In Proceedings of IEEE Conference on Systems, Man and Cybernetics, Vancouver, 1995

Application of Holonic Methodologies to Problem Diagnosis in a Steel Rod Mill

Duncan McFarlane
Industrial Control Centre,
University of Westminster,
115 New Cavendish St.,
London, UK, W1M 8JS

Bruce Marett
BHP Research -
Melbourne Laboratories,
PO Box 264, Clayton,
Australia, 3168

Gary Elsley
BHP Rod and Bar Products
Division, Newcastle, Australia

Dennis Jarvis
CSIRO Division of
Manufacturing
Technology, Adelaide,
Australia

ABSTRACT

Holonic architectures have been suggested as a possible building block for future manufacturing systems. The term holonic implies a unit in manufacturing that demonstrates the dual characteristics of autonomous behaviour (when required) and the ability to function co-operatively when the situation requires it. One of the key functions that is important to both of these properties is the need for the manufacturing unit to be able to (self) diagnose performance and functional failures in a systematic manner. This paper proposes a generic methodology for self-diagnosis within a holonic manufacturing system, in which faults and problems are analysed as a standard function of the system.

1. INTRODUCTION

In 1993/94 as part of the international Intelligent Manufacturing Systems (IMS) a feasibility study was established to determine the effectiveness of so called *Holonic Manufacturing Systems* (HMS) - a proposed architecture for manufacturing systems which emphasises the properties of co-operation and autonomy as a means of achieving flexible and modular system properties. The latter are important for industry to meet future market requirements while continuously reducing overall production costs.

In order to trial these methodologies on a real application, BHP Steel and Rockwell (USA) developed a simulation of a steel rod mill spray cooling line and developed a conceptual Holonic Cooling Control system to manage this process. Previous work ([1], [2]) has demonstrated the use of negotiation strategies as tools for achieving co-operative behaviour in control and process set-up. This work addresses another key issue - that of a generic approach to self-diagnosis which is an important aspect of autonomous behaviour. Generalised diagnosis methodologies have previously been developed for retrofitting diagnositic capabilities to existing processes and equipment (see [3] for example for an excellent description). This paper takes the framework for holonic systems developed in [1] and [2] and outlines a prototype methodology for holonic self-diagnostics which could ultimately be integrated into the design of any manufacturing system.

2. DIAGNOSIS IN HOLONIC SYSTEMS

In this section we review the work in [1],[2] where a prototype framework was developed for a holonic manufacturing system and introduce some basic concepts in problem diagnosis which are to be incorporated into this framework.

2.1 The BHP/Rockwell Holonic System Architecture

A holonic system design consists of a collection (holarchy) of holons, each of which is composed of a standard architecture consisting of a set of component functions as shown in Figure

1. Interfacing is achieved via a standardised method (e.g., function blocks for software components). These component functions may themselves be holons, although some of the functions may not be required within the base holon. The important criteria to be met are the key attributes of co-operation and autonomy. To this extent, even if some of the component functions have been reduced to zero in a particular application, as long as the key attributes are met, then the holon description is generally appropriate. In the case where the key attributes are not satisfied, then the holon reduces to a functional module.

The proposed HMS scheme has many attributes commonly found for example in a distributed operating system, where a multitasking, real-time kernel provides support for an autonomous and co-operative environment for each system component. Certain attributes and services must naturally be provided, for example some self-diagnosis capability. This can be configured manually, or in some cases automatically, to suit particular applications and the means for doing this is the subject of this paper.

The architecture of the proposed holonic control system (see Figure 2.1) consists of the following components which are integral to autonomous and cooperative behaviour:
1. *Processing holon* is responsible for the execution of algorithms (e.g., control, planning), auxiliary modelling calculations and subsidiary calculations as required, specified by a state map of the holon functions. Additionally, any sub functions which comprise the main functions of the holon are contained within the processing holon.
2. *Negotiation holon* is fundamental to the operation of any holon and implements the key attribute of co-operation, through negotiation. A global goal is given to each holon and local goals are determined for each holon by a distributed negotiated optimisation technique.

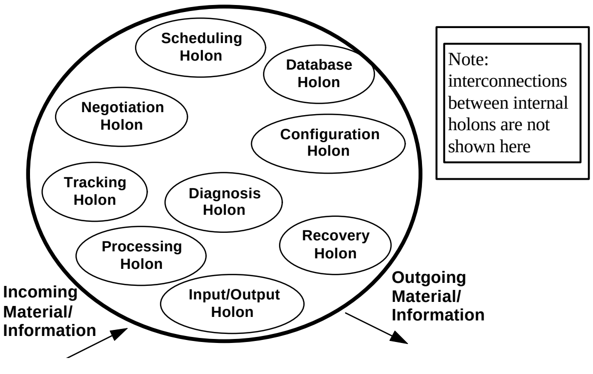

Figure 2.1 Holon Structure

3. S*cheduling holon* is responsible for scheduling activities associated with the operation of the holon system's global goals given current product parameters.
4. *Database holon* is responsible for all historical archiving of data and/or information. This information can then be accessed for use with set-up tables, tuning, negotiation,

condition monitoring; basically any task which requires historical data.

5. *Diagnosis holon* is an integral part of the holonic system providing fault monitoring and diagnosis for the control holon.
6. *Configuration holon* acts to reconfigure the system as required. It detects the addition or deletion of holons or objectives, and adds additional I/O and code as appropriate.
7. *Recovery holon* provides a means for performing component and process fault management.
8. *Input/output (I/O) holon* is fundamental for passing data or information in and out of each holon.
9. *Tracking holon* is responsible for synchronising all the sequencing operations for the base holon and its immediate surrounds.

The holonic structure is intended to allow for nesting of other holons, as the architecture is applicable at the many functional levels of a manufacturing process. (The reader is referred to [2] for more details on the architecture proposed.)

2.2 Traditional Diagnosis

Industrial equipment and process diagnosis is commonly described as being comprised of three stages: problem detection, analysis, and assignment of root cause. Note that problem detection is preferred to fault detection here as it is a term more commonly applied to both process failures and equipment faults. The analysis stage consists of comparing process data with norms to establish further evidence of the problem and to elicit all possible candidate causes. Other data, frequently of a procedural nature (e.g. sequence, timing) will provide the context often needed to establish the likely root cause.

A general event-driven diagnosis model is given in Figure 2.2. which is based on the methodologies in [3]. The diagram outlines the key actions in effective diagnosis and will be used as the basis of the holonic diagnosis described in the next section. The reader is referred to [3],[4] and [5] for further background information on conventional diagnostics.

2.3 Incorporating Diagnosis into Holonic Systems

Much of the traditional model of diagnosis in Figure 2.2 is applicable to holonic systems, the key difference being in the distribution of the tasks and the concepts of heirachy and alone PC programmed to collect data, detect problems, diagnose faults and recommend actions). In a holonic system the diagnostic function will be an integral part of the system and each holon will be responsible for supplying its own health information to the appropriate diagnosis holon.

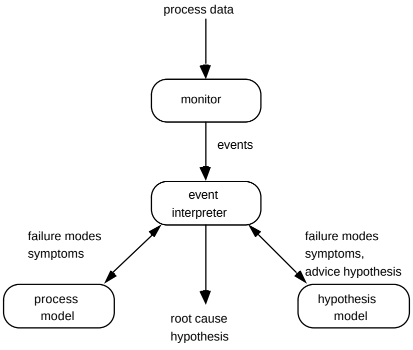

Figure 2 .2 Traditional Diagnosis Procedure

With respect to information requirements, the structure in Figure 2.2 is repeatable at different functional levels of holons, allowing a uniform diagnostic procedure to be applied at any level of holon. (This will be discussed in more detail later.) The main concern is the scoping of procedural knowledge at each holonic level, where it is noted that this knowledge can be of two forms:

1. *Process specific* - relating to the behaviour of the particular manufacturing system configuration
2. *Equipment (holon) specific* - relating to fundamental characteristics of the particular holon.

The latter may be obtained a priori from accumulated knowledge (as part of the standard capabilities of the equipment) while the former needs to be developed by process simulation and trialing. Note that a high functional level holon would see a high level event (internal or external) and the event sensor could be a message identifying neigbouring holon processes, or lower level holon events from within which it would co-ordinate and diagnose at its level.

Figure 2.3 reinterprets (a section of) the generic holon structure of Figure 2.1 from the perspective of self-diagnosis. An unsolicited self diagnosis may commence from the diagnosis holon which will provide data to the database holon. The database holon then negotiates with the schedule, process, configuration and diagnosis holons (and their diagnosis holons if applicable), to establish parameters for conducting the self diagnosis. Once this is completed, the diagnostic outcome is relayed to the database which reports the diagnostic status to the I/O holon.

This self test capability is perhaps best illustrated by example. Consider the need for the cooling box holon (described in a later section) to test its control valve for (flow) saturation. The diagnosis holon would determine what was required for the test (within itself) and request the I/O holon to monitor flow rate and valve positions, subject to any configuration and scheduling requirements of the responsible holons (e.g. wait for time window to conduct test, and reconfigure sluice valve, if necessary). The process holon conducts tests and returns data to the database holon which enables comparison of data collected with its model (characteristic flow curve) of the valve. This holon would report the outcome of the comparison to the diagnosis holon which would make the decision as to whether further diagnostic information was required, and make a report on the cooling box health status.

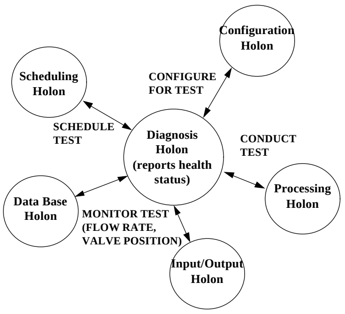

Figure 2.3 Generic holon structure & diagnostic functions

If an unhealthy (failed) status is determined then the diagnosis holon would be able to establish possible root causes (e.g. valve sticking or saturation) by reference to the database holon. Identification of most likely root cause (e.g. the function of the hypothesis model in traditional diagnosis) would be conducted by establishing process sequence and time information from other holons.

2.4 Holonic Systems Notation

The notation adopted in this paper for describing holons on different functional levels is as follows:

H(i,j,k,f(*)) = level i, holon number j (where multiples of a given type of holon exist at the same functional level), linked to parent holon k (at level I+1), performing function f(*)

Hence, the lower the level the greater the range of functionality of the holon. The following ordering will be used:

Holonic Function Ordering:
f(1) - Processing holon
f(2) - Negotiation holon
f(3) - Scheduling holon
f(4) - Database holon
f(5) - Diagnosis holon
f(6) - Configuration holon
f(7) - Recovery holon
f(8) - Input/output holon
f(9) - Tracking holon

For example, the diagnostic holon which forms part of the cooling control holon (refer to Figures 1 or 3), could be denoted H(3,1,2,f(5)) to denote a level 3 holon, with holon number =1 as there is only one diagnosis holon - f(5) - linked to the parent holon, cooling control holon 2. Hence, the reference for the cooling control holon itself is H(2,2,1,f(1)), where level i=1 refers to the higher level process line holon (number 1) and the cooling holon is in fact a processing holon for the process line.

3. DETECTION

Detection involves the signalling but not necessarily the tracing of a problem in the holonic system. The method proposed for a holonic environment is the use of a *health status* submitted by each holon to indicate normal operation or (potential) problems. The health status is the outcome of a local fault detection within the holon which could be solved by a variety of instrument or analytic (model based) redundancy techniques (refer to [4,5] for a range of approaches).

3.1 Health Status Concept

All holons calculate a health status. The health status is typically a value ranging from 0 to 1, where 1 indicates no faults detected (e.g. new system), whereas 0 is completely unavailable for use. A health status of 1 can be interpreted as a perfect match of the holon’s current operations with a model of the holon operating under normal conditions. Conversely, health status 0 denotes a gross discrepancy between current operations and model predictions. (It is precisely by comparing operations to internal models that we determine the health status.)

The analogue nature of the health status allows for statistical outcomes and predictability of the likelihood of catastrophic failures to occur. (See for example the confidence number method of [6].) Particular holons have a certain threshold below which certain tasks can no longer be performed with any degree of confidence. The notion of health status, therefore allows for degraded performance to be achieved, with confidence levels determined for the achieving targeted performance specifications. An alternative approach to looking at the health status of a component is to look at three states: healthy, not healthy and indeterminate. The third state is assigned if the diagnostic system is unable to decide the health or otherwise of the system.

The health status is calculated by each of the holons within any base holon, and reported to the base diagnostic holon. This diagnostic holon then communicates with other neighbouring holons and shares their status. The processing of the returned health statuses is the diagnostic task carried out by the diagnosis holon.

3.2 Detection at different holon levels

The aim of the holonic detection procedure is to prepare a signature of the health of the surrounding holons for analysis by the diagnostic holon. This signature is simply a configuration of health statuses compiled from contributing holons. When a health failure is detected, a signature is collected by each participating diagnostics holon, and further analysis is driven by the highest ranking holon connected to either an alarm, compensation or maintenance scheduling function. (E.g. Valve failure in a steel cooling zone would be handled at a local level if it did not endanger smooth plant operations.).

A signature is developed from two types of interholon communications:
1. **Unsolicited Information** - each neighbouring holon continually provides the diagnostic holon with its health status as is outlined in the example below.

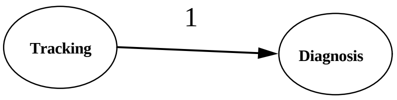

Figure 3.1 Simple Unsolicited Health Information

Note that the Input/Ouput Holon will indirectly convey health information about neighbouring holons via its own health status.
2. **Interrogated Information** - the diagnostic holon compiles local health information into a single level health signature - e.g. (1,0,1,....) - where the entries are in the same order as the ordering of the functions given in the Section 1.4. Where more than one function of a given type can exist (e.g. processing holon), the reported health is 0 unless each one of the functions has health of 1.

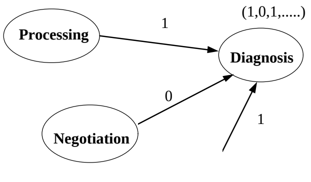

Figure 3.2 Formation of Health Signature

A health signature of (1,1,1,1, ...) indicates a healthy holon, and a health status of 1 is passed to the next level. However, any non-unitary entries in the signature, lead to a deeper interrogation to determine more details about the unhealthy holon. For, example, a 2-level signature can be developed as in Fugre 3.3. The level to which a particular signature is developed is dependent on the detail that is required to identify the particular fault. As will be discussed in the next section, the collected signature can be compared to known health signatures stored in the data base holon at the same level for diagnostic purposes.

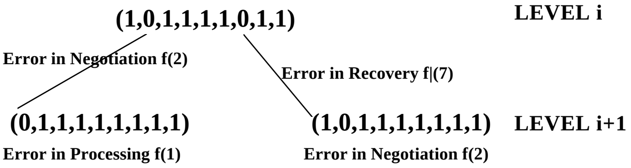

Figure 3.3 Two Level Health Signature

3.3 Health Detection Protocol

In order to develop a generic diagnostic methodology within the holonic environment, a set of principles summarising the signature system given in the previous section are now specified.

i. The diagnostic holon H(i+1,1,j,f(5)) can accept status requests from superior levels diagnostic holon H(i,1,k,f(5)) to which it will respond by reporting on the health status of holon H(i,*,j,f(1 or 8)) which in most cases will be either a processing holon or input/output holon.
ii. The diagnostic holon H(i+1,1,j,f(5)) can interrogate each of its peer level holons H(i+1,*,j,f(*)) (through their respective diagnostic functions) for deeper diagnostic information in order to develop appropriate health signatures.
iii. The diagnostic holon H(i+1,1,j,f(5)) can accept unsolicited messages from each of its peer level holons H(i+1,*,j,f(*)) detailing their health status.
iv. The diagnostic holon H(i+1,1,j,f(5)) is responsible for diagnosing and monitoring performance of all non holon components of holon H(i,*,j,f(1)) (or holon H(i,*,j,f(8)))
v. Any diagnostic holon H(i+1,1,j,f(5)) will forward up messages or health status reports to its superior diagnostic H(i,j,k,f(5)) up to the point where the message can be presented for appropriate responses (e.g. alarming, compensation or maintenance scheduling)
vi. The status of a holon will be reported either as healthy (1) or as unhealthy (0) with an accompanying signature of health of peer holons.

4. DIAGNOSIS

This section deals with the problem of diagnosing a problem (within a holonic framework) once it has been detected using the process described in Section 3.

4.1 Holonic Diagnosis

Diagnosis in a holonic system describes the operations of the dedicated diagnosis holon, interpreting the health signals provided, comparing signatures with those stored in the data base holon and interrogating neigbouring holons for further information if required in order to uniquely determine the nature and location of the fault. The diagnosis holon is also responsible for setting the health status of the upper level holon and providing diagnostic information as requested from higher levels.

4.2 Self Diagnosis Operations - A possible methodology

In this section a possible methodology for handling a fault in a low level component is outlined. Suppose that a holon has detected anomalous behaviour. There are 3 cases to consider - the fault condition will either be present in:

1. a component of the detecting holon
2. a subholon of the detecting holon
3. an input to the detecting holon

In practice, it is unrealistic for cost and configuration reasons to assume that a holon will have a sufficient number of sensors to enable it to isolate a fault condition to a single candidate. Consequently, it is assumed that while some faults will be able to be identified by the detecting holon, in general, several holons will need to co-operate in identifying the cause of anomalous behaviour. To cater for the situation where there is a less than ideal number of sensors, virtual sensors, using internal holon models and values inferred or calculated from other real/virtual sensors. Also associated with the sensor reading is a degree of confidence (ranging from 0 to 1).

A holonic diagnosis is required to be achieved in a generic manner, the conventional approach employed in empirically derived rule based systems of associating a fault with the state arising from that fault is be inappropriate, as it requires diagnostic rules to be customised for each system. An alternative approach is to reason directly about the system using internal holon models of structure, behaviour and function in conjunction with the health signature of the system developed from inter holon interrogation. As will be shown in the next section, a hybrid approach using both rules and models is most realistic, where the number of rules is minimised to allow for as generic a methodology as possible with minimum process specific training.

This approach assumes that we have knowledge (models) of

1. *characteristic faults / failure modes* (e.g. valve stuck open, valve stuck closed). Note that this information will be specified a priori and that the bulk of this knowledge will be inherited from generic holon classes.
2. the *structural context of the current holon*. If we know how a holon is physically connected to other holons and if we have mechanisms for predicting system behaviour, we can dynamically implicate or exonerate certain holons.

These two pieces of information comprise the information that will automatically be addressed within the standard specification of a holonic system as described in [1,2]. Further, it is assumed, that via (possibly automatically derived) rules, we have information regarding

3. *Problems that can occur in operation*. These may be derived via one of the follow possible methods: (i) manual matching of known process errors to fault signatures (ii) doing the same via simulations and learning of characteristic signatures (iii) error “walk through” of the actual process (iv) on the job learning.

These three pieces of information can then be used to develop candidate signatures associated with both equipment and process problems that can occur. The next issue is, for a given problem detection, to extract the correct fault signature from those held in the data base, and to handle possible ambiguities that can arise. This is done by the generation and manipulation of candidate lists of possible signatures.

5. ACTIONS

Communication of diagnosed health to alarms, maintenance and operation schedules etc. would depend on the holon level in the holonic structure. At the lowest level holons self diagnosis would be done as described above, and the diagnosed health would be translated to other (higher level) holons. Subsequent analysis of health may require some negotiation between holons in a way described in [1,2] in order to a) determine the exact location of the fault and b) determine the appropriate compensation and repair actions.

5.1 Alarming

Alarm conditions are generated based on the following:
a) *Unsolicited* - local detection of faults using standard (existing) fault detection schemes which are communicated by comparing health status conditions with threshold limits. These use conventional logic to generate alarm conditions, and includes detection of availability of resources (water, air, electricity, etc.), detection of device failure, detection of abnormal operating conditions. These fault conditions are usually associated with standard hardware sensors which are designed to detect failure of supplies, devices or components.
b) *Interrogation* - results of the diagnostic signature interpreter, which assigns a health status to particular tasks. This may be the result of interpreting diagnostic information from a range of sources depending on the tasks to be performed, and consequently is a collective result, or indeed, a prediction of the likelihood of success or failure. The resulting health status is passed up to level at which an alarm can be appropriately selected.

5.2 Responses

The response of the diagnostic system is a decision making exercise whereby, based on the degree of seriousness of the fault condition, the following actions may be initiated:

ITEM	DESCRIPTION
Graceful degradation of Performance	determine if the task can carry on anyway in some degraded capacity.
Reorganise Operations	renegotiate the task with the appropriate holons based on the new value of the health status.
Stop operation	the failure is such that the task cannot be completed. The holon reports itself as unable to continue and passes control for the other participating holons to resolve through renegotiation.
Amend maintenance schedule	send the new health status to the maintenance holon and renegotiate the maintenance contract. This determines, for example, when the components require servicing.
Detailed Diagnosis	as part of the maintenance contract negotiations it may be desirable to perform signature analysis testing, where test signals or calibration signals are sent to the suspect device to re determine the operating characteristic of the device. This would have to be scheduled at an appropriate time slice in the production cycle.
System Reconfiguration	to allow continued operation of the system; this may involve reconfiguration of hardware and software components, and utilisation of the redundancy holon.
Online performance evaluation tests	which may involve perturbing the control output by known amounts, and measuring the output, building a table of results which can be used to retune the model, determine fault conditions, or degraded conditions. These could involve other participating holons, and rely on different permutations or combinations of control shifting.

6. HOLONIC DIAGNOSIS APPLIED TO THE RBPD ROD MILL COOLING SYSTEM

In this section the holonic diagnosis methodology is evaluated in a conceptual application to the BHP RBPD Rod Mill spray cooling line. In reality, the diagnostic function would not operate as an isolated entity, but for simplicity other operating functions are not considered here.

6.1 The BHP RBPD Rod Mill

The principal function of a rod mill is to reduce steel billets of dimension 12m x 127mm x 127mm to rod of between 5.5 and 12 mm in diameter (see Figure 6.1). This process involves a heating of the billet to temperatures of 1150^{o}C, then a multi-pass rolling process which gradually reduces the bar to the required diameter. The reduced bar is then water cooled, followed by controlled air cooling.

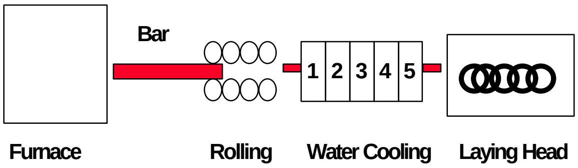

Figure 6.1 Schematic of a Rod Mill

The cooling system must be adaptive to the customer needs of rod in different shape, size and grade of high metallurgical and dimensional quality for varying lot sizes. Recovery from malfunction is essential, and flexibility in configuration is necessary to meet the demands.

6.2 The Holonic System for Rod Mill Cooling Control

The proposed architecture is comprised of a holarchy of cooling holons which negotiate with each other to achieve performance goals. The negotiation is distributed with management of the tasks and achievement of local goals for each holon carried out by consensus with respect to a distributed cost function. The key system components are:
Water Box Cooling Control Holon - Transforms bar of a given input temperature to a lower output temperature as the bar passes through the water box. Provides all control support for determining appropriate flows required to achieve rod temperature set points for a wide range of product parameters.
Valve Control Holon - Lower level holon (a Processing Holon within the Water Cooling Box Holon) for ensuring that flow requests are met and that correct water flows are provided to the cooling control holon. Interfaces with Maintenance Holon to ensure adequate condition is maintained and that sensing devices are correct.
Operator - In charge of ongoing maintenance (i.e. day-to-day) of water box performance. Negotiates maintenance contracts to automatically service failed or degraded components. Performs functions under self-diagnosis and scheduling holons in the Water Cooling Control holon.

6.3 Key Problems and Fault Maps

Some key problems experienced with the Rod Mill Cooling System are related to operating valve failures and ageing or wear of water spray nozzles. The operating valve failures are associated with either the diverter valve which shunts water from the bar to the sluice, and the control valve which has the desired flow setting appropriate to the grade and section of steel rod being processed. The diverter valve is a high speed valve which operates based on signals received by the operation of the Hot Metal Detector (HMD) which detects the presence of the rod entering the cooling zones.

The fault map in Table 6.1 shows some of the key problems identified for the Cooling System. The map shows the particular fault, and the indicator or symptom associated with identifying that particular fault. In each case, the fault is linked to the particular holon(s) in one of the Water Cooling holons which is (are) likely to reflect a reduction in health as a result of the fault.

FAULT	INDICATOR	HOLON LOCN	ERROR	OTHER_CAND
Control Valve sticks	SLOW TREND OVER, SAY, 0.5HR	STATE MAP & HISTORY (Database holon)	CONSISTENT TEMP ERROR; POSSIBLY IN WRONG DIRECTION	o NOZZLE WEAR o LEAKING HOSE &/OR CLAMPS o REDUCED WATER PRESSURE
Diverter Valve does not turn off	COBBLE FOR SMALL SECTIONS	PHYSICAL PROCESS MONITOR STATE MAP & HISTORY (Tracking holon, database holon)	BAR TAIL COLDER THAN NORMAL	o LAYING HEAD FAILURE/ PROBLEMS o OBSTRUCTION, e.g. incorrect guide positioning
	LAYING HEAD PROBLEMS	PHYSICAL PROCESS MONITOR (tracking holon)	TAIL DOESN'T LAY PROPERLY	
Diverter Valve does not turn on	HIGH TEMP. READING	STATE MAP & HISTORY (CH) (Tracking holon, database holon)	SMALLER CONSISTENT ERRORS (SPC)	o PYRO BROKEN o OTHER DV FAILURES o WORN NOZZLE o CV FAILED o BLOCKED NOZZLE o NO WATER o BROKEN HOSE; CLAMP
		CONTROL (CH)[1] (Processing holon)	LARGE TEMP. ERRORS	
Nozzle Wear	1ST BAR OUT OF SPEC.	SCHEDULE (CH) (Schedule Holon)	TEMP ERRORS cf. PREVIOUS SETUP	o WRONG NOZZLE o NO WATER o WRONG PRODUCT/GRADE o WORN NOZZLE
	1ST 5 BARS OUT OF SPEC	STATE MAP & HISTORY (CH) (Processing holon, database holon)	CONSISTENT ERROR	
	FLOW SATURATION	CONTROL (CH & VH) (Processing holon)	SAME FLOW FOR SAME INCREMENT	

Table 6.1 Fault Map for Cooling Process

Clearly, a number of these faults will not be unambiguously determined merely from the observations given above. These observations can be linked to the *unsolicited* information defined in Section 3. Further *interrogation* is required to uniquely define a fault (when indeed it is possible to do so).

6.4 Diagnostic Traces and Signatures for Key Faults

Table 6.2 illustrates the methodology for identifying the holon locations of each initial fault detection and also provides a level 2 signature of the particular fault as determined from an interrogative action by the water cooling holons diagnosis function. Only the diagnosis associated with the *Control Valve Sticks* fault is given here but such a methodology could be extended to all candidate faults. Note that for simplicity, the fault is assumed to act on water cooling holon No 1 (denoted although the analysis could equally apply to all such holons

FAULT	HOLON LOCATION WHERE HEALTH STATUS IS DEGRADED	HOLON IDENTIFIER	LEVEL 1 SIGNATURE	HOLON IDENTIFIE (LEVEL 2)	LEVEL 2 SIGNATURES
Control Valve sticks	STATE MAP & HISTORY (Database holon)	H(3,1,1,f(4))	(1,1,1,0,1,1,1,1,1)	H(2,1,1,f(4))	(1,1,1,1,0,1,1,1,1)

Table 6.2 Sample Traces: Control Valve Sticking

6.5 Candidate Lists

A map of all candidate signatures is maintained by the data base holon and would include; manufacturer's standard candidate lists (e.g. faults, chemical hazards), standard company wide in house lists (equivalent to knowledge based

standard procedures, and beyond the existing stand alone failure mode data which has little richness in context), process plant generic lists, and local area specific (e.g. models of local plant, expertise based, configuration specific).

To identify most likely fault, a negotiation scheme is used between the holons to best utilise model, process sequence, configuration and time context information. This diagnostis is equivalent to the traditional hypothesis model but executed in a distributed manner between holons.

6.6 Actions Taken for Each Fault

Specifc actions taken for each of the faults in the case of the cooling control application are now given.

<u>1. Control Valve sticks</u> -

a) As described earlier the Level 2 Configuration Holon downloads the new non-linear characterisation of the valve to the Level 2 Processing Holon, or
b) The Level 2 Configuration Holon adds a dither signal to the control output of the Level 2 Processing Holon, and
c) The Level 2 Configuration Holon reconfigures the controller with some non-linear compensation to be implemented in the Level 2 Processing Holon.
d) The Level 2 Scheduling Holon negotiates a maintenance contract to service valve next down time.

<u>2. Diverter Valve does not turn off</u> -

a) The Level 2 Scheduling Holon in concert with the Level 2 Processing Holon (and associated holons to complete the task) performs a self test by pulsing the valve when no bar is present, to check on the operation of valve. If no response (observed manually as no sensors available - requires maintenance holon input) then negotiates maintenance contract with Maintenance Holon to replace the valve at the next appropriate time, or
b) if a self repair mechanism exists implement it, or
c) if the Redundant Holon is available (with capability to perform valve operation, i.e. hardware as well as appropriate software), then after negotiation and reconfiguration, instigated by the Configuration Holon, the replacement of the Diverter Valve is effected.

<u>3. Diverter Valve does not turn on</u> - same as for 2.

<u>4. Nozzle Wear</u> -

a) The Level 2 Scheduling Holon negotiates a maintenance contract to service the nozzle, or the Scheduling Holon after negotiation with Level 1 and Level 2 Diagnostic Holons and the Maintenance Holon replaces the nozzle, or
b) replace the nozzle as in 2c.

7. REFERENCES

[1] McFarlane, “Holonic manufacturing in continuous processing: concepts and control requirements”, Proc ASI’95, Portugal, 1995

[2] Agre, J et al, Rensslaers Manufacturing Conference, New York, October, 1994

[3] F.E. Finch, O.O.Oyeleye and M.A. Kramer, “A robust event-oriented methodology for diagnosis of dynamic process systems”, Computers Chemical Engineering, 14, (12), pp1379-1396, 1990.

[4] Isermann, R., “Process fault detection based on modelling and estimation methods - a survey”, Automatica, v20, n4, pp387-404, 1984

[5] Patton, R, Frank, P.M., Clark, R., Fault Diagnosis in Dynamic Systems: Theory and Applications, Prentice Hall, 1989

[6] Isermann, R., “Fault diagnosis of machines via parameter estimation and knowledge processing - tutorial paper, Automatica, v29, n4, 1993