

# MMT を用いた端末間映像同期の Android 端末への実装

河村侑輝 大槻一博 遠藤洋介

試験放送が開始された 4K・8K 衛星放送では、IP 上のメディアトランスポート方式である MMT(MPEG Media Transport) が用いられる。MMT は映像・音声の提示タイムスタンプとして絶対時刻を用いるため、放送とインターネットなど伝送遅延の異なる伝送路から受信した複数の映像・音声を高精度に同期させることが可能であり、これを活かした新たなサービスの実現が期待される。本稿では、MMT で配信された映像・音声を端末間で高精度に同期させる機能を Android 端末向けのアプリケーションとして実装し、応用サービスの可能性を検討する。

## Implementation of Inter-Terminal Video Synchronization Using MMT on Android

YUKI KAWAMURA KAZUHIRO OTSUKI YOSUKE ENDO

In the UHDTV satellite broadcasting, MMT (MPEG Media Transport) has been adopted as the IP-based media transport scheme. Since MMT utilizes the UTC timestamp for video/audio presentation, multiple videos from different paths including broadcasting and the Internet can be presented in a synchronized manner. Then, the expectations for the new services utilizing this feature are increasing. In this paper, we implement the inter-terminal video synchronization using MMT on Android and discuss the possibilities of application services.

### 1. はじめに

2016年8月に試験放送が開始された4K・8K衛星放送では、映像・音声の符号データや制御情報を多重して伝送するためのメディアトランスポート方式としてMMT(MPEG Media Transport) [1]が用いられる。MMTは、IP(Internet Protocol)上のプロトコルであるMMTP(MMT Protocol)を規定しており、放送とインターネットとを区別することなく利用できる。このため、4K・8K衛星放送では、試験放送で既に実現した超高精細度映像の放送サービスに加え、2018年の実用放送や2020年の本格普及に向けて、高度な放送・通信連携サービスの実現が期待されている。

筆者らはこれまで、ワークステーションを用いて8Kテレビを想定したMMT受信装置の実装・検証を行い、4K・8K衛星放送の標準規格[2]および運用規定[3]の策定に寄与してきた。このMMT受信装置は、通信を用いない放送単独サービスの検証に加え、通信を用いて放送の主映像と同期する付加映像を配信する放送・通信連携サービスの検証にも対応する。具体的には、通信から受信した付加映像を放送映像上の子画面に合成し表示する機能や、放送映像の一部を付加映像に差し替える機能[4]を有する。これらは、8Kテレビ単体が放送・通信の両方からコンテンツを受信し、それらを組み合わせて利用するサービスを想定していた。一方、8Kテレビとインターネットに接続される多様なモバイル端末とを、端末間で連携させるサービスの実現も望まれている。例えば、図1に示すように、複数人が8Kテレビの主映像・主音声を視聴する環境で、各個人が好みに合

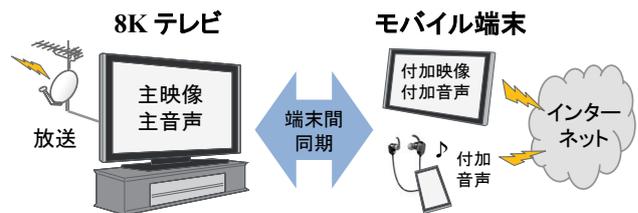


図1 端末間同期のイメージ

Figure 1 An image of inter-terminal synchronization.

わせた付加映像や付加音声をインターネットから受信し、手元のモバイル端末で同期視聴できる端末間同期サービスが考えられる。現状、MMTを用いたインターネット単独での映像配信サービスが商用化された例[5]はあるが、伝送遅延が異なる複数の映像・音声を複数の端末で同時に受信し、それらの端末間で映像・音声を高精度に同期できるMMT受信機能を実装し、検証した報告はなされていない。

そこで本稿では、モバイル端末向けプラットフォームとして広く普及しているAndroidを搭載する端末を対象に、端末間での映像・音声の高精度同期機能を備えたMMT受信アプリケーションを実装する。さらに、市販のモバイル端末を用いた実験により、伝送遅延が異なる複数の映像・音声を複数の端末で受信する場合であっても、端末間で高精度な同期提示が可能であることを確認する。また、放送に加えて、デジタルサイネージなど他分野に視野を広げて、MMTを用いた応用サービスの可能性について検討する。

## 2. 放送のメディアトランスポート方式

### 2.1 MPEG-2 TS (Transport Stream)

従来のデジタル放送システムでは、映像・音声の多重化方式として、20年以上前に標準化された MPEG-2 TS [6]が用いられている。TS では、188 バイトの固定長パケットに映像・音声の符号データや制御情報を細かく分割して格納し、シリアル信号に多重する。さらに、送信装置が生成する 27 MHz の基準クロックである STC (System Time Clock) のサンプル値が、PCR (Program Clock Reference) として信号中に多重される。受信端末は、この PCR をもとに STC を再生し、基準クロックとして用いる。受信端末が正しく STC を再生して破綻なく動作するためには、伝送遅延が固定であることが必要であり、PCR の許容ジッタは ±500 ns と規定される。

TS では映像・音声の提示タイムスタンプ (PTS, Presentation Time Stamp) が、STC のサンプル値として与えられる。ただし、STC はそれぞれの送信装置が生成するクロックであり、ある瞬間の STC は送信装置ごとに異なる値をもつ。つまり、送信元が異なる映像・音声は、PTS の基準となる STC が異なるため、そのままでは同期を合わせることができない。STC が異なる複数 TS の映像・音声を同期させるためには、STC 間の相対関係を示す補助情報を別途生成して受信端末に与える仕組みが必要となる。

また、TS をインターネット上で配信する際には、IP 上のプロトコルである RTP (Real-time Transport Protocol)/UDP (User Datagram Protocol) のパケットにカプセル化する必要がある。また、遅延変動が起こり得るインターネット上でも固定遅延を担保する必要があるため、各 TS パケットの先頭に配信タイムスタンプを付与する TTS (Timestamped TS) 形式を用い、受信端末側のバッファリングによる伝送遅延ジッタの解消が必要となる。

### 2.2 MPEG-H MMT (MPEG Media Transport)

MMT は、インターネットが広く普及した現在のコンテンツ配信環境に適する IP 上の新たなメディアトランスポート方式として、2009 年から MPEG での標準化が行われ、2014 年に国際規格として承認された。

MMT と TS の比較を表 1 に示す。MMT では、伝送路の MTU (Maximum Transmission Unit) を最大とする可変長の IP パケットに映像・音声の符号データや制御情報を格納する。IP の下位層が Ethernet の場合、MTU は 1500 バイトが標準である。また、基準クロックの伝送は規定せず、送信装置と受信端末それぞれが、NTP (Network Time Protocol) サーバ等から得た協定世界時 (UTC, Coordinated Universal Time) を基準に動作することを基本とする。

MMT では映像・音声の PTS が、UTC のサンプル値として与えられる。UTC は、送信装置が異なっても一致する絶対的な基準である。そのため、MMT に多重された映像・

表 1 TS と MMT の比較

Table 1 Comparison between TS and MMT.

比較項目	MPEG-2 TS	MPEG-H MMT
パケット長	188 バイトの固定長	MTU を上限とする可変長
基準クロック	System Time Clock (STC)	Coordinated Universal Time (UTC)
クロックの伝送	STC のサンプル値 (PCR) を信号に多重して伝送する	クロックの伝送を規定しない
映像・音声の PTS	STC のサンプル値 (符号データに付加)	UTC のサンプル値 (制御情報に記載)
放送時の形式	TS	MMTP/(圧縮 UDP/IP)/TLV
インターネット配信時の形式	TTS/RTP/UDP/IP	MMTP/UDP/IP

音声は、送信元が異なる場合であっても、補助情報を必要とせずに PTS を用いて同期を合わせることができる。

MMT では、独立して復号処理が可能な符号データの集合を MPU (Media Processing Unit) と呼び、MPU ごとに、最初に提示するフレームの PTS を制御情報に記載して伝送する。映像符号化の場合、他フレームを参照せずに復号処理を開始できるイントラフレームを復号順の先頭とする複数フレームの集合を MPU とする。MPU を構成する符号データをさらに細分化した単位を MFU (Media Fragment Unit) と呼び、IP パケットに格納するデータの基本単位となる。HEVC (High Efficiency Video Coding) の場合、NAL (Network Abstraction Layer) ユニットの MFU としてデータサイズに応じた結合または分割を行い、MTU を上限とする IP パケットに格納する。

4K・8K 衛星放送の伝送方式[7]は TLV (Type Length Value) による IP パケットの伝送に対応するため、放送とインターネットとで共通のパケット形式として、MMTP/UDP/IP パケットを用いることができる。このため、インターネットによる放送の再送信や、放送とインターネット上の映像配信とでの受信機能の共用化が TS に比べて容易である。なお、TLV は UDP/IP ヘッダの圧縮機能をもつため、放送時はオーバーヘッドを削減した効率的な伝送が可能である。

このように、MMT は多様な伝送路や多様な受信端末が存在する現在のコンテンツ配信環境に適したメディアトランスポート方式であり、今後、放送・通信の垣根を越えた新たなサービスが生まれることが期待される。

### 3. 複数伝送路配信映像の端末間同期

#### 3.1 MMT の提示制御

送信装置における映像・音声の MPU への PTS の付与方法として、下記の 2 つの方法が考えられる。

##### ① 未来の時刻を指定する

送信装置が、全ての受信環境における伝送遅延時間と処理遅延時間を考慮し、ある MPU を送信する時刻の UTC に対してこれらの遅延時間の最大値を加算した値、つまり、送信装置にとって未来の時刻をその MPU の PTS として指定する。この場合、受信端末は、PTS で指定された通りの時刻にその MPU を提示する。

##### ② MPU を送信する時刻を指定する

送信装置は、受信環境の伝送遅延時間や処理遅延時間を考慮せず、ある MPU を送信する時刻の UTC を、その MPU の PTS として指定する。この場合、受信端末にとって PTS は過去の時刻となり、PTS で指定された通りの時刻に MPU を提示することはできない。このため、各受信端末は PTS で指定された時刻の UTC に対して、それぞれの受信環境に応じた伝送遅延時間と処理遅延時間の和（以降、PTS オフセットと呼ぶ）を加算した時刻に、その MPU を提示する。

方法①では、受信端末の時計が UTC に同期していれば、伝送遅延や処理遅延が異なる複数の端末間でも MPU の提示が同期する。しかし、放送・通信連携サービスの実用化においては、サービスの対象を特定の伝送路や端末に限定せず、あらゆる受信環境を想定することが求められる。したがって、伝送遅延時間と処理遅延時間の最大値を考慮する必要がある方法①の利用は、非現実的である。

一方、方法②では、各受信端末の時計が UTC に同期していても、端末間で PTS オフセットが異なる場合、図 2 に示すように MPU の提示は同期しない。方法②において複数端末間で MPU の同期提示を実現するためには、図 3 に示すように、各端末がそれぞれの PTS オフセットを決定した後、それらの中で最大の PTS オフセットを共通の PTS オフセット（以下、共通 PTS オフセットと呼ぶ）として用いることが考えられる。この場合、各端末は、共通 PTS オフセットと自身の PTS オフセットとの差分に応じてバッファリングを行うことで、端末間でも MPU の提示が同期する。このため、映像・音声を同期させたい複数の受信端末（以下、同期メンバ端末と呼ぶ）の中で最大の PTS オフセットを決めて、それを共通 PTS オフセットとして端末間で共有する手続きが必要である。

#### 3.2 提案手法

方法②を用いることを要求条件とし、同期メンバ端末が端末間通信を行うことで共通 PTS オフセットを自動的に決定し、端末間で共有する仕組みを提案する。

提案手法では、同期メンバ端末のうち 1 台をマスタ端末、

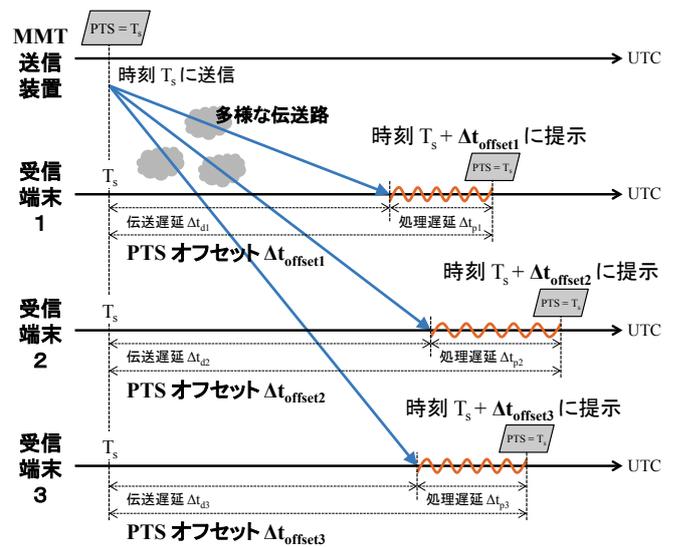


図 2 PTS オフセットが異なる場合の提示  
Figure 2 Presentation with different PTS offsets.

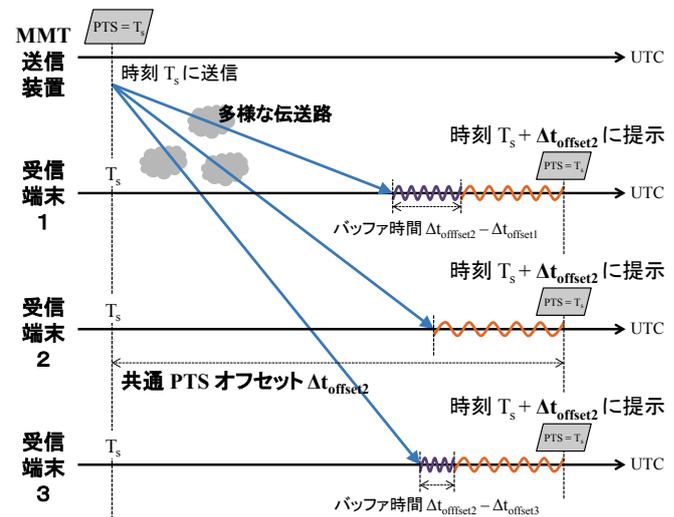


図 3 共通 PTS オフセットを用いた提示  
Figure 3 Presentation with a common PTS offset.

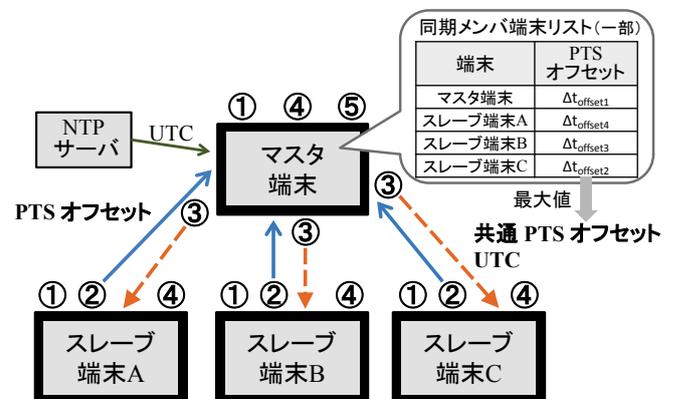


図 4 端末間通信の手続き  
Figure 4 Procedure of inter-terminal communication.

その他をスレーブ端末として動作させる。図 4 および下記に、端末間通信の手続きの概略を示す。

### ① PTS オフセットの決定 (全ての端末)

各受信端末の PTS オフセットは、伝送遅延時間と処理遅延時間をそれぞれ個別に測定し、それらの和から求めることができる。しかし、伝送遅延時間と処理遅延時間を個別に扱う必要はないため、実際に 1 つの MPU を最短時間で復号し、その MPU の先頭フレームが提示可能となった時刻とその MPU に付与されていた PTS とを比較することで直接 PTS オフセットを決定することができる。

### ② 同期パケットの送信 (スレーブ端末)

スレーブ端末は、マスタ端末に対して同期パケットを送信する。同期パケットのデータ構造を図 5 に示す。同期パケットには、ポーリング間隔、開始タイムスタンプ T1、受信中の MMT パッケージ (番組に相当) を識別する MMT パッケージ ID、PTS オフセットを記載する。開始タイムスタンプ T1 は、スレーブ端末が同期パケットを送信する瞬間の UTC とする。

### ③ 同期パケットの受信と返信 (マスタ端末)

マスタ端末は、端末内に同期メンバ端末リストを保持する。スレーブ端末から同期パケットを受信すると、同期パケットに記載された MMT パッケージ ID とマスタ端末が受信中の番組の MMT パッケージ ID の一致を確認する。一致する場合、スレーブ端末の IP アドレス、PTS オフセット、ポーリング間隔、開始タイムスタンプ T1 のセットを、同期メンバ端末リストに記録する。既に同じスレーブ端末のエントリが存在する場合には、これらの情報を更新する。次に同期メンバ端末リストを参照し、マスタ端末自身も含めた全ての同期メンバ端末の PTS オフセットの中から最大値を求め、これを共通 PTS オフセットとして決定する。その上で、同期パケットに共通 PTS オフセット、受信タイムスタンプ T2、送信タイムスタンプ T3 を記載してスレーブ端末に返信する。受信タイムスタンプ T2 は、マスタ端末が同期パケットを受信した瞬間の UTC、送信タイムスタンプ T3 はマスタ端末が同期パケットを返信する瞬間の UTC とする。マスタ端末は、共通 PTS オフセットが変更された場合、その値に応じたバッファリングを開始する。

同期パケットに記載された MMT パッケージ ID とマスタ端末が受信中の番組の MMT パッケージ ID が一致しない場合は、そのスレーブ端末を同期メンバ端末リストには加えない。また、同期パケットの PTS オフセットを共通 PTS オフセットに書き換えず、受信タイムスタンプ T2、送信タイムスタンプ T3 のみ記載して、スレーブ端末に返信する。

### ④ 同期パケットの受信 (スレーブ端末)

スレーブ端末は、マスタ端末から同期パケットを受信すると、同期パケットに記載された共通 PTS オフセットに従ったバッファリングを開始する。また、開始タイムスタンプ

0	16	24	31
Reserved	ポーリング間隔	Reserved	
Reserved (160 ビット)			
開始タイムスタンプ T1 (64 ビット)			
受信タイムスタンプ T2 (64 ビット)			
送信タイムスタンプ T3 (64 ビット)			
受信中の MMT パッケージ ID (32 ビット)			
PTS オフセット(スレーブ→マスタ) 共通 PTS オフセット(マスタ→スレーブ) (64 ビット)			

図 5 同期パケットのデータ構造

Figure 5 Data structure of the synchronization packet.

プ T1、受信タイムスタンプ T2、送信タイムスタンプ T3、スレーブ端末が同期パケットを受信した瞬間の UTC (T4) を用い、マスタ端末との時刻ずれを求めて自身の時計を補正する。スレーブ端末の時計のマスタ端末の時計に対する時刻の遅れは、次式により求められる。

$$\Delta t_{\text{diff}} = (T2 + T3) / 2 - (T1 + T4) / 2 \quad \dots (式 1)$$

### ⑤ 同期メンバ端末リストの定期参照 (マスタ端末)

マスタ端末は、定期的に同期メンバ端末リストを参照し、前回受信した同期パケットの開始タイムスタンプ T1 の時刻と現在時刻との差分が、ポーリング間隔と一定のタイムアウト時間を加えた時間以上となったスレーブ端末について、リストから削除する。この際、リストに残っている端末の PTS オフセットの最大値に応じて、共通 PTS オフセットを更新する。共通 PTS オフセットの更新は、次の同期パケットにより各スレーブ端末にも反映される。

## 4. MMT 受信および端末間同期機能の実装

### 4.1 対象端末

モバイル端末向けのプラットフォーム OS (Operating System) として Android と iOS が広く普及しているが、下記の点を考慮して、今回、Android アプリケーションとして MMT 受信機能を実装することとした。

- ・ 開発環境のオープン性が高い
- ・ SoC (System on Chip) に実装されるハードウェア復号器による 4K/60/P (解像度 3840×2160 画素、毎秒 60 フレームのプロGRESSIVE 映像) の HEVC 復号に対応する端末が存在する
- ・ アプリケーションからハードウェア復号器を制御可能な API (Application Program Interface) が存在する
- ・ スマートフォンとタブレット端末だけでなく、テレビや STB (Set Top Box) での採用例も増えている

開発にはタブレット端末 (Sony Xperia Z4 Tablet) を用い

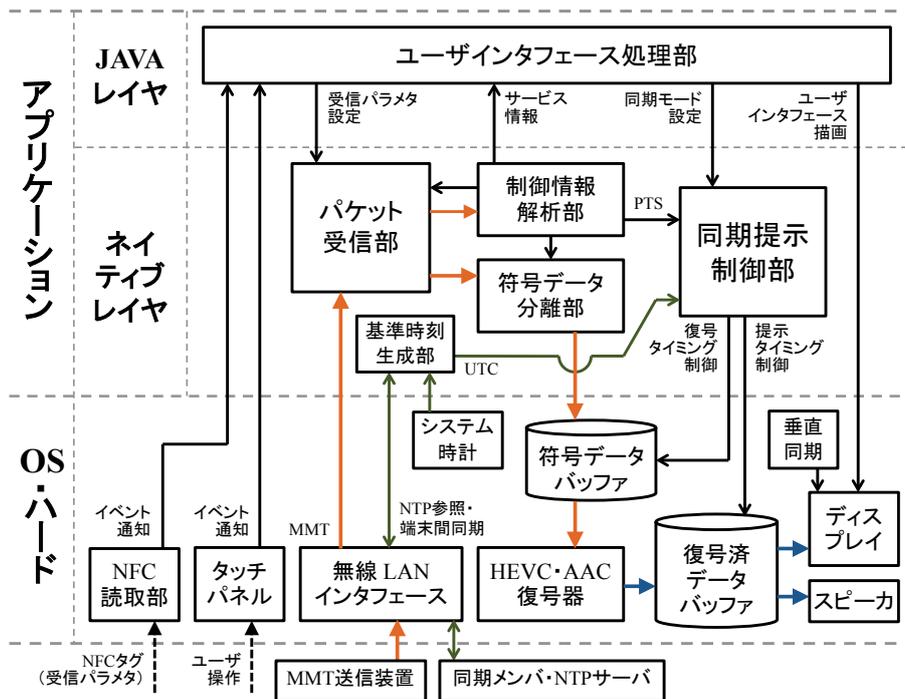


図 6 アプリケーションの構成

Figure 6 Internal block diagram of the application.

たが、同一のアプリケーションがスマートフォン（Google Nexus 6P）、スマートテレビ（Sony Bravia シリーズ）でも動作することを確認した。

#### 4.2 アプリケーションの構成

アプリケーションの内部構成を図 6 に示す。JAVA 言語で実装する範囲を JAVA レイヤ、C 言語で実装する範囲をネイティブレイヤと称することとする。JAVA レイヤとネイティブレイヤの間は、JNI（Java Native Interface）を用いて関数呼び出しや変数の共有を行う。今回、ユーザインタフェース処理部を除いて、高速な信号処理が要求されることから、ネイティブレイヤの機能として実装した。下記に、主要な機能要素の動作について説明する。

##### ユーザインタフェース処理部

Android フレームワークの機能を用いて、ディスプレイへのユーザインタフェース描画やユーザ操作のイベント処理を行う。タッチパネルでは、MMT の受信パラメタのプリセットやプリセット済チャンネルからの選局の操作が可能である。MMT の受信パラメタは、マルチキャストの場合はマルチキャストグループアドレス、送信元アドレス、宛先ポート番号、MMT パッケージ ID の組み合わせである。開発に用いる MMT 送信装置は、RTSP (Real Time Streaming Protocol) によるユニキャスト配信要求の受付機能を有するため、ユニキャストによる MMT 受信も可能とした。ユニキャストの場合、受信パラメタとして、MMT 送信装置の IP アドレス、RTSP の受付ポート番号、端末が受信する MMT の宛先ポート番号、MMT パッケージ ID の組み合わせを用いる。また、端末が NFC (Near Field Communication) 機能

をもつ場合、MMT の受信パラメタを記録した NFC タグを本体の NFC リーダ部にかざすことで、MMT 受信アプリケーションが自動起動し、NFC タグに記録された受信パラメタに従って MMT 受信を開始する機能を設けた。

マスタ端末とスレーブ端末の紐付けは、スレーブ端末にマスタ端末の IP アドレスを設定することで行う。ユーザインタフェースを用いた手動での設定以外に、NFC タグで受信パラメタと同時に設定することができ、手動設定の手間を省くことが可能である。

##### パケット受信部

ユーザインタフェース部から受信パラメタが設定されると、UDP ソケットを用いて MMTP/UDP/IP パケットの受信を開始する。所望の MMT パッケージの受信に必要なパケットのフィルタおよびバッダ部の除去を行い、制御情報パケットのペイロード部を制御情報解析部へ、符号データパケットのペイロード部を符号データ分離部に対して出力する。

##### 制御情報解析部

MMT の制御情報として伝送される MMT パッケージテーブルを解析する。映像・音声の MPU を格納するパケットのペイロード ID を特定してパケット受信部に通知、また、MPU の PTS を特定して同期提示制御部に通知する。

##### 符号データ分離部

パケットのペイロードに分割して格納された符号データを結合して、ハードウェア復号器に入力可能な形式とする。

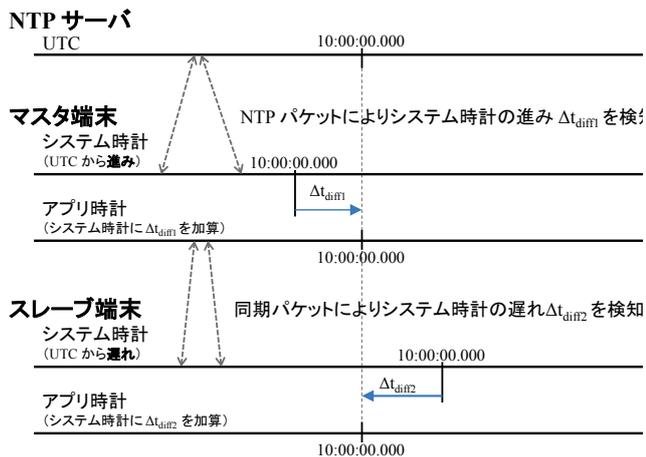


図 7 受信端末の時刻補正と時間軸

Figure 7 Time correction mechanisms and time axes.

### 基準時刻生成部

OS に管理されるシステム時計は、前回の時刻補正からの時間経過により、UTC との間に誤差が生じている可能性がある。さらに、Android では、アプリケーションの権限でシステム時計を補正することができない。そこで、アプリケーション内でシステム時計と UTC の差分値を保持し、システム時計にこの差分値を加えた値を、基準時刻生成部が出力する UTC (以降、アプリ時計と呼ぶ) とする。

基準時刻生成部は NTP サーバへの時刻参照機能を持ち、マスタ端末は任意の NTP サーバを参照してシステム時計と UTC との差分値を取得する。一方、スレーブ端末はマスタ端末との間の同期パケットによってシステム時計と UTC との差分値を取得する。図 7 に、NTP サーバがもつ UTC、マスタ端末のシステム時計・アプリ時計、スレーブ端末のシステム時計・アプリ時計について、各時刻軸の相関関係の一例を示す。同期パケットの開始タイムスタンプ T1 はシステム時計の現在時刻、受信タイムスタンプ T2 と送信タイムスタンプ T3 はアプリ時計の現在時刻とすることで、マスタ端末とスレーブ端末との間でアプリ時計の同期が実現する。

### 同期提示制御部

符号データを復号器に入力する復号タイミング制御、復号済データをディスプレイに入力する提示タイミング制御を行う。SoC に実装されるハードウェア復号器を制御するため、Android フレームワークが提供する MediaCodec クラスライブラリを用いた。MediaCodec クラスライブラリの API は JAVA レイヤで定義されるため、ネイティブレイヤの同期制御部とは JNI を介して接続する。

## 5. 端末間同期実験

### 5.1 実験方法

提案手法による端末間同期機能を検証するため、室内実験を行う。実験システムを図 8、信号形式諸元を表 2 に示す。

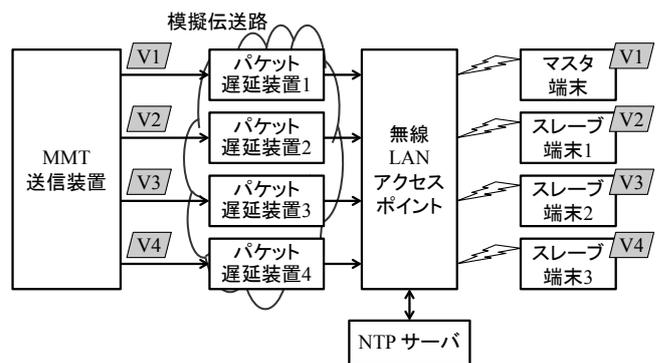


図 8 実験システム

Figure 8 Experimental system.

表 2 信号形式の諸元

Table 2 Signal format.

項目		諸元
映像	解像度	3840×2160
	フレームレート	毎秒 60 フレーム
	符号化方式	H.265   MPEG-H HEVC Main Profile Level 5.1
	ビットレート	20 Mbps
	イントラフレーム周期 (MPU のフレーム数)	32 フレーム
音声	チャンネル数	2 チャンネル
	サンプリング周波数	48 kHz
	符号化方式	MPEG-4 AAC LC
	ビットレート	160 kbps
多重・伝送	MPU のサンプル数	25600 サンプル
	メディアトランスポート方式	MPEG-H MMT ARIB STD-B60
	IP パケット MTU サイズ	1500 バイト
	IP パケット形式	MMTP/UDP/IPv6 (ユニキャスト)

MMT 送信装置は、事前に圧縮符号化された映像コンテンツを 2 次記憶に保存しており、それらをリアルタイムに MMTP パケット化して出力インターフェースから送信する。パケット遅延装置は、伝送路上での伝送遅延を模擬する装置であり、入力インターフェースから受信した MMTP パケットを設定した遅延時間に従って一定時間メモリ上に蓄えてから、順番を変えずに出力インターフェースへ送出する。最大 4 台の受信端末を用い、うち 1 台をマスタ端末、その他をスレーブ端末とする。下記の 2 項目の実験を行う。

### 実験 A. 端末間同期の表示フレームずれ測定

2 台の受信端末宛てにフレーム番号を重畳した同一内容の映像を配信し、受信中の端末画面を並べて静止画撮影し、同時に表示されているフレーム番号のずれを測定する。マスタ端末宛ての伝送遅延を 3 秒で固定し、スレーブ端末宛ての遅延を 1 秒から 5 秒まで 1 秒おきに変化させて各 20 回ずつ、合計 100 回、アプリケーション再起動を含めた受信動作と静止画撮影を行う。

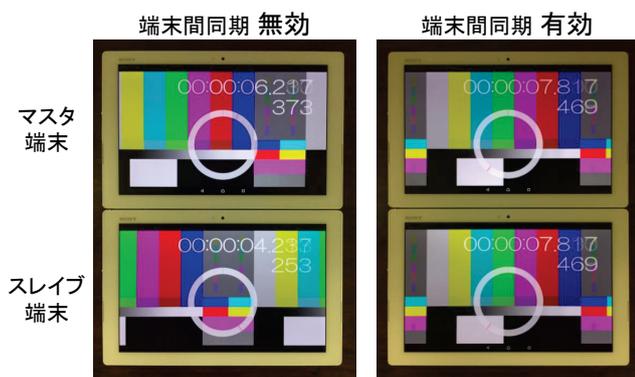


図 9 端末間同期のフレームずれ測定 (実験 A)

Figure 9 Measurement of frame deviation (Experiment A).

### 実験 B. 端末間同期による 8K 映像再構成

4 台の受信端末宛てに、8K 映像を 4 分割した各領域の 4K 映像を配信し、端末間同期により 8K 映像を再構成できることを確認する。マスタ端末宛ての伝送遅延を 1 秒、スレーブ端末 1 宛ての遅延を 2 秒、スレーブ端末 2 宛ての遅延を 3 秒、スレーブ端末 3 宛ての遅延を 4 秒とする。

### 5.2 実験結果

実験 A において、スレーブ端末の遅延を 5 秒としたときの受信端末を図 9 に示す。図の左側はスレーブ端末の同期機能が無効の状態、図の右側はスレーブ端末の同期機能が有効の状態である。同期機能が無効のとき、マスタ端末とスレーブ端末に同時に表示されているフレーム番号の差は、端末間の伝送遅延差である 2 秒間のフレーム数である 120 となっている。一方、同期機能が有効のときは、同時に表示されているフレーム番号が一致し、差は 0 であることが分かる。合計 100 回の試行により測定した表示フレームずれと、その測定回数を表 3 に示す。表示フレームずれは、マスタ端末のフレーム番号に対してスレーブ端末のフレーム番号が遅れている場合に負数、進んでいる場合に正数としている。表から、マスタ端末とスレーブ端末の伝送遅延差に関わらず、端末間のフレームずれの大きさは最大 3 フレームに収まることを確認した。また、100 回の測定によるフレームずれの平均は、-0.18 フレームであった。

実験 B における受信端末を図 10 に示す。図の上部はスレーブ端末の同期機能が無効の状態、図の下部はスレーブ端末の同期機能が有効の状態である。同期機能が有効のときには、4 台の受信端末をタイル状に合わせて 8K 映像が再構成できていることが分かる。実験 A により、端末間で最大 3 フレーム以内の表示フレームずれが生じ得ることが分かっているが、人間の見た目では違和感なく 1 つの 8K 映像として視聴できることを確認した。カット編集のタイミングが同じ映像を並べて見る場合にはフレームずれに気が付くことがあるが、別カメラの映像や編集ポイントが異なる映像の場合には、表示フレームずれに対する許容度はより高くなることが想定される。

表 3 端末間の表示フレームずれの測定結果

Table 3 Measurement result of frame deviation.

	スレーブ端末宛ての伝送遅延					計
	1 秒	2 秒	3 秒	4 秒	5 秒	
-3 フレーム	0	1	2	0	0	3
-2 フレーム	4	5	4	1	2	16
-1 フレーム	5	4	2	5	4	20
0 フレーム	5	5	5	7	6	28
1 フレーム	3	2	6	5	7	23
2 フレーム	3	3	1	2	1	10
3 フレーム	0	0	0	0	0	0

### 5.3 表示フレームずれ要因の考察

実験 A の結果から、フレームずれの平均では-0.18 フレームの精度である一方、最大 3 フレームの範囲で同期精度にゆらぎが生じていることを確認した。これらの要因として、下記が考えられる。

#### ① ディスプレイ垂直同期のタイミングずれ

アプリケーションから、ディスプレイを駆動する垂直同期のタイミングを制御することはできない。仮に、アプリケーションによる提示制御の実行タイミングが完全に同期していたとしても、最終的にディスプレイ上に表示されるタイミングには、±1 フレームの範囲でのずれが生じる。実験 A の測定結果では、71 % がこの±1 フレームの誤差の範囲に収まる。

#### ② 提示制御の実行タイミングのゆらぎ

OS や他のアプリケーションとハードウェア資源を共有しており、割り込み等で提示制御の実行タイミングにゆらぎが発生し得る。また、マルチスレッドを用いた実装のた

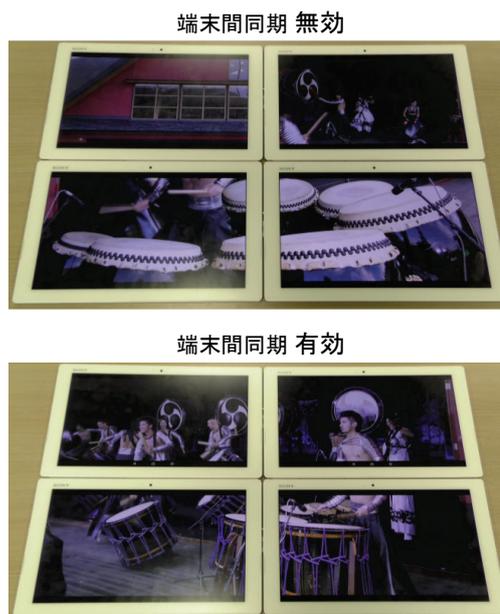


図 10 端末間同期による 8K 再構成 (実験 B)

Figure 10 Reconstruction of 8K video (Experiment B).

め、アプリケーション内のスレッド間のコンテキストスイッチも提示制御の実行タイミングにゆらぎを与える要因となり得る。

### ③ 同期パケットの伝送遅延ゆらぎによるアプリ時計の時刻同期誤差

スレーブ端末のシステム時計とマスタ端末のアプリ時計との間の時刻ずれを求める式1は、スレーブ端末とマスタ端末の間での同期パケットの送受信において、往路と復路の伝送遅延が同一であることを前提としている。したがって、同期パケットの伝送遅延にゆらぎがあり、往路と復路の伝送遅延に差が生じている場合、求めた時刻ずれには測定誤差が生じる。端末間でのアプリ時計の同期誤差は、そのまま、端末間の映像・音声の同期誤差となる。

①～③の他、受信端末として異なる機種を用いて同期を行う場合には、下記に留意する必要がある。

### ④ アプリケーションの後段の画像信号処理

端末機種によっては、アプリケーションから提示制御を実行した後段の処理として、高画質化・フレーム補完などの信号処理が行われ、実際にディスプレイに表示されるまでに処理遅延が発生する場合がある。これらの処理遅延が固定であり、大きさが既知の場合には、自身のPTSオフセットにこの処理遅延を加えることも考えられる。

## 6. MMT 端末間同期の応用サービス検討

MMT を用いた映像・音声の高精度同期を活用したサービスの具体例として、筆者らは、放送の主映像・主音声に同期する付加映像や付加音声をインターネットで配信するサービスについて機能開発・検証を進めている。例えば、サッカーなどのスポーツ中継において、特定の選手を中心にとらえた複数の付加映像をインターネットで配信すれば、視聴者が好みに応じた付加映像を手元のモバイル端末で受信して8Kテレビと同期させて視聴できる。また、国際的なスポーツイベントにおいて、多言語の実況音声をインターネットで配信することで、視聴者が言語を選択して実況音声を受信できるサービスも考えられる。これらの形態は、家庭内での放送受信に限らず、パブリックビューイングや映画上映での応用にも期待ができる。

その他の応用分野として、例えば、デジタルサイネージが挙げられる。デジタルサイネージ分野では、小・中型ディスプレイを複数組み合わせる大型化するタイル状ディスプレイへの需要が高い。タイル状ディスプレイの駆動方式としては、描画装置1台から多系統の映像信号を出力して個別のディスプレイに有線接続する方式がある。一方、本稿で実装したアプリケーションをインストールしたタブレット端末を複数台並べることで、簡易的なタイル状ディスプレイを構成できる。この場合、無線LANを用いるこ

とによる配線コストの削減、レイアウトの自由度の向上などのメリットが考えられる。また、街頭のデジタルサイネージの映像に同期する付加映像を、個人のモバイル端末向けに配信することも考えられる。この場合、デジタルサイネージをマスタ端末、個人のモバイル端末をスレーブ端末として高精度同期をすることが可能である。本稿でも実装したNFCタグによるMMT受信パラメタ設定機能を利用すれば、デジタルサイネージに設置したNFCタグに個人のモバイル端末をかざすことでMMT受信アプリケーションが自動的に起動して付加映像の受信を開始させる仕組みも実現できる。付加映像を受信することで特典情報を得られる様なコンテンツ要素と組み合わせることで、デジタルサイネージの広告効果の向上も期待できるだろう。

## 7. おわりに

本稿では、モバイル端末向けプラットフォームとして広く普及しているAndroidを対象に、端末間での映像・音声の高精度同期機能を備えたMMT受信アプリケーションを実装した。毎秒60フレームの映像を用いた端末間同期実験の結果、各受信端末宛での伝送遅延が異なっても、端末間での表示フレームのずれが最大3フレーム、平均0.18フレームという精度で同期が可能であることを確認した。これは、人間の見た目には違和感のない同期視聴が可能な精度である。

今後、4K・8K衛星放送における放送・通信連携サービスの実用化に向け、放送とインターネット回線を用いた実証実験を行う予定である。また、放送・通信連携サービスの基本機能についてさらなる実装や検証を進めるとともに、新たなサービスの創出につなげるため、コンテンツ制作者への技術周知を図りたい。

## 参考文献

- 1) ISO/IEC 23008-1, "Information technology – High efficiency coding and media delivery in heterogeneous environments – Part 1: MPEG media transport (MMT)," (2014).
- 2) ARIB STD-B60 1.0 版, "デジタル放送における MMT によるメディアトランスポート方式," (2014).
- 3) ARIB TR-B39 1.0 版, "高度広帯域衛星デジタル放送 運用規定," (2016).
- 4) Y. Kawamura, K. Otsuki, A. Hashimoto and Y. Endo, "Functional Evaluation of Hybrid Content Delivery Using MPEG Media Transport," IEEE International Conference on Consumer Electronics, pp.267-268 (2016).
- 5) SK Telecom, "SK Telecom Commercializes True Real Time Mobile Streaming Technology Named 'T Live Streaming,'" (2016) <http://www.sktelecom.com/en/press/detail.do?idx=1169>
- 6) ISO/IEC 13818-1, "Information technology – Generic coding of moving pictures and associated audio information: Systems" (1995).
- 7) ARIB STD-B44 2.0 版, "高度広帯域衛星デジタル放送の伝送方式," (2014).