2J-02

FAにおけるエッジアプリケーションのミドルウェア化に向けた検討

境　将輝†　　中島　毅†
芝浦工業大学†

高橋　和也‡
Edgecross コンソーシアム‡

1. 研究の背景

FA(Factory Automation)のIoT化が急速に進み,製造装置に近いエッジ領域においてリアルタイム性の高い分散処理を行うエッジアプリケーション技術が注目されている.従来,生産ラインから取得したデータはオフライン上で分析され,その分析結果を生産ラインの設計・オペレーションに反映させることで,生産性向上のために利用されてきた.この改善サイクルの効率を上げるために,オフライン分析の結果を直接エッジアプリケーション(以下EAP)に反映することで,リアルタイムに機器を監視・制御することが求められている.

しかし,オフライン分析には多くの手法があり,またリアルタイムに製造現場のデータを取得する方法がラインを構成する製造装置ごとに異なるため,それらに対応するアプリケーションの開発が困難となっている.そのためFAにおいて,それらの差異によって発生する作業負担を軽減させるようなEAPの開発のための支援が必要である[1].

2. 従来研究

従来技術としてリアルタイムデータアナライザというEAPが存在する[2].このEAPはデータ収集プラットフォームのEdgecross[3]からデータを取得し,直接オフラインの分析結果を反映することで機器の監視・制御を行うことができる.しかしこのEAPには以下の2つの制約事項がある.

- 分析手法が重回帰分析とMT法のみ利用可能.
- 対応する前処理として標準的な統計量のみ提供.

このEAPは,Edgecrossを用いることでデータ収集の汎用性を確保しているが,特定の分析方法に特化しているため,利用可能な分析手法やツールが限定されることが課題である.

3. 研究目的

本研究の目的は,上記の課題を解決するために,様々なEAPの開発に対応可能なミドルウェアを開発することである.この目的を達成するために実EAPを試作し,EAPのミドルウェア化に必要な要求事項を抽出する.

4. ミドルウェア化構想

提案するミドルウェアは,種々の分析手法や前処理に対応するために,オフライン分析の結果や前処理記述を用いて,柔軟にEAPを構築できるようにすることを目指す.以下はEAPのミドルウェア化に対する要求事項である.

- 製造装置から生データを取得し,監視・制御を行う.
- 分析結果をデプロイしてEAPのふるまいを実装する.
- 分析手法や分析環境(R, Pythonなど)に依存しない.

(1),(2)は本ミドルウェアを用いたEAPの仕様である.図1に提案するミドルウェア化の構想を示す.

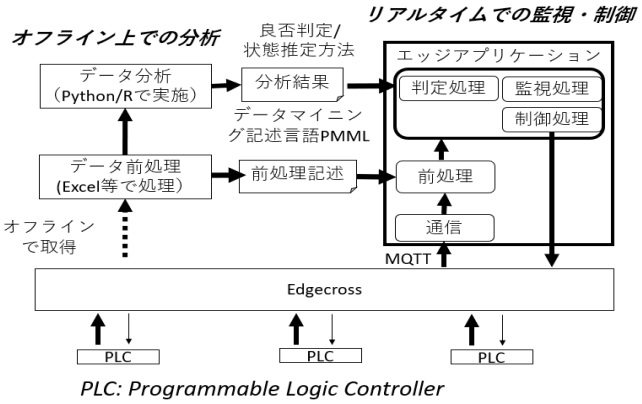


図1 FAにおけるEAPのミドルウェア化構想図

(1) オフライン上での分析 (分析者の作業)

- 製造装置からEdgecrossを通して生データをオフライン上で取得する.
- 取得したデータに対して前処理を実行する.
- 実行した前処理の処理内容を前処理の記述言語により記述する(処理内容を記述言語化することで種々の前処理に対応できる柔軟性を持つものを目指す).
- 分析結果をPMML(Predictive Model Markup Language)形式で出力する.(PMMLはデータマイニングモデルをXMLベースで記述するための言語であり,PMMLを用いることで環境に依存せずモデルを使用することができる[4].)

(2) リアルタイムでの監視・制御 (EAPのふるまい)

- Edgecrossからデータをリアルタイム上で取得する.
- 取得したデータに対して,前処理記述を読み込み,オフライン上で行った処理と同じ前処理を実行する.
- オフライン上で作成したPMMLモデルを用いてデータに対して推論を実行する.推論には製品の良否判定や機器の監視・制御などを想定している.

EAPミドルウェアを利用することにより,種々の分析手法や前処理への対応が可能になり,柔軟にEAPを構築できるようになる.

5. EAPの試作

ミドルウェア化の検討のために実EAPの試作を行った.今回試作したEAPは,実機から得られる実データと,実データを元にして作成された推論モデルを用いて,工場で製造している製品の良否判定を行う.

(1) 試作EAPのシステム構成

図2は試作したEAPのシステム構成図である.図2において太枠で覆われた部分が今回開発した試作EAPの構成である.各部の機能を以下に示す.

①データ受取部: EdgecrossからMQTT通信で複数のセンサデータを取得する.Publish-Subscribe方式のSubscriberの役割を持つ.1ms毎にデータを受け取り,合計で6000ms分のデータを取得する.

②前処理部: データ受取部が受け取ったデータに対して前処理を実行

「Examination for middleware of edge application in FA」
† 「Masaki Sakai・Shibaura Institute of Technology」
† 「Tsuyoshi Nakajima・Shibaura Institute of Technology」
‡ 「Kazuya Takahashi・Edgecross consortium」

Copyright ©2020 Information Processing Society of Japan.
All Rights Reserved.

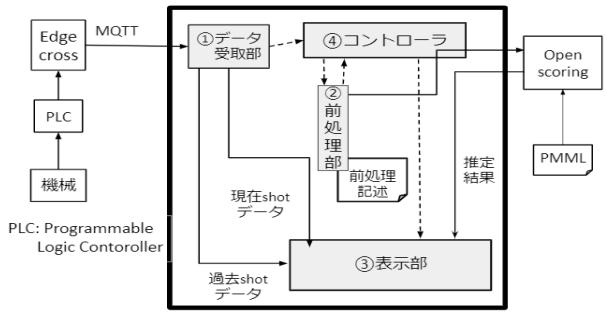


図2 試作EAPのシステム構成図

する.100ms毎に要約統計量を求める.説明変数1つ毎にセンサデータの種類,取得するデータの区間幅,実行する統計処理を逐次選択する.

③ 表示部: 3種類のデータの可視化を行う.過去に取得したセンサデータ,リアルタイムで取得しているセンサデータ,判定結果の分布を表示する.

④ コントローラ: 前処理部が作成した説明変数を PMML 実行エンジン(Open scoring[5])に送り,返ってきた判定結果を受け取る.

⑵ 前処理記述の考案

図1に示したように,EAPは前処理記述を読み込んでオフライン分析の前処理を実行する.前処理記述を用いることには以下の狙いがある.

① 前処理の意味の明示化

処理に必要な情報をタグで表現することによって,どんな処理を行って作成される変数なのかを明示化する.

② 前処理の実行の簡易化

前処理記述を作成すれば,前処理を実行できるので,EAP 開発時における負担を軽減する.

③ 前処理の変更・追加の簡易化

変数タグの中の値を変更することで,前処理の実行内容を簡単に変更することができる.また特定の変数にだけ処理を追加する場合も新たにタグを追加すれば実装できる.

以下は,考案した前処理記述の例である.

```
<variable name="チップ ストローク量 2500_min">
    <Sensor>チップ ストローク量</Sensor>
    <Range>2500</Range>
    <Statistics>min</Statistics>
</variable>
```

XML形式で,説明変数1つにつき必要な処理内容をタグで囲む構造となっている.

- variable タグ: 1つの説明変数に対して内部情報を保持(name 属性は説明変数の名前).
- Sensor タグ: 取得するセンサデータの種類を選択.
- Range タグ: 所得するセンサデータのデータ幅を選択.
- Statistics タグ: 取得したセンサデータに対して実行する統計処理を選択.

6. 試作EAPの評価と考察

試作した EAP に対してリアルタイム性に関する評価を行った.これはラインで製造される製品の判定を行う上で重要な指標である.実際の製造時間とリアルタイム処理の実行時間との比較を行った.Edgecross が6000ms分のデータを Pubilish し終えてから,EAP が6000ms分のデータを Subscribe して受け取り終えるまでの時間(1),受け取ったデータに対して前処理を実行する時間(2),判定結果を Open scoring から取得するまでの時間(3)を計測しそれらの時間を合わせた処理全体の時間(4)計測を100回行った.(CPU:Intel i7-7500U,メモリ:16GBの実行環境下で計測を行った.)

表1. 製品の製造時間とリアルタイム処理実行時間の比較

	平均	標準偏差
製造時間	300秒	
(1) データ取得	1.07秒	0.70
(2) 前処理	0.35秒	0.29
(3) 判定結果取得	0.03秒	0.28
(4) 処理時間全体	1.45秒	0.72

- 実際の製造時間300秒に対して,処理時間全体は1.45秒である.次の製品が製造されるまでに処理を終えることから,リアルタイム性は確保できていることがわかる.今後,処理時間に関する課題が生じたときデータ取得部分で時間を短縮させることは難しいため,前処理部分の処理時間を短縮させることが重要である.
- 今回の前処理では複雑な処理をしておらず,また欠損値補完を行っていない.今後は様々な分析に対応できるようにリアルタイム性を意識していく必要がある.

7. EAP試作・評価を通したミドルウェア化への課題発見

アプリケーション開発とその評価,前処理記述考案を通じて得たミドルウェア化のための課題を以下に示す.

⑴ 前処理の網羅性の確保と検証

今回試作では,要約統計量を取得する処理しか実装していない.FA現場における分析の多様性に対応した前処理モジュールの充実が必要である.例えば,種々の欠損値補完手法(統計的な補完や線形補完)などに対応する必要がある.

⑵ 種々のシナリオでのリアルタイム性の確保

製造されている製品に対してリアルタイムで判定結果を返す場合次の製品ができるまでに結果を返さなければならない.MQTT通信でデータ取得,前処理を実行,判定結果出力にかかる時間をできるだけ短縮し,種々の利用シナリオでの適合性を検証する必要がある.

8. まとめと今後の課題

FA分野におけるEAPの構築を容易にするためのEAPのミドルウェア化の構想を示し,EAP試作を実施することでEAPのミドルウェア化への課題を明確にした.その結果,前処理の網羅性と処理のリアルタイム性の確保が重要であることがわかった.上記の課題は,EAPのミドルウェア化に向けて解決すべきものであると考えられる.

参考文献

[1] 製造業におけるIoTの活用例と将来像:e-F@ctoryを例として(特集 IoTがもたらす製造業の新展開) オペーションズ・リサーチ= Communications of the Operations Research Society of Japan : 経営の科学,2018(柳生 理子 高橋 雄一 大谷 治之) 著

[2] 三菱電機「リアルタイムデータアナライザ」 https://www.edgecross.org/ja/data-download/pdf/Partner_products_5.pdf

[3] Edgecrossコンソーシアム https://www.edgecross.org/ja/

[4] Alex Guazzelli, Wen-Ching Lin, Tridivesh Jena: Pmml in Action: Unleashing the Power of Open Standards for Data Mining and Predictive Analytics, CreateSpace,2012

[5] Openscoring https://openscoring.io/

Copyright ©2020 Information Processing Society of Japan.
All Rights Reserved.