KNOWLEDGE-ENHANCED CO-MONITORING IN COAL-MINES

Wolfram Burgard[1], Armin B. Cremers[1], Dieter Fox[1], Martin Heidelbach[2], Angelica M. Kappel[1] and Stefan Lüttringhaus-Kappel[1]
[1]Institut für Informatik III, Universität Bonn, Römerstr. 164, D-53117 Bonn
[2]Ruhrkohle Bergbau AG, T 1.2, Shamrockring 1, D-44623 Herne
E-mail: {wolfram,abc,fox,heidelba,angelica,stefan}@cs.uni-bonn.de

Abstract

Detection of underground fires is an important security task in hard-coal mining. Automated fire detection systems are usually based on the monitoring of carbon monoxide (CO). Systems using conventional technology based on threshold and tendency observations, however, generate a large number of false alarms. We show how CO-concentrations can be forecast by appropriate models of the physical and chemical processes. We furthermore describe a rule-based specification system utilizing forecasting for CO-monitoring. The improvement of this approach over the conventional is threefold. First, the number of false alarms is reduced by 50%, at least. Simultaneously, the thresholds for warnings and alarms can be reduced so that, second, the detection of real fires becomes both quicker and more reliable. Third, heuristic rules for fire detection and suppression of false alarms as well as the control of the forecasting can be described in a declarative way. While our system is still in a prototypic stage, the three major German hard-coal mining companies decided to use our approach in their CO-monitoring systems.

1 Introduction

In hard-coal mining preventive techniques against fires and explosions have been very successful in the 20th century, especially in avoiding explosions by intrinsically safe (electrical) equipment and reducing damage by strong mine rescue parties. Moreover, fire barriers are installed all over the mines. Sufficient supply of fresh air avoids dangerous concentrations of methane (CH_4), while it also reduces the heat and simultaneously improves the quality of the air. However, due to the self inflammability of the sulphureous hard-coal fires occur in every modern coal mine. Furthermore, defect machines or parts of machines, as for example defective rollers of band conveyors could cause a smoldering fire.

Enormous technical effort is taken to detect fires as early as possible and to extinguish them before miners are endangered or substantial damage takes place. In order to enable a rapid detection, ventilation velocity, carbon monoxide (CO) and CH_4 concentrations are constantly monitored using several hundreds of sensors in an average coal mine.

While the ventilation velocity and CH_4-concentrations can be monitored not to exceed fixed security limits, the interpretation of CO-concentrations and CO-curves is quite more complicated. This is due to the fact that various activities carried out underground by the miners such as blasting operations or driving Diesel-powered vehicles produce similar amounts of CO as open fires. Moreover, processes outside the mine produce CO which, if drawn into the mine, increases the CO-concentration in a similar way as a smoldering fire. Such processes are smog or combustion processes, like cars on nearby roads, factories, and power plants.

In the beginning, CO was monitored by specially trained staff. Since the end of the seventies, the Ruhrkohle Bergbau AG uses process computers to control the CO-concentrations measured by the CO-sensors [SSSS82]. Such a process computer installed on a mine monitors for each CO-sensor the mean CO-values averaged over a time period of one minute. In this system [Kar87], apart from absolute thresholds increasing CO-concentrations and special properties of CO-curves are used as indicators of open or smoldering fires. Furthermore, special filtering heuristics are applied to detect CO-curves coming from blasting operations which otherwise would produce a fire alarm.

Figure 1 shows a typical line plot of the minutely averaged CO-concentrations at several CO-sensors for about 16 hours without fire (the y-axis is scaled in ppm/100 CO). On one hand, large variations can be observed, and on the other hand, a slope of just 6 ppm CO per 24 hours could indicate a smoldering fire.

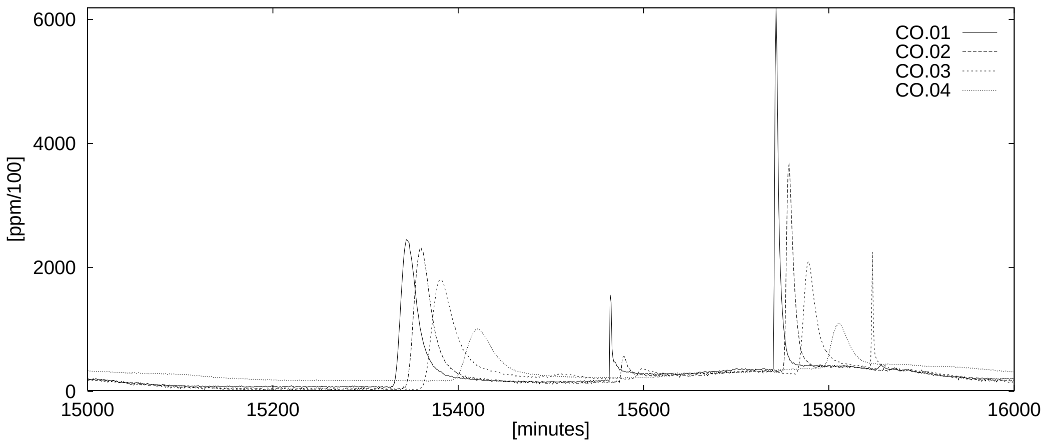

Figure 1: Typical CO-curves of different devices

To be on the safe side, thresholds for warnings and alarms in the current system have been set to very low values, causing a serious number of false alarms. The number of false alarms is further multiplied by the fact, that the clouds of CO (and other gases as well) are streaming with the air through the mine, causing series of similar (false) alarms at successive sensors. In all, there may be more than eighty warnings and alarms per day. The huge number of alarms are a critical factor burdening staff in the mine observation center. On one hand, a substantial amount of time is spent for checking CO-curves, or even for controlling the corresponding parts of the mine by the miners in order to verify the situation is really safe. On the other hand, the staff's sensitivity to actual fire events may be reduced.

In this paper we describe how the number of fire alarms can be reduced significantly by combining sub-symbolic and adaptive techniques with traditional knowledge representation methods. This is achieved by adaptive techniques forecasting CO-curves in the mine combined with a rule-based representation of heuristic rules for fire detection and the control of the forecasting. The next section describes the method to forecast CO-values. Section 3 illustrates the knowledge based part of the system. In Section 4 we describe how we integrate the forecasting and the rule-based part. Finally, in Section 5 we demonstrate the performance of the system by an application example.

2 Forecasting of Carbon Monoxide by Adaptive Techniques

In the fire detection system currently in use every CO-sensor is observed independently of the other CO-sensors. For this reason a warning or alarm has to be triggered whenever the concentration of CO reaches a fixed threshold. Because the air is streaming along the roadways, high concentrations of CO at a CO-sensor can be observed some minutes later at the following CO-sensor.

As an example consider the CO-curves in Figure 1. Near minute 15400, typical CO-curves of a single blasting operation measured at different devices located one after another in the air stream are shown. While the qualitative similarity between the different curves is obvious, the quantitative CO-curves do not match exactly. For the currently used system, much effort was spent to define heuristic classification rules to cope with this variety of CO-curves.

Our approach is to model the change of the CO-curves between two CO-sensors exploiting the similarity of the corresponding CO-curves. This allows us to predict the CO-curves at successor devices. For every time step we compute the expected CO-value of a CO-sensor based on the CO-curve of the preceding CO-sensor. For a reliable prediction it is extremely important to know the time delay of the air stream between such two devices as well as the mapping between the CO-curves. The time delay, unfortunately, is neither constant nor is it provided accurately enough by the monitoring system. We therefore developed an adaptive method that extracts this information from sample data given by the CO-curves of two adjacent devices (see Section 2.1). Concerning the mapping of CO-curves, it is obviously not possible just to shift the CO-curve of the first device by the time delay and to use these values as prediction for the second device. Our method to model this mapping based on sample data is described in Section 2.2.

For illustration we use the CO-curves of two adjacent

CO-sensors of the mine Blumenthal taken in March 1994. The devices were placed in an area of the mine with a lot of mining activities.

2.1 Determination of the time delay

To keep our method as adaptive as possible it only relies on the local topology of the mine, i.e. the predecessor/successor relation of the CO-sensors. All the information necessary to determine the time delay is contained in the sample data given by the CO-curves of the two corresponding CO-sensors. By repeated usage of this method on updated samples the time delay can be adapted to the changing environment in the mine. In the approach described in [Kar87] the determination of the time delay based on sample data depends on significant CO-values above 3.0 ppm. Especially on weekends or for devices far away from mining activities this method is not applicable. We therefore introduce a method extracting the time delay independently of such peaks.

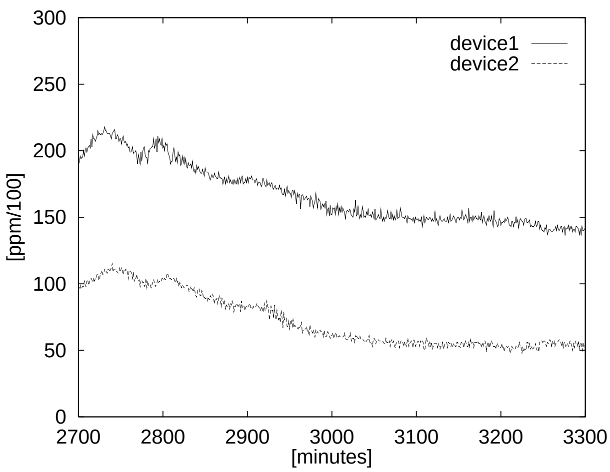

Figure 2: The noisy values are correlated

If we look at a sufficiently long time interval of ten hours as shown in Figure 2 a high similarity in the CO-curves becomes apparent even for values below 3.0 ppm. This similarity is due to the long term changes in the CO-curves. The high frequency noise is not correlated at all and has to be filtered out first. Assuming Gaussian noise in the high frequency we use a Gaussian filter to delete it. The result of this filter is given in Figure 3.

To extract the time delay from the data we only have to find the offset of the two time series leading to the best match of the values. For this task we have to explore a one dimensional search space to get the time shift with the best evaluation. To evaluate the quality of the match for each time shift we consider

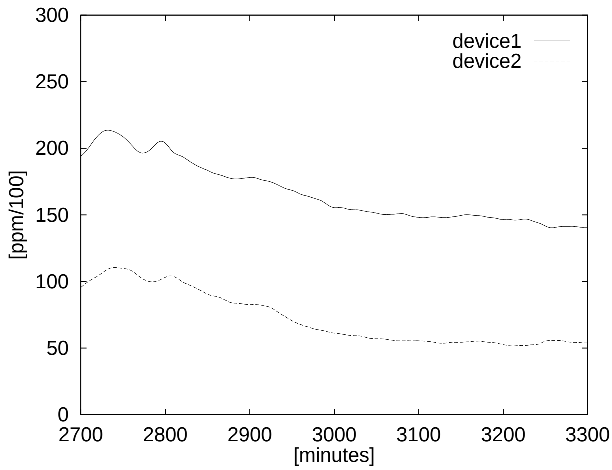

Figure 3: The values after Gaussian filter

the CO-values as n instantiations of two random variables. Thus we can use the cross-correlation between these values as evaluation function for the matches. The cross-correlation $r_{xy} \in [0;1]$, see equation (1), is a quantitative measure of the linear dependency between n values of two random variables X and Y [HEK89]:

$$r_{xy} = \frac{\sum_{i=1}^{n} x_i y_i - n\bar{x}\bar{y}}{\sqrt{\left(\sum_{i=1}^{n} x_i^2 - n\bar{x}^2\right)\left(\sum_{i=1}^{n} y_i^2 - n\bar{y}^2\right)}} \quad (1)$$

2.2 Approximation of the Mapping

The prediction of the CO-concentration at a CO-sensor is of course most important for high values. We assume the change of the CO-curves in time to be caused by two types of processes. On one side there are chemical processes leading to a loss of CO until the concentration reaches equilibrium at about 2 ppm. On the other side we have the physical process of whirling which aims at the same concentration of a gas over the whole available space. A CO-peak changes under the influence of the first process in a way such that the integral under the peak becomes smaller but the shape of the CO-curve stays constant. The process of whirling leaves the overall amount of CO unchanged but aims at a flattening of the CO-curve.

A mathematical model of these processes depends on the exact knowledge of all facts influencing loss of CO and whirling of air. Among many other influences there are the temperature of the air, the composition of gases in the air, the concentration of pulverized coal in the air and the exact geometry of the mine. These facts are not available and because of the mining activities the environment is in permanent change. For these reasons we decided to extract the parameters of the model of these processes from examples. This leads to a model which constantly adapts itself to the specific facts given in a certain place at a certain time in the mine.

In our simple model we assume a loss of CO linear with the CO-values. The whirling, in contrast, can be modeled as a Gaussian process [Ari56, Lit70, Tay53]. This leads to the formula given in (2) computing the prediction value p_t for the succeeding CO-sensor at time t.

$$p_t \quad = \quad a \left(\sum_{i=-n}^{n} \left(\frac{\omega}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}(\frac{i}{\sigma})^2} \right) c_{t-o+i} \right) + b \quad (2)$$

Mathematically the whirling process is computed by a weighted sum of the value itself and n preceding and succeeding values. The coefficients of this sum are given by a Gaussian probability density function with mean at the actual value. To ensure the commutativity of the whirling process and the loss of CO, the coefficients are normalized by ω to sum up to 1. The standard deviation σ of the Gaussian defines the strength of the whirling process. $C_{t-o-n}, \ldots, c_{t-o+n}$ are the corresponding CO-values of the preceding CO-sensor and o is the time delay between both measuring devices. Finally the decomposition process is modeled by a linear function $y = ax + b$ where a describes the decomposition in time o and b the average minimum CO-concentration.

The task is to adapt the parameters o, σ, a and b of (2) to a given set of samples. The sample consists of the CO-values of two adjacent CO-sensors taken over a period of one week. This leads to about 10000 values for each device. In order to map each value of the preceding CO-sensor on the corresponding value of the other device we first have to compute the time delay o of the air stream between the devices. This is done with the method described in Section 2.1. Given this value we still have to determine the remaining parameters of the two processes described above.

To extract the parameter σ for the whirling we perform a one dimensional search over different standard deviations of the Gaussian. A deviation of zero does not change the values at all and the higher the deviation the more the flattening influences the peaks. Thus, the starting point for the search is a standard deviation $\sigma = 0$. Having the linear loss in mind we want to find the deviation which leads to the most linear coherence between the values. This is why we use the cross-correlation as a measure for the best standard deviation. We regard a model of the whirling as the best one if it yields the highest cross-correlation value for the resulting curves.

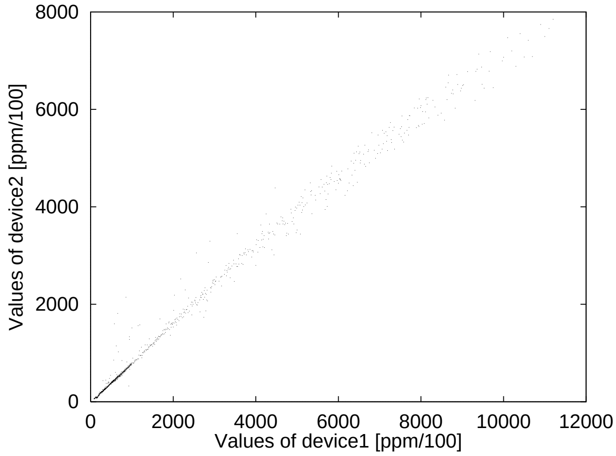

Figure 4: Mapping of shifted raw data

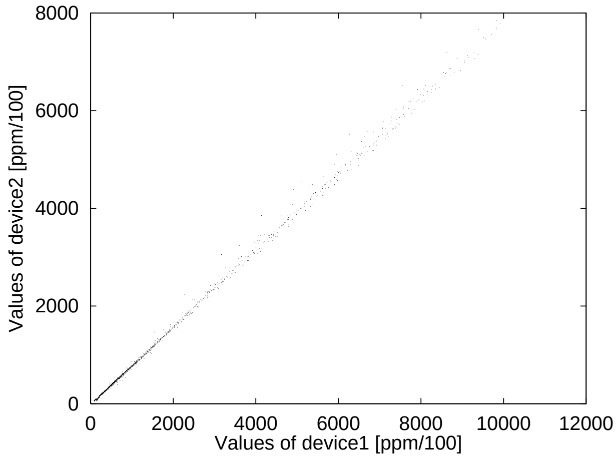

Figure 5: Mapping after Gaussian whirling

After having found the parameters of the whirling model for the two devices, we just have to apply this model to the CO-values of the first device and map the resulting values to the values of the second device, shifted by the time delay o. Such a mapping is shown in Figure 5. To demonstrate the influence of the Gaussian whirling we show in Figure 4 the input data before having computed the whirling. Obviously the Gaussian process leads to a significantly more linear dependency of the values.

The relation of the CO-values shown in Figure 5 is approximated by linear regression [HEK89]. As Figure 6 illustrates, the resulting mapping allows an accurate prediction of CO-curves. The figure contains

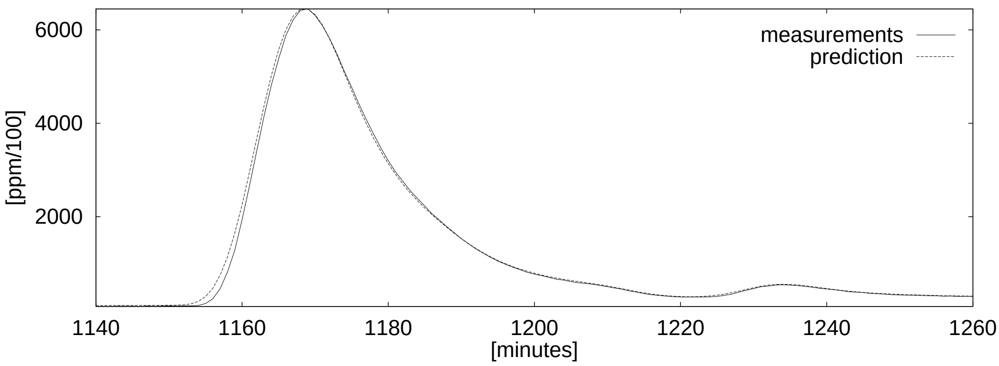

Figure 6: Predicted and measured CO-values of a blasting operation

the measured CO-curve of a CO-device and the predicted CO-curve based on the values of the preceeding device. Note that the shown time interval is different from the one used to generate the mapping.

Having provided a method for forecasting CO-values, we are next going to define a technique for the specification of the rule-based part of the fire-detection system.

3 Executable Rule-based System-Specification

High-level specification tools are essential for an appropriate specification and integration of the various modules of our system. In contrast, the basic heuristics of the system currently in use are expressed by complex flow-charts, which are hard to survey and to maintain. Our aim was to cure this deficiency using an executable, descriptive, and rule-based specification language.

Recently, *Evolving Algebras* (EAs) [Gur91, Gur94] turned out to be a powerful rule-based specification tool. The semantics of EAs have been investigated intensively within the last years [Gur94], and various applications in different fields are known [Bör94a]. EAs allow problem-oriented specifications tailored very closely to the given subject at an appropriate level of abstraction, supporting easy system verification. We decided to use EAs as their properties lead to more transparency of the underlying algorithms and heuristics for fire detection and thus to easier maintenance of the system. In this way, EAs contribute to enhance security in underground coal-mining.

EAs are suitable for an efficient implementation using logic programming. A Prolog-based EA-implementation for sequential non-modular EAs emphasizing efficiency exists [Kap93]. But for our system for fire detection, it was necessary to have modularity and parallelism. Each part of the complex analyzing methods constitutes a module of its own. Each module is executed in parallel for all sensors in the mine. Different modules represent distributed processes communicating via streams. The streams also realize the synchronization of these asynchronously running processes. Figure 7 shows the architecture of our fire detection system. The boxes represent different modules, and streams are indicated by arrows.

For this purpose we developed SEA (**S**treams in **EA**s) which provides the required means to implement this high-level system-specification.

3.1 The Specification Language SEA

SEA is a modular language allowing parallel execution of modules. In addition to partial functions, SEA supports partially bound data structures, e.g. streams. An evolving algebra is an SEA-module consisting of a module head, function declarations, a start rule and a set of transition rules.

The module head consists of the name and the interface of the module. The interface defines the name and arity with which the module will be called. It may be called either as an n-ary predicate in Prolog-programs or as an $(n-1)$-ary function in SEA-modules.

Module parameters defined in the module interface are internal SEA functions by definition. They may optionally be annotated with modes.

External functions are defined by a predicate name and arity `p/n`. Inside the actual SEA-module, an external function is visible as an $(\mathtt{n}-1)$-ary function.

Internal functions are local SEA-variables with optional mode annotations.

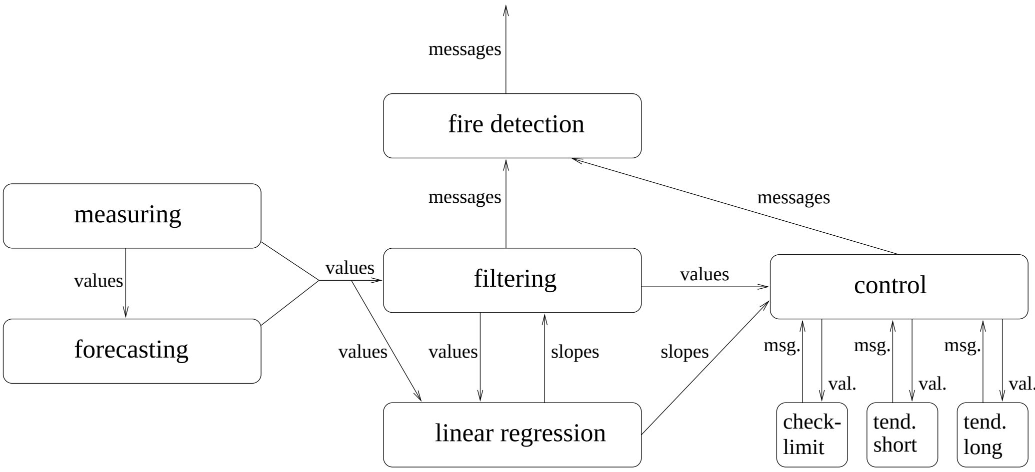

Figure 7: Module hierarchy of the fire detection system

A start rule is an action-formula, which is executed at the beginning of a computation.

A transition rule is a condition-formula followed by an action-formula, which is executed if the conditions are true.

Conditions compare terms by the use of built-in relations like `=`, `<`, `>`.

Actions are either function updates `f := t`, a `stop` operation terminating the computation, a binding of a partially bound data structure `t1 = t2` or a generation of a new temporary variable `var v` constituting an unbound part of a partial data structure.

Macros may occur either in condition or in action parts of a transition. Macros are defined outside the algebra by the command `macro Name => Replacements`, where `Replacements` are conditions or updates. Each occurrence of a macro name in a rule will be expanded at compile-time. Using macros rules become shorter and more declarative.

When a module is called at first the start rule is executed generating the initial state of the algebra. Then a loop is entered; in each round the update-part of the first rule whose conditions evaluate to true is carried out until a `stop` operation occurs. The `stop` operation terminates the module call.

SEA provides a graphical EA-visualization tool which allows a look at indicated functions of an algebra at any time. In this way, the system development became both quicker and safer.

```
algebra check_limit defines
        check_limit(device,value,message);

local warning_value,alarming_value;
external co_warning_value/2, co_alarming_value/2;

start warning_value := co_warning_value(device),
      alarming_value := co_alarming_value(device);

if   value < warning_value &
     value < alarming_value
then message = [], stop;

if   value < alarming_value &
     value >= warning_value
then message = ["Warning-level exceeded"], stop;

if   value >= alarming_value
then message = ["Alarming-level exceeded"], stop;
```

Figure 8: Fire detection module checking CO-limits

An example of an SEA-module for fire detection is given in Figure 8. The algebra `check_limit` defines the control of incoming CO-values with respect to given warning and alarm thresholds. As a more sophisticated example, we will subsequently show some rules for the filtering of blasting curves. The fire detection system identifies underground blastings and filters their high CO-values. According to [DMT93, SSSS82] a blasting curve is divided into three parts (see Figure 9): the first part starts at the beginning of the curve and ends at the first turning point `T`. The second part starts at `T` and ends at the maximum `M` of the curve. The third

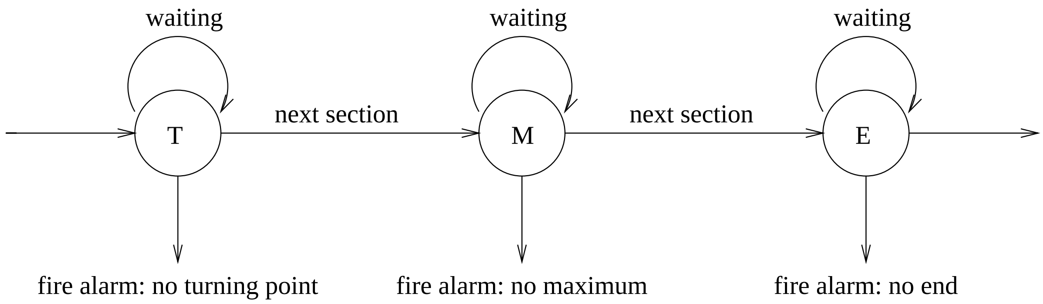

Figure 9: Graph representation of the algorithm filtering blasting curves

part starts at M and ends when the curve reaches its starting level again. `T` and `M` are detected by analyzing the slope of the curve. The end `E` of the curve, is found by comparing the current CO-value with the stored entrance level of the curve. Each part of the curve must be passed within a given time interval. Otherwise, an alarm is triggered and the filtering is stopped. Accordingly, the filtering algorithm has three SEA-rules for each section of a blasting curve: the first means waiting for `T`, `M`, or `E`, respectively, the second means going to the next section and the third gives a fire message if the time limit for this section has been exceeded. Figure 10 shows the rules for the `M`-section of a blasting.

4 Integration Level of the System

The overall system is realized by three permanently running Unix processes: the data acquisition interface, the monitoring module, and the graphical user-interface. The monitoring module receives its input as a set of streams containing CO-values, one per active CO-sensor. The monitoring module runs a considerably large set of parallel sub-processes under an and-parallel execution model. One of these parallel sub-processes is a controlling module which launches and supervises further processes monitoring a single CO-sensor each. These processes in turn fork to perform their subtasks. Synchronization between any pair of sub-processes is achieved through streams, which supply a (typically numerical) value per time unit.

The controlling module is also responsible for determining a monitoring method for each CO-sensor, i.e. whether forecasting can be applied or conventional techniques have to be used. The monitoring method is subject to change, when forecasting becomes no longer available due to the malfunction of the preceeding device. Furthermore, in case of modified forecasting circumstances like moving sensors or changed speed of air, the controlling module automatically launches temporary processes to determine new values for the whirling parameters and the time delay. Before these values are used, they are checked by special SEA-rules.

```
macro set_value =>
      head(new_values) = head(values),
      new_values := tail(new_values).

macro mark_value =>
      head(new_values) = void,
      new_values := tail(new_values).

macro next_values =>
      minutes := minutes + 1,
      slope := tail(slope),
      values := tail(values),
      flags := tail(flags).

if   filtering = on & turning_point = yes &
     maximum = no & minutes < maxminutes &
     head(slope) >= 0
then mark_value, next_values;

if   filtering = on & turning_point = yes &
     maximum = no & minutes < maxminutes &
     head(slope) < 0
then maximum := yes, mark_value, next_values;

if   filtering = on & turning_point = yes &
     maximum = no & minutes = maxminutes
then head(messages) =
        "Alarm: no maximum in blasting curve",
     messages := tail(messages),
     filtering := off, set_value, next_values;
```

Figure 10: Some rules of the module filtering blasting curves

The computation of predicted values is handled in an SEA-rule-based way, like any of the other monitoring methods. Given whirling parameters and the time delay from the sensor database, a time delay of n

Alarm type	Messages of system in operation		Messages of prototype with forecasting	
	Category A	Category B	Category A	Category B
Warning threshold exceeded	72	56	4	0
Alarm threshold exceeded	14	7	4	0
Fire tendency	20	20	4	4
Sum	106	83	12	4

Figure 11: Comparison of messages with and without forecasting

minutes is implemented by prepending a list of n `void` values to the predecessor's stream of CO-values. Due to the underlying Prolog's way of representing data structures in memory, no copying of the stream is necessary, it remains shared. Next, the shifted stream passes through a filter computing the Gaussian kernel corresponding to the whirling of the real CO concentration. The resulting stream of predicted values is subtracted from the actual CO-values of the current CO-sensor, and the difference stream is connected to the subsequent analysis modules.

5 Application Results

In order to document the enhancement of the current fire detection system in coal mining, we focused our interest on ten subsequent CO-sensors. The corresponding CO-values were measured during two days in march 1994, where several blasting operations were carried out. These blastings caused more than 100 alarms at the observed sensors. The alarms can be divided into two groups:

- Incoming values are compared with a warning and alarming threshold. A warning CO-level is announced by an optical signal, an alarming CO-level produces an optical and an acoustical signal.

- Over some given time intervals (10, 30, 60 and 180 minutes, 8, 16 and 24 hours) the slope of the CO-curve is computed by linear regression. If the slope exceeds a given threshold, a fire tendency message will be given. Computing the tendencies, it is possible to recognize even slowly but monotonous rising CO-values, which are indicating smoldering fires.

The current fire detection system filters high CO-values coming from blasting curves by some well evaluated heuristics, see 3.1. A blasting curve for example must have its turning point and maximum within fixed time intervals. All values measured by a CO-sensor that are marked as filtered are not taken into account for the computing of tendencies.

Our improved fire detection system prototype relies on the forecasting technique. Instead of analyzing the absolute CO-values we only consider the difference between the measured and the predicted value. This difference is treated like a new, virtual CO-sensor and we use the same algorithm to compute the tendencies and to monitor thresholds like before. As mentioned above, we decreased the thresholds indicating warning and alarming levels to the half.

The table shown in Figure 11 compares the messages produced by the current system with the messages given by our enhanced fire detection system. While category A contains all ten CO-sensors, category B includes only the eight CO-sensors from category A which are not next to the two blasting areas. Notice that for CO-sensors next to mining activities the additional amount of CO cannot be predicted by the preceeding CO-sensors. The sensors in category B are most interesting for the forecasting method because their CO-curves can be predicted and they generally produce sequences of alarms.

Because of the blasting operations during the two observed days the current system produced 72 threshold warnings, 14 threshold alarms, and 20 fire tendency alarms for all ten sensors (category A). These numbers are reduced significantly by the forecasting method: the number of warnings decreased from 72 to 4 which is a reduction of 95 percent. The alarms were reduced by more than 70 percent such that only 4 alarm messages remained. Only 4 out of 20 fire tendencies were given by our system.

Considering only the sensors of category B the improvement becomes even better. 56 threshold warnings and 7 threshold alarms were produced by the current system. With the forecasting method all these messages were eliminated. As can be seen in the figure all fire tendency alarms were produced by the sensors of category B so that there is no further improvement compared to category A.

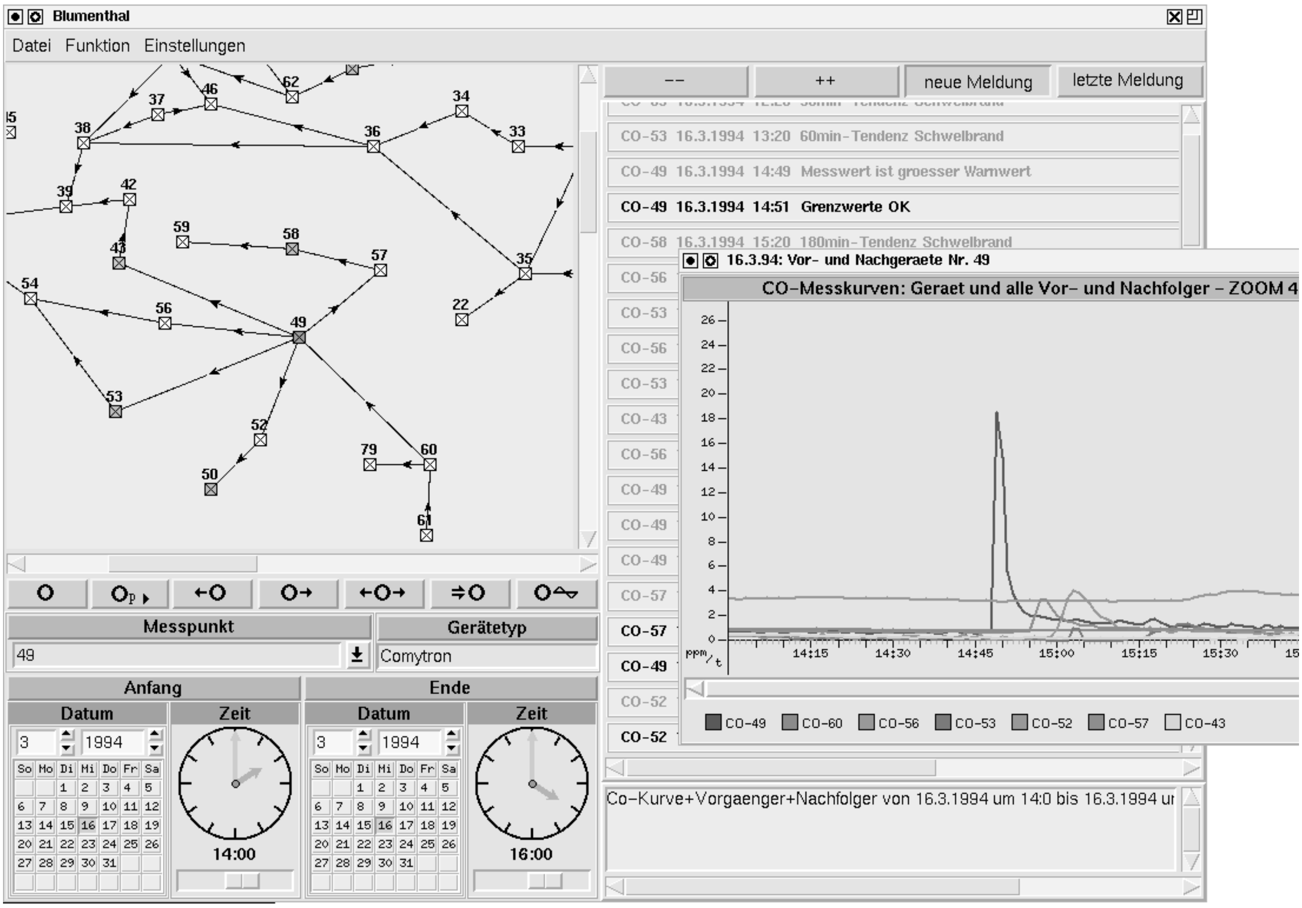

Figure 12: A snapshot of the user interface

The results of our analyzing method based on forecasting are very encouraging. By the use of our enhanced system, the staff in the mine observation center could be relieved from many false fire alarms. As a conclusion, our system based on forecasting reduces the number of false alarms to about 12%. It is safe to say, that our enhanced system for fire detection generally reduces the number of false alarms by more than 70%. This rate will be further increased, as improved classification methods for CO-curves, where no forecasting can be applied, are currently under development.

6 Graphical User Interface

The new CO-monitoring system comes along with a mouse-driven graphical user interface (see Figure 12). The user interface has been developed by the use of the modern and productive tool Tcl/Tk generating X11 applications. As the implementation of our system has mainly be done using the logic programming paradigm, we created a tool combining the languages Prolog and Tcl/Tk. More informations on our **PAT** tool can be found in the net[1].

On the right side of the system's main window actual warnings and fire alarms are displayed. The upper side of the left part shows a directed graph representing the CO-sensors of the mine and their predecessor/successor relations. This mine model is a base for the automatic computation of forecasting parameters. It may also be used to select a device for displaying its stored measuring or forecasting curves in a separate window.

7 Conclusions

The detection of underground fires is an important security task in hard-coal mining. The current fire detection system running on a process computer produces many false alarms caused by mining activities. In this paper we showed how CO-values can be forecast by appropriate models of the physical and chemical pro-

[1] `http://www.informatik.uni-bonn.de/~angelica/PAT-WWW`

cesses. This offers a great potential for the improvement of fire detection in hard-coal mines. First, the analysis of CO-curves can operate on the difference between the measured and forecast CO-concentrations. Second, heuristic thresholds for warnings or alarms can be adjusted more precisely, which allows a faster detection of fires. Practical applications of this new method show a reduction of false alarms by more than 70%.

Our specification language SEA extends the Evolving Algebra approach by stream parallelism. SEA has been used for the specification of the overall monitoring process. The high level of abstraction supports verification and improves transparency of the CO-monitoring system. At the same time, the SEA-compiler automatically compiles SEA-specifications into efficient Prolog code.

Thus, our system shows how the combination of knowledge based and sub-symbolic techniques leads to an improvement of transparency, reliability, and security.

References

[Ari56] R. Aris. On the dispersion of a solute in a fluid flowing through a tube. In *Proceedings of the Royal Society of London*, volume 253 of *A*, pages 67–77, 1956.

[BBC+90] K.-H. Becks, W. Burgard, A.B. Cremers, A. Hemker, and A. Ultsch. Parallel process interfaces to knowledge systems. In *Proceedings of the International Conference on Parallel Processing in Neural Systems and Computers*, 1990.

[BCG+91] Wolfram Burgard, Armin Bernd Cremers, Judith Grebe, Reinald Grevé, Stefan Lüttringhaus, Frank Mücher, and Lutz Plümer. Knowledge based planning of underground lighting in hardcoal mining. In R. Vichnevetsky and J.J.H. Miller, editors, *Proceedings of the 13th IMACS on Computers and Applied Mathematics*, volume 4, pages 1751–1752, 1991.

[Bör94a] Egon Börger. Annotated Bibliography on Evolving Algebras. [Bör94b].

[Bör94b] Egon Börger, editor. *Specification and Validation Methods.* Oxford University Press, 1994.

[CTB94] Armin B. Cremers, Sebastian Thrun, and Wolfram Burgard. From AI technology research to applications. In *Proceedings of the 13th World Computer Congress*, volume 3. Elsevier Science B.V. (North Holland), 1994.

[DMT93] Bericht über die Untersuchung der Prozeßrechneranlage zum Erfassen und Verarbeiten wettertechnischer Meßdaten mit Schreiberersatz auf dem Bergwerk Ewald / Schlägel und Eisen. DMT-Gesellschaft für Forschung und Prüfung mbH, August 1993.

[Gur91] Yuri Gurevich. Evolving Algebras. A tutorial introduction. *Bulletin of EATCS*, 43:264–284, 1991.

[Gur94] Yuri Gurevich. Evolving Algebras 1993: Lipari Guide. In Börger [Bör94b].

[HEK89] Joachim Hartung, Bärbel Elpelt, and Karl-Heinz Klösener. *Statistik: Lehr- und Handbuch der angewandten Statistik.* R. Oldenbourg Verlag München Wien, 9 edition, 1989.

[Kap93] Angelica M. Kappel. Executable Specifications based on Dynamic Algebras. In A. Voronkov, editor, *Logic Programming and Automated Reasoning*, volume 698 of *Lecture Notes in Artificial Intelligence*, pages 229–240. Springer Verlag, 1993.

[Kar87] Hans-Jürgen Kartenberg. Bessere Brandüberwachung in Strecken mit großen Wetterströmen. Glückauf-Forschungshefte 48, Nr. 2, 1987.

[Lit70] A.B. Littlewood. *Gas Chromatography.* Academic Press, 2 edition, 1970.

[SSSS82] Anton Stark, Horst Seyfarth, Hans Sredenscheck, and Joachim Steudel. Stand und Entwicklung der Brandfrüherkennung bei der Bergbau AG Lippe. *Glückauf*, 118(8):3–7, 1982.

[Tay53] Sir Geoffrey Taylor. Dispersion of soluble matter in solvent flowing slowly through a tube. In *Proceedings of the Royal Society of London*, volume 219 of *A*, pages 186–203, 1953.