

動的時間伸縮の時間伸縮情報によるバッチプロセスの故障検出[†]

田村 雅之*

Fault Detection of Batch Processes Using Time Warp Information of Dynamic Time Warping

Masayuki Tamura*

Abstract: A new monitoring method of batch processes using dynamic time warping (DTW) is proposed. The method utilizes relationships between time warp information of DTW and external variables. DTW has been used as a preprocessing tool for batch process monitoring based on multiway principal component analysis (MPCA) in order to align batch data from various batch lengths by expanding and/or shrinking time axis of data trajectories flexibly. Although MPCA has experienced many successes, the fault detection ability is limited if a fault affects only the progress speed of the process and doesn't change the correlation structure between variables. In this study, another usage of DTW is proposed to overcome this difficulty. If the time warped information of DTW is correlated with external variables of the process such as temperature, initial concentration of materials and so on, changes of input-output relations between external variables and time warped information become good indicator of fault occurrences. The proposed method is applied to a fed-batch penicillin fermentation process simulator to show its effectiveness.

Keywords: fault detection, process monitoring, batch process, dynamic time warping

1. はじめに

様々な工業製品を製造する上でバッチプロセスは大きな役割を持っており、その故障を早期に発見することは重要な課題である。しかしながら、プロセスに関わる各種計測値によって故障を検出しようとした場合、定常状態にあるプロセスと異なり、各変数のとるべき値が時々刻々と変化するため、容易な作業とはならない。すなわち、単純な上限値や下限値を設定した計測値の監視では故障を感度良く検出することは困難である。このことはバッチプロセス以外にも一般にプロセスの起動時や状態遷移時においても見られる、産業応用上幅広い課題である。

バッチプロセスの監視を行なうアプローチの一つとして、Gollmer ら⁽¹⁾は音声認識の手法として開発された動的時間伸縮法(Dynamic Time Warping: DTW)を応用することを提案し、バッチプロセスのプロセスフェーズの同定や異常監視が可能であることを示した。近年盛んに研究されているバッチプロセス監視手法としては、マルチウェイ主成分分析(Multiway Principal Component Analysis: MPCA)が挙げられる⁽²⁾。MPCA は、すでに多くのプラント状態監視の実績を挙げている主成分分析 (PCA) ⁽³⁾⁽⁴⁾をバッチプロセスに拡張し

たものであり、変数間の相関構造や時系列データとしての性質に着目したモニタリングが可能である。Kassidas ら⁽⁵⁾は DTW を MPCA の前処理として使用し、データの時間方向の同期法として有用であると報告している。

さて、従来法の1つである MPCA によるモニタリングを実施した場合、故障の種類によっては必ずしも検出感度が良くないケースがあった。それは故障の発生によってデータに現れる変化が時間方向のデータ伸縮のみであり、波形の大きな変形や変数間の相関の変化を生み出さないような場合である。そこで本研究では、MPCA を補完するための DTW の新しい使用法を提案する。DTW の時間伸縮プロファイルに注目し、それがプラントの置かれた環境や原料濃度その他の外部変数に依存する場合には、外部変数と時間伸縮プロファイルの関係を予め求めておくことによって、上述のような故障についても高感度な検出が可能となると考えられる。具体例として、Birol ら⁽⁶⁾によって開発提供されたペニシリン製造のセミバッチプロセスのシミュレーションプログラムである PenSim によって生成されたデータを利用し、手法の有効性を示す。

2. 手法

2.1 従来法

2.1.1 MPCA

最初に、正常なバッチについてデータを収集する。I 回の

[†] 主な部分を「第7回評価・診断に関するシンポジウム」にて発表。

* 東京ガス(株)技術開発本部 基盤技術部 技術研究所
神奈川県横浜市鶴見区末広町1-7-7

* Technical Research Institute, Tokyo Gas Co., Ltd.

(Received May 7, 2009)

正常なバッチについて J 個のセンサを対象に各バッチごとに K 回の測定を行った場合を考える。得られたデータを、等しいバッチのデータが等しい行に入るように、また各列がそれぞれあるセンサの等しい回数目の測定となるようにデータを並べ、 $I \times JK$ の行列の形とする。各列について平均 0、標準偏差 1 となるように標準化を行ない、計 JK 個の変数について I 回の観測を行ったものと考えて主成分分析を行なうことで、正常なバッチにおける各センサの値の振る舞いをモデル化できる。診断対象の新しいデータについて Q 統計量やホテリングの T^2 統計量を算出し、これらを異常監視指標としてモニタリングを行なう。

データ行列にバッチの前処理等のデータを付加することも提案されており、本研究では初期条件や環境条件のデータを加えて解析を行った。データ行列の作り方などについては上記以外の流儀もあり、手法の詳細については参考文献²⁾を参照されたい。

2.1.2 前処理としての動的時間伸縮法 (DTW)

バッチプロセスでは、たとえば原料の純度などの条件に応じて、その反応や濃縮などに要する時間が変化する場合がある。この変化は検出したいプラントの異常ではない場合が多く、これを「問題のない」挙動であると認識するためには、MPCA による解析の前にデータの時間方向の同期をとる必要がある。すなわち、前節で述べた MPCA において、同一時間間隔で測定したデータをそのまま用いるのではなく、各バッチの等しい回数目に出てくる値がプロセスの等しい「進行度」におけるデータとなるようにしなければならない。

この問題を解決するために、Kassidas ら⁶⁾は言語認識の分野で過去に使用された DTW を利用して各バッチのデータ軌跡を時間伸縮させて同期し、その後 MPCA の計算を行なうことを提案した。

2.1.3 DTW

DTW は当初言語認識の分野で研究された。同一の言葉の発話であっても、異なる話者による場合は、その人の個性や様々な条件によってスピードが異なる。これに対して一定のルールを設けて各軌跡の時間を伸縮させ、同一の波形と見なせるかどうかの判断材料とするのが DTW である。手法にはいくつもの派生形が存在し、以下に示すものはそのアルゴリズムの一例である。

2つの似通った波形について、それら間の「距離」を時間伸縮によって最小化する問題を考える。Fig. 1 に時間伸縮の概念図を示す。波形 T と R を考え、それぞれ t 点、 r 点からなっているとす。DTW は Fig. 1 に示された軌跡

$c(k)=[i(k),j(k)]$ ($k=1,\dots,K, \max(t,r)<K<t+r$)を求める作業である。 $c(k)$ はすなわち、2つの波形のどの点とどの点とが互に対応するとみなすのかを示す、サイズ $t \times r$ のグリッド上の軌跡である。 $c(k)$ を求めるルールとして、たとえば以下のようなものを考える。 T と R の間の距離として

$$D = \frac{1}{N(w)} \sum_{k=1}^K d[i(k), j(k)] \cdot w(k) \dots \dots \dots (1)$$

を定義する。 $w(k)$ は局所距離 $d[i(k), j(k)]$ に対する重みであり、 $N(w)$ は正規化のためのファクターである。 $d[i(k), j(k)]$ はたとえば

$$d[i(k), j(k)] = [T(i(k)) - R(j(k))]^T \cdot \mathbf{W} \cdot [T(i(k)) - R(j(k))] \dots (2)$$

のように定義する。 \mathbf{W} は各変数の重要性に関する重みを決める行列である。Fig. 1 では T, R はスカラー量が並んだ波形のイメージで描いてあるが、(2) 式は一般に多くのセンサによる多変数の測定を想定し、ベクトルの並んだものとして考えている。

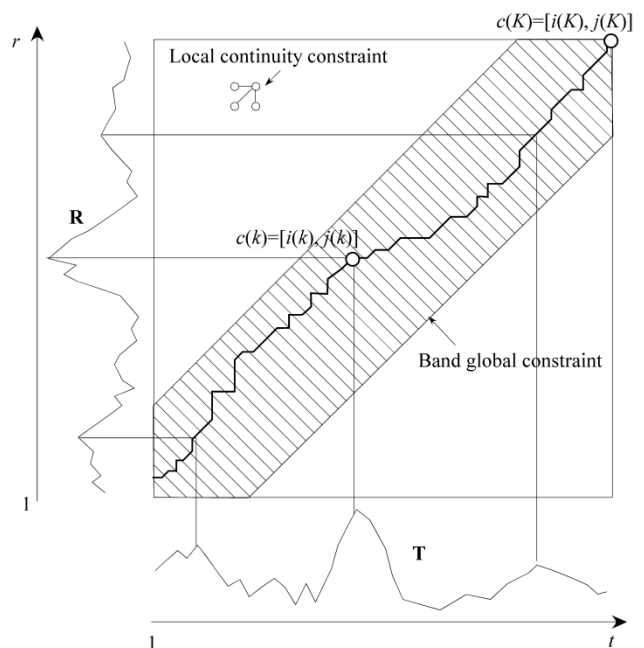


Fig. 1 Warping path of dynamic time warping.

(1)式の D を最小化するように $c(k)$ を決定する。 $c(k)$ を全く任意な軌跡から選べるようにしてしまうと、2つの波形は一致するものの、過剰な変形をしてしまうことになるため、一般に $c(k)$ の傾きを制限する Local continuity constraint と、 $c(k)$

の存在できる範囲を規定する Global constraint にしたがって計算が行われる。本研究では前者として (i, j) に遷移できるのは $(i-1, j)$, $(i-1, j-1)$, $(i, j-1)$ の3点のみとする制限を設け、後者として $c(k)$ が始点と終点を結ぶ直線から $\pm M$ の幅に収まること、という制限を設けることにする。それらを Fig. 1 に図示している。また、波形の端点についても制約を設ける。本研究では2つの波形の最終点を一致させる Fixed-endpoint 制約を採用する。

さて、上記の問題は動的計画法(Dynamic programming)によって解くことができる。すなわち、まず (i, j) に至るまでの最小累積距離を D_A として、つぎの問題を各 (i, j) について解き、その後、最終点から遡る形で総距離 D を最小とする $c(k)$ を決定することで解くことができる。ここでは (3) 式の再帰式に Kassidas ら⁽⁵⁾の提案した式を採用した。

$$D_A(i, j) = \min \begin{cases} D_A(i-1, j) + d(i, j) \\ D_A(i-1, j-1) + d(i, j) \\ D_A(i, j-1) + d(i, j) \end{cases} \dots\dots\dots(3)$$

$$D_A(1,1) = d(1,1)$$

2.1.4 従来法の問題点

バッチプロセスの進行速度が変化するのは、上記で想定したような「問題のない」初期条件や環境条件に起因する変化ばかりではない。当然、いくつかのプラント故障によっても、プロセスの進行速度は変化する。多くの場合、故障が発生するとプロセスの進行速度だけではなく、センサ間の相関も変化するため、上記の従来法で問題無く異常が検出可能であるが、ある条件においては従来法では異常を検出できない可能性がある。すなわち、測定可能なセンサすべてにおいて、プラント故障の発生の影響が、時間方向のデータ伸縮としてしか観測されないような場合である。そのような場合、前処理としての DTW によって、プラント故障の特徴は消去されてしまう。

そこで本研究では、入手可能な初期条件や環境条件といった外部変数を入力変数とし、DTW の時間伸縮プロファイルを出力変数として、何らかの経験的な入出力モデルを構築しておくことを提案する。

2.2 提案手法

従来法の手続きに加え、以下の監視を実施する。まず正常データについて、初期条件や環境条件（以下、外部変数と呼ぶ）と DTW による時間伸縮プロファイルの間に経験的な関係式を求めらる。

具体的な計算法は以下のとおりである。Fig. 1 のような $t \times r$ のグリッド上の軌跡 $c(k)$ が時間伸縮の結果を表すが、このま

までは扱わずにため、Kassidas ら⁽⁵⁾によって提案された非対称な波形同期法を参考に、以下のような時間伸縮に関する特徴量の生成を行なう。ある参照波形 \mathbf{B}_{ref} を決め、それを Fig. 1 の波形 \mathbf{R} とする。それに対して他の波形 \mathbf{B}_i を Fig. 1 の \mathbf{T} として DTW によって伸縮した結果、 \mathbf{B}_{ref} の1点目から r 点目までに対応する \mathbf{B}_i のインデックスがどうなったかを考える。Fig. 1 の軌跡 $c(k)$ 上の点を考えると、 $j(k)$ は r 通りの値をとり、それぞれについて対応する $i(k)$ があるが、同じ $j(k)$ の値が複数の $i(k)$ に対応づけられる場合がある。Fig. 1 でいえば、軌跡 $c(k)$ が水平方向にのびている部分である。

さて、 \mathbf{h} を r 個の成分を持つベクトルとする。その成分 $\mathbf{h}(\rho)$, ($\rho=1, \dots, r$) を r 通りの異なる $j(k)$ にそれぞれ対応する $i(k)$ の値であると定義し、これを時間伸縮プロファイルを表現するベクトルとして定義する。このとき、 \mathbf{h} の決定において注意が必要なのは前述の Fig. 1 で軌跡が水平方向にのびている部分である。たとえば $c(k)$ の一部を

$$[i(k), j(k)] \rightarrow [i(k+1), j(k+1)] \rightarrow \dots \rightarrow [i(k+a), j(k+a)] \quad (4)$$

と表したとき、

$$j(k) = j(k+1) = \dots = j(k+a) = \rho_o \dots\dots\dots(5)$$

すなわち

$$[i(k), \rho_o] \rightarrow [i(k+1), \rho_o] \rightarrow \dots \rightarrow [i(k+a), \rho_o] \dots\dots\dots(6)$$

のようになっていたとする。この場合、 $\mathbf{h}(\rho_o)$ は、

$$\mathbf{h}(\rho_o) = \frac{i(k) + i(k+1) + \dots + i(k+a)}{a+1} \dots\dots\dots(7)$$

のように定義する。 $a=0$ の場合は $\mathbf{h}(\rho_o) = \mathbf{h}(j(k)) = i(k)$ である。これによって、異なる $j(k)$ の個数 (r 個) の成分を有するベクトル \mathbf{h} が定義される。 \mathbf{h} の m 番目の成分が m より大きければ、 \mathbf{B}_{ref} の m 番目の点に対応づけられた \mathbf{B}_i の点は m 番目より後の点であることになり、 \mathbf{B}_i のほうがプロセスの進行速度が遅くなっていることを意味する。

以上で時間伸縮プロファイルを表現するベクトルが準備できた。つぎに、外部変数と時間伸縮プロファイルの間の経験的な関係式を求めらる。

外部変数（全部で p 個あるとする）をバッチ毎に並べたサイズ $I \times p$ のデータ行列 \mathbf{G} を訓練用入力データとして用意する。このとき、プロセスの進行速度に影響すると考えられる変数を行列 \mathbf{G} に含めることが重要である。それらのバ

ツチに対応する $\mathbf{h}(\rho)$ を並べた $I \times r$ の行列 $\mathbf{H}=[\mathbf{h}_1, \dots, \mathbf{h}_I]^T$ を用意し、訓練用出力データとする。関係式の求め方は自由であるが、ここでは一般的な非線形関数の近似法として、訓練バッチ数と等しい I 個の中間層を持った RBF ネットワークを使用した。モデルパラメタとして RBF の広がりがあり、これは故障検出性能に影響を与える。これを適切に定めるには、訓練サンプルの一部を除いて RBF ネットワークを訓練し、除いたサンプルに対する予測誤差を見るのが一つの方法である。与えた広がり小さすぎると、訓練に使用しなかったサンプルに対する予測誤差（下記の (8) 式で表される）が増大する一方で、広がり大きすぎると故障検出性能が低下するので、訓練に使用しなかったサンプルに対する予測誤差が増大しない程度に小さな値を採用すべきである。

\mathbf{h} の成分の個数 r は時間伸縮情報を表現するには充分すぎるほど大きい場合がある。Fig. 1 の $c(k)$ に相当する曲線をプロットして、これが十分になめらかであり、時間軸方向の情報が過剰であると考えられれば、計算量を節約するために、これを間引いても良い。このとき、 $c(k)$ の形状が表現できなくなるほどの間引きを行わないように注意する必要がある。

さて \mathbf{G} を入力、 \mathbf{H} を出力の訓練データとして訓練した RBF ネットワークに対して、診断対象の新しい p 個の外部変数を入力して、予想される時間伸縮プロファイル $\hat{\mathbf{h}}$ を出力し、実際の時間伸縮プロファイル \mathbf{h} との差を

$$\boldsymbol{\varepsilon} = \mathbf{h} - \hat{\mathbf{h}} \dots \dots \dots (8)$$

とする。さらに、 $\boldsymbol{\varepsilon}$ の各成分のプラントが正常な期間における標準偏差で規格化して次の (9) 式のような予測誤差ベクトルを考える。

$$\mathbf{e} = \begin{bmatrix} \frac{\varepsilon_1}{\sigma_{\varepsilon 1}} & \frac{\varepsilon_2}{\sigma_{\varepsilon 2}} & \dots & \frac{\varepsilon_r}{\sigma_{\varepsilon r}} \end{bmatrix} \dots \dots \dots (9)$$

ただし、 $\sigma_{\varepsilon m}$ は \mathbf{h} の m 番目の成分のプラントが正常な期間における標準偏差である。異常監視指標として \mathbf{e} の大きさ、具体的には \mathbf{e} の成分の二乗和 $|\mathbf{e}|^2$ を採用することを提案する。従来法である MPCA で用いられる Q 統計量は二乗予測誤差と呼ばれることもあり、これと明確に区別するために、本研究ではこの式(9)で定義される予測誤差 \mathbf{e} を WPE (時間伸縮プロファイル誤差, Warping Profile Error), $|\mathbf{e}|^2$ を SWPE (二乗時間伸縮プロファイル誤差, Squared Warping Profile Error) と呼ぶことにする。また、 \mathbf{e} の成分、すなわち時間伸縮の正常状態との差異が初期から終期に向けてどの方向にどれだ

け観測されたかは、故障の種類を診断するために有益となる可能性がある。提案する手法の概念図を、従来法の場合に加筆する形で Fig. 2 に示す。

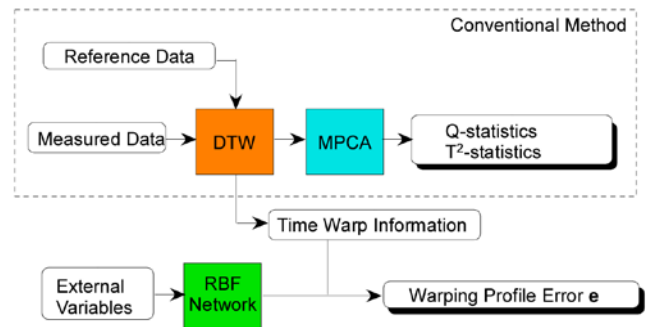


Fig. 2 Proposed method

3. 数値例

3.1 シミュレーションプログラム"PenSim"

PenSim は Birol ら⁽⁶⁾によって開発されたペニシリン培養セミバッチプロセスのシミュレーションプログラムである。異なる初期条件や制御条件のもとでプロセスに関わる変数の時間経過をシミュレートすることができる。プロセスは培養器にバイオマスおよび基質を入れて反応をスタートさせ、基質が消費されてくると途中から追加供給される。この間、温度と pH が制御される。プログラムの出力する変数のなかで、Table 1 に掲げる 9 変数を解析に用いた。

本研究は、「異常が変数の軌跡の時間伸縮としてのみ現れる場合」について議論する目的で、解析する変数を選択している。したがって、実際のプラントで計測変数としてこれらを選択することが合理的かどうかなどについては議論しない。

Table 1 Variables used in monitoring of the process

Aeration rate (l/h)
Agitator power (W)
Substrate feed temperature (K)
Dissolved oxygen concentration (g/l)
Carbon dioxide concentration (g/l)
pH *
Fermentor temperature (K)
Biomass concentration (g/l) *
Penicillin concentration (g/l) *

DTW の距離 D を算出する基準としては、プロセスの進展度合いをよく表すと考えられる*印のついた3変数のみを採用

することとし、(2) 式の \mathbf{W} はこれら以外の変数を無視するように設定した。

3.2 計算条件

シミュレーションを行なうプロセスの時間は各バッチ 400 時間とし、30 分ごとにデータを 800 点記録した。温度制御は PID とし、制御パラメタはプログラムの初期値通りである。シミュレーションの初期条件（本ケースではこれを外部変数として使用する）を変化させながら正常バッチを 85 回、異常バッチを **Table 2** に示す Fault A~C それぞれについて 5 回ずつ計算した。初期条件は、**Table 3** に示す変数に関してそれぞれ表に示した平均値と範囲にしたがって一様分布するように乱数を発生させて与えた。また、計測誤差を模擬するために各変数にそれぞれの最大値のおよそ 0.5% に相当する標準誤差で正規分布する乱数を加えた。

Table 2 Explanations of faults

Symbol	Explanations of faults
Fault A	Step change of substrate feed rate. (+2% after 10 h.) (One of the disturbances prepared in PenSim.)
Fault B	Degradation of agitator. (True power: 30W → 29.5W)
Fault C	Offset of fermentor temperature sensor. (-0.5K)

Table 3 Variations of initial conditions

Variable	Mean	Range
Substrate conc. (g/l)	15.25	±0.25
Biomass conc. (g/l)	0.1025	±0.0025
Culture volume (l)	152.5	±2.5
Temperature (K)	298.5	±0.5

DTW の参照プロファイルとして正常バッチのなかから任意に選んだ一つのバッチのデータを使用した。MPCA や提案手法の RBF ネットワークの訓練には正常バッチのなかから 50 セットを使用し、手法の評価には残りの 35 セットの正常バッチと 15 セットの異常バッチのデータを用いた。時間伸縮プロファイル \mathbf{h} および WPE は、参照プロファイルを基準に 5 時間に 1 点の間隔で 80 点抽出したタイミングについて算出した。以上の計算手順を **Fig. 3** に示す。

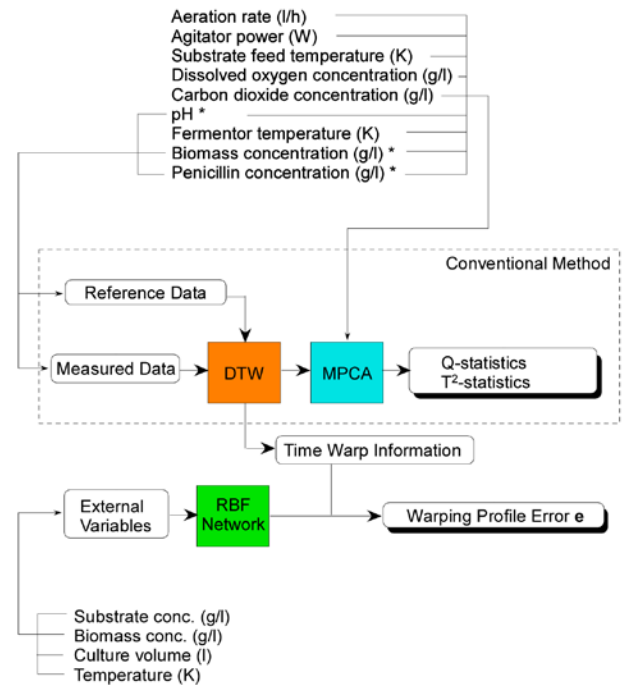
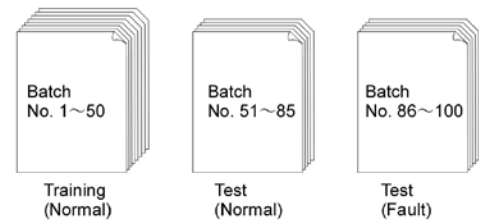


Fig. 3 Schematic of calculations.

3.3 結果

Fig. 4 に、正常状態のデータ（無作為に抽出した 5 例）と異常 C（温度設定値のオフセット）のデータ 5 例を重ねてプロットしたものを示す。ここに示さない 4 個の変数はいずれもほぼ一定値の周りにランダムに分布していた。上段が DTW 処理前、下段が処理後である。この時間スケールでは正常と異常のデータはほぼ重なり、これらを見ただけでは正常と異常の違いは不明瞭である。また、上段と下段の差異も明らかでなく、DTW の効果も見出すことができない。**Fig. 5** に各変数の変化が激しい時間帯（30~50 hour）の拡大図を示す。上段の生データを見ると異常状態の波形のほう時間が速いという特徴を有するよう見えるが、**Fig. 5** の下段からわかるように、DTW によってその特徴は消え、すべての波形が重なっている。**Fig. 5** を見る限り、本ケース（異常 C）において、計測されている変数については、正常データと異常データの間プロセス進行速度以外の差異は（少なくとも目で見て判断できるレベルでは）存在しないようである。

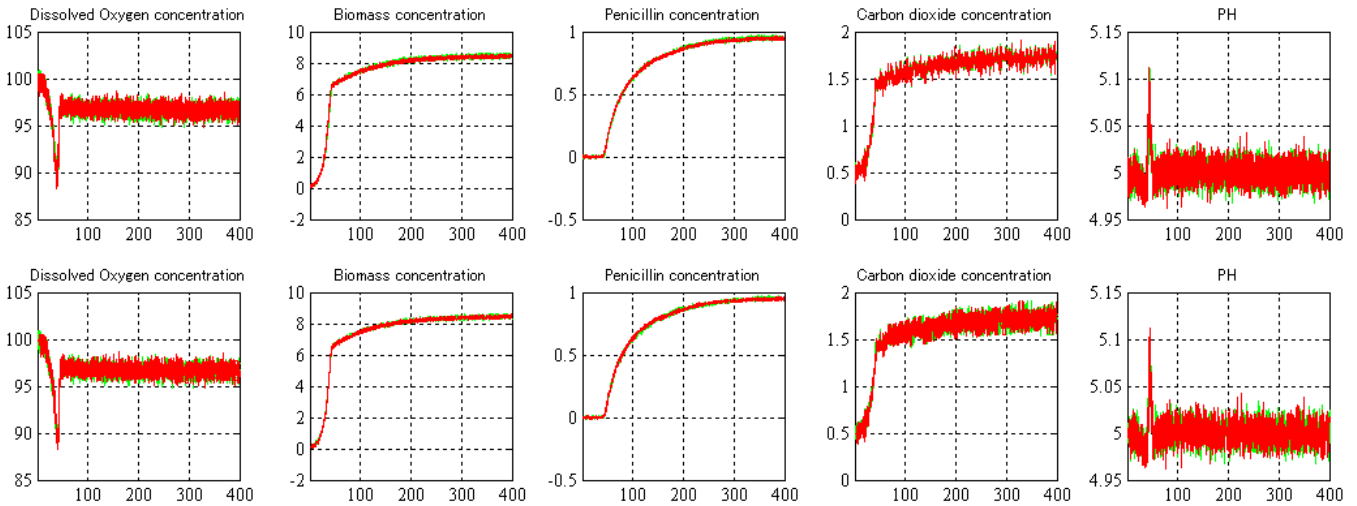


Fig. 4 Example data from normal (green) and fault C (red) batches. Randomly chosen five batches for each are plotted. Upper and lower plots are before and after DTW, respectively.

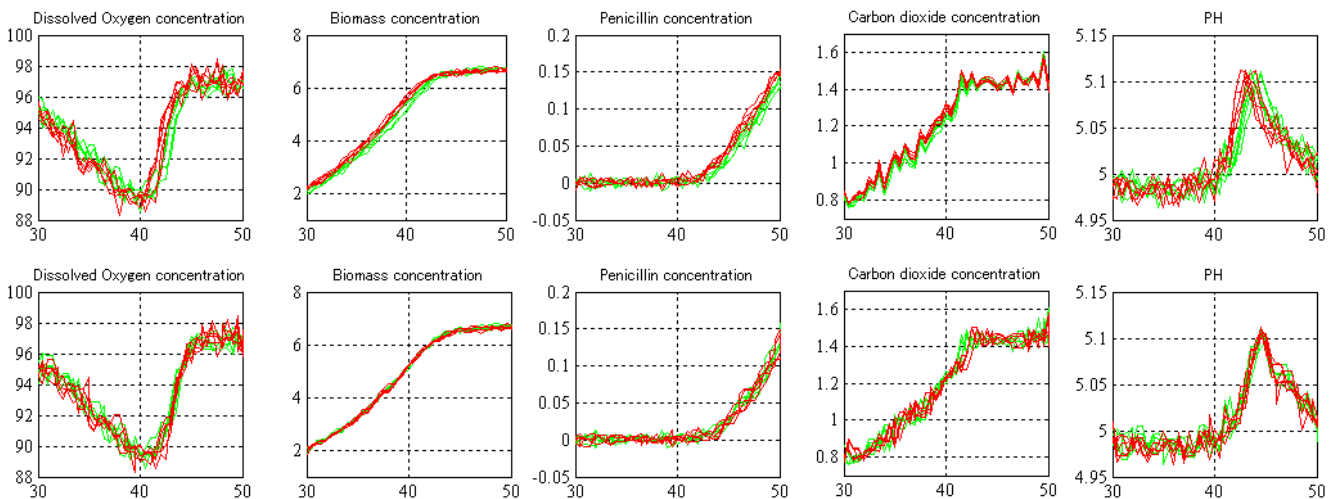


Fig. 5 Close-up of Fig. 4.

つぎに DTW によって波形の伸縮を行った波形を対象に、従来法である MPCA による異常検出を試みた。Fig. 6 に、各バッチに対する Q 統計量 (PCA で算出される異常度を表す指標) を示す。バッチ No.50~85 が正常、86~100 が異常である。Fault A は異常である可能性があるが、はっきりとした確信を持って異常と判断することは難しそうである。一方、Fault B,C については正常バッチとの差は不明瞭であり、これらについては従来法では正常と判定される可能性が高い。

さて、つぎに提案手法を評価する。DTW によってキャンセルされたプロセスの進行速度の差異が、各バッチの外部変数によって説明可能な「問題のない」挙動なのか、故障の発生によるものなのかを調べるために、時間伸縮プロファイルの $SWPE|e|^2$ を比較する。Fig. 7 に結果を示す。一部の異常バッチは異常であると判断しづらいもの

もあるが、Fig. 6 の従来法の示す結果と比較して、より多くのバッチについて正しく診断することが可能であるといえる。

ここで、提案手法の効果について詳細に見ていく。Fig. 8 は、正常バッチ (No. 81~85, 緑で表示) と異常バッチ (Fault C の No. 96~100, 赤で表示) について、DTW による時間伸縮軌跡と、それぞれのバッチの外部変数の値を RBF ネットワークに入力することによって予測された軌跡、それらの差を各時間ごとに正常時の標準偏差で規格化した WPE を示したものである。最上段のグラフは算出された時間伸縮軌跡 (Fig. 1 に相当するもの) であるが、正常バッチの緑色と異常バッチの赤色について、互いに重なり合っており、これから正常と異常を見分けることはできない。すなわち、時間伸縮の程度の大小からは正常と異常が判別できないことがわかる。一方で 2 段目の

グラフは外部変数の値から予測された時間伸縮軌跡である。これを見ると、異常バッチの後半は正常バッチと比較して一様に軌跡が下側にシフトしている。すなわち、バッチの後半においては、異常バッチについてはいずれも進行が遅めと予測されていることがわかる。3段目のWPEを見るとそのことがよりはっきりと示されている。時刻100時間以降、異常バッチは進行が遅く、WPEが正の方向に現れている。一方で、時刻30~50時間においては予測残差が逆符号となっており、この間は異常バッチの進み方が予測されるよりも早いことがわかる。このことはFig. 5において目視で確認された事実とよく対応している。

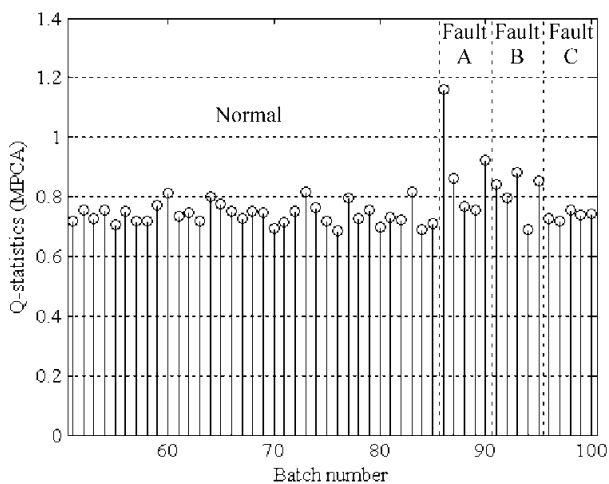


Fig. 6 Q-statistics of MPCA (conventional method).

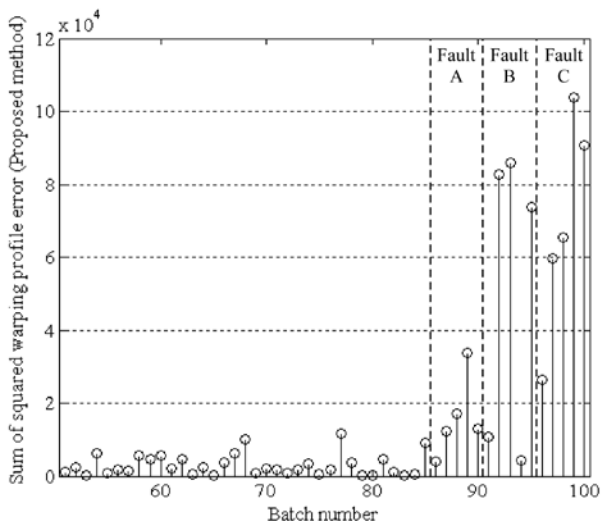


Fig. 7 Squared warping profile error $|e^2|$ of the proposed method.

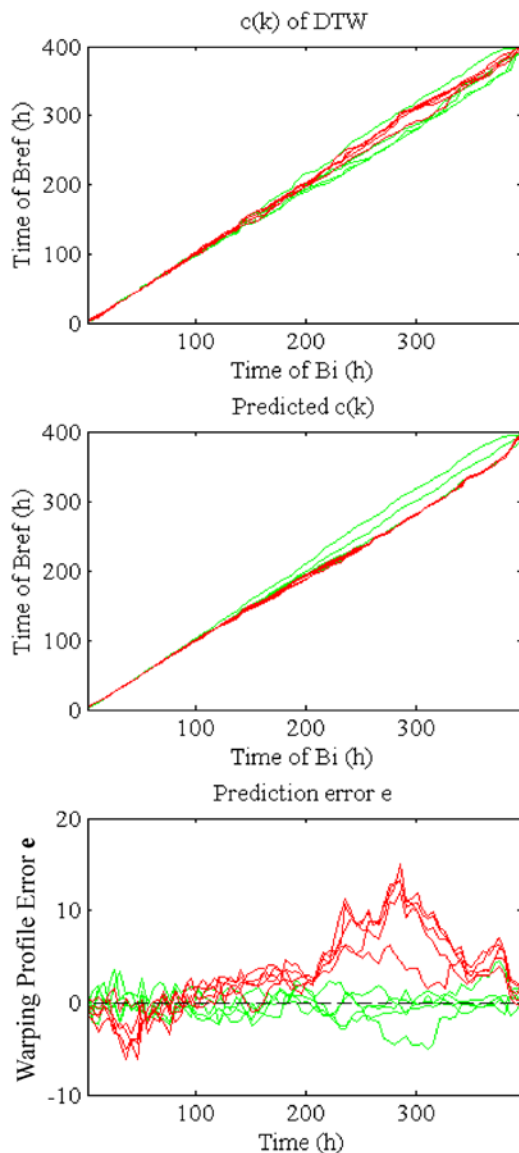


Fig.8 Calculated and predicted $c(k)$ plotted with warping profile error vector e . Normal (green) and fault C (red) batches are plotted.

以上、Fig. 8 に一例を示したように、外部変数を入力として時間伸縮軌跡を予測するモデルは、異常を診断する上で有効なツールである。

上に見た定性的な結果をさらに定量的に評価するために、Fig. 6 と Fig. 7 の指標それぞれについて正常/異常の判定閾値を変化させながら誤判定されるサンプルの数をプロットした。結果を Fig. 9, 10 に示す。図にプロットされているのは、異常を検出できなかった数(Missed detection: MD)と正常でありながら異常と誤って見なした数(False alarm: FA), およびそれらの総数であり、これらの指標が低いことが重要である。

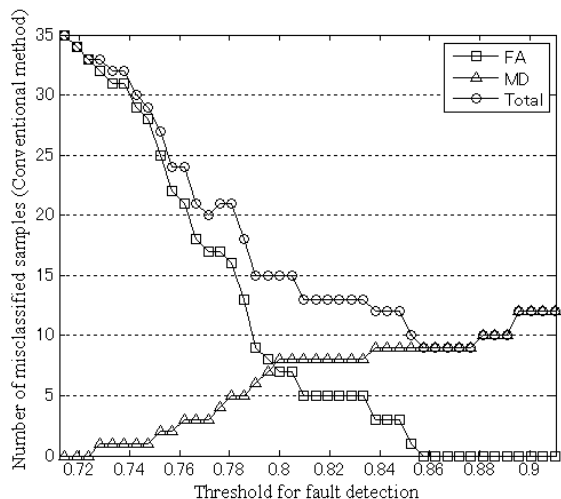


Fig. 9 Number of misclassified batches by conventional method (plotted against discriminant threshold). Δ : Missed detection: MD, \square False alarm: FA, \circ Sum of MD and FA.

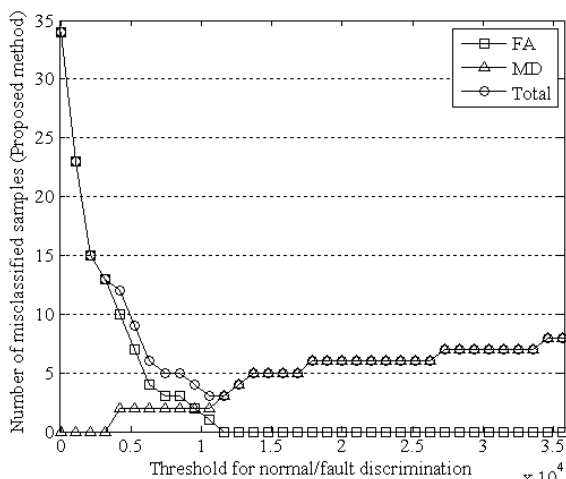


Fig. 10 Number of misclassified batches by proposed method (plotted against discriminant threshold). Δ : Missed detection: MD, \square : False alarm: FA, \circ Sum of MD and FA.

さて、Fig. 9は従来法であるMPCAのQ統計量についての結果であり、Fig. 10は提案手法のなかのSWPEを用いた結果である。ここで注意すべきことは、Fig. 2に示したように、提案手法はQ統計量とSWPEの両者を使うという点である。したがって、従来法はFig. 9によって性能を判断できるが、提案手法はFig. 9とFig. 10の両者によって評価する必要がある。さて、実際的な場合において故障検出閾値がどのように設定されるべきかを考えると、FAが容認できる程度に小さな値となる条件のなかで、見逃しが極力少なくなるように管理限界を決めるのが通常の見え方であろう。たとえばFig. 9ではFAが0個となる

0.86を少し上回る程度、Fig. 10では同様にFAが0個となる 1.1×10^4 を少し上回る程度に設定することが合理的である。このとき、従来法ではFig. 9より15個の異常サンプルの内9個以上を見逃すのに対し、提案手法ではQ統計量では同じくFig. 9より9個以上の見逃しとなるが、同時に採用しているSWPEによって、Fig. 10より4~5個程度の見逃しですむことになる。以上より、提案手法のほうが優れているといえる。

さて、異常の種類を識別できないかとの観点から、WPEの成分を評価する。WPEの成分は、プロセスの進行が、外部変数を考慮した結果予測されるものと比較して遅いか早いかを示すものである。Fig. 11に、無作為に5バッチずつ3組抽出した正常バッチのWPEと、Fault A~C各5バッチのWPEを示す。

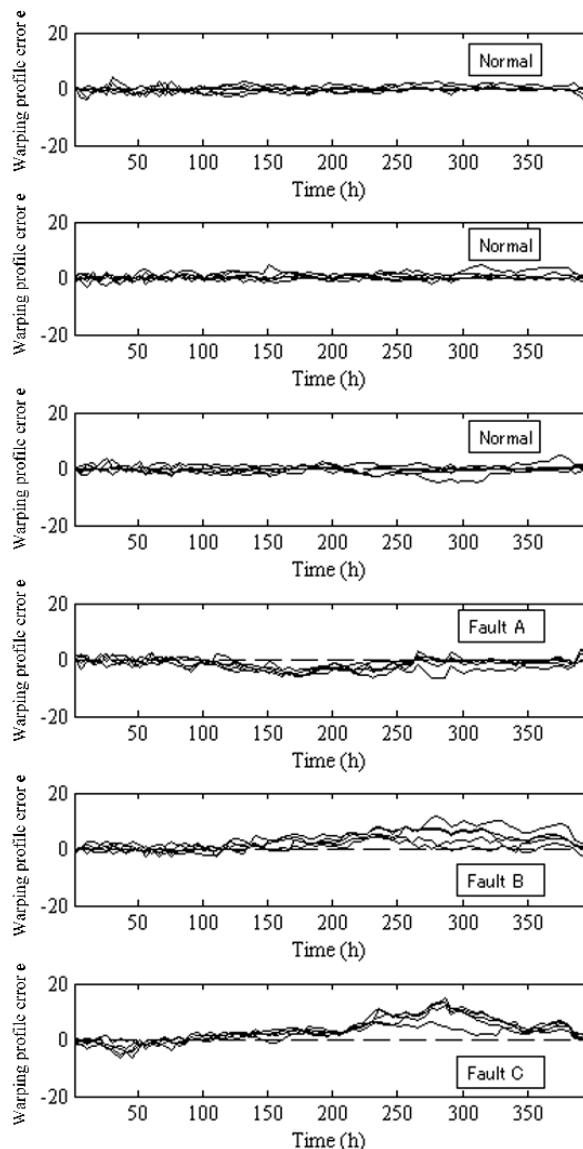


Fig. 11 Warping profile errors. 5 batches are plotted for each graph.

Fig. 11 の各グラフは、いずれも左から右に、バッチの初期から終期に向かって 5 時間おきに評価したプロセス進行予測残差を意味する。正の値は、診断対象波形の $i(k)$ とそれに対応する参照波形の $j(k)$ を比較したときに $j(k) < i(k)$ となっていることを意味するから、プロセス進行の遅れである。異常バッチにおける WPE の形状には特徴があり、一部例外はあるものの Fault A は 150~200 h 付近で進行が速く、Fault B,C はいずれも 200~350 h 付近で進行が遅い。それに加えて Fault C は 50 h の少し前に進行が速いことがわかる。これらの特徴によって、異常の種類を経験的に識別できる可能性があることがわかる。

4. まとめ

バッチプロセスの監視法として従来提案されている MPCA とその前処理としての DTW の使用において、DTW の処理における時間伸縮プロファイルとプラントの初期条件および環境条件との関係を利用した新しい監視手法を提案した。この手法は従来法に追加してプラント初期条件や環境条件といった外部変数を入力、DTW の時間伸縮プロファイルを出力とするニューラルネットワークを備えることによって構成され、時間伸縮プロファイルの予測残差を異常監視指標とする。特にプロセス進行速度が外部変数と相関を有している場合に、異常によってデータに現れる現象がプロセス進行速度の変化のみであるものを検出するのに有効な指標である。時間伸縮プロファイルの予測残差は異常種類の識別に利用できる可能性が示された。

[参考文献]

- (1) Gollmar, K., and C. Posten : Supervision of bioprocesses using a dynamic time warping algorithm, *Control Eng. Practice*, **4**, 1287/1295 (1996)
- (2) Kourti, T. : Multivariate dynamic data modeling for analysis and statistical process control of batch processes, start-ups and grade transitions, *J. Chemometrics*, **17**, 93/109 (2003)
- (3) Kourti, T. : Process Analysis and Abnormal Situation Detection: From the Theory to Practice. *IEEE Control Systems Magazine*, **22**, 10/25 (2002)
- (4) 加納学 : プロセスケモメトリクスによる統計的プロセス管理, システム/情報/制御, **48**, 165/170 (2004)
- (5) Kassidas A, J. F. MacGregor, and P. A. Taylor : Synchronization of batch trajectories using dynamic time warping, *AIChE J.*, **44**, 864/875 (1998)

- (6) Birol, G, C. Undey, and A. Cinar : A modular simulation package for fed-batch fermentation: penicillin production, *Comput. Chem. Engng.*, **26**, 1553/1565 (2002)

[著者紹介]

田村 雅之 (正会員)



1994年東京大学大学院工学系研究科物理工学専攻修士課程修了。同年東京ガス(株)入社、現在に至る。2002年博士(工学)。光計測技術の研究、設備診断手法の研究に従事。