ABSTRACTOR: AN AGGLOMERATIVE APPROACH TO INTERPRETING BUILDING MONITORING DATA

SUBMITTED: November 2006
REVISED: February 2008
PUBLISHED: May 2008
EDITOR: R. Amor

Apkar Salatian
School of Computing, The Robert Gordon University, St Andrew Street, Aberdeen, AB25 1HG, United Kingdom
Email: apkar.salatian@northampton.ac.uk

Bruce Taylor
Scott Sutherland School, The Robert Gordon University, Garthdee Road, Aberdeen AB10 7QB, United Kingdom
Email: b.taylor@rgu.ac.uk

SUMMARY: *Building operators are confronted with large volumes of continuous data from multiple environmental sensors which require interpretation. The ABSTRACTOR system under development summarises historical data for interpretation and building performance assessment. The ABSTRACTOR algorithm converts time series data into a set of linear trends which achieves data compression and facilitates the identification of significant events on concurrent data streams. It uses a temporal expert system based on associational reasoning and applies three consecutive processes: filtering, which is used to remove noise; interval identification to generate temporal intervals from the filtered data - intervals which are characterised by a common direction of change (i.e increasing, decreasing or steady); and interpretation which performs summarisation and assists building performance assessments. Using the temporal intervals, interpretation involves differentiating between events which are environmentally insignificant and events which are environmentally significant. Inherent in this process are rules to represent these events. These rules support temporal reasoning and encapsulate knowledge to differentiate between events.*

KEYWORDS: *Building, monitoring, interval identification, data mining, clustering, data compression.*

1. INTRODUCTION

Energy efficiency in buildings is moving up the political and business agendas (DTI, 2003), (CIBSE 2004)). Since only about 1% of the UK's building stock is renewed each year significant steps in reducing the UK's consumption of energy in buildings and hence CO_2 emissions depend upon improving the performance of existing buildings. The first step in addressing this is for building owners' to commission an energy audit. A detailed audit would include measuring the electrical and fossil fuel supplies for individual services (space heating, hot water, lighting, fan and pump power for ventilation and air conditioning etc) on an half hourly basis for at least one year in order to discover the seasonal variations between summer and winter. Further measurements of air temperature and humidity, space heating/cooling fluids and domestic hot water temperatures, may also be made to assist in the diagnosis of areas where energy is being used excessively or wastefully. (Field et al., 1997) found some of the main causes of poor energy performance of buildings are that the building services are not operating as they have been scheduled to operate or the building is occupied differently from the management's intentions.

Measuring and collecting large, volumes of time series data on the environmental performance is becoming increasingly affordable with falling prices of sensing and computing technology and the increasing availability of Building Energy Management Systems (BEMS). To make use of this potential of more extensive routine data collection it is necessary to apply statistical analysis and data mining techniques to extract relevant information useful to the building services operator.

Whilst energy audits can be conducted without extensive monitoring of a building (Field et al 1997), this is not the case when the building performance is being investigated as part of a research project. For research on even a domestic scale building there may be many measurements (30 would not be untypical) made and logged over extended periods of time. On a commercial building the number of variables recorded could easily be 100 or more especially if the research were to extend beyond energy use to broader environmental issues such as indoor air quality, lighting, electromagnetic radiation and human interaction with the building and the services controls.

Currently most analysis is done simply by visual inspection of time series graphs of the variables on the computer screen of the BEMS operator or energy consultant. More often than not data is not analysed (ASHRAE, 2003) and the opportunity is lost to turn data into information and knowledge. If suitable techniques could be developed for the analysis of large time series sets of building environment data and the results routinely incorporated into a BEMS then this may lead to improvements in the operating efficiency of buildings.

Building operators are confronted with large amounts of historical continuous data and they could benefit from assistance in its interpretation. When considering a given time in the past, data is available relating to times both *before* and *after* that time. Providing high level summaries of what has happened to a building in, say, the past 24 hours or 7 days, could be of significant importance in assisting building operators to decide which interventions are appropriate. It is an important feature of the summarisation of past events that the future (relative to that event) can be used to confirm the presence or absence of that event. Environmentally insignificant events must be removed as they would otherwise lead to inaccurate interpretations.

This paper presents a system called ABSTRACTOR which uses a novel agglomerative clustering algorithm for deriving temporal intervals which have the properties *increasing, decreasing* or *steady* from dense data sets. The objective is to develop a tool that will automatically determine changes in the variables that affect the energy efficiency of a building. The results of applying ABSTRACTOR to time series thermal data from a building are presented and critically appraised.

The structure of this paper is as follows: section 2 reviews literature on environmental modelling, section 3 outlines the ABSTRACTOR algorithm, describes each of the ABSTRACTOR processes and the results of testing each stage; section 4 shows the results of ABSTRACTOR and final conclusions are in section 5. This paper presents time series data collected from a low energy house that has unfired internal clay brick walls as the internal skin of the external walls and the interior walls. This work was funded under the UK's Department of Trade and Industry Partners in Innovation Competition.

2. LITERATURE REVIEW

Time series analysis techniques have a long history of application in meteorology and climatology ((Stringer, 1972), (Brooks and Carruthers, 1953)). They are used to develop design data and usual involve the production of daily, monthly or annual means of climatic variables such as air temperature solar irradiation (Page, 1986) or test reference years which will provide hourly data for a "typical" year for a given location (CIBSE, 2002) They are also used in studies of niche applications of building technologies. A system is needed for identifying, in environmental data, events relevant to the investigator. To simplify computation the events could be defined by rules. The events could be, for example, faults in the operation of the building environmental control system.

A number of different techniques for detecting and isolating faults in HVAC plant have been developed in IEA Annex 25. The techniques make use of simple, on-line models of correct operation to detect faults. Diagnosis is based either on on-line models of different faults or on expert rules and not purely on detection of trends as described in this paper. These techniques were developed using detailed computer simulation and have been tested using experimental data from laboratory HVAC plants (Liddament, 1999). IEA Annex 34 tests the fault detection and diagnosis (FDD) techniques in realistic on-line situations (Dexter and Pakanen, 2001). More recently (Braun and Li., 2003), carried out a detailed review of the literature on FDD as applied to HVAC systems in order to improve their economic viability. FDD is expensive to apply practically because of requirements for model training and large computational demand. They have only been tested in the laboratory and not developed to take account of factors that occur in the field. Nor could they handle multiple simultaneous faults. We need a system that does not require extensive computer models of the system but does require rules to identify faults from the trend intervals.

The only application of trend recognition techniques to improve engineering processes is in the chemical industry (Rengaswamy and Venkatasubramanian, 1995). Their aim was to develop real-time decision support systems for plant operators which extract qualitative features of process trends from real time sensor data and perform cause and effect reasoning to make assessments about the process behaviour. Such analysis was then used to draw conclusions about the possible course the process might take in the near future, the reasons for doing so and any appropriate control actions that might be needed. Chemical processes are highly dynamic this is in contrast to most building processes with exception of solar gains in cloudy country like UK which can have a large dynamic range even in winter.

Rengaswamy and Venkatasubramanian developed a Trend Description Language (TDL). TDL is based around a set of nine generic trend curves (called primitives) which were used to capture the essential features of real-time process data. This technique meant real-time data could be compressed into a data structure which carries the following information 1) primitive associated with the episode, 2) interval length, 3) initial and final values and 4) pointer to the next profile structure. This has similarities to ABSTRACTOR except that ABSTRACTOR uses only three primitives (linearly increasing, decreasing and steady). This is adequate for many building environment variables with periods of order 24 hours but may not so good for representing switching phenomena.

Neural networks, because they are good at pattern recognition, were used for identifying the appropriate primitives to represent the real data. A disadvantage of neural networks is that they have to be trained using input training data which covers the rage of events one wishes the device to detect.

(Liao and Dexter, 2003) proposed an Inferential Control Scheme (ICS) for use in multi-zone buildings where there is no measurement of the internal air temperature in the different zones of the building. The room temperature estimator is a simplified model of the building and the heating system. The model includes solar heat gains through glazing and walls as well as heat losses through these elements and ventilation heat losses. The estimator was commissioned using short-term monitoring data. The relevant parameters were initially calculated on the basis of known design conditions. Then a general depth-priority searching mechanism was applied to search for an optimal combination of these parameters values that minimised the root mean square of the estimation errors. Liao and Dexter concluded that the estimator could accurately estimate the long term (13-20 days) average room temperature of the building due to changes in both climatic conditions and the mode of operation of the heating system. However, it is the present authors' view that a heating control system should be self adaptive in which case changes in average room temperature would need to be estimated accurately over the short-term i.e 24 hours.

(Shen and Chouchoulas, 2000) acknowledged that most attempts to build successful intelligent monitoring and diagnostic systems tend to stumble at the knowledge acquisition bottleneck. They integrated rough set attribute reduction (RSAR) with a powerful Rule Induction Algorithm (RIA) known as NP_hard. The dimensionality-reducing rule-induction framework was applied to a set of historical data from a water treatment plant. The data comprised 521 days with one series of 38 measurements per day.

A desirable goal of rule induction algorithms is to be able to generate operational and diagnostic rules from plant data when experts are not available to undertake a detailed analysis of historical data. For real time monitoring and diagnostic systems, experts can be used only in the analysis of historical data to generate rules that are then implemented in real-time. However, these systems would not be adaptive in real time.

(Yu, et al, 2003) proposed a FDD system based model of the building and heating system which compared calculated indoor air temperature and radiator valve position with the measured values. Indoor air temperature is the parameter used to detect faults which are classified into two types: Type I (increase in room temperature), Type II (decrease in room temperature).

To diagnose the cause of the fault in the relatively simple situation of a room with a window and radiator, a parameter P is defined that allows for the compensation effect of the radiator and the influence of the outside air temperature. An open window in winter will cause indoor air temperature to drop, this will generate a signal to the thermostatic valve to drive it open. If the thermostatic valve is faulty (sticking) this will cause the difference between the required position of the valve and the actual position of the valve to increase. The thresholds at which the parameter P and the indoor air temperature residual are set for detecting faults has been set on the basis of a qualitative analysis of their respective time series over a period of 250 hours (10 days). We need a system to detect faults qualitatively using trends rather than quantitatively using thresholds which could represent noise.

A novel agglomerative algorithm, **ABSTRACTOR,** is proposed for identifying intervals using clustering and temporal reasoning. The algorithm abstracts time series data to reveal key features and events in the data. This will facilitate the identification and understanding of changes in energy use in a building.

3. THE ABSTRACTOR ALGORITHM

The ABSTRACTOR algorithm, which summarises or abstracts historical data, can be viewed as the application of a sequence of three processes (Fig 1). Data is initially *filtered* to get rid of environmentally insignificant events and noise; the resulting data stream is then *segmented* by a second process into temporal intervals over which the predicates *steady, increasing* or *decreasing* hold; these intervals and knowledge of environmentally significant events are *interpreted* by a third process.

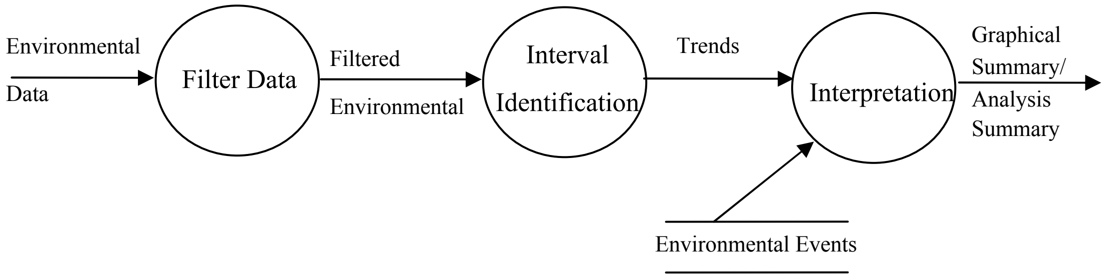

FIG 1: Processes of ABSTRACTOR

3.1 Filter data

Initially data needs to be filtered to get rid of noise e.g., non-significant events in environmental monitoring data e.g., air temperature spikes which occur during a cold spell when someone opens a window for a short period. If this happens infrequently then such events are insignificant and should be treated as noise and removed.

The following filters: *median filter*, *average filter, low-pass filter* and *high-pass filter* were investigated. All of these techniques involve a moving window. For historical data, the window can be centered on the point x_n i.e if the window is of size 2k+1 the window contains the points x_{n-k} to $x_{n+k.}$ In what follows only centered windows will be considered.

3.1.1 Filter Types

Median Filter

The median filter algorithm simply takes the unweighted median of all values within the window, i.e the value which has as many values which are greater as are less than it is taken to be the median.

$$y_n = \text{median}(x_{n-k},..., x_n,..., x_{n+k})$$

Average Filter

An average filter simply takes the mean of all the values within the window.

$$y_n = \text{average}(x_{n-k},..., x_n,..., x_{n+k})$$

Low-Pass Filter (non-recursive)

Low-pass finite and symmetric digital filter takes the form (Hamming, 1989)

$$y_n = \sum_{k=-N}^{k=N} c_k \, x_{n-k} \qquad (c_k = c_{-k})$$

Where

$$c_k = \frac{\sin(0.4\pi k)}{\pi k}\frac{N}{\pi k}\sin\left(\frac{\pi k}{N}\right) \qquad \text{k = 1 to N}$$

$$c_0 = 0.4$$

The sampling frequency, f_s , of the data set was 96 samples/day. The highest frequency that can be defined by the sampling frequency is the aliasing, f_a , or Nyquist frequency which is half the sampling frequency i.e 48 cycles/day. Therefore the frequency range of interest for this application of building monitoring data is 0 to 48 cycles/day.

The low pass filter was designed to let through frequencies $f^* < 0.2$ where the dimensionless frequency f^* is defined in terms of the sampling frequency

$$f^* = \frac{f}{f_s}$$

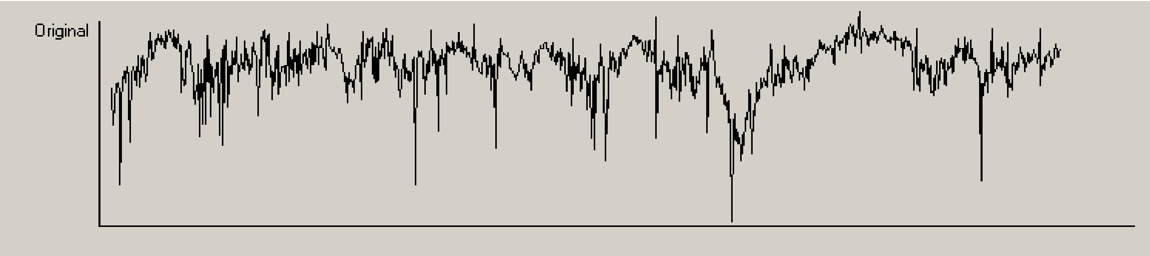

FIG 2: Original Data

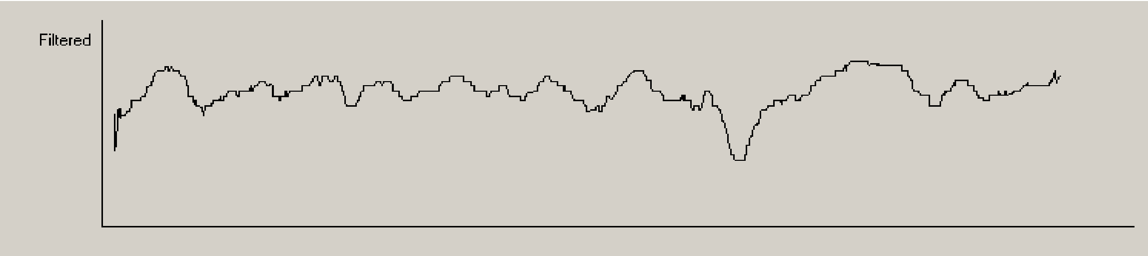

FIG 3: Application of the median filter

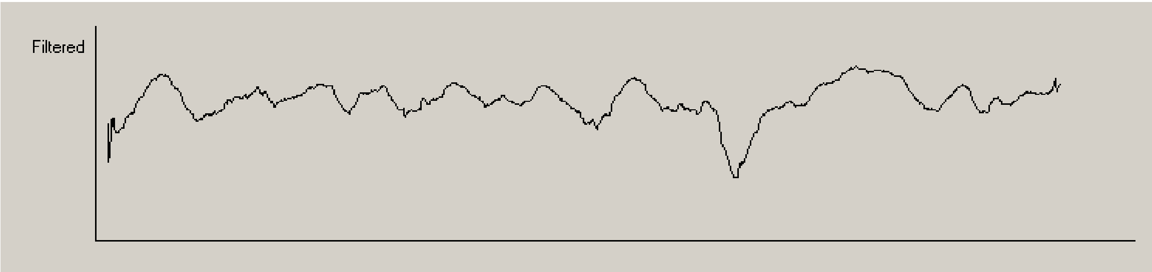

FIG 4: Application of the average filter

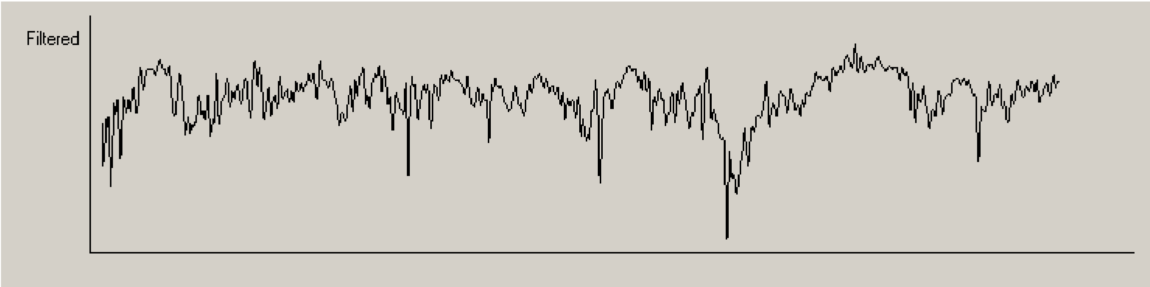

FIG 5: Application of the low-pass filter

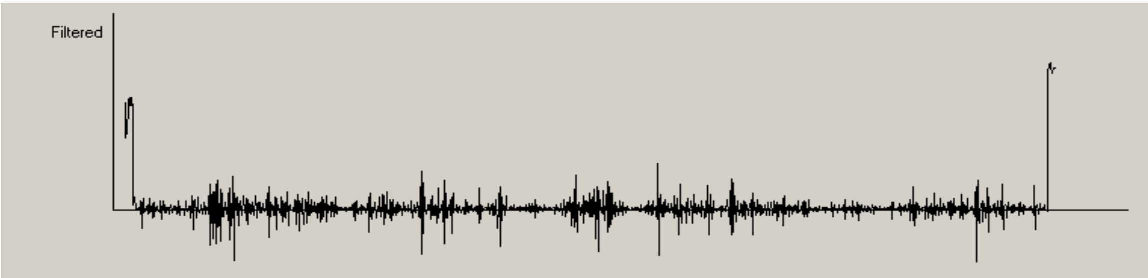

FIG 6: Application of the high-pass filter

High-Pass Filter (non-recursive)

The high pass digital filter was designed to let through frequencies in the range $0.2 < f^* < 0.5$. The high pass filter coefficients, c'_k, can be derived from the low pas filter coefficients, c_k :

$c'_k = -c_k$ for $k \neq 0$

$c'_k = 1 - c_k$ for $k = 0$

$$c_k = \frac{\sin(0.4\pi k)\cos(0.7\pi k)}{\pi k}\frac{N}{\pi k}\sin\left(\frac{\pi k}{N}\right) \qquad k = 1 \text{ to } N$$

$$c_0 = 0.6$$

3.1.2 Filter Results

The results of applying the filters to historical data are now presented. In order to analyse the results an interesting case was chosen which contains many trends and events. The data set is an external air temperature trace. The frequency of the data was one value every 15 minutes. The data was collected over a 2 week period and it contains 1155 data points. The original data set is shown in Fig 2. Figs 3 through to 6 are the graphical results of applying the various filters with k = 10, which are each looked at in turn.

Fig 3 shows the result of applying a median filter. It can be seen that spikes lasting of size k or less are removed. In Fig 4 it can be seen that the average filter smoothes the data to a greater extent than the median filter. Fig 5 shows the result of applying a low-pass filter. It can be seen that the low-frequency variations are allowed to “pass through” the filter. In a low-pass filter, the low frequency (long-period) waves are barely affected by the smoothing. The output from the low pass filter is still quite noisy compared with the either the average or median filter. In contrast Fig 6 shows that the high pass filter eliminated the low-frequency variations and the high-frequency variations are unaffected.

From the results it is concluded that the average filter with k=10 is the best filter to use as it removes all the very short duration spikes from the outdoor temperature data whilst revealing the short duration trends hidden in the raw data. This is because a median filter removes transient features lasting shorter than half the width of the window hence events lasting more than half the width of the window will not be removed, a low-pass filter attenuates noise (noise may have some low-frequency components) and a high-pass filter eliminates low frequency variations and trends leaving only the higher frequency components.

3.2 Interval Identification

This process can be divided into the sub-processes *temporal interpolation* and *temporal inference* (Fig 7).

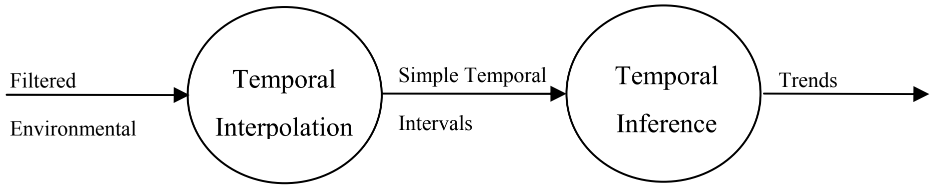

FIG 7: The sub-processes of the process Interval Identification

In what follows the constructed temporal intervals; the i^{th} temporal interval, I_i, are described as

$$I_i\ (x,\ t_{begin},\ t_{end},\ trend_i\ ,\ \alpha_i,\ \beta_i\ ,\ \mu_i,\ \theta_i,\ \sigma_i)$$

where

- x is the variable under investigation (e.g., heat flux or air temperatures)
- t_{begin} and t_{end} are the start and end times of the interval
- $trend_i$ is the interval trend i.e *increasing, decreasing* or *steady.* Note the increasing and decreasing trends can be classified as *slow, moderate or fast* depending on the value of μ_i (see below)
- α_i and β_i are the minimum and maximum values of x over the interval
- μ_i is the absolute value of the gradient of x over the interval
- θ_i is the mean of x over the interval
- σ_i is the standard deviation of x over the interval

All of the numerical properties of the interval are calculated simply from the values of the data points which it encompasses. The issues in question are (i) how to determine the extent of the individual intervals and (ii) how to determine the trend.

3.2.1 Temporal Interpolation

This is the process of generating an interval between two adjacent data points. By 'adjacent' it is meant that there are no missing data points between them. Note that in all our raw data, 0 (zero) may stand for a true zero data point or may stand for missing data; there is no way of knowing which. The value of x at t_{begin} is referred to as x_{begin} and the value of x at t_{end} as $x_{end.}$

The trend *steady* ($trend_i = steady$) is derived if $x_{end} = x_{begin} \pm \delta x$. The value of δx will depend on the resolution of the instrument supplying the data and on the number of significant figures appropriate for the application. In our case the level of quantisation is taken be zero (i.e. strict equality is applied). If the interval is not classified as *steady*, then it will be *increasing* if $x_{end} > x_{begin}$ and *decreasing* if $x_{end} < x_{begin}$

The values of α_i, β_i , θ_i, and σ_i are calculated as follows:

- $\alpha_i = \min(x_{end}, x_{begin})$
- $\beta_i = \max(x_{end}, x_{begin})$

$$\theta_i = \frac{1}{n}\sum x$$

$$\sigma_i = \sqrt{\frac{\Sigma(x-\theta_i)^2}{n-1}}$$

For *steady* intervals μ_i is set to zero. For increasing or decreasing intervals it is defined as $(| x_{end} - x_{begin} |)/(t_{end} - t_{begin})$

Temporal interpolation involves a single pass over the data set. Given *n* data points, it generates exactly *n-1 simple temporal intervals*.

3.2.2 Temporal Inferencing

Temporal inferencing involves looking for trends in the data which are steady, increasing or decreasing and on the data's rate of change. An interval is steady if the difference between any two points in the interval is below a threshold, δx. This threshold will determine a maximum and minimum value for the interval. The value of the parameter threshold, *diff*, depends on the variability of the data. The greater the data's variability, the larger the threshold.

A deviation greater than the allowable range for a steady may be considered to be either an increasing or decreasing trend. A deviation above the maximum value for a steady is considered as an increasing trend. Likewise a deviation below the minimum value for a steady is considered as a decreasing trend. The different rates of change, namely whether an increasing or decreasing trend is slow, moderate or rapid are investigated. These trends are defined by different gradient ranges and are parameter dependent.

Temporal Inferencing is the process of attempting to apply rules to merge two or three neighbouring intervals into *super-intervals*, so that a common trend can be derived. This is the process of agglomeration. We use rules to merge over two intervals to derive increasing and decreasing trends which have similar gradients (such rules are based on a *rate of change*) and to derive steady intervals where the values are within a set range (such rules are based on a *range*). The rules for merging must take account of the fact that a super-interval which is described, for example, as increasing, may actually be made up of some smaller sub-intervals in which the variable is described as steady or even decreasing. Such rules are based on *duration* and *rate of change*. For example, given three intervals which are increasing, steady and increasing respectively one can infer a possible increasing super-interval if the duration of the steady interval is shorter than the duration of the increasing intervals. Likewise a *steady* interval may be made up of many increasing, decreasing and steady intervals. Here one needs upper and lower thresholds and an appropriate metric is provided by the minimum and maximum values of the steady interval.

To derive super-intervals, the following four parameters for each variable are defined:

- *dur* - a duration; value depends on whether interested in long or short term trends
- *diff* - a range; used in the definition of *steady* intervals
- g_1 - a gradient (see below)
- g_2 - a gradient, greater than g_1 (see below)

If a gradient is less than g_1 ($\mu_i <= g_1$) this is taken as representing a *slow* rate of change. If it is between g_1 and g_2 ($g_1 < \mu_i <= g_2$) this is taken as representing a *moderate* rate of change. If it is greater than g_2 ($g_2 > \mu_i$) this is taken as representing a *fast* rate of change.

As intervals are merged into super-intervals, a new data structure is created representing the new interval, derived from the representations of its sub-intervals. Temporal inferencing is done in two ways: over *two* adjacent intervals and over *three* neighbouring intervals.

Based on similar characteristics an agglomerative clustering approach is used to merge the simple intervals generated from the temporal interpolation process into larger intervals then repeatedly merge these larger intervals into even larger intervals until no more similarities can be found. This merging algorithm is achieved using the temporal inference rules. Firstly, rules to merge two adjacent intervals to derive only increasing and decreasing trends are applied. This will provide the basis for finding potentially larger increasing and decreasing trends. The term 'apply' means trying to combine the first two intervals; if this succeeds then trying to combine this new interval with the next and so on. If two adjacent intervals cannot be merged then the interval which was to be combined is taken and used as a starting point. The rules are then applied to merge two adjacent intervals to derive only steady intervals. Merging two adjacent intervals can generate too many intervals. The inferences are then repeatedly applied to merge three neighbouring intervals followed by the inferences to merge two adjacent intervals until no more intervals can be merged.

3.2.2.1 Temporal inferencing over two adjacent intervals

For temporally inferencing over two adjacent intervals it is desirable to represent a *super-interval* as an assertion in the following form:

$$\Delta_{H2}(I_i\ (x, t_1, t_2, trend_i\ , \alpha_i, \beta_i\ , \mu_i,, \theta_i, \sigma_i), I_j(x, t_2, t_3, trend_j\ , \alpha_j, \beta_j\ , \mu_j, \theta_j, \sigma_j))$$

$$\Rightarrow I_{ij}\ (x, t_1, t_3, trend_{ij}\ , \alpha_{ij}, \beta_{ij}\ , \mu_{ij},, \theta_{ij}, \sigma_{ij})$$

Thus given intervals I_i (from t_1 to t_2) and I_j (from t_2 to t_3) which meet at t_2, a *super-interval* I_{ij} (from t_1 to t_3) can be derived by merging the intervals I_i and I_j based on the temporal inferencing function Δ_{H2}.

By symmetry, decreasing followed by decreasing is similar to increasing followed by increasing. The criterion for combining any two intervals into a steady interval is independent of the trends of the two contributing intervals. Therefore only two distinct cases need be considered.

Whenever a super-interval is generated, the values of α_{ij}, β_{ij} , θ_{ij}, and σ_{ij} are calculated as follows:

- $\alpha_{ij} = \min(\alpha_i, \alpha_j)$
- $\beta_{ij} = \max(\beta_i, \beta_j)$
- θ_{ij} is derived from the average of all the values over the 2 adjacent intervals
- σ_{ij} is derived from the sums of squares carried with representations of the individual intervals.

If a gradient is less than g_1 ($\mu_i <= g_1$) this is taken as representing a *slow* rate of change. If it is between g_2 and g_2 ($g_1 < \mu_i <= g_2$) this is taken as representing a *moderate* rate of change. If it is greater than g_2 ($g_2 > \mu_i$) this is taken as representing a *fast* rate of change.

The value of μ_{ij} depends on $trend_{ij}$. If $trend_{ij}$ is *steady* then $\mu_{ij} = 0$. If $trend_{ij}$ is *increasing* or *decreasing* then $\mu_{ij} = (\beta_{ij} - \alpha_{ij})/(t_3 - t_1)$

increasing/increasing ⇒ increasing

For successive increasing intervals, a super-interval *increasing* is inferred if the gradients of *both* intervals are within the *same* limit range. Formally this is written as:

$(\mu_i <= g_1$ **AND** $\mu_j <= g_1)$

OR $(g_1 < \mu_i <= g_2$ **AND** $g_1 < \mu_j <= g_2)$

OR $(g_2 < \mu_i$ **AND** $g_2 < \mu_j)$

If this condition is not satisfied, no inference can be performed.

any/any ⇒ steady

A super-interval is classified as *steady* if the difference between the maximum and minimum values over the adjacent intervals is less than the pre-defined constant *diff*.

$$\beta_{ij} - \alpha_{ij} <= diff$$

3.2.2.2 Temporal inferencing over three neighbouring intervals

Inferring over two adjacent intervals alone can result in too many intervals. Inferring over three neighbouring intervals in particular cases allows us to create even larger intervals which are either increasing or decreasing. For three neighbouring intervals it is desirable to represent a super-interval as an assertion in the following form:

$$\Delta_{H3}(I_i\ (x, t_1, t_2, trend_i\ , \alpha_i, \beta_i\ , \mu_i,, \theta_i, \sigma_i), I_j(x, t_2, t_3, trend_j\ , \alpha_j, \beta_j\ , \mu_j, \theta_j, \sigma_j),$$

$$I_k(x, t_3, t_4, trend_k\ , \alpha_k, \beta_k\ , \mu_k, \theta_k, \sigma_k))$$

$$\Rightarrow I_{ijk}\ (x, t_1, t_4, trend_{ijk}\ , \alpha_{ijk}, \beta_{ijk}\ , \mu_{ijk},, \theta_{ijk}, \sigma_{ijk})$$

Thus given intervals I_i (from t_1 to t_2), I_j (from t_2 to t_3) and I_k (from t_3 to t_4) a *super-interval* I_{ijk} beginning at time t_1 and ending at time t_4 can be created by merging the intervals I_i, I_j and I_k using the temporal inferencing function Δ_{H3}.

Though the inferencing is very similar to Δ_{H2} it will be seen that the size of the middle interval is critical in deciding whether to generate a super-interval. Again the aim is to create super-intervals which truly reflect rates of change and capture trends.

Considerations of symmetry allow us to consider the following three groups of possibilities for trends of three adjacent intervals:

- **increasing/increasing/increasing**
    - (decreasing/decreasing/decreasing)
- **increasing/steady/increasing**
    - (decreasing/steady/decreasing)
- **increasing/decreasing/increasing**
    - (decreasing/increasing/decreasing)

Where there are multiple possibilities, only the first will be considered, the remainder following from symmetry.

Whenever a super-interval is generated, the values of α_{ijk}, β_{ijk}, θ_{ijk}, and σ_{ijk} are calculated as follows:

- $\alpha_{ijk} = \min(\alpha_i, \alpha_j, \alpha_k)$
- $\beta_{ijk} = \max(\beta_i, \beta_j, \beta_k)$
- θ_{ijk} is derived from the average of all the values over the 3 neighbouring intervals
- σ_{ijk} is derived from the sums of squares carried with representations of the individual intervals.

The value of μ_{ijk} depends on $trend_{ijk}$. If $trend_{ijk}$ is *steady* then $\mu_{ijk} = 0$. If $trend_{ijk}$ is *increasing* or *decreasing* then $\mu_{ijk} = (\beta_{ijk} - \alpha_{ijk}) / (t_4 - t_1)$

3.2.2.2.1 increasing/increasing/increasing ⇒ increasing

Given three neighbouring intervals which are all increasing then an *increasing* super-interval can be inferred if the duration of the middle interval is less than the duration of the other two intervals by at least a factor of *dur* and the gradients of the two outside intervals are both within the same range:

$t_3 - t_2 < \min((t_2 - t_1), (t_4 - t_3))/dur$

AND

$((\mu_i <= g_1$ **AND** $\mu_k <= g_1)$

OR $(g_1 < \mu_i <= g_2)$ **AND** $g_1 < \mu_k <= g_2)$

OR $(g_2 < \mu_i)$ **AND** $(g_2 < \mu_k))$

3.2.2.2.2 increasing/steady/increasing ⇒ increasing

Given three neighbouring intervals which are increasing, steady and increasing respectively, one can infer an *increasing* super-interval if the duration of the steady interval is less than the size of it's neighbouring increasing intervals by at least a factor of *dur* and the gradients of the increasing intervals are both within the same range.

3.2.2.2.3 increasing/decreasing/increasing ⇒ increasing

Given three neighbouring intervals which are increasing, decreasing and increasing respectively, one can infer an *increasing* super-interval if the duration of the decreasing interval is less than the duration of it's neighbouring increasing intervals by at least a factor of *dur* the gradients of the increasing intervals are both within the same range, the minimum value of the decreasing interval (I_j) is greater than the minimum value of the first increasing interval (I_i) and the maximum value of the second increasing interval (I_k) is greater than the maximum value of the first increasing interval:

$(t_3 - t_2 < \min((t_2 - t_1), (t_4 - t_3))/dur)$

AND

((μ_i <= g_1 **AND** μ_k <= g_1)

OR (g_1< μ_i <= g_2) **AND** g_1< μ_k <= g_2)

OR (g_2 < μ_i) **AND** (g_2 < μ_k))

AND

(α_i < α_j **AND** β_k > β_i)

A summary of the algorithm for interval identification is shown in Fig 8.

1. Apply the inferences in Δ_{H2} which derive only increasing or decreasing trends by trying to combine the first two intervals; if this succeeds try to combine this new interval with the next and so on. If combination fails, then the interval which failed to be combined is taken and used as a new starting point.
2. Apply the inferences in Δ_{H2} which derive only steady trends.
3. Set flag *still-to-do* to **true**.
4. **while** *still-to-do* **do**
5. Set *previous* to the number of intervals generated so far.
6. Apply the inferences in Δ_{H3}
7. Apply the inferences in Δ_{H2}
8. Set *still-to-do* to *previous* = current number of intervals.
9. **endwhile**

FIG 8: Algorithm for interval identification

3.2.3 Interval Identification Results

The results of identifying temporal intervals in historical data are now presented. The types of temporal intervals generated by the kind of values assumed for the parameters *diff, dur*, g_1, and g_2 are explored. In order to analyse the effect of choosing particular values for the parameters in the temporal interfacing algorithm, an interesting case was chosen which contains many trends and events. The data set is an external air temperature trace. The frequency of the data was one value every 15 minutes. The data was collected over a 2 week period and it contains 1155 data points. An average filter of size *k=10* was chosen as a way of determining short term trends (five and a quarter hours). The original data set is shown in Fig 9. Figs 10 through to 16 are the graphical results of the intervals generated by different settings of the parameters.

Setting a low value for the parameter *diff* results in many (perhaps unnecessary) intervals being generated. Setting *diff* to 0.1 results in 202 intervals being generated (Fig 10). This is a reduction of points to intervals of 5.72. All changes of interest in the data are captured. Many intervals could be merged into larger super-intervals. Setting a high value for the parameter *diff* results in a small number of intervals being generated - these intervals are predominately long steady intervals. Setting *diff* to 10 results in only 7 intervals being generated (Fig 11). This is a reduction of points to intervals of 165. Only major increasing and decreasing trends are identified whereas slow changes are incorporated as part of a steady interval.

Setting a low value for the parameter *dur* results in many steady intervals being generated which would have otherwise been part of an increasing or decreasing trend. Setting *dur* to 1 results in 328 intervals being generated (Fig 12). This is a reduction of points to intervals of 3.52. Setting a high value for the parameter *dur* results in many steady intervals being merged into increasing and decreasing trends. Setting *dur* to 20 results in 128 intervals being generated (Fig 13). Here longer than expected increasing and decreasing trends are produced. Such trends can be considered as long term trends. This is a reduction of points to intervals of 9.

Setting low values for the parameters g_1 and g_2 results in all rates of changes in the data being captured. Setting g_1 to 0.1 and g_2 to 0.5 results in 197 intervals being generated (Fig 14). This is a reduction of points to intervals of 5.87. Here consideration is being given to the different rates of change in the data. Here many intervals could be merged into larger super-intervals. Setting high values for the parameters g_1 and g_2 results in a small number of intervals being generated. Setting g_1 to 5 and g_2 to 10 results in also 197 intervals being generated (Fig 15). This is a reduction of points to intervals of 5.87. It seems that g_1 and g_2 has no significant effect on the number of intervals generated. The best combination for this data set is to set the parameters *diff* to 1, g_1 to2, g_2 to 3 and *dur* to 10 (Fig 16). We chose these values because external air temperature has a variability of 1^{o}C, we wish to differentiate between gradients of 2 and 3 and we wish long term trends. This has reduction of points to intervals of 18.33. A summary of the effects of these setting are given in table 1.

The settings of the size of the filter window *k* for the *filter data* process and the parameters *diff*, *dur*, g_1 and g_2 for the *interval identification* process for historical data will depend on the variability of the data and how few or many intervals are desired.

TABLE 1: Results for various temporal inferencing parameters

diff	dur	g_1	g_2	Intervals	Data reduction	Comments
Low e.g., 0.1	5	1	3	202	5.7 : 1	Too many steady intervals – see Fig 10
High e.g., 10	5	1	3	7	165 : 1	Many increasing and decreasing trends missed – see Fig 11
0.5	Low e.g., 0.5	1	3	328	3.5 : 1	Too many steady intervals – see Fig 12
0.5	High e.g., 20	1	3	128	9 : 1	Misses a few rates of change – see Fig 13
0.5	5	Low e.g., 0.1	Low e.g., 0.5	197	5.9 : 1	Captures rates of change – see Fig 14
0.5	5	High e.g., 5	High e.g., 10	197	5.9 : 1	Captures many rates of change – Fig 15 similar to Fig 14
0.5	10	2	5	63	18 : 1	Best combination – see Fig 16

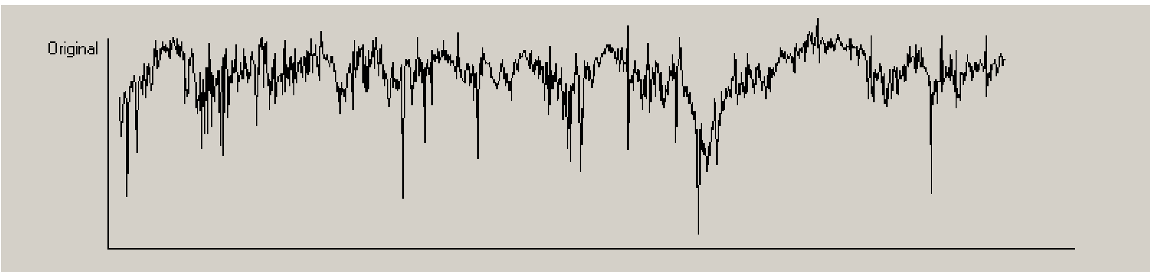

FIG 9: Original data – 1155 points.

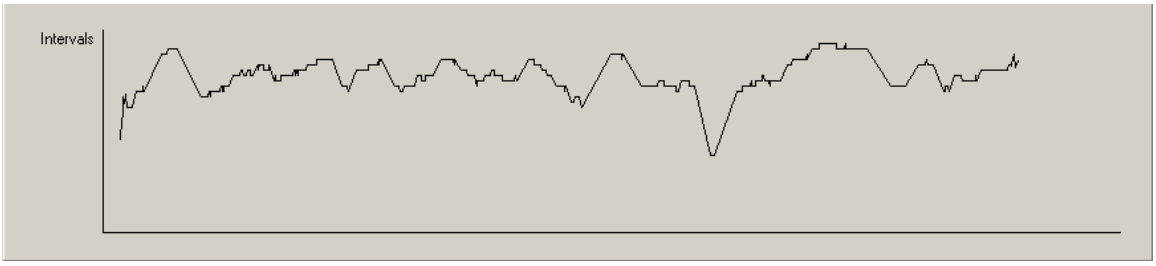

FIG 10: Low diff - diff =0.1, dur=5, g_1 =1 g_2 =3, intervals = 202

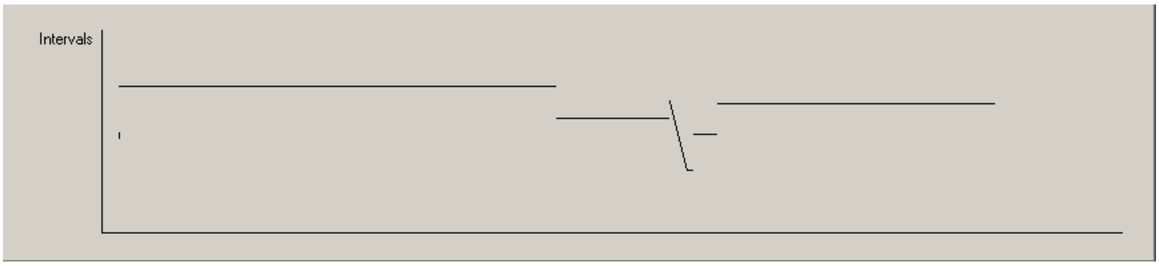

FIG 11: High diff - diff=10, dur=5, g_1 =1 g_2 =3, intervals = 7

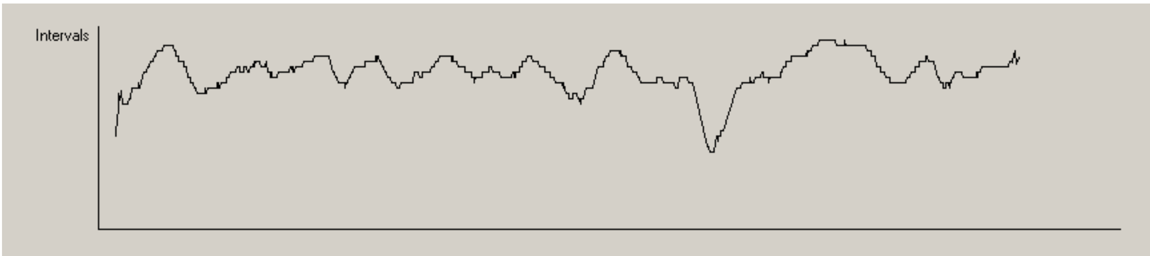

FIG 12: Low dur - diff=0.5, dur =1, g_1 =1, g_2 =3, intervals =328

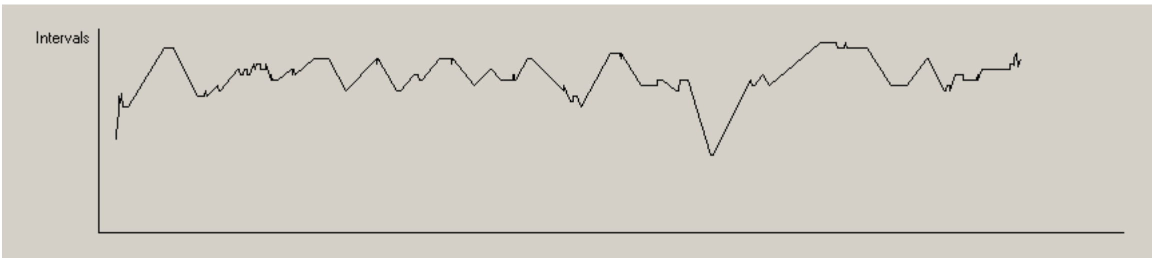

FIG 13: High dur - diff =0.5, dur = 20, g_1 =1, g_2 =3, intervals =128

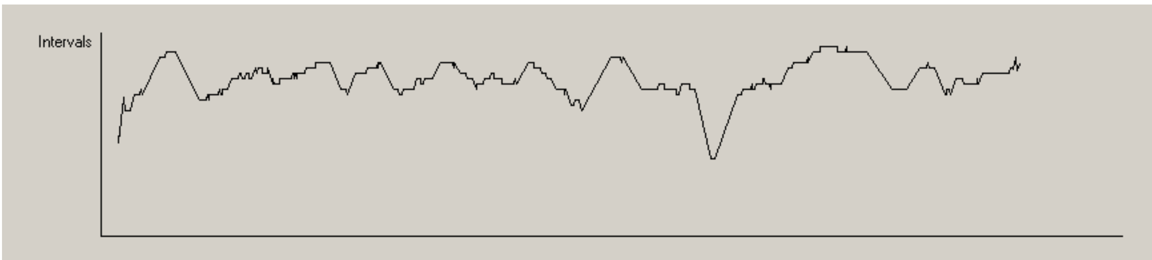

FIG 14: Low g_1 - diff=0.5, dur=5, g_1=0.1, g_2 = 0.5, intervals = 197

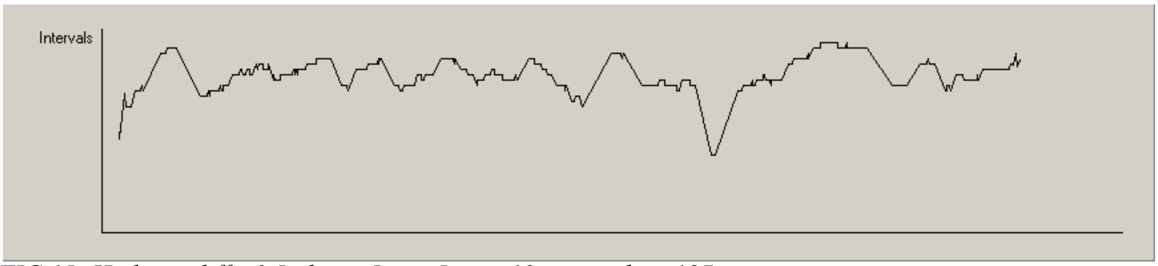

FIG 15: High g_1 - diff =0.5, dur = 5, g_1=5, g_2 =10, intervals = 197

FIG 16: Best combination - diff = 1, dur = 10, g_1 =1, g_2 = 5, intervals = 63

3.3 Interpretation of linear trends

The final stage is interpretation where linear trends in several concurrent data sets compared to identify events that are of interest.

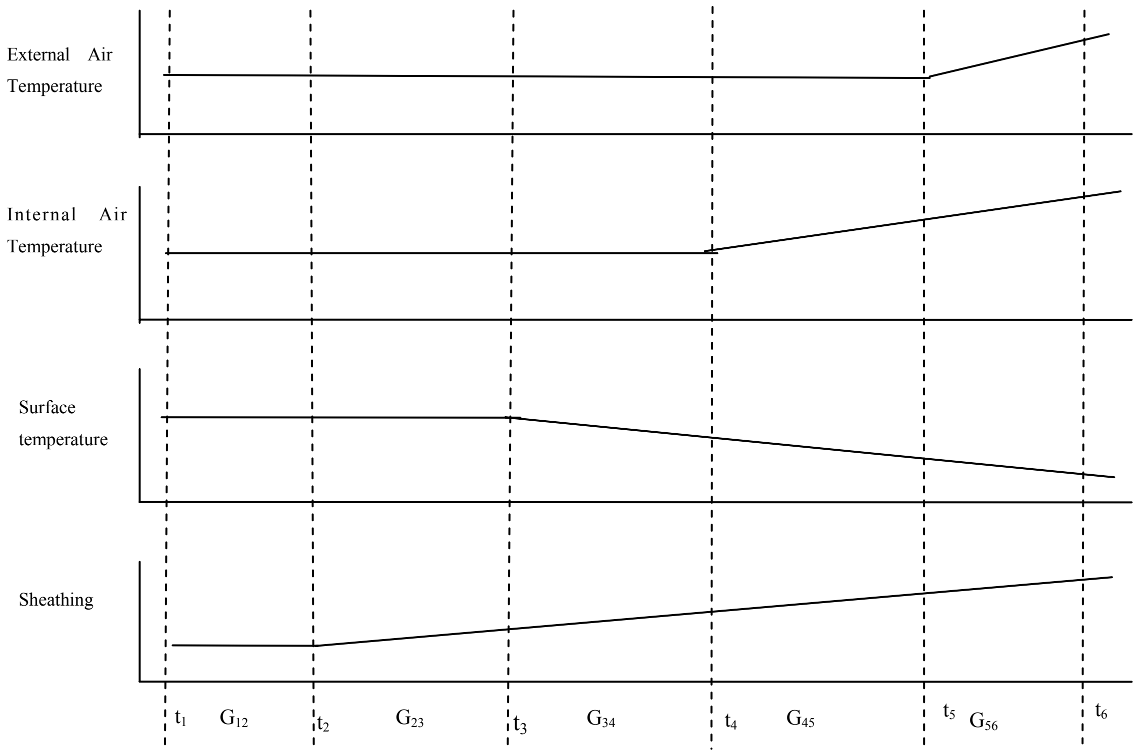

FIG 17: Splitting overlapping intervals into global segments

Given overlapping temporal intervals it is proposed, in the spirit of (DeCoste, 1991) they are split into *global segments*. A change in the direction of change (slope) of one (or more) channels or a change in the rate of change of one (or more) channels contributes to a split in the temporal intervals creating a global segment. For example, in Fig 17 a global segment, G_{12}, lasting from t_1 to t_2 would be created in which the intervals for *External Air temperature, Internal Air Temperature, Surface Temperature* and *Sheathing Temperature* all have the abstraction steady - this global segment is created because of the change in direction of change of *Sheathing Temperature* at time t_2. A global segment can be considered as being a set of intervals - one for each channel. Likewise global segments would be created lasting from t_2 to t_3, t_3 to t_4, t_4 to t_5 and t_5 to t_6.

The algorithm for interpretation involves applying the rules to the global segments. Examples of rules for identifying events are shown in Fig 18 – here a fault is declared when the heat-flux does not have the same trend as the difference in internal and external temperature (t1-t0). These rules make up the knowledge base called *Environmental Events* shown in Fig 1. If rules are true over adjacent global segments then one can determine when the environmental event started and ended.

```
If heat-flux increasing and ti-t0 decreasing then
        fault detected
end if

If heat-flux increasing and ti-t0 steady then
        fault detected
end if

If heat-flux decreasing and ti-t0 increasing then
        fault detected
end if

If heat-flux decreasing and ti-t0 steady then
        fault detected
end if

If heat-flux steady and ti-t0 increasing then
        fault detected
end if

If heat-flux steady and ti-t0 decreasing then
        fault detected
end if
```

FIG 18: Example of rules to apply to global segments

4. RESULTS OF ABSTRACTOR

ABSTRACTOR has been tested on over 8 days (12179 minutes) worth of continuous data (Fig 19a). The data was the heat-flux into a wall and the difference in internal and external temperature (ti-t0) measurements; the sampling frequency of the signals is one data item every 15 minutes. No prior knowledge of events that occurred within this data set were known to the expert or the tester. The application of the average filter (k=10 filter provides a running five and a quarter hour running average) is shown in the middle graph (b) and the intervals generated are shown in the bottom graph (c).

The output from applying the rules in Fig 18 to the abstracted temperature difference and heat flux data (Fig 19(c)) are compared with periods the domain expert identified as faults also defined by rules in Fig 18. In order to determine the efficiency of ABSTRACTOR the following parameters need to be evaluated: sensitivity, specificity, predictive value, false positive rate and false negative rate (defined in Appendix 1). These are, in turn, based on identifying true positives and true negatives from the data.

A true positive (TP) occurs when ABSTRACTOR correctly identifies periods of time where there were faults. A false positive (FP) occurs when ABSTRACTOR identifies periods of time where there were faults but actually none occurred. A true negative (TN) occurs when ABSTRACTOR correctly identifies periods of time where there were no faults. A false negative (FN) occurs when ABSTRACTOR identifies periods of time where there were no faults but there were actually faults. These can be summarised in table 2.

TABLE 2: Four categories of fault diagnosis

	Actual condition	
ABSTRACTOR identifies	**Fault**	**No fault**
Fault	TP	FP
No fault	FN	TN

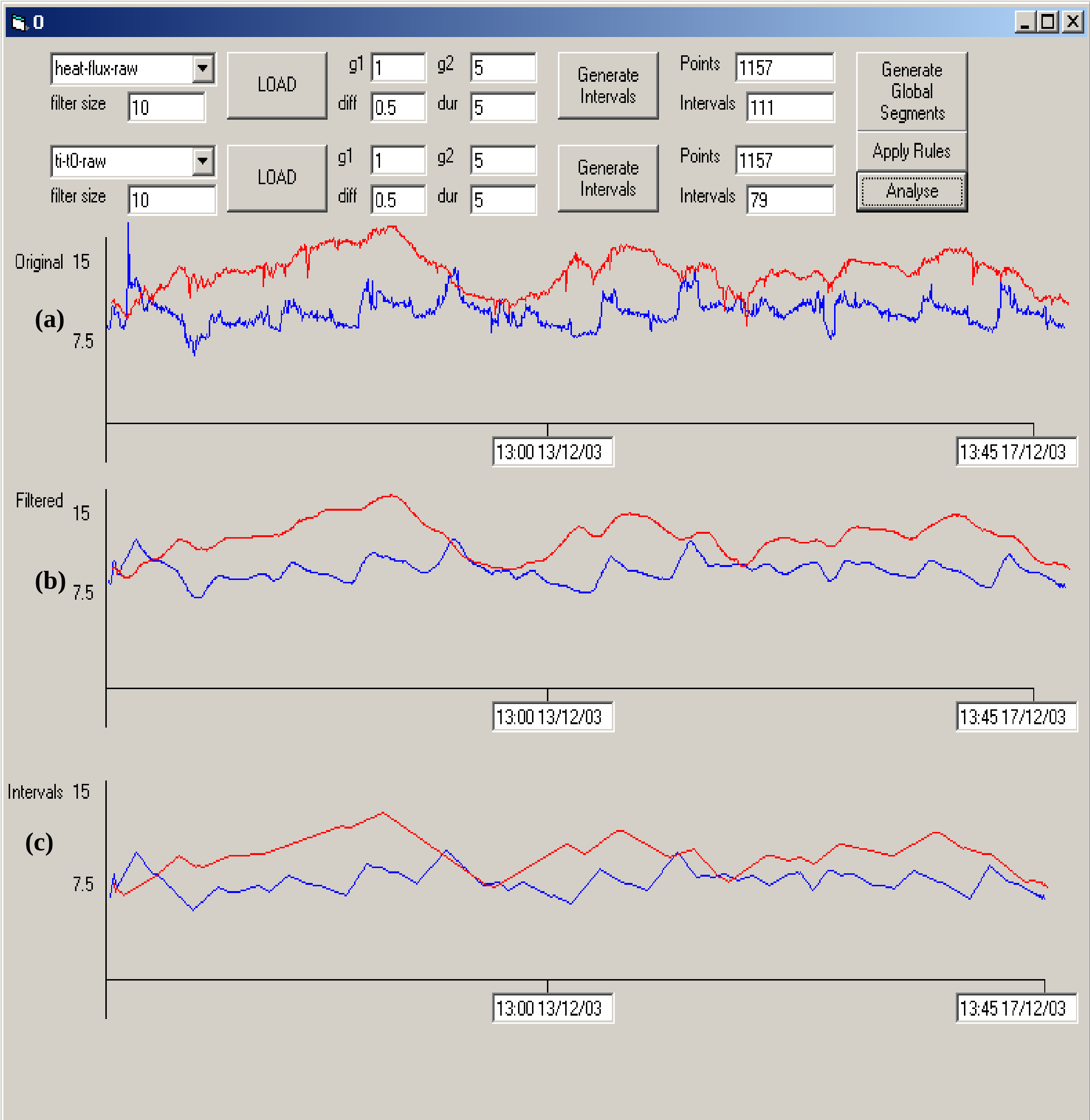

FIG 19: ABSTRACTOR applied to environmental data

Overall, ABSTRACTOR has a sensitivity of 56%, specificity of 64%, predictive value of 43%, a false positive rate of 57% and a false negative rate of 24%. These results mean that when a fault is present ABSTRACTOR is detecting it only 56% of the time but when there is no fault it will correctly identify this 64% of the time. Whilst it would seem that ABSTRACTOR is only slightly better than tossing a coin to decide the presence or absence of a fault it needs to be remembered that the actual fault conditions were derived from an expert's manual abstraction of the raw data which is dependent on the expert's attitude and experience. A direct comparison with the raw data is meaningless because the data is at intervals much shorter than the trends. If ABSTRACTOR were to be incorporated in its present state into a control system it would generate a high number of false alarms (57%) but would fail to detect a fault only 24 % of the time. These results are indicating that ABSTRACTOR is a more liberal system than a random system (Fawcett, 2003).

The raw data as can be seen from Fig 19 is non-stationary (the statistical properties of the data vary with time) and so all the standard techniques for processing stationary random data (Bendat and Piersol, 2000) are not applicable. The heat flux data is showing similarities to a pulsed or switching signal as if a source of heat is being switched off and on. This important feature is lost when any of the filters considered in this study are used. In particular a five and a quarter hour running average is too long to preserve sharp transitions whereas a shorter running average would not provide sufficient smoothing of the random noise.

Abstraction as a technique for analysing the large volume of data generated by building monitoring equipment has potential to provide insight into the data, as demonstrated by Fig 19(c). It also reduces the number of data points describing the data to number of intervals plus one which represents a greater than 10:1 compression ratio for the data.

5. CONCLUSIONS

The data routinely collected by BEMS is often not fully exploited to identify poor performance and faults in building energy systems due to the vast amount of data collected from sensors. Furthermore, with the trend away from air conditioning systems controlling building environments to within relatively tight limits, towards mixed mode buildings where internal conditions will fluctuate over a much wider range there is a need to identify and reduce concurrent time series data sets to a set of linear trends. This would facilitate the understanding of the inter-relationship between trends on different data streams and help identify significant events.

The algorithm ABSTRACTOR provides a means of doing this. The algorithm was tested on time series heat flux and internal and external temperature data. Evaluation of different filter types for pre-processing the data showed that the average filter proved the best choice for this type of data. A number of parameters which control the agglomeration process need to have values set. The optimum values for these were established by an iterative process thus enabling a set of 1155 data points to be summarised as 63 line segments (64 points). This represents a data compression ration of 18:1.

ABSTRACTOR's ability to identify significant events on concurrent data sets was evaluated using the heat flux and temperature data. Here ABSTRACTOR's performance was less encouraging. It performed moderately better at detecting faults than a purely random system. A possible explanation is that the results depend on the way the expert manually identified the environmentally significant events. A reasonable data compression of 10:1 was achieved on these two concurrent data streams.

Work is ongoing in two main areas: improving the sensitivity and specificity of the algorithm, establishing the range of validity of the algorithm parameters for much longer non-stationary data sets and to environmental variables with different statistical properties.

6. ACKNOWLEDGEMENTS

The authors thank T Morton of Arc-Architects and F. Stevenson, University of Dundee, Faculty of Duncan of Jordanstone College, for the use of time series from a DTI funded the PII project (O-02-TMA1) "Low cost earth brick construction: monitoring & evaluation".

7. REFERENCES

ASHRAE. (2003). Applications Handbook (SI), Chap 40 Building Energy Monitoring, ASHRAE, Atlanta, Georgia.

Bendat, J S. and Piersol, A G. (2000). Random Data: analysis and measurement procedures, 3rd ed, John Wiley & Sons, New York.

Braun, J. and Li, H. (2003) Automated Fault Detection and Diagnostics for Rooftop Air Conditioners, Final report for the Building Energy Efficiency Program sponsored by California Energy Commission, Purdue University.

Brooks, C E P. and Carruthers, N. (1953). Handbook of Statistical Methods in Meteorology, HMSO, London.

CIBSE. (2002). Weather, solar and illumination data: CIBSE Guide Part J, London .

CIBSE. (2004). Energy Efficiency in Buildings, CIBSE Guide F, 2nd edition, London.

Craggs, C., Conway, E M. and Pearsall, N M. (2000). Statistical investigation of the optimal averaging time for solar irradiance on horizontal and vertical surfaces in the UK , *Solar Energy*, 68, 2, pp 179-187.

DTI. (2003). Our energy future – creating a low carbon economy, Energy White paper Cm 5761, TSO (The Stationary Office), London.

DeCoste, D. (1991). Dynamic Across-Time Measurement Interpretation, *Artificial Intelligence* 51, pp 273-341.

Dexter, A. and Pakanen, J. (2001). Demonstrating Automated Fault Detection and Diagnosis Methods in Real Buildings, IEA annex 34 Energy Conservation in Buildings and Community Systems, Symposium 217, VTT, Espoo, Finland.

T. Fawcett. (2003). *ROC Graphs: Notes and Practical Considerations for Data Mining Researchers*, Intelligent enterprise Technologies Laboratory, HP Laboratories Palo Alto, HPL-2003-4, January 7.

Field, J., Soper, J., Jones, P., Bordass, W., Griggs, P. (1997). Energy performance of occupied non-domestic buildings: Assessment by analysing end-use energy consumption, *Building Services Engineering Research and Technology*, 18(1) pp39-46.

Gallagher Jr, N C. (1988). Median Filters: A Tutorial, *Proceedings of ISCAS '88*, pp 1737-44.

Hamming, R W. (1989). Digital Filters, 3rd ed., Dover Publications, New York.

Liao, Z. and Dexter, A L. (2003). An inferential control scheme for optimizing the operation of boilers in multi-zone heating systems, *Building Services Engineering Research and Technology*, 24, 4, pp 245-256.

Liddament, M W. (1999). Technical Synthesis Report: Real Time Simulation of HVAC Systems for Building Optimisation, Fault Detection and Diagnostics, *ESSU, (Air Infiltration and Ventilation Centre), Coventry.*

Page, J. (1986) Climate in the UK, HMSO, London.

Stringer, E T. (1972). Techniques of Climatology, Freeman & Company, San Francisco.

Rengaswamy, R. and Venkatasubramanian, V. (1995). A Syntactic Pattern-Recognition Approach for Process Monitoring and Fault Diagnosis, *Engineering Applications of Artificial Intelligence*, 8, 1, pp 35-51.

Shen, Q. and Chouchoulas, A. (2000). A Modular Approach to Generating Fuzzy Rules with Reduced Attributes for the Monitoring of Complex Systems, *Engineering Applications of Artificial Intelligence*, 13, pp 263-278.

Yu, B., van Paassen, AHC., Rishy, S. (2003). Open Window and Defective Radiator Valve Detection, *Building Services Engineering Research and Technology*, 24, 2, pp 117-124.

Appendix 1

Sensitivity is the proportion of faults correctly identified i.e the ability to detect true positives.

$$Sen = \frac{N_{TP}}{N_{TP} + N_{FN}}$$

Specificity is the proportion of cases correctly identified as being fault free i.e the ability to detect feature-free data.

$$Spc = \frac{N_{TN}}{N_{TN} + N_{FP}}$$

Where N_{TN}, N_{TP}, N_{FN}, N_{FP} are the number of true negatives, true positives, false negatives and false positives respectively and are measured by the duration of relevant trends since it is a time series that is being dealt with.

A positive predictive value (PPV) rate is the proportion of cases ABSTRACTOR correctly identifies as being a fault divided by the total of all positive results.

$$PPV = \frac{N_{TP}}{N_{TP} + N_{FP}}$$

False negative rates (FNR) and false positive rates (FPR) are defined as

$$FNR = \frac{N_{FN}}{N_{FN} + N_{TN}}$$

$$FPR = \frac{N_{FP}}{N_{FP} + N_{TP}}$$