The Architecture of a Knowledge Based Controller for Application to Glass Melting Furnace Operation

P. Lima, P. Oliveira, C. Ferreira, J. Sentieiro*

December 13, 1990

Abstract

A Knowledge Based Controller (KBC) specially designed for the operation and control of Glass Melting Furnaces is presented in this paper. The KBC, developed under the ESPRIT project AIMBURN, implements the first two levels of a hierarchy consisting of (from more abstract to more precise decisions) Organization, Coordination and Execution levels. The Coordination level and the related processors are described with some detail. A Data Base of general application in Knowledge-Based Control of multi-loop processes is also presented.

1 Introduction

The operation and control of large scale industrial processes such as water treatment, cement or glass processing is usually a hard task, because it involves several control loops (air, fuel, pressure, temperature) resulting in coupled multivariable systems, with difficult mathematical modeling. The solutions usually implemented are based on conventional controllers (e.g. PID), manually tuned or autotuned for each control loop. The systems are operated by experienced operators who modify the set points for each loop in order to maximize efficiency and final product quality.

Controllers based on expert system shells or fuzzy rules have emerged in the last years [Årzén89, Procyk79, Mandani77]. Advantages in terms of efficiency and final product quality have been reported for several practical applications [King88].

The main reason for these improvements lies in that Knowledge Based Systems take into account not only overall performance improvement (e.g., maximizing heat transfer to glass), but also the handling of exceptions. For example, if the heat transfer to the glass is maximized, but there is unmelted glass in certain areas due to an improper temperature profile, the main goal of the system must change, in order to deal with this problem, even sacrificing the max-heat transfer goal.

The purpose of this paper is to describe with some detail an architecture for a Knowledge Based System (KBS) suited for the operation of glass melting furnaces, proposed under the EEC ESPRIT II project AIMBURN – "Advanced Intelligent Multi-Sensor System for Control of Boilers and Furnaces".

The architecture of Knowledge-based Controller (KBC) is the result of an extension of the usual KBS components (Data Base, Rule Base and Inference Engine) to a set of processors [Doraiswami89].

The proposed control architecture for the glass melting furnace is based in the three-level (Organization, Coordination, Execution) hierarchical model proposed by [Saridis89] for the architecture of Intelligent Machines (Figure 1). The KBC includes the Organization and Coor-

---

*Centro de Análise e Processamento de Sinais and Laboratório de Robótica e Processamento de Informação, Technical University of Lisbon, Av. Rovisco Pais, 1096 Lisboa Codex, Portugal.

dination levels, the Execution level being implemented by PID controllers where the set points for each control loop are provided by the Coordination level.

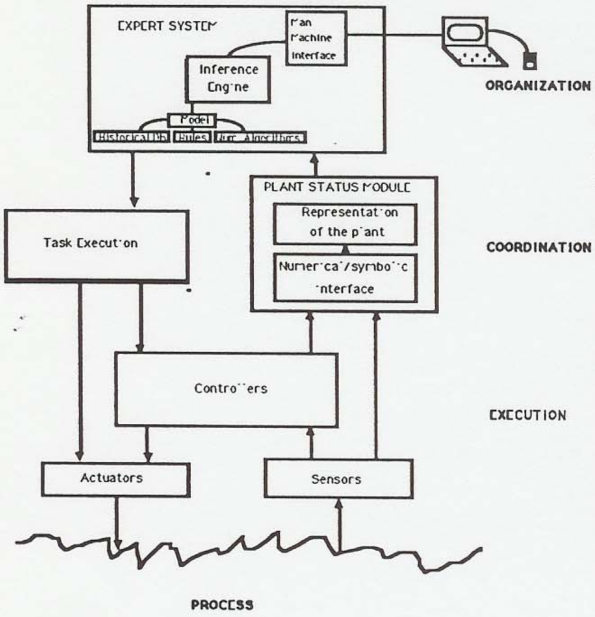

Figure 1: The 3-level conceptual diagram of the whole control architecture.

Description details for each KBC components are given in section 3, following a brief overview of the main problems in the operation of glass melting furnaces, given in section 2. Section 4 presents a descriptive example of an implementation for a subsystem of the glass furnace operation and control, which uses a vision sensor and actuates the fuel burners using differential control. In section 5 are described the remaining steps to be followed in order to achieve a full implementation and test of the whole system.

2 Operation of Glass Melting Furnaces

A glass furnace usually includes 3 areas:

- Glass Melting Area;
- Working Area;
- Fining Area;

The raw material (batch) is fed to the furnace and is transformed into molten glass in the Glass Melting Area. The homogenization of the glass occurs in the Working Area, and finally its temperature is slowly lowered in the Fining Area, where the glass softly flows to the furnace outlet.

The main goal in the operation of Glass Melting Furnaces is to achieve final glass quality, minimizing fuel consumption. To accomplish this, correct control of the temperature profile along the furnace is mandatory. However, temperature control is not enough since several factors can disturb the process and hence have to be taken into account [Pincus80]. Some examples of such disturbing factors are:

- Inadequate batch composition leading to incorrect glass color. Air/fuel ratio must then be re-tuned for the new conditions, and coloring materials may have to be introduced;
- Changes in molten glass level inside the furnace may introduce impurities in the final glass, or prevent it from reaching the outlet zone of the furnace, then the batch input rate must be controlled by the glass level;
- The air pressure inside the furnace lower than the outside pressure allows the leaking of outside cool air having as a result the decreasing of the temperature. Thus, there is a need for a pressure control loop.

The management of all these loops and disturbances is usually the responsibility of trained operators, who can control the process not only using the information supplied by the flow, the temperature, the pressure, and the glass level meters, but also using visual information about flame quality, provided by video cameras. Given that information, they may decide to change set-points and air/fuel ratios, or add colorings, to assure regular functioning.

Video cameras also allow the monitoring of the location and motion of unmelted materials which may be located in, or moving towards forbidden regions. A barrier of air bubblers is used to prevent the flow of those aggregates of unmelted material to reach the furnace outlet. The control of the bubblers air flow can take into account the vision system information.

The AIMBURN project includes a Vision System that deals with that sort of situations.

3 Extending Conventional KBS Architecture for Control Purposes

The conventional KBS components, usually including an **Inference Engine** operating over a **Data Base** of symbolic data by a set of **Production Rules**, are not sufficient for operation of real dynamic systems, no matter how sophisticated the knowledge representation/reasoning may be (*predicate calculus, frames, semantic networks*). This is due essentially to three reasons:

1. The nature of the acquired data from the process is essentially numeric (although some qualitative information can be directly provided by specific wide-sense sensors as a colored vision-based sensor) whilst a KBS deals with symbols;
2. The system is dynamic, so there is a temporal/historical component difficult to represent in the conventional Data Base;
3. Decisions based on symbolic reasoning cannot be directly provided to the low-level control loops, as they deal with numerical values characterizing set points and other parameters.

The KBC must then incorporate not only an **Organization** level of decision which essentially consists of a conventional KBS, (where symbolic actions are determined in response to specific qualitative conditions of the process), but also a **Coordination** level acting as an interface between the higher abstract level of decision and the lower-level controllers.

The Coordination level can be divided in two parts: the **Numerical–Symbolic Interface** which converts numerical data into symbolic information, by sensor integration and/or classification based on Pattern Recognition and simpler techniques; and the **Task Executor**, which translates qualitative high-level orders into several data (set points, controller gains) for the **Execution** level.

This intermediate level provides the KBC with the information that can be used to overcome some of the limitations of the KBS described before:

1'. The Numerical/Symbolic Interface is composed of several processors which extract features of the numerical data from a sensor or group of sensors and translates that data into symbolic information by posterior classification in one of previous selected classes.

2'. The information about past values of the process variables is implicitly used in some of the processors referred in 1'.

3'. The Task Executor receives as input qualitative actions to be taken and executes those orders sending set points and controller gains to the Execution level, thus performing a coordination task.

Figure 2 displays a block diagram of the knowledge-based controller, where it can be noticed that the Numerical/Symbolic Interface contains itself two sub levels in correspondence with two sets of pre-processors: the *pre-processors* and the *specialized processors.* Each of the two sets is described in the next

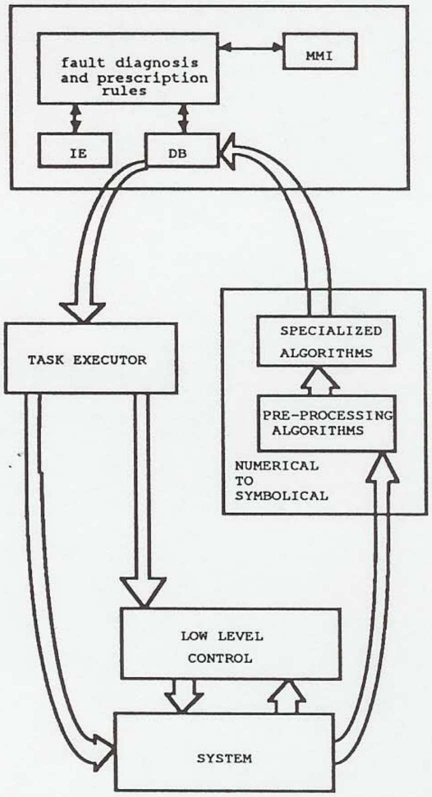

Figure 2: More detailed architecture for the glass melting furnace operation and control system

subsections. A presentation of the Data Base designed for the KBC is also included. This Data Base has a structure which can be generalized to processes other than glass melting furnaces.

The KBC is continuously monitoring the furnace state by polling the outputs of all preprocessors in order to detect fault conditions.

Every time a fault[1] is detected, the corresponding set of specialized processors is executed, in order to determine the error values, the frequency of the oscillations, the features related to unmelted glass parts (position, area, orientation) or the flame features (length, width, area).

After all necessary specialized processors were run, a set of forward/backward inference rules, which perform fault diagnosis and prescription is activated. These rules decide, based on the results of the Specialized Processors, whether the fault is due to a damaged actuator or sensor, a mistuned controller, or to an inadequate set point. These are the main causes of problems, but others can be identified, such as bad batch composition or unbalanced control of the different burners.

The syntax of the rules is the following:

IF
contingency
THEN
prescription

for *forward inference rules*, which prescribe an action to be taken, given certain *contingency*. Contingencies are tested by the *backward inference rules* which have the general form:

contingency
IF
results of specialized processors
match predefined conditions
of abnormal operation.

Underlying the whole automatic supervision, the sensor values and actuator outputs are periodically updated. Different control loops, including the so-called vision loops have different sampling periods.

3.1 Pre-Processors

Pre-processors inspect process inputs and outputs, loop errors and others, in order to detect fault conditions, such as saturation in the actuators.

A non exhaustive list of pre-processors for this particular system includes:

---

[1] Here and henceforth, *fault* stands not only for failures such as damages in sensors, but also for situations such as saturated controllers, output oscillations or improper dynamic behavior of the system.

- Detection of output oscillations and sudden transitions, as well as growing static error for the following process loops: fuel flow, air flow, crown temperature, pressure inside furnace, glass level, fuel viscosity;

- Checking of the crown temperature values in three different locations along the furnace, in order to keep temperature profile within tolerable limits;

- Checking of the presence of unmelted glass in areas too far from the back wall, of inactive bubblers and/or burners,using data provided by the Vision System;

- Asking operators for defective glass in the output (which is a task not intended to be done automatically);

3.2 Specialized Processors

When requested by the triggering of a particular pre-processor, information about the state of some plant entities is returned.

For example, in response to the detection of oscillations by a pre-processor, a specialized processor determines the frequency of oscillation, which together with the static information of the Data Base (such as if the sensor is wide band or narrow band), can determine the source of the problem.

The specialized processors allow to determine:

- when *oscillations* in control loop l are detected: determination of oscillations frequency, checking of the last values for control, set point and output signals, comparison between actuator output and actuating signal, estimation of parameters for a model of the process loop l and test the step response of the model in terms of overshoot and rise time;

- when *static error* in process loop l is detected: getting sensor characteristics from the Data Base, determination of the average error for loop l in the last sampling instants, determination of a loop model loop and test its step response, and getting control, output and set point history;

- when *sudden transition* in the output of loop l is detected: determination of transition slope and loop model, and what parameters of the model, if any, suffered the abrupt transitions – this can permit the detection of several faults and its origin;

- when *unmelted glass in dangerous areas* is detected, getting features such as unmelted glass position, area, eccentricity and present rhythm of batch feeding;

- when *defective glass* is reported, checking for the type of defect (blisters, cords, devitrification, change of color) and according to it, checking flame features (length, width, area, distance to burner), temperature profile, batch composition and/or claimed glass color.

3.3 Task Executor

The main task to be performed by processors implementing algorithms belonging to this class is the dual of the Numerical/Symbolic Interface task.

Given as input qualitative actions to be performed, numerical values of variables are provided to the control loops which implement these actions. Set points and controller gains are typically the kind of parameters changed by the output of the Task Executor (TE).

TE processors include actions such as tuning the gains of a PI controller given the step response features of the present model for the process [Oliveira90], correction of temperature profile, ordering the addition of colorings or fining products, and in general, adjustment of set points for the different loops.

3.4 Data Base

The processors described in the previous section deal with a reasonable amount of information both static (characteristics of sensors and actuators, dimensions and divisions of the furnace, units of measured values) and dynamic (control loops outputs, control and set point signals, PID gains). This justifies the need for a Data Base (DB) descriptive of the furnace control system, while specialized processors use data store in the DB by the acquisition task to generate more elaborated data for the rules. Task executor processors modify the data values, which are later put in the actuators by the actualization task.

The present state of the DB can be seen in figure 3, using the stipulated representation of the Expert System development shell KEE. In the figure, classes, subclasses and its instances (all called *units*) can be recognized. KEE represents class/subclass relations by solid lines and class/instance relation (or *member of*) by dashed lines. Each unit has its own *slots* not represented in the figure, and where each one represents a distinct attribute of the unit. An inheritance mechanism from class to subclass to instance is provided. It determines, if allowed, which values of a certain slot of a given class are automatically values of its subclasses as well as the respective instances slots.

From the figure, it can be seen that a general purpose DB for knowledge-based process control can result from this particular application. It has 5 main classes:

- **control loops**, describing the loops present in the installation, each with its controller, sensor, actuator and actuating subsystem;
- **actuating subsystem**, describing pumps, feeders, motors and other devices associated to actuators;
- **actuators**, more related to servomechanisms such as motorized valves or on-off devices;
- **sensors**, including both "conventional" sensors and more sophisticated ones, such as vision cameras associated with image processors;
- **controllers**, in this case PID's, for each of which the attributes are, for example, the proportional, derivative and integral gains.

Implicit in this representation is a relation *part of*. Each sensor, controller, actuator and actuating subsystem is *part of* a control loop. The relation is represented using the unit values of slots as instances of the control loop.

In this application, an additional class contains static data descriptive of the furnace.

4 Example

A subsystem of the Glass Melting Furnace presently under test is based in the architecture described before. It obtains information from the image processors about unmelted glass parts, stores that information in the DB and then a set of prescription and diagnosis rules balance the amount of fuel for each of the burners of the system[2] in order to melt the glass in the zone where it is unmelted.

The features provided by the processing of inside-furnace images are the area and mass center of the whole set of unmelted glass parts, a measure of its eccentricity in 2-D (a circular zone of unmelted glass parts is different from a long one), and the radius in the longitudinal direction of the furnace (which measures the extent of the danger of reaching the neighborhood of the bubblers).

Each time an Image Pre-Processor detects unmelted parts in a dangerous zone (usually the working area), it first activates some specialized processors which determine information such as area, radius, mass center and eccentricity of unmelted glass parts and then a set of rules in the form:

---

[2] There are 6 burners, installed in the back wall, divided in 2 groups of three, fired alternately each 20 minutes.

```
          Significant unmelted glass
          in middle-left with area ?a
     IF
          area ?a of unmelted parts is $> \alpha$
          mass center is working area left
          eccentricity is $< 0.5$
          and radius is $> \rho$
                        and
IF
           Significant unmelted glass
           in middle-left with area ?a
THEN
           Call Heat Transfer Evaluator
           to determine necessary
           amount of fuel $\epsilon$ to melt
           a portion of glass with area ?a
           Fire left burner with a quantity of
           $+\epsilon$ fuel and right burner
           with $-\epsilon$ fuel to maintain
           the balance of fuel and
           melt left zone glass.
```

The result of rule application is stored in Data Base. Then, the Task Executor refreshes the numerical inputs of the Execution Level, given the symbolic information stored by the Organization Level.

5 Conclusions and Future Trends

The architecture for a Knowledge Based Controller for a Glass Melting Furnace has been presented. The overall control system of the furnace is based in an hierarchy of three levels, called **Organization**, **Coordination** and **Execution**, from the more abstract to the more precise. The KBC includes the first two levels, and the Execution level is implemented by conventional control loops.

The Organization level consists of an Expert System containing operation rules of a glass furnace, normally used by specialized operators and process engineers, while the Coordination level provides the interface between the abstract symbols handled by its upper level and the numeric data used in the Execution level (set points, PID's gains, sensor measures).

Some processors to be included in the Coordination level have also been described. They are divided in Pre-Processors, Specialized Processors and Task Executors. Their task includes the handling of the dynamic data and coordination of low-level loops from the high-level symbolic orders.

A Data Base of a more general application in Knowledge-Based Process Control, its structure and the corresponding interaction with the mentioned levels, has been presented.

The architecture presented, although proposed and being implemented in the ESPRIT project AIMBURN is still conceptual, and no results but an example of application being developed have been presented. The example shows the advantages of dealing with a major amount of information such as the data provided by an image processor, using this kind of approach.

Future work will necessarily include the development of a furnace model in order to test the whole control architecture proposed. Modifications of the Data Base are expected, while testing is in progress, and also the inclusion of new processors in the Execution level, and correction of the present ones. However, this conceptual step led to the establishment of the framework within which the whole system will be developed.

References

[Årzén89] Årzén, K.-E. (1989). "An Architecture for Expert System Based Feedback Control". Automatica, vol 25, n 6, pp 813-817.

[Åström86] Åström, K. J.; Anton, J. J. e Årzén, K.-E. (1986). " Expert Control". Automatica, vol 22, n 3, pp 277-286.

[Doraiswami89] Doraiswami, R. and Jiang, J. (1989). "Performance Monitoring in

Expert Control Systems". Automatica, vol 25, n 6, pp 799-811.

[King88] King, R. E. e Karonis, F. C. (1988). "Multi-Level Expert Control of a Large-Scale Industrial Process". Fuzzy Computing, pp 323-339.

[Mandani77] King P. J., Mandani E. H. (1977). "The Application of Fuzzy Control Systems to Industrial Processes". Automatica, vol 13, pp 235-242.

[Oliveira90] P. Oliveira; P. Lima and J. Sentieiro (1990). "Fuzzy Supervision of Direct Controllers", Proceedings of the V IEEE International Symposium in Intelligent Control, Philadelphia.

[Pincus80] Ed. Pincus, A. G. (1980). " Melting Furnace Operation in the Glass Industry". Books for Industry and Glass Industry Magazine.

[Procyk79] Procyk, J. J. e Mandani, E. H. (1979). "A Linguistic Self-Organizing Process Controller". Automatica, vol 15, pp 15-30.

[Saridis89] Saridis, G. (1989). " Analytic Formulation of the Principle of Increasing Precision with Decreasing Intelligence for Intelligent Machines". Automatica, vol 25, pp 461-467.

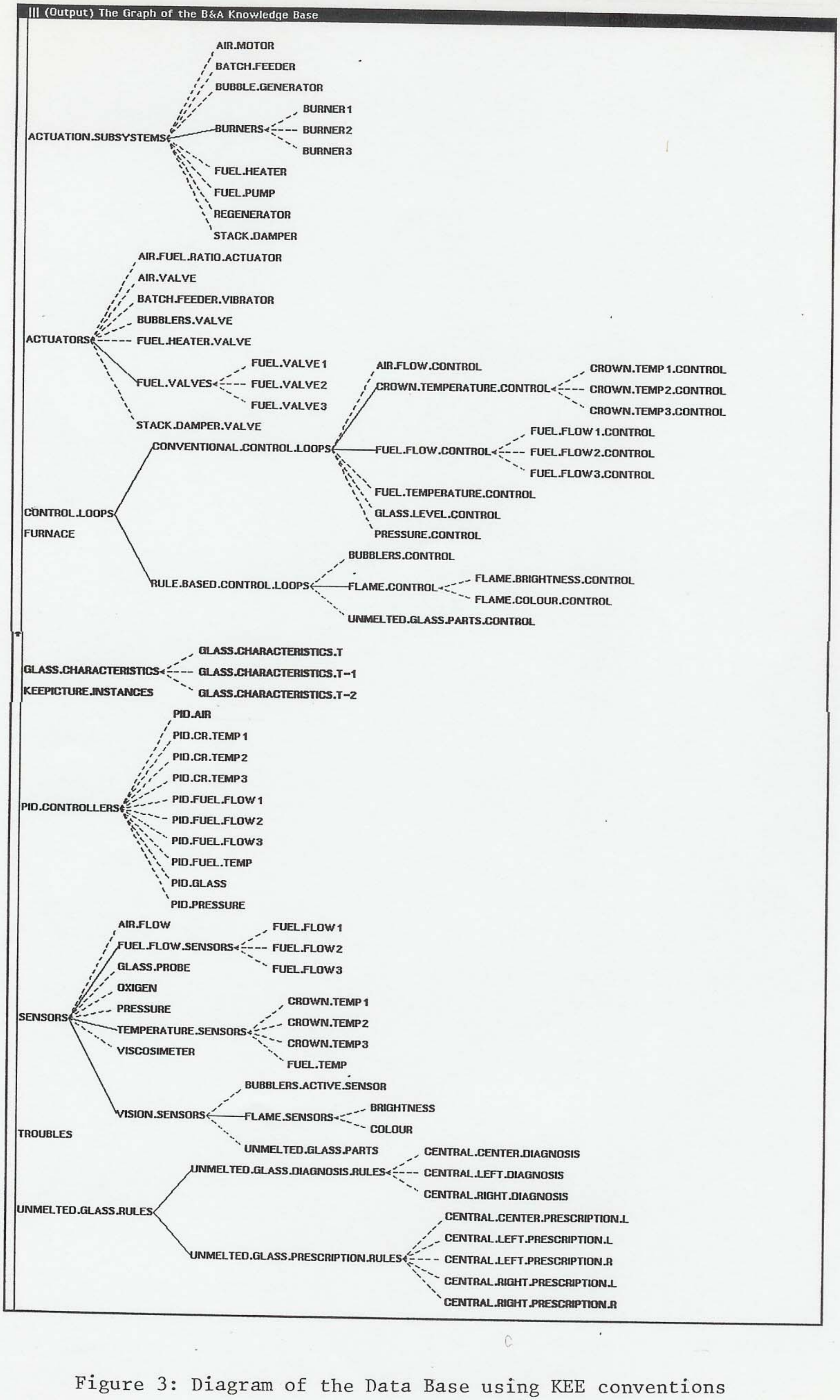

Figure 3: Diagram of the Data Base using KEE conventions