

設備診断向けの時系列データ分析支援方式 —設備異常検知用の時系列データ検索言語の提案—

今村 誠[†] 高山 茂伸[†] 撫中 達司[†]

[†] 三菱電機株式会社 〒247-8501 神奈川県鎌倉市大船 5-1-1

E-mail: [†] {Imamura.Makoto@bx, Takayama.Shigenobu@db, Munaka.Tatsuji@ct}.MitsubishiElectric.co.jp

あらまし ビルやプラント等の設備の運用やエネルギーの効率向上を目的として、設備を構成する多数の機器の温度、振動、制御用設定値などの時系列データを取得・分析する設備診断のニーズが高まっている。本論文では、人手による詳細分析の候補となる設備異常の条件を検索式として宣言的に記述し、異常を示すデータと時刻を検索するための SQL ベースの検索言語を提案する。本言語の特徴は、以下の3点である。(1) 窓関数を用いた畳み込み演算の SQL への導入により、時系列データのウインドウ処理を簡潔に記述できる。(2) 窓関数として、全称制約充足窓関数と波数カウント関数を導入することにより、設備診断で典型的な時間継続制約やハンチング異常を記述できる。(3) 設備の構成情報記述により、設備構成に依存した時系列データ項目間の関係を記述できる。

キーワード 時系列データ処理、設備診断、異常検知、時間データベース

A Method of Supporting Time Series Data Analysis for Facility Diagnosis — A Proposal of a Time Series Data Query Language for Anomaly Detection —

Makoto IMAMURA[†] Shigenobu TAKAYAMA[†] and Tatsuji MUNAKA[†]

[†] Mitsubishi Electric Corporation, 5-1-1, Ofuna, Kamakura, Kanagawa 247-8501, Japan

E-mail: [†] {Imamura.Makoto@bx, Takayama.Shigenobu@db, Munaka.Tatsuji@ct}.MitsubishiElectric.co.jp

Abstract In facility management for buildings or plants, needs of facility diagnosis for saving energy or facility management cost by analyzing time series data from sensors of equipments in facilities have been increasing. In facility diagnosis, it is hoped that anomaly data are automatically chosen for manual analysis from a large amount of time series data, because manual analysis is a time-consuming task. This paper proposes a SQL-based declarative query language TPQL (Trend Pattern Query Language) for expressing constraints in time series data for anomalies detection in facilities. The features of TPQL are the following. (1) TPQL has a convolution operator with domain dependent window functions into SQL in order to describe concisely the duration constraints and window processing. (2) TPQL has a universally quantified constraint satisfiability check and a wave counter as window functions in order to describe duration constraint and hunting constraint respectively. (3) TPQL can describe the relation among time series data depending on equipment configuration with configuration management description in XML.

Keyword Time Series Data Processing, Facility Diagnosis, Anomaly Detection, Temporal Database

1. はじめに

ビルやプラント等の設備管理分野では、設備の運用やエネルギーの効率向上を目的として、設備を構成する多数の機器の温度、振動、制御用設定値などの時系列データを取得・分析する設備診断のニーズが高まっている。設備診断では、大量の時系列データの人手詳細分析は困難なため、事前に、詳細分析の候補となる設備異常を示すデータの自動抽出が望まれる。本論文では、人手による詳細分析の候補となる設備異常の条件を検索式として記述し、時系列データを対象として、異常を示すデータと時刻を検索する言語を提案する。

2. 課題

2.1. 応用課題

(1) 課題 1: 時系列データに対する制約条件の記述

ビルやプラント等の設備診断の異常検知では、「給気温度偏差が X℃ 以上である状態が Y 分以上継続し、かつ、冷温水弁開度の偏差が Z% 以上であると、弁固着異常である可能性がある」といった複数の時系列データ間の関係制約を記述できる必要がある [1]。

従来の SQL で記述にくい典型的な時系列データ処理は、一定の時間の区間のデータ列であるウインドウに対する制約記述である。以下、設備診断の異常の条件記述でよく

用いられる時間継続制約、時間スライド制約、時間集約制約、および、ハンチング異常制約を含む検索要求の例を示す。

(i) 時間継続制約

・室内温度が温度設定値よりも $X^{\circ}\text{C}$ 以上高い状態が Y 分以上継続する時間、室内温度を求めよ。

(ii) 時間スライド制約

・室内温度が上昇している時間、室内温度、温度差を求めよ。すなわち、時間差分を求めよ。

・観測データ $f(t)$ と AR(AutoRegression)モデル式による予測値との偏差(残差)が定数 T 以上である時刻を求めよ。例えば、 $f(t) - (a_1 f(t-1) + a_2 f(t-2)) \geq T$ を満たす時刻を求めよ。

(iii) 時間集約制約

・前後 X 分の値を平均して得られる室内温度(時間平均)が Y 度以上となる時間を求めよ。また、前後 X 分の値で最大値が Y 度以上となる時間を求めよ。

(iv) ハンチング異常制約

・室内温度が X 分間で $Y^{\circ}\text{C}$ 以上の振幅で上下する凹凸の波がある時間を求めよ。

(2)課題 2:設備の機器間の構成関係に依存した時系列データ間の関係制約の記述

ビルやプラント等の設備は、多数の機器から構成されており、設備診断では、数百から数万のセンサーが取得する時系列データを参照する必要がある。例えば、ビルの空調システムは、フロア毎に、空調機を中心とする複数の系統があり、その系統毎に、可変風量制御装置(VAV: Variable Air Volume)やターミナル空調機(TAC: Terminal Air Conditioner)が設置されている[1]。

そのため、時系列データ間の関係制約を記述するには、センサーが設置されている機器の構成関係に関する知識を参照する必要がある。例えば、以下に示すような機器構成情報に依存した検索要求を簡単に記述できる必要がある。

・空調機毎に、空調機の温度が、空調機の系統に属する可変風量制御装置の平均温度よりも大きい時間を求めよ。但し、空調機は複数あり、空調機毎に可変風量制御装置が複数設置されており、空調機にも可変変量にも温度センサーが付与されているものとする。

2.2. 技術課題

2.1 節の課題 1 に対して参考になる従来技術として、SQL にウィンドウ処理の概念を拡張した検索言語がある。例えば、CQL(Continuous Query Language)[3]には、ウィンドウを扱う Range 構文や時間スライドを扱う SLIDE 構文がある。CQL は、ストリームデータ管理を主な目的としており、ウィンドウを抽出・処理のため

の基本的な演算を提供することを特徴としている。そのため、設備診断用の時系列データ検索言語として必要とされる、信号処理や時系列解析で用いる時間の関数に対する制約を簡潔に記述するための機能は、提供されていない。

2.1 節の課題 2 に対して参考になる従来技術としては、半構造データ記述の標準である XML (Extensible Markup Language) や RDF(Resource Description Framework)に対する検索言語である XQuery(XML Query)や SPARQL(SPARQL Protocol and RDF Query Language)がある[4]。XML や RDF により機器間の関係を表現すれば、XQuery や SPARQL の半構造データの検索言語により、設備の機器間の構成関係に関する制約を記述できる。しかし、機器の構成関係と、機器のセンサーがもつ時系列データとは、データモデルが異なるため、課題 1 と両立が難しいという問題がある。

3. 設備異常検知用時系列データ検索言語

3.1. 設計方針

2 章で述べた 2 つの課題を同時に満たす設備異常検知用の時系列データ検索言語を定義する。

(1) データモデル

検索言語が対象とするデータを、設備の時系列データに限定する。具体的には、対象とするデータを、「時間をキーとするテーブルデータ(時系列データテーブル)」と、「時系列データを取得するセンサーの設置機器の構成関係(設備構成情報)」に限定する。前者は関係データモデル、後者は半構造データモデルで表現し、関係データモデルのカラム名に関する関係を後者で表現することで、両者を結びつける。

(2) 検索言語の構文・目的・記述レベル

・関係データベースの標準的な検索言語 SQL(Structured Query Language)の SELECT・FROM・WHERE 構文をベースとする。

・データモデルを制限することにより、「WHERE に示す設備異常の制約条件を満たす異常時間を検索する」という検索式を簡潔に記述できるようにする。

・設備診断向けで、CQL より上流の時系列データに対する制約記述を提供する。より具体的には、WHERE 句の制約条件記述では、信号処理や時系列解析などにみられるように、時系列データを時間の関数とみなした畳み込み和やフーリエ変換などを簡潔に記述できるようにする。

(i)課題 1 への対応方針:畳み込み作用素による時系列データの制約条件の記述

課題 1 に統一的に対応するために、ウィンドウ処理を、設備異常検知の主要な道具の一つである信号処理やシステム制御[5]でよく用いられる畳み込み和の概

念を用いる。

畳み込み和とは、信号処理や微分方程式でよく用いられる畳み込み積分を離散化したものである。畳み込み和は、データ処理の観点からみると、ウィンドウをずらしながら、ウィンドウ毎に同じ処理を反復適用する処理に相当する。例えば、時系列データ $f(t)$ と矩形ウィンドウ関数 $W_R(t)$ との畳み込み和 $f * W_R(t)$ は、 $f(t)$ の前後 1 時間の時間平均を求める処理に対応する。ウィンドウ関数の台(0 でない値をとる時刻の集合)の長さがウィンドウのレンジ幅に対応している。

$$f * W_R(t) = \sum_{\tau} f(t) W_R(t - \tau) \quad (1)$$

$$\text{但し、} W_R(t) = \begin{cases} 1/60 & \text{if } |t| \leq 30\text{min} \\ 0 & \text{otherwise} \end{cases}$$

課題 1 解決のための主なアイデアは、「ウィンドウ処理を、時系列データの時間毎に、その時間を基準として抽出したウィンドウデータ列に対して、関数を反復作用させる操作として、畳み込み和として表現する」である。すなわち、ウィンドウデータ列の抽出とウィンドウデータ列に対する関数の作用を組み合わせたものを、畳み込み作用素として表現する。

すなわち、畳み込み和

$$\sum_{\tau} f(t) W_R(t - \tau) \text{ を、}$$

時系列データ $f(t)$ に対する作用として、 $\text{conv}(W_R)(f(t))$ と記述する。すなわち、 $\text{conv}(W_R)$ を、「信号関数 $f(t)$ を入力として、畳み込み和をとることにより新たな時系列データを返す作用素」としてとらえる。ここで、 W_R を窓関数、 $\text{conv}(W_R)$ を畳み込み作用素と呼ぶ。

具体的には、(1) であげた畳み込み和を、 $\text{conv}(\text{avg}(\text{range}=60, \text{slide}=-29))(f)$ で表現する。ここで、 $\text{avg}(\text{range}=60, \text{slide}=-29)$ は、サンプリング周期が分である時系列データを対象として、「ある時刻に対して、その時刻の -29 分前から 60 分間のデータを切り出し、その平均 avg を計算すること」を表現している。

次に、設備診断の異常検知のための時系列データの制約条件を簡潔に記述できるようにするために、分野特化の窓関数を定義していく。

(ii) 課題 2 への対応方針: 信号変数と機器構成情報 XML 記述による時系列データ間の関係制約の記述

・検索式で、時系列データの名前(時系列テーブルのカラム名に相当する)を変数として扱えるようにする。提案言語中では、時系列データを時間の関数である信号とみなし、時系列テーブルのカラム名を信号名、変数を信号変数と呼ぶ。本論文では、時系列データを時間の関数と解釈することを強調する際には、信号と呼ぶことにする。

・機器構成情報に基づく信号間の関係は、機器構成情報を表現する半構造データ(例えば、XML)に対する検索言語(例えば、XPath 式)で表現する。

・機器構成要素毎の異常時間検索を記述できるように、XPath 式で指定される信号毎に検索を実行する `foreach` 文を提供する。

・XPath 式で指定される信号の集合を入力として、平均、最大などの信号の集合を集約した信号を返す集約関数を提供する。

3.2. 時系列データ検索言語の構文と意味

図 1 に、提案する時系列データ検索言語の構文を BNF(Backus-Naur Form)表記にて示す。本論文では、提案言語を TPQL(Trend Pattern Query Language)と呼ぶ。図を参照しながら、構文の主要要素である検索式、信号名、時間検索式、信号関数、時間制約式の構文と意味を説明する。

(1) 検索式

・<検索式>は、`foreach` 文、または<時間検索式>である。
・`foreach` 文は、XML ファイル<信号構成情報 XML>において、XPath 式<Xpath>を満たすノードリストの要素毎、すなわち、時系列データテーブルのカラムである信号名毎に、`{}` 中の<検索式>を実行する。`{}` 中の<検索式>では、`foreach` 文の文脈で参照している<信号変数>を参照できる。

(2) 信号名

信号名は、<信号変数>、または、<信号名インスタンス>である。<信号名インスタンス>は、時系列データテーブルのカラム名を指している。

(3) 時間検索式

<時間検索式>を評価するとは、「<テーブル名>が指し示す設備の時系列データに対して、<診断条件式>を満たす時間を求め、その時間の<信号関数>の値を取り出すこと」を意味する。

(4) 信号関数

信号関数は、<信号名>、<集約信号>、<算術信号>、<畳み込み信号>、または、<信号変換>である。

・<信号名>は、時系列データテーブルのカラム名であり、時系列データテーブルのキーである時間に対して、カラムの値を対応させる時間の関数を意味する。

・<算術信号>は、信号関数の和、差、積、商などの信号関数の算術演算により信号関数を得る操作を意味する。

・<畳み込み信号>は、<信号関数>を入力として、<窓関数>に対応する畳み込み演算の結果として得られる信号関数を返す操作を意味する。畳み込み信号は、詳細を 3.3 節で説明する。

・<変換関数>は、<信号関数>を入力として、<信号関

数>を返す変換操作を意味する。ローパスフィルタなどの信号変換処理を表現するために用いる。

```
<検索式> ::= foreach {
    <信号リスト> as <信号変数>
    in <信号構成情報 XML>
    } { <検索式>
    | <時間検索式>
}
#< 信号リスト>とは、XML ファイルである<信号構成情報 XML>のノード名の集合を示す XPath 式とする。
<信号> ::= <信号変数> | <信号名>
<時間検索式> ::= SELECT TIME (","<信号関数>)*
    FROM <テーブル名>
    WHERE <時間制約式>
<信号関数> ::= <信号名> | <集約信号> | <算術信号>
    | <畳み込み信号> | <信号変換>
<算術信号> ::= <信号関数> <算術演算子> <信号関数>
<集約信号> ::= <集約関数> "(" <信号リスト> ")"
<集約関数> ::= "max" | "min" | "sum" | "avg" | "std"
    等の集約関数
<算術演算子> ::= + | - | * | /
<畳み込み信号> ::=
    conv "(" <限量子付窓関数> ")" "(" <時間制約式> ")"
    | conv "(" <集約窓関数> ")" "(" <信号関数> ")"
<限量子付窓関数> ::= <全称制約窓関数> | <存在制約窓関数>
<集約窓関数> ::= <集約関数> | <差分関数>
    | <スライド関数> | <N 変数 1 次関数> | <波数関数>
<信号変換> ::=
    変換関数 "(" <信号関数> "(" <信号関数> ")" * ")"
# 信号変換は、ローパスフィルタのように、信号を入力として、信号を返す変換である。
<時間制約式> ::= <時間基本制約式>
    | <時間制約式> and <時間制約式>
    | <時間制約式> or <時間制約式>
<基本時間制約式> ::= <信号関数> <θ> <Const>
<θ> ::= "=" | "≤" | "<" | "≥" | ">"
```

図 1 時系列データ検索言語 TPQL の構文

(5) 時間制約式

<信号関数> = 5 のように、真偽値が定まる信号関数の制約条件であり、かつ、制約条件が真となる時刻(または、時間区間)を決めることができる制約式を時間制約式と呼ぶ。時間制約式の評価結果は、制約式を満たす時刻(または、時間区間)の集合とする。以後、時刻、または、時間区間の集合を時間集合と呼ぶ。ここで、時間区間を扱う場合は、サンプリングされた時

間の間を補間する等により、離散的な時刻の値を時間の区間上の値として解釈するものとする。

foreach 文の場合は、foreach 文の文脈毎に、時間集合が定まるので、評価結果は、時間集合の集合になる。これは、設備の機器毎に、異常となる時間集合があることを表現している。

・<時間基本制約式>は、<信号関数> = 5 のように、信号関数の値に関する制約式である。=は、不等号でもよい。

・<時間制約式> の論理和 or、論理積 and は、<時間制約式>である。<時間制約式>の論理和、論理積の意味は、<時間制約式>の評価結果として得られる時間集合の集合和、集合積である。ここでは、論理和と論理積のみを扱ったが、時間データベース(例えば、TSQL2[6])で既に扱われている時間に関する比較演算子(BEFORE, AFTER, DURING, ADJACENT, EQUIVALENT, OVERLAP, FOLLOWS, および、PRECEEDINGS など)は、時刻や時間区間に関する比較演算を定義し、時間集合に対する比較演算に拡張すれば、時間制約式の関係として定義することもできる。

【基本時間制約式の記述例】

室内温度が温度設定値よりも X℃以上大きく、かつ、冷温水開度が Y 以上となる時刻、室内温度を求めよ。

```
SELECT time, 室内温度
FROM 時系列データテーブル
WHERE 室内温度 - 温度設定値 ≥ X
    and 冷温水開度 ≥ Y
```

時系列データテーブルは、time をキーとして、室内温度、温度設定値、冷温水開度をカラムとするテーブルである。カラムは、値として実数値をとる。

上記の問合せ式の評価結果は、信号を時刻の関数として、室内温度(t)のように表記すると、以下に示す集合になる。

```
{ (t, 室内温度(t)) |
    (室内温度(t) - 温度設定値(t)) ≥ 1
    and 冷温水開度(t) ≥ Y }
```

基本時間制約式の評価結果は、時刻の関数に対する制約式の解として、自然に解釈できることがわかる。

3.3. 畳み込み作用素によるウインドウ処理の表現

ウインドウ処理は、時系列データの各々の時刻(基準時刻と呼ぶ)に対して、基準時刻から相対的な時間関係にある時刻の値の列を構成するウインドウデータ列抽出(抽出した値の列をウインドウデータ列と呼ぶ)と、ウインドウデータ列を入力として値を返す関数を作用させた結果得られる値から時系列データを構成するウインドウデータ列処理からなる。

3.3.1. ウィンドウデータ列抽出

ウィンドウデータ抽出には、スライド幅 `slide`、レンジ幅 `range`、シフト幅 `shift` の3つのパラメータがある。

スライド幅 `slide` では、抽出を開始する時刻の基準時間からみた相対的な順序数、または、時間である。過去 10 単位時間前であれば、-10 とする。ここで、単位時間とは、時系列データのサンプリング周期のことをさす。サンプリング周期が 1 分であれば、10 単位時間とは 10 分をさす。

レンジ幅 `range` は、抽出するウィンドウのデータ数、または、時間である。抽出したウィンドウデータ列の長さに対応する。

図 2 に、スライド幅 `slide`、レンジ幅 `range` のイメージ図を示す。

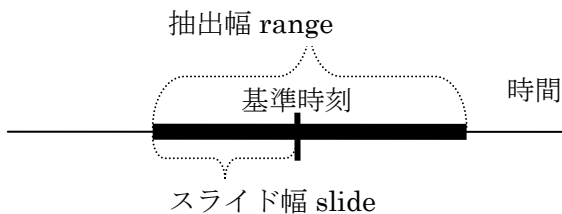


図 2 ウィンドウのスライド幅とレンジ幅

シフト幅 `shift` は、図 3 に示すように、ウィンドウ抽出の開始時刻と直前に抽出したウィンドウとの間隔の時間幅である。畳み込み和では、多くの場合、和をとる時刻は、サンプリング周期と同じ 1 単位時間である。1 以外のシフト幅 `shift` を指定することにより、ウィンドウの重なり幅を調整できる。

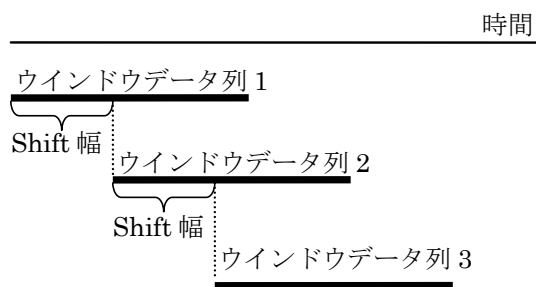


図 3 ウィンドウのシフト幅

3.3.2. 窓関数を用いたウィンドウデータ列処理

窓関数は、ウィンドウ抽出処理のパラメータと、抽出したウィンドウデータ列に対して作用させる関数を指定する。

窓関数の構文は、<関数名>[関数の引数](`slide`,`range`,`shift`)である。(slide,range,shift)は、ウィンドウ抽出のパラメータであり、窓関数の台を指定している。

以下順に、窓関数を用いた畳み込み演算によるウィンドウ処理の例を示す。シフト幅 `shift` は通常 1 なので、

以下の例では、シフト幅 `shift` の記載を省略する。

窓関数は、ウィンドウデータ列の値毎に制約条件の充足を判定する**限量子付窓関数**と、ウィンドウデータ列を一つの値に集約した後、制約条件の充足を判定する**集約窓関数**とに分類できる。

(1) 限量子付窓関数による時間継続制約の表現

ウィンドウデータ列に対して、データ列中のすべての値がある条件式を満たすとき 1(真)を返し、それ以外は 0(偽)を返す作用に対応する窓関数を、**全総制約窓関数 `all_satisfy`** と呼ぶ。同様に、データ列中のある値が条件式を満たすときに、1(真)を返し、それ以外は 0(偽)を返す作用に対応する窓関数を、**存在制約窓関数 `exist_satisfy`** と呼ぶ。

【継続時間制約の記述例】

室内温度が温度設定値よりも $X^{\circ}\text{C}$ 以上高い状態が Y 分 ($Y-1$ 分前からその時刻まで)以上継続する時刻と室内温度を求めよ。

```
SELECT time, 室内温度
FROM 時系列データテーブル
WHERE
  conv( all_satisfy(range=Y slide=-Y-1)( * > X)
      (室内温度 - 温度設定値) )
      = 1
```

但し、室内温度と温度設定値は、サンプリング周期 1 分のデータとする。

(2) 集約窓関数によるウィンドウデータ系列の制約の表現

集約窓関数には、SQL の集約関数に対応する関数 (例えば、avg, max など)、差分 `diff`、 N 変数関数、波数カウントなどがある。

(i) 時間平均窓関数 `avg`、時間最大値窓関数 `max`

ウィンドウデータ列の平均値を返す窓関数を時間平均窓関数 `avg` と呼ぶ。また、ウィンドウデータ列の最大値を返す窓関数を時間最大値窓関数 `max` と呼ぶ。SQL の集約関数である `sum`, `std`, `min` など、同様に窓関数として定義できる。

【時間平均の記述例】 各々の時刻の前後 X 分の値を平均して得られる室内温度(室内温度の時間平均)が Y 度以上となる時刻と、その時刻の室内温度の時間平均を求めよ。

```
SELECT time,
      conv(avg(range=X,slide=-X/2))(室内温度)
      as time_average
FROM 時系列データテーブル
WHERE time_average ≥ Y
```

【時間最大値の記述例】 各々の時刻の前後 X 分で、

室内温度の最大値が Y 度以上になる時刻を求めよ。

```
SELECT time,  
       conv(max(range=X,slide=-X/2))(室内温度)  
       as time_max  
FROM 時系列データテーブル  
WHERE time_average ≥ Y
```

(ii) 差分窓関数 diff

ウィンドウのレンジ幅が 2 で、ウィンドウデータ列の後ろの値から前の値を引いた値を返す窓関数を差分窓関数 diff と呼ぶ。

【例】室内温度が上昇している時刻、室内温度、温度差を求めよ。

```
SELECT time, 室内温度,  
       conv( diff (range=2, slide=-1))(室内温度) as d  
FROM 時系列データテーブル  
WHERE d > 0
```

(iii) N 変数 1 次窓関数 linear[$a_n, a_{n-1}, \dots, a_1$]

長さ N のウィンドウデータ列に対して、N 変数の 1 次関数作用させた時系列データを返す窓関数を N 変数 1 次窓関数と呼ぶ。 $[a_n, a_{n-1}, \dots, a_1]$ は、1 次関数の係数の列を示す。すなわち、関数としては、 $a_n f(t-n) + a_{n-1} f(t-(n-1)) + \dots + a_2 f(t-2) + a_1 f(t-1)$ を示す。

【例】観測データ $f(t)$ と AR(AutoRegression)モデル式による予測値 $f(t) - (a_1 f(t-1) + a_2 f(t-2))$ との偏差(残差)が定数 T 以上である時刻を求めよ。

```
SELECT time, f  
FROM 時系列データテーブル  
WHERE  
       f - conv(linear[a2,a1](range=2, slide=-2))(f) ≥ T
```

(iv) 波数カウント窓関数 wave_count[振幅 A]

長さ N のウィンドウデータ列に対して、振幅 A 以上の波がいくつあるか示す波数を返す窓関数を波数カウント窓関数と呼ぶ。振幅 A 以上である凹凸、または、凸凹のパターンを波数 1 とカウントする。

【例】室内温度が、基準時刻からみて最近の X 分間で Y℃以上の振幅で上下する凹凸の波がある場合の基準時刻と、最近の X 分間における波数を求めよ。

```
SELECT time,  
       conv(wave_count[2](range=X, slide=-(X-1)))  
       (室内温度) as w  
FROM 時系列データテーブル  
WHERE w ≥ Y
```

(v) shift パラメータの導入による重なり区分の指定

時系列データのマイニングでは、図 4 に示すように、ウィンドウ抽出時に、間の重なりなく区分で切ってウィンドウを作る場合と、ずらしながら重なりをもつウィンドウを作る場合がある。このずらし幅は、shift パラメータで指定できる。

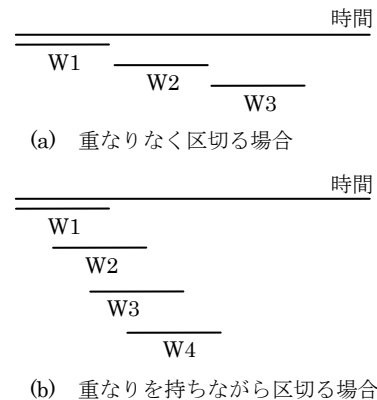


図 4 ウィンドウの作り方

レンジ幅 range とシフト幅 shift が同じ場合は、重なりなく区切る場合であり、SQL の GroupBy 演算に相当する。この意味で、畳み込み和は、GroupBy の拡張になっている。

例えば、以下のように、対象データのサンプリング周期が 1 分で、ウィンドウ切り出しの先頭が時間の開始時点とすると、以下の二つの検索式は等しくなる。

【例：畳み込みによる GroupBy の表現】

```
SELECT conv(avg(range=60,slide=0,shift=60),  
           室内温度)  
From 時系列データテーブル
```

【例：SQL による GroupBy の表現】

```
SELECT avg(室内温度)  
From 時系列データテーブル  
GROUP BY hour(time)
```

3.4. 機器構成情報 XML による時系列データ間の関係の表現例

3.1(ii)に述べたように、機器の上位下位関係、電気・給排水設備としての機器間の連結関係などの機器構成情報を表現する XML 文書を入力として、時系列データの制約式が参照する信号名を XPath 式の評価結果として取得することにより、機器構成に応じた時系列データ間の関係制約を評価する。以下に、TPQL による記述例を示す。

【例】空調機毎に、空調機の温度が、空調機の系統に属する可変風量制御装置の平均温度よりも大きい時間

を求めよ。但し、空調機は複数あり、空調機毎に可変風量制御装置が複数設置されており、空調機にも可変変量にも温度センサーが付与されているものとする。センサー間の構造は、図 5 に示す信号構成情報 XML で表現し、XML の属性に時系列データテーブルのカラム名を記載しているものとする。

```
foreach {空調機 as $A in 信号構成情報 XML }
{ SELECT time
  FROM   時系列データテーブル
  WHERE $A!温度 > avg($A /可変風量制御装置!温度)}
```

上記の“\$A /可変風量制御装置!温度”で、\$A は信号変数で、反復出現する<空調機>要素が順に束縛される。また、“/”は XML の下位要素をたどる操作、“!”は、XML の属性にアクセスする操作を示す。

foreach 文を反復実行すると、要素名が“空調機”である空調機_1、空調機_2 が順に呼び出され変数 \$A に代入される。変数 \$A に空調機_1 が代入されているときには、変数“\$A /可変風量制御装置!温度”には、空調機の下位構造にある“可変風量制御装置”の要素リスト(id が可変風量制御装置_1_1、可変風量制御装置_1_2 であるもの)が代入される。結果として、空調機_1 の温度は、「可変風量制御装置_1_1 と可変風量制御装置_1_2 の平均」と比較し、空調機_2 の温度は、「可変風量制御装置_2_1 と可変風量制御装置_2_2 の平均」と比較する、という処理を表現できる。

```
<機器構成情報>
<空調機 id="空調機_1" 温度="空調機_1_温度">
  <可変風量制御装置 id="可変風量制御装置_1_1"
    温度="可変風量制御装置_1_1_温度"/>
  <可変風量制御装置 id="可変風量制御装置_1_2"
    温度="可変風量制御装置_1_2_温度"/>
</空調機>
<空調機 id="空調機_2" 温度="空調機_2_温度">
  <可変風量制御装置 id="可変風量制御装置_2_1"
    温度="可変風量制御装置_2_1_温度"/>
  <可変風量制御装置 id="可変風量制御装置_2_2"
    温度="可変風量制御装置_2_2_温度"/>
</空調機>
</機器構成情報>
```

図 5 機器構成情報 XML

4. 時系列データ検索言語の処理方式

時系列データ検索言語の処理は、図 6 に示すように、信号変数具体化、畳み込み信号処理、および、制約充足性判定処理からなる。以下、各々の処理を順に説明する。

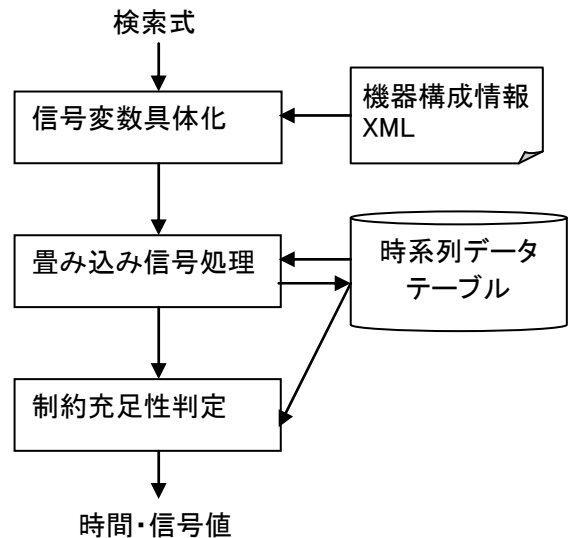


図 6 処理方式

(1) 信号変数具体化

機器構成情報 XML ファイルを参照しながら、検索式中の XPath 式のノード中のテキスト値を用いて、信号変数に具体的な信号名を割り当てる。

(2) 畳み込み信号処理

(1)で得られた検索式中の畳み込み信号に対して、畳み込み演算処理した結果得られる新たな信号を時系列データベースに登録する。

(3) 制約充足性判定

(2)で得られた検索式中の制約式を充足する時間を求め、その時間の SELECT 構文で指定する値を組とするタプルの集合を出力する。

5. 考察

TPQL は、時系列データ処理特有のウィンドウ処理を畳み込み演算として定義し、畳み込み演算を SQL に組み込み関数として追加した検索言語である。また、設備診断向けの時系列データ処理を簡潔に記述するために、全称制約、時間平均、時間最大値、差分、N 変数関数、波数カウントなどを窓関数として導入した。

本章では、ウィンドウ処理の記述が可能な従来の検索言語として典型的な CQL と比較考察する。

CQL は、ストリームデータ管理システムに対する検索言語であり、ストリームデータと SQL の関係データ間の相互変換のための作用素を提供している。ストリームデータを関係データに変換する際には、ウィンドウを抽出する機能を抽出しており、関係データへ変換後は、SQL と同様に、検索式を関係データから関係データへの関数として解釈できる。

TPQL は、関係データベースにおいて、時間をキーとする時系列データをもつテーブルに限定して、ウィンドウ処理を、テーブルのカラム(時刻の関数)に対す

る畳み込み演算により表現している。畳み込み演算は、SQLのカラムに対する算術演算と同様に、テーブルのカラムを新たに定義する操作に相当するので、SQLの意味論を保存したまま、ウィンドウ処理を導入できる。

CQLでは、ウィンドウデータ列の抽出は、ストリームデータと関係データ間の変換として表現し、ウィンドウデータ列の処理は、関係データから関係データへの変換として表現する。しかし、ストリームデータから関係データへの変換時のアプリケーション固有の処理については、対象外であった。

TPQLは、ストリームデータから関係データへの変換時に必要なアプリケーション依存の処理を、窓関数に基づく畳み込み演算として体系的に提供する枠組みを提案している。そのため、設備異常検知のための時系列データ処理を、窓関数を用いて類型化することにより、分野特化の検索式をより簡潔に記述できるようになる。また、畳み込み演算は、関数型言語のフレームワークであるMapReduceで表現できるので、分散処理化にも適していることが期待できる。

6. おわりに

本論文では、ビル・プラント設備に付与されたセンサーが取得した多数の時系列データを対象として、設備異常を検知するための制約式を記述し、その制約式を満たす時間を検索する時系列データ検索言語TPQLを提案した。

TPQLでは、対象とするデータを、「時間をキーとするテーブルデータ」と、「時系列データを取得するセンサーの設置機器の構成関係」に限定し、「時間継続や時間スライドを含む信号の制約条件の記述」と「設備の機器間の構成関係に依存した時系列データ間の関係制約の記述」の双方を満たすことを狙った。

提案言語TPQLの主な特徴は、以下の3点である。

- (1) 設備診断向けの時系列データの異常条件を記述する際に重要となるウィンドウ処理を、窓関数を用いた畳み込み演算により簡潔に記述できる。
- (2) 窓関数は、限量子付窓関数と集約窓関数に分類できる。また、設備診断向けのウィンドウ処理を記述するための窓関数として、全称制約、時間平均、時間最大値、差分、N変数関数、そして、波数カウントなどを提案した。
- (3) 関係モデルのカラム名の関係を半構造データにより表現することにより、設備の構成情報を用いた信号間の制約式を記述できる。

今後の課題は、提案言語TPQLを設備診断の実課題に適用することにより、言語の表現能力、記述の簡潔さ、処理性能(速度、データ量)が実用上十分であることを検証することである。

参 考 文 献

- [1] 木下 栄蔵 他: “ビル空調のエネルギー・環境・設備のための統計解析”, 5.2 運転データによる異常検知診断, オーム社, pp.208-236 (2006).
- [2] 新 誠一, 中野 和司: “ウェーブレット解析の産業応用”, 産業応用事例紹介(プラント分野), pp70-134, 朝倉書店 (2005).
- [3] Arvind Arasu, Shivnath Babu, Jennifer Widom: The CQL continuous query language: semantic foundations and query execution, Journal The VLDB Journal - The International Journal on Very Large Data Bases archive, Volume 15 Issue 2, (2006).
- [4] Serge Abiteboul, Peter Buneman and Dan Suciu: Data on the Web: From Relations to Semistructured Data and XML, Morgan Kaufmann Publishers (2000).
- [5] 有本卓: “信号処理とシステム制御”, 岩波書店 (1982).
- [6] Richard T. Snodgrass (Ed.): The TSQL2 Temporal Query Language. Kluwer (1995)