PERGAMON Journal of the Franklin Institute 338 (2001) 571–582

Journal *of* The Franklin Institute

www.elsevier.com/locate/jfranklin

# Information fusion techniques for pattern analysis in large sensor data networks: an overview

S.S. Iyengar[1,*], Bush Jones

*Department of Computer Science, Louisiana State University, Coates Hall, Room 298, Baton Rouge, LA 70803-4020, USA*

Received 14 May 2001

**Abstract**

Many defense and civilian applications require algorithms with great correctness and precision, and work with limited or incomplete information. This paper provides an overview of a new data analysis technique for many of these applications. 

*Keywords:* Pattern analysis; Sensor data; Networks; Algorithms

## 1. Introduction

Modern battle spaces have become technologically very large and complex. Information must be collected and put into comprehensible form. Algorithms are needed to study postulated battle space environments to reduce them into fundamental information components. Then algorithms are needed to provide real time elemental information in a concise format in actual deployment. Algorithms must adapt to new patterns in the data and provide feedback to the collection process. Military applications require algorithms with great correctness and precision, and work with limited or incomplete information (Fig. 1).

*Corresponding author. Tel.: +1-225-388-1495; fax: +1-225-388-1465.

*E-mail address:* iyengar@bit.csc.lsu.edu (S.S. Iyengar).

[1] This Research is partially funded by ONR grant No. N000014-91-J1306 and by DARPA/ITO through the SensIT program.

0016-0032/01/$20.00 
PII: S0016-0032(01)00024-2

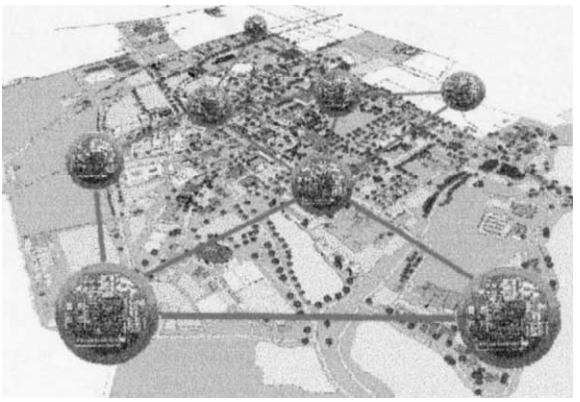

Fig. 1. Distributed MEMS (micro-electro-mechanical systems) sensors.

Traditional fusion analysis is assumption driven in the sense that a hypothesis is validated against the data. However the proposed *k*-systems analysis is pattern driven in the sense that patterns are automatically extracted from the data. The *k*-systems theory studies the relationship between parts and wholes in systems. It uses maximum entropy mathematics, and avoids contamination by a model, or the introduction of extraneous information. It brings correctness to the study of systems that was never before possible. *k*-systems has many maximum entropy algorithms, which are broadly applicable in many disciplines.

This *k*-systems theory was invented in 1985 by Bush Jones (one of the PIs of this investigation), and has undergone development over the last 14 years. Most problems in *k*-systems are solved, and there exists a 10,000-line program for doing this analysis. However, there remains a problem in information fusion that is still only partially solved. This problem is to fuse information from subsystems into total system information in its most general case. In this process many measures of system interrelationships can be produced as well, as a global system description. This leads to better understanding and predictability of the whole system.

This paper focuses on proposing a data analysis method and then proves this data analysis technique to detect, recognize and identify targets of interests from sensor data in its most general case. Currently, we are developing an interactive software visualization tool for a general use.

## 2. Identification and significance of the problem

Advances in sensor technology have led to better and cheaper sensors. These advances beget more complex tactical deployment of sensors. Such deployment requires new and sophisticated techniques for information analysis and fusion of information. The *k*-systems theory is a new branch of analysis that studies the relationships of parts and wholes in systems. In particular, it treats the fusion of information from multiple subsystems into an overall system.

*The problem*: Consider the following scenario. Sensors are grouped into nets or nodes, and these nets are deployed into "regions of darkness" in a tactical field. We refer to these as pockets on the field, and these pockets may be disjoint or overlapping. Each sensor provides information to one or more pockets, and information from the pockets fuses to form total system information (Fig. 2).

At the lowest level, the sensors provide raw data in the form of real numbers. The pockets are mathematical functions of this raw data. That is, the pockets **fuse** the real numbers into meaningful functions. Now, these pocket functions are themselves **fused** into a master function, which is descriptive of the whole scenario. This master function may be a decision table, or other concise representation of relevant decision criteria. It is particularly important that data in its most elemental form be incorporated at each level, screening out extraneous information. $k$-systems work with data "information" as developed in the Information Theory of Claude Shannon.

Allowing a sensor to be contained in more than one pocket enhances field coverage reliability; but, more importantly, mathematical accuracy of the total system is enhanced. This is a multi-function scenario where there is a function associated with each pocket, and an overall function that fuses the information from the pockets. This is referred to as the identification problem in $k$-systems theory. In $k$-systems theory, each sensor would be represented by a variable $s_i$, which is discrete or continuous. There is a real function (discrete or continuous) of the pocket variables for the $m$th pocket, ${}^m k(.)$, which would represent the information of the $m$th pocket subsystem. It is the fusion of the ${}^m k(.)$ functions of the subsystems into an overall system function $k(.)$, that is critical to the success of the identification problem. In $k$-systems theory, this is accomplished by the **maximum entropy** extension of the subsystems ${}^m k(.)$ to the overall system function $k(.)$. The ability to do this extension, for deterministic functions, ${}^m k(.)$ and $k(.)$, as well as probabilistic functions, is unique to $k$-systems. The maximum entropy extension is an unbiased estimator. It neither loses any information content nor adds any extraneous information content to the ${}^m k(.)$—a claim unique to $k$-systems.

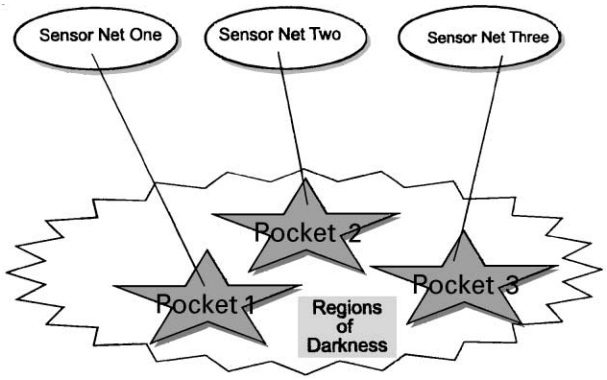


Fig. 2. Deployment of sensor clusters to detect regions of targets.

### 2.1. Introduction to k-systems

This new $k$-systems theory studies the relationships between the parts and wholes of systems. It uses the mathematics of entropy, and does not depend on an assumed model, and never introduces information extraneous to the data. $k$-systems is an analysis that is as correct as the data. There are two basic problems broached by $k$-systems.

#### 2.1.1. Reconstruction problem

Given information on the whole system, determine the most important subsystems and how they are important. Sensor problems require that only the most fundamental units of information be employed. The reconstruction problem solution allows these elements to be identified in postulated battlespaces before actual deployment in real time.

*Problem in context of investigation*: Given information on the subsystems, determine information on the whole system. This is the real time step wherein the elemental units of information are examined to yield global system information in a concise and meaningful form.

The reconstruction problem has been solved for any system described by any real valued function. The identification problem has only been solved for probabilistic functions. Part of this proposal is to solve the identification problem by entropy mathematics for any real valued functions.

*k-systems (new technique developed by Dr. Jones)*: Sensor data raw information is in the form of real numbers, and subsequently in the form of real valued functions. These functions are not necessarily probabilistic or [0,1] functions. This is where traditional information analysis comes to a crashing halt, but $k$-systems theory allows information theory to proceed by construction of a isomorphic or mirror system in [0,1] space.

Before entropy mathematics is employed on a system, the system undergoes a scalar transformation to reduce it to a [0,1] function system. If the original system functions were probability functions, the transformation leaves them invariant. In either case, no information is lost, and we say the systems are isomorphic. We call this a mapping from a real world $g$-system to a $k$-system.

We first review the definition of $g$-system and $k$-system. Associated with a system is a finite set of variables $\{v_i\}$ which take values from finite sets $\{0, 1, 2 \ldots n_i\}$. Each nonempty subset of the variables identifies one subsystem of the system. States $\{\acute{a}\}$ and substates $\{\hat{a}\}$ of the system are determined by particular value assignments to the variables.

Associated with the overall system is a behavior function $f(.)$ If $A$ is the set of all states of the system, and $R^+$ is a set of positive real numbers, then $f : A \circledR R^+$ is a function which represents information associated with system states. We define

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

as a set of functions $\{^m f(.)\}$, one for each subsystem. These functions are simply marginal totals (marginal distributions if the functions are probabilistic). We also define a parameter:

$$\tau = \sum_{\alpha/\tau} f(\alpha)$$

This is simply a sum of the system function over all system states.

**Definition.** A $g$-system is atuple:

$$(\tau, \{v_i\}, \{\propto\}, \{\beta\}, f(.), \{^m f(.)\})$$

(1) $\tau$ is a scaling parameter
(2) $\{v_i\}$ is a set of variables
(3) $\{\propto\}$ is a set of states
(4) $\{\beta\}$ is a set of substates
(5) $f(.)$ is a function on $\{\propto\}$
(6) $\{^m f(.)\}$ are functions of $\{\beta\}$

The $k$-system is directly obtained by a simple scaling. The function $f(.)$ is a measure of some type of information on system states and commonly has some type of units associated with it (e.g. temperature-degrees centigrade, etc.). We remove such units from the system and accomplish the first part of our

$$k(\alpha) = f(\alpha)/\tau$$

transformation by the normalization:

This is done for every $\propto$. Then clearly,

$$\sum_{\alpha} k(\alpha) = 1$$

Next we define the functions $^m k(.)$ for each subsystem of the original system as

We now define a $k$-system

**Definition.** We define a $k$-System as:

$$(\tau, \{v_i\}, \{\propto\}, \{\beta\}, k(.), \{^m k(.)\})$$

(1) $\tau$ is a scaling parameter
(2) $\{v_i\}$ is a set of variables
(3) $\{\propto\}$ is a set of states
(4) $\{\beta\}$ is a set of substates
(5) $k(.)$ is a function on $\{\propto\}$
(6) $\{^m k(.)\}$ are functions on $\{\beta\}$

The set of equations defined above describes the system effects and interactions. These equations are similar to the equations of Analysis of Variance (ANOVA) in statistics. However, we do not assume a model as is done in ANOVA, but we are

able to solve them directly by a maximum entropy algorithm thereby avoiding contamination by a model. These equations are linear, but the effects and interactions they describe can be nonlinear. We obtain a system function of nonlinear complexity, and our method is valid for any degree of nonlinearity. This is not the case with ANOVA, which describes only a linear function.

### 2.2. A new paradigm: maximum entropy algorithm

The maximum entropy algorithm to accomplish the extension from the ${}^{m}k(.)$ functions to the $k(.)$ function was given by Jones [1]. Also **unique** to the algorithm by Jones is the ability to accomplish the extension from only partial information on the subsystem functions ${}^{m}k(.)$. First, the equations are "partitioned" into classes, which normalizes the equations (details of this normalization can be found in Jones [1]). Then the algorithm works by iteratively scaling one side of the equations (the unknown $k(.)$ values) to satisfy the other sides (the known ${}^{m}k(.)$ values). The technique has been proven to converge, and to converge to the unique maximum entropy solution. Below is a description of the algorithm.

Let $a_i = {}^{m}k(.)$ where the $v_i$ that ${}^{m}k$ is a function of, take specific values. Denote the real values of $k(.)$ by $k_{ij}$. We can associate a set or subset of linear equations that capture system.

Interactions for any set have given $a_i$ as

$$\sum_j k_{ij} = a_i$$

Now partition the equations.

Then we solve these equations by the following iterative maximum entropy algorithm:

(1) Initialize $k(.)$ to a flat distribution (all $k_{ij}$ equal to mean).
(2) For all $i$:
New $k_{ij}$ = old $k_{ij}$ $(a_i/a_i')$ for every $j$: where $a_i'$ is derived from the current estimate of $k_{ij}$ and $a_i$ is a true value.
(3) Convergence Test:
New $k_{ij}$ − Old $k_{ij}$ less than tolerance for all $ij$?
If satisfied, stop
If not satisfied, go to (2)

### 2.3. Reconstruction algorithm

We examine independent ${}^{m}k(.)$ or $\hat{a}$ and determine which have the greatest information content towards the determination of the overall function $k(.)$. This is done as follows:

1. Initialize an approximation to $k(.)$, call it $k'(.)$, to a state of complete entropy.

2. Pick that independent ${}^{m}k(.)$ or $\hat{a}$ that does the most to bring $k'(.)$ toward $k(.)$; for example, by taking a maximum entropy approximation to $k(.)$ using each ${}^{m}k(.)$ in turn, and measuring the information distance between $k(.)$ and $k'(.)$ – picking the ${}^{m}k(.)$ or $\hat{a}$ that yields the smallest distance. (A shortcut algorithm exists to do this.)

3. Update $k'(.)$ with the information. Repeat step 2, picking new ${}^{m}k(.)$ or $\hat{a}$ until the $\{\hat{a}\}$ or $\{{}^{m}k(.)\}$ functions describe the $k(.)$ function within the desired information distance.

### 2.3.1. Time series illustration

Suppose we view a target identification function over 16 time periods. As the function increases, we become more certain of a target identification. Suppose we also view three sensors as we view the target identification function. These observations could come from field experimentation, simulations, or be postulated. We are interested to see if we can identify a target from the sensors alone for one time period of lookahead. That is, based on the current values of the sensors, can we predict the target will be present in the next time period.

We have 16 total observations and three variables besides time. These observations are given in Table 1. The program would first cluster each continuous sensor variable into an optimal number of clusters. We do our analysis for 1 time period look ahead. Output from analysis is now given for a system accuracy of 98%.

Recall that these sensor values actually represent regions—the user would refer to the cluster maps in the output to see the ranges for each sensor. The factors (sensor values) output is to view as prevailing conditions, and the effect is to be viewed as 1 time period ahead. Two factors (the first and fourth) are particularly good conditions for designating a target for the next time period since 1 time period after they occur the target value is experiencing its highest values. In the future when these conditions approximately hold, a target can be expected in the next time period. In order to determine the extent to which they should approximately hold, the user should examine the cluster maps for each of the variables concerned. Many other measures such as interactions can be obtained by $k$-systems analysis. This shows the power of $k$-systems in the reconstruction problem to extract elemental information in system behavior.

## 2.4. Identification problems and proposed information fusion

Given information on the subsystems, the identification problem is to identify the overall system. For example, suppose we have a probabilistic system with three variables: $v_1$, $v_2$, and $v_3$. We would like to know the behavior of this system ${}^{123}f(v_1, v_2, v_3)$, but we only have information on its subsystems or marginal distributions, and even this information is incomplete. This can occur due to the cost of viewing the entire system, or because the total system information is just not available. For instance, we know the probability of $(v_1 = 0,\ v_3 = 2)$, but not of $(v_1 = 0,\ v_2 = 1,\ v_3 = 0)$.

Table 1

| Time | Sensor1 | Sensor2 | Sensor3 | Target value |
|---|---|---|---|---|
| 1 | 8.2 | 4.2 | 20.6 | 11.0 |
| 2 | 7.8 | 4.1 | 23.9 | 12.1 |
| 3 | 8.1 | 4.5 | 21.1 | 12.3 |
| 4 | 9.6 | 4.4 | 21.7 | 11.6 |
| 5 | 8.4 | 4.2 | 27.1 | 12.4 |
| 6 | 7.8 | 6.6 | 22.0 | 12.0 |
| 7 | 8.1 | 6.9 | 24.1 | 17.4 |
| 8 | 8.2 | 7.1 | 28.4 | 22.7 |
| 9 | 9.2 | 5.2 | 22.1 | 11.6 |
| 10 | 9.3 | 4.7 | 23.8 | 12.2 |
| 11 | 9.7 | 4.3 | 28.7 | 12.6 |
| 12 | 9.6 | 6.1 | 22.0 | 14.2 |
| 13 | 9.8 | 6.9 | 24.2 | 19.4 |
| 14 | 9.9 | 7.2 | 28.9 | 24.7 |
| 15 | 9.8 | 7.1 | 28.8 | 21.2 |
| 16 | 9.4 | 5.2 | 27.9 | 14.1 |

| | | |
|---|---|---|
| Number of controlling factors: | | 5 |
| Number of distinct variables in controlling factors: | | 3 |
| System accuracy of controlling factors: | | 98.1% |
| Average error for the reproduced system: | 0.30 | |
| Maximum error for the reproduced system: | 0.61 | |

| Factor | Isolated effect on flat system (effect value) | System accuracy after adding this factor (%) |
|---|---|---|
| SENSOR2 = 6.843<br>SENSOR3 = 24.000 | 50.72% (23.7) | 86.08 |
| SENSOR2 = 6.843<br>SENSOR3 = 21.583 | 17.01% (18.4) | 91.81 |
| SENSOR1 = 9.589<br>SENSOR3 = 28.300 | 1.27% (15.4) | 95.12 |
| SENSOR1 = 9.589<br>SENSOR2 = 6.843 | 30.90% (20.6) | 97.58 |
| SENSOR2 = 4.533<br>SENSOR3 = 28.300 | −16.69% (13.1) | 98.11 |

The situation is given below:

| $v_1$ | $v_2$ | $^{12}f(.)$ |
|---|---|---|
| 1 | 1 | 0.25 |

| $v_2$ | $v_3$ | $^{23}f(.)$ |
|---|---|---|
| 1 | 0 | 0.14 |
| 1 | 1 | 0.18 |
| 1 | 2 | 0.23 |

| $v_1$ | $v_3$ | $^{13}f(.)$ |
|---|---|---|
| 0 | 1 | 0.14 |
| 0 | 2 | 0.18 |
| 1 | 0 | 0.20 |
| 1 | 1 | 0.20 |
| 1 | 2 | 0.17 |

we can readily determine $^{123}f(.)$. The maximum entropy algorithm given earlier can solve this problem with probabilistic functions, which for this example readily yields:

| $v_1$ | $v_2$ | $v_3$ | $^{123}f(.)$ |
|---|---|---|---|
| 0 | 0 | 0 | 0.079 |
| 0 | 0 | 1 | 0.088 |
| 0 | 0 | 2 | 0.083 |
| 0 | 1 | 0 | 0.031 |
| 0 | 1 | 1 | 0.052 |
| 0 | 1 | 2 | 0.097 |
| 1 | 0 | 0 | 0.091 |
| 1 | 0 | 1 | 0.072 |
| 1 | 0 | 2 | 0.037 |
| 1 | 1 | 0 | 0.109 |
| 1 | 1 | 1 | 0.128 |
| 1 | 1 | 2 | 0.133 |

The problem above can only be solved by Jones' algorithm, which allows a solution for incomplete information. This algorithm is very useful in pattern recognition.

We desire solutions when the functions involved are general real valued functions (as treated in the reconstruction problem earlier). There are no known solutions to this problem, despite its obvious importance. This is especially true as systems and subsystems become larger and more complex.

For one example, consider a real valued system function $f(x_1, x_2, \ldots x_n)$ over a large number of variables. This can be a pattern recognition function, decision function, etc. In most real applications, we can observe $f(.)$ only over subsets of the variables (*or indeed only over certain substates)*, and this may vary dynamically in time. We allow that these subsets may not be disjoint. We can view the observations of $f(.)$ over a subset as a subsystem function. For instance, observations of $f(.)$ over $(x_1, x_2)$ as $^{12}f(x_1, x_2)$. From subsystem functions, we would then want to characterize $f(.)$.

Another example would be where there are distinct subsystem functions over possibly intersecting subsets of the variables, and we desire to coalesce these into a unique system function with meaningful interpretations.

The solutions to these problems are not readily apparent, but the authors believe that modifications of the $k$-system framework will resolve them. Effects and interactions of the subsystems might be used to build a global system function by maximum entropy mathematics. In so doing, many useful measures descriptive of the total system would be generated as well as the global system function or functions. This would give a better understanding of the operation of the system and subsystems, and would allow greater precision in predicting or identifying system and subsystem behaviors. Problems of incomplete data, continuous as well as discrete data, time variant data, contradictions in data, and inconsistencies in functions would be treated.

## 3. New framework and conclusions

The objectives of this new frame work are to develop and prove this new data fusion techniques to detect, recognize and identify targets of interest from sensor data in its general case. In particular, an interactive software and visualization toolkit would be developed and tested on many defense installations. There are number of interesting research issues that would be investigated in this proposal are sketched in Figure $F^*$. More importantly, 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 that searching all substates should be of no 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 all the substates search, through 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 analogy. The concept of a fitness landscape is a central element of the study of complex systems and machine learning. 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 is responsible for its shape Figs. 3 and 4.

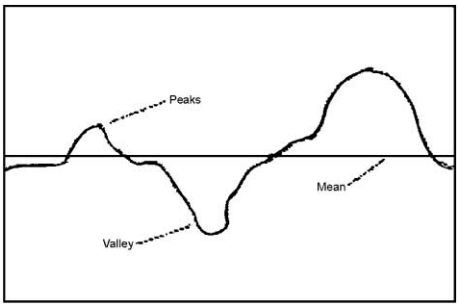


Fig. 3. Simple fitness landscape. Source: Directed search in $k$-system reconstructions, Ph.D. Thesis, Christopher W. Branton, LSU.

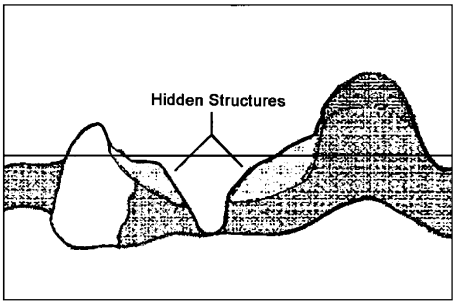


Fig. 4. Landscape showing underlying structure. Source: Directed search in $k$-system reconstructions, Ph.D. Thesis, Christopher W. Branton, LSU.

In order to validate our hypothesis we will determine appropriate optimization techniques to examine different reconstruction methodologies through a simulation technique. The simulation will be carried out using a hypothetical but a realizable tactical database from one of the defense agencies.

## Reference

[1] B. Jones, Determination of unbiased reconstructions. Internat. J. Gen. Systems 10 (1984) 303–305.

## Further reading

B. Jones, A Greedy algorithm for a generalization of the reconstruction problem, Internat. J. Gen. Systems 11 (1985) 63–68.
B. Jones, Reconstructability analysis of general functions. Internat. J. Gen. Systems 11 (1985) 133–142.
B. Jones, Reconstructability considerations with arbitrary data, Internat. J. Gen. Systems 11 (1985) 143–151.
B. Jones, K-Systems analysis versus classical multivariate analysis, Internat. J. Gen. Systems 12 (1986) 1–6.

S. Iyengar, B. Jones, S. Trivedi, Reconstruction algorithms, Proceedings of the Seventh SIAM Conference on Discrete Mathematics, 1994.
B. Jones, Deky Gouw, Using K-systems theory to compute true main and interaction effects in a design of experiments, Cybernet. systems 26(4) (1995) 481–498.
B. Jones, Patti Isles, Parallel reconstructability algorithms, Adv. Systems Sci. Appl. 8 (1996) 215–220.
B. Jones and Deky Gouw, The interaction concept of K-systems theory, Internat. J. Gen. Systems 24(1–2) (1996) 163–169.
S. Iyengar, B. Jones, S. Trivedi, Reconstructability analysis approach to biological systems modeling, Computer modeling and simulations of complex biological systems, CRC Press Inc., Boca Raton, FL, 1997, pp. 135–161.