1998

Directed Search in K-System Reconstruction.

Christopher W. Branton
Louisiana State University and Agricultural & Mechanical College

INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each original is also photographed in one exposure and is included in reduced form at the back of the book.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

UMI

A Bell & Howell Information Company
300 North Zeeb Road, Ann Arbor MI 48106-1346 USA
313/761-4700 800/521-0600

DIRECTED SEARCH IN K-SYSTEM RECONSTRUCTION

A Dissertation

Submitted to the Graduate Faculty of the
Louisiana State University and
Agricultural and Mechanical College
in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

in

The Department of Computer Science

by
Christopher W. Branton
B.S., Louisiana State University, 1992
December 1998

Table of Contents

List of Tables

List of Figures

Abstract

K-systems analysis is a factor analysis technique created through the generalization of key reconstructability analysis definitions and algorithms. The method is applied to functions on systems of discrete variables to discover a set of factors which can explain the bulk of the function's variation from the mean. K-systems analysis uses principles of information theory to reveal interactions which are often masked by the assumptions implicit in traditional methods. The method has been used successfully to analyze systems in several disciplines.

Despite the success of k-systems analysis, obstacles to the creation of a mature methodology still exist. Some issues and open questions are examined, and a requirement for creating disjoint subsets of equations for calculating the unbiased reconstruction is confirmed, at least in the context of the greedy reconstructability algorithm. There is also a need for a framework to compare reconstructions. One approach for deriving comparison measures is suggested, based on the similarity between k-systems and the concept of a fitness landscape.

One of the most serious obstacles to the generalized use of k-systems analysis is the exponential growth of system size as the number of variables and the values they assume increases. Searching the entire substate set for candidate factors limits the size of systems which can be effectively reconstructed. Methods exist which limit the search to a fraction of the substate space, but often lead to less compact reconstructions.

An algorithm is presented which performs a search of the smaller state space to choose factors to use as starting points for a directed search of the substate space. Complexity analysis and experimental evidence indicate that the directed search

technique provides a notable reduction in computation for the search process, while still providing a compact reconstruction. Combining directed search with state sampling techniques should further extend this capability.

In addition to the directed search algorithm, a technique is proposed which can significantly reduce the computation required to update substate function values. This technique is based on a substate labeling scheme which imposes a total ordering on the substate set.

Chapter 1. Introduction

The scientific method is based on the observation and analysis of measurable properties of real or theoretical objects of interest. In most cases, the investigation will involve more than one of these properties, and any interaction among them may be at least as important to the analysis as the behavior of the individual attributes. In such cases, it is often helpful to combine the properties into an information model known as a system and analyze the behavior of the system as a whole. The interrelated set of problems concerned with "determining a system on an object of investigation which is an adequate model of relevant phenomena associated with the object" is known as systems modeling [CAVA81a].

For any but the simplest systems, this commonly involves viewing the overall system in terms of subsystems. For example, probability distributions involving large numbers of variables are often constructed by combining marginal distributions. The problems of how such subsystems combine to form the behaviors associated with the overall system, and how an overall system can be represented by interacting subsystems, form the core of reconstructability analysis (RA).

An important derivative of reconstructability analysis is k-systems analysis (KSA), a data analysis technique created through the generalization of key reconstructability analysis definitions and algorithms. K-systems analysis techniques use principles of reconstructability analysis to reveal structure in data which is often masked by the assumptions implicit in traditional data analysis methodologies.

The k-system techniques are built around two general algorithms which were developed in the mid-1980's to identify and reconstruct systems using digital computers

[JONE85a][JONE85b]. While the bulk of related research has involved applying these techniques to solve problems in areas from industrial engineering to evolutionary biology, a small but active research effort has remained focused on increasing the generality, efficiency and usefulness of these algorithms. The primary goal of this research is to further that effort.

Though noteworthy results have already been produced in several disciplines using KSA techniques, the methodology is still not fully developed. Several problems currently elude general solutions, and questions remain unanswered which largely limit the use of this technique to a few researchers modeling relatively small systems. An examination of some of these issues is given in Chapter 3.

There is a set of interrelated problems related to constructing an appropriate system model from an arbitrary collection of data points. Problems involving inconsistencies, data scattering, state contradictions and missing data values must be satisfactorily resolved before reconstruction can even be attempted. Attempts to resolve one of these problems can often exacerbate one or more others. While techniques exist to deal with most of these issues, a general framework for this process does not yet exist, and a comprehensive measure of the error which is inevitably introduced has not been defined.

There are also fairly severe limitations on the size of systems that can be analyzed using this technique due to the computational intensity of the reconstructability algorithm. System state and substate sets grow exponentially as the numbers of variables and values increase, and current k-system techniques require

iterating each of these sets at least once. This requirement restricts the use of these techniques to relatively small systems.

Efforts to reduce the computation necessary to reconstruct a system inevitably lead to questions concerning which components of the algorithm can be substantially changed without harm, and which must remain essentially unchanged to retain the power of the method. One such question has concerned the necessity of partitioning the set of substate equations into disjoint subsets. Analysis is presented in Chapter 3 which indicates that this partitioning is essential to obtaining a correct reconstruction.

Important questions also exist concerning the results generated using KSA. Different models may be generated for a single system when different optimizations are applied to the algorithms. While measures exist to evaluate how well a particular model captures system behavior, there is currently no framework defined to explore the difference between two models which reproduce system behavior equally well.

A commonly employed intuition is that a reconstruction involving fewer substates is superior to one requiring more substates. This intuition can be quantified by dividing the overall reconstruction closeness by the number of factors required for the reconstruction, yielding an average contribution per factor.

Researchers in genetic algorithms, machine learning and complex systems research make heavy use of a concept known as a fitness landscape. The similarity between k-systems and fitness landscapes suggests at least the possibility that these areas of research may have measures defined which would be useful for comparing reconstructions.

The exponential growth of system size has restricted the use of these techniques to systems with only a few variables. Parallel algorithms have been developed to allow analysis of larger systems [ILES95]. But since the specialized hardware necessary to implement these algorithms is still not widely available, other optimizations should be examined.

In particular, a technique which can limit the search for candidate substates while maintaining a compact reconstruction would allow k-systems analysis of larger systems than is now possible for a given computer platform. In addition, this technique may provide a structured analysis of the data present, while avoiding the imposition of a standard model, leading to a greater understanding of the results produced by the reconstructability algorithm. Examination of one proposed algorithm for directed substate search is the topic of Chapter 4.

The directed search technique alone can modestly expand the class of systems which can be reconstructed on a given computer platform. However, the true potential of the technique has yet to be realized. If techniques for system approximation such as state sampling and substate estimation were combined with the directed search approach, a much larger class of systems could be analyzed using k-systems techniques.

Chapter 2. Overview

The study of history is often advocated as a way of supplying context for current events. K-systems analysis evolved from efforts to address basic questions in a field of general systems theory known as reconstructability analysis, and a comprehensive understanding of the k-system framework requires a study of some central concepts and definitions from the RA literature.

K-systems analysis and reconstructability analysis are still tightly coupled subjects, and the two terms may be used interchangeably in many contexts. For the most part, "reconstructability analysis" will be used here to describe concepts which are shared among the two fields, or unique to RA. The term "k-system" will be used to describe those concepts which are significantly different from their RA counterparts, or which are unique to k-systems. An overview of the principle differences between the two fields is presented in Section 2.4.3.

2.1 Reconstructability Analysis

Reconstructability analysis has been defined as "the process of investigating the possibilities of reconstructing desirable properties of overall systems from the knowledge of the corresponding properties of their various subsystems" [CAVA81b]. Significant early work in the area of reconstructability analysis was first brought together in a special issue of the *International Journal of General Systems* [CAVA81a] dedicated to RA. In particular, a series of papers by Cavallo and Klir integrated the questions, definitions and techniques of reconstructability analysis into a cohesive problem space. Cavallo and Klir described reconstructability analysis as the study of two problems associated with modeling systems as sets of coupled subsystems

[CAVA81b]. The first is the identification problem, which involves identifying the properties of an unknown overall system from known properties of its subsystems. If the overall system is known, the problem becomes one of identifying which subsystems are sufficient to reconstruct the properties of the overall system to a desired level of accuracy. This is known as the reconstructability problem.

2.1.1 Behavior Systems

In many cases, the relevant behavior of a system can be captured by a single function whose domain is the set of states and whose range is a subset of the real numbers. The corresponding model which incorporates this function is known as a behavior system, designated B. The function representing the system behavior is called the behavior function.

As an example, a biological study of the effect of certain genetic variations might associate a particular fitness value with each combination of genes. Each gene in this model would have a finite number of values it could take. Each possible combination of values for the genes under study would define a state of the system. The fitness values for the genetic combinations would define a behavior function for the genetic system.

In such an information model, the properties are represented by variables whose values represent some state of the object being modeled. Each variable of the system may assume a finite number of values. The set of variables for a given system is represented by the set V, or $\{v_i\}$. Each unique combination of variable values defines a system state, designated α.

Formally, a behavior system B is defined by:

$$B = (V, \mathcal{V}, s, A, Q, f)$$

where

- $V = \{v_i \mid i \in N_n\}$ is a set of variables
- $\mathcal{V} = \{V_j \mid j \in N_m, m \leq n\}$ is a family of state sets
- $s : V \rightarrow \mathcal{V}$ is an **onto** assignment function by which one state set from $\mathcal{V}$ is assigned to each variable in V
- $A = \{s(v_1) \times s(v_2) \times s(v_3) \times \ldots \times s(v_n)\}$ is the set of all potential aggregate states
- Q *is* a set of real numbers which includes zero
- $f : A \rightarrow Q$ is a function, commonly referred to as a behavior function [CAVA81b]

Until reconstructability techniques were shown to be applicable to a wide range of general functions [JONE85], the behavior function was normally a selection function, probability distribution function, or fuzzy set membership function [CAVA 82b].

2.1.2 Subsystems and Structure Systems

Reconstructability analysis is primarily a study of how the behavior of large systems of many variables can be explained in terms of the interaction of simpler systems of fewer variables. The smaller systems are known in RA as subsystems, since their variable sets are subsets of the variables of the larger system. The behavior system framework is intended to account for both systems and their component subsystems.

A behavior system may be viewed as a the overall system of study, or may be seen as a subsystem of a larger overall system. Given a behavior system B, and another system ${}^0B = ({}^0V, {}^0\mathcal{V}, {}^0s, {}^0A, {}^0Q, {}^0f)$, ${}^0B$ is a subsystem of B if and only if:

* ${}^0V \subset V$
* ${}^0\mathcal{V} \subseteq \mathcal{V}$ such that ${}^0s$ is onto
* ${}^0s : {}^0V \rightarrow {}^0\mathcal{V}$ such that ${}^0s(v_i) = s(v_i)$ for each $v_i \in {}^0V$
* ${}^0A = \underset{v_i \in {}^0V}{{}^0s \times}(v_i)$
* ${}^0Q = Q$
* ${}^0f = [f \downarrow {}^0V]$

This definition formalizes the idea that every non-empty proper subset of variables of an overall system B identifies a single subsystem of B [CAVA81b].

A set of behavior systems such as $S = \{{}^kB = ({}^kV, {}^k\mathcal{V}, {}^ks, {}^kA, {}^kQ, {}^kf) \mid k \in N_q\}$ is referred to as a structure system. The individual ${}^kB$ in S are known as elements of the structure system [CAVA81b].

Structure systems may be used to create a refinement lattice, a conceptual construct which places a partial ordering on models based on whether they are refinements of other models in the lattice [CAVA82]. Given the set M of all models over the variable set V, a model $X \in M$ is a refinement of model $Y \in M$ iff for every $V_x \in X$ there exists a $V_y \in Y$ such that $V_x \subseteq V_y$ [PITT90]. The lattice is intended as a tool to help researchers manage the conceptual and computational complexity of investigating large systems using RA techniques.

2.1.3 Substates

While the structure system clearly defines the relationship between systems and subsystems, there is a need to formalize the relationship between the states of an overall system and the states of its subsystems. The states of a subsystem are defined as substates of the overall system.

Recall from the definition above that the variable set of a subsystem is a subset of the overall system's variables. A given substate β is a substate of a state α if every variable of β takes the same value as the corresponding variable in α. This relationship is formalized by the following definition:

> "If $\alpha = (\alpha_i | \, i \in N_n) \in A$ is an aggregate state of a behavior system, then $\beta = (\beta_j | j \in X, X \subset N_n)$ is called a *substate* of α (or α is a *superstate* of β) if and only if $\beta_j = \alpha_j$ for all $j \in X$. The notation $\beta \succ \alpha$ is commonly used to denote that β is a substate of α" [CAVA81b].

Substates are primarily used in RA to define the projection $[f \downarrow V]$ which is required to define a subsystem. This projection for a substate β is defined to be $g(\{f(\alpha) | \alpha \succ \beta\})$, where the nature of function g depends on the nature of the system. For a probabilistic system, the substate function is the sum of the corresponding state function values [CAVA81b]. A list of the eight states and eighteen substates for a system of three binary variables is shown in Table 1 below.

2.2 The Identification Problem

As the systems under investigation become more complex, it becomes unfeasible to gather information on all of their properties simultaneously. Even if each variable takes a finite number of values, the possible number of system states

Table 1. States and Substates for Binary Variables

State α	Substates $\beta \prec \alpha$
000	$^{1}(0), ^{2}(0), ^{3}(0), ^{12}(00), ^{13}(00), ^{23}(00)$
001	$^{1}(0), ^{2}(0), ^{3}(1), ^{12}(00), ^{13}(01), ^{23}(01)$
010	$^{1}(0), ^{2}(1), ^{3}(0), ^{12}(01), ^{13}(00), ^{23}(10)$
011	$^{1}(0), ^{2}(1), ^{3}(1), ^{12}(01), ^{13}(01), ^{23}(11)$
100	$^{1}(1), ^{2}(0), ^{3}(0), ^{12}(10), ^{13}(10), ^{23}(00)$
101	$^{1}(1), ^{2}(0), ^{3}(1), ^{12}(10), ^{13}(11), ^{23}(01)$
110	$^{1}(1), ^{2}(1), ^{3}(0), ^{12}(11), ^{13}(10), ^{23}(10)$
111	$^{1}(1), ^{2}(1), ^{3}(1), ^{12}(11), ^{13}(11), ^{23}(11)$

expands exponentially as the number of variables increases. If there are n variables and each variable takes k values, there will be k^n possible system states. It is not practical to gather all possible state information experimentally for most systems of even modest complexity, and some complex real life systems will not have reached all possible system states in their history, even if someone were there to collect the data.

This is a central motivating factor in the wish to create system models from interrelated subsystems or partial information. In the RA framework, these subsystems are combined to form a structure system. The problem of identifying overall systems which can generate the information in these subsystems is known as the identification problem in RA [CAVA81b]. An example of the identification problem is shown in Table 2.

The two systems shown here can each individually define a behavior system. Variable 2 represents a property present in both systems. The identification problem in this case is to determine what overall systems can be identified that are consistent with the information in the two systems shown in Table 2.

Table 2. Example Structure System of Two Subsystems

System 1			System 2		
Variable 1	Variable 2	$f(\cdot)$	Variable 2	Variable 3	$f(\cdot)$
0	0	0.25	0	0	0.37
0	1	0.18	0	1	0.18
1	0	0.20	1	0	0.09
1	1	0.37	1	1	0.36

2.2.1 Reconstruction Families

In the general case, more than one overall system may exist which could produce the information contained in the subsystems. For a given structure system S, the set of all overall systems that are compatible with S is called the reconstruction family of S [CAVA82].

Identifying the entire reconstruction family can provide useful information concerning the uncertainty present in the overall system. Cavallo and Klir give a procedure for obtaining the reconstruction family for a system with a probabilistic behavior function, based on the fact that substate values are simply marginal probabilities. For a given overall system B, and a subsystem ${}^{k}B$, then ${}^{k}f$ must satisfy

$$ {}^{k}f(\beta) = \sum_{\alpha \succ \beta} f(\alpha). $$

The ${}^{k}B$ thus form a set of linear equations. Each non-zero solution to such a set of equations defines a probabilistic behavior function which uniquely represents a member of the reconstruction family [CAVA81b]. Jones improved the method to obtain a more efficient form of the matrix equations, greatly reducing the computation necessary for determining the reconstruction family of a structure system [JONE82].

2.2.2 The Unbiased Reconstruction

While identifying the entire reconstruction family can provide useful information about a structure system, evaluation of a reconstruction hypothesis ideally leads to the identification of a single member of the reconstruction family which in some way best utilizes the information present in the subsystems. Selecting a single overall system from the reconstruction family requires some justifying assumptions.

Cavallo and Klir make strong arguments that the best, if not only, solution for a probabilistic system is the solution which maximizes the information entropy present in the overall system, known as the unbiased reconstruction [CAVA81b]. Information entropy is a measure of uncertainty in probability distributions, defined by Shannon as the quantity

$$H = -K \sum_{i=1}^{n} p_i \log p_i ,$$

where p_i is the probability associated with event i, and K is a positive constant associated with the choice of a unit of measure [SHAN48]. In the behavior system framework, the p_i correspond to the behavior function values for state i, the constant K is normally 1, and the logarithm base is 2.

Cavallo and Klir use three arguments to justify this choice:

1) The maximum entropy distribution is the only unbiased distribution. The maximum entropy distribution takes into account all of the constraints present in the data, but does not introduce any other constraints.
2) The maximum entropy distribution is the most likely distribution. Each member of the reconstruction family could have been generated by any

number of data sets. The maximum entropy distribution is compatible with the largest number of these data sets.

3) Maximizing any other function leads to inconsistencies, unless the other function has the same maxima as entropy. [CAVA81b]

The second argument relies on the "Principle of Maximum Information", which justifies the maximum entropy choice as an extension of Laplace's "Principle of Insufficient Reason" [GUIA77]. There has been considerable debate over the use of the unbiased reconstruction as a general best solution. Pittarelli has provided a comprehensive overview of the issues involved in [PITT89]. Despite the debate, the arguments in favor of the maximum entropy solution have led to almost universal adoption of the unbiased reconstruction as the single best reconstruction.

One important implication of these arguments is that if the reconstruction family is not empty, the unbiased reconstruction will exist. In other words, if there is only one reconstruction which is consistent with the information in the subsystems, it will be the unbiased reconstruction.

2.2.3 Determining the Unbiased Reconstruction

In 1964, Ross Ashby suggested a procedure for obtaining the unbiased reconstruction for a given structure system. This procedure was implemented by Cavallo and Klir by repeatedly applying a relational join to pairs of subsystems [CAVA81b]. Jones offered a procedure which eliminates redundant substate information, improving both the efficiency and applicability of the process [JONE85a].

2.2.3.1 Independent Substates

The unbiased reconstruction is calculated for a given set D of substates. The algorithm will produce the unbiased reconstruction $U(D)$ for any set D whose reconstruction family is not empty. However, the set of substates available may contain more information than is necessary to create the unbiased reconstruction. Jones has shown that limiting D to a set of independent substates can greatly increase the efficiency of the algorithm with no loss of resolution in $U(D)$ [JONE85a].

A set of independent substates can be created using the concept of the null extension. A state α is the null extension of a substate β if $\alpha \succ \beta$ and every variable in α which is not in β takes a value of zero. A set of substates with different null extensions is an independent set [JONE85a].

The procedure for populating the set D with independent substates, presented here as algorithm ISS, involves creating equivalence classes of substates using their null extensions and selecting one substate from each equivalence class for inclusion in D. Input to ISS is a set of substates $\{\beta_i\}$. Output is the set D of independent substates.

2.2.3.2 Algorithm ISS

(1) Set D = $\{\varnothing\}$.

(2) Calculate the null extension β' of each β_i. Add β_i to a set E_i whose elements have the null extension β'. If no such E_i exists, create a new E_i containing β_i.

(3) Select one β from each E_i and add it to D.

For the example system given in Table 2, substates $^{12}(00)$ and $^{13}(00)$ share the null extension (000); $^{12}(01)$ and $^{23}(10)$ both have a null extension of (010). One substate from each of these pairs will be arbitrarily selected for addition to D, along with the remaining substates of the structure system.

2.2.3.3 Disjoint Sets

In order to ensure that the algorithm will converge to the unbiased reconstruction, the substates in D must be partitioned into disjoint sets C_l. Each C_l is formed so that no two $\beta \in C_l$ are substates of the same system state α. This step is necessary to prove convergence of the algorithm, though its practical necessity in this algorithm has been debated. This issue is discussed further in Chapter 3.

The procedure given by Jones [JONE85a] for creating the disjoint sets places each $\beta \in D$ into the lowest numbered subset in which the substate is disjoint from the other members. The process assumes the set $D = \{\beta_i\}$, $i = 1, 2, \ldots n$, is an arbitrary collection of substates of an overall system. The initial set C_1 is formed as follows:

(1) Let $\beta_1 \in C_1$. $i \leftarrow 2$.

(2) If there is no $\alpha \in C_1 : \beta_j \prec \alpha \succ \beta_i$ for some $\beta_j \in C_1$, then let $\beta_i \in C_1$.

(3) Set $i \leftarrow i + 1$. If $i \leq n$ go to (2). Else C_1 is formed.

Set C_2 is formed in the same manner from the β_i not included in C_1. The process is repeated to form $C_3, \ldots, C_m$ until no β_i remain. Since the selection of the β_i is arbitrary, the partition is not unique. For the example system from Table 2, given

$$D = \{^{12}(00), ^{23}(01), ^{12}(01), ^{23}(11), ^{12}(10), ^{12}(11)\},$$

one possible result of the partitioning process would be the sets

$$C_1 = \{^{12}(00), ^{12}(01), ^{12}(10), ^{12}(11)\}$$

$$C_2 = \{^{23}(01), ^{23}(11)\}.$$

This process has allowed the elimination of two of the eight substates from the calculations with no loss of resolution in the final reconstruction $U(D)$.

2.2.3.4 Forming Substate Equations

The subsets C_l are used to form sets of linear equations. The equations are formed from the definition of the substate function projection

$$\sum_{\alpha \succ \beta} f(\alpha) = {}^{m}f(\beta).$$

Each overall state function value $f(\alpha)$ on the left hand side of the equations is initially estimated as the system mean $1/|A|$, where $|A|$ is the number of states in the overall system. In other words, the system is initialized to a flat distribution.

In addition, one equation of the form

$$\sum_1 f(\alpha) = 1 - \sum_2 {}^{m}f(\beta)$$

is added to each C_l, where Σ_2 is taken over the β of C_l, and Σ_1 is over all α for which $\beta \succ \alpha$ is false for all β of C_l. Normally referred to as the unit normalization equation, this equation enforces the constraint

$$\sum_{\alpha} f(\alpha) = 1.$$

The estimated state function values are designated $\hat{f}(\alpha)$. The estimated $f(\beta)$ values produced by summing the appropriate $\hat{f}(\alpha)$ values are designated $\hat{f}(\beta)$.

The additional constraining equations will usually allow the elimination of one additional substate from one of the subsets C_l. Note that in the partition above, subset

C_1 contains information for every state in the overall system. In this case, the left hand side of the unit normalization equation would be empty.

If the unbiased reconstruction is being calculated from complete subsystems, then one substate whose null extension is the zero vector will be included in some C_l. If an independent set of substates has been used to generate the subsets, there will be only one such substate. By convention, this substate is normally removed from the corresponding C_l, thus ensuring that the zero vector state is on the left hand side of the unit normalization equation of each C_l.

The sets of equations for the partition given for the example system are

C_1

$$f(010) + f(011) = {}^{12}f(01)$$
$$f(100) + f(101) = {}^{12}f(10)$$
$$f(110) + f(111) = {}^{12}f(11)$$
$$f(000) + f(001) = 1 - {}^{12}f(01) - {}^{12}f(10) - {}^{12}f(11)$$

C_2

$$f(001) + f(101) = {}^{23}f(01)$$
$$f(011) + f(111) = {}^{23}f(11)$$
$$f(000) + f(010) + f(100) + f(110) = 1 - {}^{23}f(01) - {}^{23}f(11)$$

2.2.3.5 Algorithm JUR

The procedure for obtaining the unbiased reconstruction from a set of substates involves four steps [JONE85a]:

(1) Create a set D of substates using Algorithm ISS.

(2) Partition D into disjoint sets C_l.

(3) Form one equation from β in C_l. Create one unit normalization equation for each C_l.

(4) Scale left hand side of each equation to fit right hand side until convergence.

The left hand $f(\alpha)$ values are scaled by the factor:

$$new\ \hat{f}(\alpha) = \hat{f}(\alpha)\frac{{}^{m}f(\beta)}{{}^{m}\hat{f}(\beta)}$$

until the values converge.

The unbiased reconstruction for the two systems in Table 2 is shown in Table 3. By maximizing the overall entropy of the system, the algorithm has met the constraints imposed by the substates, while creating the minimum possible departure from equal probability and independence in the resulting distribution [GATL72].

Table 3. Example Solution to the Identification Problem

Variable 1	Variable 2	Variable 3	$f(\cdot)$
0	0	0	0.1500
0	0	1	0.1000
0	1	0	0.0622
0	1	1	0.1178
1	0	0	0.1200
1	0	1	0.0800
1	1	0	0.1278
1	0	1	0.2422

2.3 The Reconstructability Problem

For the reconstructability problem, the system behavior function $f(\cdot)$ is known for each $\alpha \in A$. The goal is to determine to what extent the behavior of $f(\cdot)$ can be explained by the information contained in the substate functions, ${}^{m}f(\cdot)$. If we can successfully attribute the majority of system behavior to information present in the substates, we can be reasonably confident in our ability to effectively model the system

in terms of partial models. Good solutions to this problem should allow "for the determination of the most appropriate description of an overall situation in terms of partial models, as well as for determination of the strength of structural tendencies which exist in the overall system" [CAVA82].

2.3.1 Greedy Algorithm for the Reconstructability Problem

Bush Jones developed a greedy algorithm for a general solution to the reconstructability problem for probabilistic systems in 1985 [JONE85b]. The reconstruction process described by the algorithm adds the information from a single substate at a time to a system reconstruction which is initially set to a flat distribution. A set E contains the substates to be considered for the reconstruction. The elements of E may be identified by creating a set of independent substates using Algorithm ISS presented above, or it may include all of the substates defined by the system. An independent set of substates may be created from the full substate set using the same procedure defined for the unbiased reconstruction algorithm. Function estimates $\hat{f}(\beta)$ are calculated for each $\beta \in E$, and this information is used to select the next substate and improve the reconstruction on each iteration. The substates selected form the set D, which is used to create the reconstruction.

The greedy reconstructability algorithm is outlined below as Algorithm JGR. Input is a behavior system B. Output is the reconstruction set D. In Jones' original version [JONE85b] of this procedure, E was initialized to a set of independent substates using the procedure ISS. Subsequent experience has shown that better reconstructions are normally obtained if E includes the set of all substates of the system B.

2.3.2 Algorithm JGR

1) Initialize system approximation: Set $\hat{f}(\alpha) = 1/|A| \ \forall \alpha$. Set $D = \{\varnothing\}$. Initialize E using algorithm ISS.
2) Choose the substate β in E which maximizes the choice function, $\gamma(\beta)$.
3) Add β to D. Remove β from E.
4) Compute the unbiased reconstruction $U(D)$ for the new D. The new $U(D)$ is normally computed from the $U(D)$ provided by the previous D, which greatly hastens convergence.
5) Check stopping condition. The algorithm will be stopped when the size of D has reached a predefined limit, or when $U(D)$ is sufficiently close to the true system. Otherwise, go to Step 2.

The system reconstruction is initially constrained only by the equation

$$\sum_{\alpha} f(\alpha) = 1,$$

so the function mean serves as an unbiased estimator for the function as a whole [KOLM50]. As each substate is added to the reconstruction, the function estimates will be changed only as much as necessary to meet the new constraints imposed by the additional information.

2.3.3 The Choice Function

Choosing the correct substates for inclusion in the reconstruction is essential to the reconstruction process. An implied goal of the selection process is to choose the substate which adds the most information at each iteration. If the choice function is poor, more substate equations will be required to achieve a suitable reconstruction than

is necessary with a better choice function, and a misleading model of the system may be produced [JONE85b].

The degree to which knowledge about a substate contributes to our knowledge of overall system behavior is known as the cognitive content of the substate [JONE85e]. Jones uses the information distance measure of relative entropy to derive the choice function $\chi(\beta)$:

$$\chi(\beta)={}^{t}f(\beta)\log_2\frac{{}^{t}f(\beta)}{{}^{t}\hat{f}(\beta)}+\left((1-{}^{t}f(\beta))\log_2\frac{(1-{}^{t}f(\beta))}{(1-{}^{t}\hat{f}(\beta))}\right)$$

The substate β which maximizes χ is the one whose inclusion will most improve the reconstruction [JONE85b].

Another information theoretic measure is used to measure how well the reconstruction reproduces the behavior of the original system. Known as system accuracy, this measure is defined as

$$100*\left[1-\left(\frac{\sum f(\alpha)*\log_2\left(f(\alpha)/\hat{f}(\alpha)\right)}{\sum f(\alpha)*\log_2\left(f(\alpha)/\bar{f}(\alpha)\right)}\right)\right]$$

where $\bar{f}(\alpha)$ denotes a flat distribution. System accuracy is a measure of information distance ranging from a minimum of 0.0 to a maximum of 100.0 [JONE89].

2.3.4 Reconstructability Example

The example given is from [JONE85b]. The example system models three variables, $\{v_1, v_2, v_3\}$. Variables v_1 and v_2 take values from {0,1}; variable v_3 takes values from {0,1,2}. The twelve states and their function values are shown in Table 4.

Table 4. Example for the Reconstructability Problem

State	$f(\cdot)$	State	$f(\cdot)$
000	0.079	100	0.091
001	0.088	101	0.072
002	0.083	102	0.037
010	0.031	110	0.109
011	0.052	111	0.128
012	0.097	112	0.133

The first step is to form a set E of independent substates. After eliminating redundant substate information, we have

$$E = \{^{12}(10), ^{12}(11), ^{13}(11), ^{13}(01), ^{13}(12), ^{13}(02), ^{23}(12), ^{23}(11), ^{23}(10)\}.$$

These are the only substates we will consider for inclusion in the reconstruction. The reconstruction set D is initially empty. We initialize the behavior function to a flat system:

$$\hat{f}(000) = \hat{f}(001) = \hat{f}(002) = \hat{f}(010) = \cdots \hat{f}(112) = 0.083$$

A graphical illustration of this initial setup is shown in Figure 1. The dark rectangles represent the current estimate for each state, $\hat{f}(a)$. The lighter shaded rectangles are the true function values for the corresponding states. State labels are listed along the horizontal axis and function values on the vertical axis.

The next step is to select a substate in E which maximizes the distance measure $\gamma(\beta)$. For the first iteration, the substate selected is $^{12}(11)$. The system of equations is formed by substituting the appropriate values into the two equations below and scaling the values on the left hand sides until the values converge:

$$\hat{f}(110) + \hat{f}(111) + \hat{f}(112) = {}^{12}f(11)$$
$$\hat{f}(000) + \hat{f}(001) + \hat{f}(002) + \hat{f}(010) + \hat{f}(011) + \hat{f}(012) + \hat{f}(100) + \hat{f}(101) + \hat{f}(102) = 1 - {}^{12}f(11)$$

Figure 1. Initialization of the Greedy Reconstructability Algorithm

The $\hat{f}(a)$ estimates converge to the values shown in Table 5. Note that the two constraining equations lead the function values for the corresponding states to the average of their final values. The estimated function value for each the remaining states has decreased to ensure that the system mean is maintained.

Table 5. Reconstruction After Addition of One Substate

State	$\hat{f}(a)$	State	$\hat{f}(a)$
000	0.070	100	0.070
001	0.070	101	0.070
002	0.070	102	0.070
010	0.070	110	0.123
011	0.070	111	0.123
012	0.070	112	0.123

The effect of adding the first substate to the reconstruction set D is shown graphically in Figure 2. We can clearly see that the addition of substate $^{12}(11)$ to the reconstruction has largely captured a major feature of the system.

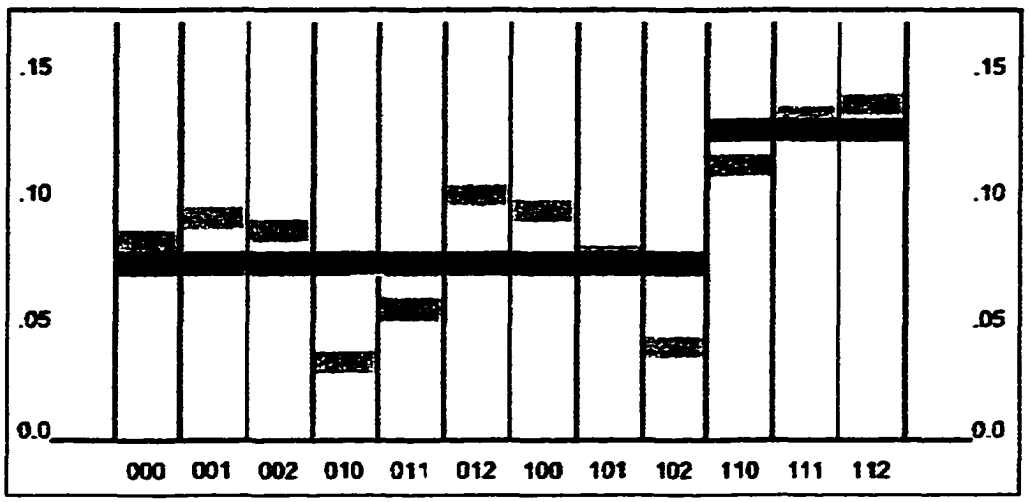


Figure 2. Reconstruction after addition of one substate

The next substate chosen is $^{23}(10)$. Since this substate is not disjoint from the one previously chosen, we form a second disjoint subset and apply a second unit normalization equation. The left hand values in the equations of the two subsets will be alternately scaled until the estimated function values converge. The resulting system of equations is

$$C_1$$
$$\hat{f}(110)+\hat{f}(111)+\hat{f}(112)={}^{12}f(11)$$
$$\hat{f}(000)+\hat{f}(001)+\hat{f}(002)+\hat{f}(010)+\hat{f}(011)$$
$$+\hat{f}(012)+\hat{f}(100)+\hat{f}(101)+\hat{f}(102)=1-{}^{12}f(11)$$
$$C_2$$
$$\hat{f}(010)+\hat{f}(110)={}^{23}f(10)$$
$$\hat{f}(000)+\hat{f}(001)+\hat{f}(002)+\hat{f}(011)+\hat{f}(012)$$
$$+\hat{f}(100)+\hat{f}(101)+\hat{f}(102)+\hat{f}(111)+\hat{f}(112)=1-{}^{23}f(10)$$

The effects of this substate interaction can be seen in Figure 3. The addition of $^{23}(10)$ to the reconstruction forces the estimate for state 110 to decrease. Estimates for 111 and 112 are forced to increase to maintain the correctness of the estimate for $^{12}(11)$.

The estimate $\hat{f}(110)$ is adjusted alternately up and down by the scaling operations on the two equations of which it is a part until the algorithm converges.

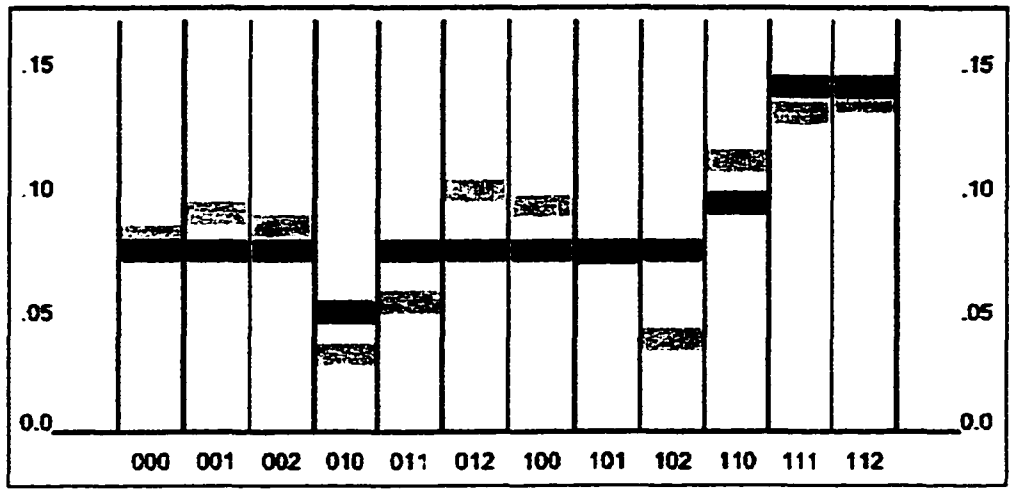


Figure 3. Reconstruction after addition of the second substate

An exact reconstruction is obtained once all nine substates from the original set E are added to the reconstruction set D, though the majority of the function's behavior is captured after only a few iterations. A summary of each iteration is given in Table 6.

2.4 K-Systems Analysis

The general algorithms presented by Bush Jones as solutions to the problems of system reconstruction form the core of k-systems analysis. These procedures do not require the reconstruction to be in the form of a structure system, but operate on an arbitrary collection of substates, shifting focus to the determination of which individual substates, or factors, are most important to the behavior of the overall system. Since many of the states in a given subsystem may contribute only minimally to overall system behavior, this concentration on individual factors enables more efficient

Table 6. Sequence of Unbiased Reconstructions

State	Function Estimate by Reconstruction Size								
	1	2	3	4	5	6	7	8	9
000	0.07	0.073	0.076	0.085	0.088	0.086	0.083	0.081	0.079
001	0.07	0.073	0.076	0.085	0.088	0.086	0.083	0.088	0.088
002	0.07	0.073	0.076	0.085	0.088	0.086	0.083	0.081	0.083
010	0.07	0.048	0.045	0.034	0.032	0.033	0.032	0.032	0.031
011	0.07	0.073	0.076	0.044	0.049	0.05	0.05	0.052	0.052
012	0.07	0.073	0.076	0.085	0.088	0.097	0.097	0.097	0.097
100	0.07	0.073	0.076	0.085	0.088	0.086	0.092	0.092	0.091
101	0.07	0.073	0.076	0.085	0.069	0.07	0.07	0.072	0.072
102	0.07	0.073	0.055	0.042	0.039	0.037	0.037	0.037	0.037
110	0.123	0.092	0.095	0.106	0.108	0.107	0.108	0.108	0.109
111	0.123	0.139	0.16	0.136	0.131	0.13	0.13	0.128	0.128
112	0.123	0.139	0.115	0.128	0.131	0.133	0.133	0.133	0.133

solutions to be specified for both the identification problem [JONE85a] and the reconstructability problem [JONE85b].

By generalizing the information system model, Jones also allows these techniques to be applied to a wider range of systems [JONE85c]. Procedures to deal with problems associated with deriving a system from arbitrary data completed the kernel of this new factor analysis technique [JONE85d].

2.4.1 G-Systems

The k-systems framework is built around two types of systems with definitions slightly different from the behavior system. The definitions and results given so far have been limited to systems with probabilistic or possibilistic behavior functions, and properties of these types of functions have been used in the development of the procedures presented for use in reconstructability analysis. Jones has shown that these techniques can be successfully applied to a much more general class of system

functions [JONE85c]. This is accomplished by transforming a general system, or g-system, to an isomorphic system which is amenable to RA techniques, known as a k-system. No information is lost via the transformation, and conversion from k-system back to g-system is straightforward [JONE85c].

A g-system can be defined for nearly any function on variables with a finite number of values. A g-system is defined as a sextuple:

$$(\tau, \{v_i\}, \{\alpha\}, \{\beta\}, f(\cdot), \{^{m}f(\cdot)\})$$

Where:

* τ is a parameter, defined as $\sum_{\alpha \in A} f(\alpha)$.
* $\{v_i\}$ is a set of variables which take values from finite sets $\{0,1,2,\ldots,n_i\}$.
* $\{\alpha\}$ is the set of states of the system.
* $\{\beta\}$ is the set of substates of the system. Each non-empty subset of the variables in $\{v_i\}$ identifies one subsystem of the system. Notation of the form $^{m}\beta$ is commonly used to identify the subset of variables that defines the substate. For instance, the substate $^{23}(10)$ denotes the set of states for which $v_2 = 1$ and $v_3 = 0$.
* $f: A \rightarrow R^{+}$ is the system behavior function. A is the set of all states of the system. R^{+} is a set of positive real numbers.
* $\{^{m}f(\cdot)\}$ is a set of functions, one for each substate, such that $^{m}f(\beta) = \sum_{\alpha > \beta} f(\alpha)$. If f is probabilistic, the $^{m}f(\cdot)$ form the marginal distributions [JONE85c].

2.4.2 K-Systems

The reconstructability algorithms require that $\tau = 1$. If this is not the case, we can define a function k, such that $k(\alpha) = f(\alpha)/\tau$, and ${}^{m}k(\beta) = \sum_{\alpha>\beta} k(\alpha)$ for each α. A new system is defined by replacing $f(\cdot)$ with $k(\cdot)$ and $\{{}^{m}f(\cdot)\}$ with the corresponding $\{{}^{m}k(\cdot)\}$. This transformation creates a k-system which is isomorphic to the given g-system. The operation removes the units from the system function, and ensures the following properties:

* $\sum_{\alpha} k(\alpha) = 1$;
* $0 \le k(\alpha) \le 1$ for all α;
* ${}^{m}k(\beta) = \sum_{\alpha>\beta} k(\alpha)$.

A k-system can thus be defined as the sextuple:

$$K = (\tau, \{v_i\}, \{\alpha\}, \{\beta\}, k(\cdot), \{{}^{m}k(\cdot)\})$$

obtained from a corresponding g-system. It is important to note that the substate equations no longer represent marginal probabilities, but that these equations are defined without adding mathematical structure that is not present in the original system [JONE85c].

The k-system transformation was proposed by Jones to allow reconstruction techniques to be used to model a wider range of functions than was previously possible [JONE85c]. In order to simplify the following discussion, and without loss of generality, we will assume $\tau = 1$, and use $f(\cdot)$ to denote the system function.

2.4.3 Comparison With Reconstructability Analysis

Though the terms k-systems analysis and reconstructability analysis are often used interchangeably, there are two primary reasons that k-systems analysis should be considered a separate field of inquiry from reconstructability analysis, despite the large overlap in the two fields.

First, the primary focus of KSA is different from that of RA. K-systems analysis is a factor analysis technique which focuses almost exclusively on identifying important substates in overall systems. Creating overall systems from partial information is normally only a step towards that goal. The concentration on substates is substantially different from the subsystem dependent structure system of reconstructability analysis.

The second reason for separating RA and KSA is a consequence of the first, and has to do with the k-system definition of a substate. The definition of a structure system requires that the variable set of a subsystem be a proper subset of the overall system variable set. This means that states cannot be substates in RA. This restriction is understandable when the idea is to model a system using subsystems.

In k-systems modeling the aim is to characterize the system in terms of important variable values and their interactions, somewhat analogous to a statistical regression. If states are not allowed in the substate set, factors which include an interaction of all the variables in the system cannot be captured. To address this problem, the g-system and k-system definitions allow substates to include any non-empty subset of the system variables.

It should be noted that the issue of restricting substates to proper subsets of system variables is not crucial to the soundness of the reconstruction algorithms, and

the restriction may be applied for applications which require it. For example, reconstructing a system with the intent of identifying possible subsystems for traditional RA modeling would best be served by the traditional RA substate definition. Of course, a different reconstruction would be expected with this restriction applied.

The k-system framework may not only be effective for the types of data analysis tasks traditionally performed using statistical regression techniques, but may prove useful in some important areas of computer science. K-system substates are essentially identical structures to the schema concept used in the analysis of genetic algorithms and classifier systems [HOLL92]. K-systems reconstruction could prove to be a useful tool for identifying important schema in these types of applications.

The substate concept might also be useful as a representation of the "partial mental states" of Minsky's coincidentally named K-line theory [MINS85]. While the current k-system framework would need to be significantly expanded to have real usefulness in these types of artificial intelligence and learning applications, the potential of this technique in these areas should not be ignored.

Chapter 3. Issues and Open Questions in K-Systems Analysis

While reconstructability analysis and k-systems analysis are important areas of systems research, there are limitations that become apparent as soon as these techniques are applied to real world systems. Attempting to reconstruct a system without properly considering the difficulties associated with modeling a system in this way can lead to a representation that is "fundamentally incorrect and, regardless of its advantages, might be vastly misleading when applied" [CAVA81a].

A review of RA and KSA would not be complete without at least a brief discussion of some of the most serious of these issues. While not completely independent, the issues can be classified according to the phase of the analysis which is most affected. Researchers must be aware of the difficulties associated with creating the system models to be reconstructed, calculating the reconstruction, and interpreting the output. While useful methods have been developed to deal with many of these problems, most are still the subject of some debate.

3.1 Creating the Model

The original data to be analyzed are not always in a form that can be directly mapped to a g-system. Data from experiments or observations do not always show the consistent behavior required to build a system model. Problems of inconsistency, state contradiction, data scattering, and missing state values require resolution before reconstruction may even be meaningfully attempted [JONE85d]. While research into definitive solutions to these problems is ongoing, effective techniques exist to minimize their impact.

3.1.1 Inconsistency

When a system is reconstructed from partial systems, it is possible that the information from the different partial systems will be inconsistent. If the projections of the behavior functions for two systems with respect to variables they share are not equal, the systems are locally inconsistent. If the reconstruction family for a set of partial systems is empty, the systems are globally inconsistent [CAVA81b].

An example of local inconsistency is shown in Table 7. In this case, there is no consistent behavior function value which can be inferred for the shared variable v_2 using the information from both systems.

Table 7. Locally Inconsistent System

System 1			System 2		
v_1	v_2	$f0$	v_2	v_3	$f0$
L	L	0.2	L	L	0.3
L	H	0.3	L	H	0.3
H	L	0.3	H	L	0.1
H	H	0.2	H	H	0.3

Strategies for resolving local inconsistencies most commonly involve defining some rational method for choosing one distribution over another, and then transmitting the choice to the overall distribution in a manner that is unbiased to the information in the system not directly involved in the inconsistency [MARI85]. There is currently no acknowledged best method for dealing with global inconsistency, and such systems are normally considered "ill-formed".

3.1.2 State Contradiction and Data Scattering

Inconsistency is not an issue for the reconstructability problem, since the overall system is already known. However, state contradiction and data scattering are two analogous problems for the reconstructability problem. State contradiction occurs when more than one function value is listed for a single state. Data scattering is a problem most commonly encountered when working with continuous variables; variable values may not have the high degree of repetition required to define a system [JONE85d].

The most common method employed to deal with data scattering is to cluster the variable values [JONE85d]. Finding the best method for clustering particular variables, and a general strategy for dealing with the resulting loss in system resolution, continue to be areas of active research.

One disadvantage of clustering variables to eliminate data scattering is that further state contradictions are often created when several data points are mapped to a single state. State contradictions are usually dealt with by averaging the different function values for a state, though choosing the minimum, maximum, or most frequent value are approaches that may also be used [JONE85d].

3.1.3 Missing State Values

When a system is being formed from arbitrary data, it is not unusual for some state function values to be missing from the data set. The usual method for replacing these missing states is a procedure known as entropy fill, whereby the missing values are assigned the mean value of the states which are present [GOUW96]. A method to

assign values to the missing states based on the values of the most closely related states has also been proposed [ASMU98].

3.2 Performing the Reconstruction

Once the data points have been transformed into a k-system model, the next phase is computerized analysis of the system. This phase brings with it a new set of problems to be addressed and decisions to be taken, each of which can affect the eventual result. While there are recommendations and heuristics that can help with this step, most are anecdotal and unpublished, and the successful completion of this phase relies heavily on the researcher's knowledge of the issues involved and the system under study.

3.2.1 Distance Functions

As in any modeling endeavor, the manner in which the results of the effort are measured is critical to the final character of the resulting system model. Derivation and evaluation of distance measures has been an active area of research in probability theory, information theory and RA since their beginnings, and the debate continues unabated [SHAN48][KOLM50][HIGA83][PITT89]. In k-system reconstruction, the two important measures are the closeness attained by the system reconstruction and the selection function used for choosing substates to include in the reconstruction. We will examine one alternative for the latter measure here.

3.2.1.1 Cognitive Content

The current substate selection function finds the substate β which maximizes the information distance measure:

$$\gamma(\beta) = {}^{k}f(\beta)\log_2 \frac{{}^{k}f(\beta)}{{}^{k}\hat{f}(\beta)} + \left(1 - {}^{k}f(\beta)\log_2 \frac{\left(1 - {}^{k}f(\beta)\right)}{\left(1 - {}^{k}\hat{f}(\beta)\right)}\right)$$

where $f(\beta)$ is the true substate value, and $\hat{f}(\beta)$ is the estimated substate value. This expression is based on the directed divergence measure from information theory, also referred to as relative entropy. Directed divergence measures the inefficiency of assuming an estimated distribution in place of the actual distribution [COVE91]. This is a min-max estimator, intended to minimize maximum error [PITT89].

3.2.1.2 Equilibrium in the Unbiased Reconstruction Process

Besides min-max estimators, there are other distance measures which could prove useful for k-system reconstructions. One way to measure the possible contribution of a substate to a system reconstruction is to measure the difference between the substate estimate and its true value. The motivation for examining this measure is rooted in the mechanisms at work in the unbiased reconstruction algorithm.

The analogy between information entropy and the entropy of physics has been the cause for considerable debate and misunderstanding [PIER80]. Though always wary of extending an analogy beyond its usefulness, we return here to the physical realm to examine the concept of equilibrium, in order to enhance our understanding of the processes involved in determining the unbiased reconstruction.

Without leaning too hard on the physical analogy, we can characterize equilibrium as a condition in which all acting influences are canceled by others, resulting in a stable, balanced, or unchanging system[AMER92]. The concepts of equilibrium, entropy and probability are entwined in physics and physical chemistry by

the second law of thermodynamics. "One statement of the second law of thermodynamics is that for an isolated system, the equilibrium state is the one for which entropy is at a maximum. From the statistical mechanical point of view, the equilibrium state of an isolated system is one that represents the most probable distribution and has the maximum randomness" [TINO95]. In the reconstructability framework this translates to the requirement that the number of ways in which constraints can be met should be maximized, one of the prime motivations for preferring the maximum entropy solution..

Equilibrium is present in the unbiased reconstruction process in some fairly obvious ways, and some which are less visible. Perhaps the most apparent is the requirement on the system function that state values sum to one, making the function isomorphic to a probability distribution. This ensures that the system mean remains constant throughout the reconstruction process; it also has the effect of reducing the amount of information that must be explicitly included in the model. "We note that by including the information [in the substate] $^{k}f(\beta)$, we also include the information $1-{}^{k}f(\beta)$ in the unbiased reconstruction" [JONE85b]. By initially setting the state values to the system mean, we create the isolated equilibrium system which characterizes maximum entropy in the physical world.

This quality persists at the subsystem level as well. Since each subsystem partitions the state set, the mean function value for each variable subset remains constant throughout the reconstruction process. And at the substate level, the linear equations which define the reconstruction ensure that every substate included in the reconstruction retains a constant value. It is two or more constraints affecting one state

that provide much of the "extra" information needed to obtain an accurate reconstruction at the state level.

3.2.1.3 Substate Centroid Distance

The manner in which the unbiased reconstruction algorithm adjusts substate values is analogous to a center of mass problem in physics. At each step, the correction of a substate function value requires the system to adjust in such a way that the center of mass (system mean) remains the same. This behavior suggests that a substate selection function based on the idea of center of mass might prove effective.

Center of mass measures have already been evaluated for use in comparing distributions. The center of mass, instead of minimizing maximum error, has been shown to minimize mean squared error [PITT89]. Center of mass expressions have the disadvantage of being hard to calculate, but the arithmetic mean of vertices (states) normally provides a good estimate for the center of mass, which neutralizes the disadvantage and confirms the intuition provided by the previous analysis that substate values would provide a reasonable distance measure in most cases [PITT89].

For a set of b states which compose a substate, the mean function value is equivalent to $f(\beta)/b$. We call this value the substate centroid. We can then define a substate centroid distance measure as

$$\delta(\beta) = \frac{\left| f(\beta) - \hat{f}(\beta) \right|}{b}.$$

There is no intent to suggest here that the substate centroid distance measure $\delta(\beta)$ is superior to the current cognitive content measure for general system reconstructions. However, in some cases, such as systems which are suspected of

containing unreliable or noisy data, minimizing mean squared error might be an approach worthy of evaluation.

The substate centroid error function is also slightly less computationally expensive to evaluate than $\gamma(\beta)$, since there are fewer terms and no logarithms to be evaluated. While the savings are not great for a single evaluation, they could be significant for the reconstruction of a very large model.

3.2.2 Disjoint Subsets and the Unbiased Reconstruction

The unbiased reconstruction algorithm developed by Jones requires partitioning the substates of the reconstruction set D into disjoint subsets C_l. This requirement was included as part of the proof that the algorithm would converge to the unbiased reconstruction [JONE85a]. It has remained an open question whether convergence of the algorithm could be guaranteed without using the partitioning procedure.

It seems that the question of whether or not the algorithm will converge is probably not especially relevant. The more important point is that without dividing the system into disjoint subsets, the algorithm is not likely to yield the unbiased reconstruction, even when it does converge.

Jones' algorithm is an adaptation of a technique for estimating probability distributions from component distributions given by Brown in [BROW59]. The component distributions are defined by subsets of the variables of the overall distribution, and are essentially identical to the subsystems of reconstructability analysis. The C_l in the Jones technique represent these component distributions.

In Brown's technique, the scaling operation is applied to each component distribution in turn. "When [component distribution] p_b is satisfied, p_a may no longer

be satisfied, so the procedure will in general require each component distribution to be employed more than once before convergence is obtained" [BROW59]. If the independent subsets are not formed, the unbiased reconstruction procedure will not be operating on a structure isomorphic to a probability distribution.

This point can be illustrated using the example system given in Table 4 in Chapter 1, taken from [JONE85b]. Consider a reconstruction using only the two substates $^{12}(10)$ and $^{12}(11)$, which are not disjoint. The problem begins as soon as the system of equations is formed. Without partitioning, the initial system of equations is:

$$\hat{f}(110) + \hat{f}(111) + \hat{f}(112) = {}^{12}f(11)$$
$$\hat{f}(010) + \hat{f}(110) = {}^{23}f(10)$$
$$\hat{f}(000) + \hat{f}(001) + \hat{f}(002) + \hat{f}(011) + \hat{f}(012) + \hat{f}(100) + \hat{f}(101) + \hat{f}(102) = 1 - {}^{12}f(11) - {}^{23}f(10)$$

Recall that the definition of a system includes the equation

$$^{m}k(\beta) = \sum_{\alpha > \beta} k(\alpha),$$

where the summation is taken over all α for which β is a substate. If we use this expression to expand the substates on the right hand side of the unit normalization equation by substituting the sum of their component states, and add the corresponding state values to both sides, we would expect to see the full set of system states on the left side equal to 1 on the right. The actual result is illustrated in (1) below. Note that because the state 110 is shared between the two substates in the system, it appears twice, and the equation does not hold. In most cases, this will lead the sum of the system states to converge to a value other than one.

(1)
$$f(000) + f(001) + f(002) + f(010) + f(011) + f(012) +$$
$$f(100) + f(101) + f(102) + f(110) + f(110) + f(111) + f(112) = 1$$

While there are several techniques one can use to try to overcome this obstacle, employing them only serves to uncover a deeper problem. A review of the scaling process shows why.

During the scaling operation for the iterative solver, each equation is transformed by an expression equivalent to

$$\text{new } \hat{a} = \hat{a}^* \frac{a}{\hat{a}}.$$

This form of the expression illustrates that each substate value will be completely corrected with each iteration. Only the interaction between equations causes the need for iterating to convergence. This is what allows the use of an expression of the form

$$(1 - (\beta_1 + \beta_2 + \ldots))$$

to evaluate the system normalization constraint

$$\sum \alpha_i = 1$$

using the same scaling technique that is employed with the other equations. Each time the solver goes through one iteration, the sum of the function estimates should be 1. If we have interacting substates, this will not be the case. In the non-disjoint case, scaling the second equation changes the value of the first equation.

If the same scaling technique is to be used on the unit normalization equation as is used for the others, the overlapping equations must be iterated until they converge before the unit normalization can be applied. Otherwise, at least one of the values in this equation will be incorrect, and the system will likely converge to some value other than 1.

Since scaling does not work, we are forced to find another way to meet the overall constraint. One way to meet the constraint under these circumstances is to change each of the excluded states by the same amount, which does not yield the same answer.

This argument is not meant to imply that there is no way to adapt the algorithm to meet the necessary constraints and still obtain the unbiased reconstruction. Rather, the intent is only to show that the current algorithm cannot be expected to create the maximum entropy reconstruction without creating disjoint subsets.

It should also be noted that this problem will likely not be an issue when creating an overall system from subsystems for a large class of problems. If information is included for all states at the in the initial reconstruction set, there might not be a need for a unit normalization equation in a single set of equations. Whether the algorithm would converge to the maximum entropy reconstruction in this case remains an open question.

When a single constraint is added to the system, the system must be adjusted in a manner that minimizes dependence between the new constraint and those previously applied if the maximum entropy system is to be obtained [GATL72]. The responsibility for restoring the system to equilibrium (sum to 1) is shared by all states not involved in the constraint, including states involved in non-disjoint constraints. Each disjoint set adjusts the values of all states not in that set. If we do not use disjoint sets, the system is adjusted based on the *cumulative effect* of overlapping constraints. If the maximum entropy reconstruction is obtained, it can only be accidentally.

Since disjoint subsets of system substates seem necessary for the greedy reconstructability algorithm, these structures may warrant evaluation as a useful tool for analysis of k-systems. While the k-system focus on substates over subsystem provides the flexibility to analyze systems which are not readily reconstructed using traditional techniques, this flexibility is not without cost. A subsystem has a well defined place as a component in the reconstructability framework; the role of a substate in defining system structure is not as clear. Disjoint sets may provide a way to further understand the structure of a system reconstructed using k-systems techniques.

Each disjoint subset C_l represents a partition of the state set, filling the role of the subsystem in reconstructability analysis. Each subset defines a component of the overall system. The unit normalization equation defines an interface between the substates of the component and the rest of the system. On this level, the subset acts as a unit, and the system reacts as a whole.

There is another level of interaction as well. The states which are shared by the various C_l define specific interactions among the components, providing more than one level of structure for the system without forcing a strict hierarchy on the system structure.

Whether the disjoint subsets contain useful information about system structure, or are simply an incidental organization imposed by the algorithm, is a question that has not even begun to be explored. However, a technique for solving to a difficult problem can often reveal something new and useful about the problem itself. The use of disjoint subsets as components for system modeling could lead to a more effective way of effectively representing complex systems.

3.2.3 System Growth Rates

One of the most important considerations for the computer analysis phase of the k-system reconstruction method is the computational complexity of these techniques, even for systems of moderate size. If a system is defined over n variables, with each variable taking k discrete values, the size of the state set is k^n.

As quickly as the size of the state set grows, the number of substates is even larger. The number of substates for this system, including the set of states, is $(k+1)^n - 1$. The number of substates which are not also system states is the difference, $(k+1)^n - k^n - 1$, a polynomial expression of degree (n-1). As the number of variables grows, the number of substates can quickly become very large in comparison to the number of states of the system. Values for the number of states and substates for some small values of n and k are given in Table 8 and Table 9.

The assumption that each variable takes the same number of values is made here only to simplify the comparison of state and substate set sizes. For variable sets whose members take different numbers of values, the expressions given here can be used with the appropriate values to compute upper and lower bounds on the sizes of the state and substate sets. The exact number of states for these systems is given by the expression $\prod_i k_i$, where k_i is the number of values for variable v_i.

3.3 Interpreting the Results

A problem that is rarely considered is determining the meaning of a reconstruction once it has been realized. If the analysis is performed using the structure system model, the strict hierarchy imposed by the model may force structure onto a

Table 8. Number of System States for Small Systems

Values per Variable	Number of Variables							
	1	2	3	4	5	6	7	8
1	1	1	1	1	1	1	1	1
2	2	4	8	16	32	64	128	256
3	3	9	27	81	243	729	2187	6561
4	4	16	64	256	1024	4096	16,384	65,536
5	5	25	125	625	3125	15,625	78,125	390,625
6	6	36	216	1296	7776	46,656	279,936	1,679,616
7	7	49	343	2401	16,807	117,649	823,543	5,764,801
8	8	64	512	4096	32,768	262,144	2,097,152	16,777,216

Table 9. Number of System Substates for Small Systems

Values per Variable	Number of Variables							
	1	2	3	4	5	6	7	8
1	1	3	7	15	31	63	127	255
2	2	8	26	80	242	728	2186	6560
3	3	15	63	255	1023	4095	16,383	65,535
4	4	24	124	624	3124	15,624	78,124	390,624
5	5	35	215	1295	7775	46,655	279,935	1,679,615
6	6	48	342	2400	16,806	117,648	823,542	5,764,800
7	7	63	511	4095	32,767	262,143	2,097,151	16,777,215
8	8	80	728	6560	59,048	531,440	4,782,968	43,046,720

system that it simply does not possess. In addition, exploring different options during the first two phases of an analysis will often yield competing models. There has been precious little published in RA and KSA research which focuses on the problem of choosing the correct model, other than an occasional reference to the need for an experienced researcher to make the correct choice.

3.3.1 Hierarchical Representation

Hierarchy is a central concept in systems science [AUGE91]. Debate on the strengths and weaknesses of hierarchical representation provide some of the most spirited debate in the literature. "Global structure, if attempted to be made local, produces contradictions and nonsense. . . . Hierarchy is characterized by a specific treelike global structure, and therefore is unable to capture other local structures obtainable through other schemes of connection" [VIXI97]. Vixie's frustration is likely the product of statements like, "The eggs of insects, fish and birds control their own development independently of external influences as long as certain environmental parameters remain constant" [TABA91]. Discussions in the literature are sometimes reminiscent of medieval astronomers arguing the exact paths that the planets take as they wind their way around the Earth.

Researchers regularly make the assumption that interactions between levels of an organizational hierarchy are not significant, even though it has been shown that the balance of entropy requires that we consider these interactions [AUGE82]. The impact of this assumption on the quality of the resulting analysis almost certainly depends on the system in question, and it seems that any attempt to define a structure which is appropriate for all systems is at best naïve.

The impact on these problems of using k-systems and the Jones algorithms is unclear, where the focus is on substates instead of subsystems. If subsystems are present, their relative importance can be evaluated *after* the reconstruction is complete by the arrangement of substates. In addition, the cognitive content and closeness measures provide an unbiased evaluation of the relevance of any reconstruction. A comprehensive investigation into the role of the disjoint subsets discussed in the previous section could provide important insights into this topic.

3.3.2 Comparing Reconstructions

Performing an analysis using the k-system framework normally requires exploring different options during the first two phases of the process, which often results in the generation of several competing models. There are accepted closeness measures that are quite useful for determining how well a reconstruction captures a system's behavior [JONE89], but the models created using variations of k-systems techniques often achieve very similar closeness values. We can measure how well a system is reconstructed, but we currently have no way to talk meaningfully about the structure of a particular reconstruction.

Unfortunately, techniques for comparing reconstructions do not tend to focus on these types of questions. Some research has been done on analogies between systems [BUNG81][LIN87][FLOO90], but this work is based on a definition of a system which is not directly transferable to the RA system model, and is focused on relations which are not even defined in the k-systems framework. More importantly, the goal of general systems analogy research is focused on comparing systems from different domains, and is of little use in comparing system reconstructions.

No ready solution to this problem has been offered to date, but the similarity between k-system substates and schema suggest one direction the investigation may take. Measures exist in the domain of genetic algorithms and classifier systems which may prove useful for detecting properties of reconstructions which would aid in their comparison.

For example, the concepts of generality and specificity have been used for generating rules in classifier systems and other rule learning applications [BOOK90]. A rule R1 is considered to be more specific (or less general) than a rule R2 if and only if R1 will apply to a proper subset of the instances in which R2 will apply [MITC77].

This idea could be adapted to the k-systems framework by defining a specificity measure as the number of variable values present in the substate label. In probability theory, this value is often called the order of the distribution [BROW59]. Calculating an average order for all of the substates in competing reconstructions should give an indication of the relative sizes of the features being represented by each model.

Another measure can be derived to quantify the intuition that a more compact reconstruction is superior to a larger one which captures the same amount of information. The difference in system closeness before and after a substate is added to the reconstruction provides an indicator of that substate's contribution to the reconstruction. The average of this value for all substates in the reconstruction set should provide a fair measure of the efficiency of the reconstruction efficiency, at least in terms of the value of the included substates.

Chapter 4. Directed Search in System Reconstruction

Since exponential system growth limits practical computerized analysis to systems of only a few variables, it is desirable to find ways around the problem. One of the most computationally expensive steps in the greedy reconstruction algorithm can be the selection of the next substate for inclusion in the reconstruction set D.

4.1 Cost of Searching All Substates

As discussed in Chapter 2, selecting the next substate to include in a reconstruction involves finding the substate β which maximizes the information distance measure:

$$\gamma(\beta) = {}^{k}f(\beta)\log_2 \frac{{}^{k}f(\beta)}{{}^{k}\hat{f}(\beta)} + \left((1-{}^{k}f(\beta))\log_2 \frac{(1-{}^{k}f(\beta))}{(1-{}^{k}\hat{f}(\beta))} \right)$$

where $f(\beta)$ is the true function value, and $\hat{f}(\beta)$ is the current estimate of the function value for substate β. Identifying the desired β normally involves evaluating $\gamma(\beta)$ for every substate of the system.

As the number of variables n increases, the number of substates will quickly become very large in comparison to the number of states of the system. This is illustrated for systems of binary variables in Figure 4.

Since the evaluation of a substate requires taking the sum of several $f(\alpha)$ values, the number of $f(\alpha)$ evaluations required seems to provide a more meaningful measure of the computational cost of searching all substates than simply counting the number of substates. In order to simplify the analysis, we will once again assume that each of the n variables of the system takes values from a set of k elements.

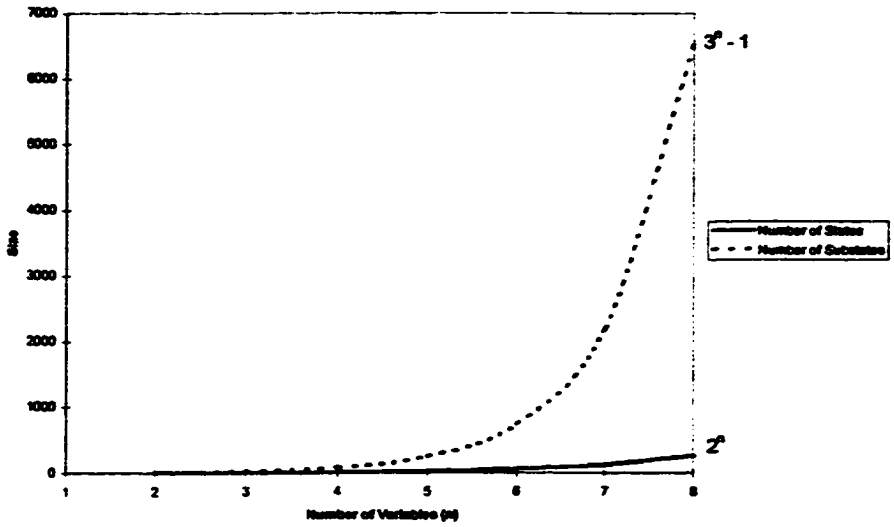


Figure 4. State vs. Substate Growth for Systems of Binary Variables

As stated previously, the number of substates for a system of n variables which take k values each is $(k+1)^n - 1$, and each would be evaluated in an all substate search. However, the number of state evaluations required to search the substate space is more conveniently derived by briefly returning to the notion of subsystems.

Recall that a subsystem is defined by any non-empty subset of the system variables. Since a k-system substate search will normally include the state set A, we do not enforce the structure system requirement that the subsystem variable set be a proper subset of the variable set. The states of a subsystem induce a partition of the states of the original system, so the evaluation of the substates in each subsystem will require evaluating each of the k^n system states exactly once. Multiplying this value by the total number of subsystems defined by the system will yield the total number of state evaluations required for a search of all substates.

The number of subsystems of size m defined for a system is the number of m-element subsets of the variable set V. Applying the binomial theorem [BOGA88] allows us to determine directly that the total number of state evaluations for an all substate search is

$$\sum_{m=1}^{n}\binom{n}{m} * k^n = (2^n - 1) * k^n.$$

4.2 Strategies for Reducing Substate Evaluations

One way to speed the reconstruction process would be to find a way to select the next substate while evaluating only a fraction of the system's total substates at each step. Reducing the size of the set E which contains the substates being considered for inclusion in the reconstruction set D would achieve this aim. For systems with a substate set which is large enough to make computerized analysis problematic, the benefits of this capability should outweigh even a significant loss of discrimination in the reconstruction process.

4.2.1 Independent Substates

One approach to reducing the number of substates considered is to limit the set E to a single set of independent substates, as described in [JONE85a] and reviewed in Chapter 2. When this method is used, the set of independent candidate substates is generated before reconstruction begins, and the reconstruction is computed using only these substates. This approach has been shown to provide a correct reconstruction, and has allowed analysis of systems that would not otherwise be practical [JONE85a].

There are, however, two disadvantages to using independent substates. First, the process requires all substates to be evaluated for membership in one of the equivalence

classes E_i. This results in a computational complexity on a scale with that required when using all substates.

The second disadvantage is that there is at present no definitive method for choosing an optimum set of independent substates to include in the reconstruction. While it is not possible in the general case to determine a unique optimal reconstruction set for a given system, it is reasonable to consider a smaller reconstruction set to be superior to a larger set which captures the same information. Experience has shown that systems can be reconstructed to a given degree of accuracy using significantly fewer substates when all β are included in E, instead of using only independent substates.

4.2.2 Substate Pruning

One variation of the independent substate technique would be to limit the substate search to substates which are independent of those already in the reconstruction. This technique is referred to here as substate pruning. This modification of the original independent substate technique begins by initializing E to include all β. When a substate β_i is selected for addition to the reconstruction set D, then β_i and all substates with the same null extension as β_i are removed from E.

A preliminary analysis of the feasibility of using this technique was conducted using data from all substate reconstructions. As expected, the results show that two or more substates with the same null extension are often included in a system reconstruction, indicating that this technique would likely lead to larger reconstruction sets than the all substate search technique. By the time redundancies were encountered in the test systems, about half of the substates had usually been eliminated from the

search space. Reduction in the search space was not as great as the directed search technique presented below, so further investigation of this technique was not pursued.

One interesting result from this investigation was the observation that redundant states normally appear late in the reconstruction process. In each of the test systems, the first appearance of non-independent substates was after the system was within a few percent of its final accuracy value. If this behavior could be shown to be a general feature of the reconstructability algorithm, the first appearance of redundant information could serve as a flag indicating a point of diminishing returns, and signal a natural place to conclude the reconstruction.

4.2.3 Directed Search

Another strategy for limiting the substate search space is to make use of the information present in the system states during the reconstruction process. While it is important to resist the temptation to assume some particular structure for the system, it is reasonable to assume some structure is present. After all, this is the assumption underlying the belief that a system can be reconstructed at all. In particular, we can reasonably assume that substates which contain the most information are more likely to be substates of the states which contain the most information. If α_{max} is the state for which $\gamma(\alpha)$ is maximized, we might expect $\gamma(\beta_i)$: $\beta_i \prec \alpha_{max}$ to be greater than $\gamma(\beta_j)$ for β_j which are not substates of α_{max}. This is the central assumption underlying the idea of the directed search technique presented here. The approach involves identifying the states α_i with the highest information distance values and expanding them to find the maximum distance substate which contains each α_i.

4.3 Directed Search of Substate Expansions

We define the state set for a substate β to be $S(\beta) = \{\alpha : \alpha \succ \beta\}$. β_j is an expansion of β_i if $S(\beta_i) \subseteq S(\beta_j)$. The expansion β_j is obtained by dropping one or more ordinates from β_i. An expansion obtained by dropping exactly one variable from β_i is called an immediate expansion. The number of immediate expansions of any β is the order of β. The total number of expansions for β is $2^{\text{order}(\beta)} - 2$.

The directed search version of the greedy reconstructability algorithm recreates the candidate substate set E at each iteration. The elements of E are determined by creating a set M of candidate states with $\gamma(\alpha)$ values which are close to $\gamma(\alpha_{max})$. A single expansion of each $\alpha \in M$ is then selected for addition to E. The $\beta \in E$ which maximizes $\gamma(\beta)$ is then selected for inclusion in the reconstruction set D.

4.3.1 Forming the Candidate State Set

Forming the set M first requires the determination of two distance values, d^+ and d^-, where

$$d^+ = \max \gamma(\alpha) : \hat{f}(\alpha) - f(\alpha) \geq 0$$

$$d^- = \max \gamma(\alpha) : \hat{f}(\alpha) - f(\alpha) < 0$$

These distances are used to evaluate each $\alpha \in A$ for inclusion in one of two sets, M^+ and M^-, which will divide the candidates based on whether their estimates are greater than or less than their true values. The elements of these sets are determined by

$$M^+ = \{\alpha : |\gamma(\alpha) - d^+| \leq tol, \hat{f}(\alpha) - f(\alpha) \geq 0\}$$

$$M^- = \{\alpha : |\gamma(\alpha) - d^-| \leq tol, \hat{f}(\alpha) - f(\alpha) < 0\},$$

where *tol* is a preset tolerance parameter which determines how restrictive the search will be. The candidate state set M is defined by $M = M^{+} \cup M^{-}$.

A set of states is chosen to expand instead of a single state for two reasons. First, in the case where α_{max} is not unique, choosing a set of states prevents an arbitrary choice from among a set of states with equal $\gamma(\alpha)$ values, and generally leads to much better reconstructions. To achieve this, the tolerance parameter *tol* is set to a value just large enough to account for round off error and sampling tolerances. A value of $tol = 10^{-10}$ was used for this purpose in each of the test runs presented here.

Adjusting the value of the tolerance parameter would allow the user to adjust the completeness of the search. Assigning a value of $tol = 0$ would limit the set to states with distance values exactly equal to the maximum distance. A large enough value would ensure that all states were expanded. While an expansion of the entire state set A would likely search all substates, the algorithm in its current form would not be guaranteed to search all of the lower order substates.

4.3.2 Expanding the States

Once the set M of candidate states is formed, a single expansion of each $\alpha_i \in M$ is selected for inclusion in the candidate substate set E. The selection process is outlined below:

(1) Set $\beta_{max} = \alpha_i$.

(2) Calculate $\gamma(\beta_j)$ for each β_j which is an immediate expansion of β_{max}. If $\gamma(\beta_j) \geq \gamma(\beta_{max})$ for any β_j, set $\beta_{max} = \beta_j$.

(3) If β_{max} is unchanged from the previous iteration, or if the order of β_{max} is 1, stop. Otherwise, go to Step 2.

4.4 Directed Search Algorithm DS1

An algorithm to implement the state expansion concept has been developed to test this approach. The process requires forming the set of states to be expanded, then expanding each state to find the maximum error substate. Input to the algorithm is the system $K = (\tau, \{v_i\}, \{\alpha\}, \{\beta\}, f(\cdot), \{^m f(\cdot)\})$, and the current reconstruction $U(D)$ for the current iteration of the algorithm JGR. Output is the set of candidate states E to be evaluated for addition to D by JGR. The procedure is carried out in four steps:

(1) Set $E = \{\varnothing\}$.

(2) Calculate d^+ and d^-.

(3) Create set $M = M^+ \cup M^-$.

(4) Selectively expand each $\alpha_i \in M$ to find $\beta_{max}(\alpha_i)$. Set $E = E + \beta_{max}(\alpha_i)$ for each $\alpha_i \in M$.

The expansion of a state α_i will not necessarily search all of the substates of α_i. If a system contains n variables, each substate of order $(n\text{-}1)$ will be evaluated before any substates of order $(n\text{-}2)$ are checked. Only the substate with the maximum $\gamma(\beta)$ value at each level is expanded further. The expansion continues only as long as substates with greater cognitive content are being discovered. The process is illustrated in Figure 5.

The example shown is an illustration of the first substate selection for one of the test systems used to evaluate the algorithm. The substates selected by expanding states

101 and 510 have equal distance values. In this case, the final choice of the substate selected for inclusion in the reconstruction is arbitrary. Since the two substates shown partition the states of the system, either one adds the same amount of information to the reconstruction.

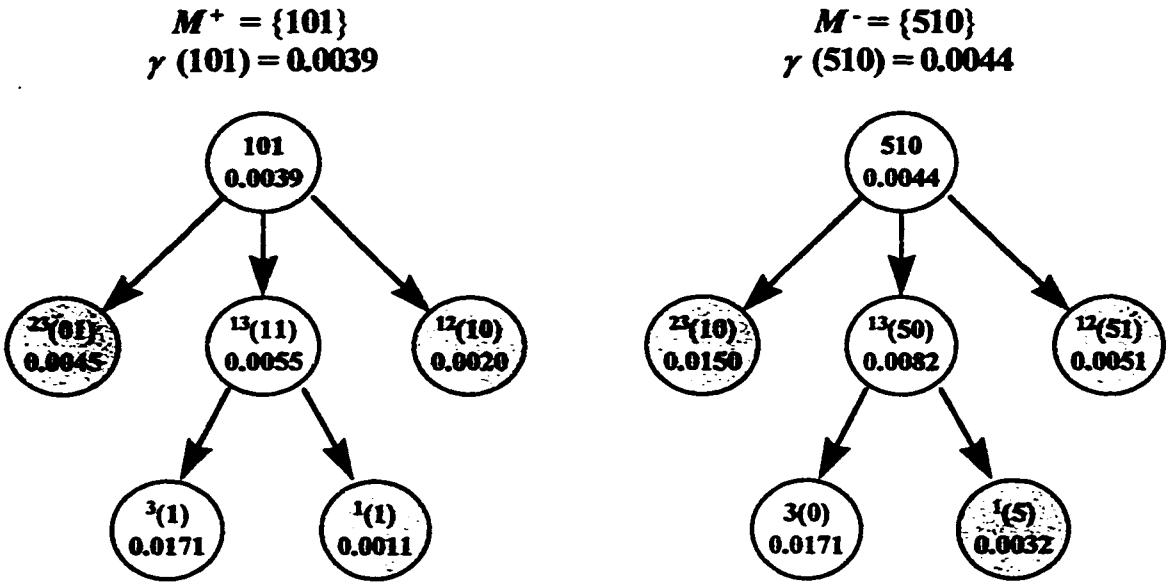


Figure 5. Example Substate Expansion

4.5 Evaluation of Directed Search

An evaluation of the directed search algorithm would ideally answer two questions. First, how much computational savings can be expected using this technique over the all substate search? Also, how good will the resulting reconstruction be in comparison to the all substate reconstruction?

Unfortunately, neither question is easy to answer. The reduction in computation depends on the number of states selected for evaluation, the number of values that each variable can take, and the particular substates that are chosen during the directed

search. And we have already seen that there are limited existing measures for comparing two reconstructions.

Because of the difficulties inherent in undertaking a theoretical analysis of the effectiveness of this technique, a preliminary experimental analysis was conducted to compare the directed search technique to the all substates search. Of primary interest in the analysis were the average reduction in state and substate evaluations, and the number of factors required to reconstruct 99.5 percent of the system's behavior. If the addition of a single factor ever produced a change in system closeness of less than 0.1, the reconstruction was halted. In these cases, final closeness values would be compared for the two techniques as another direct measure of reconstruction quality.

4.5.1 Sample Data

The program was tested on three sample systems. The systems are intended to represent a range of system types that are commonly encountered. Data Set 1 is the example system from [JONE85b]. This is an example of the types of systems often found in the literature, with a small number states and high degree of variation; qualities which most clearly illustrate the workings of the algorithm.

The other two systems are data are from the biological sciences, and represent the type of data often produced from actual surveys and studies. Data Set 2 is from a study of caloric intake, and Data Set 3 was collected as part of an oil spill bioremediation study. Data Set 2 required variable clustering, missing state replacement, and averaging of inconsistent state function values. Data Set 3 was the largest of the three systems and included a significant number of states with a function

value near zero. Summary statistics for the three k-systems are shown in Table 10. Details concerning the data and reconstructions can be found in the Appendix.

Table 10. Sample Data Summary Statistics

Data Set	1	2	3
No. of Variables	3	3	5
No. of Values	(2,2,3)	(6,3,2)	(2,5,2,2,3)
No. of States	12	36	120
No. of Substates	35	83	647
System Mean	0.08333	0.02778	0.00833

4.5.2 Results

A C++ implementation of the directed search algorithm was tested on the three sample sets. A summary of the reduction in substate evaluations is shown in Table 11.

Table 11. Reduction in Substate Evaluations Using Directed Search

Data Set	Total Evaluations Using All Substate Search	Total Evaluations Using Directed Search	Directed Search Reduction
1	280	58	0.793
2	1577	149	0.906
3	25233	1834	0.927

A summary of the reduction in state evaluations is shown in Table 12. One reason for the smaller reduction in state evaluations than in substate evaluations is that the state evaluations required to identify the maximum error values d^+ and d^- are included in these figures. This means that each state will have been evaluated once before the search process begins. In practice, these evaluations could easily be

combined with bookkeeping and reporting functions which are normally carried out between iterations.

Table 12. Reduction in State Evaluations Using Directed Search

Data Set	Total Evaluations Using All Substate Search	Total Evaluations Using Directed Search	Directed Search Reduction
1	672	356	0.470
2	4788	2171	0.547
3	145080	25186	0.826

Overall, the directed search technique produced averages of 88 percent fewer substate evaluations and 61 percent fewer state evaluations, when compared to searching all substates. In each case, the directed search algorithm was able to match the closeness of the all substates search reconstruction to within one percent, and the difference in the number of factors required was either equal or plus one for directed search.

A more efficient search technique would be of little use if it caused slower convergence of the unbiased reconstruction algorithm, so the number of iterations required for the iterative solver to converge was also tracked. No apparent pattern was evident. Data Set 3 required approximately 6% fewer iterations using the all substate search, while Data Set 2 converged in around 25% fewer iterations with directed search. The results for Data Set 1 were identical for both methods.

4.5.3 Computational Complexity

Suppose a selected state will be expanded to level m. If n is the number of variables in the system, $n - m$ will be the size of the variable subset defining the

substates generated at level m. The number of substates evaluated at level m will be $(n - m + 1)$. Evaluating substate i, generated by removing variable i from the substate representation, requires $\prod_{i=1}^{m} v_i$ state evaluations, where v_i is the number of values taken by variable i. Returning to our assumption of a constant k values per variable, we can obtain a rough estimate of k^m as the number of state evaluations per substate at level m, giving a total of $(n-m+1)*(k)^m$ state evaluations at level m. Summing this expression over the number of levels expanded gives the number of state evaluations per selected state. Multiplying by the number of states selected gives an estimate of the number of state evaluations for a single search.

This analysis demonstrates that though the number of substate evaluations is most sensitive to the number of states selected and the number of variable-value combinations in the system, the total number of state evaluations is dominated by the depth of the search. This fact partially explains why the directed search technique can achieve such dramatic reductions in total state evaluations, despite the added cost of two full iterations of the state space. The substates which are the most expensive to evaluate are the last to be explored.

In the worst case, enough states will be selected for expansion, and the expansion will proceed to a sufficient depth, that all substates will be evaluated. In this case, the directed search will actually be more expensive to compute than the all substate search, because of the required search of the state set A. This case should only occur when large numbers of states all take the maximum error value, or when the tolerance parameter is large.

The best case occurs when the state α_{max} is chosen as the best substate. The immediate expansion of α_{max} requires n substate evaluations, where n is the number of variables in the system. Since no better candidate is found, the search ends after n substates have been evaluated. Since the process will always choose at least two states for expansion, the total number of substates evaluated will always be at least $2n$.

When only the first level expansion of α_{max} is performed, $f(\beta)$ for each of the n expansions of α_{max} will sum over k states, leading to a best case of $2nk$ state evaluations for a single search.

Overall, we can expect the savings with directed search to be greatest when the system behavior is dominated by high order interactions. The technique will be less effective for systems whose behavior is caused by single variable effects or interactions of just a few variables, due to the need to search the lower order substates more often.

4.5.4 Reconstruction Quality

The reconstructions obtained by the directed search algorithm capture as much of the system's behavior as the all substate search using almost the same number of factors. In this respect, the fact that the technique may produce a different reconstruction than searching all substates should be of no more concern than the fact that a given number may be decomposed into more than one set of factors. In a very real and important sense, the directed search technique can be considered an effective alternative to searching all substates.

However, if the goal is to explore system structure, it seems that merely measuring the degree to which a system's behavior is reproduced is only part of what is needed.. It seems irresponsible to assert that we can impose a structure on the search

process without imposing some structure on the resulting reconstruction. The directed search given here is essentially an "outside-in" process which explores the most visible structures until deeper order is revealed. This mechanism also seems to be present to some degree in the all substates search, though the fact that the two techniques do not always choose the same factor implies that other mechanisms are also at work in the more general search technique.

While there is no existing framework for a rigorous comparison of the two techniques, the principle differences can be illustrated through the use of an analogy. The concept of a fitness landscape is a central element of the study of complex systems and machine learning [HOLL95]. In this context, the goal is normally to find the highest or lowest point on a landscape which has not been fully explored. However, solving the reconstructability problem can be viewed as an attempt to identify features of a known landscape which are responsible for its overall shape.

The analogy from function optimization to landscape features is a powerful one, and brings concepts which are difficult to visualize into familiar territory. Not surprisingly, extending the metaphor only slightly brings quickly to light at least two differences between the all substate search mechanism and the directed search.

Suppose that a system function defines the landscape pictured in cross-section in Figure 6. The goal is to identify the land forms which cause the landscape to differ from its average elevation. Looking only at the surface of the landscape, we would naturally frame our explanation in terms of the two hills and the valley that are evident. Geographers create contour maps to highlight just these types of features.

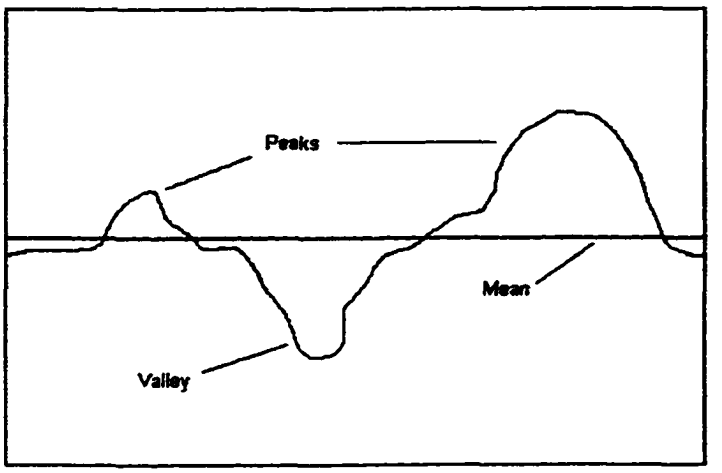


Figure 6. Simple Fitness Landscape

Geologists explain landscape features in different terms. Their approach is normally to explain surface features in terms of processes operating on the underlying rock strata [BIRK78], as illustrated in Figure 7. Each of the shaded regions represents a different rock layer. This additional information suggests an alternate explanation for the landscape features than that provided by viewing only the surface features.

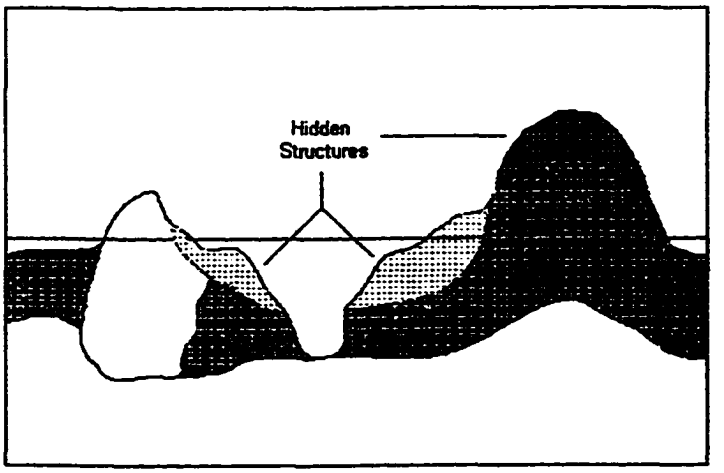


Figure 7. Landscape Showing Underlying Structure

Two concepts illustrated by Figure 7 can be mapped directly to behavior exhibited by the directed search technique. First, the small layer at the top of the tallest peak conceals a more significant feature below. The directed search algorithm, which will begin its search at the peak, will be unable to detect the deeper structure.

The second limitation of directed search is analogous to the layer which is bisected by the valley towards the center of the landscape. Two or more substates which do not contain enough information to justify further exploration may share an expansion which *is* significant, but will not be detected. An illustration of this phenomenon is shown in Figure 8.

As the example shows, neither B_{11} or B_{12} are selected for further investigation by the search algorithm, even though they share a substate which would be selected if all substates were searched. Examination of the sample data indicates that this phenomenon is likely to be the most common factor leading to different reconstructions from the two search techniques.

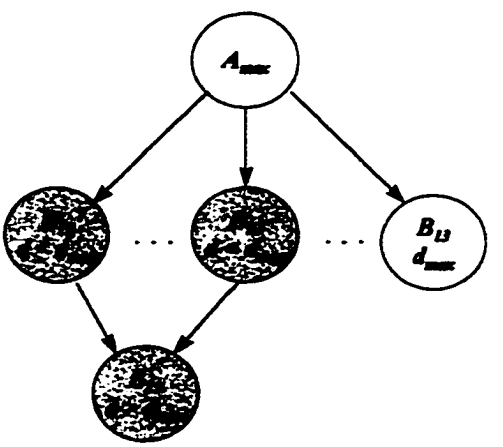


Figure 8. Directed Search Failure to Detect Maximum Distance Substate

In light of previous discussion, we might expect the directed search technique to yield substates which were more specific than those created by the all substate technique. This intuition seems to be confirmed by the test runs, though the effect is not as pronounced as might be expected. Recall that the average order for a reconstruction was defined in Chapter 3 to be the average number of variables present in the substates which are included in a reconstruction. The average order of the all substates reconstruction for Data Set 2 was 2.60; the same value for the directed search was 2.63. For Data Set 3, both techniques yielded an identical value of 4.13.

A measure of average substate contribution was also defined in Chapter 3 to be the average accuracy added to a system representation by each substate in the reconstruction. The average substate contribution for Data Set 2 using the directed search was 4.98. The all substate technique produced a slightly higher value of 5.24. For Data Set 3, the values were 2.55 for the all substate search and 2.48 for the directed search. Since both techniques produced essentially identical closeness values for all of the data sets, the difference in substate contribution is wholly attributable to the ability of the all substate search to produce slightly smaller reconstructions in some cases.

The ability to evaluate different models is not only important for determining important factors for a system. Many modeling systems, such as the Copycat system for analogy formation [HOFS95], operate by generating several models simultaneously and evaluating each for strengths and weaknesses. Exploring the issues involved with evaluating competing models is essential for creating a generalized modeling system from the k-system framework.

4.6 Other Optimizations

While factor search is one of the more computationally intensive steps of the reconstruction process, there are other optimizations that can improve the usefulness of the algorithm. One is to parallelize the entire process. Another helpful technique would be to reducing the number of additions required to calculate the function values for the entire system.

4.6.1 Parallel Reconstruction

One strategy for dealing with system growth is to provide a parallel version of the k-system algorithms [ILES95], which allowing analysis to be performed on larger systems than was previously feasible. Patti Aymond (formerly Patti Iles), has developed parallel versions of the unbiased reconstruction algorithm [AYMO97] and the greedy reconstructability algorithm [ILES95]. The parallel algorithms are designed for a hypothetical multiple instruction multiple data stream system with a dynamic bus system, and addresses the problems encountered when the number of data values is greater than the number of processors in the system [ILES95].

There is at least one element of the parallel algorithms which will likely prove useful in sequential implementations as well. The parallelization of the algorithms requires an efficient method for distributing tasks among the processors of the system. Since the tasks are centered around the states and substates of the system, a way had to be found to directly determine a processing element number for any given substate.

The solution to this problem recommended by Aymond is a substate enumeration based on a string encoding technique known as Godel numbering. A

discussion on the usefulness of this technique for sequential implementations is given in the next section.

4.6.2 Minimizing State Evaluations in K-System Reconstruction

For a k-system, the function value of a substate β is defined to be $\Sigma\alpha_i$, where the α_i are all of the states for which β is a substate. The unbiased reconstruction algorithm calculates this sum for each substate function estimate $\hat{f}(\beta)$ in the reconstruction set D on every iteration. Any one of these substates may sum over as many as half of the states in the system. In addition, the sum is calculated for each substate being considered for addition to the reconstruction set by the greedy reconstructability algorithm.

An enumeration technique is presented here which may reduce the number of state evaluations required for this process, as well as simplifying the notation for state and substate labeling. The motivation is an observation concerning the hierarchical relationship of substates of different orders.

Consider a substate ${}^k\beta$ of order $m < n$. If v_i is a variable not included in ${}^k\beta$, then $S({}^k\beta)$ is equivalent to the union of all $S({}^j\beta)$: ${}^k\beta$ is an immediate expansion of ${}^j\beta$, and ${}^j\beta$ includes v_i. This implies that the function value $f(\beta)$ for any substate of order m can be calculated by the sum of a relatively few substates of order $m + 1$.

A system can be structured so that this substate hierarchy can be exploited. The process begins by imposing a total ordering on the substates (including the system states). This ordering is a variation of the godelization technique described by Iles for the parallel reconstruction algorithm [ILES95].

Instead of the traditional method of labeling a variable with k values from 0 to k-1, we use labels (1, 2, ... k). A zero value in a label denotes a substate which excludes that variable. A label of this form is created for each substate, and the labels are ordered in the traditional way. For example, the substate traditionally described as $^{23}(00)$ in a three variable system would be labeled (011).

Once the states have been labeled and ordered, we assign an integer index $g(\beta)$ to each substate. This process is illustrated in Table 13 below. The first column shows the traditional notation for each of the substates. The value (and inverse) of this index can be computed in $O(n)$ time, where n is the number of variables.

The last column of the table lists expressions which can be used to calculate the substate values. The numbers shown are the substate indexes for the corresponding states/substates.

Inspection of Table 13 confirms that each substate can be calculated using only values with a higher index. The ordering has imposed a hierarchy on the substates; each substate which spans more than one variable can be partitioned by substates with a higher index. Thus, if we work up from the bottom of the structure, each substate can be calculated using values which have already been calculated. This will reduce the number of state evaluations significantly, since only substates which aggregate a single variable must be calculated directly from state values. All other substate values can be computed from previously computed substates. In the example given, the number of state evaluations required was reduced from 84 to 36. The net reduction in computation is not quite as large, since substate values must still be calculated from other substates

Table 13. Example of Substate Numbering Scheme

Traditional Denotation	Substate Index $g(\beta)$	Substate Label		Equivalent Sum
	0	000	=	1+2+3
$^{3}(0)$	1	001	=	5+9
$^{3}(1)$	2	002	=	6+10
$^{3}(2)$	3	003	=	7+11
$^{2}(0)$	4	010	=	5+6+7
$^{23}(00)$	5	011	=	17+29
$^{23}(01)$	6	012	=	18+30
$^{23}(02)$	7	013	=	19+31
$^{2}(1)$	8	020	=	9+10+11
$^{23}(10)$	9	021	=	21+33
$^{23}(11)$	10	022	=	22+34
$^{23}(12)$	11	023	=	23+35
$^{1}(0)$	12	100	=	13+14+15
$^{13}(00)$	13	101	=	17+21
$^{13}(01)$	14	102	=	18+22
$^{13}(02)$	15	103	=	19+23
$^{12}(00)$	16	110	=	17+18+19
	17	111		
	18	112		
	19	113		
$^{12}(01)$	20	120	=	21+22+23
	21	121		
	22	122		
	23	123		
$^{1}(1)$	24	200	=	25+26+27
$^{13}(10)$	25	201	=	29+33
$^{13}(11)$	26	202	=	30+34
$^{13}(12)$	27	203	=	31+35
$^{12}(10)$	28	210	=	29+30+31
	29	211		
	30	212		
	31	213		
$^{12}(11)$	32	220	=	33+34+35
	33	221		
	34	222		
	35	223		

which partition them. Still, the savings in computation could be significant for a large system.

More careful study reveals a pattern in the sums that is relatively simple to characterize. The value for $f(\beta)$ may be calculated for any substate β by taking a sum over any variable i for which the label of β is 0. This sum, denoted here $f_i(\beta)$, can be calculated by the expression

$$f_i(\beta) = \sum_{j=1}^{|v_i|} f({}^{j}\beta) \text{ where } g({}^{j}\beta) = g(\beta) + j * step$$

where i is the variable over which the sum is taken. The variable *step* is calculated as

$$\prod_{k>i} (|v_k| + 1).$$

Note that all substates whose labels contain more than one zero value have more than one possible way to calculate their sums. For example, substate 24, labeled 200, can be written either as the sum (201 + 202 + 203), or as (210 + 220). Both characterizations are correct, and each has its own advantage. The sum over v_3 includes the next three substate numbers in order. In general, summing over the rightmost zero value involves the simplest step calculation. However, taking the sum over v_2 in this case involves summing fewer states. For a system with high variance in the number of values taken by the different variables, this approach may prove worth the extra computational expense of identifying the smallest set.

Also note that once a substate has been included in a system reconstruction, its estimated value will not change significantly from one iteration to the next. The value is fixed by the corresponding substate equation. Therefore, these substates need not be

evaluated again during the search process, though their estimates may be used in sums for those substates which precede them in the ordering.

Use of this structure allows reduction of additions for substate function values to the number of values taken by one of the system variables. In addition, the structure is compact and the labeling notation straightforward. Thus, this ordering provides both a compact data structure and an efficient evaluation technique computation involving k-systems.

Chapter 5. Conclusions

5.1 Significance of K-Systems Analysis

The k-systems analysis techniques described here have already had an impact in the area of data analysis. When coupled with procedures to deal with problems associated with arbitrary data [JONE85d], these algorithms provide a useful tool for real world data analysis, and a powerful adjunct to classical techniques [JONE86]. Evidence exists that the greedy reconstruction algorithm may provide a more correct characterization of interaction effects than classical regression techniques [GOUW96]. A program implementing k-systems analysis [JONE89] has been used successfully to glean information on important interactions in ecological data which standard regression techniques were unable to detect [SHAF97].

While k-systems techniques are unlikely to replace classical statistics in practice, they can provide a useful tool in many areas besides traditional data analysis. For example, the concept of a substate is essentially equivalent to a schema in genetic algorithms and classifier systems. The greedy algorithm's ability to explain fitness values in these terms could lead researchers in these fields to new insights into the fitness landscapes on which their systems evolve.

5.2 Significance of the Current Research

For the benefits of k-systems analysis to be generally useful, techniques must be developed which allow the investigation of larger systems than is now practical. The directed search technique is a step towards this goal. The algorithm provides better results than can be expected using arbitrarily chosen independent substates, while requiring only a fraction of the state evaluations required by the all substate search.

While the directed search algorithm cannot be expected to provide better results than the all substate search, it is an attractive alternative when searching the entire substate space is not practical.

It is also important that the results provided by different reconstruction methods are clearly understood. Besides allowing for analysis of larger systems, the directed search technique, as well as the use of alternate distance measures, highlight a need for new ways to compare alternative reconstructions for a single system. Executing the greedy algorithm on a single system using different variations of measures and search techniques will provide different models of system behavior, and current tools are inadequate to meaningfully compare these different models. Measures such as average substate order and average substate contribution are first steps toward a more powerful framework for evaluating competing models than currently exists.

The work on disjoint subsets in the unbiased reconstruction partially answers a long standing question concerning the algorithm, and may lead to new understanding of the meaning of the system produced using this technique. Current reconstruction of systems provides little more than an ordered list of substates. The use of disjoint subsets as components of the system model could lead to a more structured approach which avoids the strict hierarchy imposed by the structure system of reconstructability analysis.

The substate labeling scheme and corresponding data structure presented in Chapter 4 should enhance efficient implementation of both the unbiased reconstruction algorithm and the greedy reconstructability algorithm by reducing the number of state

evaluations necessary to update the system model. This labeling method may also provide a more concise and understandable notation.

5.3 Future Work

K-systems analysis provides a powerful tool for discovering the important factors in determining a system's behavior. The potential of this technique is greater still. The generality of the k-system model and the power of the reconstructability algorithm make the technique potentially useful in several diverse areas of computer science. Applications of k-systems techniques in areas such as fuzzy rule bases, genetic algorithms, unsupervised learning systems and robot control can be envisioned.

The success of the directed search algorithm can likely be at least partially attributed to its compatibility with the min-max distance function $\gamma(\beta)$. The performance of this technique with different system measures remains an open question.

The directed search technique allows analysis of systems for which the substate set is too large to compute effectively, but for which the state set is computationally tractable. This difference is most pronounced when the number of variables is significantly larger than the number of values taken by the variables. For example, a system of eight binary variables has over twenty-five times as many substates as states, while there are less than six times as many substates as states for a system of eight variables which take four values each.

For a given computational platform, this is a relatively small class of problems. To be truly effective, this technique should be coupled with a method for exploring a state set which is too large to be easily enumerated. Integration of effective techniques

for state sampling and substate estimation would allow k-system reconstruction to be performed on much larger systems than is now practical.

References

[AMER92] *The American Heritage Dictionary of the English Language*, 3rd ed., s.v. "equilibrium."

[ASMU98] Asmus, Gary J. (1998). "Techniques For Resolving Incomplete Systems In K-Systems Analysis." Ph.D. diss., Louisiana State University.

[AUGE82] Auger, P. (1982). "Order, Disorder in Hierarchically Organized Systems." *International Journal of General Systems*, 8, 109-113.

[AUGE91] Auger, Pierre (1991). "Introduction to the Special Issue on Hierarchy Theory and its Applications." *International Journal of General Systems*, 18, 189-190.

[AYMO97] Aymond, Patti Iles (1997). "Parallelization of The Determination of Unbiased Reconstructions." *Advances in Systems Science and Applications*, 1997 Special Issue, 387-392.

[BIRK78] Birkeland, Peter W., and Edwin E. Larson (1978). *Putnam's Geology*, 3d ed. Oxford University Press, New York, NY.

[BOGA88] Bogart, Kenneth P. (1988). *Discrete Mathematics*. D. C. Heath and Co., Lexington, Massachusetts.

[BOOK90] Booker, L. B., D. E. Goldberg and J. H. Holland (1990). "Classifier Systems and Genetic Algorithms." In *Machine Learning: Paradigms and Methods*, ed. Jaime Carbonell, The MIT Press, Cambridge, MA.

[BROW59] Brown, David T. (1959). "A Note on Approximations to Discrete Probability Distributions." *Information and Control*, 2, 386-392.

[BUNG81] Bunge, Mario (1988). "Analogy Between Systems." *International Journal of General Systems*, 7, 221-223.

[CAVA81a] Cavallo, Roger E., and George J. Klir (1981). "Reconstructability Analysis: Overview and Bibliography." *International Journal of General Systems*, 7, 1-6.

[CAVA81b] Cavallo, Roger E., and George J. Klir (1981). "Reconstructability Analysis: Evaluation of Reconstruction Hypothesis." *International Journal of General Systems*, 7, 7-32.

[CAVA82] Cavallo, Roger E., and George J. Klir (1982). "Decision Making in Reconstructability Analysis." *International Journal of General Systems*, 8, 243-255.

[COVE91] Cover, Thomas and Joy A. Thomas (1991). *Elements of Information Theory*. John Wiley & Sons, New York.

[FLOO90] Flood, R. L. (1990). "New Domains for Analogy: Systemic Dialectics and Theory Development." *International Journal of General Systems*, 18, 113-123.

[GATL72] Gatlin, Lila L. (1972). *Information Theory and the Living System*. Columbia University Press, New York.

[GOUW96] Gouw, Deky and Bush Jones (1996). "The Interaction Concept of *K*-Systems Theory." *International Journal of General Systems*, 24, 163-169.

[GUIA77] Guiasu, Silviu (1977). *Information Theory with Applications*. McGraw-Hill, New York, NY.

[HIGA83] Higashi, Masahiko and George J. Klir (1983). "On the Notion of Distance Representing Information Closeness." *International Journal of General Systems*, 9, 103-115.

[HOFS95] Hofstadter, Douglas (1995). *Fluid Concepts and Creative Analogies*. BasicBooks, New York, NY.

[HOLL92] Holland, John H. (1992). *Adaptation in Natural and Artificial Systems*. The MIT Press, Cambridge MA.

[HOLL95] Holland, John H. (1995). *Hidden Order*. Addison-Wesley Publishing, Reading, MA.

[ILES95] Iles, Patti (*see also Aymond, Patti Iles*) (1995). "Parallelization of Reconstructability Analysis Algorithms." Ph.D. dissertation. Louisiana State University, Baton Rouge, LA.

[JONE82] Jones, Bush (1982). "Determination of Reconstruction Families." *International Journal of General Systems*, 8, 225-228.

[JONE85a] Jones, Bush (1985). "Determination of Unbiased Reconstructions." *International Journal of General Systems*, 10, 169-175.

[JONE85b] Jones, Bush (1985). "A Greedy Algorithm for a Generalization of the Reconstruction Problem." *International Journal of General Systems*, 11, 63-68.

[JONE85c] Jones, Bush (1985). "Reconstructability Analysis for General Functions." *International Journal of General Systems*, 11, 133-142.

[JONE85d] Jones, Bush (1985). "Reconstructability Considerations With Arbitrary Data." *International Journal of General Systems*, 11, 143-151.

[JONE85e] Jones, Bush (1985). "The Cognitive Content of System Substates." *Proceedings of the 1985 IEEE Workshop on Languages for Automation*, IEEE Computer Society Press, 11-13.

[JONE86] Jones, Bush (1986). "*K*-Systems Versus Classical Multivariate Systems." *International Journal of General Systems*, 12, 1-6.

[JONE89] Jones, Bush (1989). "A Program For Reconstructability Analysis." *International Journal of General Systems*, 15, 199-205.

[KOLM50] Kolmogorov, A. N. (1950). *Unbiased Estimates*. American Mathematical Society, Translation 98, 1953.

[KLIR85] Klir, George J (1985). "Cognitive Aspects of Reconstructability Analysis." *Proceedings of the 1985 IEEE Workshop on Languages for Automation*, IEEE Computer Society Press, 6-10.

[LIN87] Lin, Yi (1987). "Remarks on Analogy Between Systems." *International Journal of General Systems*, 13, 135-141.

[MARI85] Mariano, Matthew (1985). "The Problem of Resolving Inconsistency in Reconstructability Analysis." *Proceedings of the 1985 IEEE Workshop on Languages for Automation*, IEEE Computer Society Press, 14-17.

[MINS85] Minsky, Marvin (1985). *The Society of Mind*. Simon & Schuster, New York, NY.

[MITC77] Mitchell, Tom M. (1977). "Version Spaces: A Candidate Elimination Approach to Rule Learning." Proceedings of the International Joint Conference on Artificial Intelligence – 1977, 1, 305-319.

[PIER80] Pierce, John R. (1980). *An Introduction to Information Theory: Symbols, Signals, and Noise*. Dover Publications, New York, NY.

[PITT89] Pittarelli, Michael (1989). "Uncertainty and Estimation in Reconstructability Analysis." *International Journal of General Systems*, 15, 1-58.

[PITT90] Pittarelli, Michael (1990). "A Note on Probability Estimation Using Reconstructability Analysis." *International Journal of General Systems*, 18, 11-21.

[SHAN48] Shannon, C. E. (1948). "A Mathematical Theory of Communication." The Bell Systems Technical Journal, 27, 379-423.

[TABA91] Tabary, J.C. (1991). "Hierarchy and Autonomy." *International Journal of General Systems*, 18, 241-250.

[TINO95] Tinoco, Ignacio Jr., Kenneth Sauer, James C. Wang (1995). *Physical Chemistry: Principles and Applications in Biological Sciences.* Prentice Hall, Englewood Cliffs NJ.

[TRIV93] Trivedi, Sudhir K. "Reconstructability Theory for General Systems and its Application to Automated Rule Learning." Ph.D. dissertation, Louisiana State University, Baton Rouge, LA, 1993.

[VIXI97] Vixie, Kevin R. (1997). "The Generalization of Mathematical Description." *Advances in Systems Science and Applications*, 1997 Special Issue, 66-73.

Appendix. Directed Search Test Data

The performance of the directed search algorithm presented in Chapter 3 was compared to the all substate search approach using three k-system descriptions derived from experimental data. Descriptions of the experimental data sets, along with summaries of the reconstructions, are given below.

Data Set 1

Data Set 1 is an example system taken from Bush Jones' 1985 paper, "A Greedy Algorithm for a Generalization of the Reconstruction Problem," though it has also appeared in other scholarly papers on the subject. It was chosen for inclusion in the test set for two reasons. First, it is small, simple system which allows relatively straightforward illustration of the mechanisms of the algorithms, and is small enough to solve by hand. Second, the system served as a baseline test case for the implementation of the all substates search option of the greedy reconstructability algorithm.

System Description

The k-system derived from Data Set 1 includes three variables. Variables v_1 and v_2 are binary; v_3 takes three values. The function values for the resulting 12 states range from a low of 0.031 to a high of 0.133, with a mean value of 0.083. The normalized function values for the states are shown in Table 14. The states are labeled using the scheme presented in Chapter 3.

Table 14. Data Set 1 Behavior Function Values

State	$f(\alpha)$	State	$f(\alpha)$
(1,1,1)	0.079	(2,1,1)	0.091
(1,1,2)	0.088	(2,1,2)	0.072
(1,1,3)	0.083	(2,1,3)	0.037
(1,2,1)	0.031	(2,2,1)	0.109
(1,2,2)	0.052	(2,2,2)	0.128
(1,2,3)	0.097	(2,2,3)	0.133

All Substate Search Reconstruction

The all substate search reconstruction of Data Set 1 required 8 substates to achieve a final closeness value of 99.6203, though the additional information provided by new substates falls of sharply after the fifth substate is added. A summary of the reconstruction is shown in Table 15.

Table 15. Data Set 1 All Substate Reconstruction Summary

No. of Substates	Substate Selected	Substate Evaluations	State Evaluations	Iterations to Converge	System Closeness
1	220	35	84	2	47.8321
2	121	35	84	2	68.3133
3	213	35	84	2	86.1491
4	122	35	84	2	94.9979
5	023	35	84	9	97.0358
6	212	35	84	8	98.2822
7	222	35	84	19	99.2118
8	211	35	84	17	99.6203

Directed Search Reconstruction

The directed search reconstruction of Data Set 1 selected the same substates in the same order as the all substate search above. A summary of the reconstruction is shown in Table 16.

Table 16. Data Set 1 Directed Search Reconstruction Summary

No. of Substates	Substate Selected	Substate Evaluations	State Evaluations	Iterations to Converge	System Closeness
1	220	8	50	2	47.8321
2	121	8	48	2	68.3133
3	213	8	48	2	86.1491
4	122	8	48	2	94.9979
5	023	8	48	9	97.0358
6	212	6	38	8	98.2822
7	222	6	38	19	99.2118
8	211	6	38	17	99.6203

Data Set 2

Data Set 2 is based on survey data from a study of calorie intake for persons from various demographic categories. Variable 1 takes values from six income levels. Variable 2 is one of three residence zones, and Variable 3 is subject age. Variable 3 was clustered into two categories from original data of age in years.

System Description

The k-system derived from Data Set 2 includes three variables. Variable v_1 takes six values, v_2 takes three values, and v_3 is binary. The function values for the resulting 36 states range from a low of 0.0165 to a high of 0.0416, with a mean value of

0.028. The normalized function values for the states are shown in Table 17. The states are labeled using the scheme presented in Chapter 3.

Table 17. Data Set 2 Behavior Function Values

State	$f(\alpha)$	State	$f(\alpha)$
(1,1,1)	0.03639	(4,1,1)	0.033479
(1,1,2)	0.024953	(4,1,2)	0.026649
(1,2,1)	0.027778	(4,2,1)	0.026334
(1,2,2)	0.023122	(4,2,2)	0.027778
(1,3,1)	0.027778	(4,3,1)	0.034167
(1,3,2)	0.023418	(4,3,2)	0.022562
(2,1,1)	0.027609	(5,1,1)	0.032909
(2,1,2)	0.01654	(5,1,2)	0.027778
(2,2,1)	0.028345	(5,2,1)	0.031464
(2,2,2)	0.022546	(5,2,2)	0.021075
(2,3,1)	0.035799	(5,3,1)	0.03052
(2,3,2)	0.021218	(5,3,2)	0.023578
(3,1,1)	0.032056	(6,1,1)	0.031944
(3,1,2)	0.018427	(6,1,2)	0.023714
(3,2,1)	0.028417	(6,2,1)	0.041573
(3,2,2)	0.022618	(6,2,2)	0.034287
(3,3,1)	0.029384	(6,3,1)	0.040869
(3,3,2)	0.023322	(6,3,2)	0.019603

All Substate Search Reconstruction

The all substate search reconstruction of Data Set 2 required 19 substates to achieve a final closeness value of 99.5543. A closeness value of 95 was achieved with around half as many states. A summary of the reconstruction is shown in Table 18.

Directed Search Reconstruction

The directed search reconstruction of Data Set 2 required 20 substates to achieve a final closeness value of 99.5693. A closeness value of around 95 was

attained with the addition of twelve states, somewhat later than with the all substates technique. A summary of the reconstruction is shown in Table 19.

Table 18. Data Set 2 All Substate Search Reconstruction Summary

No. of Substates	Substate Selected	Substate Evaluations	State Evaluations	Iterations to Converge	System Closeness
1	002	83	252	2	52.7667
2	620	83	252	2	68.5613
3	631	83	252	4	74.8125
4	212	83	252	7	79.5168
5	312	83	252	7	82.176
6	010	83	252	12	85.9685
7	231	83	252	11	88.818
8	422	83	252	10	91.1113
9	431	83	252	12	93.2052
10	211	83	252	13	95.1892
11	632	83	252	12	96.0425
12	111	83	252	14	96.8509
13	501	83	252	10	97.6486
14	612	83	252	16	98.0867
15	522	83	252	15	98.3957
16	232	83	252	15	98.7125
17	421	83	252	17	99.0254
18	512	83	252	21	99.2688
19	410	83	252	32	99.5543

Table 19. Data Set 2 Directed Search Reconstruction Summary

No. of Substates	Substate Selected	Substate Evaluations	State Evaluations	Iterations to Converge	System Closeness
1	002	12	214	2	52.7667
2	620	10	130	2	68.5613
3	631	8	112	4	74.8125
4	212	8	112	7	79.5168
5	312	6	94	7	82.176
6	021	8	124	8	85.5457
7	211	8	112	2	87.0893
8	131	9	105	2	88.8252
9	632	9	105	9	90.5204
10	422	9	105	10	91.9842
11	512	6	94	11	93.6929
12	110	8	112	10	94.822
13	412	6	94	12	96.1166
14	331	6	94	9	96.9254
15	521	6	94	13	97.8181
16	231	6	94	11	98.4189
17	531	6	94	11	98.76
18	522	6	94	14	99.0571
19	232	6	94	15	99.3606
20	421	6	94	17	99.5693

Data Set 3

Data Set 3 was provided by Gary P. Shaffer of Southeastern Louisiana University. The data are taken from an oil spill bioremediation study. This system was chosen for inclusion because of its significantly larger size than the other two systems, and the large number of function values near zero. This system proved the most difficult to reconstruct using simpler directed search techniques than the final version presented here.

System Description

The k-system derived from Data Set 3 includes five variables. Variables v_1, v_3 and v_4 are binary; v_2 takes five values and v_5 takes three. The function values for the resulting 120 states range from a low very near zero to a high of 0.03067, with a mean value of 0.0083. The normalized function values are shown in Table 20.

All Substate Search Reconstruction

The all substate search reconstruction of Data Set 3 required 39 substates to achieve a final closeness value of 99.5009. A closeness value of around 95 was attained with the addition of only eighteen states, similar to the results with the other systems. A summary of the reconstruction is shown in Table 21.

Directed Search Reconstruction

The directed search reconstruction of Data Set 3 required 40 substates to achieve a final closeness value of 99.5358. The closeness value exceeded 95 with the addition of the nineteenth substate, slightly earlier than the all substates search. A summary of the reconstruction is shown in Table 22.

Table 20. Data Set 3 Behavior Function Values

State	f(α)	State	f(α)	State	f(α)
(1,1,1,1,1)	0.016921	(1,4,1,2,2)	6.20953e-006	(2,2,2,1,3)	0.00334694
(1,1,1,1,2)	0.00805221	(1,4,1,2,3)	6.20953e-006	(2,2,2,2,1)	0.0104196
(1,1,1,1,3)	0.0171073	(1,4,2,1,1)	6.20953e-006	(2,2,2,2,2)	0.0123694
(1,1,1,2,1)	0.00807239	(1,4,2,1,2)	6.20953e-006	(2,2,2,2,3)	0.00790473
(1,1,1,2,2)	0.0151512	(1,4,2,1,3)	6.20953e-006	(2,3,1,1,2)	0.0118975
(1,1,1,2,3)	0.0162348	(1,4,2,2,1)	6.20953e-006	(2,3,1,1,2)	0.0116511
(1,1,2,1,1)	0.0164863	(1,4,2,2,2)	6.20953e-006	(2,3,1,1,3)	0.0165267
(1,1,2,1,2)	0.0104351	(1,4,2,2,3)	6.20953e-006	(2,3,1,2,1)	0.00786903
(1,1,2,1,3)	0.0182871	(1,5,1,1,1)	0.0212055	(2,3,1,2,2)	0.0119409
(1,1,2,2,1)	0.0156791	(1,5,1,1,2)	6.20953e-006	(2,3,1,2,3)	6.20953e-006
(1,1,2,2,2)	6.20953e-006	(1,5,1,1,3)	6.20953e-006	(2,3,2,1,1)	0.0138266
(1,1,2,2,3)	0.0109101	(1,5,1,2,1)	0.0306751	(2,3,2,1,2)	0.012922
(1,2,1,1,1)	0.0155362	(1,5,1,2,2)	6.20953e-006	(2,3,2,1,3)	6.20953e-006
(1,2,1,1,2)	0.0120341	(1,5,1,2,3)	0.00601393	(2,3,2,2,1)	0.00890136
(1,2,1,1,3)	0.00303336	(1,5,2,1,1)	0.025397	(2,3,2,2,2)	0.00760667
(1,2,1,2,1)	0.0263377	(1,5,2,1,2)	6.20953e-006	(2,3,2,2,3)	0.0130431
(1,2,1,2,2)	0.0178493	(1,5,2,1,3)	6.20953e-006	(2,4,1,1,1)	6.20953e-006
(1,2,1,2,3)	6.20953e-006	(1,5,2,2,1)	0.0225095	(2,4,1,1,2)	6.20953e-006
(1,2,2,1,1)	0.0169179	(1,5,2,2,2)	6.20953e-006	(2,4,1,1,3)	6.20953e-006
(1,2,2,1,2)	0.0084077	(1,5,2,2,3)	0.0172004	(2,4,1,2,1)	6.20953e-006
(1,2,2,1,3)	0.0117888	(2,1,1,1,1)	0.0186596	(2,4,1,2,2)	6.20953e-006
(1,2,2,2,1)	0.0134002	(2,1,1,1,2)	0.0108729	(2,4,1,2,3)	6.20953e-006
(1,2,2,2,2)	0.0101867	(2,1,1,1,3)	0.0147787	(2,4,2,1,1)	6.20953e-006
(1,2,2,2,3)	6.20953e-006	(2,1,1,2,1)	0.0142509	(2,4,2,1,2)	6.20953e-006
(1,3,1,1,1)	0.0100004	(2,1,1,2,2)	0.0144992	(2,4,2,1,3)	6.20953e-006
(1,3,1,1,2)	0.00396478	(2,1,1,2,3)	0.0187528	(2,4,2,2,1)	6.20953e-006
(1,3,1,1,3)	0.0147632	(2,1,2,1,1)	6.20953e-006	(2,4,2,2,2)	6.20953e-006
(1,3,1,2,1)	0.0112579	(2,1,2,1,2)	0.0160516	(2,4,2,2,3)	6.20953e-006
(1,3,1,2,2)	0.0135212	(2,1,2,1,3)	0.0162069	(2,5,1,1,1)	0.017573
(1,3,1,2,3)	0.0104662	(2,1,2,2,1)	0.0118478	(2,5,1,1,2)	6.20953e-006
(1,3,2,1,1)	0.00936086	(2,1,2,2,2)	6.20953e-006	(2,5,1,1,3)	6.20953e-006
(1,3,2,1,2)	0.0112951	(2,1,2,2,3)	6.20953e-006	(2,5,1,2,1)	0.0252417
(1,3,2,1,3)	0.0118354	(2,2,1,1,1)	0.0102799	(2,5,1,2,2)	6.20953e-006
(1,3,2,2,1)	0.00758388	(2,2,1,1,2)	0.0091901	(2,5,1,2,3)	6.20953e-006
(1,3,2,2,2)	0.0135181	(2,2,1,1,3)	0.0124718	(2,5,2,1,1)	0.0208454
(1,3,2,2,3)	0.00927083	(2,2,1,2,1)	0.00792646	(2,5,2,1,2)	6.20953e-006
(1,4,1,1,1)	6.20953e-006	(2,2,1,2,2)	0.00362015	(2,5,2,1,3)	6.20953e-006
(1,4,1,1,2)	6.20953e-006	(2,2,1,2,3)	0.0115125	(2,5,2,2,1)	0.0104631
(1,4,1,1,3)	6.20953e-006	(2,2,2,1,1)	0.0145893	(2,5,2,2,2)	6.20953e-006
(1,4,1,2,1)	0.0229132	(2,2,2,1,2)	0.00828041	(2,5,2,2,3)	6.20953e-006

Table 21. Data Set 3 All Substate Search Reconstruction Summary

No. of Substates	Substate Selected	Substate Evaluations	State Evaluations	Iterations to Converge	System Closeness
1	04000	647	3720	2	26.5093
2	05002	647	3720	2	41.9305
3	14121	647	3720	1868	55.0497
4	05013	647	3720	2	63.2965
5	05001	647	3720	2	69.0824
6	01222	647	3720	2	72.9674
7	12023	647	3720	2	76.9591
8	25023	647	3720	2	81.0638
9	21211	647	3720	2	83.1605
10	21223	647	3720	2	85.2879
11	23123	647	3720	2	87.4469
12	23213	647	3720	2	89.6385
13	12120	647	3720	1265	90.8719
14	01000	647	3720	1492	92.3408
15	12113	647	3720	1055	93.049
16	25221	647	3720	1178	93.7841
17	22213	647	3720	1053	94.4452
18	22122	647	3720	1053	95.0713
19	13112	647	3720	1050	95.6511
20	11012	647	3720	1168	96.0831
21	12001	647	3720	838	96.5883
22	11121	647	3720	1109	96.9602
23	03113	647	3720	1100	97.287
24	15223	647	3720	1051	97.5895
25	15123	647	3720	1004	97.8108
26	15121	647	3720	1140	98.0368
27	00220	647	3720	1284	98.2098
28	21112	647	3720	1090	98.3743
29	13022	647	3720	1101	98.5305
30	02212	647	3720	963	98.6657
31	25111	647	3720	1080	98.7992
32	23121	647	3720	900	98.905
33	22121	647	3720	910	99.0155
34	23223	647	3720	1093	99.1148
35	13201	647	3720	1063	99.1957
36	22211	647	3720	923	99.2639
37	22222	647	3720	1090	99.3353
38	23210	647	3720	1123	99.4033
39	10200	647	3720	1169	99.5009

Table 22. Data Set 3 Directed Search Reconstruction Summary

No. of Substates	Substate Selected	Substate Evaluations	State Evaluations	Iterations to Converge	System Closeness
1	04000	600	9330	2	26.5093
2	05002	245	2066	2	41.9305
3	14121	133	930	1868	55.0497
4	05013	142	1058	2	63.2965
5	05001	88	666	2	69.0824
6	25220	14	301	1276	71.0419
7	01222	70	500	421	74.9635
8	12023	52	424	427	78.9941
9	25023	38	381	986	81.7985
10	21211	29	343	2	83.8952
11	21223	24	329	2	86.0226
12	23123	19	315	2	88.1816
13	23213	14	301	2	90.3732
14	12120	14	301	1265	91.6066
15	12113	10	268	682	92.4706
16	22213	10	268	684	93.2775
17	22122	10	268	682	94.0404
18	13112	10	268	679	94.747
19	15123	10	268	639	95.1241
20	15121	14	292	1141	95.3502
21	01000	27	580	1449	96.1623
22	11012	18	316	1168	96.5943
23	11121	14	292	1109	96.9662
24	05223	14	292	1019	97.194
25	12001	17	332	889	97.7036
26	03113	14	292	1110	98.0366
27	21112	10	268	1049	98.1825
28	11223	10	268	1065	98.3429
29	22211	10	268	980	98.4589
30	25111	10	268	1090	98.5678
31	00021	23	495	1506	98.7519
32	25121	10	268	1196	98.8357
33	23222	14	301	1080	98.9396
34	23210	14	301	1216	99.0339
35	13022	14	292	1191	99.1571
36	22223	10	268	1088	99.2287
37	02212	18	316	1189	99.3415
38	15201	14	292	947	99.4305
39	21221	10	268	1241	99.4661
40	01102	17	332	829	99.5358

Vita

Christopher W. Branton received his bachelor of science degree in Computer Science from Louisiana State University in 1992. He joined the doctoral program at L.S.U. in 1994 as a Board of Regents Graduate Fellow.

He is currently employed as a systems analyst and software designer for Innovative Emergency Management. His work there includes design and construction of hazard and emergency response simulation software systems.

Apart from k-systems analysis, his research interests include artificial intelligence, machine learning, complex systems and theory of computation. He has been a member of the Association for Computing Machinery since 1992. His degree of Doctor of Philosophy will be awarded at the December Commencement 1998.

DOCTORAL EXAMINATION AND DISSERTATION REPORT

Candidate: Christopher W. Branton

Major Field: Computer Science

Title of Dissertation: Directed Search in K-System Reconstruction

Approved:

Major Professor and Chairman

Dean of the Graduate School

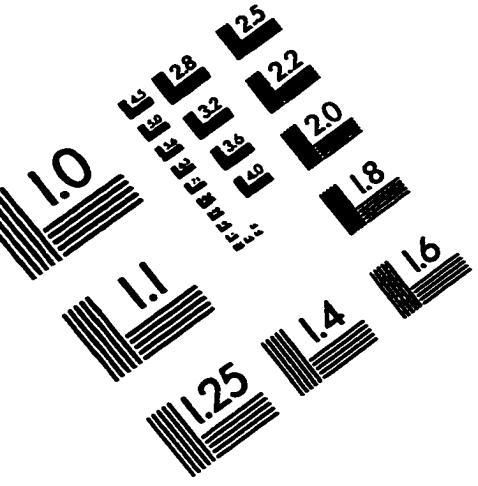
EXAMINING COMMITTEE:

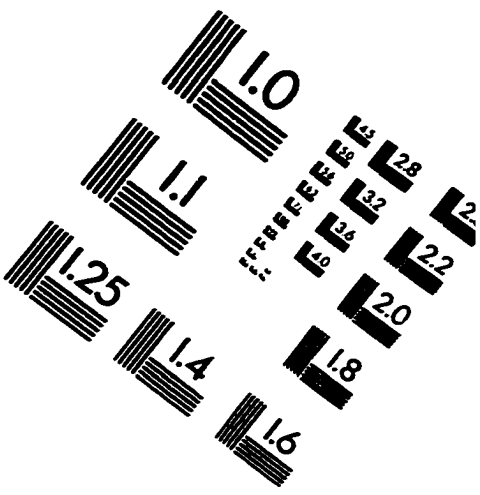
MT

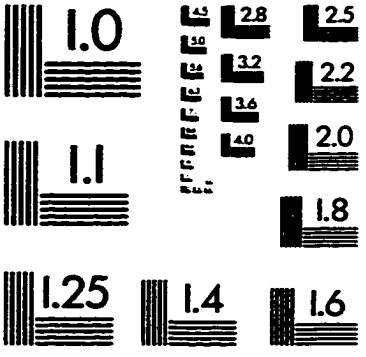
Date of Examination:

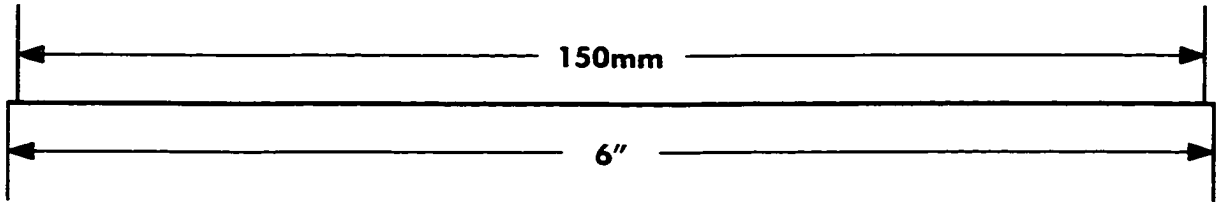
November 3, 1998

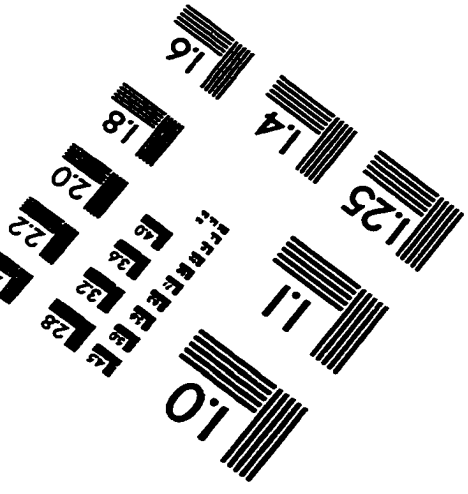
IMAGE EVALUATION
TEST TARGET (QA-3)

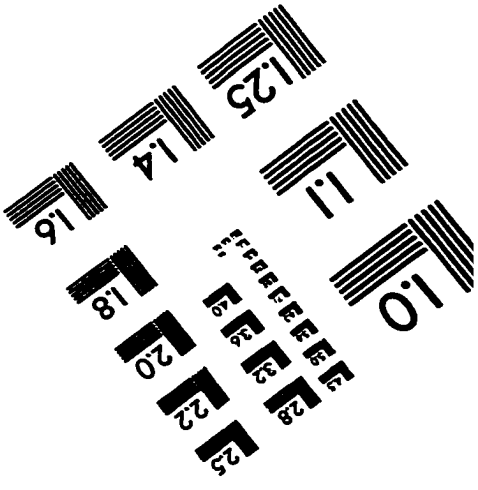












APPLIED IMAGE, Inc
1653 East Main Street
Rochester, NY 14609 USA
Phone: 716/482-0300
Fax: 716/288-5989