SMVLite: Reduced Complexity Selectable Mode Vocoder

C.V.Goudar, Pankaj Rabha, Murali Deshpande, Ajit Rao

Texas Instruments Inc.

ABSTRACT

This paper describes the recently developed SMVLite speech codec. SMVLite is the reduced complexity variant of the new 3GPP2 (3rd generation partnership project 2) CDMA standard SMV (Selectable Mode Vocoder). SMV provides superior speech quality at low bit-rates compared to other CDMA codecs. However, its computational complexity is significantly higher than other CDMA standards, thereby rendering it inefficient for real-time implementation. We have developed a lower complexity version of the SMV called SMVLite. SMVLite is bit-stream interoperable with SMV and its voice quality is perceptually equivalent to SMV in all modes & conditions of interest. The computational complexity of SMVLite is 25% lower than SMV. The voice quality equivalence of SMV and SMVLite has been conclusively proven in a formal subjective listening test conducted at Dynastat.

1. INTRODUCTION

The SMV (Selectable Mode Vocoder) [1][2] was standardized in 2001 by the 3GPP2 (3rd Generation Partnership Project 2) as the next generation speech codec for CDMA networks. Prior to SMV, two other speech codecs - Q13 (Qualcomm 13) & EVRC (Enhanced Variable Rate Codec) [3] - were the standard codecs allowed in CDMA. SMV provides good voice quality with highest possible bandwidth utilization among the three CDMA codecs. However SMV has a higher computational requirement when compared to EVRC and Q13. The high complexity results in fewer simultaneous SMV channels being supported on a single chip compared to other codecs. This results in reduced efficiency and higher cost of mobile infrastructure equipment. The need to reduce the cost of these systems has motivated us to modify SMV to reduce its complexity while continuing to meet the Minimum Performance Specification (MPS) [6] requirement for SMV implementations. Our modified codec, called SMVLite, is fully compatible with SMV - speech coded by an SMVLite encoder can be decoded by standard SMV and vice versa. Moreover, the subjective voice quality of SMVLite is equivalent to SMV in the same conditions & at the same average bit-rate.

This paper describes the SMVLite approach. While SMVLite builds on SMV, several modifications result in approximately 25% reduction in computational complexity. Formal subjective listening tests conducted at Dynastat Labs shows that SMVLite is equivalent to the SMV for all test conditions.

This paper is organized as follows. Section II covers a brief overview of standard SMV. Section III describes algorithmic changes to SMV that result in SMVLite and finally Section IV summarizes the results of the subjective listening tests.

2. SMV OVERVIEW

SMV is a multiple rate, multi-mode codec. The mono input speech signal is sampled at 8 kHz and segmented into non-overlapping frames of length 20ms (160 samples) each. Each frame is encoded at one of four bit-rates: (i) 8.55 kbps i.e. 171 bits per frame (full-rate), (ii) 4.0 kbps i.e. 80 bits per frame (half-rate), (iii) 2.0 kbps i.e. 40 bits per frame (quarter-rate) and (iv) 0.8 kbps i.e. 16 bits per frame(one- eighth rate). The bit-rate used for each frame depends on the voice activity in the speech signal during that frame as well as a user & system dependent “mode”. SMV defines 3 official modes – modes 0, 1 & 2. Typically, the mode is determined during call-setup although it may change occasionally during the call due to changes in network conditions or user preferences. Amongst the 3 modes, mode 0 corresponds to the highest average bit-rate (ABR) and therefore the highest voice quality. Mode 1 has an intermediate ABR and voice quality, while Mode 2 is the most efficient in bit-rate with the lowest quality.

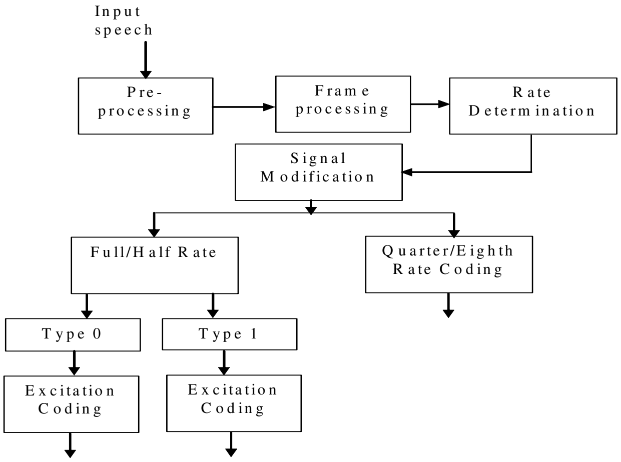

FIGURE 1: HIGH-LEVEL BLOCK DIAGRAM OF SMV

Figure 1 shows a high-level block diagram of the SMV codec. SMV is based on eXtended CELP (eX-CELP) technology [4] which is a variant of the popular Codebook Excited Linear Prediction (CELP) approach common to most speech codecs. The input speech signal is subjected to a pre-processing algorithm that includes high-pass filtering, noise suppression (similar to EVRC), and adaptive tilt compensation. Pre-processing cleans up the background noise in the voice signal and prepares it for the subsequent coding steps. Next, frame-level processing is performed. This includes Linear Predictive Coding (LPC) analysis & open-loop pitch analysis. LPC and pitch-analysis are standard steps common to CELP-based codecs. They help identify the parameters that allow the codec to exploit short-term and long-term correlation between the samples of the voice signal.

Next, the frame classification algorithm is invoked. The speech signal in each frame is analyzed and classified depending on the nature of the voice. The permitted classes are (i) silence or background noise, (ii) noise-like unvoiced, (iii) unvoiced, (iv) speech onset, (v) non-stationary voiced, or (vi) stationary voiced. Next, the rate determination algorithm (RDA) selects one of the 4 possible bit-rates (full, half, quarter, one-eighth) depending on the class of the frame and the current mode. Silent / background noise frames and stationary unvoiced frames are coded at either the one-eighth rate (0.8 kbps i.e. 16 bits per frame) or quarter rate (2.0 kbps or 40 bits per frame) depending on the mode. All other frames may be encoded at 4.0 kbps or 8.55 kbps depending on the mode.

The concept of signal modification is integral to the eX-CELP algorithm just as it is to EVRC. The inspiration for eX-CELP is the RCELP (Relaxed Code Excited Linear Prediction) [5] coding strategy used in EVRC. SMV follows the signal modification strategy of RCELP but unlike RCELP where the residual is modified, in eX-CELP, the weighted speech signal is non-linearly warped to regularize its pitch contour. Following the signal modification step, the half and full-rate frames are reanalyzed to determine a "frame type". Frames which demonstrate a high long-term correlation after signal modification are declared as Type 1 frames while those with less long-term correlation are declared as Type 0 frames. Combined with the bit-rate assigned, this results in four distinct types of frames at half and full-rates: Type 0 half-rate, Type 1 half-rate, Type 0 full-rate and Type1 full-rate.

LPC parameters in SMV are encoded using a predictive LSF (Line Spectral Frequencies) scheme. The frame LPCs are transformed into LSFs through a root-search algorithm. A predictor is used to predict the frame LSFs from the past frames and the residual prediction error is vector-quantized using a standard weighted-mean-squared error (WMSE) measure.

2.1 Excitation Coding

In the next important step, we code the speech excitation signal. The one-eighth and quarter rate modes are the easiest and least complex to code. In SMV, these frames are coded using spectrum and energy modulated random noise models.

The full-rate and half rate codecs in the SMV are coded using the eX-CELP excitation coding algorithm. Here, traditional analysis-by-synthesis (closed loop) search that is common to most CELP codecs is combined with perceptually based decisions (open loop) for improved voice quality.

Type 0 Full-rate frames are divided into 4 sub-frames. As in traditional CELP codecs, an adaptive codebook (ACB) search is performed for each subframe and followed by a closed loop fixed codebook (FCB) search. The adaptive and the fixed codebook gains for each subframe are jointly quantized using a 2-dimensional vector quantizer. Type 0 Half-rate frames are coded similarly, except 2 subframes are used instead of 4.

Type 1 frames are divided into 4 subframes (at full-rate) and 3 subframes (at half-rate). The pitch value derived from the open loop pitch search performed during frame-level processing is used to represent the long-term correlation. Closed-loop pitch is not necessary due to the high long-term pitch prediction gains and stable pitch contour of type 1 frames. FCB search is performed for each subframe. The pitch gains for all the subframes are jointly vector-quantized.

The FCB approach is to have multiple sub-codebooks which must be searched for highest optimality. The only exception is the full rate Type 1 sub-frames where only one codebook is used. Each sub-codebook has been pre-designed to provide the best possible representation of a particular type of speech excitation. Depending on the frame type and the rate, the sub-codebooks may consist of pulse excitations or random excitations. The choice of the best codebooks to search through and best code vectors to use is based on the closed loop metric (weighted error measure), as well as additional information such as background noise conditions, peakiness of the speech etc. The search process through pulse-based codebooks is iterative. Pulse locations are added sequentially or two-at-a-time but re-optimized in subsequent passes. The codebook search is performed via a closed loop error minimization. Perceptual considerations are used to minimize search and improve overall voice quality.

While standard SMV offers superior voice quality at low bit-rates, this comes at the cost of increased computational complexity which taxes real-time implementations resulting in lower channel densities (number of simultaneous voice calls possible on a single real-time system), and thereby leading to increased system cost. For the purpose of this discussion, we measure computational complexity via the Weighted Million Operations Per Second (WMOPS) approach that is reasonably independent of the choice of the voice processor. Table 1 shows a break-up of the worst case computational complexity of SMV for the computationally expensive sub-modules.

Module	WMOPS
	SMV
Signal Modification	6.11
LSF Quantization	3.26
Fixed Codebook Search	11.32

TABLE 1: COMPLEXITY BREAK-DOWN OF SMV

3. SMVLITE

We propose next, a sequence of steps, some simple, others innovative to significantly reduce the algorithmic complexity of an SMV implementation without impacting its compatibility with standard SMV, nor degrading its voice quality or average bit-rate characteristics. The resulting implementation, which we refer to as SMVLite is compliant with the minimum performance specifications (MPS) of the 3GPP2 standards body. 3GPP2 requires SMV implementations to be equivalent in average bit-rate and voice quality with the standard while being fully inter-operable. Bit-exact implementations are not required.

We focused our complexity reduction efforts on the three critical sub-modules: (a) signal modification, (b) LSF quantization and (c) FCB search. We describe the specific changes proposed in the following sub-sections.

3.1 Signal Modification

Signal modification in SMV is performed via continuous warping of the weighted speech signal. First, the weighted speech signal is analyzed to locate significant pitch pulses. Next, the frame is divided into variable-length sub-frames centered around the pitch pulses identified. Careful checks ensure continuity with the past frames. It is important to note that the length of the pitch sub-frames depends on the local pitch value. Note also that the variable length sub-frames identified here are used only for signal modification purposes and are distinct and different from the fixed sub-frames used for excitation coding subsequently. Having identified the variable length subframes, we next search and determine a more accurate value for the local pitch delay by performing fractional pitch search in each variable length subframe. The accurate pitch values thus obtained are used to perform non-linear warping on the speech signal to regularize the pitch contour for low-rate excitation coding.

The complexity of the pitch search & signal modification steps is proportional to the total number of the variable length subframes. In particular, for shorter pitch lags (female speech), there are a large number of short subframes resulting in very high computational complexity for the pitch search & warping algorithms. In SMVLite, we propose an innovative search reduction method for signal modification. Regardless of pitch value, we lower-bound the size of the variable size subframe by 12 samples. Careful steps allow us to ensure that the warping regions thus identified do not split any pitch peaks. This leads to significant reduction in the worst-case WMOPS – the worst-case cost of 6.11 WMOPS for signal modification is reduced by 46% to 3.26 WMOPS. (see Table 1).

3.2 LSF Quantization

Our second complexity improvement was achieved on the LSF quantization algorithm. As noted earlier, the LSFs are coded via vector quantization of the predictive residual with a WMSE metric:

$$WMSE_{LSF}^{j,l,r} = \sum_{i=1}^{10} w_{LSF}(i)\left(I_{LSF}^{j,l}(i) - C_{LSF}^{j,r}(i)\right)^2 \quad \ldots(1)$$

In the equation above, the vector l represents the frame LSF and C represents the LSF codevector. j represents stage of predictive quantizer and l represents index of the input vector. In the standard SMV implementation, the above search is implemented using a fixed-point algorithm to directly minimize equation (1) thereby resulting in high computational complexity. In order to reduce the computational burden of the WMSE calculation, we propose a simple pre-computation based approach. In our SMVLite implementation, equation (1) can be expanded[1] as

$$\begin{aligned} WMSE_{LSF}^{j,l,r} = & \sum_{i=1}^{10} w_{LSF}(i)\left(I_{LSF}^{j,l}(i)\right)^2 + \sum_{i=1}^{10} w_{LSF}(i)\left(C_{LSF}^{j,r}(i)\right)^2 \quad \ldots(2) \\ & - 2\sum_{i=1}^{10} w_{LSF}(i)\left(I_{LSF}^{j,l}(i) \cdot C_{LSF}^{j,r}(i)\right) \end{aligned}$$

Clearly, the first two terms in (2) can be very efficiently pre-computed and stored prior to initiating the search steps thereby resulting in reduced complexity. While this simple pre-computation has no impact on the choice of the LSF code vector for a floating point implementation, there are differences between the results of equations 1 & 2 in fixed-point. The use of equation (2) instead of equation (1) results in a reduction of greater than 25% in WMOPS cost of LSF computation with imperceptible impact in the voice quality.

3.3 Fixed codebook (FCB) search

Fixed codebook (FCB) search is computationally, the most expensive block in SMV as well as most other CELP codecs. SMV uses multiple sub-codebooks for FCB search. There are rate & type specific constraints on the use of the sub-codebooks. We have used three specific techniques to reduce the FCB search complexity. These are (i) Reduced Backward Pitch Enhancement, (ii) Selective Joint Search and (iii) Codebook Search Space Reduction.

3.3.1Reduced Backward Pitch Enhancement

SMV uses an innovative backward pitch enhancement approach [1] in addition to the standard forward pitch enhancement used in many CELP codecs. In SMV as well as most other speech codecs, forward pitch enhancement usually costs little in additional computational complexity since it can be incorporated by simply modifying the impulse response for the weighted filter used in FCB search. However backward pitch enhancement cannot be easily incorporated in a similar manner and thereby results in higher computational cost. The additional complexity is especially high when the frame is divided into three subframes (i.e subframe length is 53 or 54 samples) and the pitch lag is very small. This is unusual in SMV by design. However it does occur occasionally, resulting in very high WMOPS for the worst case. In order to limit the worst-case WMOPS, we have used an approach that limits somewhat the use of backward pitch enhancements during FCB search. Standard SMV allows an impact of up to three pitch periods with exponentially decaying amplitude during backward pitch enhancement. However, in SMVLite, during certain circumstances that correspond to high computational complexity (three subframes per frame and pitch period less than a small threshold), we limit the impact to two pitch periods. We observe that the worst case computational complexity is reduced significantly as a result of this approximation with almost no impact on voice quality.

3.3.2 Selective Joint Search

SMVLite incorporates a novel pulse search strategy termed as the Selective Joint Search (SJS)[1]. SJS builds on the standard joint search methodology used in SMV. To understand SJS, let us understand the FCB search used in standard SMV. Consider Type1 Full-rate voice sub-frames - a single fixed codebook with 8 pulses is used. The standard search procedure used in SMV during codebook search for these pulses is to do two "turns" of iterations. In the first "turn", pulses are added one pair at a time (in four pairs). Each time a new pulse pair is added, all possible pulse positions for this new pair are considered and the best positions are chosen (full search) by minimizing the standard weighted error function in CELP. Once chosen, no further modifications are allowed to the pulse positions during the first turn. During the second turn, however, the positions of the pulses in each pair are allowed to be re-optimized. This re-optimization is performed sequentially for three pulse pairs. This is referred to as the second turn of the pulse search. Having completed the two turns above, two more turns of re-optimization are conduced. However in these (third and fourth) rounds, pulse positions are refined one at a time. The above algorithm is clearly very demanding computationally.

In SMVLite, we first conduct one turn of pulse search. During this search, pulses are introduced one at a time and pulse positions are optimized sequentially. The goal is only to get an estimate of the likely pulse position.

Next, the SJS approach is used. The philosophy of SJS is that instead of blind full-search optimization of pulse positions, we select and re-optimize pulse positions on only tracks which are likely to have a large impact on the voice quality when re-optimized. Tracks are selected based on the minimum contribution criterion i.e. we select only a small subset of tracks which contribute the least to the weighted squared-error during the first turn.

SJS is used not just in Type 1 Full-rate frames but other frame types as well. In Type 0 frames for example, SMV first selects one of three sub-codebooks as the "winner" sub-codebook before searching for the best code-vector in that sub-codebook. The winner sub-codebook is picked based on generating the least weighted error in a pre-search. The pre-search involves searching through each of the three sub-codebooks with one turn of single pulse search. The sub-codebook which minimizes the error criterion is picked as the "winner". Once the winner is determined through this pre-search, the best set of pulses may be re-optimized via joint search strategy involving multiple turns with pairs of

[1] Patent application submitted

pulses. In the SMVLite the procedure for picking the "winner" sub-codebook is identical to SMV. However, instead of blind full-search refinement in the second stage, SJS as described above is used. This approach leads to a huge savings in computational complexity with no impact on voice quality.

3.3.3 Codebook Search Space Reduction

Limiting the search space for code-vectors can significantly reduce the complexity of FCB search. In SMVLite, we reduce the search space for Type 1 frames by exploiting the periodic nature of the FCB in these frames. One of the sub-codebooks allowed in Type 1 sub-frames has only 2-pulses. To search through this sub-codebook, SMV conducts a pre-search that selects the best 16 or 19 positions prior to performing full-search in this reduced pulse space. During this search procedure, the pitch information can be used effectively to reduce the search space and hence the computational complexity[1]. Since Type 1 frames are characterized by high pitch gain, each FCB pulse candidate is virtually repeated through the rest of the sub-frame as a result of pitch enhancement. We have observed through experiments that it is redundant to search through many positions in the codebook since those positions have been already covered through pitch-enhanced replicas of other pulses. We have applied this approach to significantly reduce the computational complexity of FCB search in Type 1 frames for the 2-pulse sub-codebook.

The three improvements to FCB computational complexity described in this paper result in a net improvement of approximately 42% in the FCB search complexity (from 11.32 to 6.60 WMOPS). Combining all the techniques in SMVLite (Signal modification, LSF quantization & FCB search), we have obtained a total WMOPS of 27.1 which is 25% better than SMV (36). The improvements in computational complexity are summarized in Table 2. We have also implemented both SMV and SMVLite algorithms in fixed-point on the TMS320C55x digital signal processor. The improvements in WMOPS shown in table 2 are also seen to reflect in a 25% improvement in the case of actual TI c55x implementations.

Module	WMOPS	
	SMV	SMVLite
Signal Modification	6.11	3.26
LSF Quantization	3.26	2.40
Fixed Codebook Search	11.32	6.60
Total WMOPS	36	27.1
Total TMS320C55x	**30.6**	**23.0**

TABLE 2: COMPLEXITY BREAK-DOWN OF SMV

4. RESULTS OF SUBJECTIVE LISTENING TESTS

The subjective quality of SMVLite was measured through formal Mean Opinion Score (MOS) tests which rate the codec on a five-point scale. The MOS tests were conducted in the Dynastat Listening Laboratories in accordance with ITU-T recommended subjective test procedures [7] for evaluating speech quality. The tests were conducted also within the framework of SMV MPS (Minimum Performance specifications) as defined by the 3GPP2 [6]. The MPS specifies three independent experiments to evaluate SMVLite (test codec) against the standard floating-point SMV reference (master codec). Experiment 1 is the comparison for clean speech (no background noise). Experiment 2 compares master and test codecs when there are frame erasures while Experiment 3 covers performance in noisy background. In each experiment, 64 listeners rated speech material for 8 talkers in each of 48 conditions. These include all possible combinations of Master/Test Encoder and Master/Test Decoder [6].

The results of the MOS test conducted are summarized in table 3. These include results of experiments 1-3 in all SMV modes. For the purpose of brevity, in each experiment, we have averaged the MOS score over multiple conditions corresponding to a particular SMV mode. Statistical cross-checks by Dynastat conclusively prove that SMVLite is perceptually similar or better than SMV in all test conditions and all modes in all experiments.

	Mode 0	Mode 1	Mode 2
Experiment 1 (Clean Speech)			
SMV	4.012	3.889	3.726
SMVLite	4.005	3.879	3.727
Experiment 2 (Frame Erasure)			
SMV	3.615	3.523	3.411
SMVLite	3.654	3.538	3.381
Experiment 3 (Noisy Speech)			
SMV	3.572	3.561	3.481
SMVLite	3.646	3.565	3.498

TABLE 3: COMPARISON OF MOS SCORES OF SMV AND SMVLITE

5. CONCLUSIONS

This paper describes the SMVLite codec whose complexity is 25% less than the SMV standard. SMVLite is fully compatible with SMV. Subjective tests compliant with 3GPP2 Minimum performance specifications conclusively prove that SMVLite is perceptually equivalent or better than SMV in all conditions & modes.

REFERENCES

[1] Selectable Mode Vocoder (SMV) Service Option for Wideband Spread Spectrum Communications Systems, 3GPP2-C.S0030-0.

[2] Yang Gao et al., "The SMV Algorithm Selected by TIA and 3GPP2 for CDMA Applications, Proc ICASSP-2001, pp. 709-712.

[3] Enhanced Variable Rate Codec (EVRC), 3GPP2 C.S0014-0.

[4] Yang Gao et al., "Ex-Celp: A Speech Coding Paradigm", Proc ICASSP-2001, pp. 689-692

[5] W.B. Kleijn, R.P. Ramanchandran and P. Kroon, "Generalized Analysis-by-Synthesis Coding and its Application to Pitch Prediction" Proc ICSSAP, 1992, pp. 1337-1340.

[6] Recommended Minimum Performance Standard for Selectable Mode Vocoder, Service Option 56. 3GPP2-C.S0034-0.

[7] ITU-T Rec. P.800: Methods for Subjective Determination of Transmission Quality (08/96).

[8] Dynastat TI India SMV MPS Report, August 2004.

[9] TI Press Release Texas Instruments Enhances its VoIP Gateway, 8th March 2005.