CONCEPTUAL DIAGNOSIS OF SIGNAL PROCESSING SYSTEMS

HAMID NAWAB [1], VICTOR LESSER [2], EVANGELOS MILIOS [3]

[1] ECS Department, Boston University, Boston, MA. 02215
[2] COINS Department, University of Massachusetts, Amherst, MA. 01003
[3] Research Laboratory of Electronics, M.I.T., Cambridge, MA. 02139

ABSTRACT

Signal processing systems usually have parameters whose settings are selected on the basis of the class of expected input signals. Finding the appropriate parameter settings for a class of inputs usually involves testing the system against typical and/or important inputs from that class. Whenever the system produces an incorrect output, the parameter settings responsible for the fault are identified. The system user can then adjust the system parameters in order to ensure correct system behavior. The diagnostic process of identifying the parameters responsible for system faults is generally difficult because the signal processing system carries out a complicated mathematical transformation involving a multi-stage algorithm that generates an enormous amount of intermediate data. We develop a new approach to the diagnosis of such systems. The approach is based on the availability of an abstract and possibly qualitative description of the input scenario and the use of an alternative system model derived from the underlying mathematical theory that explicitly represents the phenomena responsible for any incorrect processing.

1. INTRODUCTION

An acoustic signal processing system [Nawab] for determining the directions of low-flying aircraft is an example of a problem-solving system which carries out transformations based on Fourier theory. The system receives its inputs from a small array of microphones used to detect the sounds of nearby low-flying aircraft. The system maps these time-domain input signals into their frequency domain representations which are interpreted to indicate the number of aircraft and their directions.

In designing such a real-time signal processing system, a tradeoff is made between the scope of the input scenarios that can be correctly identified and the computational resources required. One way of gaining efficiency without limiting generality is to use a-priori knowledge about the input scenarios to focus the system resources for efficient recognition. This is accomplished by introducing parameters into the signal processing system for adjusting its behavior to a specific class of inputs (e.g. scenarios involving only helicopters). Changing to a different class of input scenarios is accomplished by changing the values of some of the parameters.

Parameter settings can be selected by testing the system against typical scenarios from a class of interest. If the system fails for a scenario, the cause of that failure is determined to assess whether it can be eliminated by parameter adjustment. It is advantageous if *the cause of failure is described in a way that clearly identifies the system parameter settings responsible for the failure* . We have developed and implemented a general strategy for carrying out this type of diagnosis.

Our diagnosis approach is predicated on the availability of abstract and possibly qualitative descriptions of the input scenario. For our particular diagnosis system in the acoustic signal processing context, a human observer provides this information. If there are inconsistencies between the system output and the user-specified description of the input scenario, the diagnosis system attempts to find an explanation for why the signal processing system failed in that particular instance. The human operator or an automatic system can consequently adjust the signal processing system parameters. We are currently developing such an automatic system to adjust parameters in response to information provided by our diagnosis system.

In this paper, we present an approach to diagnosis which uses a system model that describes how the system may *distort* the scenario characteristics of interest in mapping the input signals to the frequency domain. In particular, we use the underlying Fourier theory of the application domain to generate an alternative model for system processing than that used to perform the actual data transformation. This model represents the processing not as a transformation from the time-domain microphone signals to the frequency domain output but rather as a transformation from a frequency domain representation of the input scenario to the frequency domain output of the system. In this alternative model, we describe the system as a set of processes (e.g. aliasing, resolution, filtering) that transform the user-specified description of the input scenario to the system output. Each process is described in terms of how it transforms its input into its output. This transformation depends upon the values of the parameters associated with a process. Further, the parameters of the processes have direct correlates in the parameter set of the actual system. When the system is operating correctly (the system parameters are appropriately set), each of the processes performs an identity transformation from its input to its output. System faults are thus modelled as the result of non-identity transformations in one or more of the processes. Diagnosis from this perspective involves finding, through a means-ends analysis type of search [Newell], a minimal set of non-identity processes that explain the discrepancy between the user-specified information and the system output.

This work was sponsored at MIT Lincoln Laboratory by the Defense Advanced Projects Agency under contract F19628-85-C-0002 and at the Research Laboratory of Electronics by the Advanced Research Projects Agency monitored by ONR under contract N00014-81-K-0742 NR-049-506. The views expressed are those of the authors and do not reflect the official policy of the U.S. Government.

CH2243-4/86/0000-2911 $1.00 © 1986 IEEE

2. THE ACOUSTIC SIGNAL PROCESSING SYSTEM.

The acoustic signal processing system determines the directions of low-flying aircraft from recordings made by a small array of microphones on the ground. Signal analysis is done repeatedly every few seconds to keep the direction estimates updated. Each time the analysis is done, the output is a data structure representing a *direction spectrum* [Nawab]. This data structure explicitly represents the directions of each of the aircraft detected by the microphones.

Our diagnosis strategy, which we describe in the next section, makes use of multiple levels of abstraction for the Fourier domain representation of the direction spectrum. We use five distinct levels of abstraction: direction, power, frequency, band and gaussian levels. Their refinement hierarchy is illustrated in FIGURE 1. At the direction level, each aircraft is associated with just one characteristic -- its direction in the direction spectrum. Other characteristics of the signals are suppressed at this level. At the band level, we may either choose to describe the maximum and minimum frequencies of the signal (frequency level) or just the net power in the signal (power level). The next level combines the frequency and power level descriptions. Each signal is modelled as a rectangle with a minimum and a maximum frequency and whose area represents the net power in the signal. The height of the rectangle represents the average amplitude of the signal. At the lowest level of abstraction, the rectangle is replaced by a gaussian whose maximum amplitude equals the average amplitude of the rectangle. The sides of the rectangle correspond to the standard deviation of the gaussian.

3. DIAGNOSIS BASED ON FOURIER DOMAIN MODELS

We now describe our general strategy for diagnosis in terms of how it has been applied in the case of the acoustic signal processing system of Section 2. We assume that the user provides the diagnosis system a rough description of the correct answer (in the form of a direction spectrum) for the situation under diagnosis. A *plan-and-verify* paradigm is used by the diagnosis system to identify a sequence of Fourier domain processes that are responsible for the discrepancy between the system output and the correct answer.

Given the Fourier domain representation of the acoustic signal processing system, the problem of determining the non-identity processes involved in a particular situation can be viewed as a state space search; the states (to be called signal-states for our application) are Fourier domain directional spectra and the operators are the Fourier domain processes. The initial state is derived through a straightforward computation from the user-specified description of the correct answer. The goal state is a description of the directional spectrum at the output of the signal processing system.

We now phrase our diagnosis task as the following search problem:

Identify a sequence of processes such that

(i) the sequence maps the initial signal-state into the goal signal-state and

(ii) No proper subsequence maps the initial signal-state into the goal-state.

We talk about a *sequence* of processes since, as we shall see later in this section, the order in which the processes are applied to the signal-states is important. The requirement that there be no subsequence that can map the initial state into the goal state ensures that we exclude any unnecessary process in explaining the discrepancy between the correct answer and the system output. It should also be noted that there may be more than one sequence of processes satisfying our search criterion. This happens when more than one explanation can justify the differences between the system output and the correct answer. Unless intermediate data states of the actual system are available for inspection, such multiple explanations cannot be disambiguated. For our signal processing system, the intermediate data states are not available for inspection. Consequently, we have designed our search strategy so that it generally finds the sequence with the smallest number of processes. We were guided in this selection by the heuristic that *the simplest explanation is the most likely explanation for the cause of a system fault* .

The search is an iterative two-step process. The first step is a search based on an abstracted and qualitative view of the states and operators. We use a GPS approach for this planning phase. An example of a GPS operator description is given in FIGURE 2 for the *equal-resolution* process. The description is *qualitative* because the pre and post-conditions are specified in terms of ranges rather than specific values. We employ a range-intersection criterion for testing the pre-conditions. That is, if a pre-condition requires that a parameter be within a certain range, this condition is considered to be satisfied by any state for which that parameter has a range that intersects with the range specified in the condition. This is sufficient for our purpose because we are seeking any plan that may be possible within the specified uncertainty (expressed as numerical ranges) for the initial state. It should also be noted from FIGURE 2 that an operator may have *scenario-preconditions* . These are descriptions of the scenario (e.g. aircraft velocity) which are not captured by the direction spectrum representation. Our diagnosis system contains a database in which the user may enter such information about the scenario. The second phase of the two-step iterative strategy involves a detailed verification of the plan generated in the first step. The verification is carried out at the lowest level of abstraction at which the correct answer has been specified.

The planning process essentially forms *crude* plans (ordered sets of processes) by using as high a level of abstraction as possible for its pre- and post-conditions and by ignoring the precise values of signal-state parameters. This is done because the highest levels of abstraction hide most of the details and thus give rise to plans with the smallest number of operators. In cases where plans fail, the planner has the option of selecting different plans at the same level of abstraction or dropping to a lower level of abstraction and forming new plans there. In forming the new plans, our strategy also attempts to make use of the nature of failures in previous verifications.

The planning phase uses the generic means-ends analysis technique of GPS at the various levels of abstraction. The planner classifies differences between signal-states into seven categories -- missing-signal, unassociated-signal, direction-shifting, amplitude-scaling, frequency shifting, resolution and ghosting. It selects the most important difference and an operator that is likely to reduce the difference in the current situation. Control of the GPS search is accomplished through two important mechanisms.

First, no operator is allowed to appear more than once in a particular plan. This follows from the fact that each operator represents a single process in the signal-processing system. This is in contrast to GPS search strategies in problems such as algebraic simplification where the same operator can be used many times over. In those situations, the GPS strategy requires "depth heuristics" to avoid fruitless searches involving many instances of the same operator. In our problem, however, the constraint that an operator may appear only once in a plan removes the need for depth heuristics.

A second mechanism for controlling the GPS search in our system is the use of an *ordering relationship* among classes of signal-states. In particular, there are five classes of states. *Propagation domain* states represent plane wave signals propagating through the atmosphere and they must precede all other states. *Continuous-temporal* domain states come next and they represent one-dimensional analog signals. Next, *Discrete-temporal* domain states represent one-dimensional digital signals. *Continuous-spatial* domain states represent two-dimensional analog wavenumber spectra and finally *Discrete-spatial* domain states represent digitized wavenumber spectra. Each operator specifies the allowable classes of input and output signal-states. Thus our strategy does not permit plans in which the operators violate the domain requirements for their input and output states. This helps to reduce the search space considerably. It should be noted, however, that operators whose input and output states belong to the same domain can appear in any order with respect to each other.

When a candidate plan is generated by the GPS planner, a *verification* of the plan is attempted as the next step. The abstraction level of this verification is selected to be the lowest one at which a description of the input signal state is known. Our verification procedure makes use of the operator and state representation mechanisms used in the GPS planner. The verification process can be viewed as a slightly modified version of the GPS planner, but at the detailed abstraction level. The difference is that in this case the planner is provided with an already formed plan to guide its operator selection. If the plan is successful in reaching the goal state, our plan-and-verify strategy ends with the executed plan representing the desired diagnosis. On the other hand, if the verification fails at some point, further diagnosis is guided by the nature of the failure as discussed below.

There are two basic types of failures. In one case, the pre-conditions of an operator in the plan are not satisfied by the state preceding the operator. For such a situation, a plan re-adjustment is attempted by finding a plan for linking the state and the pre-conditions of the failed operator. In the second type of plan failure, the state at the output of an operator does not match the qualitative description anticipated for it in the original plan. For such a situation, a plan readjustment is attempted by eliminating the failed operator from the plan and devising a different plan to replace its position in the original plan. Note that the adjusted portions of a plan can be at lower levels of abstraction than the current level of abstraction chosen for the entire planning process. If local readjustment of a plan is not possible, the basic level of abstraction is lowered and a new planning process is started.

Our planning and verification strategy has much in common with the ABSTRIPS problem-solving system [Sacerdoti]; as well as a number of important differences. Both systems use GPS for candidate generation and use a verification environment to test the plans. They differ in that ABSTRIPS uses resolution theorem proving as the basis of the test in the verification environment, our system uses the successive application of plan operators to test whether the initial state is transformed into the goal state. More importantly, however, they differ in how they exploit the abstraction space.

Both systems use multiple levels of abstraction for their operator descriptions. In our system, we carry out verification even for plans which are constructed using the highest levels of abstraction in the operator descriptions. In contrast, the ABSTRIPS system does not execute the plans formed at the higher levels of abstraction. Instead it uses them in a process of *refinement* to produce plans at lower levels of abstraction. Thus no portion of a plan is considered ready to be executed in the verification environment until it has been refined to the lowest level of abstraction. Thus, in some sense, the ABSTRIPS system does a depth-first search along degrees of abstraction for plan portions to be executed in the verification environment. On the other hand, our system searches breadth-first across a single level of abstraction in search of complete plans to be executed in the verification environment. Our system drops to a lower level of abstraction only when it cannot form a complete plan at the higher levels of abstraction.

4. EXAMPLE

In this section, we use an example to illustrate our plan-and-verify problem-solving strategy. Let the two aircraft in our example be AIRCRAFT-1 and AIRCRAFT-2, respectively. The initial state is a user-provided description of these aircraft. We will denote this initial state by *S0* and its *qualitative* description at an abstraction level *i* by *Qual(S0, i)* . With this notation in mind, we consider the qualitative description of the initial state at the direction-level in our example:

Qual (S0, direction):

AIRCRAFT-1 at direction D1
AIRCRAFT-2 at direction D2
D1=[0,10] deg.
D2=[35,50] deg.

Let us denote the goal state for our problem by *Sf* . where *Qual (Sf, direction)* is given by:

Qual (Sf, direction):

AIRCRAFT-3 at direction D3
D3 = [20,20] deg.

The program's goal is to transform *Qual (S0 , direction)* to *Qual (Sf, direction)* . A set of rules determines the differences that exist between *S0* and *Sf* . Two of the differences detected are resolution and direction-shift. Resolution differences are characterized by having two directions in the input state while the output state has a direction between those two directions. A direction-shift is a difference characterized by a direction in the input state shifting to a different direction in the output state.

Another set of rules assigns priorities to the selected differences. In our example, let us assume that the resolution difference is given the highest priority. The next step is to *select* an operator Q that might reduce the highest priority difference. In our example, the selected operator would be one called "Equal-Resolution". This operator acts upon two signals whose directions are close to each other and have the same minimum and maximum frequencies. The result is a single signal with the same frequencies but a direction enclosed by the original two directions. While the equal-resolution pre-conditions, *qual-pre (Equal-resolution, direction)* require that the two aircraft be closer than 20 degrees apart, directions in *Qual (S0, direction)* are specified to be between 25 and 50 degrees apart. A difference of direction-shift thus exists between *Qual(S0, direction)* and the *qual-pre (Equal-resolution, direction)* . The planner then selects the Fast-velocity operator to reduce this difference. The fast-velocity operator represents the effects of fast aircraft velocities on the direction measurements. The *qual-pre (Fast-velocity, direction)* conditions require that an aircraft have a velocity greater than (200 / analysis-interval) m/sec, where analysis-interval is one of the system parameters. Suppose that in our example, the analysis-interval parameter is set at 4 secs. and AIRCRAFT-1 has a velocity of 150 m/sec with increasing angle. The *qual-post(Fast-velocity, direction)* conditions specify that in such a case the fast-velocity operator will create a new state, say *S1* , in which the direction of AIRCRAFT-1 changes by 6 degrees, making it fall in the interval [12,22]. The direction difference between [12,22] and the direction [35,50] of AIRCRAFT-2 falls in the interval [13, 38]. Since this intersects with [0,20], the planner concludes that *Qual(S1, direction)* matches the *qual-pre (Equal-resolution, direction)* conditions. Furthermore, the output of Equal-resolution will be a signal whose direction falls in the interval [12,50]. Thus *qual-post (Equal-resolution, direction)* conditions match the *Qual (Sf, direction)* conditions. Thus, a complete plan at the direction level has been formulated to connect *S0* and *Sf* . The plan consists of the fast-velocity operator followed by the equal-resolution operator.

The plan generated above is then passed to the verification stage which tries the plan at the gaussian level of abstraction. It is found that the output state *S1* produced by the fast-velocity operator does not match the pre-conditions of the equal-resolution operator due to the fact that the two aircraft do not have the same minimum and maximum frequencies In accordance with our diagnosis strategy, we move to the plan adjustment phase. Our goal now is to reduce the frequency-shift difference between *S1* and the pre-conditions of the Equal-resolution operator. The frequency-shift difference cannot be dealt with at the direction level because frequency ranges cannot be described at that level. The planner therefore drops to the frequency-level. The Elevation-compression operator is selected for reducing this difference. The Elevation-compression operator represents the phenomenon of the compression of frequency ranges for aircraft that are at high elevations with respect to the microphone array. Elevation-compression applies to *S1* and produces another state *S2* . It turns out that the *Qual (S2, frequency)* conditions and the *qual-pre (Equal-resolution, frequency)* conditions match each other. We therefore have an adjusted candidate plan consisting of three operations: fast-velocity, elevation-compression, and equal-resolution. The adjusted plan is then passed and successfully verified at the gaussian level of abstraction.

REFERENCES

[Nawab] S.H. Nawab, F.U. Dowla, R.T. Lacoss: "Direction Determination of Wideband Signals", IEEE Transactions on ASSP, pp. 1114-1122, October, 1985.

[Newell] A. Newell and H.A. Simon: "Human Problem Solving", Englewood Cliffs N.J., Prentice Hall. 1972

[Sacerdoti] E. Sacerdoti: "Planning in a Hierarchy of Abstraction Spaces", Artificial Intelligence, vol. 5, pp. 115-135, 1974.

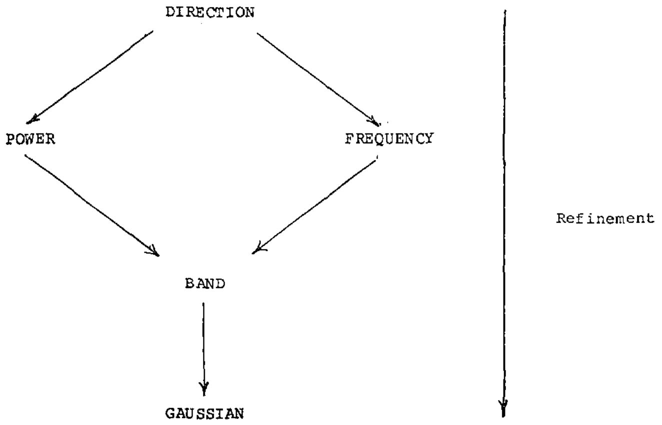

FIGURE 1

```
EQUAL RESOLUTION OPERATOR

     INPUT SIGNAL TYPE:      propagation, continuous-temporal,
                             discrete-temporal, continuous-spatial

     OUTPUT SIGNAL TYPE:     continuous-spatial

     DIFFERENCES REDUCED:    resolution

     OPERATOR PARAMETERS:

          DIRECTION          array-aperture
          POWER              array-aperture
          FREQUENCY          array-aperture, epsilon
          BAND               array-aperture, epsilon
          GAUSSIAN           array-aperture, epsilon

     STATE PRECONDITIONS:    Per pair of input signals

          DIRECTION          Direction difference intersects
                             [0, 100/array-aperture].
          POWER              Direction level preconditions.
                             Power in [0, inf].
          FREQUENCY          Minimum-freq's intersect.
                             Maximum-freq's intersect.
                             Direction difference intersects
                             [0,1000*epsilon)/(array-aperture*.0001*maximum-freq)].
          BAND               Power level preconditions.
                             Frequency level pre-conditions.
                             Amp in [0, inf]
          GAUSSIAN           Frequency level pre-conditions with
                             gaussian model.

     SCENARIO PRECONDITIONS: none

     STATE POSTCONDITIONS:   Per pair of input signals

          DIRECTION          Delete input signals.
                             Create signal whose direction is the
                             cover of the two input directions.
          POWER              Direction level postconditions.
                             Power of output signal in
                             [0, sum of maximum powers in signals].
          FREQUENCY          Direction level postconditions.
                             Minimum-freq of output same as input
                             Maximum-freq of output same as input
          BAND               Frequency level postconditions.
                             Power level postconditions.
                             Amp of output signal in
                             [0, sum of maximum amps in signals].
          GAUSSIAN           Band level postconditions with gaussian
                             model.
```

FIGURE 2