

Title	様々なデータを扱えるストリーミング技術を用いて機器間の通信・制御を行う研究
Author(s)	明壁, 祐基
Citation	
Issue Date	2004-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/1806
Rights	
Description	Supervisor:丹 康雄, 情報科学研究科, 修士

修 士 論 文

様々なデータを扱えるストリーミング技術を用いて
機器間の通信・制御を行う研究

北陸先端科学技術大学院大学
情報科学研究科情報システム学専攻

明壁 祐基

2004年3月

修 士 論 文

様々なデータを扱えるストリーミング技術を用いて
機器間の通信・制御を行う研究

指導教官 丹 康雄 助教授

審査委員主査 丹 康雄 助教授
審査委員 篠田 陽一 教授
審査委員 日比野 靖 教授

北陸先端科学技術大学院大学
情報科学研究科情報システム学専攻

210001 明壁 祐基

提出年月: 2004 年 2 月

概要

本論では、従来のクライアントから要求を受けるたびにサーバーが答えを返すリクエスト - レスポンス型の通信ではなく、クライアントが一度情報を要求すると、サーバーが常にリアルタイムな情報を流し続けることによってクライアントが必要な情報を継続して受信する、ストリーミング配信型の通信を用いて、映像・音声のみではなく、センサ情報や機器の制御情報を配信することで、無線 LAN で接続された機器間がリアルタイムな情報を利用し、そのときに最適な機器制御を構成し、協調動作を可能にすることを提案する。

目次

第1章	はじめに	1
1.1	背景と目的	1
1.1.1	ユビキタス環境	1
1.1.2	従来の機器間の通信	2
1.1.3	ストリーミング配信型の機器間の通信	2
1.1.4	研究の目的	3
1.2	本論文の構成	3
第2章	ストリーミング	5
2.1	ストリーミングとは	5
2.2	IP ネットワーク上でのストリーミング	6
2.2.1	UDP と TCP	7
2.2.2	RTP	8
2.2.3	RTP のパケットフォーマット	9
2.2.4	RTP 定義のペイロードタイプ	11
2.2.5	RTCP	12
2.2.6	RTCP が提供する機能	12
2.2.7	RTCP パケットの種類	13
2.3	一般的なストリーミングでの RTP の使い方	14
2.3.1	映像データの場合	14
2.3.2	音声データの場合	16
2.4	ストリーミングの基本的な問題点	18
第3章	ストリーミング技術を用いた機器間の通信・制御	20
3.1	ストリーミング技術を用いる利点	20
3.2	ストリーミング技術を用いる問題点	21
3.2.1	センサ情報や機器の制御情報をストリーミング配信する方法	21
3.2.2	輻輳問題を解決する方法	21
3.3	想定する環境のシステム	22
3.3.1	構成	22
3.3.2	想定する自走可能な掃除ロボットの動き	23
3.3.3	想定する協調動作	24

第 4 章	センサ情報及び機器の制御情報のストリーミング方式	26
4.1	ストリーミングを用いた機器間の通信・制御で扱うデータ	26
4.1.1	センサ情報	26
4.1.2	機器の制御情報	27
4.2	テキストデータとしてストリーミング	27
4.2.1	既存のテキストデータストリーミング方式	29
4.2.2	新たに提案するテキストデータストリーミング方式	31
4.3	センサ情報や機器の制御情報を XML を用いて記述	33
4.3.1	XML とは	33
4.3.2	XML を用いて記述する利点	33
4.3.3	想定する環境のシステムでの XML 記述	33
4.3.4	XML 記述テキストデータでの RTP パケットのデータ量	38
4.4	提案したストリーミングシステムの手順	38
第 5 章	アプリケーションレベルでのアドミSSION制御	40
5.1	アドミSSION制御を行う意義	40
5.2	場合分け	41
5.2.1	情報の変化頻度	41
5.2.2	情報の重要性	42
5.2.3	想定する環境のシステムでの場合分け	43
5.3	クラス分け	46
5.3.1	想定する環境のシステムでのクラス分け	46
5.4	ポリシーに従ったアドミSSION制御の手順	47
5.4.1	想定する環境のシステムでの手順	47
第 6 章	評価	50
6.1	評価環境	50
6.1.1	ns-2	50
6.2	ポリシーに従ったアドミSSION制御の評価	51
6.2.1	ポリシーに従ったアドミSSION制御を行わない場合	51
6.2.2	ポリシーに従ったアドミSSION制御を行う場合	52
6.2.3	測定方法	53
6.2.4	測定結果	54
6.3	シミュレーション結果についての評価と考察	59
第 7 章	おわりに	60
7.1	まとめ	60
7.2	今後の課題	60

目次

1.1	リクエスト - レスポンス型の通信	2
1.2	ストリーミング配信型の通信	3
2.1	ダウンロードとストリーミングの違い	6
2.2	RTP のプロトコルスタック	7
2.3	映像ストリームの送信の仕組み	9
2.4	RTP パケットの内容	10
2.5	H.261 におけるピクチャの分割方法	15
2.6	H.261 における映像の RTP パケット化の考え方	16
2.7	G.723.1 における音声の RTP パケット化の考え方	17
3.1	想定する環境のシステム	23
4.1	温度センサ機構のデータ変換プロセスの例	28
4.2	RFC2793 が想定するテキストストリーミングの例	30
4.3	RFC2793 で定義された RTP パケットの内容	30
4.4	新たに提案するテキストデータストリーミングのプロトコルスタック	32
4.5	新たに提案するテキストデータストリーミングの RTP パケットの内容	32
4.6	想定する環境のシステムでの XML の構造	34
4.7	想定する環境のシステムでの XML の構造の DTD	34
4.8	一例として作成するシステムでの温度情報の XML 記述例	36
4.9	一例として作成するシステムでの位置情報と制御情報の XML 記述例	37
4.10	提案したテキストデータストリーミングでの 1 パケットのデータ量	38
4.11	提案したストリーミングシステムの手順	39
5.1	情報の変化頻度の違い	42
5.2	情報の重要性の違い	43
5.3	想定する環境のシステムでの場合分け	43
5.4	想定する環境のシステムでのクラス分け	47
5.5	想定する環境のシステムでのポリシーに従ったアドミッション制御	49
6.1	毎秒ごとに自走可能な掃除ロボットが受信するデータ量の違い	55

6.2	毎秒ごとに自走可能な掃除ロボットが受信する情報数の違い(位置・制御情報)	56
6.3	毎秒ごとに自走可能な掃除ロボットが受信する情報数の違い(人の動き検知情報)	56
6.4	毎秒ごとに自走可能な掃除ロボットが受信する情報数の違い(温度情報)	57
6.5	毎秒ごとに自走可能な掃除ロボットが受信する情報数の違い(照度情報)	57
6.6	毎秒ごとに自走可能な掃除ロボットが受信する情報数の違い(音情報)	58

表 目 次

2.1	RTP 定義のペイロードタイプ一覧	12
3.1	想定する環境のシステムに存在するセンサ及び機器	23
4.1	想定する環境のシステムでセンサ情報を提供するセンサ及び機器	27
4.2	想定する環境のシステムで制御情報を提供する機器	27
4.3	想定する環境のシステムで対象となるテキストデータ	29
6.1	シミュレーションを行った PC のスペック	50
6.2	シミュレーションの内容 (ポリシーに従ったアドミッション制御を行わない場合)	52
6.3	シミュレーションの内容 (ポリシーに従ったアドミッション制御を行う場合)	54

第1章 はじめに

近年，ネットワーク，計算機，携帯端末などの進歩により，様々な機器が計算機能を備えネットワークに接続されることで，「いつでも，どこでも，誰とでも」情報をやりとりできるユビキタス環境 [1] が実現されつつあり，その環境を前提とした研究を多くなされている．

従来，ユーザが機器やサービスを利用するためには，ユーザは利用したい機能を備えた機器のある場所まで出向く必要があった．しかし，ユビキタス環境では，ユーザや機器の位置や嗜好，様々なセンサ情報などを利用することで，ユーザや機器の状況や環境に基づいて動的に機器を構成し，サービスを提供することが考えられる．

このような環境において，刻々と変化する空間のリアルタイムな状況や環境を把握をするために，様々なセンサ情報をリアルタイムに通信できる技術が必要となる．

一方では，ネットワーク技術の進歩に伴い，テレビ番組やラジオ番組の生放送や各地に設置されたライブカメラからのライブ映像などを IP ネットワークを通じて楽しむことが出来る，ストリーミング技術が開発され，注目を浴びている．ストリーミング技術は，映像や音声などの情報をリアルタイムに配信することができる．

本論では，ストリーミング技術がリアルタイムな情報の配信を可能にする点に注目し，空間に存在するセンサや機器から得られるセンサ情報や機器の制御情報など，様々なデータを扱えるストリーミング技術を用いて，刻々と変化する空間のリアルタイムな状況や環境の情報をその情報を必要とするユーザや機器に，リアルタイムにストリーミング配信することで，機器の協調動作を実現できると考える．

1.1 背景と目的

1.1.1 ユビキタス環境

ユビキタス環境の特徴のひとつとして，空間に小型 PC やセンサが埋め込まれており，ユーザや機器が小型 PC やセンサを持っていることがあげられる．この特徴を生かすことで，今まで独立して動作していたデバイス同士が結びつくことによる協調動作が可能になる．このようなユビキタス環境において，刻々と変化する空間のリアルタイムな状況や環境を把握をするために，空間に埋め込まれた小型 PC やセンサからの様々な情報をリアルタイムに通信できる技術が必要となる．

しかし，各空間を構成する機器やセンサ，ユーザが保持する小型 PC やセンサはそれぞ

れ異なる．よってそれらの違いを吸収し，統一した通信方法を提供する技術が必要になる．

1.1.2 従来の機器間の通信

今までに，一方の機器が要求を出すたびにもう一方の機器が答えを返す，リクエスト - レスポンス型の通信については盛んに研究がなされている．リクエスト - レスポンス型の通信では，クライアントは必要となる情報を得たいとき，サーバに要求を出し，サーバが要求に見合った情報を送信するといった2ステップの通信を毎回行う必要がある．

また，通信を行うたびにコネクションを確立するための応答確認を行うので，通信の確実性は高まるが，通信のリアルタイム性は損なわれる場合がある．リクエスト - レスポンス型の通信について，図 1.1 で示す．

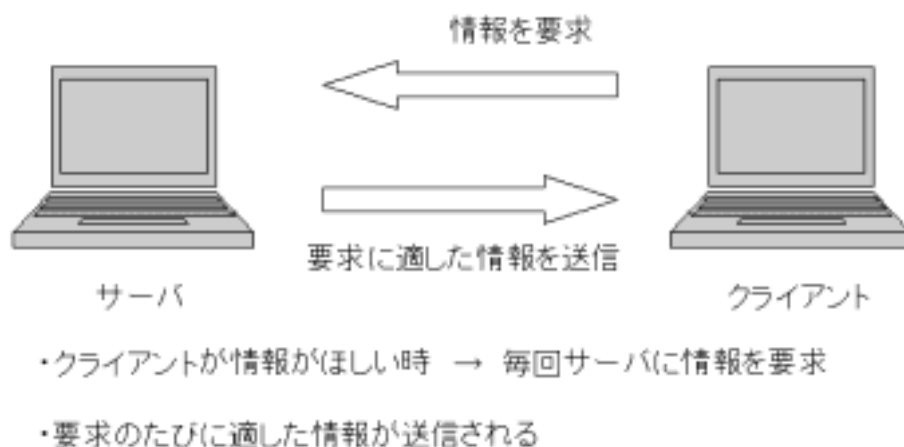


図 1.1: リクエスト - レスポンス型の通信

1.1.3 ストリーミング配信型の機器間の通信

リクエスト - レスポンス型の通信に対して，ストリーミング技術を用いたストリーミング配信型の通信を用いて機器間の通信・制御を行うことが考えられる．

クライアントが一度，情報の配信を要求すると，サーバが情報を連続して配信し続けることによって，クライアントは情報を受信し続けるストリーミング配信型の通信では，通信を行うたびにコネクションを確立しないので，通信の本質的な遅延を省くことができる．そのため，情報を必要とする機器は常にリアルタイムな情報を取得できる．

その結果，機器はサーバから配信され続けるリアルタイムな情報を利用することによって，その時々で最適な動作または機能を計画することができ，機器の協調動作が可能になる．ストリーミング配信型の通信について，図 1.2 で示す．

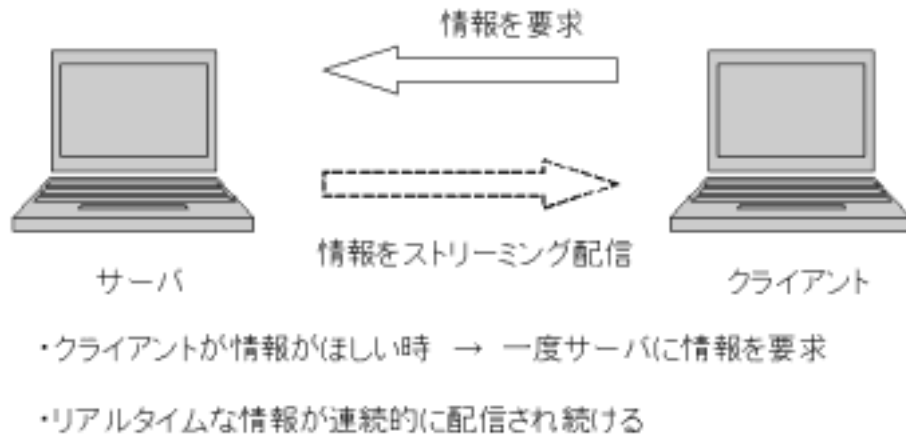


図 1.2: ストリーミング配信型の通信

1.1.4 研究の目的

本論では、機器が無線 LAN で接続された室内などの限られた空間を対象として、映像・音声のみではなくセンサ情報や機器の制御情報をストリーミング配信型の通信を用いて扱うことで、機器間でのリアルタイムな情報の配信、及び協調動作の実現を目的とする。

ストリーミング配信は、送信側の各センサや機器から、受信側の機器へ一方的にリアルタイムな情報を流し続ける場合のみを想定する。受信側の機器は連続的に送られてくるリアルタイムなデータを自身で処理することで、動的な動作の計画を実現することができる。

室内などの限られた空間を対象とすることで、存在する機器の数が制限される。室内などの空間に存在する機器間をつなぐ近距離無線通信技術としては、802.11 無線 LAN、Bluetooth、IrDA、PHS などが既存の技術として考えられる。本論では、室内などの空間に存在する機器は IP ネットワークを構築されている場合を前提とする。そのため、空間に存在する機器が IP ネットワークを構築するための一般的な近距離無線通信技術として、無線 LAN を用いることとする。

1.2 本論文の構成

本論の構成は以下の通りである。第 2 章では一般的なストリーミング技術について述べる。第 3 章では、機器間通信・制御にストリーミング技術を用いる意義を説明する。ストリーミング技術を機器間の通信・制御に用いる上での利点と問題点をあげ、その解決法を提案する。また、具体的な例として提案した、想定する環境のシステムについて説明する。第 4 章では、新たに提案したセンサ情報や機器の制御情報をテキストデータとしてストリーミング配信する方法についてを説明する。第 5 章では、第 3 章であげた輻輳問題の

解決法として，ポリシーに従ったアドミッション制御について提案方法を説明する．第6章では，想定する環境のシステムを用いて，各センサや機器からのセンサ情報及び機器の制御情報の同時ストリーミング配信時における，ポリシーに従ったアドミッション制御の有効性について，シミュレーションを行い，結果を示す．結果について考察を行う．最後に第7章では，まとめと及び今後の課題について述べる．

第2章 ストリーミング

本章では、一般的なストリーミング技術について説明する。また、ストリーミング配信の基本的な問題点について説明する。

2.1 ストリーミングとは

ストリーミングとは、映像や音声などのデータをダウンロードしてから再生するのではなく、映像や音声などのデータを受信しながら同時にリアルタイムに再生する方式である。

従来、映像や音声などのデータを再生するためには、映像や音声などのデータをすべてダウンロードして受信するまで待たなければならなかった。ストリーミングでは、送信側が映像や音声などのデータを複数のパケットに分割して送信し、受信側は分割された映像や音声などのデータを受け取ると同時に再生することができるので、リアルタイムに映像や音声などのデータを再生できる。ダウンロードとストリーミングの違いについて図 2.1 で示す。

また、ライブ中継の映像や音声をリアルタイムで符号化し、そのままストリーミング配信することも可能である。ライブ中継のため、サーバ側はマスターとなる動画や音声クリップを持たない。したがって、ダウンロードしてからの再生はできず、クライアント側はストリーミング再生方式でコンテンツを再生することになる。

その他のストリーミングの特徴としては、受信した映像や音声などのデータを保存をしなくてもよいため、流れてきた映像や音声などのデータは順次利用され、そのまま保存されることなく消えていく。これにより、機器にデータが残らないといったことがあげられる。

ストリーミング技術を利用する一般的なアプリケーションソフトとしては RealOnePlayer[2]、WindowsMediaPlayer[3]、QuickTimePlayer[4] などがある。

・ダウンロード



・ストリーミング



図 2.1: ダウンロードとストリーミングの違い

2.2 IP ネットワーク上でのストリーミング

本論では、機器が無線 LAN で接続された室内などの限られた空間を対象として、映像・音声のみではなくセンサ情報や機器の制御情報をストリーミング配信型の通信を用いて扱うこと目的としており、無線 LAN で接続された機器は IP ネットワークを構築していることを前提としている。

IP ネットワークでは、おもに UDP/IP 上で動作する RTP(Real-time Transport Protocol)[5] と RTP の補助を行う RTCP(RTP Control Protocol)[5] を用いてストリーミングを行う。RTP ではコネクションの概念がないので、UDP/IP 上の場合、下位層のプロトコルである UDP に従いコネクションレス型の通信を行う。RTP を用いたストリーミングのプロトコルスタックは図 2.2 のようになっている。

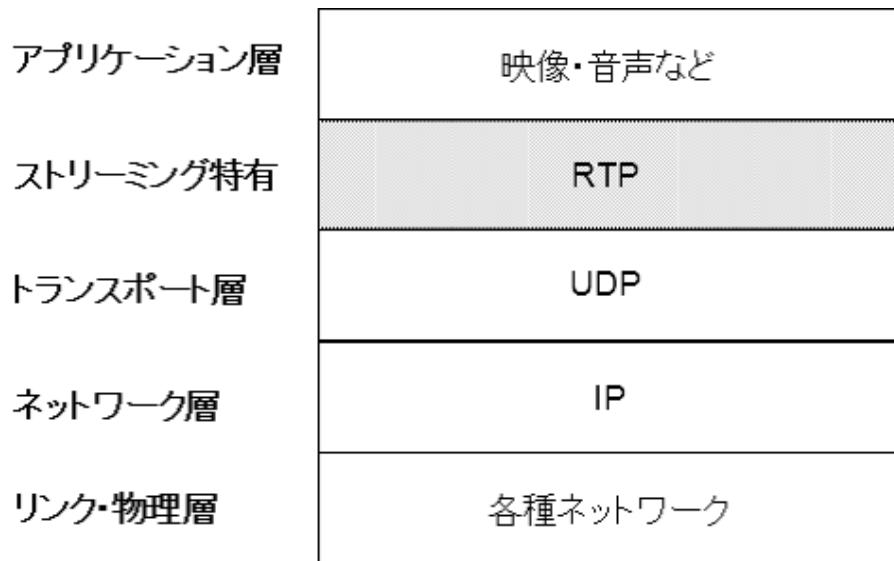


図 2.2: RTP のプロトコルスタック

2.2.1 UDP と TCP

UDP はコネクションレス型のトランスポートプロトコルである。UDP の特徴としては、以下のような特徴があげられる。

UDP の特徴

- 一方的にデータを送るだけで、必ずしも応答を要求しないような通信や（少々エラーが発生してもよいので）なるべく素早く相手にデータなどを届ける通信のために用いる。
- プロトコルのオーバーヘッドが小さい。接続の確立処理などを行うことなく、すぐにデータ送信ができる。応答確認やエラー検出なども行わない。
- 信頼性が低い。エラーが発生しても再送処理などは行われぬ（必要ならば上位のアプリケーションで対応する）
- 1対多のブロードキャストや、信頼性よりもリアルタイム性が要求されるような通信に適する。

UDP と対をなすトランスポートプロトコルとして、TCP がある。TCP はコネクション型のトランスポートプロトコルで、以下のような特徴があげられる。

TCP の特徴

- 2つのアプリケーション間で確実にデータを（双方向で）やり取りするために使う。

- プロトコルのオーバーヘッドは大きい．通信する相手との接続を確立してから通信を開始する．また応答確認やエラー検出，再送，エラー訂正処理などを行う．
- 信頼性が高い．エラー検出や，エラー訂正処理などが行われる．
- データを間違いなくやりとりする必要がある通信に適する．

以上のような UDP と TCP の特徴から，IP ネットワークでは，リアルタイムな通信を行うために，UDP 上で RTP を用いてストリーミングを行う．また，UDP と TCP の違いが，第 1 章の図 1.1 と図 1.2 で示した，リクエスト - レスポンス型の通信とストリーミング配信型の違いの根源となっている．

2.2.2 RTP

RTP は，IP ネットワーク上で，映像や音声などのリアルタイム性のあるデータを，end-to-end に転送するためのプロトコルである．RTP は映像や音声などを運ぶために，データを識別する共通の方法とパケットのタイムスタンプのつけ方を定義している．また，映像や音声などのデータの圧縮方式ごとにペイロードフォーマットが定義されている．RTP には以下のような特徴がある．RTP を用いたストリーミングの仕組みの例として，映像情報の場合を図 2.3 で示す．

RTP の特徴

パケット化

映像符号化データ（アナログの映像信号をデジタル信号に変換したデータ）や音声符号化データを，パケット転送に適した長さのブロックに分割する．

パケット破棄対策

例えば，前のパケットが破棄されてもデータを復号できるようにするための情報を，パケット・データに付加する．この情報の構造は圧縮符号化方式ごとに定められている．

ジッタ吸収

RTP の送信側と受信側は同じ速度で進むクロックを備える．送信側は，データ送信時のクロックのカウント値をタイムスタンプとして付加しながら，データを送信する．受信側は，受信したデータを復号する際に，データ復号時のクロックのカウント値とタイムスタンプ値との差が一定となるようにデータの復号時刻を調整する．この仕組みによりジッタを吸収できる．

メディア間同期

タイムスタンプと RTCP からのデータを利用して、メディア間同期を保つ必要がある映像ストリームと音声ストリームとの対応関係を認識し、タイムスタンプにより対応するストリーム間の時間合わせを行う。

UDP による転送

RTP パケットは UDP/IP を用いて転送する。そのため、時間を厳守した配信や、サービスの品質を保証するメカニズムはない。それらはアプリケーションで実現する必要がある。

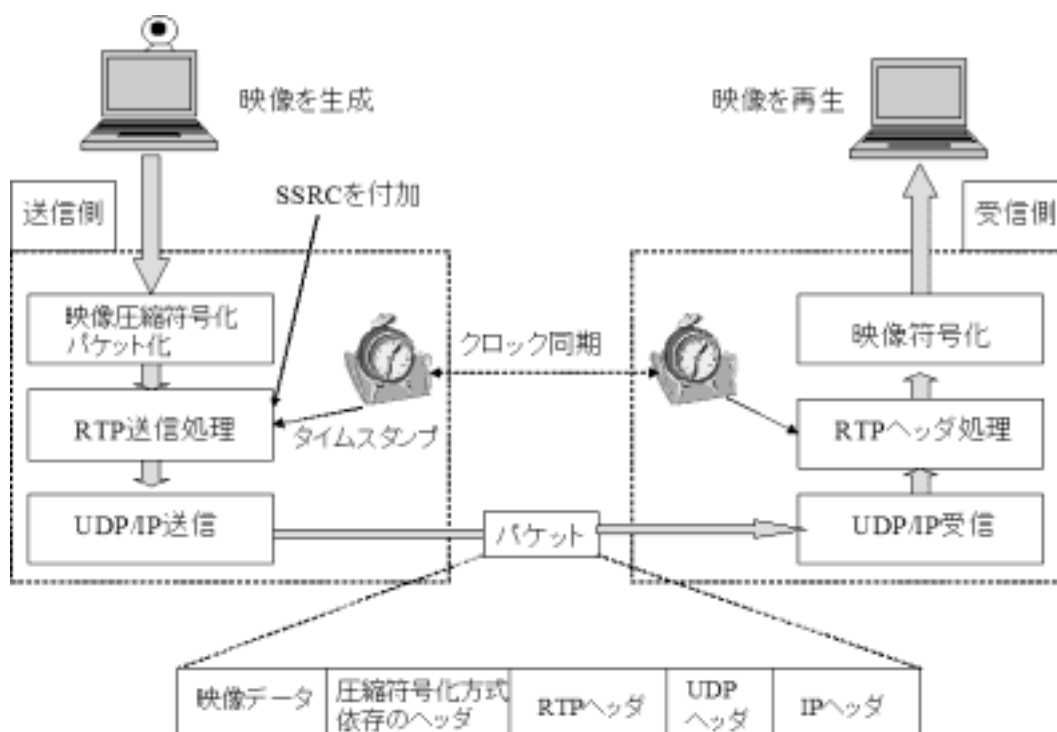


図 2.3: 映像ストリームの送信の仕組み

2.2.3 RTP のパケットフォーマット

RTP は、その全メッセージで同じフォーマットを使用する。アプリケーション層フレーム化をサポートするために、このメッセージフォーマットによって多様な解釈ができ、特定のアプリケーションで必要とする追加を行うことができる。

RTP ヘッダは、映像や音声などのデータを同期化して表示し、パケットが欠落していないか、順序どおりに到着したかを定めるのに必要なタイミング情報を提供する。また、RTP ヘッダはペイロードタイプを含み、複数のデータや圧縮タイプに対応することができる。

RTP パケットは、12 バイトのヘッダとそれに続くペイロード（ビデオフレームまたは音声 f フレームなど）で構成される。ペイロードには、符号化の種類によって、特有の拡張が含まれる場合もある。RTP パケットの内容を図 2.4 で示し、その内容について説明する。

v=2	P	X	CSRC カウント	M	ペイロードタイプ	シーケンスナンバ
タイムスタンプ						
同期送信元識別子 (SSRC)						
コントリビュート送信元識別子 (CSRC) ※CSRCの数に依存						
ペイロードフォーマット拡張 ※ペイロードの種類により任意						
アプリケーションデータ						

図 2.4: RTP パケットの内容

RTP ヘッダ及びペイロードには、以下の情報が含まれている。

バージョン

1 ビット。RTP のバージョンを示す。現状では ver.2 である。

パディング

1 ビット。このビットがセットされているとパケットの最後に 1 オクテット以上のパディングオクテットが挿入されていることを示す。

拡張

1 ビット。このビットがセットされていると、ヘッダのあとに拡張ヘッダが付加される。

CBRC カウント

4 ビット。コントリビュート送信元識別子を何個含んでいるかを示す。最大 15 個。

マーカービット

1 ビット．マーカービットの解釈はペイロードタイプによって異なる．例えば，ビデオのペイロードの場合，フレームの最後をマークし，音声の場合は，音声のかたまりの始まりをマークする．

ペイロードタイプ

1 バイト．パケットに含まれるペイロードの種類を識別する．

シーケンス番号

16 ビット．同一のタイムスタンプを持つ一連のパケットにおける欠落の検出と，順序指定を行う．初期値は乱数によって定める．

タイムスタンプ

32 ビット．パケットに含まれるデータの発生時間が記述される．タイムスタンプの周波数はペイロードタイプにより異なる．

同期送信元識別子 (SSRC)

32 ビット．送信者の ID．セッション内で送信元を一意的に認識するための乱数値．

コントリビュート送信元識別子 (CSRC)

32 ビット × (0 ~ 15)．送信関係者を表す．ミキサーによってミックスされた場合の元の送信者 ID．

ペイロードフォーマット拡張

各種ペイロードタイプによって任意に拡張される．

アプリケーションデータ

映像や音声などのペイロード．

2.2.4 RTP 定義のペイロードタイプ

ペイロードタイプは，映像や音声などの符号化等，特定のペイロードを RTP で輸送する方式を定義する．表 2.1 のように定義されている，RTP のペイロードタイプを一覧する．この他にも，H.263 ビデオストリーム，MPEG1/MPEG2 ビデオなど各種定義されている [7]．

表 2.1: RTP 定義のペイロードタイプ一覧

ペイロードタイプ	符号化の種類	A / V	クロック周波数 (Hz)	チャンネル (音声)
0	PCMU	A	8000	1
2	G.721	A	8000	1
3	GSM	A	8000	1
4	G.723	A	8000	1
8	PCMA	A	8000	1
9	G.722	A	8000	1
14	MPA	A	90000	
15	G.728	A	8000	1
26	JPEG	V	90000	
31	H.261	V	90000	
32	MPV	V	90000	
33	MP2T	AV	90000	
96 ~ 127	動的に割当			

2.2.5 RTCP

RTCP は、RTP のデータ伝送を制御するために使用される。RTP はデータのリアルタイムな転送を行うのみで、データ転送の遅延やロス率などの転送状況の送受信には RTCP メッセージを用いる。RTCP メッセージは、お互いを埋め込むことが出来る複数のパケットで構成され、それぞれの RTCP パケット種類ごとに自身のタイプコードと長さが指定されている。また、RTCP パケットは、同じホストグループに周期的に転送される。このため、実際のデータ伝送がなくても、RTCP パケットによって接続が確立されていることやセッションメンバの存在がわかる。RTCP も RTP と同様に UDP/IP を用いて転送する。

2.2.6 RTCP が提供する機能

RTCP は以下の機能を提供することができる。

アプリケーションへの情報提供

データ配信の品質についてアプリケーションに情報を提供する。各 RTCP パケットには、送信レポート (SR)、受信レポート (RR) の両方またはどちらかが含まれており、これにはアプリケーションにとって有益な、送信されたパケット数、欠落したパケット数、到着

間ジッタなどの統計情報が含まれている．これによりストリームを送出する機器はフロー制御を行うことができる．

クロック同期

各機器のクロックにはどうしても誤差が生じてしまう．このため，RTP パケットに記入されたタイムスタンプだけでは，受信側における情報の再生のタイミングはずれていき，フレーム落ちが発生してしまう．そこで，RTCP パケットが持つ，RTCP パケット送出時刻の RTP タイムスタンプを用いることでクロック同期を可能にしている．

メディア間同期

RTCP パケットは，NTP タイムスタンプを含み，これによって各メディアの再生時刻を，メディアの違いによらず共通に定義可能なクロックに変換して，メディア間の再生時刻を比較することで，メディア間同期を実現する．

RTP 送信元の識別

RTCP は正規名 (CNAME) と呼ばれる，RTP 送信元を識別するトランスポートレベルの識別子を運ぶ．CNAME は，RTP セッションの参加者を把握するために使われる．

RTCP 伝送間隔の制御

ネットワーク資源を圧倒する制御トラフィックが発生するのを防ぐために，また，RTP セッションが多数の参加者を持てるようにするために，セッショントラフィック全体の 5 % に RTCP 制御トラフィックを限定する．また参加者が多くなると，それに対し RTCP パケットの量が線形的に増加する．この RTCP パケットの数によって，セッション内の参加者数を判断できる．

小規模のセッション制御情報の伝達

選択的な機能として，小規模な情報を全てのセッション参加者に伝達することもできる．例えば，RTCP は，ユーザーのディスプレイ上で参加者を識別するための個人名を伝達できる．

2.2.7 RTCP パケットの種類

RTCP は，以上のような機能を提供するために，5 種類の RTCP パケットが定義されている．それぞれの内容を説明する．

SR(Sender Report)

タイプコードは 200 . ストリームを送出した機器が , 送出したストリームに関する情報と他の機器から受信したストリームに関する情報を通知するためのパケット . SR の送出時刻における RTP タイムスタンプの値が記入され , 受信機器にとってクロック同期を行うための時報の役割をする . さらに SR は NTP タイムスタンプを含み , これによって各メディアの再生時刻を , メディアの違いによらず共通に定義可能なクロックに変換して , メディア間の再生時刻を比較することで , メディア間同期を実現する .

RR(Receiver Report)

タイプコードは 201 . ストリームを送出していない機器が , 他の機器から受信したストリームに関する情報を通知するためのパケット . RR 送出時におけるパケット廃棄率 , ジッタを通知することで , ストリームを送出する機器のフロー制御を可能にする .

SDES(Source Description)

タイプコードは 202 . RTP パケットの SSRC/CSRC の値と電子メールアドレスなどのユーザ情報との関係を知覚するためのパケット . SDES からどの SSRC 識別子がどの参加者に対応しているかを認識する . 正規名 (CNAME) を必ず含む .

BYE

タイプコードは 203 . 通信から離脱する機器が , その旨を他の機器に通知するためのパケット .

APP(Application)

タイプコードは 204 . RTCP 規定外のアプリケーション固有の制御情報を通知するためのパケット

2.3 一般的なストリーミングでの RTP の使い方

ストリーミングで扱われるデータとしては , 映像や音声のデータが一般的である . 映像や音声は各種圧縮符号化方式により , RTP で映像や音声のデータを転送するための , データのパケット化のルールや前のパケットが廃棄された場合の復号方法などが定義されている . そこで , 映像については H.261 を , 音声については G.731 を例として , 映像と音声の場合における RTP の使い方を簡単に紹介する .

2.3.1 映像データの場合

H.261 は , ISDN によるテレビ電話・会議用の映像圧縮符号化方式である . H.261 で符号化された映像データを RTP で転送する方式は , 詳細に定義されている [8] . この規格の要旨を図 2.5 と図 2.6 に示す .

H.261 では図 2.5 と図 2.6 で示したように、ピクチャ(画面)、GOB (Group of Blocks)、MB (Macro Block) からなる階層構造の情報として映像をデータ化する。パケット化のルールとして、ひとつの MB に関するデータがふたつのパケットにまたがらないようにすることが定義されている。

ピクチャの最後のパケットでは RTP ヘッダのマーカビットを 1 にすること、また H.261 のパケットタイプにより、90,000Hz を基準クロック周波数として RTP ヘッダのタイムスタンプを記入することが定義されている。

前のパケットが廃棄されても復号できるようにするための情報を、H.261 ヘッダとして付加することも定義されている。

これらの定義により H.261 では、映像の RTP を使ったストリーミングを可能にしている。

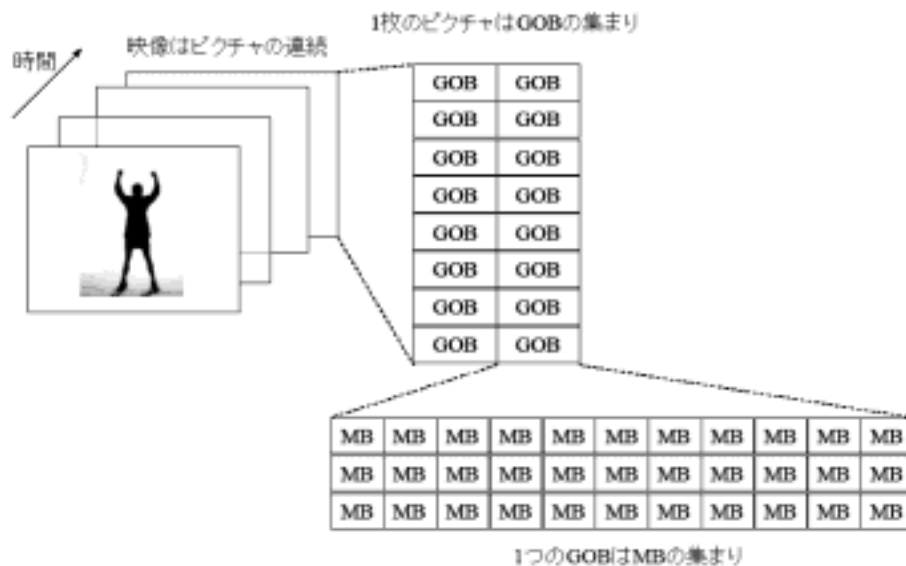


図 2.5: H.261 におけるピクチャの分割方法

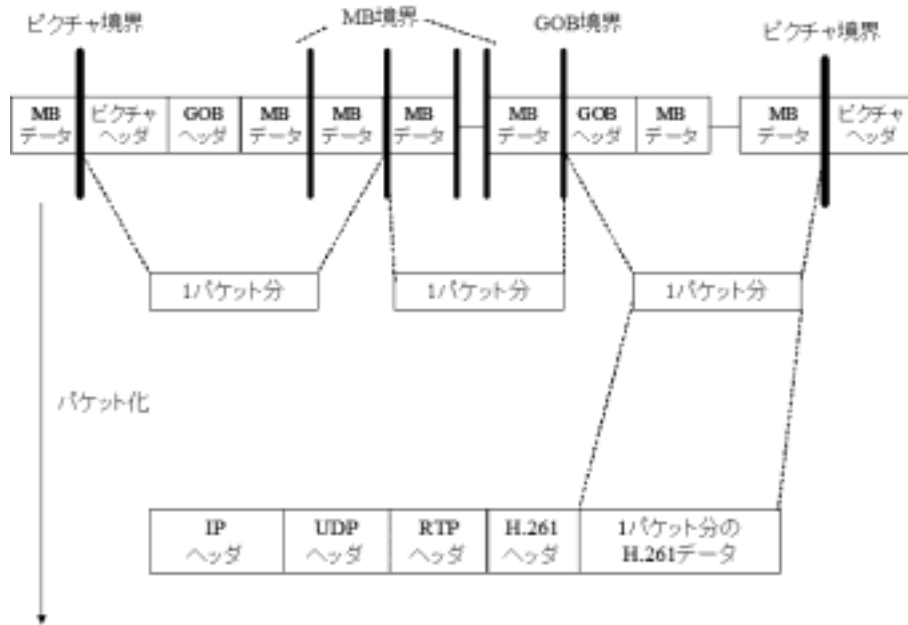


図 2.6: H.261 における映像の RTP パケット化の考え方

2.3.2 音声データの場合

G.723.1 は、アナログ電話網やインターネット上でテレビ電話用の音声を低伝送速度 (5.3kbps/6.3kbps) で実現する音声圧縮符号化方式である [9]。G.723.1 におけるパケット化方式を図 2.7 で示す。音声データのうち、人が話している時間に相当する区間を有音区間、人が黙っている時間に相当する区間を無音区間と呼ぶ。G.723.1 符号化は、30ms ごとにフレームと呼ばれるデータを生成する。このフレームを単位としてパケット化するが、180ms 分を最大として 1 パケットに複数のフレームを含めることが許されている。

RTP ヘッダのマーカビットは、有音区間の始まりにおいて 1 となる。また、タイムスタンプの基準クロック周波数は 8000Hz とする。

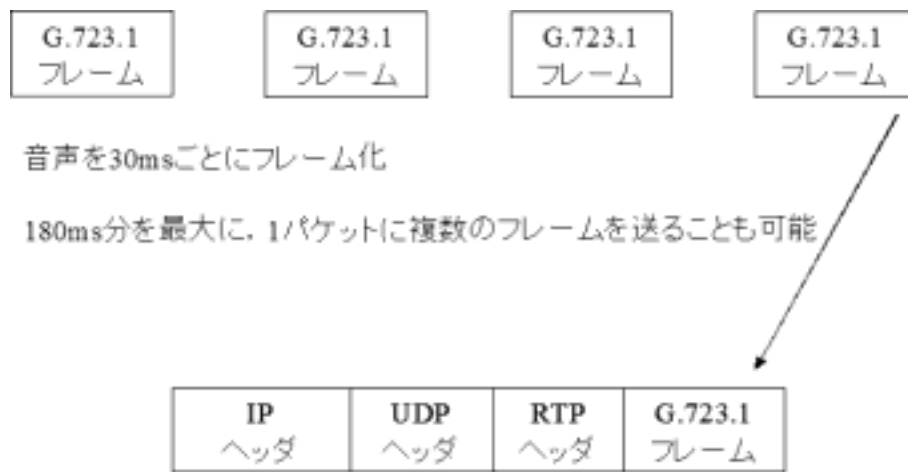


図 2.7: G.723.1 における音声の RTP パケット化の考え方

2.4 ストリーミングの基本的な問題点

RTP/RTCP を用いた IP ネットワーク上におけるストリーミングには、以下のような問題がある。これらの問題をできるだけ解決できるようにストリーミング技術を利用すべきである。

細切れとユニキャスト

ストリーミングを広域ネットワークで適用した場合、細切れとユニキャスト伝送により、パケット喪失の確率が高くなる。つまり、パケット落ちによる映像の劣化、品質低下が前提となる。逆に言えば、広域ネットワークでのストリーミング技術は、多少のフレーム落ちがあってもよいテレビ会議やひとつひとつのフレームがそれほど重要でないアプリケーションに限定して適用されるものになる。

表示開始・再開や切り替えの遅延

通常、ネゴシエーション（相互接続性を保証する）やバッファリングなどで時間がかかるため、表示の開始や停止後の再開に時間がかかる。そこで一般的には、ユーザー自身もあきらめて待つか、別の作業をすることになります。

帯域幅の確保

映像などのストリーミング配信では、帯域幅に応じて画質（速度）調整が可能ではあるものの、帯域幅はだいたい連続的に確保されなければいけない。つまり、ベストエフォート型で、利用ユーザ数の増減によって頻繁に帯域幅が変動する可能性のあるものは、この調整が頻繁に発生するため、あまり適切ではない。

輻輳問題

多数のストリーミング配信が同時に行われると、バックボーンに深刻な影響を与える。つまり、例えばそれらがアクセス回線でメガビットクラスの伝送レートを使うとすれば、そのためにだけでバックボーンでは利用ユーザ数によっては最悪、テラビットクラスの伝送レートが要求され、輻輳を起こす原因となる。

また、インターネット上では、ストリーミング配信はマルチキャスト・ルーティングを行わない限り、ネットワークに悪影響を与え続けることになる。結果としてはネットワークの輻輳や閉塞を引き起こすことになる。しかし、現状のインターネットでは、実験的なものを除けば、マルチキャスト・ルーティング機能がないため、多くのストリーミング配信が LAN に限定されている。

苦手なアプリケーション・サービス

複数対複数のアプリケーション・サービスには、帯域幅を連続的に使用するストリーミングでは現実性に乏しいと考えられる。

また、複数のユーザが異なるさまざまなコンテンツに頻繁にアクセスする対話型アプリケーションへの適用も難しい。これは、もともとストリーミングが不得手とするところで、レスポンスが悪くなり、待ち時間が多くなる。同様に、画面の早送りや巻き戻し、あるいはシーンの変更などへの即応機能に不向きである。したがって、動的シナリオへの適用も難しいと考えられる。

第3章 ストリーミング技術を用いた機器間の通信・制御

本章では、機器間通信・制御にストリーミング技術を用いる意義を説明する。機器間通信・制御にストリーミング技術を用いる利点をあげ、機器間でセンサ情報や機器の制御情報をストリーミング配信する有効性を考察し、その利点をあげる。また、機器間でセンサ情報や機器の制御情報をストリーミング配信するうえでの問題点として、センサ情報や機器の制御情報をストリーミング配信するための規格の定義、及び多数機器のストリーミング配信による輻輳問題に着目し、その解決方法を提案する。

また、本論の目的である、機器が無線 LAN で接続された室内などの限られた空間を対象として、映像・音声のみではなくセンサ情報や機器の制御情報をストリーミング配信型の通信を用いて扱うことで、機器間でのリアルタイムな情報の配信、及び協調動作を実現するための一例として具体的なシステムを提案する。一例として具体的なシステムを提案し、紹介することで、機器間通信・制御にストリーミング技術が用いられる状況のイメージを明確にする。

3.1 ストリーミング技術を用いる利点

ストリーミング技術を機器間通信・制御に用いることで、以下のような利点がある。

- ストリーミング配信型の通信（図 1.2）を行うことで、通信を行うたびにコネクションを確立しないので、通信の本質的な遅延を省くことができる。また、ポーリングのように対象となるすべての機器に対して順番に要求を聞いて回る必要がなくなるので、情報の伝達の遅れを防ぐことができる。
- 空間内に存在するセンサや機器から出される、センサ情報及び機器の制御情報をストリーミング技術を用いることで、センサや機器からリアルタイムに配信することができる。また、それらの情報をユーザや機器はリアルタイムに受信すると同時に利用することができる。
- センサや機器からリアルタイムな情報が流され続けることで、ユーザや機器はリアルタイムな情報に従った動作計画を立てることができるようになる。よって、空間内に存在するストリーミング配信可能なセンサまたは機器と、そのストリーミング配信を受信可能なユーザまたは機器との協調動作を実現することができる。

以上の利点の基づき、機器間でのリアルタイムな情報の配信、及び協調動作の実現するために、センサ情報や機器の制御情報などをストリーミング技術を用いて配信することが有効であると考えられる。

3.2 ストリーミング技術を用いる問題点

ストリーミング技術を機器間通信・制御に用いる上で、センサ情報や機器の制御情報をストリーミング配信する方法を定義すること、また、第2章であげたようなストリーミング配信の基本的な問題点を解決することは重要である。本論では、大きく分けて以下の2点に着目し、その解決方法を提案する。

3.2.1 センサ情報や機器の制御情報をストリーミング配信する方法

映像・音声のみではなくセンサ情報や機器の制御情報をストリーミング配信するためには、センサ情報や機器の制御情報をストリーミング配信するための方法が必要になる。

従来、ストリーミング技術は、映像や音声を配信することを主な目的として利用されているため、本論で用いるような、センサ情報や機器の制御情報をストリーミング配信するための規格は定義されていない。

そこで本論では、センサ情報や機器の制御情報をストリーミング配信するための方法を新たに提案する。詳しくは第4章で説明する。

3.2.2 輻輳問題を解決する方法

第2章でストリーミング配信の基本的な問題点のひとつとしてあげたように、ストリーミング配信を行う上で、輻輳問題の解決は重要である。

近い将来実現するかもしれないユビキタス環境の下で、本論で提案するセンサ情報や機器の制御情報をストリーミング配信することによる協調動作を適応することも考えられる。その場合、ユビキタス環境が構築された室内には、無数の小型PCやセンサが埋め込まれている状況が考えられるため、それら全ての小型PCやセンサがセンサ情報や機器の制御情報を同時にストリーミング配信しようとしたとき、多数のストリーミング配信が同時に行われることで、バックボーンに深刻な影響を与え、ネットワークの輻輳や閉塞を引き起こす可能性が高い。

そこで本論では、輻輳問題を解決するひとつの方法として、各機器がポリシーに従ったアドミッション制御を行うことを提案する。詳しくは第5章で説明する。

3.3 想定する環境のシステム

本論の目的は、機器が無線 LAN で接続された室内などの限られた空間を対象として、映像・音声のみではなくセンサ情報や機器の制御情報をストリーミング配信型の通信を用いて扱うことで、機器間でのリアルタイムな情報の配信、及び協調動作を実現である。この目的を実現するためのシステムとしては、様々なものが考えられる。そこで、具体的な一例として、図 3.1 で示すシステムを想定する。これにより本論が目的としているストリーミング技術を用いた機器間の通信・制御が用いられる状況のイメージを明確にする。本論では以降、このシステムを前提として話を進める。

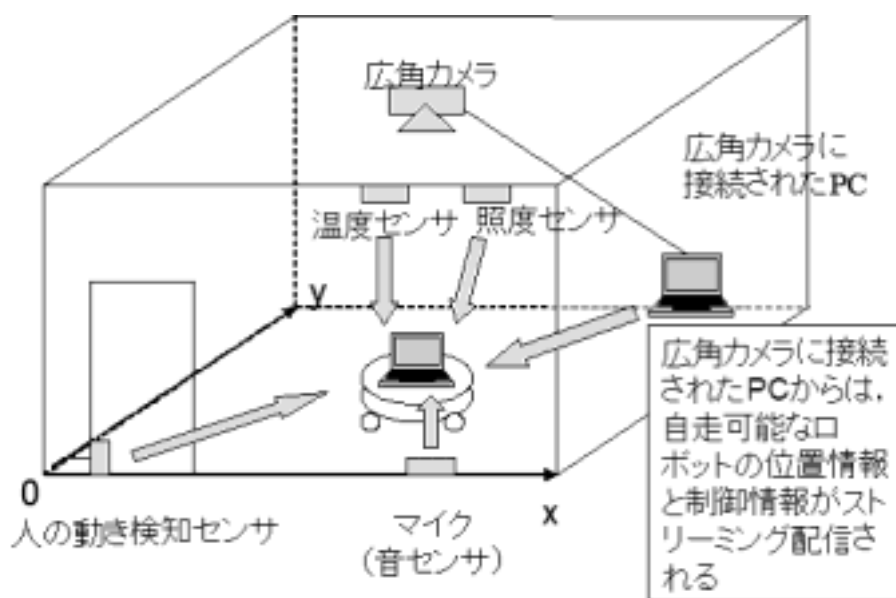
3.3.1 構成

既存の近距離無線通信技術である 802.11 無線 LAN を用いて IP ネットワークが構築された室内に、複数の固定センサと自走可能な掃除ロボットを配置し、機器間でストリーミング配信型の通信を行なうことで、その時々で最適な動作の計画を実現するシステムを想定する。802.11 無線 LAN を用いることで、リンク層以下は 802.11 無線 LAN の規格に依存する。ここでは 802.11 無線 LAN を用いたブロードキャストを行う。これにより、ネットワーク内のすべての機器は、ネットワーク内の配信されるすべての情報を得ることができる。また、新しいセンサや機器、またはユーザが室内に加わった場合の参加が容易になる。IP ネットワーク上で RTP を用いたストリーミング配信を行うものとし、トランスポート層以下は UDP/IP の規格に依存する。よって本論では、RTP 以上の層（アプリケーション層含む）を対象に設計を行う。想定する環境のシステムの構成を図 3.1 で示す。

室内に配置する固定センサと機器、そのセンサ情報と機器の制御情報の内容を表 3.1 で示す。自走可能なロボットの位置情報を提供する機器として、天井に広角カメラを配置する。この広角カメラで撮影された映像は接続された PC で位置情報として処理されてからストリーミング配信される。また、広角カメラに接続された PC からは位置情報以外に自走可能な掃除ロボットの制御情報も提供可能とする。自走可能な掃除ロボットの制御情報の例としては、広角カメラに接続された PC が蓄積した過去の自走可能なロボットの移動履歴を参照することで室内の障害物のある程度の位置を予想し、障害物を避けるような制御情報が提供ができる。制御情報はストリーミングで連続して配信される。

各センサや機器はセンサ情報や機器の制御情報をストリーミング配信するための PC を、自走可能な掃除ロボットはストリーミング配信を受信するために PC を付属しており、その PC を用いてセンサ情報や機器の制御情報を利用した動作計画を立てる。

以上より、各センサや機器からのセンサ情報や機器の制御情報をストリーミング配信を用いることで、自走可能な掃除ロボットはリアルタイムな情報を利用した動作計画を立てることができ、協調動作が可能となる。



© 802.11無線LANによるブロードキャスト通信

図 3.1: 想定する環境のシステム

表 3.1: 想定する環境のシステムに存在するセンサ及び機器

センサ及び機器名	利用内容
温度センサ	室内の温度を測定
照度センサ	室内の明るさを測定
音センサ	室内の音の有無を判別
人の動き検知センサ	出入り口での人の出入りを判別
広角カメラ	掃除ロボットの位置情報を取得
広角カメラの接続された PC	広角カメラからの位置情報と計画した制御情報を提供

3.3.2 想定する自走可能な掃除ロボットの動き

自走可能な掃除ロボットとしては、ルンバ[10]の動きを想定する。ルンバは単純なアルゴリズムに従い、室内を自走し掃除を行う掃除ロボットである。動作の特徴としては、以下のようなものがあげられる。

- 想定する室内の広さに応じて S,M,L のボタンがあり、押されたボタンに応じて室内の広さを想定した動く範囲を設定する。

- 装備されたタッチセンサにより，室内の壁に接触すると，進行方向を変更する．
- 装備された段差センサにより，階段など段差のある場所に接近すると，進行方向を変更し，段差からの落下を避ける．
- 動作開始点より，らせん状の動きをはじめめる．S,M,L ボタンで想定された室内の広さに到達あるいは壁に接触するまで，らせん状の動きを続ける．
- 想定された室内の広さに到達あるいは壁に接触したあとは，壁に接触しながら進行方向を変え，直線的な動き続ける．

ルンバはこれらの動きの繰り返すことで，室内全体の掃除を実現しようとする．ルンバの動きのアルゴリズムは単純であるため，各センサや機器からのセンサ情報や機器の制御情報を利用すれば，ルンバと各センサや機器との協調動作により，ルンバの動きを賢くすることができる．

3.3.3 想定する協調動作

想定する環境のシステムでは，以下のようなシナリオを想定して自走可能な掃除ロボットの動きを協調動作によって賢くすることを提案する．

人が出かけたときあとに掃除を開始する自走可能な掃除ロボット

- 部屋が暗くなったら自走可能な掃除ロボットが動き出す．反対に，部屋が明るくなったら自走可能な掃除ロボットが止まる（照度センサ）
- 人が室内にいる場合，出入り口で人の動きを検知したら自走可能な掃除ロボットが動き出す（動き検知センサ）
- 部屋が暗くなりかつ人の出入りがある場合，自走可能なロボットが動き出す．また，人の出入りがありかつ部屋明るくなったら自走可能なロボットが止まる（照度センサと動き検知センサの複合）
- あらかじめ自走可能な掃除ロボット側 PC に始点となるべきエリアを数箇所与えるそのエリア内に侵入したら，再度らせん動作を開始することで，従来の動作よりも掃除のムラを減らす（位置情報の利用）
- 広角カメラに接続された PC が自走可能な掃除ロボットの位置情報履歴を蓄積，蓄積された位置情報の座標をマッピングすることである程度室内の障害物を認識できる．それにより，障害物をよける動作を広角カメラに接続された PC で計画し，そのための制御情報を自走可能な掃除ロボットに提供する（機器の制御情報の配信）

- 室内の温度による自走可能な掃除ロボットの動作の開始，室内で物音がしたとき自走可能な掃除ロボットの動作の停止といったことも考えられる（温度センサ・音センサ）

一例として以上のような動作を想定する．しかし，自走可能な掃除ロボットと各センサや機器との協調動作による，自走可能な掃除ロボットの動きの賢さは，各センサや機器からの情報をいかにうまく利用すべきかに依存する．そのため，これら以外にも様々な自走可能な掃除ロボットと各センサや機器との協調動作は考えられる．

第4章 センサ情報及び機器の制御情報の ストリーミング方式

本章では，空間に存在するセンサや機器から得られる，センサ情報及び機器の制御情報をストリーミング配信するため，テキストデータストリーミングの方式を新たに提案し，それについて説明する．

4.1 ストリーミングを用いた機器間の通信・制御で扱うデータ

本論では，機器が無線 LAN で接続された室内などの限られた空間を対象として，映像・音声のみではなくセンサ情報や機器の制御情報をストリーミング配信型の通信を用いて扱うことで，機器間でのリアルタイムな情報の配信，及び協調動作の実現を目的としている．そのため，空間内に存在するセンサや機器から得られる，センサ情報及び機器の制御情報を無線 LAN で構築された IP ネットワーク上でストリーミング配信する必要がある．しかし，既存の IP ネットワーク上のストリーミング技術は，映像や音声のストリーミングを主な対象としており，センサ情報及び機器の制御情報をストリーミングするための方式は定義されていない．そこで空間に存在するセンサや機器から得られる，センサ情報及び機器の制御情報のストリーミング配信するための方式を新たに提案する必要がある．

ストリーミング配信の対象となるデータである，センサ情報や機器の制御情報にはどのようなデータがあるのか紹介する．

4.1.1 センサ情報

空間に存在するセンサや機器には，温度センサ，照度センサ，位置情報を取得するためのカメラなどの機器をはじめ，ユーザや機器の位置や嗜好，空間の環境を知るために，様々な種類，用途のセンサや機器が用いられる場合が考えられる．これらのセンサや機器から得られるセンサ情報からユーザや機器の状況や環境に把握し，機器を制御することで，協調動作を実現できる．センサ情報とは，空間の状況や環境に把握するための情報である．空間に存在するセンサや機器の例として，想定する環境のシステムでセンサ情報を提供するセンサ及び機器を表 4.1 で示す．

表 4.1: 想定する環境のシステムでセンサ情報を提供するセンサ及び機器

センサ及び機器名	情報の提供内容
温度センサ	室内の温度
照度センサ	室内の明るさ
音センサ	室内の音の有無
人の動き検知センサ	出入り口での人の出入り
広角カメラに接続された PC	広角カメラからの位置情報

4.1.2 機器の制御情報

空間に存在するセンサからは，そのセンサの特性に基づいたセンサ情報が得られる．しかし，空間に存在する機器からは，ある機器ともう一方のある機器がお互いの関係に基づき協調動作を行うために，機器の制御情報が提供される場合も考えられる．機器の制御情報とは，機器を直接的に制御するための情報である．その例として，想定する環境のシステム制御情報を提供する機器を表 4.2 で示す．

表 4.2: 想定する環境のシステムで制御情報を提供する機器

センサ及び機器名	情報の提供内容
広角カメラに接続された PC	自走可能なロボットの移動履歴に基づく制御情報

4.2 テキストデータとしてストリーミング

空間に存在するセンサや機器から得られるセンサ情報及び機器の制御情報は，それぞれ各センサや機器ごとにフォーマットが異なる．各センサ単体は図 4.1 の温度センサの例で簡単に示すように，センサがアナログデータを取得，そのデータをアンプを用いて増幅，A/D コンバータによるセンサデータのアナログ/デジタル変換を経てセンサ付属の PC にデータが送られ，そのデータを PC が読み直すといったプロセスを経て，人が読めるテキストデータとして PC に取得することが出来る．このように各センサ単体から得られるセンサ情報は，アナログデータからデジタルデータへと様々な変換を伴うので，どの時点でのデータを扱うべき決定する必要がある．

本論では，各センサ単体からのセンサ情報は，各センサ付属の PC のデータ変換プロセスを経ることで，最終的にはテキストデータに変換してからストリーミング配信で扱うことにする．また，機器の制御情報に関しても，機器付属の PC 上でテキストデータとして機器の制御情報を生成するしてからストリーミング配信で扱うことにする．よって，本

論で新たに提案するセンサ情報や機器の制御情報のストリーミング配信では，データ変換のプロセスを経て最終的にPC上に得られた，テキストデータがストリーミング配信の対象データとする．これにより，各センサや機器から得られるセンサ情報及び機器の制御情報を，人が読めるテキストデータという共通のフォーマットとして一括で扱えるようになる．

空間に存在するセンサや機器から得られるセンサ情報及び機器の制御情報の統一したフォーマットとして人が読めるテキストデータを対象としたことにより，IP ネットワーク上でストリーミング配信を行うために，RTP を用いてテキストデータをストリーミング配信するため方式の設計が必要となる．

想定する環境のシステムでは，表 4.3 のようなテキストデータがストリーミング配信の対象となる．

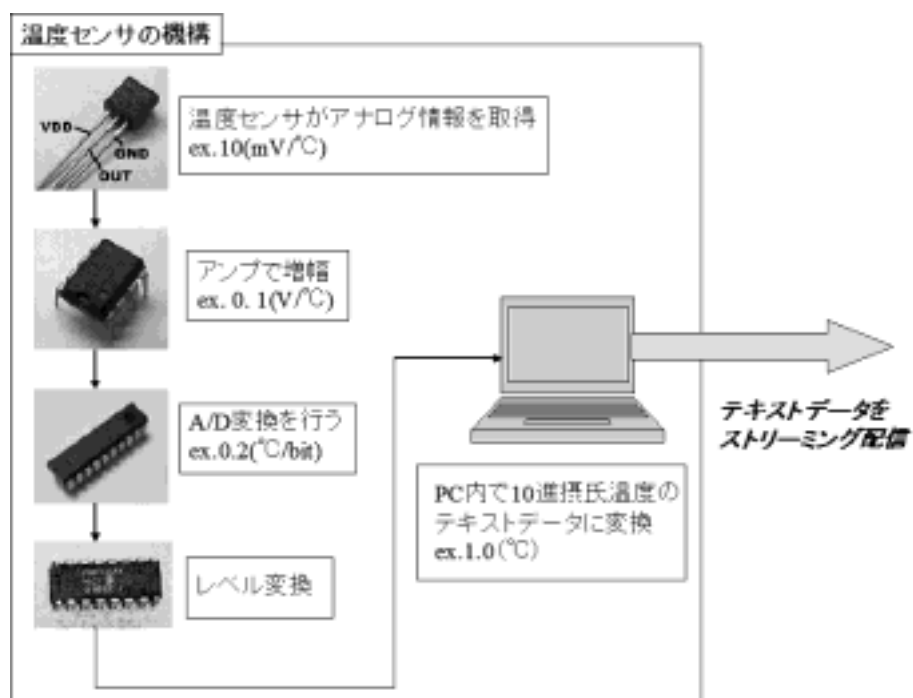


図 4.1: 温度センサ機構のデータ変換プロセスの例

表 4.3: 想定する環境のシステムで対象となるテキストデータ

情報の種類	単位・対象	テキストデータの例
温度情報	摂氏温度 ()	25.3
照度情報	側光量 (lx)	50
音情報	音圧 (dB)	46.1
人の動き検知情報	動きあり, なし	on,off
位置情報	xy 座標	x="100" y="200"
機器の制御情報	自走可能な掃除ロボットの動き	go,back,left.right

4.2.1 既存のテキストデータストリーミング方式

IP ネットワーク上で RTP を用いてテキストデータをストリーミングするための既存の規格として, RFC2793[11] がある. ここでは RFC2793 規格の特徴を紹介する.

RFC2793 規格の特徴

- ストリーミング技術を用いたテレビ会議などにおいて, チャットのようなテキストコミュニケーションをストリーミング技術を用いて行うことを目的としている (図 4.2).
- チャットで人が入力できる文字数は数文字であるという考えに基づき設計がなされており, タイムスタンプのクロック周波数は独自の 1000Hz に設定されている.
- 1 パケット内の収めることが出来る文字数は数文字までと決められている.
- ITU-T で定義された T.140[12] というテキストコミュニケーション専用のコードに従って, テキストのストリーミングを行うことが定義されている.
- T.140 では, テキストコードとして UTF-8 を用いること, また, チャット時に必要となる改行やバックスペースなどのキーコードが独自に定義されている.
- RTP パケットの内容を図 4.3 のように定義されている.
- ペイロードタイプの値は定義されていないので, ペイロードタイプは動的に 96 ~ 127 を割り当てる.

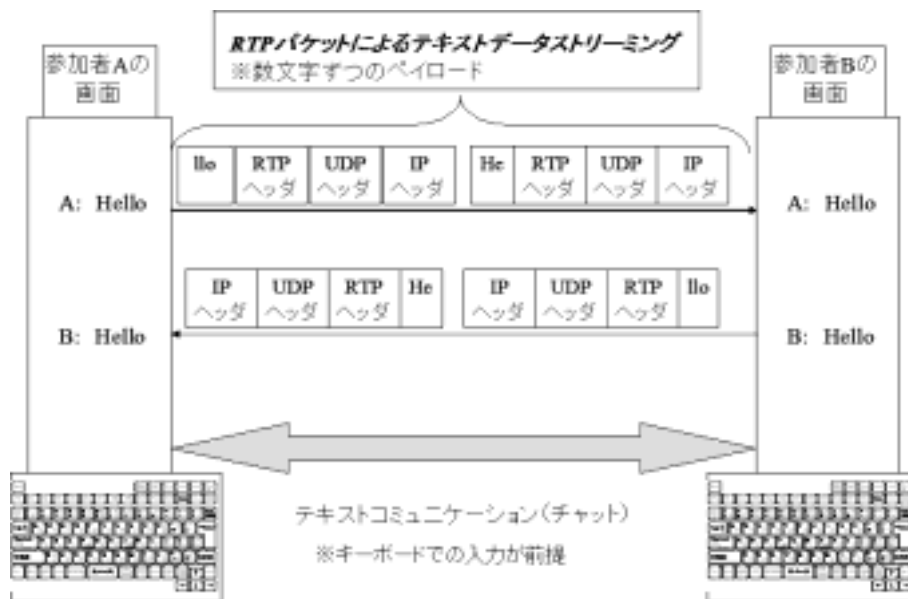


図 4.2: RFC2793 が想定するテキストストリーミングの例



図 4.3: RFC2793 で定義された RTP パケットの内容

4.2.2 新たに提案するテキストデータストリーミング方式

IP ネットワーク上で RTP を用いてテキストデータをストリーミングするための既存の規格として、RFC2793 を紹介したが、本論が目的とする、センサ情報や機器の制御情報をテキストデータとしてストリーミング配信するとき RFC2793 の規格を用いるには、以下のような問題点がある。

RFC2793 の問題点

- チャットを想定したテキストコミュニケーション特化しているため、1 パケットに数文字しか転送できない。
- RFC2793 では T.140 のテキスト形式を用いることを絶対として定義しているため、自由度が低い。
- タイムスタンプ周波数が独自の 1000Hz に設定設定されているため、1 秒間にパケット化できるデータ数が少ない。

以上のような理由により、RFC2793 の規格はあまりにもチャットのようなテキストコミュニケーションと、T.140 を用いることに特化されているため、本論が提案するテキストデータストリーミングとは相性が悪いとされる。

そこで本論では、新たなテキストデータストリーミング方式を提案する。IP ネットワーク上で RTP を用いて、センサ情報や機器の制御情報のテキストデータをストリーミング配信するためのストリーミング方式である。図 4.4 のようなプロトコルスタックとなる。新たに提案するテキストデータストリーミング方式の RTP パケットでは、次のような定義を行う。

新たに提案するテキストデータストリーミング方式の RTP パケットの定義

- 今までにないペイロードタイプなので、ペイロードタイプ値は動的に 96 ~ 127 を割り当てる。
- タイムスタンプのクロック周波数は一般的な値として、音声のタイムスタンプのクロック周波数に従い、8000Hz とする。
- 英数文字のテキストデータのみを対象とするため、UTF-8 の文字コードを用いる。
- 各センサや機器からのセンサ情報や機器の制御情報のテキストデータは、各センサや機器ごとに扱い、一回で得られた情報を 1 パケット内に収める。
- センサ情報や機器の制御情報の変化に必要十分に対応できる情報量として、1 秒間に 10 パケットのストリーミング配信を行う（ちなみに圧縮符号化形式にもよるが、大まかな値として、映像データでは 1 秒間に 30 パケット程度、音声データでは 1 秒間に 10 パケット程度のデータがストリーミング配信されている）

- センサ情報や機器の制御情報のテキストデータは、十分に少ないデータ量なので、圧縮は行わない。
- 本論では、センサ情報や機器の制御情報を扱ううえで、わずかなパケットが廃棄されたとしても、後から配信されるデータを参考にすれば、システムに大きな影響がおきる可能性は低いという考えに基づき、前パケットが廃棄された場合、復号は行わない。そのため、新たなペイロード拡張の追加を必要としない。

以上の定義に基づき、新たに提案したテキストデータストリーミング方式の RTP パケットの内容を図 4.5 で示す。

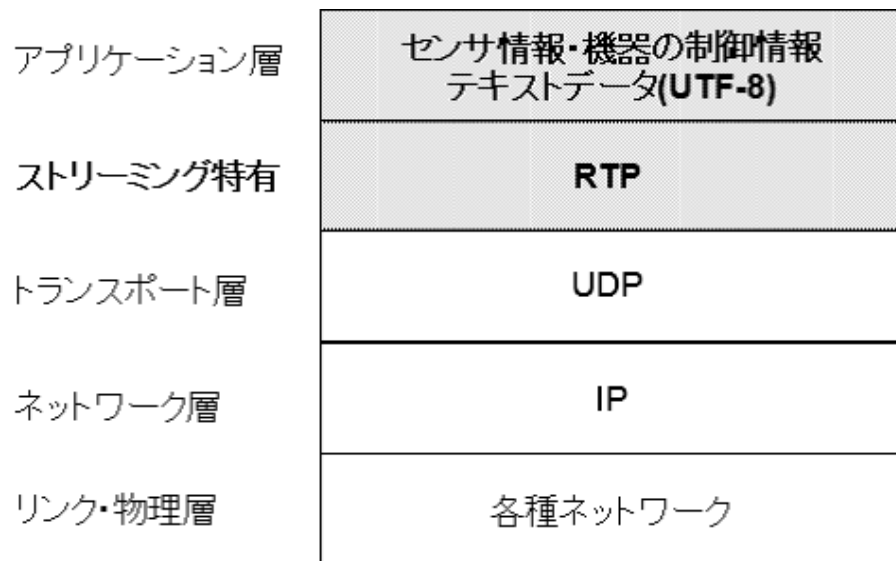


図 4.4: 新たに提案するテキストデータストリーミングのプロトコルスタック



図 4.5: 新たに提案するテキストデータストリーミングの RTP パケットの内容

4.3 センサ情報や機器の制御情報を XML を用いて記述

各センサや機器からのセンサ情報や制御情報をテキストデータとして，RTP を用いてストリーミング配信を行うとき，テキストデータを共通の形式で整理するために，XML[13] を用いてセンサ情報や機器の制御情報を記述する．

4.3.1 XML とは

XML とは，データをネットワーク経由で送受信するための言語である．ユーザが独自のタグを指定できる，メタ言語の一種である．XML はその性質上，他のマークアップ言語の骨組みとして使用されることが多い．マークアップ言語として用いられる XML 文書はテキストデータになるため，テキストエディタを使って人間が普通に読むことが可能である．XML で扱うテキストデータの文字コードは UTF-8 である．

4.3.2 XML を用いて記述する利点

XML でセンサ情報や制御情報を記述する利点としては，データを構造が整理されたテキストデータとして扱えることがある．また，DTD(Document Type Definition) を作成し，XML の構造を検証可能しておくことができる．新たなセンサ情報や機器の制御情報を追加する際，DTD に従って新しいセンサ情報や機器の制御情報のタグ付けを行うことで，新しいセンサ情報や機器の制御情報の追加を比較的簡単に行うことができる．また，XML で記述することで，XML を理解できる各種アプリケーションで，本論で提案したセンサ情報や機器の制御情報を利用できる可能性が広がる．

4.3.3 想定する環境のシステムでの XML 記述

想定する環境のシステムにおいて XML で記述したときの構造を図 4.6 で示す．また，この構造の DTD を図 4.7 で示す．各センサ情報及び機器の制御情報について定義した XML の記述の方法について説明する．その中で，一例として作成するシステムにおける温度センサ情報が，実際にどのようなテキストデータとして RTP ペイロードに記述されるのかを例をとして紹介する（図 4.8）．また，広角カメラに接続された PC からは，位置情報と制御情報をストリーミング配信することができる．XML 記述を用いることで，ひとつの RTP パケットのペイロードに位置情報と制御情報を記述することができる．その例を図 4.9 で示す．その他の一例として作成するシステムにおける，各センサ情報や機器の制御情報の記述方法についても例と同様に記述することができる．

```

センサ情報・機器の制御情報<info>属性:priority
| ーセンサ情報<sensor-info>
|   | ー温度情報<temperature>属性:unit
|   | ー照度情報<brightness>属性:unit
|   | ー音情報<sound>属性:unit
|   | ー人の動き検知情報<motion>属性:unit
|   | ー位置情報<position>属性:x,y
| ー制御情報<control-info>
|   | ー動作命令情報<operation>属性:unit

```

図 4.6: 想定する環境のシステムでの XML の構造

```

<!DOCTYPE info[
<!ELEMENT info (sensor-info?,control-info?)>
<!ELEMENT sensor-info
(temperature?,brightness?,sound?,motion?,position?)>
<!ELEMENT control-info (operation?)>
<!ELEMENT position (x,y)>
<!ELEMENT temperature (#PCDATA)>
<!ELEMENT brightness (#PCDATA)>
<!ELEMENT sound (#PCDATA)>
<!ELEMENT motion (#PCDATA)>
<!ELEMENT operation (#PCDATA)>
<!ELEMENT x (#PCDATA)>
<!ELEMENT y (#PCDATA)>
<!ATTLIST info priority CDATA #REQUIRED>
]>

```

図 4.7: 想定する環境のシステムでの XML の構造の DTD

各センサ情報及び機器の制御情報について定義した XML の記述の方法

センサ情報・機器の制御情報 < info >

すべてのセンサ情報・機器の制御情報はタグ < info > に集約される。属性として優先度”priority”を記述することができる。優先度の値は0,1,2,3のいずれかが与えられる。この優先度とは、第5章で説明するポリシーに従ったアドミッション制御で必要となる情報である。

センサ情報 < sensor-info >

すべてのセンサ情報はタグ < sensor-info > に集約される。

温度情報 < temperature >

属性：unit で、摂氏温度 () の単位を記述できる。有効数字は0.1までとして、- は利用するが+は使わない。

- 例： < temperature unit=” ” > 25.3 </temperature >

照度情報 < brightness >

属性：unit で、側光量 (lx) の単位を記述できる。整数値を扱う。

- 例： < brightness unit=”lx” > 50 </brightness >

音情報 < sound >

属性：unit で、音圧 (dB) の単位を記述できる。有効数字は0.1までとする。

- 例： < sound unit=”db” > 46.1 </sound >

人の動き検知情報 < motion >

人が出入り口を通過したかしないかの情報をテキストデータで記述できる。人の出入りがあった場合を”up”，人の出入りがない場合は”off”と記述する。属性:unit は”onoff”とする

- 例： < motion unit=”onoff” > up </motion >

位置情報 < position >

一例とし提案したシステムでは、広角カメラからの映像を、接続されたPCで画像解析することで、自走可能な掃除ロボット位置情報を生成する。その位置情報は、x軸・y軸の値として与えられる。x軸・y軸の値はタグ < position > の属性として記述できる。値は整数値とする。

- 例： < position x="100" y="200" / >

機器の制御情報 < control-info >

すべての機器の制御情報はタグ < control-info > に集約される。

操作命令情報 < operation >

自走可能な掃除ロボットの操作命令情報をテキストデータで記述できる。前進の命令は"go"，後退の命令は"back"，右回りの命令は"right"，左回りの命令は"left"，電源を入れる命令は"poweron"，電源を切る命令を"poweroff"としてそれぞれ記述する。ここでは，属性:unitをroombaとして，想定する環境のシステムでの，ルンバの動きを想定した操作命令を定義している。

- 例： < operation unit="roomba" > back < /operation >

```
<?xml version= "1.0"?>
<info priority="0">
  <sensor-info>
    <temperature unit="°C">25.3</temperature>
  </sensor-info>
</info>
```

図 4.8: 一例として作成するシステムでの温度情報の XML 記述例

```
<?xml version= "1.0"?>
<info priority="3">
  <sensor-info>
    <position x="100" y="200"/>
  <control-info >
    <operation unit="roomba">back</operation>
  </control-info>
</sensor-info>
</info>
```

図 4.9: 一例として作成するシステムでの位置情報と制御情報の XML 記述例

4.3.4 XML 記述テキストデータでの RTP パケットのデータ量

XML 記述テキストデータを RTP のペイロードとしてストリーミングするときのデータ量について、説明する。XML 記述テキストデータは、UTF-8 の文字コードで記述することを定義したので、1 文字あたり 1 バイトのデータ量を持つことになる。例えば、想定する環境のシステムにおける温度情報の RTP ペイロードのデータ量は、温度情報の XML 記述テキストデータが図 4.8 より、114 文字で記述されていることから、114 バイトであることがわかる。ちなみに、図 4.8 では、テキストデータが読みやすいように TAB 文字や改行などを行っているが、実際に温度情報のテキストデータを RTP ペイロードとして記述するときは、TAB 文字や改行などの記述はしないので、その分の文字数は RTP ペイロードのデータ量に含まない。

XML で記述されたテキストデータの RTP ペイロードの他に、12 バイトの RTP ヘッダ、8 バイトの UDP ヘッダ、20 バイトの IP ヘッダを付加することで、センサ情報や機器の制御情報をストリーミングする際の 1 パケットのデータ量は、想定する環境のシステムにおける、温度情報を例とすると、154 バイトとなる（図 4.10）。この 1 パケットあたりのデータ量は、ストリーミングで扱う 1 パケットあたりのデータ量としては十分に小さい。そのため、ストリーミング配信をするのに問題のないデータ量であると言える。



図 4.10: 提案したテキストデータストリーミングでの 1 パケットのデータ量

4.4 提案したストリーミングシステムの手順

新たに提案したテキストデータストリーミング方式を利用すると、図 4.11 のような手順を経て、データを送信する各センサや機器から、データを受信する機器へストリーミング配信が行われる。

まず、送信側の各センサや機器が空間の情報を取得する、それらの情報は各センサや機器付属の PC に持ち込まれる。その付属 PC 内で、各センサや機器から情報はテキストデータ化される。テキストデータ化された情報は定義された XML で記述される。XML で記述された各センサ情報や機器の制御情報のテキストデータに新たに提案した RTP ヘッ

データを付加する。さらにUDPヘッダとIPヘッダも付加され、802.11無線LANを用いて受信側の機器にストリーミング配信される。受信側の機器では、そのストリーミング配信されたパケットを受信し、RTPヘッダの解釈を行い、XMLテキストデータも解釈をされて、最終的に受信側で利用可能な情報として受け取られる。

本論で提案するストリーミング配信は、送信側から受信側への一方的なリアルタイムな情報の配信のみを行うため、受信側はストリーミング配信されたデータを受け取り続けるのみである。そのため、以上で説明した手順に従って、送信側の各センサや機器から、受信側の機器へと、ストリーミング配信が継続される。

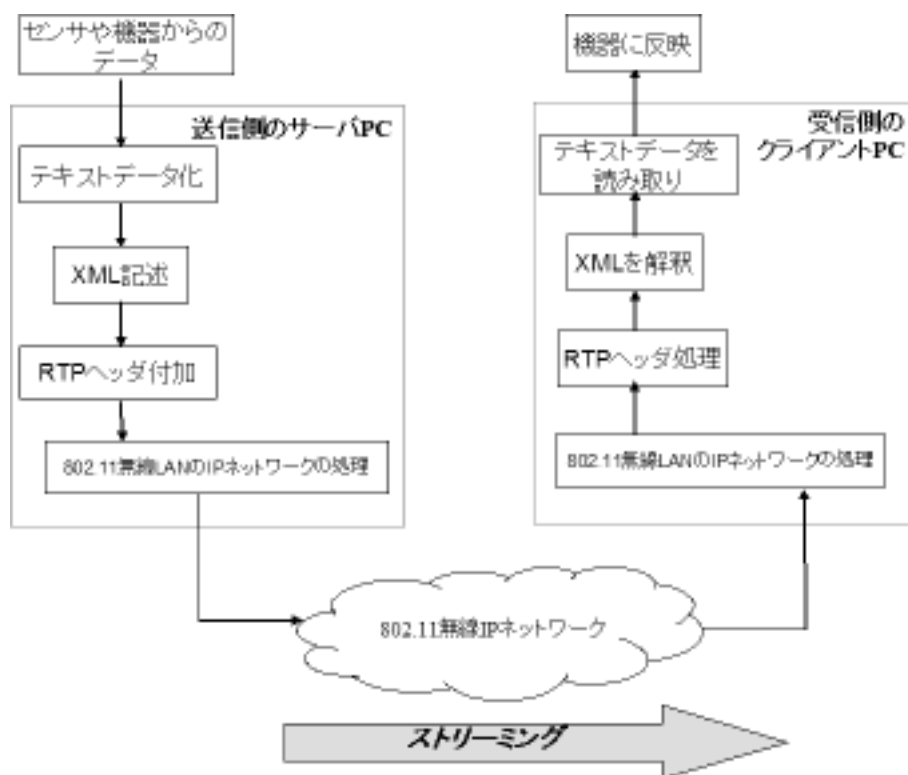


図 4.11: 提案したストリーミングシステムの手順

第5章 アプリケーションレベルでのアドミッション制御

本章では、第3章であげた輻輳問題の解決方法として、各センサや機器のアプリケーションレベルでのアドミッション制御を行うこと提案し、それについて説明する。

5.1 アドミッション制御を行う意義

第3章で述べたように、近い将来実現するかもしれないユビキタス環境の下において、本論で提案するセンサ情報や機器の制御情報のストリーミング配信の方法を適応する場合について考えてみる。ユビキタス環境が構築された室内には、無数の小型PCやセンサが埋め込まれている状況が考えられる。そのため、それら全ての小型PCやセンサがセンサ情報や機器の制御情報を同時にストリーミング配信を行った場合、大多数のストリーミング配信が同時に行われることで、バックボーンに深刻な影響を与え、ネットワークの輻輳や閉塞を引き起こす可能性が高い。この問題を解決するためのひとつの方法として、各センサや機器のアプリケーションレベルでアドミッション制御を行うことを提案する。

各センサ情報や機器の制御情報は、情報の性質によって、その情報の変化の頻度やが一定時間あたりの情報の重要性が異なることに着目し、情報の性質について各センサ情報や機器の制御情報を分類する。まず、各センサ情報や機器の制御情報を利用されるタイミングに基づいて分類する。本論ではこれを場合分けと呼ぶ。その結果、各情報の利用されるタイミングの違いに基づき、各センサや機器から得られる、各センサ情報や機器の制御情報のストリーミング配信の優先度の設定を行うことができる。本論ではこれをクラス分けと呼ぶとする。

これらにより、各センサや機器から得られる、各センサ情報や機器の制御情報にストリーミング配信に優先度を設定することができる。その優先度を利用した機器間の共通のルールにより、ストリーミング配信を行っている各センサや機器は、アプリケーションレベルで空間の通信の状況を確認し、それに応じて、優先度の高低に基づき、ストリーミング配信を開始したり、中止したりをアプリケーションレベルで判断してアドミッション制御を行う。これにより、ポリシーに従ったアドミッション制御を実現できる。これにより、ネットワーク全体の総通信量を軽減することができる。

5.2 場合分け

各センサや機器からの各センサ情報や機器の制御情報を，各情報が情報の性質に基づいて場合分けを行う．各センサ情報や機器の制御情報について，情報の変化頻度と情報の重要性の2点（それぞれ相反する2種類のパターンを持つ）に着目して，場合分けを行う．

場合分けの種類

情報の変化頻度（バンド幅を変更）

- A 情報の変化が頻繁なため，連続してデータを配信する必要がある．1秒間に10パケット送出する．
- B 情報の変化が乏しいため，ある程度間隔のある周期でデータを配信しても支障がない．1秒間に2パケット送出する．

情報の重要性（必要となるデータの連続発生時間の長短の違い）

- i 必要となるデータの発生時間が短く，情報が発生した時間通りにデータを配信し続ける必要がある
- ii 必要となるデータの発生時間が長く，多少なら情報の配信が遅れても支障がない

5.2.1 情報の変化頻度

情報の変化頻度とは，各情報のデータが頻繁に変化するか，そうでないかの分類である．情報の変化頻度の分類によって，ストリーミング配信するとき，1秒間に送出するデータ量を変更する．本論では，各センサ情報や機器の制御情報のストリーミング配信は1秒間に10パケット送出されることを基本としている．1つのパケットには送出された時間での1つのセンサ情報または機器の制御情報が含まれている．連続してデータを配信する必要がある場合は，1秒間に10パケット送出する．そのため頻繁な情報の変化に対応できる．反対に，ある程度間隔のある周期でデータを配信する場合とは，1秒間に2パケットのデータを送出することにする．そのため頻繁な情報の変化に対応できないが，送出するデータ量は減少する．つまり，1秒間に送出すべきデータ量に違いができる．ネットワークに送出される総データ量の低下は，輻輳を起こさないためには有効であると考えられる．そのために，必要ないデータの送出をなるべくやめることで，1秒間に送出するデータ量自体をできるだけ減らすように試みる．情報の変化頻度については図5.1で示す．

◎情報の変化頻度

・Aの特徴

◎情報の変化が頻繁⇒通常の定義とおりの情報量でストリーミング
1秒間に10の情報(10/パケット)



・Bの特徴

◎情報の変化が乏しい⇒情報量を減らしてストリーミング
1秒間に2つの情報(2/パケット)



データの
バンド幅に
差をつける

図 5.1: 情報の変化頻度の違い

5.2.2 情報の重要性

情報の重要性とは、1パケット情報あたりの情報の発生時間の時間の違い、つまりは、1パケットあたりの情報の重要性の違いによる分類である。ストリーミング配信のアドミッション制御が行われた場合、各センサや機器がランダム時間、ストリーミング配信を停止してしまう状況が考えられる。送信側のセンサや機器で、ストリーミング配信を受信する側の機器の、新たな動作開始のトリガーとなる情報が得られた場合、その情報の発生している時間が短いので、すぐにストリーミング配信する必要がある場合と、またはその情報の発生している時間が長いので、多少ストリーミング配信が停止されていても、その必要となる情報は頻繁に変化は起きていないので、多少遅れたタイミングでストリーミング配信が行われても支障がない場合とで分類する。つまり、必要となるデータの発生時間の長短の違いである。この分類により、アドミッション制御時にストリーミング配信を停止してもなんとかなる情報と、停止すると必要な情報を取り逃してしまう情報とに分けられる。情報の重要性については図 5.2 で示す。

◎情報の重要性

・iの特徴



・iiの特徴



図 5.2: 情報の重要性の違い

5.2.3 想定する環境のシステムでの場合分け

想定する環境のシステムについて、場合分けを行なった結果を図 5.3 に示す。想定する環境のシステムでの場合分けを決定した理由を説明する。

情報の種類	場合分け			
	A		B	
	i	ii	i	ii
室内の温度				○
室内の照度				○
室内の音				○
人の動き検知			○	
位置情報(天井に固定した広角カメラ)	○			
制御情報	○			

図 5.3: 想定する環境のシステムでの場合分け

室内の温度

- 情報の変化頻度

室内の温度は頻繁に変化しない。また想定する環境のシステムの場合、1 刻みの温度変化には対応する必要はあまりない。B の特徴を選ぶ。

- 情報の重要性

室内の温度が、10 から突発的に 30 に変化することは考えにくい。つまり、必要な情報の発生時間が長い。ii の特徴を選ぶ。

室内の照度

- 情報の変化頻度

室内の照度は頻繁に変化しない。想定する環境のシステムの場合、部屋の明暗がおおまかにわかればよい。B の特徴を選ぶ。

- 情報の重要性

室内の照度が、想定する環境のシステムの場合、部屋が暗くなったらそのまま暗く、また明るくなったらそのまま明るいままである場合が大半だと考える。つまり、必要な情報の発生時間が長い。ii の特徴を選ぶ。

室内の音

- 情報の変化頻度

室内の音は、一例提案するシステムでは、人が室内にいる場合の生活音と、人が室内にいないときの室外からの多少の騒音などが考えられる。これらの音は頻繁に大きな変化をすることは考えにくい。B の特徴を選ぶ。

- 情報の重要性

室内の音情報は、人が室内にいる場合の生活音と、人が室内にいないときの室外からの多少の騒音を対象とする場合、必要な情報が発生する時間が長い。しかし例えば、大きな音などは突発的に発生すると考えられるが、一例提案するシステムでその情報の必要性は低い。ii の特徴を選ぶ。

人の動き検知

- 情報の変化頻度

人の動き検知センサは、想定する環境のシステムにおいて、部屋の出入り口に配置されている。外出または帰宅しようとしている人が頻繁に出入り口を行き来するとは考えにくい。そのため情報の変化は少ないと考える。Bの特徴を選ぶ。

- 情報の重要性

人の動き検知は、外出または帰宅しようとしている人の出入りを見逃すべきではない。人の出入りによって、発生する必要な情報は、発生する時間が短い。iの特徴を選ぶ。

位置情報

- 情報の変化頻度

一例提案するシステムでは、自走可能な掃除ロボットの移動により位置情報は頻繁に変化する。Aの特徴を選ぶ。

- 情報の重要性

自走可能な掃除ロボットは移動し続けるため、必要な情報の発生時間は短くかつ連続的に発生する。iの特徴を選ぶ。

制御情報

- 情報の変化頻度

一例提案するシステムでは、自走可能な掃除ロボットの移動履歴に基づいて、障害物を避けるための制御情報を配信する場合、制御情報は移動方向を伝えるため頻繁に変化する。Aの特徴を選ぶ。

- 情報の重要性

自走可能な掃除ロボットが、障害物を避けるための情報は連続的に変化しかつ必要な情報の発生時間は短い。iの特徴を選ぶ。

5.3 クラス分け

場合分けに基づきクラス分けを行うことで、各センサや機器得られる各センサ情報や機器の制御情報について各ストリーミング配信の優先度を設定する。各センサ情報や機器の制御情報の情報の変化頻度と情報の重要性を評価して、ストリーミング配信の停止に対応できる情報の優先度が低くなるように優先度を設定する。

5.3.1 想定する環境のシステムでのクラス分け

情報の変化頻度と情報の重要性を評価について説明する。情報の変化頻度で A の特徴を持つ場合は、B の特徴と比べて情報の変化が頻繁に起きる。つまり、なるべく逃さずそのデータの変化を受信側に伝えたい。そのため、A の特徴のほうが B の特徴よりも優先度が高い。

情報の重要性で i の特徴を持つ場合は、ii の特徴と比べてストリーミング配信の停止に対応しにくい。本論で提案するアドミッション制御では、ストリーミング配信の停止をする場合がある。そのため、ストリーミング配信の停止に対応しがたい、i の特徴のほうが ii の特徴よりも優先度は高くなる。

これらの優先度の設定方法は、システムの動作になるべく影響がでないことを大前提として、その中でストリーミング配信を停止してもなんとかなる情報ほど優先度を低く設定しアドミッション制御することで、同時にストリーミング配信を行う機器数を減らそうという考えに基づいている。そのため、A i の特徴がもっとも優先度が高く、B ii の特徴がもっとも優先度が低くなる。一例として提案したシステムにおけるクラス分けを図 5.4 で示す。

クラス分けにより、各センサや機器からの各センサ情報や機器の制御情報のストリーミング配信に優先度を設定できる。各センサや機器が共通のルールに従い、ストリーミング配信の開始及び停止を決定することで、各センサや機器はストリーミング配信時に、ポリシーに従ったアドミッション制御を行することができる。当然、優先度が高いほど情報はストリーミング配信は継続されやすく、優先度の低くなるに従い、情報はストリーミング配信を停止しなければならない場合が起きやすくなる。

場合分け の条件	優先度	想定する環境で 利用する情報
A [∧] i	3 (高い)	位置情報 制御情報
B [∧] i	2	人の動き検知
A [∧] ii	1	
B [∧] ii	0 (低い)	室内の温度 室内の照度 室内の音

図 5.4: 想定する環境のシステムでのクラス分け

5.4 ポリシーに従ったアドミッション制御の手順

場合分け，そしてクラス分けを経て，各センサや機器からの各センサ情報や機器の制御情報のストリーミング配信に優先度を付けることができた．しかし，その優先度を利用して，ポリシーに従ったアドミッション制御を行うためには，各機器間で統一されたアドミッション制御のルールを決定する必要がある．本論でのアドミッション制御とは，動作を行うか，行わないかについて制御することをいう．そのため，ある状況においては，機器 A は動作を行うが，機器 B は動作を行わないといったルールを優先度に基づき決定すればよい．アドミッション制御のルールの例として，一例として作成するシステムでのポリシーに従ったアドミッション制御の手順を説明する．

5.4.1 想定する環境のシステムでの手順

想定する環境のシステムにおいて，802.11 無線 LAN を用いることで，各センサや機器からのセンサ情報や機器の制御情報は，室内に存在するすべての機器にブロードキャストされている．また，場合分け，クラス分けを経て設定されたストリーミング配信の優先度は，優先度が低い順に，0,1,2,3 といった数字を優先度の情報として，RTP のペイロードにテキストデータとして XML で記述してある（第 4 章参照）．そのため，各センサや機

器は、室内にどんな優先度が設定されたストリーミング配信が行われているのか知ることができ、これにより、室内に存在する各センサや機器は、ポリシーに従ったアドミッション制御が可能になる。

想定する環境のシステムでのポリシーに従ったアドミッション制御のルール

アドミッション制御の共通のルールは次のように設定する。このルールに従ってアドミッション制御を行う

1. まずはいつもと同じように、室内のすべてのセンサや機器がストリーミング配信を開始する。
2. ストリーミング配信開始から 5 秒後には、室内のすべてのセンサや機器に、空間のストリーミング配信の優先度の状況が伝わると想定する。アドミッション制御を開始する。
3. クラス分けで、優先度がもっとも高い、優先度 3 と設定された、位置情報と制御情報はストリーミング配信を続ける。
4. クラス分けで、優先度が高い、優先度 2 と設定された、人の動き検知情報もストリーミング配信を続ける。
5. クラス分けで、優先度がもっとも低い、優先度 0 と設定された、温度情報、照度情報、音情報の 3 つの情報に関してアドミッション制御を行う。この 3 つの情報が同時にストリーミング配信を行わないようにする。
6. 温度情報、照度情報、音情報をストリーミング配信する各センサは、1 秒おきに交代してストリーミング配信を行う。つまり、それぞれ最大 2 秒間はストリーミング配信が行われなくなる。

このようなアドミッション制御のルールに従って各センサや機器が、ポリシーに従ったアドミッション制御を行うことで、ネットワーク全体の通信量を減らすことができる。

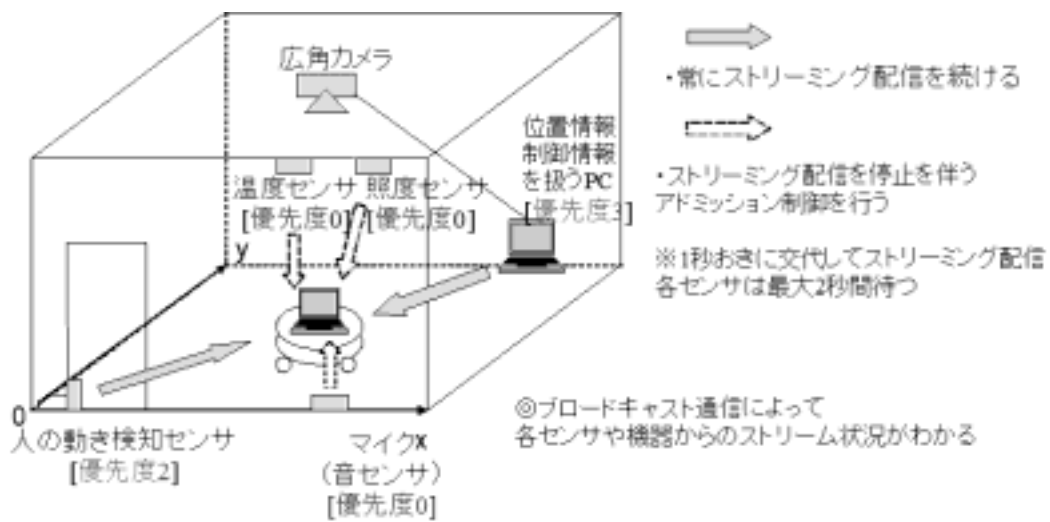


図 5.5: 想定する環境のシステムでのポリシーに従ったアドミッション制御

第6章 評価

本章では、機器間通信・制御におけるストリーミング配信の具体例として、想定する環境のシステムについてのシミュレーション実験を行い、シミュレーションの結果を示す。想定する環境のシステムのシミュレーションの結果から、ポリシーに従ったアドミッション制御を行うことが、リアルタイムな情報の配信による協調動作を行う上で、妥当な方法であるのかを評価する。

本シミュレーション評価では、ポリシーに従ったアドミッション制御を行う場合と行わない場合について、以下の項目を対象として評価を行う。

評価1 毎秒ごとに自走可能な掃除ロボットが受信するデータ量の違い

評価2 毎秒ごとに自走可能な掃除ロボットが受信する情報数（パケット数）の違い

6.1 評価環境

単体PC上で一例として提案したシステムを、ns-2(Network Simulator version 2)[14]を用いて仮想的に構築した上で、シミュレーション評価を行う。シミュレーションを行ったPCのスペックは表 6.1 で示す。

表 6.1: シミュレーションを行ったPCのスペック

CPU	AMD Duron 900MHz
メモリ	256MB
OS	Vine linux 2.6r1(kannel-2.4.20)
ns-2 のバージョン	ns-2 ver2.26

6.1.1 ns-2

想定する環境のシステムを評価する際、実機を用いて提案するシステムを構成し、提案手法を実装し、データを流す実験を行い、ネットワーク上でデータの実測し、得られた

データから提案するシステムを検証することは重要である。しかし、実機による提案するシステムの実験では再現性がないことから、得られたデータを検証することが難しい。そこで、ネットワークをシミュレートするシミュレータとして用いることが出来る ns-2 を用いて想定する環境のシステムを検証を行うことで、想定する環境のシステムを評価する。

ns-2 は、カリフォルニア大学バークレー校で開発されたネットワークに関する研究のための離散イベント型のシミュレータである。プロトコル部分は C++ 言語、インターフェース部分は otcl(Objecttcl) 言語で実装され、物理層、データリンク層、トランスポート層、及び、アプリケーション層における様々なプロトコルに対応している。また、複数のプラットフォーム (FreeBSD, Linux, Solaris, Windows) に対応している。ns-2 は、グラフ作成ソフトである Xgraph を併用することで、評価結果をグラフ表示できる。また、nam (Network AniMator) を用いると、移動体の移動やパケットの伝播状態を記録したログから、ネットワークトポロジの変化やパケットの伝播状態をアニメーション表示できる。

6.2 ポリシーに従ったアドミッション制御の評価

想定する環境のシステムでのポリシーに従ったアドミッション制御について、ポリシーに従ったアドミッション制御を行う場合及び行わない場合におけるシミュレーション実験を行い、お互いを比較することでポリシーに従ったアドミッション制御の評価する。

6.2.1 ポリシーに従ったアドミッション制御を行わない場合

ポリシーに従ったアドミッション制御を行わない場合における、一例として提案システムのシミュレーションモデルについて説明する。ポリシーに従ったアドミッション制御を行わない場合のシミュレーションで設定した項目とその内容について表 6.2 で一覧を示す。

- 各センサや機器、及び自走可能な掃除ロボットを表すノードを合計で 6 つのノードを生成し、802.11 無線 LAN (帯域 2Mbps) で IP ネットワークを構築する。
- 10m × 10m の正方形の部屋を想定し、センサや機器の各ノードは、自走可能な掃除ロボットのノードから 2m 以内の距離の十分に近い位置に配置する。これにより、802.11 無線 LAN の電波が届かなくなることがなく、トポロジーも変化しない状況となる。
- 各センサや機器を想定した各ノードが送出する 1 パケットは、20 バイトの IP ヘッダ、8 バイトの UDP ヘッダ、12 バイトの RTP ヘッダを共通に持つ。また RTP ペイロードは、各センサが持つ XML 記述のテキストデータ文字数ごとに文字数 × バイトとする。ここではそれぞれ 100 バイトとする。また、広角カメラに接続した PC に関しては、位置情報と機器の制御情報を 1 パケットで同時に含むので、XML 記

述のテキストデータ文字数約 150 文字であるとし，RTP ペイロードは 150 バイトとする．

- ストリーミング配信を行うために，アドミッション制御を行わない場合は，各センサや機器を想定した各ノードは UDP/IP で，RTP パケットを 1 秒間に 10 パケットを送出する．
- 各センサや機器は自走可能な掃除ロボットを想定するノードに向けて，シミュレーション開始から 0.1 秒後にストリーミングを開始する．
- 以上の設定で 60 秒間のシミュレーションを行う．

表 6.2: シミュレーションの内容 (ポリシーに従ったアドミッション制御を行わない場合)

設定項目	内容
配置するノード数	6
構築する IP ネットワーク	802.11 無線 LAN(2Mbps)
構築する部屋の広さ	10m × 10m
各センサの 1 パケットでのデータ量	240 バイト
広角カメラに接続された PC の 1 パケットでのデータ量	290 バイト
各センサや機器が 1 秒間に送出手数	10 パケット
ストリーミングの開始時間	シミュレーション開始から 0.1 秒後
シミュレーション時間	60 秒

6.2.2 ポリシーに従ったアドミッション制御を行う場合

ポリシーに従ったアドミッション制御を行う場合における，一例として提案システムのシミュレーションモデルについて説明する．ポリシーに従ったアドミッション制御を行う場合のシミュレーションで設定した項目とその内容について表 6.3 で一覧を示す．一例として提案システムの基本となるシミュレーションモデルについて説明する．

ポリシーに従ったアドミッション制御を行う場合のシミュレーションモデル

- 各センサや機器，及び自走可能な掃除ロボットを表すノードを合計で 6 つのノードを生成し，802.11 無線 LAN (帯域 2Mbps) で IP ネットワークを構築する．

- 10m × 10m の正方形の部屋を想定し，センサや機器の各ノードは，自走可能な掃除ロボットのノードから 2m 以内の距離の十分に近い位置に配置する．これにより，802.11 無線 LAN の電波が届かなくなることがなく，トポロジも変化しない状況となる．
- 各センサや機器を想定した各ノードが送出する 1 パケットは，20 バイトの IP ヘッダ，8 バイトの UDP ヘッダ，12 バイトの RTP ヘッダを共通に持つ．また RTP ペイロードは，各センサが持つ XML 記述のテキストデータ文字数は平均は約 100 文字であるとし，それぞれ 100 バイトとする．また，広角カメラに接続した PC に関しては，位置情報と機器の制御情報を 1 パケットで同時に含むので，XML 記述のテキストデータ文字数約 150 文字であるとし，RTP ペイロードは 150 バイトとする．
- アドミッション制御を行う場合は，第 5 章で設定したアドミッション制御のルールに従い，各センサや機器を想定した各ノードは UDP/IP で，RTP パケットを温度・照度・音情報を想定する 3 つのノードは，それぞれ 1 秒おきに交代してストリーミング配信を行う．3 つの各ノードは最大 2 秒間待つことになる．この 3 つのノードは RTP パケットを 1 秒間に 2 パケットを送出する．また，その他の位置・制御情報，人の動き検知情報を想定する 2 つのノードについては，RTP パケットを 1 秒間に 10 パケットを送出する．
- 各センサや機器は自走可能な掃除ロボットに向けて，シミュレーション開始から 0.1 秒後にストリーミングを開始する．シミュレーション開始から 5 秒後にアドミッション制御を開始する．
- 以上の設定で 60 秒間のシミュレーションを行う．

6.2.3 測定方法

毎秒ごとに自走可能な掃除ロボットが受信するデータ量の違い

各センサや機器から自走可能な掃除ロボットにストリーミング配信が行われたとき，自走可能な掃除ロボットが 1 秒間ごとに受信する総データ量を算出する．横軸にシミュレーション時間，縦軸に自走可能な掃除ロボットが 1 秒間に受信する総データ量を取り，シミュレーション終了までにおける，毎秒ごとに自走可能な掃除ロボットが受信する総データ量をグラフにまとめる．これにより，アドミッション制御を行わない場合と行う場合における，毎秒ごとに自走可能な掃除ロボットが受信するデータ量の違いを測定する．

毎秒ごとに自走可能な掃除ロボットが受信する情報数（パケット数）の違い

表 6.3: シミュレーションの内容 (ポリシーに従ったアドミッション制御を行う場合)

設定項目	内容
配置するノード数	6
構築する IP ネットワーク	802.11 無線 LAN(2Mbps)
構築する部屋の広さ	10m × 10m
各センサの 1 パケットのデータ量	240 バイト
広角カメラに接続された PC の 1 パケットのデータ量	290 バイト
位置・制御情報と人の動き検知情報を 1 秒間に送出するパケット数	10 パケット
温度・照度・音情報を 1 秒間に送出するパケット数	2 パケット
ストリーミングの開始時間	開始から 0.1 秒後
アドミッション制御の開始時間	開始から 5 秒後
シミュレーション時間	60 秒

各センサや機器から自走可能な掃除ロボットにストリーミング配信が行われたとき、自走可能な掃除ロボットが 1 秒ごとに受信した総パケット数を各センサや機器ごとに測定する。各センサや機器からストリーミング配信される 1 パケットは 1 タイミングの情報のみを含むので、毎秒ごとの総パケット数は毎秒ごとに得られる総情報数に相当する。横軸にシミュレーション時間、縦軸に自走可能な掃除ロボットが 1 秒間に受信する総パケット数をとり、シミュレーション終了までにおける自走可能な掃除ロボットが毎秒ごとに受信した総パケット数を各センサや機器ごとにそれぞれグラフにまとめる。これにより、アドミッション制御を行わない場合と行う場合における、毎秒ごとに自走可能な掃除ロボットが受信する情報数の違いを測定する。

6.2.4 測定結果

測定結果のグラフはそれぞれ以下ようになった。

グラフ 6.1 は毎秒ごとに自走可能な掃除ロボットが受信するデータ量の違いを示す。また、毎秒ごとに自走可能な掃除ロボットが受信する情報数 (パケット数) の違いを示すグラフはそれぞれ、グラフ 6.2 は位置・制御情報の情報数、グラフ 6.3 は人の動き検知情報の情報数、グラフ 6.4 は温度情報の情報数、グラフ 6.5 は照度情報の情報数、最後にグラフ 6.6 は音情報の情報数とした。

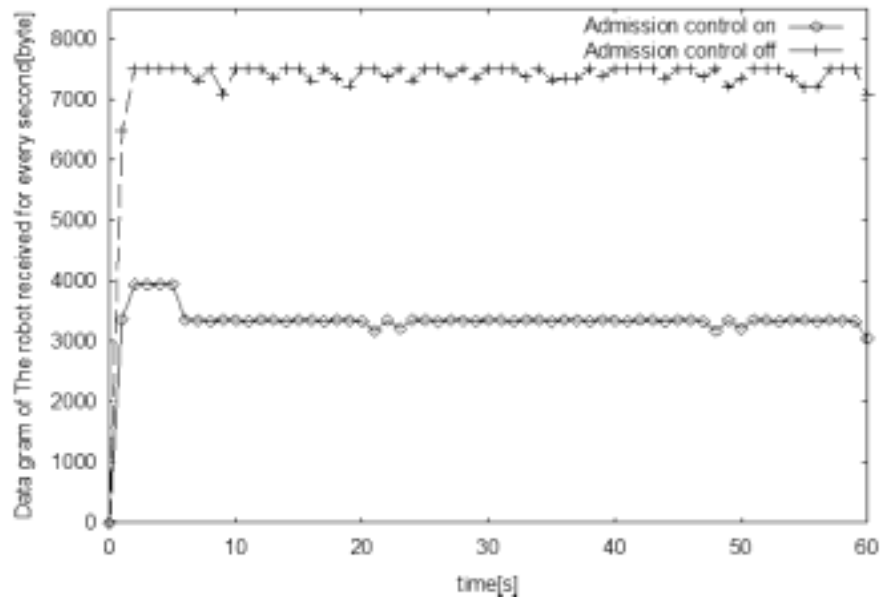


図 6.1: 毎秒ごとに自走可能な掃除ロボットが受信するデータ量の違い

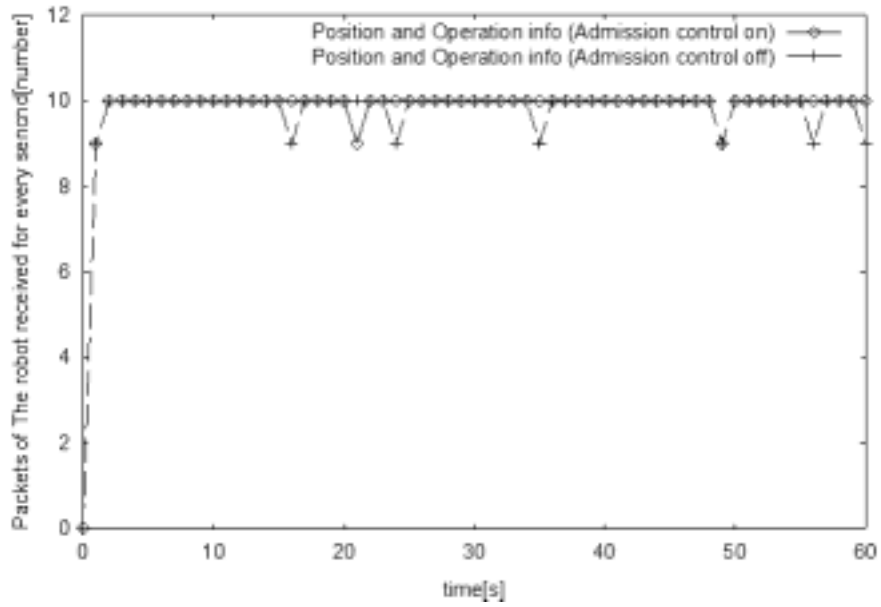


図 6.2: 毎秒ごとに自走可能な掃除ロボットが受信する情報数の違い (位置・制御情報)

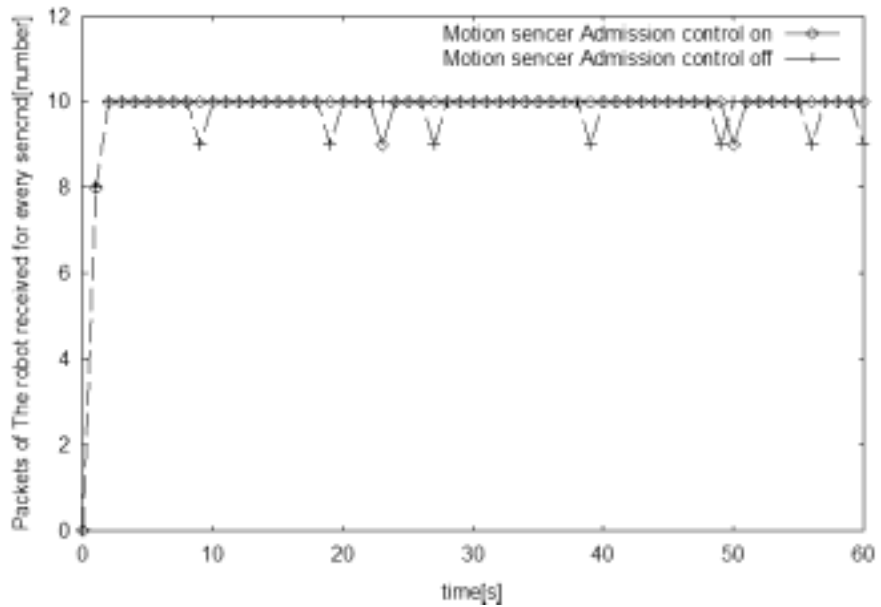


図 6.3: 毎秒ごとに自走可能な掃除ロボットが受信する情報数の違い (人の動き検知情報)

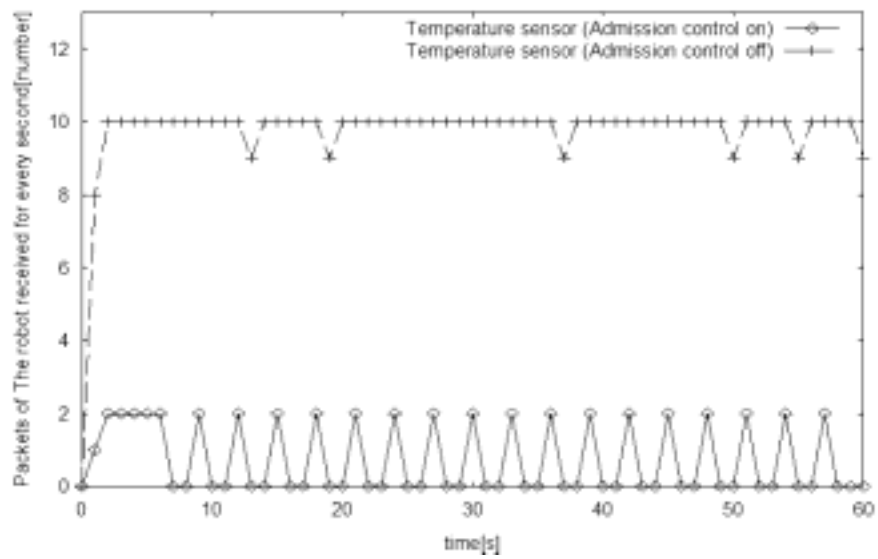


図 6.4: 毎秒ごとに自走可能な掃除ロボットが受信する情報数の違い (温度情報)

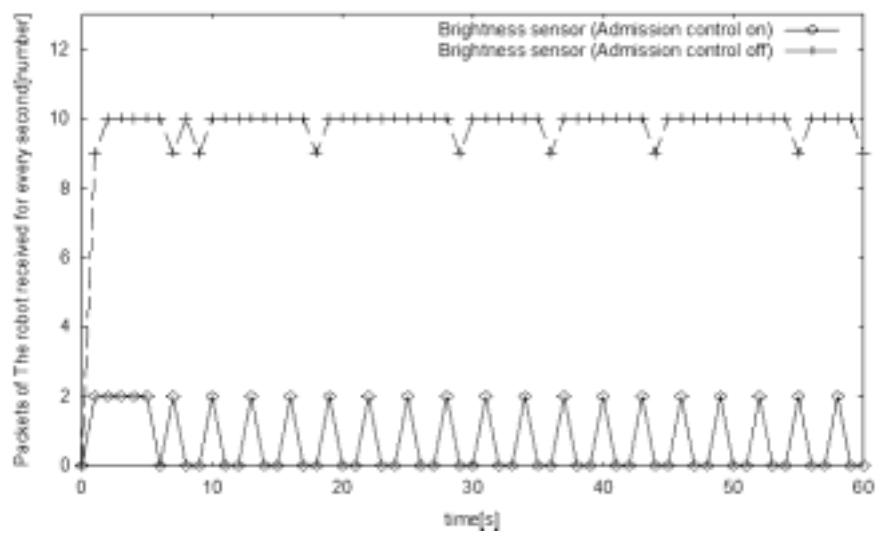


図 6.5: 毎秒ごとに自走可能な掃除ロボットが受信する情報数の違い (照度情報)

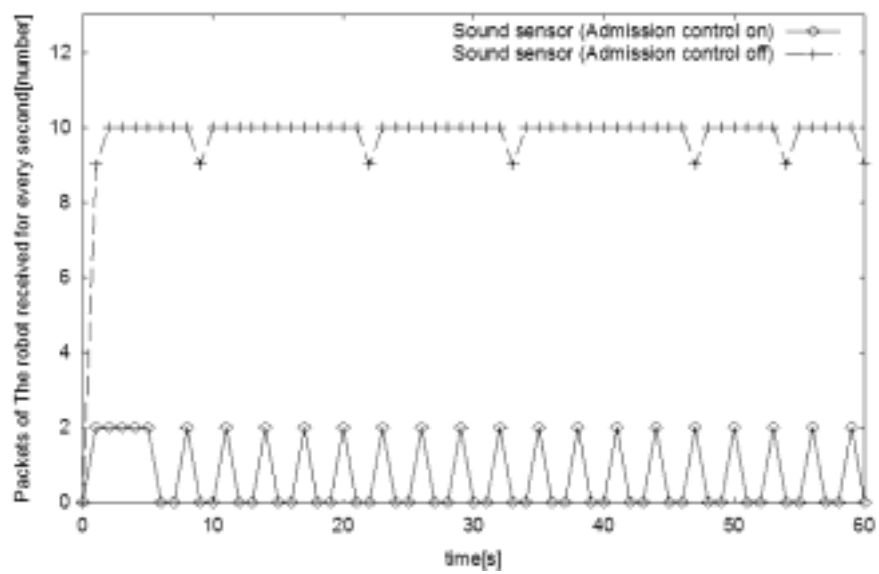


図 6.6: 毎秒ごとに自走可能な掃除ロボットが受信する情報数の違い (音情報)

6.3 シミュレーション結果についての評価と考察

毎秒ごとに自走可能な掃除ロボットが受信するデータ量の違いについての評価と考察

グラフ 6.1 の通り，アドミッション制御を行うことで，アドミッション制御を行わない場合と比べて，自走可能な掃除ロボットが毎秒受信する総データ量を減少させることができた．このことは，輻輳を事前に回避することにつながると考えられる．よって自走可能な掃除ロボットに対して行われているストリーミング配信に対して，アドミッション制御を行うことは有用であるといえる．

毎秒ごとに自走可能な掃除ロボットが受信する情報数（パケット数）の違いについての評価と考察

グラフ 6.2 とグラフ 6.3 で示したように，位置・制御情報を扱う PC と人の動き検知センサを想定したノードはアドミッション制御を行う対象ではないので，情報数に変化はない．よってアドミッション制御を行うことによって，情報数が足りなくなってしまうようなことは起きない．

また，温度・照度・音の情報に関しては，グラフ 6.4 とグラフ 6.5 とグラフ 6.6 で示すようにアドミッション制御を行うことで自走可能な掃除ロボットが毎秒ごとに得られるパケット数が減少してしまう．このことは毎秒ごとに自走可能な掃除ロボットが得られる情報数の減少を意味する．しかし，毎秒ごとに得られるセンサからの情報の数が減少することになるが，アドミッション制御の対象となった各センサを想定したノードは各グラフで示すように 60 秒間に約 40 の情報は確保できているので十分は情報数だと考えるので，アドミッション制御を行うことで絶対的な情報数が足りなくなってしまう心配は少ないと考えられる．つまり，アドミッション制御を行うことは，情報数の面から見ても十分に有用であるといえる．

第7章 おわりに

7.1 まとめ

本論では、ストリーミング技術がリアルタイムな情報の配信を可能にする点に注目することで、空間に存在するセンサや機器から得られるセンサ情報や機器の制御情報など、様々なデータを扱えるストリーミング技術を用いて、刻々と変化する空間のリアルタイムな状況や環境の情報をその情報を必要とするユーザや機器に、リアルタイムにストリーミング配信することができれば、機器の協調動作を実現できると考え、機器が無線 LAN で接続された室内などの限られた空間を対象として、映像・音声のみではなくセンサ情報や機器の制御情報をストリーミング配信を用いて扱うことで、機器間でのリアルタイムな情報の配信、及び協調動作の実現することを目的とした。この目的を実現するために、ストリーミング技術に関する新たな提案を行った。センサ情報や機器の制御情報を扱えるストリーミング技術として、センサ情報や機器の制御情報をテキストデータとしてストリーミングする方式を新たに提案し、定義した。また、ユビキタス環境など、無数のセンサや機器が存在する空間で、本論で新たに提案した、センサ情報や機器の制御情報をテキストデータとしてストリーミングする方式を適応したとき起こり得る輻輳問題のひとつの解決法として、ポリシーに従ったアドミッション制御を行うことを新たに提案した。新たに提案したセンサ情報や機器の制御情報をテキストデータとしてストリーミングする方式について、シミュレーション実験を行い、その結果に基づき評価を行った。その結果、本論で提案したセンサ情報や機器の制御情報のストリーミング配信方式は有用であることがわかった。

7.2 今後の課題

今後の課題として以下の点をあげる。

- RTP だけでなく RTCP を用いたストリーミング配信

本論では、IP ネットワーク上で RTP を用いてストリーミング配信を行うことを提案したが、RTP のみではなく、RTP を補助するためのプロトコルである、RTCP もいっしょに用いることで、RTP の機能だけでは不可能だった機器間のクロック同期が可能になり、ストリーミング配信の送信側のセンサや機器と受信側の機器とでのセンサ情報や機器の

制御情報の利用のタイミングがずれることを防ぐことができるなど、より確実なストリーミングが可能になると考えられる。

- 想定する環境のシステムの実装に伴う、想定した協調動作の評価

想定する環境のシステムを実装するためには、第4章の図4.11で示したセンサ情報や機器の制御情報をストリーミングするための手順を可能にするプログラム及び機構を作成する必要がある。また、自走可能な掃除ロボットや各センサや機器について、ハードウェアの改良も必要となる。

想定する環境のシステムを実装することで、第3章で想定した協調動作のシナリオを実際に試すことができる。例えば、自走可能な掃除ロボットにとって、想定した協調動作のトリガーとなる情報が各センサや機器で発生した場合、その情報が発生した時間からなるべく遅れないタイミングで、自走可能な掃除ロボットに届いているかどうかの測定するなど、想定する環境のシステムの実用性を評価することができる。

- ポリシーに従ったアドミッション制御の方法をより詳しく考察する必要性

本論で提案したポリシーに従ったアドミッション制御は、802.11無線LANを利用したブロードキャスト通信により、空間に存在する各センサや機器が、空間内で行われているストリーミング配信をすべて受信することで、空間内のストリーミング配信の状況および優先度を知ることができた。しかし、空間内に存在するセンサや機器の数が増加した場合にもブロードキャスト配信を続けることは難しいと考えられる。そのため、空間の状況に応じて、ブロードキャスト、マルチキャスト、ユニキャストを駆使した通信を行う必要が出てくる可能性が考えられる。また、実際にブロードキャストにより各センサや機器が空間内のストリーミング配信の状況が把握できたとしても、それに基づき各センサや機器が分散合意によりポリシーに従ったアドミッション制御を行うことはとても難しい問題である。そのため、ポリシーに従ったアドミッション制御の方法に関しては、集中制御を行うことも考えつつ、詳細な考察を行う必要がある。

詳細な考察を行った上で、ユビキタス環境での適応を前提として、大多数のセンサや機器が存在する空間で、センサ情報や機器の制御情報をストリーミング配信する場合の、輻輳の可能性とポリシーに従ったアドミッション制御の有効性を評価する必要がある。

謝辞

本研究を行うにあたり、御指導御鞭撻を戴いた北陸先端科学技術大学院大学丹 康雄 教授に深く感謝致します。

また、サブテーマで御指導を戴いた野田 五十樹 助教授に厚く御礼申し上げます。

最後に、日頃よりお世話になった丹研究室の皆様に厚く御礼申し上げます。

参考文献

- [1] Weiser, M., The Computer for the 21st Century, ScientificAmerican, 265 (3), pp.94-104, 1991.
- [2] RealNetworks, RealOnePlayer, <http://www.real.com/>.
- [3] Microsoft, WindowsMediaPlayer, <http://www.microsoft.com/windows/windowsmedia/>.
- [4] Apple, QuickTimePlayer, <http://www.apple.com/quicktime/>.
- [5] Shulzrinne,H., Casner,S., Frederick,R., and Jacobson,V., RTP:A Transport Protocol for Real-Time Application, RFC1889, 1996.
- [6] Shulzrinne,H., RTP Profile for Audio and Video Conferences with Minimal Control, RFC1890, 1996.
- [7] RTP PARAMETERS, <http://www.iana.org/assignments/rtp-parameters>.
- [8] Turetti,T.,Huitema,C., RTP Payload Format for H.261 Video Streams, RFC2032, 1996.
- [9] ITU-T: TELECOMMUNICATION STANDARDIZATION SECTOR OF ITU, Dual rate speech coder for multimedia communications transmitting at 5.3 and 6.3kbit/s, G.723.1, 1996.
- [10] Roomba, <http://www.roombavac.com/buyroomba/defaultB.asp>.
- [11] Hellstrom,G., RTP Payload for Text Conversation, RFC2793, 2000.
- [12] ITU-T: TELECOMMUNICATION STANDARDIZATION SECTOR OF ITU, Protocol for multimedia application text conversation, T.140, 1998.
- [13] Bray,T., Paoli,J., Sperberg-McQueen,C.,and Maler,E., Extensible Markup language(XML)1.0(Second Edition), <http://www.w3.org>, 2000.
- [14] The Network Simulator - ns-2, <http://www.isi.edu/nsnam/ns/>.

- [15] 大久保榮, 川島正久 監修, MCR 編, H.323/MPEG-4 教科書, IE インスティテユート, 2001.
- [16] Kosiur, D. 著, 荻田幸雄 監訳, マスタリング TCP/IP IP マルチキャスト編, オーム社, 1999.
- [17] The ns Manual, <http://www.isi.edu/nsnam/ns/ns-documentation.html>.