**AshMod: Knowledge Model Based Operator Guidance System for Coal Washing**

Dr Hernan Villanueva  
*Department of Electrical and Computer Engineering, The University of Newcastle, Callaghan 2308, Australia*  
*Email: eehev@cc.newcastle.edu.au*

Harmeet Lamba  
*BHP Research, Newcastle Laboratories, PO Box 188, Wallsend 2287, Australia*  
*Email: lamba.harmeet.hs@bhp.com.au*

**Abstract**

We present the analysis, design and implementation of a knowledge based operator guidance system (OGS) in the coal washing domain. AshMod integrates the KADS methodology for the development of knowledge based systems with the G2 real time implementation environment. It assists the operator in monitoring the plant, performing fault diagnosis, and in plant optimisation. With the decision support provided by AshMod, the operator is able to control the amount of ash (impurity) in clean coal, hence increasing plant productivity.

One of our goals in developing AshMod was to implement a high quality knowledge based system that would assist validation and maintenance. This can be very difficult in a traditional rule based systems, but is made substantially easier through the graphical model based representations of domain knowledge.

AshMod can perform deep reasoning through the use of knowledge models that capture purpose, function, structure, behaviour and heuristics. The KADS methodology provides a systematic approach to the development of such knowledge models. Knowledge validation and maintenance is facilitated through the development of graphical object oriented knowledge models in G2.

AshMod is currently undergoing offline testing at the coal washing plants operated by BHP at Port Kembla, Australia.
1. Introduction

BHP is a large private company involved in the development of steel, mineral and petroleum products [3]. This operator guidance system (OGS) has been designed to assist operators at the coal washery (plants B and C) run by the BHP Slab and Plate Products Division (SPPD) of BHP Steel, in Port Kembla, Australia. The essential purpose of the coal washery is to improve the quality of raw coal by removing impurities [1]. The complexity of the coal washing domain is such that mathematical models are difficult to develop. Traditional artificial intelligence techniques are challenged by the incomplete and imprecise nature of existing domain knowledge. A systematic knowledge modelling approach was needed for the development of this operator guidance system. This OGS (AshMod) helps plant operators in maximising the clean coal yield while keeping ash (impurity) content within acceptable limits.

AshMod contains knowledge for interpretation of process data, and is intended to advise operators on process conditions and control actions [6]. It enhances the performance of the plant by allowing the operator to make more informed decisions. By providing consistent and timely guidance to the operator, AshMod assists an operator in getting higher yield while maintaining consistent product quality. It is not intended to replace the operator, but rather to act as an assistant.

AshMod applies the knowledge modelling principles propounded by the KADS methodology [10][11] and is implemented using G2 [7][8]. Sections 2 and 3 provide an introduction to the problem domain and an overview of AshMod. The knowledge models are detailed in Section 4, and the task, inference and strategy models are described in Section 5. Section 6 discusses knowledge validation and maintenance. Finally Sections 7 and 8 wrap things up with a conclusion and a discussion of the current status of this application and possibilities for future work.

2. Problem Domain (Coal Washing at BHP)

The B and C plants process 250 and 350 tons of raw coal per hour. Unwashed coal is fed into these washeries to reduce the ash (impurity) content in the coal. The coking coal and energy coal output from the washery is used for export and for internal use. High ash content decreases the thermal value of coal. Hence the plant operation is geared towards controlling the ash content in the washed coal.

Sizing screens are used to stratify the coal on the basis of size and washed by separate processes optimised to remove impurities from coal of that size. Jigs are used to wash large coal, cyclones to wash small coal, and flotation cells are used to wash fine coal (Figure 1). The jigs and cyclones are gravity stratification processes that exploit the fact that all removable impurities associated with coal are higher in specific gravity than the coal. Flotation cells, on the other hand, exploit the hydrophobic nature of coal to separate it from impurities.
Operators continuously monitor the plant by looking at large volumes of sensor data. Sensors operating in a rough industrial environment sometimes record incorrect readings. Operators validate sensor readings to determine whether the measurements can be relied upon. The readings are subject to noise, which the operators filter out. For safety and cost considerations, sensors are difficult to calibrate. Operators perform mental adjustments to sensor readings to account for calibration offsets. They follow statistical process control guidelines to isolate plant components that are out of control. AshMod automates this monitoring task, to reduce the cognitive load on the operator.

Operators recognise process trends and perform fault detection by using associations between trend patterns and faults. They use knowledge about cause and effect relationships to diagnose the root cause of detected faults. Operators have knowledge about the action needed to correct an identified malfunction. AshMod encapsulates this knowledge and assists the operator in performing quick and consistent fault detection, diagnosis and removal. Operators identify opportunities for fine tuning and optimising the process. AshMod assists the operator in this task by providing optimisation recommendations.

3. AshMod Overview

Conceptually, the design of a knowledge based system involves the development of three components. The knowledge base stores the domain knowledge models. In AshMod, these consist of the Goal Tree, Success Tree, Fault Cause Network, Plant Schematic and the Plant Component Hierarchy. These are described in Section 4. The inference engine contains inferencing
mechanisms and control knowledge needed to manipulate the knowledge models. The human interface provides mechanisms for knowledge acquisition and for operator interaction.

Operator knowledge is lost when an operator retires. This knowledge loss is prevented by capturing it in an OGS. Due to differences in levels of expertise, operators introduce quality variability into the process. By providing consistent recommendations, an OGS reduces this variability, hence improving product quality. During process upsets, operators have to digest large volumes of sensory data and take prompt corrective action. An OGS acts as an assistant, thus reducing the cognitive overload. Fault related shutdowns result in loss of production. Such shutdowns can be prevented by the timely identification of process upsets by an OGS. An OGS can help continuously improve and optimise a process thus increasing product quantity and quality.

An overview of AshMod’s functionality can be provided by means of an example. For each plant component, AshMod monitors a set of process indicators in real time. By applying SPC criteria on these indicators, AshMod is able to determine whether a plant component is in control. The process component in the schematic workspace, as well as the goals associated with this component, are highlighted in red when the component fails to satisfy the SPC criteria. AshMod performs trends analysis on the process indicators associated with the suspected components, to identify process faults. The fault cause workspace is displayed and the identified faults are highlighted in red. For every identified symptom, there are many faults and malfunctions that could have been the cause. Prior probabilities are used to identify the most likely cause which is highlighted in blue. Messages are sent to the message workspace to inform the operator about the suspected components, the identified faults and the potential faults and malfunctions. The operator is kept in the loop by allowing him to accept or reject an AshMod recommendation. When the operator accepts a potential fault it becomes an identified fault, and AshMod examines its possible causes. This process continues till a malfunction message is accepted by the operator to indicate that he agrees that the malfunction was the root cause of the identified fault(s). AshMod then displays the recommended action that the operator needs to perform to correct the malfunction.

If all plant components satisfy the SPC criteria then there is no need for fault diagnosis, and AshMod turns to optimisation. The plant optimisation workspace is displayed. Here, the current operating point of each plant circuit is compared against the target. The circuit with the largest ‘optimality coefficient’ is the one that is furthest away from the target operating point and offers maximum scope for improvement. Hence this circuit is highlighted in red on the optimisation workspace, and the appropriate perfective action needed to adjust the circuit setpoint is displayed to the operator.
4. Multiview Domain Representations

Application domain knowledge consists of concepts, relations, and facts captured in knowledge models. Control knowledge describes the inference structures used by tasks which reason with application domain knowledge. The separation of application domain and control knowledge enhances modularity and enables reuse. Control knowledge and the knowledge model structures are domain independent. Only the content of the knowledge models is domain dependent. Hence much of AshMod can be reused in future OGS developments in BHP operated coal washeries, sinter plants, blast furnaces, coke ovens etc.

The focus of this section is the application domain knowledge models. The control knowledge is covered in Section 5. Multiview domain representations [12][13] use different knowledge models together with a whole-part decomposition. Knowledge models are viewed by operators at different cognitive levels. The whole-part decomposition allows an operator to focus attention on the knowledge model at the appropriate cognitive level.

Subsection 4.1 details the domain knowledge models developed for AshMod. These are the means-end model, the structure model, the heuristic model and the behaviour model. Subsection 4.2 details the plant hierarchy as a whole-part decomposition.

4.1 Knowledge Models

![Diagram of Knowledge Models]

Figure 2: The Relationship between different Knowledge Formalisations

This subsection details the different formalisations of the coal washing domain knowledge. The purpose and function (Figure 2) reflect the intentional aspects of the plant and its components, while the behaviour, structure and heuristics
reflect the causal aspects [12][13]. An object’s purpose captures human intent in designing an object. How the object acts in response to its environment is captured by its behaviour. Structure is the organisation or arrangement of constituents of the object. Function is what an object does to fulfil its purpose. Heuristics capture means of approximating object behaviour where accurate structure models do not exist or where it is too complex to derive behaviour from structure. Hence structure ‘exhibits’ behaviour, heuristics ‘approximate’ behaviour, behaviour ‘achieves’ function, and function ‘enables’ purpose. Subsections 4.1.1 to 4.1.4 detail the specific knowledge models developed for AshMod.

4.1.1 Means End Model (Goal Tree Success Tree)

The Goal Tree Success Tree (GTST) [14] represents the mental model used by plant superintendents and operators. The upper section (goal tree) consists of goals that capture purpose, while the lower section (success tree) consists of the success criteria that captures function (Figure 3).

![Figure 3: The ‘Large Coal Circuit’ Subtree from the AshMod GTST](image)

Each goal in the goal tree is divided into subgoals that are necessary and sufficient for its achievement. Each goal in the success tree has a list of sensors, operator activities, events and calculated parameters that effect the success or failure of the goal.

4.1.2 Structure Model (Plant Schematic)

The Plant Schematic (Figure 4) is a Structure Model in which the connectivity of plant components and the flow of information, control and material is captured graphically.

4.1.3 Heuristic Model (Fault Cause Network)

Due to the mechanical complexity of jigs, cyclones, centrifuges and flotation cells, they do not have accurate structural knowledge models. In practice, operators use heuristic models to perform diagnosis. A fault cause network
(Figure 5) provides a heuristic means for identifying the root cause of an observed symptom.

![Figure 4: AshMod High Level Schematic](image4.png)

![Figure 5: Fault Cause Network for the Small Coal Circuit](image5.png)
4.1.4 Behaviour Model (Signed Directed Graph)

Qualitative reasoning techniques utilise physical models of process knowledge. Signed directed graph models (Figure 6) allow AshMod to understand the relationship between process variables.

Empirical models are based on statistical observations of behavioural relationships between plant components. For example, AshMod uses the following empirical model: fine coal tons per hour = 0.025 * (conveyor belt speed * depth of fine coal on conveyor belt).

Mathematical models are based on simplified representations of plant components. The following model captures the relationship between target ash and target yield [1]. $R_c = Y_p \times \frac{(100 - P_A)}{(100 - F_A)}$ where $R_c$ = Carbon Recovery, $Y_p$ = Target Yield, $P_A$ = Target Ash, $F_A$ = Feed Ash.

4.2 Whole - Part Decomposition

A whole - part decomposition (Figure 7) captures the is-part-of relationships between plant components.

5. Task, Inference and Strategy Representations

The KADS library [10][11] provides a generic set of interpretation models. These models are high level descriptions of various tasks. The knowledge engineer may select suitable task models from this library and instantiate them to the specific domain. A subset of the collection of KADS tasks (Figure 8) shows that many of the tasks performed by AshMod (Figure 9) are supported by the KADS library. The inference structures detailed in this section have used the framework provided by the KADS interpretation models.

Information flow between AshMod tasks and subtasks can be illustrated through the KADS Communication Model (Figure 10).
Figure 7: Part of the Plant Component Hierarchy

Figure 8: The KADS Classification of Generic Tasks
Operator Guidance

Monitoring
Validation
Calibration
Filtering
Trending
Classification

Diagnosis
Fault Detection
Fault Diagnosis
Fault Removal

Optimisation
Plant Target Computation
Circuit Target Evaluation
Strategy Presentation

Validation Filtering        Fault Detection     Plant Target Computation
Calibration          Trending        Fault Diagnosis     Circuit Target Evaluation
Classification        Fault Removal         Strategy Presentation

Figure 9 : AshMod Tasks and Subtasks

Monitoring

hard & soft
sensor data
validation
validated data
filtering
filtered data
calibration
calibrated data
trending
data & trends
classification
data, trends & states

Optimisation

plant target computation

Diagnosis

fault identification
circuit target evaluation
fault diagnosis
strategy presentation

fault removal

root cause

Figure 10 : AshMod Communication Model
5.1 Monitoring

The monitoring task compares domain object properties against expectations and looks for anomalies [10]. AshMod employs a data driven or forward chaining strategy with the monitoring task. The results of the monitoring task are immediately made available to the diagnosis and optimisation tasks.

5.1.1 Data Validation

To identify sensor failure, AshMod validates raw data before it is preprocessed and passed on to other tasks. The following G2 rules are used for validation:

1) for any gfi-quantitative-variable v if abs(the current value of v - the aim of v) \( \leq 3 \times \) the stdev of v then conclude that the validated-value of v = the current value of v and conclude that the validity of v = the symbol valid

2) for any gfi-quantitative-variable v if abs(the current value of v - the aim of v) > 3 \times \) the stdev of v then conclude that the validated-value of v = the aim of v and conclude that the validity of v = the symbol invalid

5.1.2 Noise Filtering

AshMod uses the exponentially weighted moving average (EWMA) filter which is a good filter in situations where the process mean moves slowly relative to the movements caused by measurement noise [1]. A benefit of using this filter is that it has an incremental formulation, so that the computation time required for real time filtering is not excessive. The current value of the process variable depends in an exponentially weighted manner on all the prior values of the variable.

\[ x_k = w \times x_k + (1 - w) \times x_{k-1} \text{ where } 0 < w < 1 \]

5.1.3 Calibration

Poorly calibrated sensors produce signal values that have a constant offset. AshMod averages out signal variability over a long time. If the sensor is correctly calibrated then this average would be the same as the target value of the process variable. The following G2 rule is used for calibration:

for any variable v unconditionally conclude that the calibrated-value of v = the validated-value of v - (the avg value of v during the last 6 hours - the aim of v)

5.1.4 Trend Detection

The detection of a process trend (Figure 11) is based on the first derivative over a recommended window. This slope is compared against a trend limit to determine an appropriate trend class. The qualitative features of process trends are provided to AshMod through the use of multiresolution analysis [2]. Multiresolution analysis uses wavelets to extract features, and represents these features using triangular episodes [19].
Figure 11: Inference Structures for Trend Detection

5.1.5 Classification
AshMod applies SPC criteria on each subsystem in the process. To do so, it examines the current as well as historical values and trends of all process variables. When the subsystem fails to satisfy the SPC, the diagnosis task is initiated to identify, diagnose and remove fault.

5.2 Diagnosis
AshMod performs diagnosis through causal tracing. AshMod identifies the components suspected to contain faults, the known and suspected faults (Figure 12), the root causes and the corrective action needed. The essential strategy employed with the diagnosis task is event driven.
5.2.1 Fault Detection
AshMod maintains a list of potential faults that can be detected by the system through the use of available process trends and states. A fault detection procedure is specified for each fault. For example, a fault S1 is detected using the following G2 rule: *if the level of smallb is high and the trend of smallb is increasing then conclude that the status of S1 is alarm*

5.2.2 Fault Diagnosis
Causal tracing uses the fault cause network to progress from an observed symptom through intermediate faults and causes, till the root cause is identified (Figure 13).

5.2.3 Fault Removal
Once the root cause of a fault has been determined, AshMod recommends an appropriate corrective action to the operator. Due to time constants, it takes time before an operator action has an observable effect. During this period the same action recommendation is not repeated.

5.3 Plant Optimisation
At the coal washery, optimisation is seen along two dimensions. First, the product ash must be kept within a specified range. To maximise the yield, it is desirable to operate all the plant circuits at the target ash all the time. The larger the product ash variation between circuits and over time, the greater the loss in product yield. Second, assuming that the product ash is on target, the product yield should be maximised. This optimisation objective is in fact indirectly satisfied by the fulfilment of the first objective. If all plant circuits are producing ash at target, then the yield will automatically be maximised.

To achieve this objective, AshMod must first determine the optimum operating point of the plant. Thereafter, the optima for each of the six circuits (the large, small and fine coal circuits of plants B and C) must be evaluated. Finally an optimisation strategy is determined and presented to the operator (Figure 14). This strategy identifies the circuit whose setpoint the operator
should modify. The optimisation subtasks are elaborated in following subsections.

5.3.1 Plant Target Computation

The optimisation function has only one decision variable. Achieving the ash aim automatically results in maximum yield. The plant ash is a function of the yield and vice versa. A similar relationship exists for each circuit within the plant. The two decision variables become one. The plant optimum must lie on the target ash line (Figure 15). The actual position of the optimum depends on the feed quality.
5.3.2 Circuit Target Evaluation

Based on the feed quality and the plant optimum, we can estimate the ash and yield target for each circuit (Figure 15). The determination of the circuit target states is essential, because each of the individual circuits must be optimised to optimise the entire plant. The current state of each circuit is compared against its goal state to evaluate its performance. The circuits that are performing poorly are the suitable candidates for optimisation.
5.3.3 Strategy Presentation

The optimisation strategy synchronises the twin circuits (eg Jig_B and Jig_C) (Figure 15). For optimum operation both plants must operate in synchronisation (at the same operating point). Once all circuits are synchronised, they are jointly pushed towards the target. For each circuit an optimality coefficient is defined by the expression (| target yield - current yield |) / (target yield). The circuit with the largest optimality coefficient is optimised first, since it provides maximum potential for improvement.

6. Validation and Maintenance

The capture of knowledge through knowledge models and its subsequent graphical representation in G2 has allowed operators to understand the knowledge captured within the OGS, to comprehend the reasoning process used by the system, and hence to validate and correct the knowledge. The systems reasoning processes are made transparent through knowledge model animations. Goals in the goal tree and nodes in the fault cause matrix are highlighted to allow the operator to understand the logic being applied by the system. The integrated environment provided by G2 has allowed the knowledge engineers to extend and modify the knowledge models with ease.

Domain experts often find it difficult to ‘trust’ operator guidance systems because they are unable to ‘visualise’ and validate the knowledge stored in the OGS. In AshMod, we have made sure that the plant operators are kept in the loop in the early stages of the implementation so as to allow them to become familiar with the knowledge models, and to satisfy themselves of the validity of the encapsulated knowledge. As operators gain confidence in AshMod’s abilities, they can be taken out of the loop, allowing them to concentrate on finer issues. Once validated, the system need only present final recommendations to the operator. Explanations can be provided if the operator explicitly asks for justification.

It has been a goal of AshMod to encourage the end user to perform as much as possible of the simpler fine tuning and knowledge maintenance tasks. The G2 natural language interface and the graphical object oriented knowledge models make this relatively easy for computer literate operators and plant superintendents. This makes it substantially easier for the end users to experiment with the system, thus gaining a better understanding of the encapsulated knowledge.

7. Conclusions

AshMod bridges the gap between theory and practice by applying the sound knowledge modelling principles propounded by the KADS methodology in an OGS that will operate in an industrial environment. AshMod is perfectly suited for implementation on G2 which supports the development of object oriented knowledge models [7][8]. For this reason, we chose to use G2 as the expert
system shell for the development of this OGS. AshMod is currently undergoing offline testing and validation.

8. Future Work
We hope to achieve the following in the next phase of development:

• Gain access to a larger number of sensor data values and integrate these inputs into the OGS.
• Run the OGS using live on-line data.
• Validate and modify the knowledge models.
• Develop an integrated uncertainty management approach into the OGS.
• Develop detailed and comprehensive qualitative models and put them into a framework that allows integration with more quantitative approaches.
• Look into the safety, economic, environment and moisture related goals of the coal washery.
• Investigate automated learning and pattern recognition strategies through the application of machine learning and neural networks.
• Integrate multiresolution analysis [2] (a method for the extraction of quantitative and qualitative knowledge from process trends) into the OGS.

Acknowledgments
This work was supported by BHP Research, Newcastle Labs, Australia and by the Centre for Industrial Control Science at the University of Newcastle, Australia. We thank Gensym for providing excellent support for G2. Most importantly, we express our appreciation to John Ellison, Rob Sedmak and Steve Single at BHP SPPD for being patient during the knowledge acquisition exercises and in awaiting the results of this effort.

Key Words
operator guidance system, process monitoring, fault diagnosis, plant optimisation.

References


