L-024

有線インターネット VPN 回線の性能評価と閾値処理による障害判断手法
Evaluation for VPN over Ethernet Internet and Congestion Detection Using Threshold Processing

柏岡 秀哉† 平分 亮† 高平 寛之† 河合 英宏†
Shuya Kashioka Ryo Hirana Hiroyuki Takahira Hidehiro Kawai

1. はじめに

情報通信技術の発展により，従来オンプレミス環境を前提として設計されてきた制御システムをクラウド環境へ移行する需要が高まっており，クラウドから制御を行うシステムやアーキテクチャに関する研究・開発・実用化が盛んに行われている[1][2][3]。日立製作所では既存システムをクラウドリフトし，クラウド環境から制御をおこなう制御クラウドを提案している[4]。制御クラウドでは，クラウド上のコントローラからエッジ環境に配置したアクチュエータに対して制御指令を送信し，また，エッジ環境のコントローラからセンサデータや実行履歴をクラウド上の司令部に送信し，クラウド上の司令部は受け取った情報をもとにエッジ環境へ制御指令を送信する。制御クラウドの概要図を図 1 に示す。

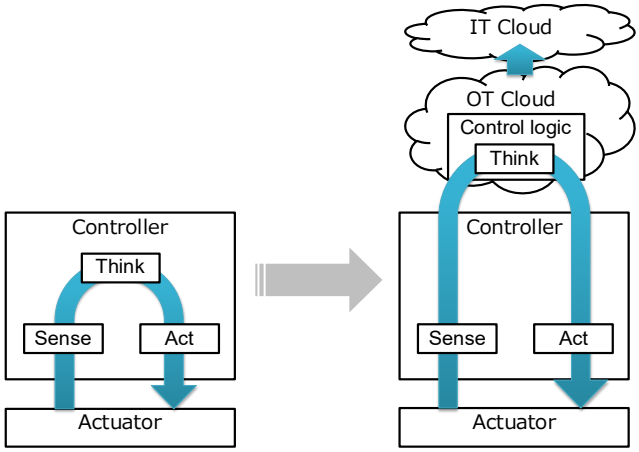

図 1　制御クラウドの構成の概要例

こうした構成において，クラウド-エッジ間をつなぐネットワークの信頼性は運用において重要な指標である。特に，制御システムは，システムの停止・遅延・誤りが起きないことが要件となるため，低遅延かつ単一障害点の無いネットワーク構成が求められる。この要件を満たすことが制御クラウド全体の信頼性を確保するための基盤となる。信頼性を保証するためには，専用線の利用が一つの選択肢として考えられる。しかし，敷設にかかるコストが高く，システム構成の拡張性にも制約がある。これに対して，公衆網や閉域網，特に有線のインターネット VPN (Virtual Private Network) の利用はコスト効率や柔軟性の観点から望ましい。

ネットワークを保有せず他者と共有する場合，ノイジーネイバや急激な利用者数の増加など他者の行動により，ネットワークに予期せぬ高負荷が生じる懸念がある。これにより，スローダウンやパケットロスの頻発といった異常が発生することが懸念される。こうした異常は制御システムの運用において致命的な影響を及ぼしうる。そのため，公衆網や閉域網の利用においても高い信頼性を確保するためには自分以外の利用者がネットワークに影響を与えうることを考慮する必要がある。

さらに，ネットワークの提供者と利用者が異なるため，利用者はネットワークを構成するハードウェア情報やネットワーク経路を知ることはできない。そのため，具体的なネットワークの運用状況や障害の発生した経路，ハードウェア情報の取得は限定的となる。こうした情報の欠如によりネットワークの運用やトラブルシューティングを困難にし，信頼性向上を妨げる要因となる。そのため，高い信頼性を確保するためには，システム構成に非依存な手法でシステムがネットワークの状態を継続的に監視し，適切な障害対応を行うことが必要となる。

2. 関連研究

クラウド環境において障害検知技術を提供するサービスとして Splunk Cloud や AWS(Amazon Web Services)の AWS CloudWatch が存在し，これらは機械学習などにより高度な障害検知サービスを提供している[5][6]。しかし，これらのサービスは収集したメトリクスを特定のサーバに集約し分析する中央集中型のアーキテクチャを採用している。そのため，メトリクスの収集経路や障害判断を行うサーバとの通信が不調の場合には，自動的かつ即座の回復処置の実行が難しくなる。また，仮に障害判定機能を制御システムのアプリケーションが動作するノードに配置した場合には，異常判定処理に要する負荷によりリアルタイム処理に影響が出ることが懸念される。

また，汎用的なメトリクスによる輻輳障害の検知手法の一つとして，通信レイテンシの分布形状から導出される歪度を用いる手法が報告されている[7]。この報告によれば LTE 回線において歪度が負の値のとき，つまり通信レイテンシの分布形状が右側に偏っているとき，輻輳障害と判断できると報告されている。しかし，今回想定する有線 VPN 回線の通信レイテンシの分布形状や通信特性が LTE 回線と同一とみなせるか不明であり，輻輳障害の検知に適用可能かどうかは不明である。

更に，障害検知・対策における制御システム特有の課題としてネットワークレベルでは障害とみなされない障害の検知がある。本研究ではこうしたネットワークの状態を半死障害と定義する。例えば，半死障害にはネットワーク利用者の増加やノイジーネイバに起因するネットワークの輻輳，ケーブルの接触不良による断続的なネットワーク障害の頻発などが含まれる。こうした障害は，クラウド環境からの制御指令の遅延や欠損，センサデータの欠落による誤った指令の作成などの問題を誘発させ，制御システムの要件を不足させる脅威となる。

† (株)日立製作所　研究開発グループ
Hitachi, Ltd. Research & Development Group

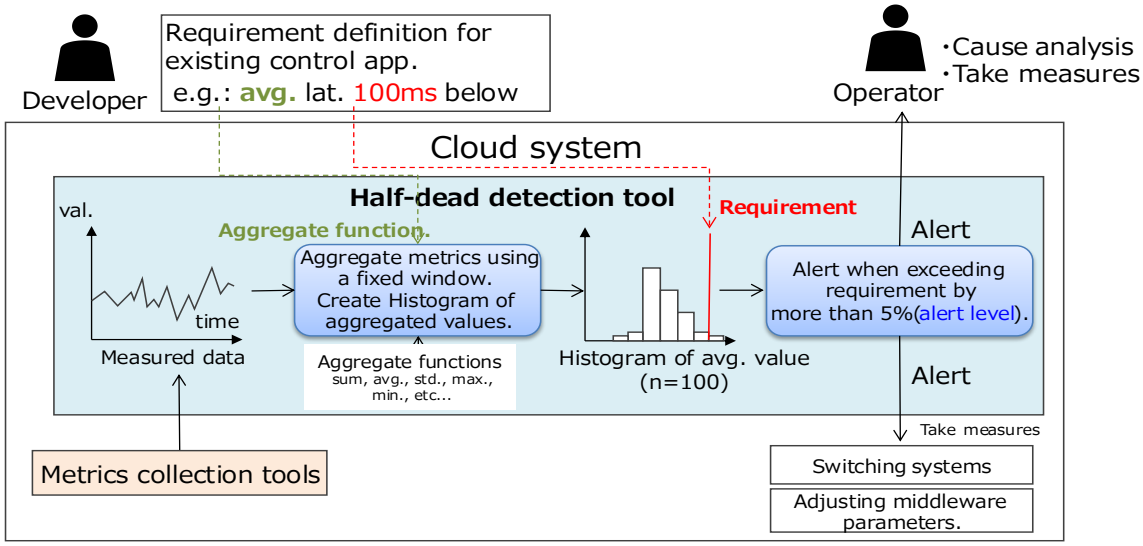

図 2　半死障害検知の概要

また, 運用上の課題として, 通信レイテンシのサンプリング周期 w [sec]の設定の難しさもある。障害判定処理のため何らかの平均値や歪度の導出を行うが, 周期 w が小さい場合には突発的なネットワークの悪化や計測上の外れ値による誤報が懸念される。一方, 周期 w が大きい場合には障害検知に要する時間が増加し障害対策の遅延によりシステムの信頼性が低下する。こうしたトレードオフから, 周期 w の適切な設定が困難であるという課題もある。

3. 研究手法

こうした課題に対して, 本研究はクラウド環境においても制御システムの要件を満たすことをめざし, プローブパケットによる通信レイテンシやパケットロス率などのメトリクスの取得と閾値ベースでの障害判定を行う障害検知手法を提案する。本手法はシステム構成に依存せず低負荷な処理であるため, リアルタイム処理を行う制御用のアプリケーションと同一ノードでの障害検知処理が可能である。これにより障害の影響を受けるノード自身が障害検知を行う自律分散型のアーキテクチャを採用でき, エッジ環境とクラウド環境双方から自律的にネットワーク障害の検知と迅速な障害対策の実施が可能となる。

更に, 本手法は周期 w [sec] に対するトレードオフを解消するため, 周期 w [sec] で取得されたサンプル値を元に計算されるメトリクスを直近 n 個だけ収集する。さらに収集された n 個のメトリクスに対して正常・異常の閾値判定を行い, 異常と判断された割合がある閾値以上の場合に半死障害と判断する。これより, メトリクスの収集数 n により障害の発生頻度と持続性に対して検出感度を調整でき, 誤報の低減が可能となる。

本手法は, レイテンシなどの通信性能を示すサンプル値を収集する機能, サンプル値から統計処理によりメトリクスを計算する機能, 直近に出力された所定個数のメトリクスを用いて障害か否かを判断するアラート機能の 3 つにより構成される。本機能の概要図を図 2 に示す。図 2 では直近 n 個のメトリクスをヒストグラムの形式で表している。

まず, 第 1 の機能は, サンプル値取得用のプローブパケットを送受信する機能である。本研究ではプローブパケットの送受信の方式として, 上りと下りの双方向で UDP (User Datagram Protocol) による通信を行う方式を採用する。プローブパケットに ID とタイムスタンプを付与し, 双方のノードがこれらを元に往路/復路の通信レイテンシ, 往復時間(以降 RTT: Round Trip Time)とパケットロスを検知する。本稿において, レイテンシは RTT の値を指すが, 提案する障害検知の手法は往路/復路の片方向のレイテンシに対しても適用可能である。

第 2 の機能は, プローブパケットにより取得したサンプル値に対して, 一定周期 w [sec] で平均や最大・最小, 分散などメトリクスを計算する機能である。ここで, 本手法で用いるメトリクスは, 平均や最大・最小, パーセンタイルといった計算量が $O(N)$または $O(Nlog\ N)$ (ここで, N はデータ数を表す) の統計処理に限定して採用し, メトリクス計算に必要な負荷を低減させる。

第 3 の機能は, 直近の n 個のメトリクスに対して閾値を超過した個数を計算し, 閾値を超えたメトリクスが M 個 (ただし, M<N) 以上であるとき半死障害と判断する機能である。半死障害の検知は n 個のメトリクスにより実施されるため, 1 個のメトリクスにより障害検知を行う場合より外れ値や突発的な通信品質の悪化による誤報を低減する。また, M=1 とすれば閾値を超えたメトリクスが一つでもあれば即座に半死障害と判断することが可能である。

これらの機能により, システム構成に依存せずアプリノード上で自律的に障害を検知し, 回復措置を実施することでシステムに対するネットワークの信頼性を保証する。

4. 評価

本手法の評価に当たり, 2023 年度に有線インターネット VPN 回線の測定を行った。測定は 0.1 秒毎に行われ, ある 1 日の 0 時から 24 時までの通信レイテンシとパケットロス数が取得される。本測定を数ヶ月にわたって連続的に行った。測定期間のある 1 日について, 1 秒毎に直近 60 秒間に測定された通信レイテンシから計算された歪度を図 3 に示す。Szilágyi らによれば LTE 回線において, 歪度が負の値を取るとき輻輳が発生していると見做すことができると報告されている[7]。図 3 において横軸は時刻, 縦軸は歪度を表しており, 12 時から 14 時の 2 時間において負の値を取っている。この 2 時間の通信レイテンシを見ると, 平常時の平均レイテ

ンシである 10 [msec] に比べおよそ 6 倍となる 60 [msec]の平均レイテンシが観測された。そのため, 本稿では大胆ではあるが, この 2 時間において輻輳障害が発生していると仮

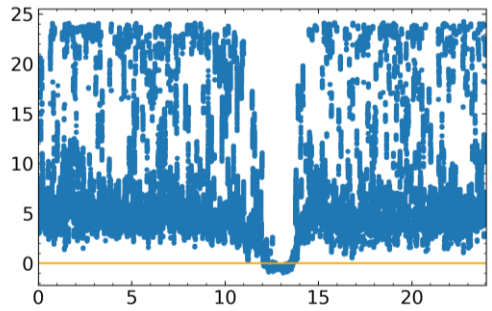

図 3　輻輳発生時の歪度の分布

定し, 障害検知機能の評価を行う。つまり, この 2 時間の障害を検知し, 他の時刻では検知しないことを真として, これに当てはまらない検知を誤報, 未検知を見逃しとして扱う。

このデータセットに対して, 統計処理関数, 閾値, 集計処理を行うメトリクス数, 集計処理の間隔を表 1 のように設定し評価を行う。本稿では, 集約されたメトリクスのうち閾値を超えたメトリクスの割合が 5%を超えたとき障害が発生していると判定する。

表 1 半死障害検知の試験パラメータ

#	集計処理関数	閾値 [msec]	周期 w [sec]	メトリクス数 n
1	Max	75	{1, 5, 50}	{10,50,100}
2	Mean	50	{1, 5, 50}	{10,50,100}
3	95 percentiles	100	{1, 5, 50}	{10,50,100}
4	95 percentiles	60	{1, 5, 50}	{10,50,100}

プローブパケットは 0.1 秒毎に送信し, サンプル値を取得するため, 1 周期内には周期 w*10 個のサンプル値が含まれる。平常時には平均レイテンシが 10 [msec], 最大レイテンシが 100 [msec] , 95 パーセンタイルが 60 [msec] であった

まず, 障害検知において周期ごとの集計値を用いることによる誤報低減効果を評価するため, 平常時にも観測されるレイテンシの値(75 [msec])を障害検知の閾値として設定し, 誤報の発生状況を検証する。次に, 障害の見逃しを防止するための閾値設計について評価を行う。輻輳時には平常時よりも通信性能の悪化が生じることが予想されるため, 平常時の最大レイテンシやその半値, 95 パーセンタイル値を参考に閾値を設定し, その効果を検証する。なお, 最大値の半値も閾値として選択した理由は, 外れ値の影響や平常時の通常変動をある程度許容しつつも, 障害発生時のレイテンシ上昇に反応し障害検知を可能とすると予想したためである。

それぞれの評価結果のうち, n=50 の結果を図 4 にまとめる。図 4 の横方向に周期 w が 1, 5, 50 [sec]の結果を, 縦方向にそれぞれの試験番号を並べている。各グラフの横軸は 0 時から 24 時までの時刻を表しており, 縦軸の 0 と 1 は半死障害の検知結果(ただし 1 のとき半死障害と判断)を表す。

例えば, 図 4 の#1 の w=1, 5 では, 平常時に観測される値を閾値と設定したにも関わらず誤報を発していない。これは, 集計処理やアラート処理による誤報低減の効果が期待通りに働いたと考えられる。一方で, w=50 の場合には誤報が多発している。これは, 周期 w の値が大きすぎる場合には, 観測された最大値が長期間残るため, 誤報低減の効果が十分に得られず, 誤報が頻発したと考えられる。次に, #3 では障害をまったく検知せず障害を見逃している。この原因として, 閾値を高く設定しすぎたことや, 輻輳中に一時的な通信の回復が発生し, 平常時と同程度の通信レイテンシが観測されたことが挙げられる。その結果, 集約されたメトリクスの中で, 閾値を超過するメトリクスの割合が減少し, 障害の見逃しが発生したと推察される。一方, 集計処理関数として#2 の平均値や#4 のパーセンタイル値を用いた場合, 誤報の発生を低減し, 障害の見逃しも少ない, 安定した障害検知が実現された。この要因として, 平均値やパーセンタイル値は一時的な外れ値や短期的な変動の影響を受けにくく, 障害検知の安定性が向上する点が挙げられる。さらに, 閾値を平常時に観測される 95 パーセンタイル値や最大値の半値を基準として設定することにより, 平常時の通常の変動を許容しつつ, 輻輳時に発生するレイテンシの上昇を的確に検知できたことが, 誤報や障害の見逃し低減に寄与したと考えられる。

次に, この閾値設定についての考察を述べる。一般に, ネットワーク輻輳はパス上の特定箇所で発生し, 該当するネットワーク機器が大量のパケットを受信すると, キューイング処理によりレイテンシが増大する。このキュー長は機器ごとに固定であるため, そのサイズに応じて最大レイテンシが決定される。キュー長から溢れたパケットはパケッ

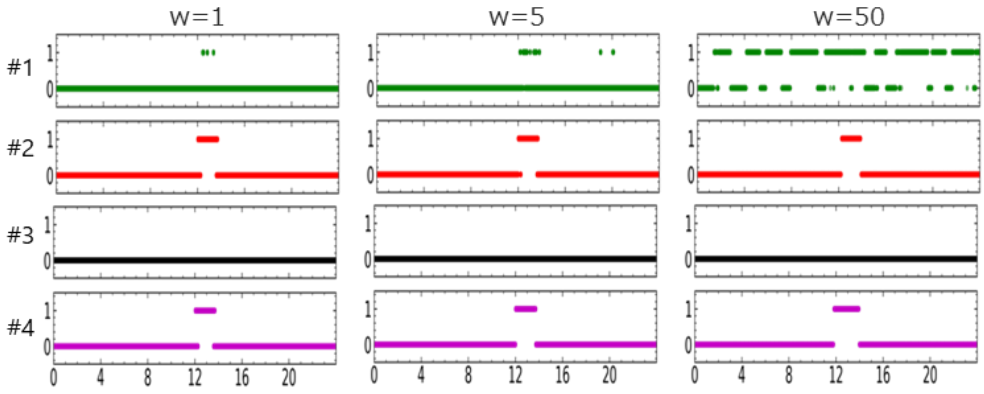

図 4　半死障害検知の検知結果

トロスとなるため, 輻輳発生時にはこの最大キュー長が通信レイテンシに対して支配的な影響を及ぼす。この当該機器は平常時においてもパケットを最大キューまで溜めることがある。以上のことから, 平常時の最大レイテンシや 95 パーセンタイル値が半死障害である輻輳障害の検知閾値の設定に活用できたと推察する。

一方で, ネットワークパス全体で考えた際には, ボトルネックとなる当該機器以外でも確率的にレイテンシの増加は生じるため, 輻輳時の最大レイテンシは平常時の最大レイテンシを上回る場合がある。また, ネットワーク経路の変更によりボトルネックとなる箇所が変わると, 最大レイテンシも変動する。そのため, 閾値として用いる値はネットワークの状態に応じて定期的に見直す必要があると推察される。

5. 結論

本研究では, ネットワークの半死障害を検知するための手法を提案した。本手法は, システム構成に依存せず低負荷な処理と自律的判断により半死障害の検知を行うため, アプリケーションと同一ノードで動作させることができ, 即座に回復措置を行うことが可能である。具体的には, プローブパケットを定期的に送信し, 取得したメトリクスから閾値ベースの障害検知を行うことで, 通信レイテンシやパケットロス率を取得し, 半死障害を検知する。

評価結果から, 提案手法は輻輳を安定して検知できることが確認された。特に, 平均やパーセンタイルを用いた統計処理と適切な閾値設定により, 誤報を低減しつつ迅速な障害検知が可能であることが示された。また, 輻輳の発生原理から適切な閾値設定の方針を示した。これにより, 半死障害検知を適用する前に短時間のネットワーク性能の計測などにより適切な閾値の設定が可能であると考える。

本手法の導入により, クラウド環境およびエッジ環境の双方において自律的なネットワーク障害検知が可能となる。これにより, 障害発生時には迅速な対応が実現され, クラウド環境においても従来のオンプレミス環境と同等の高い信頼性を担保できるようになる。この結果, 従来は厳しい信頼性要件によりクラウド利用が困難であった制御システムに対しても, クラウドの活用が期待できる。さらに, 本手法の適用により, オンプレミス前提のシステムと比較して, コスト効率, システムの柔軟性・拡張性, 保守性の向上といった利点が得られることが見込まれる。

参考文献

[1] Jan Peleska, “New Distribution Paradigms for Railway Interlocking”, ISoLA 2020, Leveraging Applications of Formal Methods, Verification and Validation: Applications, vol 12478. Springer, Cham. doi: https://doi.org/10.1007/978-3-030-61467-6_28

[2] B. Chen, J. Wan, L. Shu, P. Li, M. Mukherjee and B. Yin, "Smart Factory of Industry 4.0: Key Technologies, Application Case, and Challenges," in IEEE Access, vol. 6, pp. 6505-6519, 2018, doi: 10.1109/ACCESS.2017.2783682.

[3] D. Cotroneo, L. De Simone, P. Liguori, R. Natella and N. Bidokhti, "Enhancing Failure Propagation Analysis in Cloud Computing Systems," 2019 IEEE 30th International Symposium on Software Reliability Engineering (ISSRE), Berlin, Germany, 2019, pp. 139-150, doi: 10.1109/ISSRE.2019.00023.

[4] 日立評論, “2025 年日立技術の展望:情報制御システムの新価値創生を支えるクラウドベース高信頼プラットフォーム.” [Online]. Available: https://www.hitachihyoron.com/jp/archive/2020s/2025/01/16/index.html. [Accessed: Mar. 26, 2025].

[5] Splunk Inc., “Splunk Cloud.” [Online]. Available: https://www.splunk.com. [Accessed: Mar. 25, 2025]

[6] Amazon Web Services, Inc., “Amazon Cloud Watch.” [Online]. Available: https://aws.amazon.com/jp/cloudwatch/. [Accessed: Mar. 25, 2025]

[7] P. Szilágyi and C. Vulkán, "LTE user plane congestion detection and analysis," 2015 IEEE 26th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC), Hong Kong, China, 2015, pp. 1819-1824, doi: 10.1109/PIMRC.2015.7343594