

## 分散制御システムの間欠接続リンク用プロトコルに関する研究\*

有田 敬

電気通信大学 情報通信工学専攻†

高田 昌之

電気通信大学 情報基盤センター‡

## 1 序論

## 1.1 研究の背景

近年ネットワークは大きく発展し、分散処理環境が容易に構築可能となった。分散処理環境を構成する要素は多岐にわたり、その中には無線ネットワークを用いて接続する移動端末も含まれる。移動端末は基地局と通信可能な範囲の内外へ移動することで、ネットワークとのリンクの喪失と確立を繰り返す。この移動端末のようにネットワークとのリンクの状態が動的に変化するものを、間欠接続リンクと呼ぶことにする。通常、機器制御を目的とした分散処理環境で機器同士が連携し、求められる機能を実現している場合、機器は常時通信を行える必要がある。そのため、通信の失敗などが起きる間欠接続リンクは利用できない。

しかし、上位の制御階層からの命令に従って動作し、それ単体で必要な機能を実現できる機器が分散処理環境上にある場合を考える。この場合、この機器は他の機器群と常時通信できる必要はなく、通信ができない場面ではその事を見出し、状況に応じた自律動作が行える事が重要となる。そこで、このような機器には、間欠接続リンクが非常に有用であることから、間欠接続リンクに適したプロトコルが必要であると考えた。以上から、間欠接続リンクを分散処理環境で容易に利用できるよう、通信ができる事に見切りをつけ、機器本来の自律動作を促す事に主眼を置いたプロトコルを設計する。

## 1.2 研究の目的

間欠接続リンクを容易な利用を実現するプロトコルを設計し、間欠接続リンクを用いた応用システムを実現できるように分散システム上でプロトコルの実装を行うことを本研究の目的とする。本研究での基盤システムとして、高田研究室で継続して研究が行われている分散制御システム Glue Logic を用いる。

## 2 Glue Logic について

Glue Logic は製造用セル制御システムの構築を容易にするために設計された、複数のエージェントが並列に協調動作するための「場」を提供する制御用基盤システムであり、エージェント間で共有する情報を一元的に管

\*Study on Sporadic Link Protocol for Distributed Control System

†Takashi Arita, The University of Electro-Communications, Department of Information & Communication Engineering

‡Masayuki TAKATA, The University of Electro-Communications, Information Technology Center

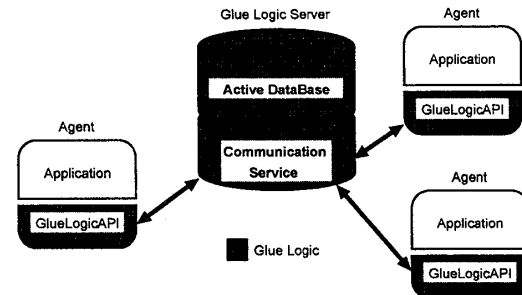


図 1: Glue Logic の構成

理する Glue Logic Server(以下サーバとする)と、Glue-LogicAPI(Application-ProgramInterface)を含んだ複数のエージェント群で構成される。ここで用いるエージェントとは、何らかの機能を持ったアプリケーションプログラムを指す。エージェント同士の通信は全てサーバの共有情報を介したメッセージのやり取りによって行われる。

サーバは、Glue Logic 内のデータが変更されたとき、変更通知要求を登録されているエージェントに対して、データが更新された事を通知する機能を持つ。これにより、エージェントは各々の処理に関連の深い情報を常に監視しておく必要がなくなるため、ネットワークのトラフィックを激減させ、更に必要最小限の CPU パワーでシステムを維持できる。この変更通知要求の登録にはエージェントに固有のエージェント識別子を用いる。

## 3 プロトコル設計

## 3.1 設計指針

基盤システムとなる Glue Logic の設計に合わせ、プロトコルの設計指針として既存の Glue Logic との互換性を維持すること、ネットワークのトラフィック、サーバの動作への影響を考慮し設計を行うことを定めた。

## 3.2 要求要件

以下を本研究での要求要件とした。

- エージェントはリンクの喪失の検出時に応答メッセージの受信を諦め、接続を切断し、対応する内部状態へ移行できること
- リンクが喪失した状況で、エージェントが意識することなくサーバへの再接続を行う事ができること
- 再接続を行ったエージェントがリンクの喪失以前と同一のエージェントとして動作できること

## 4 システムの実装

### 4.1 Glue Logic のマルチスレッド化

予備実験により、エージェントの再接続を待ち受ける際に、エージェントへのサービスが全て一時的に停止する問題があることが判明した。

そこで、問題がシステム全体に波及しないよう、サーバをマルチスレッド化し、エージェント毎に対応する通信スレッドを割り当てるようにした。実装には Java 言語を用いた。このサーバの概要を以下の図 2 に示す。

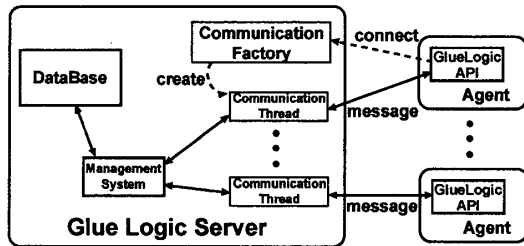


図 2: MT 化した Glue Logic

### 4.2 Java による API の設計、実装

システムの構築を容易にするため、サーバのマルチスレッド化に合わせて、Java による GlueLogicAPI の設計及び実装を行った。この API に対し、要求要件を満たすよう設計したプロトコルを実装した。

実装したプロトコルは、リンクの喪失によってユーザの設定した時間内に通信が行えない場合には、エージェントに対し通信が行えなかった結果を返すこととした。エージェントはこの結果を受け、対応する動作へ移行できるように記述する。また、エージェントが通信を望んだ時点でリンクが喪失していれば、API によって自動的に再接続作業を行われた後、要求されたメッセージのやり取りを行う事とし、通信路の確立などの作業に関して、開発者が意識する必要がないよう設計した。

### 5 サーバ間接続へのプロトコルの適用

Glue Logic では通常、高機能な単体のエージェントによってではなく、複数の簡単な機能を持ったエージェントの協調動作によって、高度な機能を実現する。

そこで、間欠接続リンクを利用したシステムを容易に構築できるように、間欠接続リンクを利用する機器で動作するエージェントに共有情報空間を提供することを考えた。その方法として、名前空間を複数のサーバ上に階層化したモデルに対しても適用し、エージェントだけでなくサーバ間の通信にも本プロトコルを適用できるように、プロトコルの拡張を行った。これを行う事で、共有情報空間の提供だけでなく、Glue Logic で標準的に容易されている機能や、既存のエージェント群を間欠接続リンクを利用する機器で利用できるようになる。

#### 5.1 多階層にわたる情報へのアクセス制限

情報が複数の場所に存在する場合に不可分動作を行うには、情報の整合性のために、一度の情報の要求に対する返答は全て同時に得られたものでなければならない。

そのため、階層化されたサーバにおいては、必要な情報が存在するサーバ全てにロックをかけ、一度に情報を取得する必要がある。しかし、これを間欠接続リンクを利用するサーバで行うと、図 3 のように間欠接続リンクを利用しないリアルタイム性の高いエージェントまでブロックされ、リンクの喪失の影響を受けてしまう。

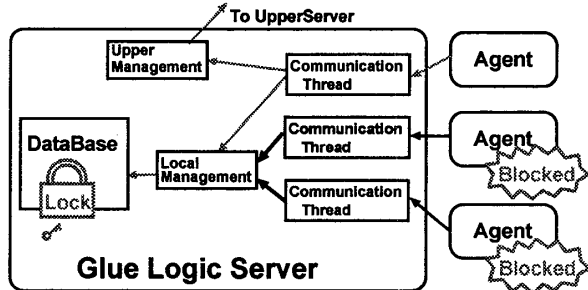


図 3: 上位へのアクセスによるエージェントのブロック

そこで、本プロトコルを利用する階層化されたサーバ間の接続では、複数のサーバにまたがった情報の要求は受け付けられないこととし、リアルタイム性の高いエージェントの動作を妨げないようにした。

## 6 システムの動作実験

実装したシステムを用い、間欠接続リンクで接続した異なる 2 つのホスト間でそれぞれサーバ・エージェント間、サーバ・サーバ間のプロトコルの動作実験を行った。行った実験は以下の通り。

- エージェントのリンク喪失時の動作、再接続時の環境復元
- 既存の Glue Logic の要素との互換性
- サーバ間接続でのアクセス制限、他のエージェントへの影響の検証

実験の結果、プロトコルは要求要件を全て満たしており、サーバの一時停止の問題も発生しない事を確認した。また、サーバ間接続ではアクセス制限が実現できており、間欠接続リンクを利用しないエージェントへの影響がない事を確認した。

## 7 結論

Glue Logic のマルチスレッド化、Java 言語による API の設計及び実装を行い、API の機能の一つとして、間欠接続リンク用プロトコルを実装した。サーバ間接続に対してプロトコルを適用し、システムの構築を容易に行えるようにした。また、これらの実装したシステムの動作実験を行い、期待した動作が行える事を確認した。

今後の課題として、より高速に動作できるサーバ内でスレッドとして動作するエージェントのための API の実装などが考えられる。

### 参考文献

- [1] 高田昌之, "分散型セル制御用基盤システムの実現", 1996 年度精密工学会 春期大会講演論文集 B-21, pp53-54
- [2] .Richard Stevens 著, 篠田 陽一 訳, "UNIX ネットワークプログラミング第 2 版 Vol.1", ピアソンエデュケーション, 2000