat artificial. But they are useful for describing a model's strengths and limitations.

### 2.1 Statistical Models

Although numerous statistical models are used to project future demands on the criminal justice system (see Block, Miller and Hudson, 1983), they all have at least one element in common: all use data to discern past patterns and project those patterns into the future.

Beyond this commonality, statistical models take several forms. Some are simple trends: if in 1989 there were 100 robbery cases, and in 1990 there were 110 robbery cases, the projection might be that in 1991 there would be 121 robbery cases (a 10% growth.) Trend models can be more realistic than this illustration, but the point is still that the projections will be accurate only when past trends actually extend into the future.

Unfortunately, for the purposes of impact analysis, disruption of a trend is often the reason for and the result of a policy intervention. Simple trend analysis, then, may have only limited utility for impact analysis.

Multivariate statistical models can be more valuable than simple trend models, because multivariate models can account for conditions under which the trends hold. For example, a multivariate model might account for future population growth, especially changes in the population's age and race composition, or for the resources devoted to law enforcement. The projections would be conditional on demographic and resource changes.

However, predictions based on a multivariate model sometimes have a fatal flaw. Unless the conditions that are taken into account by the statistical analysis are those that are subject to policy manipulation, the predictions based on multivariate statistical analysis cannot be used readily to simulate the consequences of those policy changes. For example, a statistical model of arrests for transporting drugs in Arizona cannot project the effects of increasing the number of border guards when the statistical model is based only on the state's race, age and sex population distributions.[2]

Even more sophisticated models--called structural equation models by some--have been advocated or developed for the criminal justice setting. A distinctive feature of structural equation models is that they capture, through multiple equations, the complexity of a criminal justice system. Although structural equation models have been developed for justice systems (Fox, 1978), and although some have advocated the use of such models for judicial impact analysis (anonymous, 1980), others have concluded that data

---

[2] Other, more technical, problems limit the use of multivariate models for impact analysis. Essentially, a multivariate statistical model relates a dependent variable (Y) to one or more independent variables (X1, X2, and so on). Predictions based on this relationship are the most reliable when the independent variables have values that are within the ranges or close to the ranges that have been observed historically. As the values for the independent variables change to lie more-and-more beyond the range of historical precedents, predictions become increasingly inaccurate. This technical problem is especially salient for impact analysis, where policy interventions often modify one or more independent variables.

limitations and the lack of adequate theory preclude the use of structural equation modeling as a tool for impact analysis (Boyum and Krislov, 1980.)

Aside from the above comments, this paper does not review the use of statistical modeling in support of justice impact analysis. Nevertheless, the topic is pertinent for several reasons. One reason is that the General Accounting Office, in response to a Congressional mandate to develop Federal impact models, is developing statistical models[3] of Federal case processing (Bill Jenkins, personal communication, Feb. 1990). A second reason is that many computer simulation models, which are the focus of the following discussion, embody within the computer program the results of univariate, multivariate and structural equation models. Thus, to say that statistical models have had little or no role in impact analysis would be misleading. A third reason is that all the models reviewed subsequently rely on some type of statistical analysis or expert opinion to project future inputs into the model. That is, computer simulations require their users to provide projections of future crime, or future arrests, or future convictions; these projections are not (as a rule) provided by the computer model itself.

Thus, statistical analysis is central to computer simulations and we emphasize that sophisticated statistical analysis often lies behind models that otherwise may appear uncomplicated.

## 2.2 Disaggregated Models

The predominant characteristic of disaggregated flow models is that they classify units of analysis into groups and simulate CJS operations by modeling flows between important processing stages. Three disaggregated flow models are described in this section: JUSSIM, IMPACT, and the Community Corrections Planning Simulation.

---

[3] These statistical models are based on regression analyses of past Federal caseloads. Expenditures for law enforcement, and other relevant variables, are being used to explain past trends and to project future caseloads. The GAO report is not yet available.

### 2.2.1 JUSSIM

JUSSIM is probably the best known CJS computer simulation. With an initial field implementation in the early 1970s (Cohen et al., 1973), it is certainly one of the most venerable. We discuss two operational versions of the JUSSIM model, both of which are actually based on a JUSSIM look-alike, PHILJIM (Renshaw et al., 1972). The first operational version has been developed for microcomputers using a UNIX operating system (Cushman, 1989); the second is a version developed for microcomputers using a DOS operating system (Institute for Law and Justice, 1990).

focus--JUSSIM allows the CJS to be modeled from arrests through corrections. The reach of the JUSSIM model can be tailored by the user, so versions of JUSSIM differ among sites. A JUSSIM simulation used in Dade County, Florida (Silbert, 1989) begins prior to arrests--at crime incidents--and ends at jail or probation. Similarly, the JUSSIM model used in Santa Clara County begins at crimes and traffic infractions and terminates with corrections.

flows, stocks and systems resources--JUSSIM is a flow model; it does not accumulate stocks. For example, the model might be used to predict that street sweeps designed to arrest drug dealers will increase the number of inmates entering prison by 25%. However, JUSSIM would not show how the prison stock (total prison population) would be affected.

JUSSIM does simulate resource demands. As cases flow through a CJS, those cases generate resource demands: prosecutors are needed to prosecute the cases, defense attorneys are required to defend the case, judges are needed to try the cases, and so on. The JUSSIM model allows the user to associate resource demands with flows. When the program is run, the resource demands are accumulated and reported.

JUSSIM is a steady state model. As a change is made at one point in the system (say arrests are doubled), the system impact is instantaneous (say the prosecutor's workload increases by 35 percent.) JUSSIM cannot model the dynamics of the change; it cannot model the distribution of system impact over time. Because JUSSIM is a steady-state model, it does not need procedures for introducing existing stocks into the simulation.

level of detail of the units of analysis--Units of flow are divided into a small number of groups. For example, burglary might be defined as a group. If so, the program would use branching ratios assigned to burglary to simulate the flow of burglars though the system. No smaller unit of analysis would then be possible, so the user could not ask the program questions about burglars with firearms or burglars with prior records. However, higher levels of aggregation are permitted; it is possible to ask questions about all felons or all felons accused of property crimes.

level of detail of CJS process--JUSSIM is rich in process detail. Figure 4 shows the flow model used by Santa Clara County's JUSSIM model. The JUSSIM user can specify the resource demands made by cases processed through each step of the CJS, so the model can be used to estimate resource demands at various stages.

A distinctive characteristic of JUSSIM is an extensive and detailed reporting capability focused on resource consumption at each stage of the CJS. Above all, JUSSIM is a model of resource demands.

method for predicting growth--JUSSIM is a steady-state model, meaning that it does not report the dynamics by which the CJS is affected by an intervention. For example, if the number of drug arrests increases from 1,000 per year to 1,250 per year, JUSSIM will indicate where this increase affects demands on system resources, but it will not indicate when these demands will be felt.

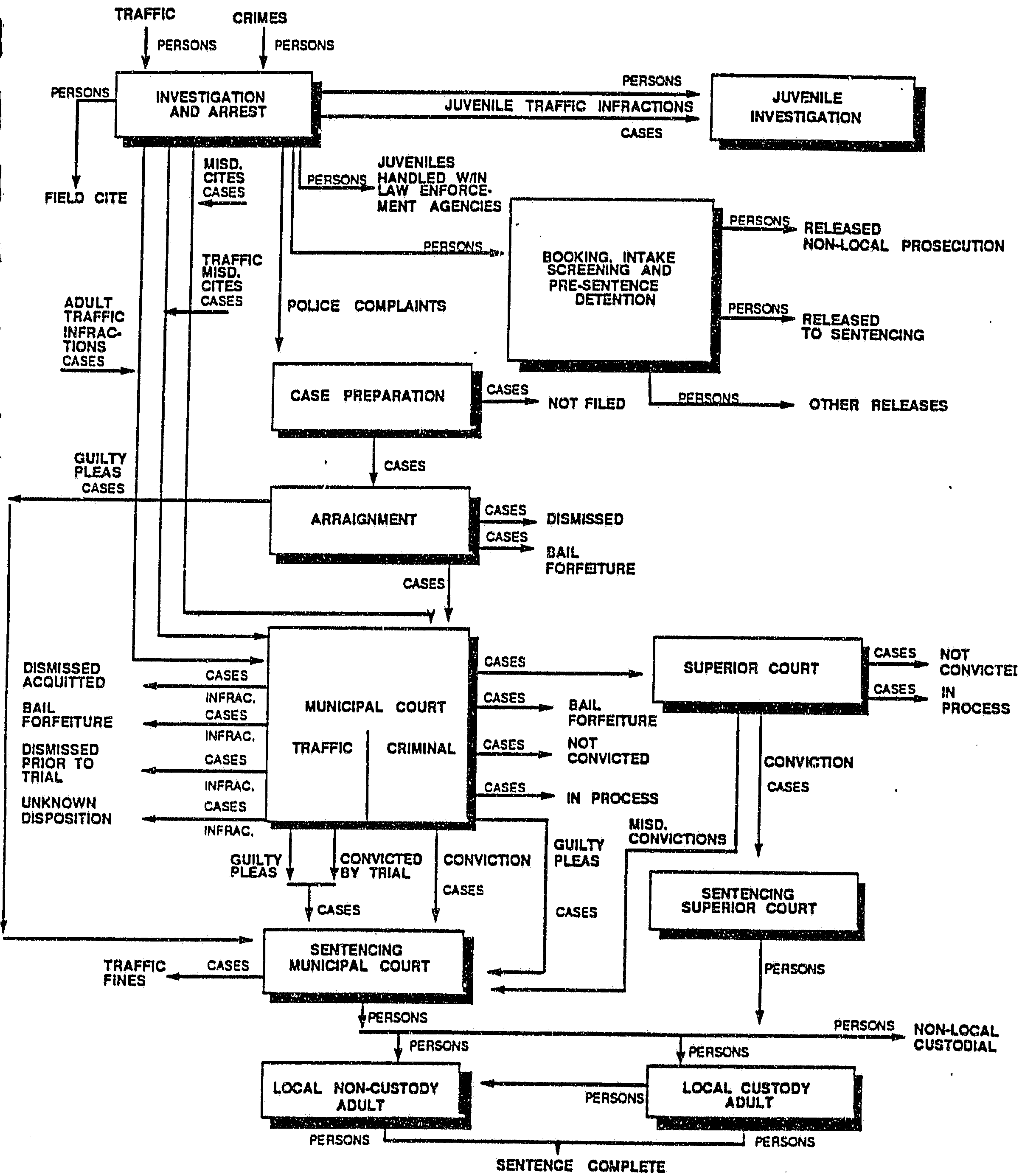
JUSSIM does not have any special internal facility to project case growth. For instance, if the user wants to simulate an increase in drug cases from 1,000 to 1,250, the user must run the program twice: once with 1,000 drug cases as input and once with 1,250 drug cases as input. However, JUSSIM's system designers have greatly facilitated the user's ability to conduct multiple runs and to compare the results, so projecting case growth does not greatly burden the user.

Blumstein, Cohen and Miller (1980) have used demographic information to predict growth in future cases, and they have entered these predictions as input into JUSSIM. Thereby, projections can be incorporated into JUSSIM, but the projections are not an integral feature of the model itself.

method of estimating impact--JUSSIM has three methods for simulating the effects of policy changes. The first method, already discussed, is to change estimated inputs into the criminal justice system, possibly by running JUSSIM multiple times. The second method is to change branching ratios. Versions of JUSSIM available for microcomputers allow the user to interactively modify branching ratios, so any policy change that can be represented as resulting in new branching ratios can be readily simulated in JUSSIM. The user can also easily add new stages to the system, and new branches, when impact is expected to take this form. A third method of anticipating the impact of policy changes is to modify the resource costs to process defendants along existing branches. The model can be run with old resources costs and then rerun with new resource costs; the results may then be compared.

implementation effort--The Center for Urban Analysis has developed a user interface for using JUSSIM on a microcomputer running under a UNIX operating system. A detailed user guide and tutorial is available. The user guide and tutorial were written for people with no prior computer experience.

Figure 4--JUSSIM Flow Model for Santa Clara County




Source: Center for Urban Analysis (1988) p. B2

The Center for Urban Analysis program can be used to tailor the JUSSIM program for that user's CJS. The user may specify the stages that will be included in the simulation, the branches that will exist among these stages, the units that will flow through these stages, and the resources that will be consumed by these flows. No programming experience is required.

Although programming is minimized, the user must provide considerable information. The number of cases entering the system must be provided, as must the branching ratios and system costs. Because data about branching ratios may not be available for all units in the analysis, the program allows the branches to depend on larger aggregates (such as all felony cases rather than all burglary cases.)

The Center for Urban Analysis program assists with this data entry. The program prompts the user to enter data and saves the data for recall and modification.

The Institute for Law and Justice (ILJ) is completing development of a microcomputer version of JUSSIM. Our review is based on a test version of the program, the introductory chapter of the user's manual, and test results provided by ILJ.

The ILJ version of JUSSIM, which operates under the popular DOS operating system, probably has an advantage over the Center for Urban Analysis's version, which operates on the less widely used (on a microcomputer) UNIX operating system.

It seems to offer other advantages. The ILJ version is menu driven, meaning that a user can perform most operations by making selections from a menu of choices that appear on the computer screen. The ILJ version has a graphical orientation, which allows the user to construct, run, examine and modify his or her flow model interactively. In contrast, the Center for Urban Analysis version assists the user to write a control file that can be used in a separate program to execute JUSSIM.

cost of using--A run of the JUSSIM program requires no more than a few seconds of computer time. It would seem that computer usage is an insignificant consideration in using JUSSIM.

As we stated earlier, the Center for Urban Analysis version of JUSSIM requires a microcomputer operating under a UNIX operating system. This requirement may increase the computer sophistication of the program's users. The ILJ version, which operates under DOS, requires WINDOWS (an operating environment designed for DOS) and a mouse (a pointing device much like a joystick used to operate computer games.) WINDOWS and a mouse together cost less than $500.

### 2.2.2 IMPACT: A computer simulation from CJSA

The Center for Decision Support of the Criminal Justice Statistics Association has developed a computer simulation for microcomputers using the DOS operating system. Our description of this program is based on review of a demonstration disk and the IMPACT User's Guide.

focus--IMPACT can be used to model any stage or multiple stages of the criminal justice process that generate a stock. The model's greatest utility is projecting jail, prison, probation or parole populations, however. Through the rest of this section, we will discuss IMPACT as if it were designed exclusively to project prison populations.

IMPACT is a flow model. The user must design and interactively program the flow. A typical application would have the user specifying a flow from arrests, through convictions and sentences, into a stock of prison inmates. As a flow model, IMPACT is more limited than JUSSIM. The flow in IMPACT is linear with only one branch indicating that the defendant continues in the CJS; contributions to the prison stock can be made only along this linear flow. (Other branches would indicate diversion, dismissal, acquittal, nonprison sentence, or any other event that does not contribute to the stock in question, namely the prison population.)

stocks, flows, and systems resources--Although IMPACT is more limited than JUSSIM as a flow model, IMPACT--unlike JUSSIM--is programmed to project stocks.

The first method for building stocks is to assume that the time served in prison is a random variable distributed exponentially. This assumption, coupled with data about beginning prison populations, average time served, and the future arrival rate of new prisoners, suffices to project future prison populations.

The second method for building stocks is to use the distribution of time served by sentenced offenders to infer both the residual prison time for offenders who are in prison at the beginning of the population and the time to be served by people just entering prison.

Neither the first nor the second method requires the user to measure the residual length of time for offenders who are in prison at the beginning of the simulation. That is, the user need not measure prison stocks at the beginning of the simulation.

IMPACT allows the user to specify system capacity. Based on that specification, IMPACT will report the difference between capacity and projected demands. IMPACT allows the user to specify resource costs. IMPACT will project future costs.

level of detail of the unit of analysis--The user can determine the unit of analysis. Typically, offense types would be an appropriate unit, although the simulation might be conducted with age, sex, or race as the unit of analysis. The unit of analysis can have two or more dimensions, for example, offense type by age.

A special utility allows the user to aggregate detailed units into less detailed units (men and women can be combined, for example) and rerun the model. A second utility allows the user to run separate simulations (on men and women, say) and then combine the results into an aggregate profile of demands on the prisons.

level of detail of CJS processes--Process detail is limited in IMPACT. As mentioned, the flow is linear, with branching ratios used to prune the number of people who will enter prison. The program does not appear to have the ability to report flow statistics with the exception of the number of people entering prison.

The user controls the time units (years, months, days), the length of the projection period, and the total number of time points to be reported during that projection period. The user controls the branching ratios. IMPACT provides a utility for computing branching ratios from raw data. The user also controls whether the branching ratios will change across time and whether the branching ratios will differ across the units of analysis. As examples, a conviction rate might have been 0.75 in the past and might be 0.85 in the future; a conviction rate might have been 0.75 for robbers and 0.65 for burglars.

method for predicting growth--IMPACT gives the user two methods for projecting future arrests (or other entry points into the model). The first method is trend analysis. IMPACT uses either geometric or additive weights to project past arrival rates into the future. A growth path can be added by the user. The second method uses linear regression to project future arrivals using explanatory variables provided by the user. Whichever method is used, the user can override a projection by substituting the user's own estimate of future arrests.

methods for estimating impact--Policy changes can be introduced into IMPACT using one of three techniques. The first technique is to use the program's interactive dialogue to change the branching ratios. The second method is to change the arrival rates. The third method is to alter the average (and distribution of) time served in prison. A special utility allows projections under different simulations to be compared in a "difference table" that contrasts the demands on prison resources for two simulations.

implementation effort--The program is menu driven. Although we reviewed a demonstration program rather than an operational version, the menu choices appear suitably designed so that a nonprogrammer could construct and execute an IMPACT run. Data requirements are minimal. Documentation is thorough.

The program provides several special features. One feature is a macro capability that allows the user to reissue frequently used commands without having to enter all the required keystrokes. A second utility graphs trends revealed by the simulation.

cost of using--IMPACT currently sells for $750. It runs on a microcomputer. A typical run requires no more than a few seconds.

### 2.2.3 Community Corrections Planning Simulation (CCPS)

CCPS was developed by Ken Carlson, of Abt Associates, for the National Institute of Corrections. The program and documentation are available from NIC. The simulation can be run on a personal computer equipped with Lotus 1-2-3 or other spreadsheet that is capable of importing Lotus files.

focus--CCPS is limited to community corrections, especially probation and parole supervision. The program's user specifies the number of people received for supervision during a year. The program estimates the resource demands that these people will make on the supervision system. To determine these demands, the program requires information about supervision levels, length of time at specified supervision levels, special conditions of supervision, revocation rates, and costs associated with providing supervision and services associated with special conditions of supervision.

flows, stocks and systems resources--CCPS is a model of stocks. Based on the number of convicted robbers entering the supervision system, for example, the model reports the number of robbers at each supervision level (intensive, maximum, medium and minimum, for instance), the services received by these robbers (hours of drug testing, counseling, and outpatient treatment, for example), and the total costs of providing these services. Costs are classified by personnel costs (direct and indirect) and by other costs.

CCPS provides a steady state solution. Consequently, it makes no provisions for existing stocks at the start of the simulation.

level of detail of the unit of analysis--The unit of analysis in CCPS is a group of offenders. An example from the documentation uses offense groups: homicide, burglary/larceny, robbery, assault, sexual assault, drugs, DWI/stolen car/traffic, and other offenses. However, the program allows its user to specify the unit of analysis, suggesting that the user might consider risk level as an alternative to offense type. The user also controls levels of supervision (intensive supervision, maximum supervision, and so on) and the special conditions assigned.

level of detail of CJS process--The unit of analysis can be tracked by supervision level, which can change during the course of supervision and may include revocation. Services are provided to offenders. Costs are assigned to the delivery of supervision and services.

method for predicting growth--The model does not project growth. Given the number of people who are expected to be received during the year, the model provides a steady-state assessment of the yearly demands that will be placed on the supervision system. Given a growing number of people entering supervision, the model would likely overstate the resource demands for the first year and understate demands for latter years. The user is required to determine how the steady state solution will be reached.

By running the model repeatedly, the user can simulate the effect of increases or decreases in the number of people received for supervision, thereby using the model to project future demands. The model does not provide any special facility for anticipating increases or decreases in people to be supervised, although the documentation that accompanies the model makes some suggestions.

methods for estimating impact--The model readily simulates the effects of policy changes that take the following forms: increases and decreases in the number of people who enter supervision, changes in the average length of time under supervision, changes in special conditions imposed on supervisees, changes in the level of supervision (or the length of time that level is provided), and changes in the costs of providing supervision and services. The steady-state consequences of these changes are reported.

implementation effort--CCPS is a clever use of readily available computer software. Programming is based on Lotus 1-2-3 software, perhaps the most widely known computer language for microcomputers. The choice of languages to implement CCPS has several advantages. One is that a user can readily modify the assumptions made by the CCPS program, a prospect that the program's designer recognizes and encourages. A second advantage is that a user can readily change the program itself, adding additional tables, for example, or expanding the number of groups allowed in the programs' current version. A third advantage is that CCPS can be used as an illustration by users who want to design their own simulations. Given the ease with which spreadsheets can be developed with Lotus 1-2-3 (or other spreadsheet programs) and add-on products (such as spreadsheet auditors), CCPS and CCPS look-alikes may be especially attractive when programmer assistance is limited.

cost of using--CCPS is available free from NIC. The cost of Lotus 1-2-3 is \$300-\$400. Computational time is insignificant.

### 2.3 Microsimulations

A microsimulation differs from a disaggregated flow model in that the computer program processes entities, such as defendants or cases, one at a time rather than in groups. From the user's perspective, this distinction can be unimportant. A disaggregated model can potentially divide subgroups so finely that the model virtually simulates the behavior of individuals. On the other hand, a microsimulation that simply processes large numbers of identical individuals in identical fashions is--for all practical purposes--simulating the behaviors of groups.

However, in the models discussed in this section, the processing of individual units is important to the characteristics and capabilities of the models. Adapting a disaggregated flow model to perform a similar simulation would be cumbersome at best.

#### 2.3.1 The NCCD stochastic entity process model

The National Council for Crime and Delinquency's model is a microsimulation of prison populations based on an earlier program developed for the California Department of Corrections. In its mainframe version, NCCD customizes the model to represent the user's CJS, so there is no single NCCD model.

Our review is based primarily on Virginia's (mainframe) version of the model, to a lesser extent, on a (mainframe) version used in Florida, and on a reading of the program's computer code. NCCD has recently released a microcomputer version, for which it has provided a user manual for our review (NCCD, 1990); we comment on how the microcomputer version is likely to change the program's operation.

focus--The NCCD model is focused on prison populations. The program simulates correctional experiences for individuals: time until release, method of release, time on parole, method of release from parole, and additional time spent in prison as a consequence of a parole revocation.

flows, stocks and systems resources--The NCCD model is especially strong when reporting stocks and flows. People entering jail, prison, and parole can be reported on a monthly basis for a 10 year projection period. The total numbers of people in jail, in prison and on parole can be reported for each month of a 10 year reporting period. The dynamics of the adjustment process can be examined by inspection of the monthly reports of stocks and flows. The model does not report system resource demands beyond the numbers of prisoners and parolees under supervision per month.

As input from the user, the program requires current prison populations, current parole populations, and expected future arrivals at prison. The user must indicate how these stocks will be dissipated through time. (The microcomputer version does not require these data about existing stocks; however, the microcomputer version assumes that the dissipation of all stocks follows an exponential process. Such a restrictive dissipation pattern was not required by the mainframe version.)

level of detail of the units of analysis--The NCCD model simulates flows and stocks based on individuals as the unit of analysis. The user determines the number of individuals who enter prison. The model randomly assigns to each individual specific characteristics, such as a date for being released from prison, a release date from parole, potential parole violation dates, dates returning to prison for revocation, and release dates following parole revocation. These random assignments are made so that the simulation of time spent in prison and on parole has a distribution that corresponds to that seen in practice. The model's user must provide these historical distributions. (As mentioned earlier, the microcomputer version presumes an exponential distribution. This assumption reduces data input at some loss of generality.)

The user controls the level of detail by which individuals are distinguished. In Virginia's version of the NCCD model, offenders are classified as "people who received a life term," "juveniles," and "others." The category "others" was broken into grades for purposes of simulating sentences. Because the NCCD model is customized for each user, the units of analysis differ. For example, Florida's version of the NCCD model can distinguish males and females; Virginia's version cannot make this distinction. (In the microcomputer version, the user can modify the units of analysis and the process detail without intervention by NCCD.)

level of detail of CJS processes--Given an entering cohort of prisoners, the NCCD model simulates prison terms, simulates release dates based on good-time provisions and parole decision making, and simulates the outcomes of parole based on historical patterns. The Virginia model distinguishes jail from prison, but it does not model any subprocess within prison, such as correctional security level. Essentially, the model determines entry and release dates, and all output is derived from these determinations.

methods for predicting growth--Given the existing stock of prisoners, the existing stock of probationers, and the future flow of entering prisoners, the NCCD model distributes prison impact over time. The model does not provide any methodology for projecting the future flow of incoming prisoners. These data must be provided by the user.

methods for estimating impact--The NCCD model incorporates policy changes in two ways. First, the user can modify the data fed into the model. For example, the model can be run with prison arrivals growing at a compound rate of 5% per year; then, to simulate a policy change that increases the prison arrivals by 2%, the model could be rerun with a growth rate of 7%. Second, the model allows statutory changes in prison sentences, and these can be phased in over time at the discretion of the user.

The NCCD computer program can be modified by a programmer to introduce different policy interventions into the model. The logic of the model is that the effect of any policy intervention can be simulated provided that the intervention can be represented by a distribution in outcomes. For example, changes in sentencing or parole practices can be simulated provided the user can specify the distribution of sentences that will prevail after the changes have been introduced and the timing of those changes. (The microcomputer version allows all system parameters to change at a time specified by the user. No reprogramming is required.)

implementation effort--The mainframe version of the NCCD simulation is written in PL/1. Changes in the program require an experienced programmer and detailed knowledge of the program's code. As a practical matter, NCCD customizes its generic mainframe model following specifications provided by users. Users do not have access to the source code.

The microcomputer version--written in "C" and dBase--allows the user extensive control over the model's design and operation. As we mentioned above, increasing the user's control seems to have come at the expense of some reduction in generality. Furthermore, it appears from the documentation (p.36) that reporting capabilities must be customized by NCCD for a user's environment.

cost of using--The NCCD model provides a great deal of detail, which is purchased at a price. The user must provide extensive input into the simulation. For example, in the Virginia version of the NCCD model, the user must provide data for 130 variables. Most of these variables have multiple categories, each of which requires datum--for example, a sentence length variable requires the percentage of offenders with one-month terms, two-month terms, and so on.

Fortunately, the program reduces data entry time by facilitating the entry of repetitive data. Nevertheless, data necessary to "prime" the NCCD model is a major cost of using the model.

As a microsimulation, the NCCD model processes each unit of analysis case-by-case. A case-by-case method is an expensive way to simulate CJS processing. Because of the cost of mainframe computing time, a microcomputer version of the NCCD model--which requires less data--seems especially welcome. We were unable to determine the extent to which the microcomputer version gains or loses functionality compared to the mainframe version.

### 2.3.2 Justice Impact Analysis

A model to conduct justice impact analysis (JIA) was developed by Celeste and Douglass (1980) for the U.S. Department of Justice. The model appears to have had little or no use during the last decade. Our review is based on the authors' final report.

focus--JIA is a microsimulation of the Federal justice system from the point that a "matter" is initiated until the case is terminated.[4] Matters and cases are classified as criminal, civil, Federal

---

4 A "matter" is any item received by an assistant U.S. Attorney for his attention. An arrest is one type of matter. So too are requests from federal agencies for the U.S. Attorney's attention to a potential legal issues.

jurisdiction, and diversity. The model purports to represent virtually every stage in the justice process where these matters and cases can affect resources.

According to the model's developers: "In order to account for the interaction and competition between cases for finite judicial resources (judges, attorneys, staff, courtrooms, etc.)...this approach modeled the justice system in terms of resources available. The model was programmed to represent a medium-sized Federal District Court.... Cases of different types were then allowed to compete for judge and attorney time for courtroom services...as well as for services outside the court." (Celeste and Douglass, 1980, p. I-16.)

flows, stocks and systems resources--JIA analyzes the flow of criminal and civil cases through a Federal court. Flows depend on both the branching ratio and the resource demands required to complete the various stages in the case's life. Because resources in the form of judges and attorneys are limited, not all cases can be processed simultaneously. Because cases must wait for resources to become available, delays occur at each stage in the justice process. A notable feature of JIA is its ability to model these delays.

JIA records flows, stocks and system resource demands. Stocks are mostly in the form of queues, as cases wait for the availability of resources required to process those cases. Growing delays reflect demands that are disproportionately large relative to the available resources.

JIA has a feature that is unique among the models considered here. Criminal cases--the domain of the other models discussed here--tend to be processed sequentially. That is, for criminal cases, stage C virtually always follows stage B, which almost always follows stage A, and so on. Case processing for civil cases is not linear.

Recognizing this, JIA allows civil cases to cycle through stages until all resource demands have been satisfied.

The beginning stock seems to be established by running the program to establish a steady state. Likewise, dynamic adjustments can be modeled by starting and stopping the model at different times, while observing the level of stocks.

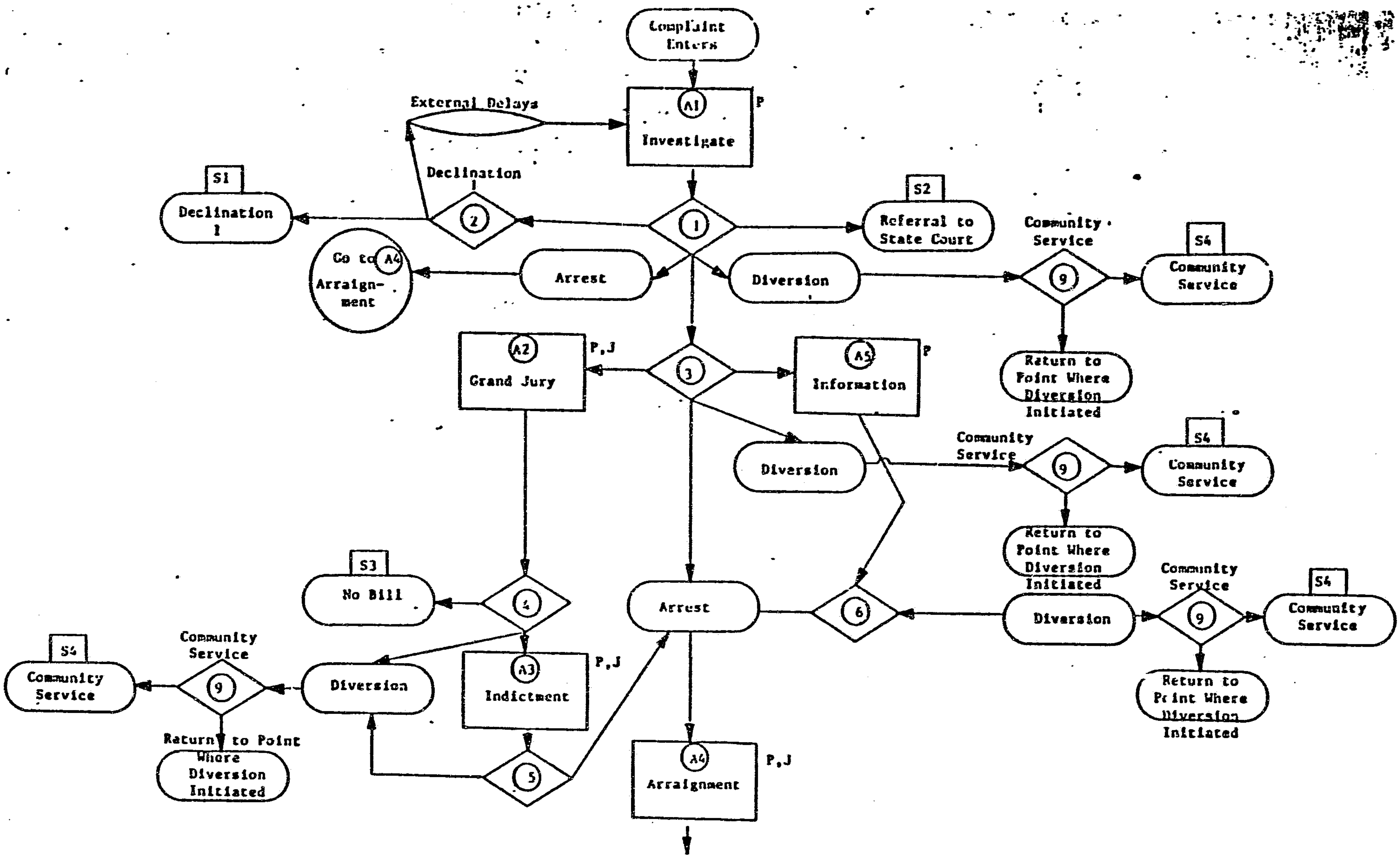
level of detail of the units of analysis--Cases are the unit of analysis. Each case is assigned attributes that determine a distribution that, in conjunction with a random number generator, is used to select for each case the branching ratios and resource demands placed on the system.

level of detail of CJS processes--To provide a feel for the level of detail in the JIA model, figure 5 reproduces the flow model for criminal cases from the point at which a complaint enters the system (the point where an assistant U.S. Attorney first knows about the matter) through arraignment. The detail is apparent. An investigation alone can have the following outcomes: declination, referral to state prosecutors, diversion, arrest and information, or Grand Jury.

methods for predicting growth--JIA does not seem to have a provision to project growth in criminal or civil cases. The user could simulate such conditions, however. JIA simulates the arrival of matters so that the arrival rate is consistent with historical observations. Presumably, this arrival rate could be changed to reflect increases or decreases in specified types of cases, so two runs with different assumptions about case arrivals could be compared.

methods for estimating impact--Policy changes could be modeled in one of several ways. The model's case arrival rate could be modified to reflect growth or decline in cases received by the courts. For example, the model's authors used this technique to simulate the

Figure 5--A Flow Model from Complaint Through Arraignment




Source: Celeste and Douglass (1980) p II-32 effect of removing certain diversity cases from Federal jurisdiction. The attributes of cases could be modified, or the effect of given attributes could be modified, to change the branching ratio and resource demands. It is not clear from the documentation whether these types of changes require programming modifications.

JIA was one part of a process by which Celeste and Douglass proposed to conduct legislative impact. An additional important ingredient in this process was expert opinion. Once the model was run, experts were expected to inspect the output and suggest how the system might adapt to the demands on system resources. For example, increasing court delay for major frauds might trigger an increase in the declination of minor larceny cases. If an expert assistant U.S. Attorney indicated that this adaptation would be likely, JIA would be modified to increase the probability of declinations for larceny cases, and the simulation would be rerun. Another round of inspection--adaptation--modification--rerun would follow. The process would continue until no additional adaptation seemed likely.

implementation effort--JIA is a detailed computer program written in FORTRAN and Q-GERT, a simulation programming language that is especially well adapted for simulations that involve queues. Although development time is unknown, it is apparent that programming or modifying a model such as JIA requires detailed knowledge of case processing and the employment of professional programmers.

cost of use--JIA appears to be expensive to run. According to the government's project monitor, this expense was an important factor that caused the Department of Justice to abandon use of the model. Additionally, cost of use would probably be higher than test costs experienced by the model's developers, as the existing version of JIA is limited to simulating the workload of a medium sized Federal district court.

### 2.3.3 Structured Sentencing Simulation Model

The Structured Sentencing Simulation (SSS) model was developed by Kay Knapp and Ron Anderson of the Institute for Rational Public Policy. Designed specifically for anticipating the consequences of sentencing reforms on correctional resources, SSS is an adaptation and extension of an earlier simulation used by the Minnesota Sentencing Guidelines Commission (Parent, 1988). Our review is based on the user's manual, several updates that have been reported in notes to users, and a demonstration of the program using data from Minnesota.

focus--SSS is a microsimulation that has been rewritten and ported from a mainframe computer to a microcomputer environment. The program was originally designed to simulate the impact of changes in Minnesota's sentencing guidelines on Minnesota's prisons. While sentencing guidelines remain at the model's core, the model's developers have added parole guidelines as an additional feature. Also, SSS provides a more expansive view of "correctional resources" than did its progenitor. Specifically, SSS simulates the impact of sentencing on prisons, jails (county-by-county), probation and parole supervision, and several intermediate sanctions.

The simulation begins in year one with a sample of sentenced offenders, each of whom has a weight assigned so that the population of offenders can be reconstructed from this sample. The sample is used to project future populations of sentenced offenders by using a population growth path that is under the user's control.

Each sentenced offender has attributes that are provided through the program's data file. Some of these attributes are used to establish reporting categories: gender, race, age, offense type, and region of the state. Otherwise, with the exception of age, these categories have no affect on the program's operation.

Other attributes have an important affect on the program's operation. In the data file, each sentenced offender has a length of time that he or she would serve in prison were a prison term imposed, a length of time that he or she would be on probation or parole, and so on for the other correctional alternatives. To simulate changes in these variables, the data file itself must be changed by increasing or decreasing time served in prison and under community supervision for each offender. In this regard, the program is a calculator that accumulates demands on correctional resources, while many important system changes are reflected by modifications to the program's data.

However, the program goes beyond being a calculator by giving the user control over the probabilities that certain events will happen (such as the probability of a probation revocation) and the consequences when these events occur (such as the time that will be spent in prison following a probation revocation.) The program simulates the consequences of different assumptions.

One of the most important parameters that are under the user's control is the probability of a prison sentence being imposed. This probability is specific to a cell of a hypothetical sentencing guideline grid (discussed below), and thus, varies across offense seriousness and criminal record categories. When a prison term is imposed, the length of time to be served is determined from the data file (not from a sentencing grid.) The indices of a guideline grid must be specified in the data file. A similar approach is used to simulate the effects of parole release guidelines.

flows, stocks and systems resources--SSS is a stock and flow generator. It has extensive, easily accessible reporting capabilities for offenders entering and leaving a correctional alternative during a specified time period and for building stocks.

At the user's choice, one of three methods can be used to determine the stock of prisoners at the time that the simulation begins. In the first method, the user must specify when each of the prisoners who are incarcerated at the beginning of the simulation will be released on a month-by-month basis over a 61-month period. This method requires the user to specify a data file with one record per offender (weights can be used to specify multiple offenders.)

When using this first method, it is necessary to establish the stock only for prison inmates. The program infers the beginning stock for the other correctional alternatives based on the sentences observed in the input data file.

As an alternative to developing a data record for each inmate, the user can specify the stock of prisoners at the beginning of the simulation. Distributions for expected release times are used to infer when those offenders would be released.

Still a third method is to equate the beginning prison stock to the steady state that would have been generated had the sentencing simulation been run in the past. The user need not provide data when using this option.

SSS does not attribute costs to the consumption of system resources.

level of detail of the units of analysis--Ostensibly, the unit of analysis is the individual sentenced offender. The simulation proceeds, on an offender-by-offender basis, to project the demands on prison and other correctional resources five years into the future.

However, the simulation makes little use of the attributes of individual offenders. Sex, race, age, offense type, felony type and regions are used to assign reporting categories (such as prison composition by age, probation composition by offense type, and so on). Otherwise these factors do not influence the simulation.

level of detail of CJS process--The model's greatest strength is its attention to detail. As the model's developers expressed it: "Early guidelines efforts focused primarily on prison sentences... Much less attention has been given to non-imprisonment sanctions such as local jail, residential and nonresidential correctional programs and probation... The interest in comprehensive structured sentencing issues was the primary impetus for the development of the Structured Sentencing Simulation Model. The model simulates the impact of proposed sentencing policies on state prison, local jails, various correctional programs and supervision." A later version of SSS expands the model's scope to include intermediate sanctions.

The program's method for simulating changes in sentencing guidelines is cumbersome. The user must specify in the data file the time that an offender would serve were a prison term imposed. A useful program feature would be to allow the user to specify expected sentences as a parameter determined by the sentencing grid (the way that the probability of prison is determined.) Also, the user must specify in the data file the guideline grid that would apply to the offender. A useful program feature would be to allow the grid's horizontal dimension, vertical dimension, or both dimensions, to be specified as system parameters. Without these features, the user must conduct potentially extensive "front-end processing" to manipulate the data file prior to running the simulation.

This same technique was used for simulating the impact of parole guidelines. Consequently, the same comments apply.

methods for predicting growth--When appropriately weighted, the data represent the number of offenders sentenced in one year. To simulate the population of offenders two, three, four, and five years into the future, the user has two choices. The first is to assume a constant growth rate of convictions, which may be zero. The second is to assign specific multipliers to represent the relative size of the population for years two through five compared to the first year's population.

methods for estimating impact--Certain policy changes must be simulated by changing the data file. For example, the data specify the sentencing guideline cell that applies to the offender. To shift offenders among guideline categories requires changing the applicable cell specification in the data file.

The simulation of many policy changes does not require modifications to the data. Rather, the program uses a series of well designed menus to guide the user through the process of changing the model's parameters. Changes can be saved, recalled, modified and rerun.

implementation effort--One of the greatest difficulties in implementing the program is likely to be the development of the data file. Although the program requires a minimum amount of data, it is necessary to assign where the offender's case would fall in a sentencing guideline grid. To make that placement, it is necessary to collect all the data that are pertinent to the guidelines and this may be a laborious task.

A second major problem is likely to be the development of procedures (likely a computer program) for conducting the "front-end" data manipulation necessary to simulate variations of a sentencing guideline scheme. The original version of this program, as it was implemented in Minnesota, had such a front-end processor. The current version does not.

Beyond providing these data, the difficulty of developing the model is likely to be minimal. The program allows the user to provide data about the stock of prisoners. However, if these data are not readily available, they can be inferred from the sentencing data in the main data file. The model requires other data, in the form of probabilities of probation revocations, prison terms served following a revocation, and so on. However, the demands for detail made by the program are minimal and likely to be easily satisfied by extant reports of cognizant agencies.

The documentation is incomplete. A potential user would find a consulting session with the program's developers to be worthwhile.

cost of using--The authors recommend that the program be run on an AT computer with a microprocessor rated at 16 to 20 megahertz with a math coprocessor. The program requires 640K RAM. The designers recommend a 40-60 megabyte hard disk, an EGA monitor and graphics board, and a laser printer. Excluding a laser printer, this configuration can be purchased for under $4,000. Our experience running the program on a 386-based microcomputer at 25 megahertz (2,000 cases from Minnesota) was that the program executed in under one hour. The recommended AT will execute at a somewhat slower speed.

### 2.3.4 JUSTICE

JUSTICE, a microsimulation of the Texas correctional system, was written in dBase for a microcomputer by Fabelo, Gidseg and Martinez of the Texas Criminal Justice Policy Council. Our review is drawn from discussions with Fabelo and two papers: Scott and Champion (1989) and Texas Criminal Justice Policy Council (1990).

focus--JUSTICE is comprehensive of the Texas correctional system, except JUSTICE excludes offenders who were <u>sentenced</u> to jail and pretrial detainees. Offenders who are in jail awaiting a transfer to prison are included. The model appears to apply exclusively to felons.

The JUSTICE model receives as input projections of the number and types of future arrests based on statistical projections of the Texas population. These future arrests are translated into prison populations, probation populations, parole populations, offenders in jail awaiting transfer to prison, and populations in intermediate sanctions: shock probation, half-way houses, and so on.

Of the simulation models that we reviewed, JUSTICE has a unique feature. Like other simulations, JUSTICE uses historical data to determine system parameters and to simulate the mix of offenders that will be handled by the CJS. Unlike other simulation models, however, the developers of JUSTICE have integrated a data base within their simulation. Consequently, simulation parameters and the data used to drive the simulation can be constantly updated.

flows, stocks and systems resources--JUSTICE simulates the flow of offenders through several stages of the CJS. The simulation begins with arrests, which are projected from forecasts of the Texas population. Arrests become convictions and convictions become sentenced offenders. Some convicted offenders are sentenced to probation, from which they may complete their terms and exit from the system or have their terms revoked and enter the prison system. Other convicted offenders are sentenced to prison. Recognizing real-world adjustments to the State's court-mandated prison cap, the model assigns felons to jails until the prison system is able to receive them. Offenders leave prison by way of intermediate sanctions and parole. Parolees either complete their terms or they are revoked for technical violations or new crimes.

The program accumulates the stock of offenders who are on probation and parole and who are in prison, jail, or a prison alternative. The program requires the user to provide the initial level of stocks. Many of these initial stocks can be computed from the program's data base; others (such as the jail populations) must be determined from other sources.

To simulate future arrestee cohorts, the model projects the characteristics of current arrestees, as reflected in the data base. Our information is incomplete on this subject, but it appears that the characteristics of past robbers as described in the data are presumed to be characteristics of future robbers.

The program does not assign dollar costs to stocks and flows. However, it monitors system capacity for prisons and other correctional resources. For example, the program will not allow the prison usage to grow beyond a 95% cap imposed by the courts. When the cap is binding, offenders who are sentenced to prison are held in jail pending the availability of a prison bed. The program allows the user to simulate expanded prison capacity at points in time that are under the user's control. As another example, the program recognizes the limited space available at prerelease centers (halfway houses) and will not assign offenders to unavailable spaces. As a final illustration, in recognition of the use of parole as a back-end release valve, the program can be used to extend prison terms (by retarding parole terms) as prison capacity expands.

level of detail of the units of analysis--Convicted offenders (felons) are the unit of analysis. The documentation is not sufficiently detailed to allow us to determine the number of variables used in the analysis, except for offense type and sentence. According to documentation:

> The actual records of the offenders in prison and on parole, as well as the records of offenders admitted to prison, can be automatically entered in JUSTICE. The on-hand population records are used to duplicate the characteristics of the actual prison and parole populations at the beginning of the projections while the admission records are used to assign the characteristics of the most recent admissions to prison to future admissions. The offender records are automatically entered using computer data from the Texas Department of Corrections (TDC) and from the Texas Board of Pardons and Paroles (BPP)... Sample studies of the on-hand population of felons in jails awaiting transfer to prison in Dallas and Houston...are used

> to assign key characteristics to this population. Only the probation system is modeled using aggregate data. (Scott and Champion, 1989, p. 2)

level of detail of CJS process--The model encompasses stocks and flows for the primary components of the Texas corrections system: jail (offenders awaiting prison transfer), probation, parole, shock probation from prison, pre-parole halfway houses, and prison. The model does not distinguish types of prisons or level of security. The program's reporting capabilities are extensive.

method for predicting growth--Demographic projections are used to forecast the number and charge-mix of future arrests, which in turn are used to forecast future convictions. These forecasts are used to drive the model. Projections are made for 10 years.

method for estimating impacts--According to documentation, the following parameters can be changed "interactively":

- number of arrests
- probation probabilities
- incarceration probabilities
- length of stay on probation
- probation recidivism rates
- shock probation eligibility
- probabilities of release on shock probation
- probabilities of offenders being admitted to prison in specific offenses and sentence categories
- prison capacity
- capacity in half-way houses, pre-parole facilities and technical violators parole or probation facilities
- average good time earned in prison
- eligibility for parole
- eligibility for mandatory release
- eligibility for release on pre-parole
- eligibility for parole-in-absentia
- offenses classified as aggravated
- eligibility for receiving Prison Management Act credits
- Minimum calendar time served in prison by offense type
- Minimum sentences for specific offenses
- failure rate for pre-parole
- parole and mandatory supervision recidivism time
- time under parole supervision
- overall parole recommendation rate

- release rate of those recommended for parole
- triggering of special parole reviews
- utilization of the Prison Management Act
- probability of annual parole review status
- proportion of offenders revoked from parole supervision with a new sentence

Because we were unable to inspect an operational version of this model or its source code, we are unsure of how readily or how accurately these impacts can be simulated.

implementation effort--Because JUSTICE has been tailored for the Texas correctional system, we are unsure how easily the program could be transported to a different correctional system. The fact that JUSTICE operates on a microcomputer using dBase software (perhaps the most widely used data base program for microcomputers), suggests that the conceptual design of JUSTICE may be readily transferable even if the actual software is specific to Texas.

cost of using--With only limited documentation and no operational software, we have not estimated the cost of using JUSTICE. It is noteworthy, however, that JUSTICE would seem to reduce the largest cost incurred in many impact models--collecting data necessary to run the model. This advantage is derived from the data base that is part of JUSTICE's design.

### 2.3.5 Federal Sentencing Simulation (FEDSIM)

FEDSIM is a computer simulation developed jointly by the U.S. Sentencing Commission and the Federal Bureau of Prisons to project the demands on the Federal prisons of the sentencing guidelines, the Anti-Drug Abuse Act of 1986, and the repeat offender provision of the Crime Control Act of 1984. The program is described in U.S. Sentencing Commission (1987), Block and Rhodes (1989), and Gaes, Rhodes and Simon (forthcoming).

focus--FEDSIM is a microsimulation of the sentencing and punishment of convicted Federal offenders. The program reads a data file that the Sentencing Commission constructed as a probability sample (N=10,500) of roughly 40,000 offenders who were sentenced during 1985. As a first step, the program projects the expected correctional experiences of all offenders in the input data set based on the sentences imposed, good-time provisions as followed by the Bureau of Prisons, release practices as followed by the Parole Commission, and revocation practices from probation and parole as observed from historical data.

FEDSIM repeats these projections after simulating sentencing behavior under the Anti-Drug Abuse Act of 1986, then under the repeat offender provisions of the Crime Control Act of 1984, and finally under the sentencing guidelines of 1987. The program compares the demands on correctional resources resulting from preguideline sentencing practices with those anticipated to result from postguideline sentencing practices.

FEDSIM does not include petty cases heard by magistrates. Such cases rarely result in prison time. These cases do result in probation supervision, however; so in its present form, FEDSIM is not complete in its simulation of probation populations.

flows, stocks and system resources--FEDSIM is a flow model. Offenders conduct plea negotiations, after which they are sentenced and confined or placed on community supervision. Offenders who are placed on community supervision, either directly through probation or indirectly through parole, exit the system after completing their supervision period, or they return to prison following a revocation. The model allows multiple cycles of imprisonment and release following a parole revocation.

FEDSIM maintains a record of the flow of offenders, but the primary purpose of the model is to project the stock of offenders in prison

5, 10, and 15 years in the future. The program builds these stocks dynamically to show, for example, that the demands on community corrections resources will greatly expand, but not during the first two years of the guidelines, and that the demands on the prisons will double or triple, but not for 5 to 10 years.

FEDSIM can report dynamic adjustments through attention to detail. Ex post facto provisions of the Constitution prevent the guidelines, the drug laws and the repeat offender provision from being applied to offenders who committed their crimes prior to the applicability of these new laws. FEDSIM recognizes this constraint and phases the guidelines into effect based on known information about when the offenders committed their crimes relative to when they were convicted. (These dates are included as attributes of the 10,500 offenders in the input data set.) For some offenses, such as bank robbery, the difference between the offense date and conviction date are usually no more than a few months; for some other offenses, such as income tax violations, the span is typically years.

Another model detail is the distribution of prison demands over time. For example, repeat offenders typically served lengthy sentences prior to the repeat offender provisions. An enhanced sentence that extended prison time from 5 to 10 years would double the prison demands for these offenders, but the effect would not be felt for 5 years and would not be completely felt for 10 years. FEDSIM reports the distribution of prison demands over time.

To show the dynamic adjustment, FEDSIM must determine the projected release dates for the stock of offenders in prison at the beginning of the simulation. The method by which FEDSIM estimates this initial stock differs from the method used in many other simulations. First, FEDSIM assumes that offenders convicted in 1985 represent earlier cohorts in every regard except numbers. For example, suppose 1,000 robbers were convicted during 1985 and that 500 of these used firearms while 500 did not use firearms. If 900 robbers had been convicted during 1984, FEDSIM would assume that 450 used firearms and 450 did not use firearms. With this assumption, FEDSIM could estimate the characteristics of past conviction cohorts based on (1) data from the 1985 sample and (2) data about the number of people convicted of specific offenses during past years. Second, FEDSIM assumes that sentencing practices observed during 1985 reflected sentencing and correctional practices from earlier years. Building on these two assumptions, FEDSIM infers the stock of offenders under correctional supervision at the beginning of the simulation.

level of detail of the units of analysis--Convicted offenders are the unit of analysis. Because the present input data set is a probability sample of offenders who were convicted in district courts during 1985, results are weighted to reflect the entire 1985 population.

Extensive data were collected about each of the 10,500 offenders in the sample. When imposing simulated sentences and projecting the resulting demands on correctional resources, the model makes use of detailed characteristics of the offender (each prior conviction, when it occurred, what was the charge, what was the sentence, and when was the sentence completed), the offense (each charge, amount of money stolen, use of a weapon, type and extent of injury to victims, and so on), and the method of conviction (guilty plea, cooperation with the prosecution, and so on.) These data were used to simulate the time each offender would spend institutionalized and under probation and parole supervision.

level of detail of CJS process--The program simulates plea bargaining practices, sentencing practices, probation and parole experiences including multiple instances of being returned to prison for a parole violation, time spent in prison, and time spent in community corrections. Because, at the time the model was designed, case processing was uncertain under the guidelines, so the model could not be based on observed practices, FEDSIM provides for sensitivity analysis. Several alternative assumptions can be made about plea bargaining, about compliance with the guidelines, and other features of case processing.

method for predicting growth--FEDSIM uses aggregate data from the Administrative Office of the U.S. Courts to project future caseloads. The Administrative Office provided 20 years of data about the number of convictions for 20 offense categories (robbery, embezzlement, and so on.) The designers of FEDSIM made several projections based on these data. Two projections were used as upper and lower limits.

The lower limit was straight forward. The model assumes that criminal cases will grow at a 1 percent compound rate. Because criminal convictions have always grown at least by 1 percent rate for a sustained five year period, the model's developers claimed that a 1 percent growth rate was a reasonable lower limit on growth.

The higher limit was more complicated. Basically, it assured that growth during the projected fifteen year period corresponded to the observed growth during the prior fifteen years (a period of high growth in the Federal criminal caseload.)

method for estimating impact--Policy changes are simulated by making changes to the computer code, recompiling the code, and executing the program with these changes. A professional programmer with a thorough understanding of the program's elements is required to modify and run the program.

implementation costs--FEDSIM was written in the "C" programming language for a microcomputer. Changes to the program require the assistance of a professional programmer. FEDSIM reports output in the form of a spreadsheet, which can be imported into Lotus 1-2-3. Thus, FEDSIM's reporting capability is accessible to a researcher who is conversant with spreadsheets.

Because FEDSIM was written to simulate expected plea bargaining, sentencing and correctional practices prior to implementation of the guidelines, the program would be most useful were it rewritten to incorporate knowledge about actual practices under the guidelines. Nevertheless, FEDSIM--in its current form--is still sufficiently accurate that both the Sentencing Commission and the Bureau of Prisons rely on the model to make prison projections.

cost of operation--A complete run of FEDSIM can be completed in under one hour on a 386 microcomputer; runs that are limited to specific offense types can be completed in less time. Because most simulations require the modification of source code, a professional programmer is required.

## 3. Potential For of a Federal Impact Model

Several lessons can be learned from reviewing extant computer impact models of the criminal justice system. We recount some lessons here, and use them to recommend the form of a Federal impact model.

### 3.1 Computer simulations are special purpose programs

Computer simulations have been designed to solve specific problems. As a corollary, no existing computer simulation model can solve every problem that might be encompassed by the rubric "justice impact." A first step in impact modeling is to delimit the problem domain.

What impacts should a Federal justice impact model address? The National Drug Control Strategy provides some guidance[5]:

> The creation of criminal justice simulation models by the Department of Justice will permit us to estimate the impact of policy changes on various parts of the existing system. Models will indicate, for example, the likely consequences that more drug-related arrests (or fewer probationers or longer sentences) will have on the courts, the jails and prisons, and the probation and treatment systems. Policy makers at the national, state, and local levels will use these models to anticipate the need for shifts in resources and help them plan a more coherent and productive criminal justice system.

Thus, the desired model must handle stocks as well as flows, must assess the dynamics of change, and must serve to analyze components of the CJS as well as the CJS as a whole. It is unlikely that any one model could satisfy the modeling needs implied by this broad agenda.

Above all, then, a Federal justice impact model must be flexible and adaptable--for the model will surely be called upon to evolve and address a wide range of situations, many of which cannot be anticipated at this time.

## 3.2 Extant models do not meet the "flexibility" criterion

We have reviewed some exceptionally well-designed computer impact models, many of which have proven their utility in public policy applications. It is no discredit to the models' designers to conclude that none of these models

---

[5] Additional guidance comes from the Anti-Drug Abuse Act of 1988, H. 11215, sec. 9201, which instructs the Comptroller General:

> (1) to determine the impact of additional resources to certain components of the Federal criminal justice system on other components of the system and of enhanced or new Federal criminal penalties or laws on the agencies and offices of the Department of Justice, the Federal courts, and other components of the Federal criminal justice system; and
>
> (2) use the data derived from the impact analysis to develop a model that can be applied by Congress and Federal agencies and departments to help determine appropriate staff and budget responses in order to maintain balance in the Federal criminal justice system and effectively implement changes in resources, laws or penalties.

meets the flexibility criterion. After all, none were designed with that criterion as a goal.

As we mentioned above, purely statistical models do not seem to be a satisfactory solution. They fail to account for changes that are the very purposes of policy interventions.

Disaggregated models (JUSSIM, IMPACT, CCPS) have played an inarguably important role in impact analysis. But these models purposefully minimize the details about units of analysis, necessarily restricting the policy interventions that can be simulated. Furthermore, disaggregated models--which concentrate on steady-state solutions--lack the richness of process detail to describe interventions that unfold over time. Designers have increased the accessibility of disaggregated models through menu driven systems and graphics, but at a cost. The user must confine his investigations to the range of queries for which the model is programmed. As wide as this range is for many CJS-related queries, it is too narrow to satisfy the flexibility criterion.

Microsimulations seem to have an advantage over disaggregated models, at least for present purposes. Several microsimulations have demonstrated and continue to demonstrate their utility in the policy arena: NCCD, SSS, JUSTICE and FEDSIM. Because these simulations incorporate considerable detail about the units of analysis, they have at least the potential to address policy changes affecting defendants with specific characteristics (drug sellers who deal 100 grams of cocaine, repeat bank robbers who use a weapon, and so on). However, like their disaggregated counterparts, microsimulations have been designed for specific purposes. Because microsimulations tend to focus on narrow aspects of justice administration (typically corrections), attempts at modification may be so contrived as to invite the development of an entirely new model.

We can illustrate these limitations. FEDSIM was designed to address the impact on prisons of three policy interventions: the career criminal provisions of the Crime Control Act of 1984, the enhanced criminal penalties prescribed by the Anti-drug Abuse Act of 1986, and Federal sentencing guidelines as implemented in 1987. Simulation of these three policy interventions requires detailed knowledge of the defendant's criminal record (including the number and type of prior convictions) and detailed knowledge of his crime (including the amount of drugs sold, where they were marketed, and to whom.) Simulation also requires detail of the criminal justice process--plea bargaining, departures from the guidelines, parole revocation practices, and so on. With the exception of FEDSIM, the computer models that we reviewed lacked adequate details regarding the unit of analysis and CJS process to simulate these policy interventions; although FEDSIM can simulate these policies, FEDSIM is structured to simulate only these and similar policy interventions, and hence, violates the flexibility criterion.[6]

It appears that no extant computer simulation model meets the requirements stipulated in the National Drug Control Strategy. BJS should coordinate the development of a Federal impact model that departs in concept and design from extant computer simulations. The model should be a microsimulation that is inexpensive to reprogram and that maximizes access by users who are not programmers. The BJS model should be a complement to, not a substitute for, models that are used by other Federal authorities.

## 3.3 Important design lessons

Although extant simulation models fail to meet the flexibility criterion, they provide guidance when judging what model features are desirable. The ideal model would possess the flexibility of CCPS, the graphics capability of SIMFLOW/JUSSIM, and the ease of use of IMPACT. The model must deal with the data at hand.

---

[6] The limited scope of FEDSIM is not a permanent problem. The Bureau of Prisons has continued to develop this model. Future adaptations may overcome a number of FEDSIM's limitations.

### 3.3.1 Data limitation and uses

Data availability is the first consideration when designing an impact model. Disaggregated models minimize data needs and therefore may be preferable to their data-hungry microsimulation counterparts, especially in state and local courts, where detailed data are unavailable or too expensive to collect.

Minimizing data needs is less of a consideration on the Federal level. Since 1980, the Bureau of Justice Statistics has supported the development of an integrated data base of Federal justice sources. Although this integrated file is not without gaps (especially at the investigation stage), it provides the working material for a microsimulation. Additionally, there exist special purpose files (we have mentioned the Sentencing Commission's sample of 10,500 offenders) that can be used to augment the BJS data, and Federal agencies are designing and developing future reporting systems. A model that was developed without anticipating future data availability could become dated before it was implemented. Our conclusion is that a Federal impact model should be able to make full use of available and future data sets.

No existing data file is likely to contain all the information that would be desirable for a microsimulation. Many applications would likely require some special purpose, narrowly focused data collection. By building on the BJS integrated data and on other extant data files, sampling frames can be developed and data collection costs can be minimized.

We are impressed with the Texas simulation model, JUSTICE, which integrates a simulation program with a dynamic data base that changes as information is updated. While the JUSTICE data base is integrated into the computer simulation, any design that affords access to a data base, so that simulation model parameters can be computed readily, would serve the same purpose. Given the availability of the BJS Federal data, the bridge between data and the computer simulation should be a design priority of a Federal impact model.

### 3.3.2 Flexibility

A Federal impact model should be flexible, but flexibility comes at a cost. An impressive feature of IMPACT, SSS and the microcomputer versions of JUSSIM are their exceptional user interfaces.[7] IMPACT and SSS use pull-down menus; one microcomputer version of JUSSIM uses menus and a graphics display. Output is printed both on the computer's screen and in a file, from which it can be incorporated into spreadsheets and word processing documents. A user does not need to understand the model's mechanics to "make the model work."

These impressive user interfaces are possible because the programs limit the user's choices, forcing the user into a preconceived problem solving structure. These structure "fit" for the problems anticipated by the programs' designers; they do not necessarily fit for the problems anticipated by the National Drug Control Strategy. Hence, a basic conflict exists between user interface and flexibility.

Our own view is that there is room for compromise between flexibility and user interface. We are impressed with the program CCPS, which was written for a microcomputer using Lotus 1-2-3. This program allows the user to modify the simulation's details, yet the program can be run with a minimum of effort: the user need only provide the correct parameters where requested by the program. There are several advantages to programming an impact model with a spreadsheet, such as Lotus 1-2-3.

---

[7] By a "user interface," we mean the way that a user views and operates the computer simulation. As an analogy, a television viewer interfaces with his television by using an on/off switch, a volume control, a channel selector, and other instruments. A computer keyboard, a mouse, and other physical devices are used to interface with the computer; menus, graphics, and other software support assist the user's access to the physical devices.

### 3.3.3 Using a spreadsheet

If one were forced to characterize CJS computer simulation models in no more than a few words, the best description might be to call them complex adding machines. All CJS simulations receive input in the form of arrestees, sentenced offenders, or other units of analysis. These inputs are converted into flows and stocks. Based on these flows and stocks, the program proceeds to anticipate the demands that will be made on the CJS system and then to accumulate those demands for relevant categories (prisons, probation, and so on) and to dissipate them over time. It is in this regard that the simulation models are complex calculators; by necessity, the programs' designers have implicitly written computer code implementing a spreadsheet.

Our suggestion is to <u>start</u> with the spreadsheet and to build the other parts of the simulation around the spreadsheet structure. One advantage to this approach is that spreadsheets can be interfaced (that is, connected implicitly or explicitly) with data base programs.[8] At this time, Lotus 1-2-3 can interface with dBase, and future routines for matching Lotus with other data base systems are promised. It is possible to develop both the data base and the calculator component of the simulation with minimal programming.

Spreadsheets afford modular[9] programming. By modular programming, we

---

[8] The connection between the simulation and the data base is "explicit" when the simulation model and the data base are part of the same computer program. It is "implicit" when one program produces output that is the input of a second program. When the data base and the simulation are connected implicitly, software development may be facilitated, because development of the data base and the simulation can proceed independently.

[9] Modular programming is a technical term meaning (crudely) that sections of the computer program operate independently of other parts of the rest of the program. Hence, a module can be extracted, modified, and replaced without the rest of the program being effected.

Spreadsheets can be written using modular programming concepts. Additionally, modern spreadsheet programs allow multiple spreadsheets to be linked, allow a single spreadsheet to have multiple levels, allow macros and subroutines to be written in an external library, allow special add-in mean that a basic simulation structure can be augmented by adding components that represent specific aspects of the CJS, such as pretrial detention or drug treatment services. When these components play no significant role in the policy impact being simulated, they can be "turned off" and not affect the simulation. When new components are desired, they can be developed and added without major modifications to the basic computer program. By using modular programming techniques, a microsimulation can grow to accommodate new needs and yet not be encumbered by irrelevant components.

Spreadsheets do not hinder the development of user interfaces that will facilitate a policy analyst's access to the model. As we mentioned earlier, we are impressed with the user interface provided by IMPACT and SSS.

Spreadsheets have built-in graphics capabilities which can be accessed by a user through a pull-down menu, printed, or incorporated into a word processor document. With a minimum of programming, stocks and flows can be shown in a variety of charts that will help the user understand his or her results and will assist the user to communicate his or her findings to others.

Finally, spreadsheets can be understood by researchers and analysts with minimal programming experience. Policy specialists can work meaningfully with program developers to produce models that best meet the analysts' needs.

### 3.3.4 Disadvantages of using a spreadsheet

A spreadsheet program is the best way to develop a Federal justice impact model, because the spreadsheet and data base accompaniment already provide the simulation's basic structure. There are some disadvantages to using a spreadsheet, however, although these disadvantages are not determinative.

---

products that enhance the spreadsheets capabilities, and allow an interface with a data base or record manager. Spreadsheets now compete with "programming languages."

A principal disadvantage of using a spreadsheet and data base accompaniment is that neither the spreadsheet nor the data base provide all the programming tools desirable when developing a simulation. As examples, a typical spreadsheet has only rudimentary implementations of computing loops and complex branching logic, two functions of basic importance to most computer programs. Additionally, spreadsheets are general purpose programs, so the computing time required to perform a simulation would be greater for a spreadsheet than it would be for a program that was designed specifically as a CJS simulation. The degree to which these disadvantages are handicaps is unknown at this time, but it is prudent to anticipate problems.

Although spreadsheet technology has limitations, it is not unreasonable to presume that complex data manipulation can be done outside the spreadsheet, using a computer program to serve as an interface between the data base and the spreadsheet. This intermediate, interface program could be written in a powerful general purpose computer language such as "C". The ability to add an intermediate program, using highly sophisticated programming tools, effectively removes any limitations inherent in building the simulation around a spreadsheet and data base.

## 3.4 Recommendations

The models that we reviewed for this paper are for the most part complex programming exercises with computer code written in "C", FORTRAN, Q-GERT and PL/1. Using a spreadsheet may be seen as a step backward in terms of technical sophistication. However, as we have emphasized throughout this review, a simulation model should not be judged by its level of technical sophistication, but rather, by its utility. At this time, nobody can stipulate the technical specifications for a Federal impact model, because nobody can identify with certainty the extent of questions that the modelers will be forced to address. Our solution is to recommend versatility and flexibility, which can best be assured by writing the impact model in a language that is accessible to a range of researchers and policy analysts who lack refined skills in a narrow computer language.

A Federal justice impact model will evolve over time as policy issues become better defined, as data become more readily available, and as Federal agencies become increasingly vested in using results from simulations. We should not invest heavily in writing powerful computer programs that will be scuttled shortly by evolutionary changes. Rather, we should write simulations that, like a disposable razor, can be discarded as sharper models are required.

Of course, using a spreadsheet during this developmental stage does not preclude the eventual adoption of a different approach as needs become better defined. Rather, adopting a spreadsheet approach during the developmental stage will remove "programming technicalities" as an impediment to addressing the many other problems of model development, many of which have nothing to do with designing, writing, debugging and running computer programs.

CJS impact models cannot be written by technicians alone; they must be developed in concert with CJS practitioners and policy planners. A daunting challenge faced by BJS when developing and using Federal impact models is to gain the early cooperation and active assistance of Federal criminal justice agencies. Toward this end, an operational if somewhat unsophisticated CJS impact model is to be preferred over a sophisticated but nonoperational one. Consequently, we recommend an approaches that shortens the development period.

To summarize, we recommend that BJS launch the development of an impact model that draws heavily on existing data. We recommend that a premium be placed on the versatility of this model. We recommend that program development be accelerated by using spreadsheet and data base software available for microcomputers. We recommend that the task of developing an impact model be expanded beyond the narrow technical programming issues involved to encompass the equally important needs to work with multiple Federal agencies whose cooperation and assistance are essential to the success of any policy analysis tool.

## References


Anonymous (1980) "Structural Models: A Primer for the Courts and an Illustration" in Forecasting the Impact of Legislation on Courts, Boyum and Krislov, ed. National Academy Press: Washington, D.C.:109-124.

Bales, W. (1989) personal communication and unpublished documents pertaining to Florida's version of the NCCD model.

Block, C., Miller, L. and Hudson, D. (1983) Manual for the Pattern Description of Time Series Illinois Criminal Justice Information Authority.

Block, M. and Rhodes, W. (1989) "Forecasting the Impact of the Federal Sentencing Guidelines," Behavioral Sciences and the Law, 7(1).

Blumstein, A., Cohen, J. and Miller, H. (1980) Demographically Disaggregated Projections Urban Systems Institute, Carnegie-Mellon University.

Blumstein, A. (1983) "The Impact of Changes in Sentencing Policy on Prison Populations" in A. Blumstein, J. Cohen, S. Martin and M. Tonry, ed. Research on Sentencing: The Search for Reform National Academy Press: Washington, D.C.

Boyum, K. and Krislov, S. (1980) Forecasting the Impact of Legislation on Courts National Academy Press: Washington, D.C.

Center for Urban Analysis (1989) JUSSIM User Interface: User Guide & Tutorial Office of the County Executive, Santa Clara County, San Jose, California.

Celeste, J. and Douglass, K. (1980) A Study of Justice Impact Analysis Office for Improvements in the Administration of Justice, FJRP-80/003, U.S. Department of Justice, Washington, D.C.

Chaiken, J., Crabill, T., Holliday, L., Jaquette, D., Lawless, M. and Quade, E. (1975) Criminal Justice Models: An Overview Rand: Santa Monica.

Cohen, J., Fields, K., Lettre, M., Stafford, R. and Walker, C. (1973) "Implementation of the JUSSIM Model in a Criminal Justice Planning Agency" Journal of Research in Crime and Delinquency 10(2):117-131.

Cushman, R. (1989) untitled paper presented at the Annual Meeting of the American Society of Criminology, Reno, Nevada.

Dall, W. and Marx, E. (1989) IMPACT User's Guide (Version 2.0) Criminal Justice Statistics Association, Washington, D.C.

Fox, J. (1978) Forecasting Crime Data Lexington, Mass: Lexington Books.

Institute for Law and Justice (1990) FLOWSIM: Interactive Criminal Justice System Case Flow Simulator--User Manual Draft of chapter 1.

Institute for Law and Justice (1990) FLOWSIM computer program.

Institute for Rational Public Policy, "Design Plan for a Postprison Extension to the Structured Sentencing Simulation (SSS)" undated manuscript.

Jarvis, J. (1989) "Evaluating Criminal Justice Policies: A Simulation Model Approach" paper presented at the 41st Annual Meeting of the American Society of Criminology in Reno, Nevada.

Johnson, J., Eidson, J., Angell, J. and McCleary, R. (1983) Bibliography on Crime Forecasting and Related Topics Justice Center, School of Justice, University of Alaska, Anchorage.

Kramer, J. and Lubitz, R. (1985) "Pennsylvania's Sentencing Reform: The Impact of Commission Established Guidelines" Crime and Delinquency 31(4).

NIC (1988) "Community Corrections Planning Simulation Version 1.0" unpublished documentation to computer program.

NCCD (1989) Computer program written for the Virginia version of NCCD's computer simulation.

NCCD (1990) The PROPHET: The NCCD's Interactive Microcomputer Simulation System User's Manual & Quick Reference Guide unpublished draft.

Parent, D. (1988) Structuring Criminal Sentences: The Evolution of Minnesota's Sentencing Guidelines Butterworth: MA.

Pannell, W. (1978) "Population Projection Methodology with Emphasis on Simulation Techniques" paper presented at Offender Based State Corrections Information Systems III Seminar, Denver, Colorado.

Renshaw, B., Goldman, C., Braybook, W. and Mitchell, E. (1972) "PHILJIM: The Philadelphia Justice Improvement Model, a general computer model for simulation of the total criminal justice system of the State of Alaska and the City of Philadelphia" paper presented at the International Symposium in Criminal Justice Information and Statistical Systems, New Orleans.

Scott, R. and Champion, R. (1989) "JUSTICE: A simulation model for the Texas adult correctional system" unpublished paper, Texas Criminal Justice Policy Council, Austin, Texas.

Silbert, J. (1989) "The Dade Justice Improvement Model (DADEJIM)" unpublished paper.

Structured Sentencing Simulation Model: User's Manual April 4, 1988.

Structured Sentencing Simulation (1989) Computer program dated 2/19/1989.