5R-01

エージェントに基づくネットワークデータ分析支援基盤

山田 良介 † 谷村 優介 † 笹井 一人 ‡ 髙橋 秋典 § 北形 元 ‡ 五十嵐 隆治 § 木下 哲男 ‡

† 東北大学大学院情報科学研究科 ‡ 東北大学電気通信研究所 § 秋田大学大学院工学資源学研究科

1 はじめに

近年のネットワーク管理においては，ネットワークシステムを構成する様々な機器の稼働状況を把握するのみならず，ユーザの行動や外部ネットワークからの通信といった高次の要素を把握する必要性が指摘されている．昨今のコンピュータの計算能力やネットワーク帯域の増大によって，これまで常時利用されることのなかった詳細情報を常に収集分析し，高度なネットワーク管理を実現することが可能になりつつある [1]．その一方で，増大するデータ量，監視対象，分析手法などに対処することは，元来ネットワーク技術者である管理者にとって容易ではなく，新たな負担を生んでいる．そこで，本研究では，ネットワークシステムの高度な管理を容易に実現することを目的とした新しいネットワークデータ分析基盤を提案する．

既存のネットワーク管理支援基盤として Zabbix[2] などのネットワーク監視ソフトウェアが存在しているが，これらの持つ機能はデータの収集及び可視化と簡易なアラート機能に限定されており，高度な分析への対応が難しい．一方で時系列トラフィックデータから特徴量を計算し周期的攻撃を検知する研究 [3] など，統計手法を用いた異常状態検出の研究が行われているが，これら手法の利活用には使用者に統計に関する専門知識を要する．

他方でデータ解析の分野では，Apache Storm[4] など，Complex Event Processing(CEP) に基づくデータのストリーム処理が注目されている．これらはオンメモリで処理を行うことで低遅延を実現しており，リアルタイムでの対応が求められるネットワーク管理において有用である．また，処理を細かなイベント処理に分割することで，分散環境上での実行に対応可能である．しかしこれら CEP システムは分析の前処理を含めた処理フローをプログラミングせねばならず，システム設計者への負担が大きいほか，アプリケーションの追加や重複する処理の共通化などを行う際の負担が大きく，データ解析の非専門家が携わるネットワーク管理に直接適用することは困難である．本稿では，エージェント技術に基づいたデータ処理モジュールを設計し，処理フローの柔軟かつ容易な生成・変更を可能とする，CEP ライクなネットワークデータ分析基盤を提案する．

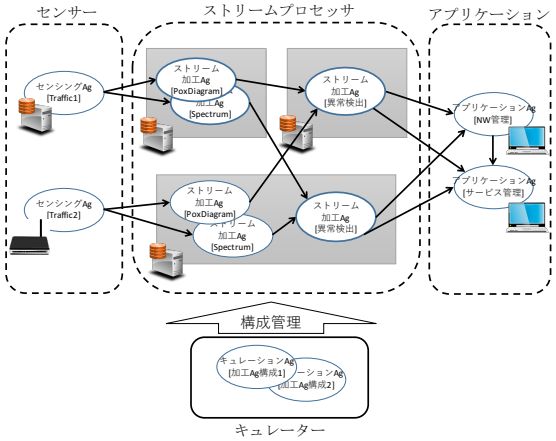

図 1: エージェントに基づくネットワークデータ分析基盤の概要

2 エージェントに基づくネットワークデータ分析基盤の提案

本稿で提案する分析基盤の概要を図 1 に示す．本基盤ではネットワークデータの分析にかかわる作業をキュレーション（データを収集・分析し意味のある結果を生むための，処理フローの設計を意味する），センシング，データ加工，アプリケーション (応用) の 4 つのステージに分類し，それぞれをエージェント (Ag) として設計することで，エージェントの持つ自律性，社会性に立脚した，柔軟な処理フローの生成，実行を実現する．分析処理はセンシング Ag から継続的に送信されるデータを加工 Ag が逐次加工，送信するストリーム処理として実現される．処理フローはアプリケーション Ag からの要求に基づき，キュレーション Ag の持つ構成知識に従って分散加工ノード上に加工 Ag が自動配置されることにより生成されるため，ネットワーク管理者の要求に従って容易にフローの生成が可能となる．

Agent-based analysis support infrastructure for network data
†Ryosuke YAMADA †Yusuke TANIMURA ‡Kazuto SASAI §Akinori TAKAHASHI ‡Gen KITAGATA §Ryuji IGARASHI ‡Tetsuo KINOSHITA
†Graduate School of Information Sciences, Tohoku University
‡Research Institute of Electrical Communication, Tohoku University
§Administration Department Faculty of Engineering and Resource Science, Akita University

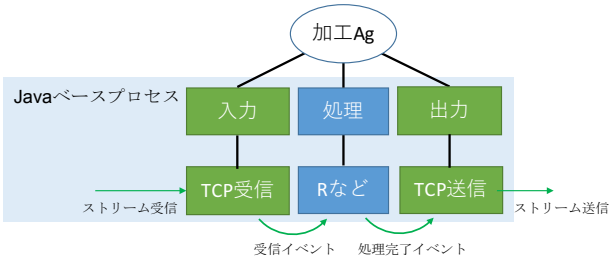

図 2: 加工 Ag のストリーム処理設計

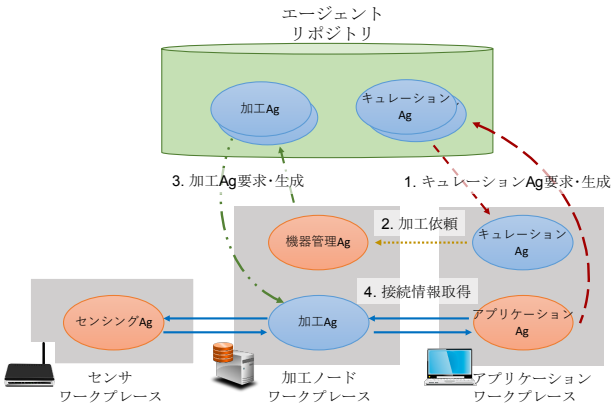

図 3: エージェント間の協調連携

3 試作システムの設計

エージェント開発基盤 IDEA[5] を用いて提案システムの試作を行った.

3.1 ストリーム処理の設計

加工 Ag におけるストリーム処理の設計を図 2 に示す. 加工 Ag は Java ベースプロセスとして入力, 処理, 出力の 3 モジュールを持つ. 入力, 出力モジュールはストリームデータの送受信を行うモジュールであり, ZeroMQ[6] を用いた Publish/Subscribe モデルによる TCP ソケット通信として実現される. 入力モジュールは他 Ag の出力モジュールが公開するソケットに接続することによりデータ受信を行う. 処理モジュールは受信データをもとにデータ加工を行うモジュールであり, Java プログラム, または R など他の処理系に処理を委譲し, 結果を受け取るプログラムとして実現される. 同一 Ag 内での各モジュールはイベント処理により接続される. センシング Ag, アプリケーション Ag も同様の入出力モジュールを持つことでストリーム処理が行われる.

3.2 エージェント構成の設計

加工 Ag がエージェントの連携により分散環境上に配置される様子を図 3 に示す. 各機器はそれぞれにエージェントの活動場所 (ワークプレース) を持ち, センシングを行う機器にセンシング Ag, 処理されたデータを受け取る機器にアプリケーション Ag, 加工ノードに機器管理 Ag が予め配置される. 機器管理 Ag は OS の種類, 言語処理系の有無などの知識を持ち, 加工 Ag が動作可能かの判断を行う. キュレーション Ag 及び加工 Ag はエージェントの格納場所 (エージェントリポジトリ) から加工ノードに配置される.

加工 Ag が配置される手順は以下の通りである.

1. アプリケーション Ag がリポジトリからキュレーション Ag を生成する
2. 生成されたキュレーション Ag の持つ構成知識に従い, 加工ノードの機器管理 Ag に対して加工依頼を行う
3. 機器管理 Ag は加工動作が可能かどうか判断し, 可能なら依頼を受諾し加工 Ag を生成する
4. 生成された加工 Ag はストリーム受信元となる Ag に接続情報 (ポート番号) を問い合わせ, 動作を開始する

4 おわりに

本稿ではエージェントを用いた構成の容易なネットワーク分析基盤を提案し, その試作システムの実装を行った. 今後は試作システムの定量的評価を行うとともに, 多種の分析手法に適用することで有効性を確認し, 拡張を行っていきたい.

謝辞

本研究の一部は, 総合科学技術・イノベーション会議の SIP(戦略的イノベーション創造プログラム)「レジリエントな防災・減災機能の強化」(管理法人:JST) および東北大学電気通信研究所における共同プロジェクト研究によって実施されました.

参考文献

[1] Kalegele, K. et al. ”Four Decades of Data Mining in Network and Systems Management”. *IEEE Trans. Knowl. Data Eng.*, 27(10), pp. 2700–2716, 2015.

[2] Zabbix. http://www.zabbix.com/.

[3] 髙橋秋典, 五十嵐隆治, 上田浩, 岩谷幸雄, 木下哲男. R/S Pox レッグライン特性. 情報科学技術フォーラム講演論文集, Vol. 11, No. 4, pp. 9–16, 2012.

[4] Toshniwal, A. et al. ”Storm@twitter”. In *Proc. SIGMOD 2014*, pp. 147–156, 2014.

[5] IDEA. http://www.k.riec.tohoku.ac.jp/s/idea/index.html.

[6] ZeroMQ. http://zeromq.org/.