STUDIA PHONOLOGICA VI (1972)

On-Line, Real-Time Multiple Speech Output System and Its System Evaluation

Toshiyuki SAKAI, Kenji OHTANI and Shinji TOMITA

A new multiple speech output system is described. This system is based on a compilation method. Fundamental speech elements used for the synthesis are the speech waves within one cycle at larynx frequency. They are stored on a secondary memory such as a magnetic drum or disk in the form of digitalized zero-crossing intervals. Speech elements are read out from the secondary memory and are connected in succession according to the connection rules by the computer program. The resulting sequences are transformed into the speech sounds by synthesizers. System evaluation is also performed in real time mode to measure the load of the computer and the maximum number of multiplicity.

1. Introduction

Among various approaches to speech synthesis, the voice-recording-reproducing-type synthesizers[1] have been in practical use, such as in the phone number inquiry service. Speech production of this type utilizes spoken words stored in a large secondary memory of a computer. When a punched text is given, synthesis of speech is to take out the corresponding words from a synthetic library, connect them properly and sent them to the terminal acoustic devices. This system, in principle, can reproduce synthetic speech of very high quality. However, its capability is naturally limited because the large storage capacity is required for spoken word data, and the device for the generation of connected speech becomes complicated as the variety of messages to be synthesized is increased.

The new system we have developed is a kind of speech synthesizer by rule, assisted by a small synthetic vocabulary. It can generate speech sound from any Japanese text punched in KANA letters. The system is designed to provide several different speech outputs to different terminals at the same time. The minimal acoustic unit for the compilation by rule is a sequence of zero-crossing intervals for one cycle at larynx frequency. By adopting this short speech element as a fundamental compilation unit, it becomes possible to connect them so that formant loci are continuous in the transitional parts and the variety of

Toshiyuki SAKAI (坂井利之): Professor, Department of Information Science, Kyoto University.
Kenji OHTANI (大谷謙治): Assistant, Department of Information Science, Kyoto University.
Shinji TOMITA (富田真治): PhD Student, Department of Electrical Engineering, Kyoto University.

messages to be synthesized is unlimited. The zero-crossing intervals of the compilation unit is quantized, and the sequence of these quantized data is stored in the synthetic vocabulary. This saves the memory capacity enormously, and makes it easy to connect speech elements.

The computer used in this system is the NEAC-2200 model 200 (character machine, same as the Honeywell 200 series: core memory 32 k characters).

2. General Description of the System

Some minimal compilation units were selected from the utterances of a male speaker and the other units were produced by simulating a terminal analog synthesizer by the computer. The former units are such parts of speech sound as one cycle at larynx frequency of voiced sounds, a burst interval of stop consonants, fricative noise and so on, which we henceforth will call "wave element". In synthesizing the stationary parts of speech sound wave, for instance, the retention of the vowels, a wave element is repeatedly connected several times. The intensity and the duration of the wave is controlled by two parameters; the amplitude "a" and the number of repetition "r". We call the wave element and the two associated parameters "segment".

At the speech synthesis stage, syllables corresponding to (V-C) context (V: vowel, C: consonant), (V-V) context, or (C-V) context are used as the actual compilation units in order to save the necessary compilation time. These are constructed with the segment sequences in advance, and are stored in the secondary memory. According to the input sentences, the segment sequences of these di-grams are read into core memory in succession, and the arranged sequences are transmitted to the terminal equipments with every segment, and transformed into the speech sound.

The number of wave elements prepared is about 550, for which 70 k bits are necessary. On the other hand, the total size of the memory of the actual compilation units is about 170 k bits.

3. Synthesis of the Speech Sound

It is well known that the voiced sounds like vowels, nasals, and voiced plosives have in their waveforms a sequence of quasi-periodic patterns repeated at the frequency of vocal chord vibration. The sequence of their zero-crossing intervals, therefore, is in good correspondence with the above-mentioned repetition of voiced sounds. This means that those voiced sounds can be expressed by a typical sequence of zero-crossing widths (hereafter abreviated as OXW) for only one pitch period (wave element P) and two parameters; the number (r) of repetitions necessary to reproduce the adequate length of the voiced sounds and the wave intensity (a). Therefore, a segment is denoted as "aP^r". By this principle, each voiced sound is converted into a very simple form, particularly suited for storage and

Table 1. Expression of phonemes by segment sequence

phoneme	expression	phoneme	expression
/a/	$17P_a^8$	/d/	$3P_d^5 \cdot 3P_t^1$
/e/	$13P_e^8$	/g/	$3P_g^5 \cdot 3P_k^1$
/m/	$3P_m^7$	/s/	$1P_s^{10}*$
/n/	$3P_n^7$	/z/	$2P_{z1}^4 \cdot 2P_{z2}^2 \cdot 2P_s^4*$
/p/	$0P_{sp}^9 \cdot 2P_p^1$	/ch/	$0P_{sp}^7 \cdot 2P_{ch}^6*$
/t/	$0P_{sp}^9 \cdot 2P_t^1$	/ts/	$0P_{sp}^7 \cdot 2P_{ts}^6*$
/k/	$0P_{sp}^9 \cdot 2P_k^1$	/h/	$1P_h^2*$
/b/	$3P_b^5 \cdot 3P_p^1$	/r/	$2P_r^5$

processing by digital data processor. Thus in this speech synthesis method, a phoneme is represented by a sequence of the segments. Several phonemes expressed by segment sequences are shown in Table 1.

The duration of the retention of a vowel, which is about 70 msec in the normal context, can be changed by controlling the value of the repetition parameter "r" of the wave element. Nasal sounds /m/ and /n/ are also produced by the repetition of wave elements P_m and P_n respectively. The unvoiced stop consonants /p/, /t/ and /k/ consist of the wave elements P_p, P_s and P_k preceded by a silent interval respectively. Voiced stops /b/, /d/ and /g/ are constructed with five repetitions of buzz elements P_b, P_d and P_g, followed by burst parts P_p, P_t and P_g, respectively, which are identical with the wave elements used in unvoiced stops.

In the expressions of /s/, /z/ and so on, which have relatively long-continued noise periods, the number of the repetition times is accompanied by a star (*) as is shown in the case of /s/ in Table 1. The expression 10* of $1P_s^{10}*$ does not mean that the elementary sound P_s is simply repeated 10 times, but instead, that the sequence of OXW of P_s is connected 10 times, being rearranged at random at every connection. Wave element P_{ch} and P_{ts} in affricative sounds /ch/ and /ts/ are also processed in the same manner. The reason for this is that those sounds must not have the periodicity.

All of the wave elements listed in Table 1 are selected from Japanese mono-syllabic sounds uttered by a male. A speech sound wave is pre-emphasized with an RC-high pass filter whose cut-off frequency is 1.6 kHz, and is sampled at 20 kHz rate. The amplitude is quantized into 10 bits plus a sign bit with an A-D converter connected to the computer and is subjected to infinite peak clipping by computer programming. The OXW patterns within one cycle at larynx frequency of the vowels are selected by hand from the retentive parts, and are also selected from the OXW patterns of the mono-syllabic sounds. Only 32 of 550 wave elements, are obtained in this way. Others are produced by the computer as followings.

When two different phonemes are uttered continuously, articulatory organs

move necessarily under some constraints. In the glide part corresponding to this transitional movement, the appropriate segment sequence must be interpolated so that the formant loci are continuous.

Since it is difficult to select the transitional wave elements from natural speech for all of the contexts, some of the wave elements were produced by the computer simulation of the terminal analog speech synthesis model, which consists of the serial concatenation of 4 single tuned circuits (S.T.C.). The center frequencies f_1 and f_2 of the lower two S.T.C. correspond to the frequencies of the first and second formants. Here the center frequencies f_1 is varied by 50 Hz from 300 Hz to 950 Hz and f_2 from 700 Hz to 2500 Hz, the center frequencies of the third and forth S.T.C. being fixed at constant values. For each pair of f_1 and f_2 parameters, the simulation program is executed and the sound wave produced for one steady larynx cycle is transformed into a zero-crossing wave, and is registered as the wave element data.

These wave elements are inserted in the glide part. For example, the VCV context of "oda" is transformed into a segment sequence

$$10P_0^{8}\cdot 9P^{1}_{(550,1100)}\cdot 8P^{1}_{(500,1200)}\cdot 6P^{1}_{(450,1300)}\cdot 3P_d^{5}\cdot 3P_t^{1}\cdot 9P^{1}_{(400,1450)}\cdot$$
$$12P^{1}_{(600,1350)}\cdot 15P^{1}_{(750,1300)}\cdot 17P_a^{8}$$

where $P_{(400,1150)}$ means a computed one-pitch wave whose first and second formants are 400 Hz and 1450 Hz respectively. Here the values 400 and 1450 and so on are computed by the linear interpolation between the formants of vowel /a/ and offset formant frequencies of /d/. The offset and onset formant frequencies used for this interpolation are based on the measured formant loci of the natural speech.

In Fig. 1, a part of the synthetic speech obtained from the above segment sequence is illustrated shematically.

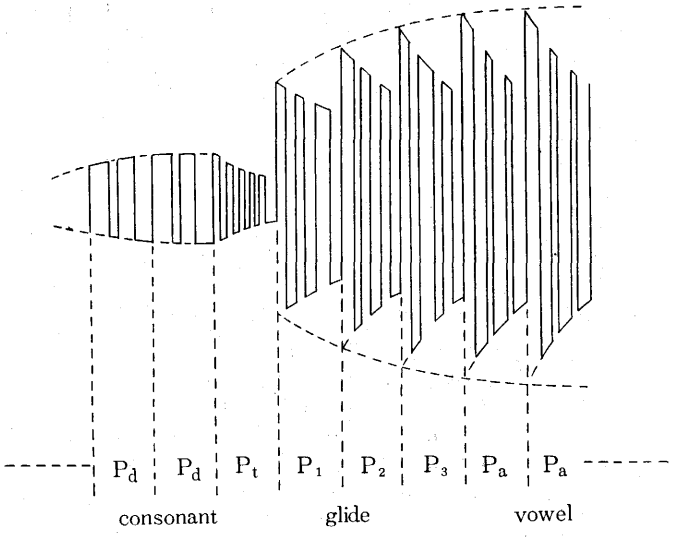

Fig. 1. Synthesized speech sound (da).

Here, (V-C) di-gram $9P^1_{(550,1100)}$... $6P^1_{(450,1300)}$ and (C-V) di-gram $3P_d^5$... $15P^1_{(750,1250)}$ are the actual compilation units used in the speech-synthesis stage.

4. ON-LINE, REAL-TIME, MULTIPLE SPEECH OUTPUT SYSTEM

In this section an on-line, real-time, multiple speech output system and its system evaluation are described. This system, accepting arbitrary Japanese sentences dispatched from many users, transforms them into speech sounds and feeds them back simultaneously. This can be implemented economically.

4.1 System Configuration and Speech-Synthesis Method

The configuration of the system is shown in Fig. 2. The dictionary of the actual compilation units stored on the secondary memory consists of about 180 Japanese di-gram (V-C), (V-V) and (C-V), which have been prepared by the compilation method described in section 3.

The operations of the synthesis program are;

(1) divide messages into a consecutive di-gram sequence;

(2) find the segment sequence corresponding to a di-gram sequence on the secondary memory;

(3) transfer the segment sequence to one of the output double-buffers A_i (B_i) allocated to user (i).

The segment sequences, for example,

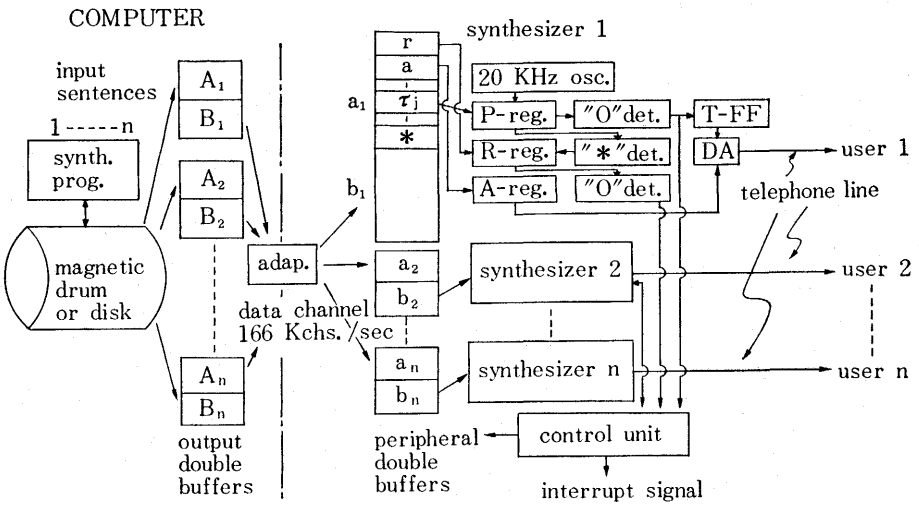

Fig. 2. System configuration.
synth. prog. : synthesis program
adap. : peripheral adaptor
"*" det. : end marker detector
"0" det. : zero dector
reg. : register
T-FF : T-flip flop
20KHz osc. : 20 KHz oscillator

$$10P_0^{8}\cdot 9P^{1}_{(550,1100)}\cdot 8P^{1}_{(500,1200)}\cdot 6P^{1}_{(450,1300)} \text{ and}$$
$$3P_d^{5}\cdot 3P_t^{1}\cdot 9P^{1}_{(400,1450)}\cdot 12P^{1}_{(600,1350)}\cdot 15P^{1}_{(750,1250)}.$$

which correspond to the di-gram (o-d) and (d-a) are stored in the area A_1 and B_1 respectively. Each segment in $A_1(B_1)$ is transferred to one of the peripheral double buffers $a_1(b_1)$ (64×2 characters/user) through the data channel (166 k characters/sec) in response to the interrupt signal from the control unit of the perpheral equipment.

Under the multiple control of the control unit, one digit of OXW "τ_j" stored in $a_1(b_1)$ is set in P-register of user (1) and is reduced by one at every clock pulse of 20 kHz (sampling rate) oscilator. When the content of the P-register becomes zero, the state of the T-flip flop is reversed, and at the same time the next digit "τ_{j+1}" is read into the P-register from $a_1(b_1)$. The content of the R-register is reduced by one at the time that the end marker "*" of the segment stored in $a_1(b_1)$ is found. If it is not zero, the OXW sequence in $a_1(b_1)$ is transmitted again to the P-register from its beginning. During this operation, the content "a" of A-register gives the amplitude of the zero-crossing waves. If the content of R-register is zero, the OXW sequence, "r" parameter, and "a" parameter in $b_1(a_1)$ are read into P-, R-, A-registers respectively and the interrupt signal is transmitted to the computer requesting that the next segment be transferred into $a_1(b)_1$.

Repeating the same operation many times, output speech sounds such as Fig. 1 are obtained at the synthesizer, and are transmitted to the user (1) via telephone lines. The envelope of the wave is modified by several kinds of filters in "DA" block. Speech sounds are also produced in the same way for the other users. The processing speed of the synthesis program is so high that any user can obtain the required audio response with no delay.

4.2 Evaluation of the System

In the experimental system, two synthesizers which are connected to the computer, are constructed in hardware to produce speech sounds requested to be simultaneously output. When many synthesizers are supposed to be attached, traffic congestion often sets up a long queue because they may request at a time next compilation units to be transferred which are stored on the secondary memory. Now we measure the load of the computer and maximum number of simultaneous output for evaluation of the system already made, and search better queue disciplines by which the traffic congestion must be reduced as much as possible. For this investigation, queuing theory is not so available as simulation techniques. The system simulation is performed in on-line, real-time mode. Of many synthesizers of the system, two are the actual ones constructed in hardware and others are simulated by the program and by the simple devices. For each number of multiplicity (number of simultaneous outputs), simulation is excuted for 5 minutes to measure the ratio of the total CPU time spent on the speech synthesis to the total effective time of computer operation (CPU occupancy rate) and the

total time per sec during which the excessive demmands of the secondary memory access cause discontinuance in speech sounds (Queuing error rate).

Both program of multiple speech output and the background are supposed to be running. CPU occupancy rate can be computed by reducing from the total effective operation time of the computer, the total time during which the computer is running in the program of the background. In the fololwing, two kinds of queue disciplines are discussed. One is based on first in first out service discipline (case 1), and the other on moving server queuing model (case 2).

Case 1

Queuing model for the system described in section 4.1 is shown in Fig. 3 a. In this system, requests for next compilation unit (segment sequence) to be transferred, are sent to the computer when the segment sequence in one of the double buffers is consumed. Because synthesizers act independently of each other, traffic contention occasionally sets up a long queue which causes queuing errors. Queuing errors are counted when the double buffers have been completely consumed but the next segment sequence has not yet been prepared. Normally, there exists only a small size queue that may be processed in so short a time as not to cause queuing errors. A main queue in Fig. 3 a corresponds to the queue for requests for next compilation units, and its service time probability is almost similar to the probability of the access time of the secondary memory. Size of the user queue is limited to two when double buffers are used to store segment sequences. And service time probability of the user queue is similar to that of the actual dura-

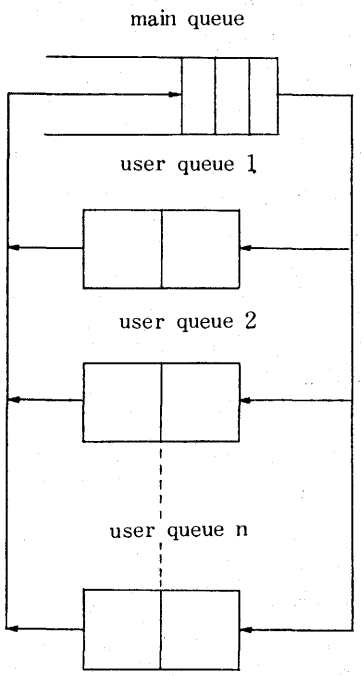

Fig. 3a. Queuing model for case 1

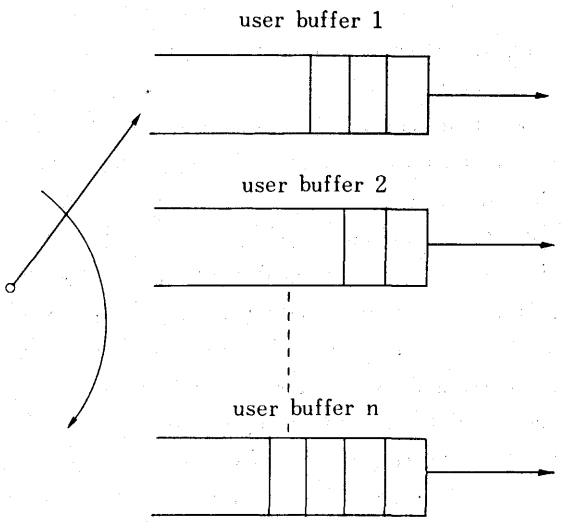

Fig. 3b. Queuing model for case 2

tion time of the segment sequences. Input process to the main queue consists of the pooled output process of the user queues. It is very difficult to solve such a model precisely by using queuing theory, but some approximations make it possible to solve easily as follows.

$$\text{Queuing error rate} = \frac{1}{A_m}(n-1)!(n\rho)^{(m-1)n}\sum_{i=1}^{n}\frac{i\rho^i}{(n-i)!}$$

$$\text{CPU occupancy rate} = 1-\frac{1}{A_m}$$

$$Am = 1+\sum_{i=1}^{(m-1)n}(n\rho)^i + n!(n\rho)^{(m-1)n}\cdot\sum_{i=1}^{n}\frac{\rho^i}{(n-i)!}$$

Here n is a multiplicity number, ρ is a ratio of mean service rate of the user queue to that of the main queue, and m is a limited number of the user queue size (in the case where double buffers are used, m equals two.) Results of calculation of the above equations are shown in Fig. 4 as the function of the multiplicity numbers with parameter ρ. It shows qualitatively that increase in m reduces queuing errors by a certain amount. Fig. 5 shows the results of simulation, when the magnetic drum is used as the secondary memory. It is found that; (1) CPU occupancy rate increases almost linearly at the rate of 5.5% per multiplicity number. (2) The queuing error rate is zero up to the 9 users. For simultaneous outputs of 10 users, it becomes to about 0.4% and increases rapidly for more than 10 users. (3) The maximum available number of simultaneous output is about 10. It mainly depend on the mean access time the magnetic drum (10 msec). So we must investigate the proper scheduling technique on it.

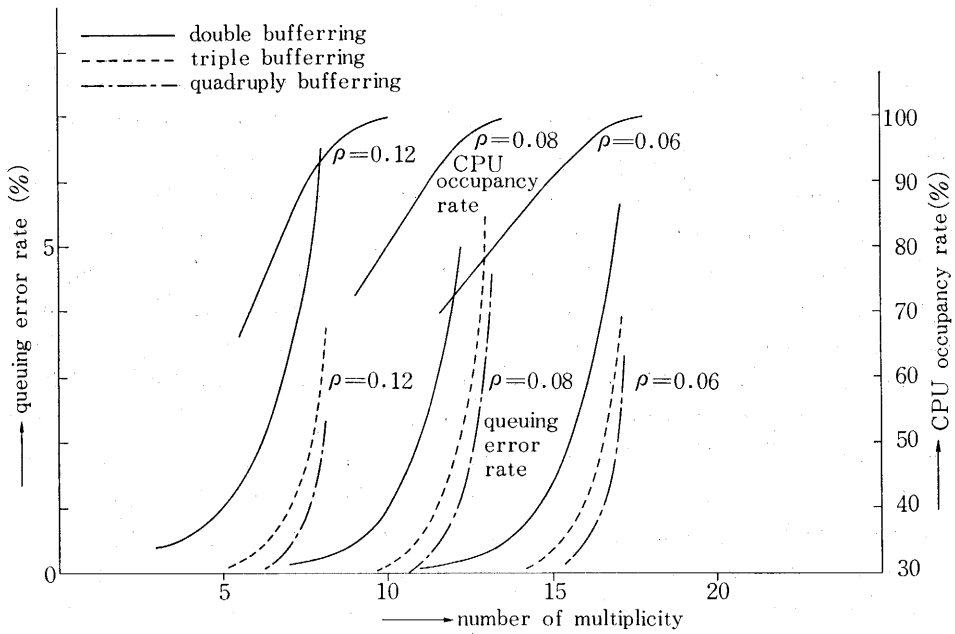

Fig. 4. Results of calculation for case 1

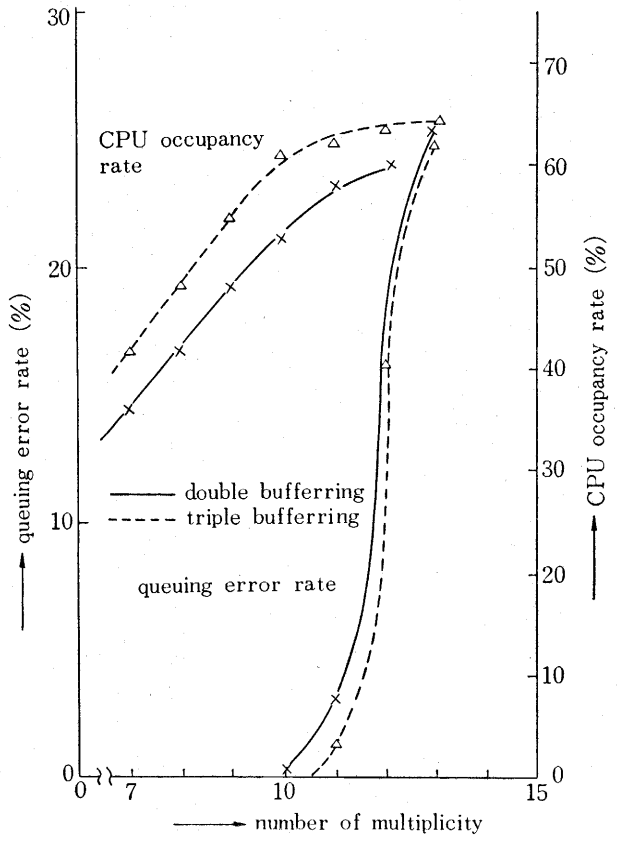

Fig. 5. Results of simulation for case 1

Case 2

In order to avoid queuing errors which cause discontinuance in output speech sounds, it is better to control the queuing process so that input intervals of the requests are scheduled regularly not to be excessively small in a period. Now, case 2 is considered where the computer can serve to transfer one segment sequence at a time to each user cyclically until each user buffer becomes full. When buffers of all users become full, the computer operates on the background job and waits till one of the user buffers can be available for data transfer. This queuing model is shown in Fig. 3 b. Queuing errors are counted every 10 msec during which data in the user buffer are fully consumed. Fig. 6 shows the results of simulation where the magnetic disk (access time, 20 msec) is used as the secondary memory and user buffer size is in the case of 1 Kchs and 2 Kchs. It is found that; (1) When the user buffer size is 1 Kchs, maximum number of multiplicity is about 6. In comparison with the case 1, the case 2 is superior in system multiplicity when both cases use buffers of the same size. (2) But, the case 2 requires more CPU time, and CPU occupancy rate increases non-linearly. (3) Where the user buffer size is 2 Kchs, the maximum number of multiplicity is about 7, and

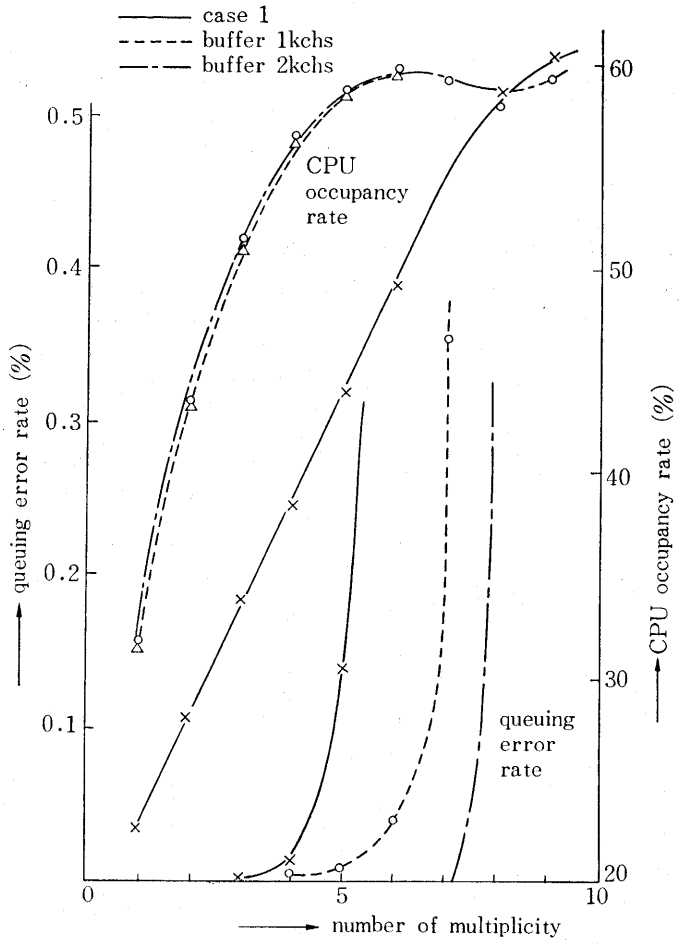

Fig. 6. Results of simulation for case 2

queuing error is completely zero up to 7 users. It rapidly increases for more than 7 multiplicity.

5. Conclusion

This new speech output system produces multi-channel outputs (10 or more) of synthetic speech sounds in real-time. The score of the intelligibility test for the synthesized speech sounds in this system, measured for 67 Japanese monosyllabic sounds, is about 75%. The followings are advantages of this system: (1) It can synthesize any Japanese sentences; (2) In comparison with other compilation methods,[1),2),3)] smaller memory capacity is sufficient to store all of the compilation units; (3) The synthesis program is simple; and (4) The speech synthesizer is very simple and small, compared to other well-known speech synthesizers[4)], and can be attached to most computer systems.

REFERENCES

1. L. Lee and R. Mulvany, Now a Talking Computer Answers Inventory Inquiries, Electronics (August 16, 1963) p. 30.
2. G.E. Peterson and W.S.Y. Wang, Segmentation Techniques in Speech Synthesis, JASA vol. 30 No. 8 (1958) p. 739.
3. G.L. Francini and G.B. Debiasi, Study of a System of Minimal Speech-reproducing Units for Italian Speech, JASA vol. 43 No. 6 (1968) p. 1282.
4. J.C.W.A. Liljencrants, The Ove III Speech Synthesizer, IEEE vol. AU-16 No. 1 (1968) p. 137.

(Aug. 31, 1972, received)