

クラウド型連動装置の構成と処理手法

寺田 夏樹* 潮見 俊輔* 遠山 喬*

Study on Interlocking Device on Cloud Computing Environment

Natsuki TERADA Shunsuke SHIOMI Takashi TOYAMA

We report our study and proposal on interlocking devices in a cloud computing environment, which realizes resilience, reduction of time and cost for replacement, and provides the interlocking process as a service. The proposed system has the feature of processing interlocking logics of different stations independently. The proposed system has three layers: terminal devices to interface track circuits, signals, and switches, etc., logic units to process interlocking logics, and controllers to assign the interlocking table to logic units. We defined the specification of the terminal devices, logic units and controllers, and carried out verifications on them.

キーワード：連動装置，クラウド，Interlocking as a service，セグメント，非同期処理

1. はじめに

連動装置は、列車の位置や進路に関する指示に基づき、信号機や転てつ機等の現場機器相互の連鎖を確保しながら、駅構内の現場機器制御を行うものである。その機能はリレーや電子計算機などにより実現されてきた。従来から存在するリレーによる連動装置（継電連動装置）では、駅毎もしくは駅構内を分割した小単位毎に設ける構成が採られてきた。近年電子計算機による連動装置（電子連動装置）も広く利用されるようになった。電子連動装置も当初は駅ごとの構成のみであったが、最近では同一線区上の複数駅を制御する集中連動方式の開発や実用化が進んでいる。集中連動方式は、連動装置の維持管理の集中化など保全面等での効果があり、特に地方線区における運営の低コスト化に貢献している。

鉄道総研では、集中連動の概念を拡張し、クラウドコンピューティングの概念を導入することで、電子連動装置の論理演算機能を異なる線区の複数の駅に対してサービスとして供給する、クラウド型連動装置の開発を進めている。本報告では、最初に現状の連動装置の課題をもとにクラウド型連動装置のコンセプトをまとめ、解決すべき技術課題について述べる。次に、クラウド型連動装置の実現方法として提案する構成と各部の機能仕様の概要を示す。最後に、その実現性についてシミュレーションにて検討した結果を示す。

2. 連動装置のクラウド化のコンセプト

クラウド型連動装置の開発を進めるにあたり、各種連

動装置の現状を把握し、こういった形態での装置の実現が望まれるか検討を行った。

2.1 連動装置の現状

駅構内の列車の進路を制御する連動装置は、論理演算を行うハードウェアによって、機械連動装置、継電連動装置、電子連動装置等に大別される。

現在では電子連動装置は、JR各社の連動装置の約40%を占める。1985年の東神奈川駅での実用化から35年を経ており、今後高経年の設備が増加することから、寿命延伸に対する技術開発のニーズが高まっている。一方、中小鉄道事業者の連動装置の80%を占める継電連動装置は、1960年代に多く導入された。現在、その多くが50年を越える経年と推察される。連動装置の更新は、結線や配線の作成と審査、施工のための設置スペースや建屋の確保、新旧を並立させるための配線工事、新旧切り替えての動作確認等、更新までの工数が多い。更新の容易化に対する技術開発のニーズは高いと考える。

連動装置は駅構内で1台、進路数の多い大駅では複数台用いる構成が一般的であるが、近年、同一線区の複数の駅を1台の電子連動装置で所掌する「線区集中連動装置」が開発、導入されている¹⁾²⁾。線区集中連動装置では、論理の演算を行う1台の論理部と、現場機器を制御するための電子端末を各駅に設置し、論理部と電子端末間を伝送回線に接続している。線区全体を1つの駅構内と同様に処理することで、論理部の台数削減に加えて、駅間の閉そく装置の機能の内包が可能となった。集中連動装置は、進路数が少ない地方線区のほか、整備新幹線でも導入されている³⁾。

また、連動装置と現場機器、現場機器相互の制御や表示情報の通信手段に従来の1:1の接続に代わり、ネッ

* 信号技術研究部 信号システム研究室

トワークを利用する「ネットワーク信号」が2004年から開発され⁴⁾、一部線区で導入されている。

2.2 連動装置の課題とクラウド化

現在の連動装置は制御対象の駅や線区沿線に設置する制約があり、電子連動装置のように信頼性確保のための冗長系を設ける場合であっても、同一機器室内に設ける構成がとられている。そのため、大規模災害や地絡事故等による連動装置の全損が発生した場合に復旧に多大な時間を要する問題が顕在化している。また、前述の通り更新時の工数や機器室等の付帯費用の高さから、特に資金力の弱い中小鉄道事業者においては、更新を行う行為そのものに困難さがあることが課題である。

ネットワークを活用した計算資源の提供方法の一つとして、ソフトウェアやデータをネットワーク上のサーバ群から提供する「クラウドコンピューティング」がある。サーバの物理的位置や台数、個々のサーバの稼働状況を意識せずに、利用者にサービスを提供する形態は、サーバに対する災害や更新に関して頑健であることが期待される。この利点は、現在の連動装置における課題点に対しても有効となりうる。そこで、連動装置の機能をクラウドコンピューティングに類似した形態で提供する装置を「クラウド型連動装置」と定義し、その利点と課題について検討を行った。

2.3 クラウド型連動装置の実現形態と期待される効果

図1にクラウド型連動装置の実現イメージを示す。鉄道事業者やメーカ等が保有する「連動センター」に連動装置の論理演算機能を集約し、各駅の現場機器を直接制御する形態としている。また、現在の連動装置とは異なり、連動センターが所掌する被制御駅は単一の駅や線区に限定されず、異なる線区の複数駅を被制御駅とすることを可能とする。連動センター内では、処理対象駅を正副複数の連動論理部に対して割り当てることにより、現在の電子連動装置と同様に故障時の冗長性を確保する。

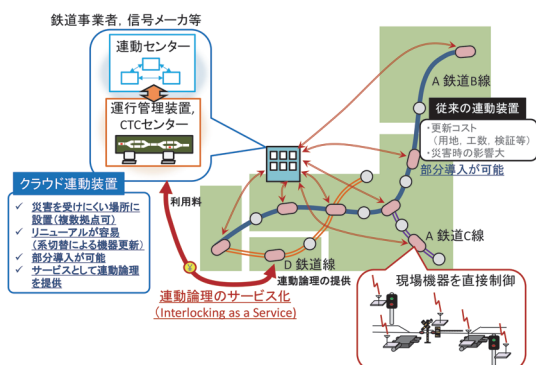


図1 クラウド型連動装置の実現イメージ

また、運行管理装置やCTC等に対して進路制御の入力と、在線位置や進路、信号機の現示等の表示情報の出力を行う機能を有する。

期待される効果としては、以下が挙げられる。

① 災害への強靱化

連動センターの物理的位置が、特定の駅や線区の沿線でなければならないという制約を受けないため、地震や水害を受けにくい箇所へ連動センターを設置することによる災害の強靱化が期待できる。また、異なる箇所に設置されている連動論理部を連携させて仮想的な「連動センター」を構成する形態では、稼働系と冗長系が異なる箇所での処理を行うため、更なる強靱化が期待できる。現場機器に関しても災害発生時にケーブルが断線しても、無線で接続することにより回復するということが期待できる。

② 機器更新に伴うコストの削減

処理対象駅を稼働中の連動論理部に移動させることが可能となるため、連動論理部の更新に伴う処理の停止は最小限となる。また、ハードウェアやソフトウェアの動作確認について、更新した系に対して稼働系と同じ入力を与えて照合することにより、動作試験で照合すべき進路や操作の一部を代替することが可能となり、照合試験における工数削減が期待される。さらに連動装置更新に伴う配線作業や機器室新設等の工数が削減されるため、クラウド化以降の更新については、機器と付帯工事に関するコスト低減が期待される。

③ サービスとしての連動論理処理の提供

複数駅の連動論理処理を行う機能により、クラウド型連動装置の余剰な処理能力の他社や他線への提供や、複数鉄道事業者の連動論理処理の実施を前提とした、連動論理処理のサービス化が実現できる。異常時の対応や責任分界等の課題はあるものの、連動論理処理の対価として利用料を徴収する、サブスクリプション方式のInterlocking (連動) as a Service (IaaS) と呼べるビジネスモデルが成立しうる。

2.4 海外の動向

日本国内とほぼ同時期に Interlocking as a Service の概念が提唱された欧州では、既存の電子連動装置をベースに遠隔、集中配置した連動装置とIPネットワークを介した現場機器の制御技術を組み合わせた連動装置の開発と実用化が進められている。ドイツ鉄道の DSTW⁵⁾ (Digitales Stellwerk, デジタル連動) や、Siemens 社の SIMIS IaaS⁶⁾ が代表例であり、2018年頃から実証試験を兼ねた実用化が進められている。DSTW は保守の集中化や遠隔制御などのクラウド連動のメリットを、1箇所に電子連動装置を多数配置することで実現する形態である。

3. クラウド型連動装置の基本構成の検討

3.1 現在の電子連動装置の処理手順とクラウド型連動装置における技術的課題

現在の電子連動装置の論理部は通常、稼働系と待機系の1組の論理部から構成されている。装置が処理する連動論理は、内部に各駅の論理をROMやストレージ等により入力し、処理時には高速なRAM上に読み出した上で実行している。連動装置と処理対象駅は1:1ないし1:Nとして固定的に対応づけられており、稼働しながら対象駅を変更することはできない。

入出力や論理演算等の処理は、設計時に定めた一定周期毎に実行される。処理対象駅が複数存在する場合であっても、入出力や論理演算は一定時間周期毎に1回のタイミングで全駅をまとめて実行する。そのため、工事等で処理を停止させる場合に特定の駅のみ処理を停止させることはできず、処理対象駅全てを停止させる必要がある。また、論理を変更する際には、変更箇所のみならず他駅の制御に影響する可能性がある。

現在の電子連動装置は安全性の確保のために、2組ないし3組のCPUとメモリを備え、処理過程や処理結果を照合して不一致があった際に安全側動作に移行する、フェイルセーフCPUを用いている。

また、装置間の通信手段にネットワークを用いる連動装置等は、専用のネットワークを構成し、装置間の通信遅延に上限を設けてそれ以上の遅延を許容しない構成を採用している。

したがって、クラウド型連動装置を実現するためには、以下の(a)~(c)の技術的課題を解決することが必要となる。また(d)については、処理対象駅が数百を超える大規模システムを低い初期コストで実現する際に必要となる。

- (a) 駅ごとの処理の独立性を確保した処理手法
- (b) 連動論理部への処理の割当および管理手法
- (c) ネットワークの遅延を許容する処理手法
- (d) 汎用CPUによる高安全な連動論理処理手法

本稿では、(a)~(c)の開発について述べる。以下各節において検討概要を示す。なお、(d)についてはネットワークを介して多数決を行うフェイルセーフ構成を提案しているが⁷⁾、まだ基礎検討の段階である。そのため連動論理の処理については、既存のフェイルセーフCPUを用いる前提でシステムの検討を行った。

3.2 駅ごとの処理の独立性を確保した処理手法

現在の電子連動装置で複数駅を取り扱う場合、論理演算や入出力は駅ごとに細分化せず一括で処理を行う方式を採用しているが、クラウド型連動装置においては駅ごとの処理の独立性を確保することが機能上必要とな

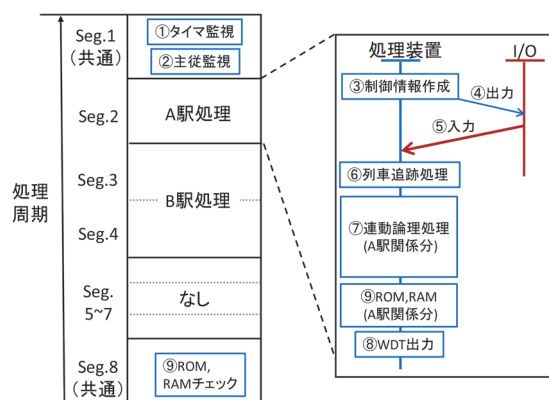


図2 セグメント分割と処理の割り当て

る。そこで、電子連動装置が周期的に行っている処理を、共通処理と各駅個別の処理に分けた上で、それぞれを連動装置の処理能力を細分化した単位(セグメント)に割り当てて順次実行する方式を提案する。

図2にセグメントの分割と処理の割り当ての概念を示す。図2の「seg.」はセグメントの略である。各駅で個別に実行する処理としては、制御情報の作成、入出力、列車追跡処理、論理演算、各駅固有のデータに関するROMおよびRAMのチェック、ウォッチドックタイマの出力が挙げられる。各駅の進路数やリレー数によって入出力や論理演算に必要な処理能力は異なるため、進路数の多い駅にはセグメントを多く割り当てる必要がある。図2に示す例では、一定処理周期で実行している処理を8セグメントに分割し、1番目と5番目のセグメントを共通処理に割り当てて、2番目にA駅の処理を3~4番目にB駅の処理を割り当てている。また、5~7番目は余剰の処理能力である。このようにセグメント単位で処理を割り当てることにより、処理順序を定義づけることができる。更に、連動論理部毎の余剰処理能力を定量的に示すことが可能となる。

3.3 連動論理部への処理の割当・管理手法

複数の連動論理部を組み合わせる多数の駅の連動装置の機能を提供するクラウド型連動装置では、複数の連動論理部に処理対象の駅を割り当てる機能が必要となる。また、連動論理部の動作が停止した場合等の処理が正常に実施できない場合には、他連動論理部の余剰セグメントへ処理を動的に再割り当てする機能が必要となる。これらの実行制御機能を担う「コントローラ」を連動論理部と対で設けることで、連動論理部の機能は現在の電子連動装置を踏襲しながらも、多数の連動論理部を連携させる機能を実現する手法を提案する。ここでコントローラは連動論理データを保有し、連動論理部は必要に応じてそのデータをダウンロードする。複数のコントローラが存在する場合は連動論理データを共有し、協調して連

動論理部の計算資源の割り当てを行う。

コントローラによる動的割り当て機能により、連動論理部については必要とする負荷に応じてその数量の増減が可能、すなわちスケーラブルとなり、保守による部分的な停止や更新も実現可能となる。

図3にシステム構成を示す。システムは、コントローラと連動論理部のほか、現場機器と接続する現場端末、連動用ネットワーク上の情報を取得し記録する監視端末、進路制御端末、データ書き換え装置から構成される。コントローラ相互間、連動論理部と現場端末間、進路制御端末と連動論理部間はそれぞれネットワークで接続する構成としている。

ネットワークについては、図3では有線としているが、無線による接続も可能である。接続する機器はホワイトリスト形式で共有し、リストにない機器からの接続は無視する。新たな機器の追加時はデータ書き換え端末を通じてリストを更新する。暗号化や認証といったセキュリティ対策が必要であり、認証サーバも設ける。ただし、クラウド型連動装置に特有となるセキュリティ要件はなく、利用するネットワークの種類や特性に応じてセキュリティ要件を定める必要がある。特に無線を使用する場合などはセキュリティ対策の強度を上げる必要がある。

3.4 伝送遅延の影響を小さくする連動論理

現在の電子連動装置等では、連動論理部で扱う現場端末の情報が所定の周期内に全て揃う（同期する）ことを前提に、軌道回路や転つ機等の表示情報等を入力とした論理演算を一定周期で行っている。現場端末からの情報の欠落や遅延が発生した場合、連動論理部の処理が停止するため、伝送遅延や端末の処理時間に対しては相應の性能と信頼性が要求されている。

しかし、連動論理部と被制御駅、連動論理部相互を物理的に離れた位置に配置することを許容するクラウド型

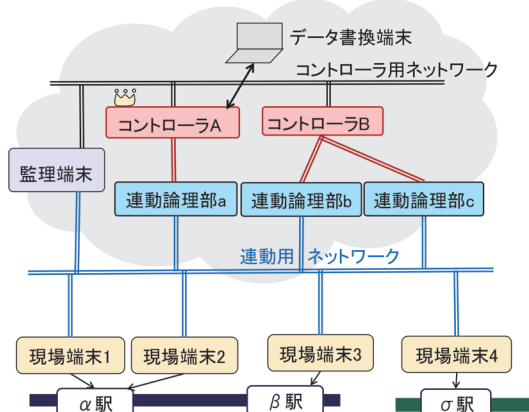


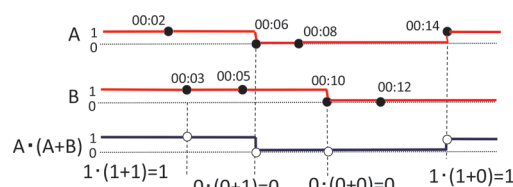
図3 連動論理部間の連携を可能とするシステム構成

連動装置において同様の前提条件を置くことは、ネットワーク設備等のコスト増加の要因となりうる。また、信号機器をネットワークで接続する考え方がより一般化した際に、低コスト化や保守性向上の観点から、有線に限らず、無線等の多様な伝送方式に対応できることが望ましい。このため、連動論理部で扱う現場端末の情報が一定周期内に全て揃うことを前提としない、連動論理部と現場機器の非同期動作を実現し、かつ情報の欠落や遅延が連動装置の機能に与える影響を低減する方法を開発することが必要である。

現在の連動論理は、バイナリ表現の「1」と「0」、リレーの「動作」と「落下」等の2値論理を基本としている。非同期動作では、伝送の遅延による情報の「不定」の状態を考慮する必要があるため、コンピュータを用いた信号装置の黎明期において研究がなされた3値論理による手法を提案する。3値論理は、「不定」状態に対する演算方法の違いにより、C形フェイルセーフ論理（「不定」に基づく演算結果はすべて「不定」）やφ形フェイルセーフ論理がある。後者は入力に「不定」が含まれていても、出力に影響がなければ「0」や「1」に確定することから、「不定」状態による処理の遅延を改善する効果が見込まれる。また、連動論理を構成する論理式には変更を必要としないため、現在の電子連動装置で用いられる結線入力方式やマトリクス方式のデータをそのまま適用することが可能である。

図4に入力A、Bに対する出力A・(A+B)について、入力が揃ってから処理を実施する現在の連動装置をベースとした処理と、3値論理を用いた非同期動作のタイムチャートを示す。ここで図4下図のφは「不定」を表す。従来の手法ではA、Bそれぞれが「0」と確定した「00:10」の段階で出力が確定したが、3値論理では、Aが「0」と確定した「00:08」で出力が得られる。

従来の連動装置の論理演算(2値論理)



3値論理を用いた非同期動作

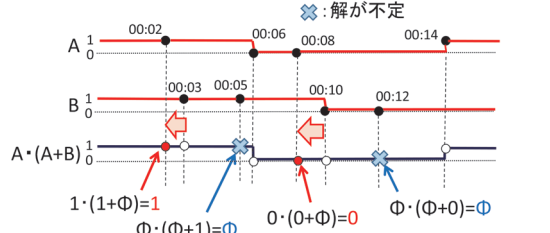


図4 非同期連動処理の例

4. 各構成要素の機能仕様の検討

図3に示したシステムを構成する各部の機能仕様を検討した。上流となるコントローラ、連動論理処理の中心となる連動論理部、そして現場機器と接続される現場端末の順にその検討結果を説明する。

4.1 コントローラの機能仕様

本章では、3.3節で紹介したコントローラの機能仕様について検討した結果について述べる。検討の前提として、コントローラは複数台を設置可能とした。また、コントローラと連動論理部の関係は1:nの関係とした。すなわち各コントローラに対して、その配下となる連動論理部が複数台存在し、逆に連動論理部から見たコントローラは特定の1台となる。

4.1.1 コントローラ相互の死活監視、データ共有機能

コントローラは相互にその死活監視を行う。また、コントローラや連動論理部のリストおよびその状態、連動論理データやセグメント割り当てデータを共有する。この死活監視とデータの共有は同じ枠組みの中で実施する。コントローラのうち、任意の1台をマスターコントローラ、その他をスレイブコントローラと定め、マスターがスレイブに対して順次問い合わせを行う、マスタースレイブ方式でのポーリングを基本的な死活監視の仕組みとした。

死活監視とデータ共有の基本的な流れを図5に示す。マスターコントローラは保有するデータのリストをスレイブコントローラに送る。スレイブは、自身が保有するデータと比較しマスタよりも新しいデータのリストを、自身および配下の連動論理部の状態と合わせてマスタに返送する。マスタからの保有データリストに対して、スレイブが保有していないデータがあった場合は、マスタに別途データの要求をし、スレイブにダウンロードする。逆にスレイブが保有するデータの中にマスタが保有しない情報があれば、返送されたリストに従い、マスタがスレイブに要求を送りダウンロードする。

コントローラ間の死活監視において、マスタからの問

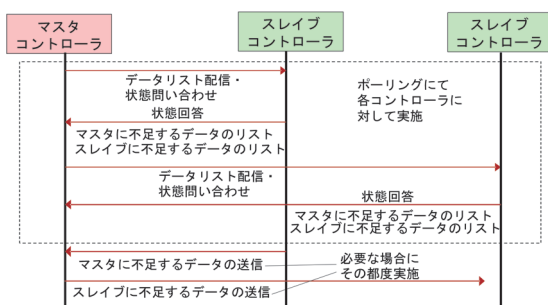


図5 コントローラ総合の死活監視とデータ共有

い合わせに対し、スレイブから反応が一定時間返ってこない場合は、マスタがスレイブの故障と判断する。一方、スレイブにおいてマスタからの問い合わせが一定時間来ない場合は、マスタの故障と判断する。この場合、残されたスレイブの中から新たにマスタを選定する。

4.1.2 マスタ権の選出およびマスタ統合機能

マスタ故障時の新マスタ選出にあたっては、選出ルール（例えば、IDが若いサーバを選定する）を決めたうえで、その選出ルールで最優先となるサーバがマスタとして立候補し、それを承認することで、以後そのサーバがマスタとして動作するというのが基本的な考え方である。この新マスタ選出にあたって、コントローラが動作していないなどの理由で、一部コントローラからのマスタ承認が得られない可能性も想定されるため全コントローラからの承認は必須とはしない。一方、スレイブが停止している間にマスタが交替した場合などで、旧マスタからの死活監視問い合わせが停止している間に別のコントローラからの死活監視問い合わせがある場合は、そのコントローラをマスタとして認識するものとする。

ここでコントローラの故障判定を通信の有無で判断した場合、ネットワークの障害と装置の故障とが区別できない。そのため、ネットワークの障害が発生しコントローラが2つ以上のクラスタに分離した場合は、実際にはマスタが動作し続けているのにも関わらず、マスタから分離されたクラスタにおいて別途マスタが立ち上がることになる。つまりネットワーク分離を考慮すると、再統合の際のマスタ統合処理について検討が必要となるが、これらの処理についても手順を定めた。

4.1.3 連動論理部の状態監視

各コントローラは配下の連動論理部のリストを所有しており、そのリストに従い配下の連動論理部に対しポーリングをかける。連動論理部に対して現在のセグメント割り当て状態を送るとともに、連動論理部の自身およびセグメントの状態に対する照会をかける。連動論理部は、それに対して応答をする。

4.1.4 監視端末(コントローラネットワーク上の仕様)

コントローラが連動論理部の死活監視を行うとはいへ、連動論理部はコントローラが停止しても動作し続ける前提としている。しかしコントローラが停止すると、その配下の連動論理部の死活監視機能も停止してしまう。そこでコントローラによる連動論理部の状態監視機能を補う装置として監視端末を設ける。

監視端末はコントローラ用ネットワークと連動論理用ネットワークの双方に接続し、連動論理部の死活監視を行うものである。仮にコントローラが停止しても、監視端末が連動論理部の状態を取得し、他のコントローラにその情報を送ることで、連動論理部が停止した場合でもそれを検出することが可能となる。

コントローラが停止している場合にその配下の連動論理部に新しいセグメントを割り当てたり、すでに割り当て済みセグメントの解放を行ったりすることはできないが、さらにその連動論理部の一部が停止した場合にはその代替となる新しいセグメントの割り当てを他のコントローラ配下の連動論理部に行うことが可能となる。

監理端末が全ての連動論理部を監視できるとは限らないため、監理端末についても複数台配置して、それぞれが、担当範囲の連動論理部の状態を監視する形をとる。

4.2 連動論理部の機能仕様

連動論理部は、連動用ネットワークを介して、現場端末からの表示情報を受信し、各駅の連動論理処理を実行し、現場端末へ制御情報を送信する。クラウド型連動装置における連動論理部の機能は、大きくセグメント機能と連動論理処理機能に分けられる。

連動論理部のセグメント機能は、上位のコントローラの持つセグメント割り当て機能と連動論理部の状態監視機能によってセグメント割り当て状態を受信し、連動論理部の計算資源を割り振るものである。図6に、1台の連動論理部全体の状態遷移図を示す。現在の一般的な電子連動装置では、連動論理データの変更時に、連動論理部の再起動を行うのに対し、クラウド型連動装置では連動論理部自体の再起動は行わず、定常処理中に駅単位で初期化を行うのが特徴である。

連動論理処理機能については、駅単位に割り振られた計算資源の中で、現場端末から受信した表示情報を解析し、連動論理を適用し、制御情報を生成する、連動装置の中核となる機能である。クラウド型連動装置においては、現場端末と連動論理部が非同期で動作を行う非同期連動処理に対応している点が特徴である。

4.2.1 セグメント機能の詳細

セグメント機能においては、セグメント間で相互に影響する部分を最小化することが安全性、アベイラビリティの観点で重要である。このため、計算資源の分割と保護の考え方について整理を行った。

大規模な駅等では、1つの駅の処理に複数のセグメン

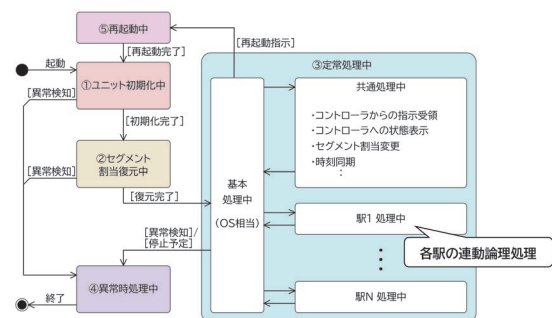


図6 連動論理部の状態遷移図

トを使用する。このため、1つの駅用に割り当てられたセグメントの組を「タスク」として定義し、同一のタスクに所属するセグメントに対し、コントローラ側でコンテキストIDを付与する方法を採用した。なお、各セグメントで状態の復元、保存を行うため、同一タスクに属するセグメントが連続している必要はない。

連動論理部は、OSに相当する基本機能によって、セグメント単位の一定時間毎にタスクを切り替えながら実行する。各処理は、時間内に完了することを前提とするが、異常の発生により完了しない場合には、基本機能により処理時間に応じて強制的にタスク切り替え（プリエンプション）を行う。当該タスクは異常検知状態となり入出力は停止する。ただし、現場機器における安全は現場端末側で確保する仕組みとなっているため問題はない。

計算資源のうちメモリ（不揮発性のROMと揮発性のRAM）については、セグメント単位で独立した領域を固定的に確保し、他のタスクからはアクセスできないように保護する。なお、クラウド型連動装置は連動論理データを変更する際に、連動論理部自体の再起動は行わない。このため、クラウド型連動装置では、RAM上のデータが主体であり、ROM上のデータはコントローラが停止中でも連動論理部を起動可能にするためのバックアップとして機能する。

図7にセグメント（タスク）の初期化から解放までの状態遷移を示す。各タスク内で異常を検知した場合には、連動処理を停止し、セグメントを解放する。この解放は、新たにセグメント割り当てが可能な状態にすることを意味し、データの消去等を行わない。これは故障原因究明等のために状態保持できるようにするためである。

4.2.2 非同期連動処理の詳細

非同期連動処理については3.4節で紹介したが、非同期連動処理をフェイルセーフCPU上で実装できるよ

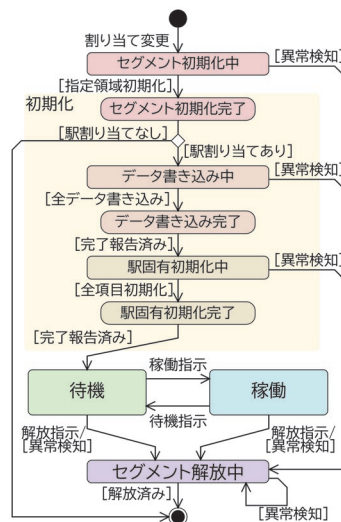


図7 セグメントの状態遷移図

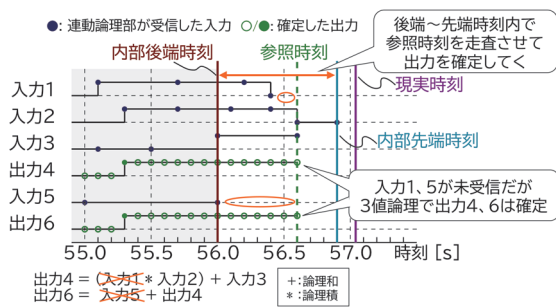


図8 連動内部時刻の定義と出力確定手順

う、時間の扱いについて具体化を行った。非同期連動処理では、現場機器の状態が変化した時刻、その情報が連動論理部で受信された時刻、その情報を使って制御出力を決定する時刻が、それぞれ異なる。このような時間の概念に対処するために、連動論理部内部の時間体系を定義した。図8にその概念を示す。図8は、現実時刻、内部先端時刻、参照時刻、内部後端時刻の4つの時刻を示しており、現実時刻以外は、タイムスタンプに基づく時刻である。内部後端時刻は、全ての入出力が確定している時刻であり、これより過去は、既に全ての状態が確定しているため再計算が不要である。内部先端時刻は、連動論理部が受理している最も現実時刻に近い入力の時刻であり、これより未来は、情報不足で状態の確定が期待できないため計算が不要である。このように、連動内部時間を有限の範囲に設定することで、ある現実時刻において、不要な処理の実行を避けることができる。参照時刻は、連動論理部が連動論理の計算を行う連動内部時間における時刻である。連動論理部は、1処理周期のタスクに割り当てられた時間の許す限り、内部後端時刻と内部先端時刻の間で参照時刻を走査させながら、出力を確定させる。また、仮想的なリレー単位でタイムアウト判定を行う仕様とした。

4.3 現場端末の仕様

現場端末は、軌道回路や信号機、電気転つ機などの現場機器の制御や表示情報の取得を行う。そのため、①制御情報の受信、②優先度に従った制御情報の処理、③制御情報の照査、④現場機器の制御、⑤現場機器からの表示の取得と出力に関する処理機能を備える。クラウド型連動装置では、信号機器室や器具箱等に現場端末を配置して多数の現場機器を集中制御する形態と、現場機器内やその近傍に現場端末を配置して1台～数台の現場機器を制御する分散的な形態のいずれも取りうる。したがって、機能仕様は両形態に適用できるものとした。

現場端末に対し、複数の連動論理部から出力される制御情報に不一致があった場合の異常検出の機能については、4.1節において紹介したコントローラによる連動論

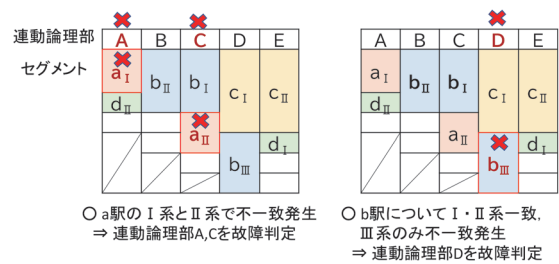


図9 監視端末による連動論理部の不良判定

理部の死活監視機能を補う監視端末に設けた。

4.3.1 制御情報の受信、優先度に沿った制御情報の処理

連動論理部から送信された制御情報は、現場端末ですべて受信する。一定時間周期で処理を行う現場端末は、処理周期のはじめに受信した制御情報から自端末が属する駅の制御情報の有無を判定し、該当する制御情報が含まれる場合は表示情報の出力を行う。また、端末配下の現場機器に対する制御情報が含まれる場合は、制御情報の照査を行う。また、上記のいずれにも関係しない制御情報については破棄する。

4.3.2 制御情報の照査と現場機器の制御

通常、1つの連動論理データは複数台の連動論理部で演算を行うため、現場機器に対する制御情報は複数の連動論理部から送信される。複数の連動論理部から送信された制御情報のうち、制御情報に付与された「優先度」が最も高いものを現場端末側で選択して制御に用いる。

また、現場端末は複数の連動論理部からの制御情報を受信し、情報の不一致や不着が検出された際には監視端末に通知する。各連動論理部の演算や制御出力のタイミングは非同期であり、かつネットワーク上の遅延の影響を各制御情報が受けるため、現場端末への制御情報の到達は同期しない。そのため、情報の不一致や不着の検出は制御情報のタイムスタンプ時刻に基づき行われる。また、情報の不一致や不着は監視端末に伝達する。

4.3.3 監視端末（連動用ネットワーク上の仕様）

監視端末は、コントローラによる死活監視を補うものであるが、連動用ネットワークにおいては動作ログの取得とその解析による連動論理部の動作判定を行う。さらに現場端末から送信される連動論理部からの制御情報の不着や不一致に関する情報を受信する。監視端末はセグメント割り当てデータを用いて、制御情報の不着や不一致が発生した論理部とセグメントを特定する(図9)。

こうして異常を検出した際に、これを取得した監視端末がコントローラ用ネットワークを通じて連動論理部の動作状態をコントローラに共有する。

5. クラウド型連動装置の実現性の検討

5.1 コントローラ機能仕様の検証

コントローラの機能について、モデル検査手法を利用して機能検証を実施した。モデル検査とは、システムを記述したモデル（一般的に状態遷移モデルとして記述される）の性質を計算機により網羅的に調べる手法の総称である。古典的なモデル検査手法ではあるが、元々が通信プロトコルの検証手法として開発が進められた SPIN⁸⁾ を使用して、コントローラ機能、例えば死活監視の問い合わせ、マスタが停止したときの新マスタの再選定、さらにはネットワーク分離時やマスタ統合時の振る舞いについて検証を実施し、仕様に問題ないことを確認した。

5.2 非同期連動処理

連動論理部における非同期連動処理に特化したシミュレータを作成し、その実現性を確認した。シミュレータでは伝送遅延の発生をシナリオ形式で設定可能とし、実際の条件設定でのシミュレーションが可能である。

シミュレーション事例として列車追跡機能の例を示す。車両追跡論理は、車両の移動に伴う隣接軌道回路の状態変化を参照することで、軌道回路の状態を不正落下と不正上を含む5状態で表現するものである。ここで、隣接軌道回路の落下から当該軌道回路の落下までの時間（進入時隔）が、既定値（ここでは0.9秒）未満の場合に、車両による短絡ではないと判断して不正落下とする論理があるが、この時間が0.5秒の不正な時に当該軌道回路の落下の情報が1秒遅延しても、見かけの時間によらず不正落下を誤りなく判断できることを確認した。このように、非同期連動論理処理が時間の判定が必要な論理を含めて問題なく適用できることを確認した。

5.3 全体構成のシミュレーション

クラウド型連動装置のコントローラ、連動論理部、現場端末、監視端末の機能を模擬するシミュレータを構築した。シミュレータは、コントローラや連動論理部の停止、ネットワークの分断や統合に伴うマスタ選定やセグメント再割り当てに関する動作のほか、連動論理部内の3値論理に基づく論理演算、非同期制御、制御情報と表示情報の授受、現場端末における制御情報の照査、現場機器の制御模擬（シナリオデータや制御情報に基づいた表示出力、制御情報のログ記録等）、監視端末による連動論理部停止の検出に関する機能と動作を模擬する機能を備える。また、ネットワーク上の通信遅延に対する動作への影響についても評価する機能を持つ。

セグメント再割り当て動作に関しては以下のとおりである。コントローラ配下の連動論理部の停止の検出と、

マスタコントローラによる稼働中の連動論理部上へのセグメントの割り当てを行い、セグメント割り当てデータを各コントローラに配信し連動論理部上で動作を開始する、一連の動作がシミュレータ上で行われている事を確認した。

6. まとめ

連動装置の論理処理をクラウド環境上に構築するクラウド型連動装置について検討した。提案したシステムはコントローラ・連動論理部・現場端末の3階層からなる。各連動論理部の処理周期をセグメントと呼ばれる単位に分割し、セグメント毎に連動論理処理を独立して行わせる。また、連動論理データはコントローラで保有し、処理対象の動的割り当てと連動論理データのアップロードをコントローラで行う。また、同一のデータを複数の連動論理部で処理させ、その結果を現場端末に集めることで連動論理処理の多重化を実現する。必要な処理量に応じて連動論理部増やせるスケーラブルな構成と連動論理処理の独立性が特徴となっている。また、伝送遅延の対策として非同期論理処理についても提案した。

なお、本報告では連動装置を対象としているが、個々の階層内で検討された処理方法を活用することで、他のシステムへの適用も可能であると考えられる。

連動装置のクラウド化の最終形は、連動論理部に汎用CPUを使う技術を確認することで実現すると考えている。この場合、連動論理部そのものがクラウド上に仮想化されると考えられるが、汎用CPUを用いてフェイルセーフを確保する技術については、引き続き検討を行う。

文 献

- 1) 奥谷民雄, 島添敏之: 鉄道信号用線区集中電子連動システムの開発と評価, 電気学会論文誌D, Vol.119, No.11, pp.1307-1314, 1999
- 2) 中野浩之: 加古川線 CTC 化 (線区集中連動方式), 鉄道と電気技術, Vol.15, No.11, pp.41-45, 2004
- 3) 大野聡: 九州新幹線 (博多・新八代間) 集中連動方式の開発と実用化, 鉄道と電気技術, Vol.24, No.8, pp.11-16, 2013
- 4) 西山淳, 国藤隆: ネットワーク信号制御システム—システム概要とモニターラン試験について—, 鉄道と電気技術, Vol.17, No.4, pp.28-31, 2006
- 5) A. Kaldenbach.: Die Digitale LST bei der DB Netz AG Eine Einführung in die Digitale Stellwerkstechnik, Fachvortrag BF Bahnen, 2019.
- 6) Siemens: Trackguard® Simis Interlocking as a Service (SIaaS), <https://www.mobility.siemens.com/ch/en/portfolio/>

- rail/automation/interlocking-systems.html (参照日：2020年8月17日).
- 7) 遠山喬, 福本聡：ネットワークを介した多数決を利用した鉄道信号システムの基礎的 FMEA, 電子情報通信学会技術研究報告, Vol.120, No.288, DC2020-67, pp.43-48, 2020
- 8) G. J. Holzman. : The SPIN Model Checker, Addison-Wesley, 2004.