空間分散アクチュエーションモジュールの開発 —多数の分散アクチュエータを用いるための通信手法—

Development of Distributed Actuation Module -Communication Technique to Use Huge Number of Distributed Actuator-

○高木和貴（芝浦工大） 富沢哲雄（産総研） 正 谷川民生（産総研）
正 大場光太郎（産総研） 正 水川真（芝浦工大）
Kazutaka TAKAKI, Shibaura Institute of Technology
Tetsuo TOMIZAWA, Advanced Industrial Science and Technology
Tamio TANIKAWA, Advanced Industrial Science and Technology
Kohtaro OHBA, Advanced Industrial Science and Technology
Makoto MIZUKAWA, Shibaura Institute of Technology

Ubiquitous robots consist of distributed and networked robot elements such as sensor, computer, actuator, and etc. Distributed actuator is one of the ubiquitous robot elements, which provides physical services. We are developing an actuation module which is distributed easy form. Here, communication channel between actuators and controller need to be improved in more easy dispersion of actuators. Wireless communication can distribute actuator to space easy. When huge number of actuators is distributed, necessary communication will be obstructed by communication traffic. And operating many actuators is over capacity for controller. This paper solves communication problem using by multicasting communication and the time sync system.

Key Words: Ubiquitous Robot, Distributed Actuator, cooperative action, Wireless Communication

1. はじめに

ユビキタスロボットはセンサ、コンピュータ、アクチュエータといったロボット機能要素を空間上に分散させたロボットである[1]。小型のロボット要素を分散し、それぞれをネットワークにより相互接続することで、空間そのものをロボットとしてサービスを行う。これまでセンサネットワークや分散コンピューティングなどは実用化に向けた研究が進んでいる[2]。これらの技術に加えて、アクチュエータ要素をセンサやコンピュータと同様に分散配置できれば、ユーザが自由にアクチュエータを配置でき、技術に偏らない真のニーズに応じたロボットシステムとすることができる。また、大まかなタスクごとに分散アクチュエータを用意し、それら独立性が高い部品として提供することにより、要素や機能の追加、変更が容易となる。センサ・コンピュータ・アクチュエータの3つの要素がモジュールとして揃えば、ロボットの段階的な導入が可能となり、生活支援ロボットを家庭内にも普及しやすいと期待できる[3]。

分散アクチュエータには、アクチュエータ・コントローラ・通信機能が必要となる。我々はこれらの基本的な要素をまとめてモジュール化することで図1の様に汎用性の高い基盤技術を提案してきた[3]。この分散アクチュエータはコンピュータ等で構成されるマスタモジュールから指示を受け動作する。分散アクチュエータとマスタ間においては相互の通信が不可欠であり、通信路やプロファイルについても、分散的かつ汎用的なものを開発する必要がある。

そこで本稿ではアクチュエータの多数使用による通信帯域の圧迫、通信路でのディレイと欠損について取り上げ、2章で解決する。3章では本稿の通信手法に必要な同期について述べ、その手法を提案する。4章はまとめである。

2. 通信法

分散アクチュエータは通信や電源の確保が欠かせないが、配線によって設置場所を制約されるべきではない。そのためにはセンサネットワーク等と同様に無線通信を用いる。アクチュエータを制御するためには、センサに比べより高度な通

Fig.1 Distributed Actuation Module

信の速度、確実性が要求され、一方で通信遅延が存在すると全体の協調作業に支障をきたす。そこで以下に分散アクチュエータの通信における問題点を挙げ、解決方法を検討する。

分散アクチュエータはそれぞれが1つのマスタモジュールから指示を受けて動作する。マスタモジュールは複数のアクチュエータを統括制御している。以降の説明ではマスタモジュールから分散アクチュエータへの指令値を送ることを指示と呼ぶ。アクチュエータの内部パラメータを必要に応じてマスタモジュールに送ることをフィードバックと呼ぶ。

2.1 多数のアクチュエータへの通信

一般に、センサネットワークの分野ではリアルタイム性はさほど重視されないが、アクチュエータをフィードバック制御するためには最低でも数十ms単位でのリアルタイム性が求められる。複数のアクチュエータに対して、マスタモジュールが高周期に個別の命令を送信しようとする、ネットワークは必然的に通信量が増加する(図2)。だが、帯域の広い通信手法は消費電力や大きさの面から分散アクチュエータに適用できない。アクチュエータを分散させて用いるには、多くのアクチュエータを動作させても通信帯域を圧迫させない仕組みが必要である。

無線通信は送信側が空間上にブロードキャストした情報を選択的に受信することで、ユニキャストやマルチキャストを実現している。そのため、時空間上の情報量は宛先の数ではなく、送る情報と回数に依存する。送信回数を減少させれば、通信量を削減できる。通信量を大幅に削減するためには、マスタから下位の分散モジュールの送信回数を減らすことが効果的である。そこで、マスタモジュールは全ての分散モジュールに対して同一の抽象的な命令を同時に送信する。各分散モジュールには、命令文を解釈し自己位置や他のモジュールとの関係を考慮して適切な動きを生成するインテリジェンスを持たせる。具体的には、分散型の動輪モジュールが取り付けられた台車を動かす場合、それぞれの動輪モジュールに対して個別の命令を指示する（"Motor 0:0.14m/s", "Motor1:0.11m/s", "Motor2:0.08m/s", "Motor3:0.02m/s"）よりもマルチキャストにより"Cart: forward0.10m/s"と指示した方が通信回数を少なくできる(図3)。この手法においては、同じ象徴化された指示で動作できるものであれば、アクチュエータが増えてもマスタ側の指示の出し方を変更する必要はない。さらに情報の絶対量を減らすことができるので、低ビットレートな通信手法を用いることができ、小型、低消費電力化を図ることができる。

2.2 マスタへの通信

複数の分散アクチュエータを用いて、協調作業を行おうとした場合、マスタはシステム全体の状態を知るために各分散モジュールから情報を逐次収集する必要がある。したがって、分散アクチュエータの数に比例して通信量が増加してしまう。通信量の増加は、通信遅延や情報欠落といった障害を引き起こす。一方で、複数のモジュール間に機械的な拘束が存在する場合には、必ずしも全てのモジュールから全ての情報を収集する必要の無いケースも多く存在する。

そこでフィードバックはベストエフォート型通信とみなしてシステムを構成する。つまり通信が行えないことを前提とし、通信を行えた中で最も確からしい情報をもとに制御を行う。フィードバックされる値はユビキタスロボットのシステム内のパラメータの1つであり、それぞれの値が完全に独立していることは多くない。得られた値の中から最も確からしいパラメータを推定して用いればよい(図4)。

通信帯域に対して非常に多くのアクチュエータがある場合は、マスタがアクチュエータからフィードバックされる情報量を規制する。これには情報の精度を利用する。アクチュエータから送信される情報はアクチュエータが置かれた状況によりその精度が異なる。マスタはアクチュエータに対し、精度の閾値を提示する。アクチュエータは自身が送信する情報の精度と閾値を比較し、情報の精度が閾値を上回った場合のみ送信する。精度が閾値を下回った場合は、マスタはそのアクチュエータからの情報がなくても十分に動作できるということである。アクチュエータが送信を打ち切っても問題はない。

また、パラメータによっては単体の分散アクチュエータのみで有効な値を推定できることがある(図4では動輪モジュールが取り付けられた台車の速度を、それぞれのモジュールが推定している)。その場合は分散アクチュエータ側でパラメータを推定し、その値を送信する。マスタモジュールは、個々の推定済みのパラメータから平均などの簡単な処理で真値を推定する。こうすれば分散アクチュエータが増えても、マスタモジュールはパラメータ推定の処理が殆ど増えない。パラメータ推定以外の処理は変更の必要がないため、アクチュエータが増減しても全くシステム構成を変更せずに対応できる。

Fig.2 Command by Unicast

Fig.3 Command by Multicast

Fig.4 Estimate of the System Parameter

Fig.5 Delay Effect on System

Fig.6 Constant Delay

2.3 通信のディレイ

近年の屋内無線通信はマルチホップなど複雑な通信経路をとるものが多い。複雑な通信経路をとるものは障害等に柔軟に対応できる半面、状況により通信のディレイは大幅に変動することがある。

分散アクチュエータはユビキタスロボットの物理的な出力を担う要素としてシステムの制御ループに組み込まれている。単純にマスタからの指示をアクチュエータに反映するだけの

通信では、ディレイはそのまま制御の安定性に影響を及ぼす。遅延量が変動するとシステムの応答性も変動することになり、安定なシステムを設計することは難しい(図5)。システムの応答性をディレイに依存させない仕組みが必要となる。

そこで、各通信プロトコルにはタイムスタンプをつけることとする。アクチュエータへの指示には、実際にコマンドが実行されるべき時刻を付加する。一方で、アクチュエータからフィードバックする場合はセンサ情報を観測した瞬間の時刻を付加する。マスタは送られた情報とスタンプされた時刻に基づき次の指示値を決定する。このようにすることで制御ループ内の通信遅延に対してロバストなシステムを構築できる(図6)。

センサ情報のディレイがなくなることはパラメータの推定にも役立つ。2.2節で述べたようにマスタへフィードバックした情報はパラメータ推定に用いられる。その際にタイムスタンプがあれば、情報を取得時間ごとにソートできる。ソートされた情報に時間窓関数を適用すれば、移動平均などのセンサ処理やパラメータ推定によく用いられる値が取得できる(図7)。

2.4 通信の確実性

ユビキタスロボットは対人サービスを行うことが期待されている。そのためアクチュエータの動作には高い信頼性が要求される。アクチュエータの動作は通信を介して制御されるので、通信も確実なものでなくてはならない。しかし信頼性の高い通信手法はコスト・大きさ・消費電力などの点で不利である。分散アクチュエータに信頼度の高い通信手法を用いるのはこれらの欠点のため難しい。そこで通常の通信手法とタイムスタンプを合わせて利用することで確実な通信を行う。

アクチュエータへの指示には実行されるべき時刻がスタンプされている。ここで指示が送信される間隔を決めておけば、指示が送られていない時刻が判明する。このとき再送要求を行うかどうかはアプリケーションによるが、要求を行えば通信の確実性を高めることができる。要求しない場合でも、通信が不安定であることをアクチュエータ側が把握することができる。

さらにこの考えを拡張すれば通信をあらかじめ行い、指示をスタックしておくことも可能である。指示はタイムスタンプどおりに実行すればよいので、前もって数回分の指示をスタンプと共に送ることができる。アクチュエータ側では送られた指示をタイムスタンプどおりにソートして順次実行すればよい(図8)。

一方で無線通信は原理的に通信を完全に保証できないため、通信が行えなかった場合に対応も必要となる。この方法は強くアプリケーションに依存するため、唯一解はない。状況によりアクチュエータにブレーキをかける、フリーにするなどの対応をとる。致命的な破壊に至らないよう、十分なリスクアセスメントを行う必要がある。

3. 同期法

通信遅延を完全に排除することはできない環境で、複数の分散アクチュエータを協調作業するためには、時刻付きの指示が有効であることは2章で述べた。ただし、タイムスタンプを利用するためには、お互いが同じ時計をもっていることが前提条件となる。本章では、複数の分散アクチュエータに同期した時刻を持たせるための方法について述べる。

Fig.7 Utilize of Time Window

Fig.8 Sort Command by Time

Fig.9 Method for Syncing Timer

3.1 タイマ

通常のロボットにおいて制御周期は1ms~100ms程度である。同期用の時計の精度は1ms以下でることが望ましい。分散アクチュエータに使われるマイコンのタイマは1ms程度の精度を十分にもつ。これらのタイマを同期用に用いればよい。

3.2 時刻同期機能

時刻を同期するには電波時計を用いる方法やGPSを用いた方法がよく用いられる。通信モジュールを介したNTPの利用も考えられる。しかし電波時計やNTPは1msの精度を出すことが難しい。時刻同期のみに用いるにはGPSは高価である。そこで我々はビーコンと通信を複合した手法を提案する。

提案する手法を図9に示す。全ての分散アクチュエータにOOK(On Off Keying)受信モジュールを搭載する。マスタから送信モジュールを介してビーコンを周期的に送信する。このときビーコンはマルチホップなどをせず、送信モジュールから受信モジュールに直接送られるものとする。アクチュエータはビーコンを受信するとそれをトリガにしてタイマをクリアする。その後分散アクチュエータに搭載した通信モジュールからトリガの時刻を受信する。タイマの値とトリガの時刻を足すことにより現在の時刻が取得できる。

この手法の特徴は他の手法に比べ低コストである点である。通信モジュールは分散アクチュエータに搭載されているもの

を用いるので追加コストはない。追加するのは OOK 受信モジュールおよびタイマ用のマイコンのみでよい。受信モジュールもタイマの倍精度である 0.5ms が読み取れればよく、2kbps 程度のものでよい。

3.3 同期機能の実装

上述の同期法に基づき、同期モジュールを設計しアクチュエータに内蔵して用いる。これを図 10 に示す。モジュールは OOK モジュールとマイコンにより構成される。アクチュエータ間での同期ずれを防ぐため、OOK モジュールには複雑な通信プロトコルが付加されていないものを用いる。このような OOK モジュールは 5\$ 程度で市販されている。マイコンは 8bit マイコンの PIC もしくは PsoC を用いる予定である。分散アクチュエータ用コントローラとの通信は、シリアル通信もしくは I2C で行う。本モジュールが有効に機能すれば 1ms の精度で全アクチュエータおよびマスタを同期できる。

Fig.10 Sync Module

4. おわりに

分散アクチュエータにおける通信と同期法について述べた。これらの手法を用いることにより複数のアクチュエータを同時に動作させることが可能となる。今後はアクチュエータの実装と共にこれらの手法の有効性を確認する。

文 献

- [1] 大原, 金, 谷川, 大場, 域検知機能を有する空間機能モジュールの開発, Robomec2007 論文集, 2A2-J02
- [2] 菅原, 李, 金, 大原, 大場, RT ミドルウェアを用いた複数サービスに対応するフレームワークの提案, SICE SI2006 講演論文集, 1B3-5
- [3] 高木, 富沢, 谷川, 大場, 水川, 分散アクチュエータの運用環境とアクティブ・キャストの実装, SICE SI2007 講演論文集,