                            RTP Payload Format
        for the Mixed Excitation Linear Prediction Enhanced (MELPe) Codec

Abstract

   This document describes the RTP payload format for the Mixed
   Excitation Linear Prediction Enhanced (MELPe) speech coder. MELPe's
   three different speech encoding rates and sample frame sizes are
   supported. Comfort noise procedures and packet loss concealment are
   described in detail.

Status of This Memo

   This is an Internet Standards Track document.

   This document is a product of the Internet Engineering Task Force
   (IETF). It represents the consensus of the IETF community. It has
   received public review and has been approved for publication by the
   Internet Engineering Steering Group (IESG). Further information on
   Internet Standards is available in Section 2 of RFC 7841.

   Information about the current status of this document, any errata,
   and how to provide feedback on it may be obtained at
   http://www.rfc-editor.org/info/rfc8130.

Table of Contents

1. Introduction

   This document describes how compressed Mixed Excitation Linear
   Prediction Enhanced (MELPe) speech as produced by the MELPe codec
   may be formatted for use as an RTP payload. Details are provided to
   packetize the three different codec bitrate data frames (2400, 1200,
   and 600) into RTP packets. The sender may send one or more codec
   data frames per packet, depending on the application scenario or
   based on transport network conditions, bandwidth restrictions, delay
   requirements, and packet loss tolerance.

1.1. Conventions

   The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
   "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this
   document are to be interpreted as described in RFC 2119 [RFC2119].

   Best current practices for writing an RTP payload format
   specification were followed [RFC2736].

2. Background

   The MELP speech coder was developed by the US military as an upgrade
   from the LPC-based CELP standard vocoder for low-bitrate
   communications [MELP]. ("LPC" stands for "Linear-Predictive Coding",
   and "CELP" stands for "Code-Excited Linear Prediction".) MELP was
   further enhanced and subsequently adopted by NATO as MELPe for use by
   its members and Partnership for Peace countries for military and
   other governmental communications [MELPE]. The MELP speech coder
   algorithm was developed by Atlanta Signal Processing (ASPI), Texas
   Instruments (TI), SignalCom (now Microsoft), and Thales
   Communications, with noise preprocessor contributions from AT&T,
   under contract with NSA/DOD as international NATO Standard
   STANAG 4591 [MELPE].

   Commercial/civilian applications have arisen because of the
   low-bitrate property of MELPe with its (relatively) high
   intelligibility. As such, MELPe is being used in a variety of wired
   and radio communications systems. Voice over IP (VoIP) / SIP systems
   need to transport MELPe without decoding and re-encoding in order to
   preserve its intelligibility. Hence, it is desirable and necessary
   to define the proper payload formatting and use conventions of MELPe
   in RTP payloads.

   The MELPe codec [MELPE] supports three different vocoder bitrates:
   2400, 1200, and 600 bps. The basic 2400 bps bitrate vocoder uses a
   22.5 ms frame of speech consisting of 180 8000-Hz, 16-bit speech
   samples. The 1200 and 600 bps bitrate vocoders each use three and
   four 22.5 ms frames of speech, respectively. These reduced-bitrate
   vocoders internally use multiple 2400 bps parameter sets with further
   processing to strategically remove redundancy. The payload sizes for
   each of the bitrates are 54, 81, and 54 bits for the 2400, 1200, and
   600 bps frames, respectively. Dynamic bitrate switching is permitted
   but only if supported by both endpoints.

   The MELPe algorithm distinguishes between voiced and unvoiced speech
   and encodes each differently. Unvoiced speech can be coded with
   fewer information bits for the same quality. Forward error
   correction (FEC) is applied to the 2400 bps codec unvoiced speech for
   better protection of the subtle differences in signal reconstruction.
   The lower-bitrate coders do not allocate any bits for FEC and rely on
   strong error protection and correction in the communications channel.

Comfort noise handling for MELPe follows the procedures in Appendix B
of SCIP-210 [SCIP210]. After Voice Activity Detection (VAD)
no longer indicates the presence of speech/voice, a minimum of two
comfort noise vocoder frames (serving as a grace period) are to be
transmitted. The contents of the comfort noise frames are described
in the next section.

Packet loss concealment (PLC) exploits the FEC (and, more precisely,
any combination of two set bits in the pitch/voicing parameter) of
the 2400 bps speech coder. The pitch/voicing parameter has a sparse
set of permitted values. A value of zero indicates a non-voiced
frame. At least three bits are set for all valid pitch parameters.
The PLC erasure indication utilizes any errored/erasure encodings of
the pitch/voicing parameter with exactly two set bits, as described
below.

3. Payload Format

   The MELPe codec uses 22.5, 67.5, or 90 ms frames with a sampling rate
   clock of 8 kHz, so the RTP timestamp MUST be in units of 1/8000 of a
   second.

   The RTP payload for MELPe has the format shown in Figure 1. No
   additional header specific to this payload format is needed. This
   format is intended for situations where the sender and the receiver
   send one or more codec data frames per packet.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          RTP Header                           |
+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
|                                                               |
+                  one or more frames of MELPe                  |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
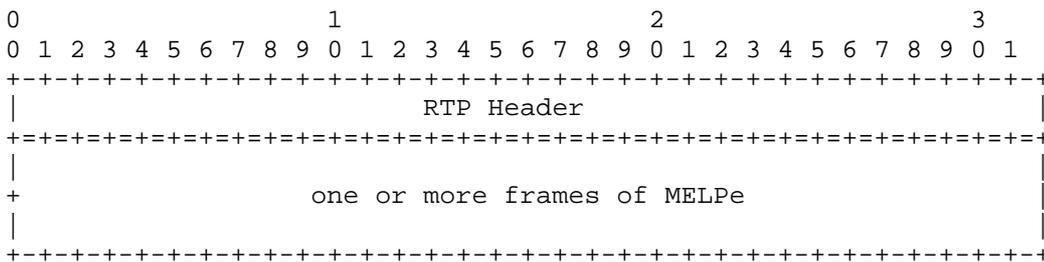
                   Figure 1: Packet Format Diagram

   The RTP header of the packetized encoded MELPe speech has the
   expected values as described in [RFC3550]. The usage of the M bit
   SHOULD be as specified in the applicable RTP profile -- for example,
   [RFC3551], where [RFC3551] specifies that if the sender does not
   suppress silence (i.e., sends a frame on every frame interval), the
   M bit will always be zero. When more than one codec data frame is
   present in a single RTP packet, the timestamp is, as always, that of
   the oldest data frame represented in the RTP packet.

The assignment of an RTP payload type for this new packet format is
outside the scope of this document and will not be specified here.
It is expected that the RTP profile for a particular class of
applications will assign a payload type for this encoding, or if that
is not done, then a payload type in the dynamic range shall be chosen
by the sender.

3.1. MELPe Bitstream Definitions

The total number of bits used to describe one frame of 2400 bps
speech is 54, which fits in 7 octets (with two unused bits). For
1200 bps speech, the total number of bits used is 81, which fits in
11 octets (with seven unused bits). For 600 bps speech, the total
number of bits used is 54, which fits in 7 octets (with two unused
bits). Unused bits, shown below as RSVA, RSVB, etc., are coded as
described in Section 3.3 in support of dynamic bitrate switching.

In the MELPe bitstream definitions, the most significant bits are
considered priority bits. The intention was that these bits receive
greater protection in the underlying communications channel. For IP
networks, such additional protection is irrelevant. However, for the
convenience of interoperable gateway devices, the bitstreams will be
presented identically in IP networks.

3.1.1. 2400 bps Bitstream Structure

   According to Table 3 of [MELPE], the 2400 bps MELPe bit transmission
   order (for clarity, the bit priority is not shown) is as follows:

```
            +--------+-------------+-------------+
            |  Bit   |   Voiced    |  Unvoiced   |
            +--------+-------------+-------------+
            |  B_01  |     g20     |     g20     |
            |  B_02  |     BP0     |    FEC10    |
            |  B_03  |     P0      |     P0      |
            |  B_04  |    LSF20    |    LSF20    |
            |  B_05  |    LSF30    |    LSF30    |
            |  B_06  |     g23     |     g23     |
            |  B_07  |     g24     |     g24     |
            |  B_08  |    LSF35    |    LSF35    |
            +--------+-------------+-------------+
            |  B_09  |     g21     |     g21     |
            |  B_10  |     g22     |     g22     |
            |  B_11  |     P4      |     P4      |
            |  B_12  |    LSF34    |    LSF34    |
            |  B_13  |     P5      |     P5      |
            |  B_14  |     P1      |     P1      |
            |  B_15  |     P2      |     P2      |
            |  B_16  |    LSF40    |    LSF40    |
            +--------+-------------+-------------+
            |  B_17  |     P6      |     P6      |
            |  B_18  |    LSF10    |    LSF10    |
            |  B_19  |    LSF16    |    LSF16    |
            |  B_20  |    LSF45    |    LSF45    |
            |  B_21  |     P3      |     P3      |
            |  B_22  |    LSF15    |    LSF15    |
            |  B_23  |    LSF14    |    LSF14    |
            |  B_24  |    LSF25    |    LSF25    |
            +--------+-------------+-------------+
            |  B_25  |     BP3     |    FEC13    |
            |  B_26  |    LSF13    |    LSF13    |
            |  B_27  |    LSF12    |    LSF12    |
            |  B_28  |    LSF24    |    LSF24    |
            |  B_29  |    LSF44    |    LSF44    |
            |  B_30  |     FM0     |    FEC40    |
            |  B_31  |    LSF11    |    LSF11    |
            |  B_32  |    LSF23    |    LSF23    |
```

```
                +--------+------------+------------+
                |  B_33  |    FM7     |    FEC22   |
                |  B_34  |    FM6     |    FEC21   |
                |  B_35  |    FM5     |    FEC20   |
                |  B_36  |    g11     |     g11    |
                |  B_37  |    g10     |     g10    |
                |  B_38  |    BP2     |    FEC12   |
                |  B_39  |    BP1     |    FEC11   |
                |  B_40  |   LSF21    |    LSF21   |
                +--------+------------+------------+
                |  B_41  |   LSF33    |    LSF33   |
                |  B_42  |   LSF22    |    LSF22   |
                |  B_43  |   LSF32    |    LSF32   |
                |  B_44  |   LSF31    |    LSF31   |
                |  B_45  |   LSF43    |    LSF43   |
                |  B_46  |   LSF42    |    LSF42   |
                |  B_47  |    AF      |    FEC42   |
                |  B_48  |   LSF41    |    LSF41   |
                +--------+------------+------------+
                |  B_49  |    FM4     |    FEC32   |
                |  B_50  |    FM3     |    FEC31   |
                |  B_51  |    FM2     |    FEC30   |
                |  B_52  |    FM1     |    FEC41   |
                |  B_53  |    g12     |     g12    |
                |  B_54  |   SYNC     |    SYNC    |
                +--------+------------+------------+
```

Notes:
g = Gain
BP = Bandpass Voicing
P = Pitch/Voicing
LSF = Line Spectral Frequencies
FEC = Forward Error Correction Parity Bits
FM = Fourier Magnitudes
AF = Aperiodic Flag
B_01 = least significant bit of data set

Table 1: Bitstream Definition for MELPe 2400 bps

The 2400 bps MELPe RTP payload is constructed as per Figure 2. Note
that bit B_01 is placed in the least significant bit (LSB) of the
first byte with all other bits in sequence. When filling octets, the
least significant bits of the seventh octet are filled with bits B_49
to B_54, respectively.

```
     MSB                                                 LSB
      0      1      2      3      4      5      6      7
    +------+------+------+------+------+------+------+------+
    | B_08 | B_07 | B_06 | B_05 | B_04 | B_03 | B_02 | B_01 |
    +------+------+------+------+------+------+------+------+
    | B_16 | B_15 | B_14 | B_13 | B_12 | B_11 | B_10 | B_09 |
    +------+------+------+------+------+------+------+------+
    | B_24 | B_23 | B_22 | B_21 | B_20 | B_19 | B_18 | B_17 |
    +------+------+------+------+------+------+------+------+
    | B_32 | B_31 | B_30 | B_29 | B_28 | B_27 | B_26 | B_25 |
    +------+------+------+------+------+------+------+------+
    | B_40 | B_39 | B_38 | B_37 | B_36 | B_35 | B_34 | B_33 |
    +------+------+------+------+------+------+------+------+
    | B_48 | B_47 | B_46 | B_45 | B_44 | B_43 | B_42 | B_41 |
    +------+------+------+------+------+------+------+------+
    | RSVA | RSVB | B_54 | B_53 | B_52 | B_51 | B_50 | B_49 |
    +------+------+------+------+------+------+------+------+
```
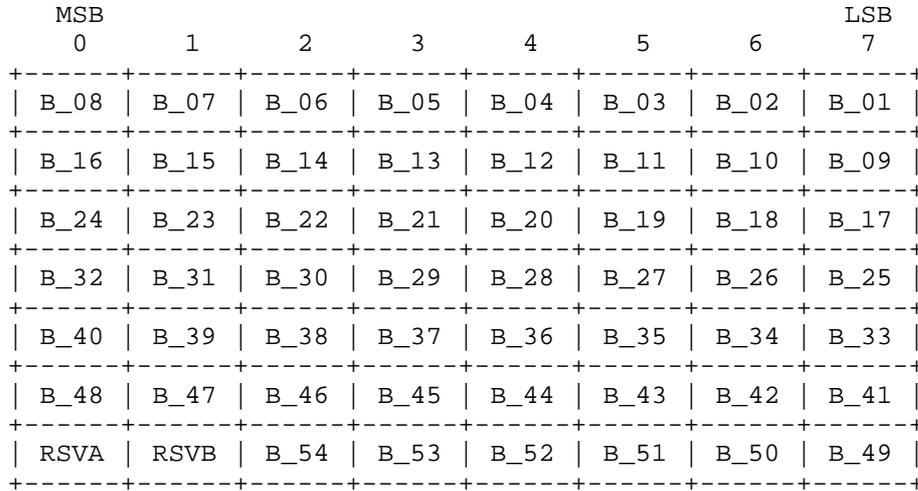
Figure 2: Packed MELPe 2400 bps Payload Octets

3.1.2. 1200 bps Bitstream Structure

   According to Tables D-9a and D-9b of [MELPE], the 1200 bps MELPe bit
   transmission order is as follows:

Bit	Modes 1-4 (Voiced)	Mode 5 (Unvoiced)
B_01	Syn	Syn
B_02	Pitch&UV0	Pitch&UV0
B_03	Pitch&UV1	Pitch&UV1
B_04	Pitch&UV2	Pitch&UV2
B_05	Pitch&UV3	Pitch&UV3
B_06	Pitch&UV4	Pitch&UV4
B_07	Pitch&UV5	Pitch&UV5
B_08	Pitch&UV6	Pitch&UV6
B_09	Pitch&UV7	Pitch&UV7
B_10	Pitch&UV8	Pitch&UV8
B_11	Pitch&UV9	Pitch&UV9
B_12	Pitch&UV10	Pitch&UV10
B_13	Pitch&UV11	Pitch&UV11
B_14	LSP0	LSP0
B_15	LSP1	LSP1
B_16	LSP2	LSP2
B_17	LSP3	LSP3
B_18	LSP4	LSP4
B_19	LSP5	LSP5
B_20	LSP6	LSP6
B_21	LSP7	LSP7
B_22	LSP8	LSP8
B_23	LSP9	LSP9
B_24	LSP10	LSP10
B_25	LSP11	LSP11
B_26	LSP12	LSP12
B_27	LSP13	LSP13
B_28	LSP14	LSP14
B_29	LSP15	LSP15
B_30	LSP16	LSP16
B_31	LSP17	LSP17
B_32	LSP18	LSP18

```
+--------+------------+------------+
|  B_33  |    LSP19   |    LSP19   |
|  B_34  |    LSP20   |    LSP20   |
|  B_35  |    LSP21   |    LSP21   |
|  B_36  |    LSP22   |    LSP22   |
|  B_37  |    LSP23   |    LSP23   |
|  B_38  |    LSP24   |    LSP24   |
|  B_39  |    LSP25   |    LSP25   |
|  B_40  |    LSP26   |    LSP26   |
+--------+------------+------------+
|  B_41  |    LSP27   |    GAIN0   |
|  B_42  |    LSP28   |    GAIN1   |
|  B_43  |    LSP29   |    GAIN2   |
|  B_44  |    LSP30   |    GAIN3   |
|  B_45  |    LSP31   |    GAIN4   |
|  B_46  |    LSP32   |    GAIN5   |
|  B_47  |    LSP33   |    GAIN6   |
|  B_48  |    LSP34   |    GAIN7   |
+--------+------------+------------+
|  B_49  |    LSP35   |    GAIN8   |
|  B_50  |    LSP36   |    GAIN9   |
|  B_51  |    LSP37   |            |
|  B_52  |    LSP38   |            |
|  B_53  |    LSP39   |            |
|  B_54  |    LSP40   |            |
|  B_55  |    LSP41   |            |
|  B_56  |    LSP42   |            |
+--------+------------+------------+
|  B_57  |    GAIN0   |            |
|  B_58  |    GAIN1   |            |
|  B_59  |    GAIN2   |            |
|  B_60  |    GAIN3   |            |
|  B_61  |    GAIN4   |            |
|  B_62  |    GAIN5   |            |
|  B_63  |    GAIN6   |            |
|  B_64  |    GAIN7   |            |
+--------+------------+------------+
|  B_65  |    GAIN8   |            |
|  B_66  |    GAIN9   |            |
|  B_67  |     BP0    |            |
|  B_68  |     BP1    |            |
|  B_69  |     BP2    |            |
|  B_70  |     BP3    |            |
|  B_71  |     BP4    |            |
|  B_72  |     BP5    |            |
```

```
           +--------+------------+------------+
           | B_73   |   JITTER   |            |
           | B_74   |    FS0     |            |
           | B_75   |    FS1     |            |
           | B_76   |    FS2     |            |
           | B_77   |    FS3     |            |
           | B_78   |    FS4     |            |
           | B_79   |    FS5     |            |
           | B_80   |    FS6     |            |
           +--------+------------+------------+
           | B_81   |    FS7     |            |
           +--------+------------+------------+

           Notes:
           BP = Bandpass voicing
           FS = Fourier magnitudes
           LSP = Line Spectral Pair
           Pitch&UV = Pitch/voicing
           GAIN = Gain
           JITTER = Jitter
```

             Table 2: Bitstream Definition for MELPe 1200 bps

   The 1200 bps MELPe RTP payload is constructed as per Figure 3. Note
   that bit B_01 is placed in the LSB of the first byte with all other
   bits in sequence. When filling octets, the least significant bit of
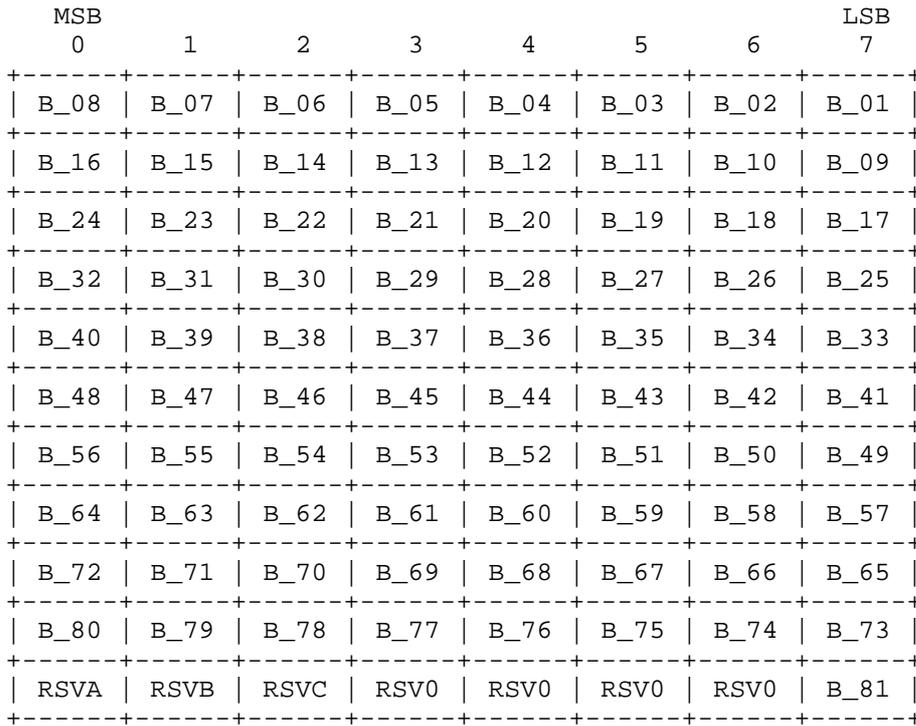   the eleventh octet is filled with bit B_81.

```
      MSB                                                        LSB
       0       1       2       3       4       5       6       7
     +------+------+------+------+------+------+------+------+
     | B_08 | B_07 | B_06 | B_05 | B_04 | B_03 | B_02 | B_01 |
     +------+------+------+------+------+------+------+------+
     | B_16 | B_15 | B_14 | B_13 | B_12 | B_11 | B_10 | B_09 |
     +------+------+------+------+------+------+------+------+
     | B_24 | B_23 | B_22 | B_21 | B_20 | B_19 | B_18 | B_17 |
     +------+------+------+------+------+------+------+------+
     | B_32 | B_31 | B_30 | B_29 | B_28 | B_27 | B_26 | B_25 |
     +------+------+------+------+------+------+------+------+
     | B_40 | B_39 | B_38 | B_37 | B_36 | B_35 | B_34 | B_33 |
     +------+------+------+------+------+------+------+------+
     | B_48 | B_47 | B_46 | B_45 | B_44 | B_43 | B_42 | B_41 |
     +------+------+------+------+------+------+------+------+
     | B_56 | B_55 | B_54 | B_53 | B_52 | B_51 | B_50 | B_49 |
     +------+------+------+------+------+------+------+------+
     | B_64 | B_63 | B_62 | B_61 | B_60 | B_59 | B_58 | B_57 |
     +------+------+------+------+------+------+------+------+
     | B_72 | B_71 | B_70 | B_69 | B_68 | B_67 | B_66 | B_65 |
     +------+------+------+------+------+------+------+------+
     | B_80 | B_79 | B_78 | B_77 | B_76 | B_75 | B_74 | B_73 |
     +------+------+------+------+------+------+------+------+
     | RSVA | RSVB | RSVC | RSV0 | RSV0 | RSV0 | RSV0 | B_81 |
     +------+------+------+------+------+------+------+------+
```

        Figure 3: Packed MELPe 1200 bps Payload Octets

3.1.3. 600 bps Bitstream Structure

   According to Tables M-11 to M-16 of [MELPE], the 600 bps MELPe bit
   transmission order (for clarity, the bit priority is not shown) is as
   follows:

```
        +--------+-------------+-------------+-------------+
        |  Bit   |    Mode 1   |    Mode 2   |    Mode 3   |
        |        |   (Voiced)  |   (voiced)  |   (voiced)  |
        +--------+-------------+-------------+-------------+
        |  B_01  | Voicing (4) | Voicing (4) | Voicing (4) |
        |  B_02  | Voicing (3) | Voicing (3) | Voicing (3) |
        |  B_03  | Voicing (2) | Voicing (2) | Voicing (2) |
        |  B_04  | Voicing (1) | Voicing (1) | Voicing (1) |
        |  B_05  | Voicing (0) | Voicing (0) | Voicing (0) |
        |  B_06  |  LSF1,4 (3) |  Pitch (5)  |  Pitch (7)  |
        |  B_07  |  LSF1,4 (2) |  Pitch (4)  |  Pitch (6)  |
        |  B_08  |  LSF1,4 (1) |  Pitch (3)  |  Pitch (5)  |
        +--------+-------------+-------------+-------------+
        |  B_09  |  LSF1,4 (0) |  Pitch (2)  |  Pitch (4)  |
        |  B_10  |  LSF1,3 (3) |  Pitch (1)  |  Pitch (3)  |
        |  B_11  |  LSF1,3 (2) |  Pitch (0)  |  Pitch (2)  |
        |  B_12  |  LSF1,3 (1) |  LSF1,3 (3) |  Pitch (1)  |
        |  B_13  |  LSF1,3 (0) |  LSF1,3 (2) |  Pitch (0)  |
        |  B_14  |  LSF1,2 (3) |  LSF1,3 (1) |  LSF1,3 (3) |
        |  B_15  |  LSF1,2 (2) |  LSF1,3 (0) |  LSF1,3 (2) |
        |  B_16  |  LSF1,2 (1) |  LSF1,2 (3) |  LSF1,3 (1) |
        +--------+-------------+-------------+-------------+
        |  B_17  |  LSF1,2 (0) |  LSF1,2 (2) |  LSF1,3 (0) |
        |  B_18  |  LSF1,1 (5) |  LSF1,2 (1) |  LSF1,2 (4) |
        |  B_19  |  LSF1,1 (4) |  LSF1,2 (0) |  LSF1,2 (3) |
        |  B_20  |  LSF1,1 (3) |  LSF1,1 (5) |  LSF1,2 (2) |
        |  B_21  |  LSF1,1 (2) |  LSF1,1 (4) |  LSF1,2 (1) |
        |  B_22  |  LSF1,1 (1) |  LSF1,1 (3) |  LSF1,2 (0) |
        |  B_23  |  LSF1,1 (0) |  LSF1,1 (2) |  LSF1,1 (5) |
        |  B_24  |  LSF2,4 (3) |  LSF1,1 (1) |  LSF1,1 (4) |
        +--------+-------------+-------------+-------------+
        |  B_25  |  LSF2,4 (2) |  LSF1,1 (0) |  LSF1,1 (3) |
        |  B_26  |  LSF2,4 (1) |  LSF2,3 (3) |  LSF1,1 (2) |
        |  B_27  |  LSF2,4 (0) |  LSF2,3 (2) |  LSF1,1 (1) |
        |  B_28  |  LSF2,3 (3) |  LSF2,3 (1) |  LSF1,1 (0) |
        |  B_29  |  LSF2,3 (2) |  LSF2,3 (0) |  LSF2,3 (3) |
        |  B_30  |  LSF2,3 (1) |  LSF2,2 (4) |  LSF2,3 (2) |
        |  B_31  |  LSF2,3 (0) |  LSF2,2 (3) |  LSF2,3 (1) |
        |  B_32  |  LSF2,2 (3) |  LSF2,2 (2) |  LSF2,3 (0) |
```

```
+--------+------------+------------+------------+
|  B_33  | LSF2,2 (2) | LSF2,2 (1) | LSF2,2 (4) |
|  B_34  | LSF2,2 (1) | LSF2,2 (0) | LSF2,2 (3) |
|  B_35  | LSF2,2 (0) | LSF2,1 (6) | LSF2,2 (2) |
|  B_36  | LSF2,1 (5) | LSF2,1 (5) | LSF2,2 (1) |
|  B_37  | LSF2,1 (4) | LSF2,1 (4) | LSF2,2 (0) |
|  B_38  | LSF2,1 (3) | LSF2,1 (3) | LSF2,1 (5) |
|  B_39  | LSF2,1 (2) | LSF2,1 (2) | LSF2,1 (4) |
|  B_40  | LSF2,1 (1) | LSF2,1 (1) | LSF2,1 (3) |
+--------+------------+------------+------------+
|  B_41  | LSF2,1 (0) | LSF2,1 (0) | LSF2,1 (2) |
|  B_42  | GAIN2 (5)  | GAIN2 (5)  | LSF2,1 (1) |
|  B_43  | GAIN2 (4)  | GAIN2 (4)  | LSF2,1 (0) |
|  B_44  | GAIN2 (3)  | GAIN2 (3)  | GAIN2 (4)  |
|  B_45  | GAIN2 (2)  | GAIN2 (2)  | GAIN2 (3)  |
|  B_46  | GAIN2 (1)  | GAIN2 (1)  | GAIN2 (2)  |
|  B_47  | GAIN2 (0)  | GAIN2 (0)  | GAIN2 (1)  |
|  B_48  | GAIN1 (6)  | GAIN1 (6)  | GAIN2 (0)  |
+--------+------------+------------+------------+
|  B_49  | GAIN1 (5)  | GAIN1 (5)  | GAIN1 (5)  |
|  B_50  | GAIN1 (4)  | GAIN1 (4)  | GAIN1 (4)  |
|  B_51  | GAIN1 (3)  | GAIN1 (3)  | GAIN1 (3)  |
|  B_52  | GAIN1 (2)  | GAIN1 (2)  | GAIN1 (2)  |
|  B_53  | GAIN1 (1)  | GAIN1 (1)  | GAIN1 (1)  |
|  B_54  | GAIN1 (0)  | GAIN1 (0)  | GAIN1 (0)  |
+--------+------------+------------+------------+
```

Table 3: Bitstream Definition for MELPe 600 bps (Part 1 of 2)

Bit	Mode 4 (voiced)	Mode 5 (voiced)	Mode 6 (voiced)
B_01	Voicing (4)	Voicing (4)	Voicing (4)
B_02	Voicing (3)	Voicing (3)	Voicing (3)
B_03	Voicing (2)	Voicing (2)	Voicing (2)
B_04	Voicing (1)	Voicing (1)	Voicing (1)
B_05	Voicing (0)	Voicing (0)	Voicing (0)
B_06	Pitch (7)	Pitch (7)	Pitch (7)
B_07	Pitch (6)	Pitch (6)	Pitch (6)
B_08	Pitch (5)	Pitch (5)	Pitch (5)
B_09	Pitch (4)	Pitch (4)	Pitch (4)
B_10	Pitch (3)	Pitch (3)	Pitch (3)
B_11	Pitch (2)	Pitch (2)	Pitch (2)
B_12	Pitch (1)	Pitch (1)	Pitch (1)
B_13	Pitch (0)	Pitch (0)	Pitch (0)
B_14	LSF1,3 (3)	LSF1,3 (3)	LSF1,3 (3)
B_15	LSF1,3 (2)	LSF1,3 (2)	LSF1,3 (2)
B_16	LSF1,3 (1)	LSF1,3 (1)	LSF1,3 (1)
B_17	LSF1,3 (0)	LSF1,3 (0)	LSF1,3 (0)
B_18	LSF1,2 (3)	LSF1,2 (4)	LSF1,2 (4)
B_19	LSF1,2 (2)	LSF1,2 (3)	LSF1,2 (3)
B_20	LSF1,2 (1)	LSF1,2 (2)	LSF1,2 (2)
B_21	LSF1,2 (0)	LSF1,2 (1)	LSF1,2 (1)
B_22	LSF1,1 (5)	LSF1,2 (0)	LSF1,2 (0)
B_23	LSF1,1 (4)	LSF1,1 (5)	LSF1,1 (6)
B_24	LSF1,1 (3)	LSF1,1 (4)	LSF1,1 (5)
B_25	LSF1,1 (2)	LSF1,1 (3)	LSF1,1 (4)
B_26	LSF1,1 (1)	LSF1,1 (2)	LSF1,1 (3)
B_27	LSF1,1 (0)	LSF1,1 (1)	LSF1,1 (2)
B_28	LSF2,3 (3)	LSF1,1 (0)	LSF1,1 (1)
B_29	LSF2,3 (2)	LSF2,3 (3)	LSF1,1 (0)
B_30	LSF2,3 (1)	LSF2,3 (2)	LSF2,3 (3)
B_31	LSF2,3 (0)	LSF2,3 (1)	LSF2,3 (2)
B_32	LSF2,2 (4)	LSF2,3 (0)	LSF2,3 (1)
B_33	LSF2,2 (3)	LSF2,2 (4)	LSF2,3 (0)
B_34	LSF2,2 (2)	LSF2,2 (3)	LSF2,2 (4)
B_35	LSF2,2 (1)	LSF2,2 (2)	LSF2,2 (3)
B_36	LSF2,2 (0)	LSF2,2 (1)	LSF2,2 (2)
B_37	LSF2,1 (6)	LSF2,2 (0)	LSF2,2 (1)
B_38	LSF2,1 (5)	LSF2,1 (5)	LSF2,2 (0)
B_39	LSF2,1 (4)	LSF2,1 (4)	LSF2,1 (6)
B_40	LSF2,1 (3)	LSF2,1 (3)	LSF2,1 (5)

```
+--------+------------+------------+------------+
|  B_41  | LSF2,1 (2) | LSF2,1 (2) | LSF2,1 (4) |
|  B_42  | LSF2,1 (1) | LSF2,1 (1) | LSF2,1 (3) |
|  B_43  | LSF2,1 (0) | LSF2,1 (0) | LSF2,1 (2) |
|  B_44  | GAIN2  (4) | GAIN2  (4) | LSF2,1 (1) |
|  B_45  | GAIN2  (3) | GAIN2  (3) | LSF2,1 (0) |
|  B_46  | GAIN2  (2) | GAIN2  (2) | GAIN1  (8) |
|  B_47  | GAIN2  (1) | GAIN2  (1) | GAIN1  (7) |
|  B_48  | GAIN2  (0) | GAIN2  (0) | GAIN1  (6) |
+--------+------------+------------+------------+
|  B_49  | GAIN1  (5) | GAIN1  (5) | GAIN1  (5) |
|  B_50  | GAIN1  (4) | GAIN1  (4) | GAIN1  (4) |
|  B_51  | GAIN1  (3) | GAIN1  (3) | GAIN1  (3) |
|  B_52  | GAIN1  (2) | GAIN1  (2) | GAIN1  (2) |
|  B_53  | GAIN1  (1) | GAIN1  (1) | GAIN1  (1) |
|  B_54  | GAIN1  (0) | GAIN1  (0) | GAIN1  (0) |
+--------+------------+------------+------------+
```

```
Notes:
xxxx (0) = LSB
xxxx (nbits-1) = MSB
LSF1,p = MSVQ* index of the pth stage of the two first frames
LSF2,p = MSVQ index of the pth stage of the two last frames
GAIN1 = VQ/MSVQ index of the 1st stage
GAIN2 = MSVQ index of the 2nd stage
* MSVQ: Multi-Stage Vector Quantizer
```

Table 4: Bitstream Definition for MELPe 600 bps (Part 2 of 2)

The 600 bps MELPe RTP payload is constructed as per Figure 4. Note
that bit B_01 is placed in the LSB of the first byte with all other
bits in sequence. When filling octets, the least significant bits of
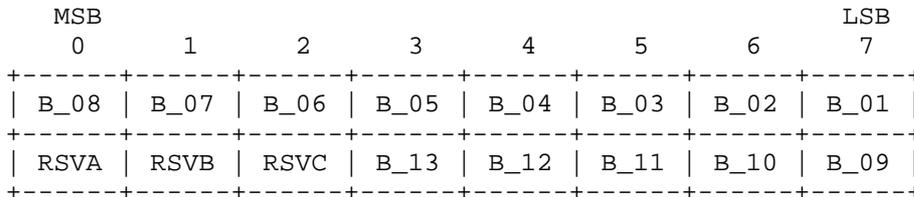the seventh octet are filled with bits B_49 to B_54, respectively.

```
   MSB                                                        LSB
    0       1       2       3       4       5       6       7
 +------+------+------+------+------+------+------+------+
 | B_08 | B_07 | B_06 | B_05 | B_04 | B_03 | B_02 | B_01 |
 +------+------+------+------+------+------+------+------+
 | B_16 | B_15 | B_14 | B_13 | B_12 | B_11 | B_10 | B_09 |
 +------+------+------+------+------+------+------+------+
 | B_24 | B_23 | B_22 | B_21 | B_20 | B_19 | B_18 | B_17 |
 +------+------+------+------+------+------+------+------+
 | B_32 | B_31 | B_30 | B_29 | B_28 | B_27 | B_26 | B_25 |
 +------+------+------+------+------+------+------+------+
 | B_40 | B_39 | B_38 | B_37 | B_36 | B_35 | B_34 | B_33 |
 +------+------+------+------+------+------+------+------+
 | B_48 | B_47 | B_46 | B_45 | B_44 | B_43 | B_42 | B_41 |
 +------+------+------+------+------+------+------+------+
 | RSVA | RSVB | B_54 | B_53 | B_52 | B_51 | B_50 | B_49 |
 +------+------+------+------+------+------+------+------+
```
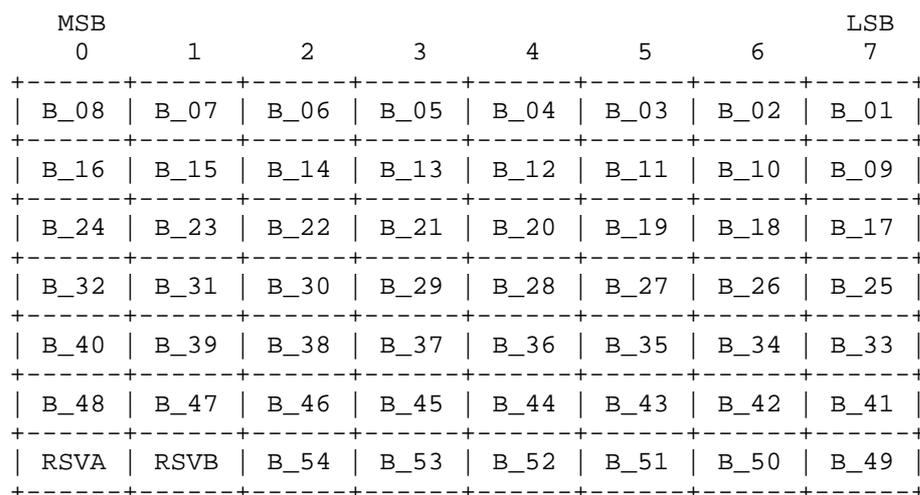
              Figure 4: Packed MELPe 600 bps Payload Octets

3.2. MELPe Comfort Noise Bitstream Definition

   Table B.3-1 of [SCIP210] identifies the usage of MELPe 2400 bps
   parameters for conveying comfort noise.

MELPe Parameter	Value
msvq[0] (line spectral frequencies)	* See Note
msvq[1] (line spectral frequencies)	Set to 0
msvq[2] (line spectral frequencies)	Set to 0
msvq[3] (line spectral frequencies)	Set to 0
fsvq (Fourier magnitudes)	Set to 0
gain[0] (gain)	Set to 0
gain[1] (gain)	* See Note
pitch (pitch - overall voicing)	Set to 0
bp (bandpass voicing)	Set to 0
af (aperiodic flag/jitter index)	Set to 0
sync (sync bit)	Alternations

   Note:
   The default values are the respective parameters from the
   vocoder frame. It is preferred that msvq[0] and gain[1]
   values be derived by averaging the respective parameter from
   some number of previous vocoder frames.

                  Table 5: MELPe Comfort Noise Parameters

Since only msvq[0] (also known as LSF1x or the first LSP) and gain[1]
(also known as g2x or the second gain) are needed, the following bit
order is used for comfort noise frames:

```
              +--------+-------------+
              |  Bit   |  Comfort    |
              |        |   Noise     |
              +--------+-------------+
              |  B_01  |    LSF10    |
              |  B_02  |    LSF11    |
              |  B_03  |    LSF12    |
              |  B_04  |    LSF13    |
              |  B_05  |    LSF14    |
              |  B_06  |    LSF15    |
              |  B_07  |    LSF16    |
              |  B_08  |     g20     |
              +--------+-------------+
              |  B_09  |     g21     |
              |  B_10  |     g22     |
              |  B_11  |     g23     |
              |  B_12  |     g24     |
              |  B_13  |    SYNC     |
              +--------+-------------+

              Notes:
              g = Gain
              LSF = Line Spectral Frequencies
```

        Table 6: Bitstream Definition for MELPe Comfort Noise

The comfort noise MELPe RTP payload is constructed as per Figure 5.
Note that bit B_01 is placed in the LSB of the first byte with all
other bits in sequence. When filling octets, the least significant
bits of the second octet are filled with bits B_09 to B_13,
respectively.

```
      MSB                                              LSB
       0      1      2      3      4      5      6      7
     +------+------+------+------+------+------+------+------+
     | B_08 | B_07 | B_06 | B_05 | B_04 | B_03 | B_02 | B_01 |
     +------+------+------+------+------+------+------+------+
     | RSVA | RSVB | RSVC | B_13 | B_12 | B_11 | B_10 | B_09 |
     +------+------+------+------+------+------+------+------+
```

        Figure 5: Packed MELPe Comfort Noise Payload Octets

3.3. Multiple MELPe Frames in an RTP Packet

   A MELPe RTP packet MAY consist of zero or more MELPe coder frames
   followed by zero or one MELPe comfort noise frame. The presence of a
   comfort noise frame can be deduced from the length of the RTP
   payload. The default packetization interval is one coder frame
   (22.5, 67.5, or 90 ms) according to the coder bitrate (2400, 1200, or
   600 bps). For some applications, a longer packetization interval is
   used to reduce the packet rate.

   A MELPe RTP packet comprised of no coder frame and no comfort noise
   frame MAY be used periodically by an endpoint to indicate
   connectivity by an otherwise idle receiver.

   All MELPe frames in a single RTP packet MUST be of the same coder
   bitrate. Dynamic switching between frame rates within an RTP stream
   may be permitted (if supported by both ends) provided that reserved
   bits RSVA, RSVB, and RSVC are filled in as per Table 7. If bitrate
   switching is not used, all reserved bits are encoded as 0 by the
   sender and ignored by the receiver. (RSV0 is always coded as 0.)

Coder Bitrate	RSVA	RSVB	RSVC
2400 bps	0	0	N/A
1200 bps	1	0	0
600 bps	0	1	N/A
Comfort Noise	1	0	1
(reserved)	1	1	N/A

                  Table 7: MELPe Frame Bitrate Indicators

   It is important to observe that senders have the following additional
   restrictions:

   Senders SHOULD NOT include more MELPe frames in a single RTP packet
   than will fit in the MTU of the RTP transport protocol.

   Frames MUST NOT be split between RTP packets.

It is RECOMMENDED that the number of frames contained within an RTP
packet be consistent with the application. For example, in telephony
and other real-time applications where delay is important, then the
fewer frames per packet the lower the delay, whereas for bandwidth-
constrained links or delay-insensitive streaming messaging
applications, more than one frame per packet or many frames per
packet would be acceptable.

Information describing the number of frames contained in an RTP
packet is not transmitted as part of the RTP payload. The way to
determine the number of MELPe frames is to count the total number of
octets within the RTP packet and divide the octet count by the number
of expected octets per frame (7/11/7 per frame). Keep in mind that
the last frame can be a 2-octet comfort noise frame.

When dynamic bitrate switching is used and more than one frame is
contained in an RTP packet, it is RECOMMENDED that the coder rate
bits contained in the last octet be inspected. If the coder bitrate
indicates a comfort noise frame, then inspect the third last octet
for the coder bitrate. All MELPe speech frames in the RTP packet
will be of this same coder bitrate.

3.4. Congestion Control Considerations

The target bitrate of MELPe can be adjusted at any point in time,
thus allowing congestion management. Furthermore, the amount of
encoded speech or audio data encoded in a single packet can be used
for congestion control, since the packet rate is inversely
proportional to the packet duration. A lower packet transmission
rate reduces the amount of header overhead but at the same time
increases latency and loss sensitivity, so it ought to be used
with care.

Since UDP does not provide congestion control, applications that use
RTP over UDP SHOULD implement their own congestion control above the
UDP layer [RFC8085] and MAY also implement a transport circuit
breaker [RFC8083]. Work in the RMCAT working group [RMCAT] describes
the interactions and conceptual interfaces necessary between the
application components that relate to congestion control, including
the RTP layer, the higher-level media codec control layer, and the
lower-level transport interface, as well as components dedicated to
congestion control functions.

4. Payload Format Parameters

   This RTP payload format is identified using the MELP, MELP2400,
   MELP1200, and MELP600 media subtypes, which are registered in
   accordance with RFC 4855 [RFC4855] and per the media type
   registration template from RFC 6838 [RFC6838].

4.1. Media Type Definitions

   Type name: audio

   Subtype names: MELP, MELP2400, MELP1200, and MELP600

   Required parameters: N/A

   Optional parameters:

      ptime: the recommended length of time (in milliseconds)
         represented by the media in a packet. It SHALL use the nearest
         rounded-up ms integer packet duration. For MELPe, this
         corresponds to the following values: 23, 45, 68, 90, 112, 135,
         156, and 180. Larger values can be used as long as they are
         properly rounded. See Section 6 of RFC 4566 [RFC4566].

      maxptime: the maximum length of time (in milliseconds) that can be
         encapsulated in a packet. It SHALL use the nearest rounded-up
         ms integer packet duration. For MELPe, this corresponds to the
         following values: 23, 45, 68, 90, 112, 135, 156, and 180.
         Larger values can be used as long as they are properly rounded.
         See Section 6 of RFC 4566 [RFC4566].

      bitrate: specifies the MELPe coder bitrates supported. Possible
         values are a comma-separated list of rates from the following
         set: 2400, 1200, 600. The modes are listed in order of
         preference; first is preferred. If "bitrate" is not present,
         the fixed coder bitrate of 2400 MUST be used. The alternate
         encoding names "MELP2400", "MELP1200", and "MELP600" directly
         specify the MELPe coder bitrates of 2400, 1200, and 600,
         respectively, and MUST NOT specify a "bitrate" parameter.

   Encoding considerations: These media subtypes are framed and binary;
      see Section 4.8 of RFC 6838 [RFC6838].

   Security considerations: Please see Section 8 of RFC 8130.

   Interoperability considerations: Early implementations used MELP2400,
      MELP1200, and MELP600 to indicate both coder type and bitrate.
      These media type names should be preserved with this registration.

   Published specification: N/A

   Applications that use this media type: N/A

   Additional information: N/A

      Deprecated alias names for this type: N/A

      Magic number(s): N/A

      File extension(s): N/A

      Macintosh file type code(s): N/A

   Person & email address to contact for further information:

      Victor Demjanenko, Ph.D.
      VOCAL Technologies, Ltd.
      520 Lee Entrance, Suite 202
      Buffalo, NY 14228
      United States of America
      Phone: +1 716 688 4675
      Email: victor.demjanenko@vocal.com

   Intended usage: COMMON

   Restrictions on usage: These media subtypes depend on RTP framing and
      hence are only defined for transfer via RTP [RFC3550]. Transport
      within other framing protocols is not defined at this time.

   Author: Victor Demjanenko

   Change controller: IETF Payload working group delegated from the
      IESG.

   Provisional registration? (standards tree only): No

4.2. Mapping to SDP

   The mapping of the above-defined payload format media subtypes and
   their parameters SHALL be done according to Section 3 of RFC 4855
   [RFC4855].

The information carried in the media type specification has a
specific mapping to fields in the Session Description Protocol (SDP)
[RFC4566], which is commonly used to describe RTP sessions. When SDP
is used to specify sessions employing the MELPe codec, the mapping is
as follows:

o The media type ("audio") goes in SDP "m=" as the media name.

o The media subtype (payload format name) goes in SDP "a=rtpmap" as
   the encoding name.

o The parameter "bitrate" goes in the SDP "a=fmtp" attribute by
   copying it as a "bitrate=<value>" string.

o The parameters "ptime" and "maxptime" go in the SDP "a=ptime" and
   "a=maxptime" attributes, respectively.

When conveying information via SDP, the encoding name SHALL be "MELP"
(the same as the media subtype). Alternate encoding name subtypes
"MELP2400", "MELP1200", and "MELP600" MAY be used in SDP to convey
fixed-bitrate configurations. These names have been observed in
systems that do not support dynamic frame-rate switching as specified
by the parameter "bitrate".

An example of the media representation in SDP for describing MELPe
might be:

    m=audio 49120 RTP/AVP 97
    a=rtpmap:97 MELP/8000

An alternative example of SDP for fixed-bitrate configurations
might be:

    m=audio 49120 RTP/AVP 97 100 101 102
    a=rtpmap:97 MELP/8000
    a=rtpmap:100 MELP2400/8000
    a=rtpmap:101 MELP1200/8000
    a=rtpmap:102 MELP600/8000

If the encoding name "MELP" is received without a "bitrate"
parameter, the fixed coder bitrate of 2400 MUST be used. The
alternate encoding names "MELP2400", "MELP1200", and "MELP600"
directly specify the MELPe coder bitrates of 2400, 1200, and 600,
respectively, and MUST NOT specify a "bitrate" parameter.

The optional media type parameter "bitrate", when present, MUST be
included in the "a=fmtp" attribute in the SDP, expressed as a media
type string in the form of a semicolon-separated list of

parameter=value pairs. The string "value" can be one or more of
2400, 1200, and 600, separated by commas (where each bitrate value
indicates the corresponding MELPe coder). An example of the media
representation in SDP for describing MELPe when all three coder
bitrates are supported might be:

```
m=audio 49120 RTP/AVP 97
a=rtpmap:97 MELP/8000
a=fmtp:97 bitrate=2400,600,1200
```

Parameter "ptime" cannot be used for the purpose of specifying the
MELPe operating mode, due to the fact that for certain values it will
be impossible to distinguish which mode is about to be used (e.g.,
when ptime=68, it would be impossible to distinguish if the packet is
carrying one frame of 67.5 ms or three frames of 22.5 ms).

Note that the payload format (encoding) names are commonly shown in
upper case. Media subtypes are commonly shown in lower case. These
names are case insensitive in both places. Similarly, parameter
names are case insensitive in both the media subtype name and the
default mapping to the SDP a=fmtp attribute.

4.3. Declarative SDP Considerations

For declarative media, the "bitrate" parameter specifies the possible
bitrates used by the sender. Multiple MELPe rtpmap values (such as
97, 98, and 99, as used below) MAY be used to convey MELPe-coded
voice at different bitrates. The receiver can then select an
appropriate MELPe codec by using 97, 98, or 99.

```
m=audio 49120 RTP/AVP 97 98 99
a=rtpmap:97 MELP/8000
a=fmtp:97 bitrate=2400
a=rtpmap:98 MELP/8000
a=fmtp:98 bitrate=1200
a=rtpmap:99 MELP/8000
a=fmtp:99 bitrate=600
```

4.4. Offer/Answer SDP Considerations

In the Offer/Answer model [RFC3264], "bitrate" is a bidirectional
parameter. Both sides MUST use a common "bitrate" value or values.
The offer contains the bitrates supported by the offerer, listed in
its preferred order. The answerer MAY agree to any bitrate by
listing the bitrate first in the answerer response. Additionally,
the answerer MAY indicate any secondary bitrate or bitrates that it
supports. The initial bitrate used by both parties SHALL be the
first bitrate specified in the answerer response.

For example, if offerer bitrates are "2400,600" and answer bitrates
are "600,2400", the initial bitrate is 600. If other bitrates are
provided by the answerer, any common bitrate between the offer and
answer MAY be used at any time in the future. Activation of these
other common bitrates is beyond the scope of this document.

The use of a lower bitrate is often important for a case such as when
one endpoint utilizes a bandwidth-constrained link (e.g., 1200 bps
radio link or slower), where only the lower coder bitrate will work.

5. Discontinuous Transmissions

   A primary application of MELPe is for radio communications of voice
   conversations, and discontinuous transmissions are normal. When
   MELPe is used in an IP network, MELPe RTP packet transmissions may
   cease and resume frequently. RTP synchronization source (SSRC)
   sequence number gaps indicate lost packets to be filled by PLC, while
   abrupt loss of RTP packets indicates intended discontinuous
   transmissions.

   If a MELPe coder so desires, it may send a comfort noise frame as per
   Appendix B of [SCIP210] prior to ceasing transmission. A receiver
   may optionally use comfort noise during its silence periods. No SDP
   negotiations are required.

6. Packet Loss Concealment

   MELPe packet loss concealment (PLC) uses the special properties and
   coding for the pitch/voicing parameter of the MELPe 2400 bps coder.
   The PLC erasure indication utilizes any of the errored encodings of a
   non-voiced frame as identified in Table 1 of [MELPE]. For the sake
   of simplicity, it is preferred that a code value of 3 for the
   pitch/voicing parameter (represented by the bits P6 to P0 in Table 1
   of this document) be used. Hence, set bits P0 and P1 to one and bits
   P2, P3, P4, P5, and P6 to zero.

   When using PLC in 1200 bps or 600 bps mode, the MELPe 2400 bps
   decoder is called three or four times, respectively, to cover the
   loss of a MELPe frame.

7. IANA Considerations

   IANA has registered MELP, MELP2400, MELP1200, and MELP600 as
   specified in Section 4.1. IANA has also added these media subtypes
   to the "RTP Payload Format media types" registry
   (http://www.iana.org/assignments/rtp-parameters).

8. Security Considerations

   RTP packets using the payload format defined in this specification
   are subject to the security considerations discussed in the RTP
   specification [RFC3550] and in any applicable RTP profile such as
   RTP/AVP [RFC3551], RTP/AVPF [RFC4585], RTP/SAVP [RFC3711], or
   RTP/SAVPF [RFC5124]. However, as discussed in [RFC7202], it is not
   an RTP payload format's responsibility to discuss or mandate what
   solutions are used to meet such basic security goals as
   confidentiality, integrity, and source authenticity for RTP in
   general. This responsibility lies with anyone using RTP in an
   application. They can find guidance on available security mechanisms
   and important considerations in [RFC7201]. Applications SHOULD use
   one or more appropriate strong security mechanisms. The rest of this
   section discusses the security-impacting properties of the payload
   format itself.

   This RTP payload format and the MELPe decoder do not exhibit any
   significant non-uniformity in the receiver-side computational
   complexity for packet processing and thus are unlikely to pose a
   denial-of-service threat due to the receipt of pathological data.
   Additionally, the RTP payload format does not contain any active
   content.

   Please see the security considerations discussed in [RFC6562]
   regarding VAD and its effect on bitrates.

9. References

9.1. Normative References

   [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
              Requirement Levels", BCP 14, RFC 2119,
              DOI 10.17487/RFC2119, March 1997,
              <http://www.rfc-editor.org/info/rfc2119>.

   [RFC2736] Handley, M. and C. Perkins, "Guidelines for Writers of RTP
              Payload Format Specifications", BCP 36, RFC 2736,
              DOI 10.17487/RFC2736, December 1999,
              <http://www.rfc-editor.org/info/rfc2736>.

   [RFC3264] Rosenberg, J. and H. Schulzrinne, "An Offer/Answer Model
              with Session Description Protocol (SDP)", RFC 3264,
              DOI 10.17487/RFC3264, June 2002,
              <http://www.rfc-editor.org/info/rfc3264>.

   [RFC3550] Schulzrinne, H., Casner, S., Frederick, R., and V.
              Jacobson, "RTP: A Transport Protocol for Real-Time
              Applications", STD 64, RFC 3550, DOI 10.17487/RFC3550,
              July 2003, <http://www.rfc-editor.org/info/rfc3550>.

   [RFC3551] Schulzrinne, H. and S. Casner, "RTP Profile for Audio and
              Video Conferences with Minimal Control", STD 65, RFC 3551,
              DOI 10.17487/RFC3551, July 2003,
              <http://www.rfc-editor.org/info/rfc3551>.

   [RFC3711] Baugher, M., McGrew, D., Naslund, M., Carrara, E., and K.
              Norrman, "The Secure Real-time Transport Protocol (SRTP)",
              RFC 3711, DOI 10.17487/RFC3711, March 2004,
              <http://www.rfc-editor.org/info/rfc3711>.

   [RFC4566] Handley, M., Jacobson, V., and C. Perkins, "SDP: Session
              Description Protocol", RFC 4566, DOI 10.17487/RFC4566,
              July 2006, <http://www.rfc-editor.org/info/rfc4566>.

   [RFC4855] Casner, S., "Media Type Registration of RTP Payload
              Formats", RFC 4855, DOI 10.17487/RFC4855, February 2007,
              <http://www.rfc-editor.org/info/rfc4855>.

   [RFC5124] Ott, J. and E. Carrara, "Extended Secure RTP Profile for
              Real-time Transport Control Protocol (RTCP)-Based Feedback
              (RTP/SAVPF)", RFC 5124, DOI 10.17487/RFC5124,
              February 2008, <http://www.rfc-editor.org/info/rfc5124>.

   [RFC6562] Perkins, C. and JM. Valin, "Guidelines for the Use of
              Variable Bit Rate Audio with Secure RTP", RFC 6562,
              DOI 10.17487/RFC6562, March 2012,
              <http://www.rfc-editor.org/info/rfc6562>.

   [RFC6838] Freed, N., Klensin, J., and T. Hansen, "Media Type
              Specifications and Registration Procedures", BCP 13,
              RFC 6838, DOI 10.17487/RFC6838, January 2013,
              <http://www.rfc-editor.org/info/rfc6838>.

   [RFC8083] Perkins, C. and V. Singh, "Multimedia Congestion Control:
              Circuit Breakers for Unicast RTP Sessions", RFC 8083,
              DOI 10.17487/RFC8083, March 2017,
              <http://www.rfc-editor.org/info/rfc8083>.

   [RFC8085] Eggert, L., Fairhurst, G., and G. Shepherd, "UDP Usage
              Guidelines", RFC 8085, DOI 10.17487/RFC8085, March 2017,
              <http://www.rfc-editor.org/info/rfc8085>.

   [MELP]   Department of Defense Telecommunications Standard,
               "Analog-to-Digital Conversion of Voice by 2,400 Bit/Second
               Mixed Excitation Linear Prediction (MELP)", MIL-STD-3005,
               December 1999.

   [MELPE]   North Atlantic Treaty Organization (NATO), "The 600 Bit/S,
               1200 Bit/S and 2400 Bit/S NATO Interoperable Narrow Band
               Voice Coder", STANAG No. 4591, January 2006.

   [SCIP210]   National Security Agency, "SCIP Signaling Plan", SCIP-210,
               December 2007.

9.2. Informative References

   [RFC4585]   Ott, J., Wenger, S., Sato, N., Burmeister, C., and J. Rey,
               "Extended RTP Profile for Real-time Transport Control
               Protocol (RTCP)-Based Feedback (RTP/AVPF)", RFC 4585,
               DOI 10.17487/RFC4585, July 2006,
               <http://www.rfc-editor.org/info/rfc4585>.

   [RFC7201]   Westerlund, M. and C. Perkins, "Options for Securing RTP
               Sessions", RFC 7201, DOI 10.17487/RFC7201, April 2014,
               <http://www.rfc-editor.org/info/rfc7201>.

   [RFC7202]   Perkins, C. and M. Westerlund, "Securing the RTP
               Framework: Why RTP Does Not Mandate a Single Media
               Security Solution", RFC 7202, DOI 10.17487/RFC7202,
               April 2014, <http://www.rfc-editor.org/info/rfc7202>.

   [RMCAT]   IETF, RTP Media Congestion Avoidance Techniques (rmcat)
               Working Group,
               <https://datatracker.ietf.org/wg/rmcat/about/>.

Authors' Addresses

    Victor Demjanenko, Ph.D.
    VOCAL Technologies, Ltd.
    520 Lee Entrance, Suite 202
    Buffalo, NY 14228
    United States of America

    Phone: +1 716 688 4675
    Email: victor.demjanenko@vocal.com

    David Satterlee
    VOCAL Technologies, Ltd.
    520 Lee Entrance, Suite 202
    Buffalo, NY 14228
    United States of America

    Phone: +1 716 688 4675
    Email: david.satterlee@vocal.com