



הטכניון – מכון טכנולוגי לישראל
Technion – Israel Institute of Technology

ספריות הטכניון
The Technion Libraries

בית הספר ללימודי מוסמכים ע"ש ארווין וג'ואן ג'ייקובס
Irwin and Joan Jacobs Graduate School

©

All rights reserved

*This work, in whole or in part, may not be copied (in any media), printed, translated, stored in a retrieval system, transmitted via the internet or other electronic means, except for "fair use" of brief quotations for academic instruction, criticism, or research purposes only.
Commercial use of this material is completely prohibited.*

©

כל הזכויות שמורות

אין להעתיק (במדיה כלשהי), להדפיס, לתרגם, לאחסן במאגר מידע, להפיץ באינטרנט, חיבור זה או כל חלק ממנו, למעט "שימוש הוגן" בקטעים קצרים מן החיבור למטרות לימוד, הוראה, ביקורת או מחקר. שימוש מסחרי בחומר הכלול בחיבור זה אסור בהחלט.

טכניון
פקולטה להנדסת חשמל
פקולטה למדעי מחשב
ספרייה מס' 959

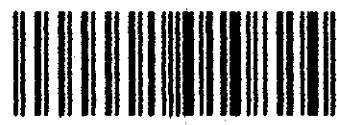
בחינת גישות לשיפור הביצועים של מקודדי
אות השארית מחיזוי לינארי של אותות דיבור

חבור על מחקר
לשם מילוי חלקי של הדרישות לקבלת התואר
מגיסטר למדעים
בהנדסת חשמל

מאת
אלפי יצחק

959

2059431



000001010824

10.1.89

הוגש לסנט הטכניון - מכון טכנולוגי לישראל
סיוון - תשמ"ח ח"פ 1988 - יוני

המחקר נעשה בהנחיית פרופסור דוד מלאך
במעבדה לעיבוד אותות בפקולטה להנדסת חשמל

תודתי העמוקה נתונה לפרופסור דוד מלאך על
הנחייתו המועילה והמסורה בכל שלבי המחקר.

כמו כן ברצוני להודות לגב' זיוה אבני, גב' אורית
מידזינסקי, מר יורם אור-חן, מר נמרוז פלג, מר משה
נמר, מר דן טל ולכל סגל המעבדות לעבוד אותות
ולתקשורת שעזרו להשלים עבודה זו.

תוכן ענינים

עמוד

1	תקציר
3	רשימת סמלים וקיצורים
5	פרק 1 : מבוא
		פרק 2 : קידוד אותות דיבור
7	2.1 מבוא
7	2.2 מקודדי מקור - Vcoders
8	2.3 קביעת פרמטרי ה-LPC
9	2.4 זיהוי Pitch
9	2.5 סיכום - מקודדי מקור
10	2.6 מקודדי צורת גל
10	2.7 אות השארית מחיזוי לינארי
		פרק 3 : מקודדי שארית
11	3.1 מבוא
12	3.2 Adaptive Predictive Coder - APC
13	3.3 APC עם עיצוב ספקטרום שגיאת הקונטיזציה
15	3.4 Multi-Pulse Excitation - MPE
15	3.4.1 מודל ה-Multi-Pulse לעירור
18	3.4.2 אלגוריתם ה-MPE
20	3.4.3 שיפורים לאלגוריתם הבסיסי
22	3.4.4 תוצאות סימולציות

תוכן ענינים (המשך)

23 Code Excited Linear Prediction - CELP 3.5
23 3.5.1 בחירת סדרת עירור אופטימלית
25 3.5.2 סיבוכיות אלגוריתם החיפוש
26 3.5.3 תוצאות סימולציות

פרק 4 : יישום רשת נוירונים לפתרון בעית ה-Multi-Pulse

27 4.1 מבוא
4.2 יישום רשת נוירונים לפתרון בעיות
27 אופטימיזציה
4.3 פתרון בעית עירור ה-Multi-Pulse בעזרת רשת
30 נוירונים
32 4.4 הסבר לדינמיקה של הרשת
33 4.5 תוצאות סימולציות

פרק 5 : עדכון תנאי התחלה למסנן הסינטזה

34 5.1 מבוא
35 5.2 פיתוח סט המשוואות לתנאי התחלה אופטימליים
5.2.1 קבלת תגובת הזכרון כפונקציה של דגמי
36 הזכרון
5.3 שילוב סכמת עדכון הזכרון בסכמת עירור
37 ה-Multi-Pulse
5.3.1 ה-אופטימיזציה לתנאי ההתחלה
41 ולאמפליטודות

תוכן ענינים (המשך)

פרק 6 : קוונטיזציה וקטורית לוקטור תנאי-ההתחלה

43	6.1 מבוא
43	6.2 קוונטיזציה וקטורית
44	6.2.1 הגדרת קוונטיזציה וקטורית (VQ)
46	6.2.2 קוונטיזציה וקטורית עם בקרת הגבר
47	6.3 תכנון מילון אופטימלי לוקטור תנאי-התחלה
49	6.4 קידוד וקטור של תנאי ההתחלה
	6.5 קידוד P דגמים ראשונים בתגובת הזכרון
53	האופטימלית המשוקללת
54	6.6 קידוד וקטור הכניסה השקול \underline{x}

פרק 7 : שילוב ספר קוד ופולסים לקידוד אות השארית

57	7.1 מבוא
58	7.2 שילוב ספר קוד אחרי קביעת הפולסים
	7.3 שילוב קוד ופולסים לעירור מסנן סינטזה עם
63	Pitch Predictor

פרק 8 : סיכום ומסקנות

67	נספח א : הקצאת סיביות במקודדים שהוצגו
69	נספח ב : חישובי סיבוכיות
78	רשימת מקורות :

תקציר

בעבודה זו אנו בוחנים מספר גישות לשיפור הביצועים של מקודדי אות שארית מחיזוי לינארי בתחום קצבי הבינים 9.6-16 Kbps -

טכניקות הקידוד המקובלות כיום דורשות קצבי שידור בתחום 2.4-64 Kbps. בראש התחום ניצבים מקודדי צורת גל בהם האיכות גבוהה והסיבוכיות נמוכה. חסרונם הבולט הוא הקצב הגבוה יחסית הנדרש להשגת איכות דיבור גבוהה: 32-64 Kbps והנפילה המהירה באיכות כשיררדים בקצב.

בקצה השני של תחום הקצבים נמצאים ה-Vocoders בהם מתאימים מודל לאות הדיבור (מודל חיזוי לינארי - LPC) ומשדרים את פרמטרי המודל. במקודדים אלו איכות הדיבור סנטטית, הם אינם מתפקדים בתנאי רעש רקע או ריבוי דוברים והגדלת קצב השידור מעל לקצב אופייני של 2.4 Kbps אינה משפרת משמעותית את איכות האות המשוחזר.

כדי להשיג איכות טובה בתחום הקצבים 4.8-16 Kbps פותחה משפחת מקודדי אות שארית (Residual Coders). במקודדים אלו משדרים מידע על אות השארית מהחיזוי הלינארי בנוסף למידע על מקדמי מסנן החיזוי.

בראש משפחה זו הן מבחינת קצב והן מבחינת איכות נמצא מקודד ה-APC (Adaptive Predictive Coder) בו מקודד כל דגם באות השארית בנפרד. כדי לרדת לקצב של 9.6 Kbps ומטה לא ניתן להישמש בשיטה זו ונאלצים להפעיל טכניקות אחרות בהם מקודד אות השארית.

גישה אחת היא ה-MPE (Multi-Pulse Excitation) שבה אות השארית מיוצג על ידי מספר קטן של דגמים חשובים. כיוון שמציאת המיקומים והאמפליטודות של הדגמים הטובים ביותר היא בעיה מסובכת מאוד, הוצע אלגוריתם איטרטיבי (תת-אופטימלי) אשר קובע בכל איטרציה פולס נוסף. בדרך כלל מסתפקים בכ 8 פולסים למסגרת של 5-8 msec להשגת איכות טובה (קצבי שידור המתאימים הם בתחום 9.6-16 Kbps).

גישה נוספת היא ה-CELP (Code Excited Linear Prediction), במקודד זה העירור האופטימלי נבחר מתוך מילון של סדרות גאוסיות וחסרות קורלציה (לבנות). החיסרון העיקרי של המקודד הוא הסיבוכיות הגבוהה הכרוכה בתהליך חיפוש הסדרה האופטימלית.

בעבודה זו מועלים ונבדקים מספר רעיונות לשיפור ביצועי מקודד ה-MPE:

ניסינו לפתור את הבעיה של מציאת הפולסים והאמפליטודות האופטימליים בעזרת רשת נוירונים - Neural Network (במקום האלגוריתם התת-אופטימלי) בתקווה להשיג איכות טובה יותר עם אותו מספר פולסים או לחילופין להקטין את קצב השידור עבור איכות נתונה. ניסיון זה לא עלה יפה: הפתרון שהתקבל מהרשת היה בדרך כלל גרוע מהפיתרון שהתקבל בהפעלת האלגוריתם התת-אופטימלי.

בדקנו כיצד לנצל את העובדה שתגובת מסנן הסינטזה מורכבת מתגובה לתנאי התחלה ומהתגובה לעירור. הרעיון הוא למצא את תנאי ההתחלה האופטימליים בנוסף לעירור כך שסך התגובות יתן אות משוחזר באיכות טובה יותר. בעבודה אנו מציגים מספר דרכים לשילוב של עדכון תנאי התחלה באלגוריתם ה-MPE ואת התוצאות שהתקבלו בכל אחת מהדרכים. התוצאות הצביעו על שיפור ניכר באיכות בהשוואה לאלגוריתם המקורי. (לפני קדוד תנאי ההתחלה).

טכניקת הקידוד שננקטה לתנאי ההתחלה האופטימליים היתה וקטורית. מציאת הקוונטיזציה האופטימלי היא בעיה מורכבת שלא נבדקה. תחת זאת נבדקו מספר קוונטיזציות אחרים אשר בגבול (עם מילונים אינסופיים) מתלכדים עם הקוונטיזציה האופטימלי. התוצאות שהתקבלו לא הצביעו על כדאיות המערכת (בהשוואה לקידוד פולסים נוספים).

נסיונות אלו הובילו למודל חדש לאות העירור. לפי מודל זה אשר מהווה הכלאה בין שיטות ה-MPE וה-CELP, משלבים פולסים וקוד לעירור מסנן הסינטזה. נתאר מספר אלגוריתמים המבוססים על מודל זה, ולסיכום העבודה נציג מקודד חדש - SPACE (Switched Pulses) And Code Excitation אשר עולה בביצועיו על מקודד ה-MPE ב-9.6 Kbps ושסיבוכיותו קטנה מזה של מקודד ה-CELP, באותו הקצב.

רשימת סמלים

- פונקציית התמסורת של מסנן הסינטזה ב-LPC.	$H(Z)$
- הגבר אות העירור למסנן הסינטזה ב-LPC.	G
- פונקציית התמסורת של המסנן ההפוך.	$A(Z)$
- דגם אות דיבור מקורי.	$s(n)$
- דגם אות דיבור משוחזר.	$\hat{s}(n)$
- מקדם של מסנן החיזוי.	a_i
- דגם של אות עירור.	$u(n)$
- סדר מסנן ה-LPC.	P
- מספר הפולסים במסגרת זמן אנליזה באלגוריתם MPE.	K
- דגם של אות השארית.	$e(n)$
- אנרגיית שגיאה.	E
- דגם אוטוקורלציה.	ϕ
- דגם קרוסקורלציה.	α
- התמרת Z של לאות השארית.	$E(Z)$
- התמרת Z של אות הדיבור.	$S(Z)$
- וקטור זכרון של מסנן סינטזה.	\underline{M}
- פונקציית תמסורת של המסנן המשקלל.	$W(Z)$
- התגובה להלם של מסנן הסינטזה.	$h(n)$
- התגובה להלם של מסנן הסינטזה המשוקלל.	$h_w(n)$
- התגובה של מסנן הסינטזה לוקטור הזכרון \underline{M} .	$l(n)$
- אות הדיבור המקורי לאחר הפחתת $l(n)$.	$s'(n)$
- אמפליטודה של פולס עירור.	b_i
- מיקום של פולס במסגרת זמן.	m_i
- וקטור זכרון של מסנן הסינטזה שמעודכן בכל מסגרת.	\underline{M}
- התגובה של מסנן הסינטזה לוקטור הזכרון \underline{M} .	$l_m(n)$
- התגובה $l_m(n)$ אחרי המעבר דרך המסנן המשקלל.	$l_w(n)$
- אות כניסה למסנן הסינטזה.	$x(n)$
- גודל מסגרת האנליזה.	N
- מספר הוקטורים במילון.	L
- גורם הגבר.	g_i
- קוונטיזציה.	$Q\{\cdot\}$
- קצב ממוצע (סיביות לשניה).	R
- מספר פעולות כפל לחישוב דגם באות המוצא.	T
- עיזות ממוצע.	D
- זרם כניסה לנוירון.	I_i
- המתח בכניסת הנוירון.	U_i
- המתח ביציאת הנוירון.	X_i
- פונקציית תמסורת של נוירון.	$S_i(t)$
- אופרטור תוחלת.	$E\{\cdot\}$
- קבוצת ההחלטה ה- i -ית של קוונטיזציה וקטורי.	S_i
- שונות.	σ

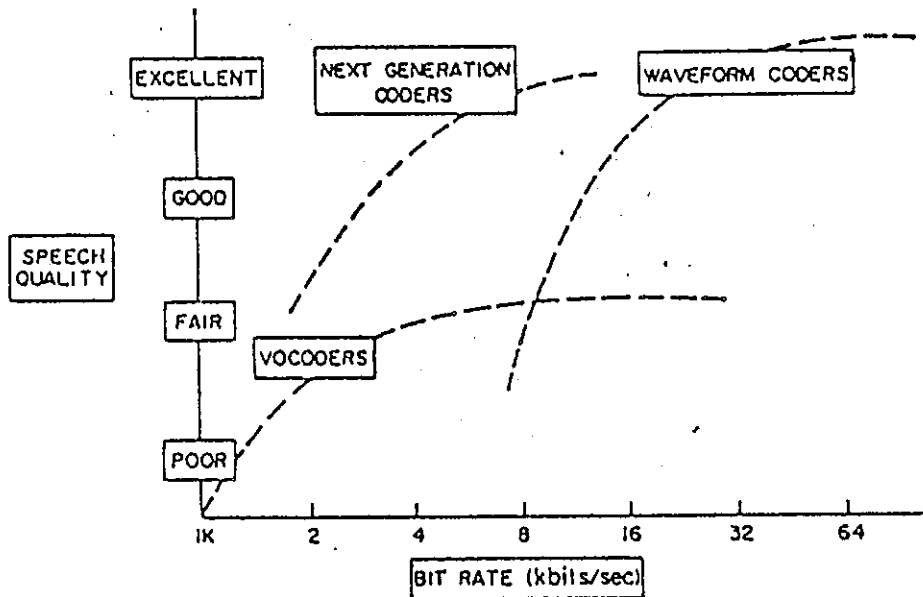
LPC	- Linear Prediction Coding.
Kbps	- K bit per second.
V/UV	- Voiced / Un-Voiced.
APC	- Adaptive Predictive Coder.
CELP	- Code Excited Linear Prediction.
MPE	- Multi-Pulse Excitation.
SNR	- Signal to Noise Ratio.
SEGSNR	- Segmented Signal to Noise Ratio (seg-size 256).
VQ	- Vector Quantization.
LBG	- Linde , Buzo, and Gray (Algorithm).
APC	- Adaptive Predictive Coder.
SPACE	- Switched Pulse And Code Excitation.

פרק 1 - מבוא

קידוד דיבור הוא שם כולל לתחום העוסק בהעברת אותות דיבור דרך ערוצים ספרתיים. שלושת הקריטריונים לפיהם נבחנים ביצועי המקודדים הם:

1. הקצב הדרוש להעברת האות בערוץ.
2. סיבוכיות אלגוריתם הקידוד.
3. האיכות האובייקטיבית והסובייקטיבית של האות המשוחזר.

בציור 1.1 מתואר עקום קצב איכות של משפחת מקודדי אותות דיבור המדגים חלוקה של המשפחה לשלוש מחלקות עקריות.



ציור 1.1 - איכות הדיבור כפונקציה של קצב השידור במקודדי אותות דיבור.

Fig. 1.1 - Speech quality versus bit rate for different types of coders.

מקודדי צורת גל (Waveform coders) משיגים איכות גבוהה בתחום קצבי שידור 16-64 Kbps אך ביצועיהם נופלים במהירות מתחת לתחום זה. במקודדי מקור (Source Coders או Vocoders) קצב העבודה האופיני הוא 2.4 Kbps אך איכות הדיבור סינטטית. אחד מכיווני המחקר העיקריים כיום, אליו משתיכת עבודה זו, הוא פיתוח של מערכות לתחום 9.6-16 Kbps באיכות גבוהה.

במקודדי מקור שיטת הקידוד מבוססת על התאמת מודל חיזוי לינארי (LPC) לאות הדיבור ושידור פרמטרי המודל. לפי מודל זה האות המשוחזר מתקבל על ידי מעבר סדרה מחזורית או סדרת רעש לבן דרך מסנן all pole לינארי משתנה בזמן. הסיבה העקרית שבגללה אין אפשרות לקבל איכות טובה במקודדים אלו היא שהעירור פשטני ואינו מייצג כראוי את אות השארית. אות השארית הוא העירור למסנן שהיה מאפשר שחזור מושלם של אות המקור.

שתי שיטות מבטיחות שהוצעו בספרות לקידוד אות השארית הם: MPE - Multi-Pulse Excitation ו- CELP - Code Excited Linear Prediction.

ב-MPE מיוצג אות השארית על ידי מספר קטן של פולסים (בכל מסגרת זמן) והוא מתאים לעבודה בתחום 9.6-16 kbps. (איכות גבוהה בקצה התחום העליון ו"איכות תקשורת" בקצה התחתון). ב-CELP מיוצג אות השארית על ידי סדרת רעש לבן והיא נבחרת מתוך אוסף גדול של סדרות (ספר קוד). מקודד זה מתאים לעבודה בתחום 4.8-9.6 kbps אך חיירונו הבולט הוא הסיבוכיות הגבוהה הכרוכה בחיפוש הסדרה הטובה ביותר.

במסגרת עבודה זו ניסינו לשפר את ביצועי שני המקודדים בקצב הביניים של 9.6 Kbps.

מבנה העבודה

בפרק 2 מוצגות המחלקות השונות לפיהן מחלקים את משפחת מקודדי דיבור, ובפרק 3 נסקור מקצת מהשיטות שהוצעו לקידוד אות השארית.

האלגוריתם האיטרטיבי שהוצע ב-[12] לחיפוש הפולסים במקודד ה-MPE הוא תת-אופטימלי, ובפרק 4 מתואר ניסיון לשפר את אלגוריתם החיפוש על ידי שימוש ברשת נוירונים לפתרון הבעיה הכללית.

בפרק 5 מתוארות הדרכים שבהן ניתן לעדכן את תנאי ההתחלה של מסנן הסינטזה במקודד ה-MPE כך שסכום התגובות לעירור ולתנאי ההתחלה יוביל לשחזור טוב יותר (כלומר מלבד המידע על העירור משודר למקלט מידע גם על תנאי ההתחלה האופטימליים). מוצגות שתי סכמות, האחת עם עדכון תנאי ההתחלה לפני תהליך מציאת הפולסים והשנייה עם עדכון תנאי ההתחלה אחרי מציאת הפולסים. הסכמה הראשונה נתנה ביצועים טובים יותר והיא זו ששמשה אותנו להמשך העבודה.

בפרק 6 מתוארות בקצרה שיטת קוונטזציה הוקטורית ואלגוריתם ה-LBG לתכנון מילון אופטימלי, ומוצגות התוצאות שהתקבלו כשהופעלה קוונטזציה ווקטורית על וקטור תנאי ההתחלה. התוצאות אינן מצביעות על כדאיות מימוש סכמה המשלבת עדכון וקידוד תנאי התחלה, בהשוואה לאלו המתקבלים במקודד ה-MPE המקורי.

בפרק 7 מתוארים מספר אלגוריתמים בהם מיוצג אות השארית על ידי שילוב של פולסים וספר קוד. חיפוש הקוד האופטימלי נעשה בדומה לאלגוריתם ה-CELP אך עם ספר קוד קטן בהרבה, דבר המביא לחיסכון ניכר בחישובים.

בפרק 8 מסוכם המחקר ומוצגים מספר כיוונים אפשריים למחקר המשך במטרה לשפר את המערכות מבחינת סיבוכיות ואיכות.

פרק 2 - קידוד אותות דיבור

2.1 מבוא

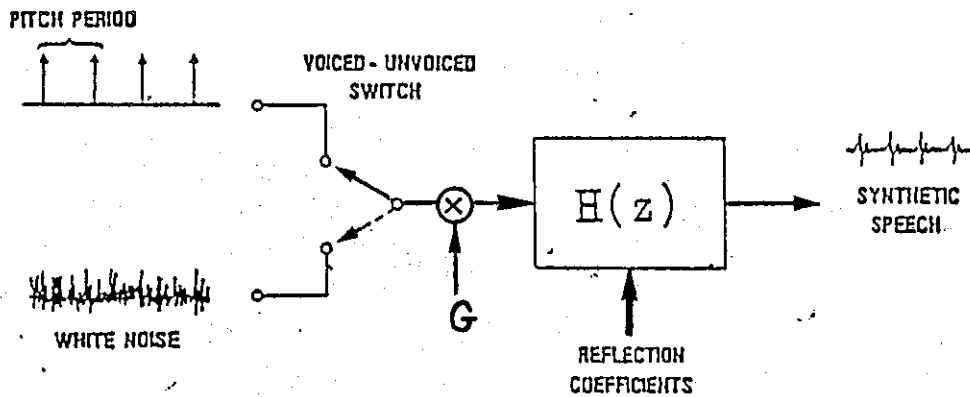
משפחת מקודדי אותות דיבור נחלקת לשלוש מחלקות עיקריות הנבדלות בקצב השידור, בסיבוכיות מערכת הקידוד ובאיכות ואופי הדיבור המשוחזר:

- א. מקודדי מקור - Source Coders או Vcoders.
- ב. מקודדי צורת גל - Waveform Coders.
- ג. מקודדי אות שארית - Residual Coders.

בפרק זה נתאר בקצרה את השתיים הראשונות ונציג את מגבלותיהן. נסכם בהצגת מקודדי אות שארית אשר ידונו בהרחבה בפרק הבא.

2.2 מקודדי מקור - Vcoders

במקודדי מקור מתאימים מודל מתמטי למערכת הביולוגית שמיצרת את אות הדיבור ומשדרים את פרמטרי המודל. שיטה מקובלת ופשוטה למימוש בקצבי שידור נמוכים היא שיטת החיזוי הליניארי (Linear Prediction) [1]. הרעיון מבוסס על המודל המוצג בציר 2.1.



ציר 2.1 - מודל LPC ליצירת אות הדיבור.
Fig. 2.1 - LPC model of speech production.

אות הדיבור מתקבל על ידי מעבר ערוך דרך מסנן all pole לינארי משתנה באיטיות בזמן - $H(z)$, שתפקידו ליצג את המעבר הקולי (Vocal Tract). העירור ממותג בין סדרת הלמים במחזור Pitch, שהוא התדר היסודי של אות דיבור קולי, לקבלת צלילים קולים (Voiced), לבין סדרת רעש לבן לקבלת צלילים אל-קולים (Un-Voiced).

בכל מסגרת זמן של 10-20 msec. נקבעים ומקודדים הפרמטרים הבאים
: [2]

- א. מקדמי המסנן $H(Z)$.
- ב. קבוע הגבר G להתאמת הספקים בין המקור לבין המקלט.
- ג. החלטה אם המסגרת קולית או לא וקביעת מחזור ה-Pitch אם המסגרת קולית.

מנקודת ראות ספקטרלית תורם המסנן את העוטפת של ספקטרום האות, בעוד שהעירור משלים את האיפיון העדין של הספקטרום [1].

2.3 קביעת פרמטרי ה-LPC

הרעיון שעומד מאחורי קידוד בעזרת חיזוי לינארי הוא שניתן לנצל את הקורלציה בין דגמי אות סמוכים כדי לרשום את דגם האות הנוכחי כקומבינציה לינארית של דגמי האות הקודמים ודגם ערוך נוכחי:

$$(2.1) \quad s(n) = \sum_{i=1}^P a_i s(n-i) + e(n)$$

נגדיר את $\hat{s}(n)$ כאותו חלק של דגם הדיבור שניתן לחיזוי על סמך P דגמים קודמים:

$$(2.2) \quad \hat{s}(n) = \sum_{i=1}^P a_i s(n-i)$$

ואת שגיאת החיזוי:

$$(2.3) \quad e(n) = s(n) - \hat{s}(n)$$

למציאת מקדמי המסנן מבצעים מינימיזציה על סך ריבועי שגיאות החיזוי:

$$(2.4) \quad E = \sum_n e(n)^2$$

סט המשואות המתקבל על ידי גזירת E לפי הפרמטרים a_i ($i=1, \dots, P$) והשוואה לאפס:

$$(2.5) \quad \sum_{i=1}^P a_i \phi(i, j) = \phi(j, 0) \quad j=1, \dots, P$$

$$\phi(i, j) = \sum_n s(n-i)s(n-j)$$

ונקרא סט המשואות הנורמליות. תחום הסכימה נקבע על סמך אחת משיטות האנליזה - אוטוקורלציה או קווריאנס [3]. פתרון בשיטת האוטוקורלציה מבטיח תיאורתית את יציבות המסנן. המטריצה המתקבלת במערכת המשואות היא ϕ טויטלצ' ϕ סימטרית, וקיים פתרון רקורסיבי פשוט על פי אלגוריתם Levinson-Durbin. בשיטת הקווריאנס לא מובטחת היציבות ויש לודא שכל שרשי המסנן נמצאים בתוך מעגל היחידה.

מקדם ההגבר G שווה לאנרגיית השגיאה המינימלית המתקבלת על ידי הצבת המקדמים האופטימליים לביטוי 2.4:

$$(2.6) \quad G = \phi(0, 0) - \sum_{i=1}^P a_i \phi(i, 0)$$

Pitch זיהוי 2.4

ה-Pitch הוא כאמור התדר היסודי של אות דיבור בהיגוי קולי ומיצג את מחזוריות בצורת הגל. תחום הערכים האופייניים הוא בין 50-450 Hz השקול ל- 160-20 דגמים בקצב דגימה של fs=8 KHz. קביעת גודל זה מהווה מרכיב חשוב במספר מערכות שנתאר בהמשך, המנצלות קורלציה בין דגמים במחזור Pitch (long term predictor) בנוסף לקורלציה בין דגמים סמוכים שממוצת על ידי מקדמי ה-LPC (short term predictor).

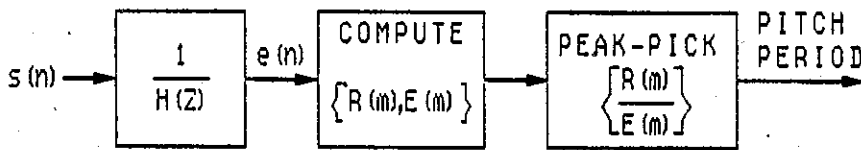
בספרות הוצעו מספר אלגוריתמים למציאת מחזור ה-Pitch [4][5]. אנו נתאר בקצרה את שיטת הקורלציה בה נעשה שימוש בעבודה זו. בשיטה זו מחשבים בכל מסגרת זמן של כ- 10 msec את סדרת מקדמי הקורלציה באות השארית - e(n) (זהו הערוור שהיה מביא לשיחזור מדויק של אות הדיבור), ומזהים את השיא בסדרה עם מחזור ה-Pitch. אות השארית מתקבל על ידי העברת אות הדיבור דרך המסנן ההפוך 1/H(Z) כפי שיובהר בסעיף 2.7 ומודגם בצירור 2.2. הדגם ה-m בסידרת הקורלציה - {r(m)} מחושב לפי:

$$r(m) = R(m) / E(m)$$

$$R(m) = \sum_n e(n)e(n-m)$$

$$E(m) = \sum_n e(n-m)^2$$

(2.7)



ציור 2.2 - זיהוי Pitch בשיטת הקורלציה.
Fig. 2.2 - Correlation Pitch detector.

החסרונות העיקריים של האלגוריתם הם: 1. סיבוכיות גדולה הכרוכה בחישוב סידרת הקורלציה ו 2. זיהוי שגוי של מחזור Pitch כפול כתוצאה מכך שפונקציית הקורלציה מחזורית במחזור Pitch, אך בכך דומים ביצועיו לאלו המתקבלים בטכניקות אחרות [5]. להשלמת מודל ה-LPC יש להחליט לגבי כל קטע אם הוא קולי או לא.

2.5 סיכום - מקודדי מקור

יתרונותיה העיקריים של אנליזת ה-LPC נעוצים הן בפשטותה והן ביכולתה לקודד ביעילות את פרמטרי המודל תוך שמירה על מובנות גבוהה בקצב שידור נמוך - 2.4 Kbps [6]. מצד שני, איכות הקול סינטטית ורגישה לריבוי דוברים ורעש רקע - גם בשיחזור מדויק של פרמטרי המודל (ללא קידוד). תופעות אלו ניגזרות בעיקר מפשטות המודל לעירור: חוסר הטיפול בקטעים שאינם משתייכים לקטגוריות V/UV וההתעלמות מהמיקום והאמפליטודה המדויקים של פולסי העירור [7].

2.6 מקודדי צורת גל

בניגוד לגישה שננקטה ב-Vocoders, שבה כופים מודל על אות הדיבור, במערכות השיכות למשפחה זו מקודדים את האות במטרה לשחזר במדויק ככל האפשר את צורת הגל [8]. מקובל לחלק את המשפחה לשתי תת-מחלקות:

- א. מקודדים בתחום הזמן (כגון מקודדים פרדקטיביים).
- ב. מקודדים בתחום התדר (כגון מקודדי התמרה).

המקודדים בתחום הזמן מנצלים את הקורלציה בין דגמים סמוכים כדי לקבל את אות ההפרש בין האות לחיזויו. הפוטנציאל לקידוד יעיל טמון בכך ששונות אות ההפרש קטנה מזו של אות הדיבור ולכן דרוש מספר קטן יותר של סיביות לייצוג האות.

במקודדים בתחום התדר (או בתחום התמרה כלשהוא) מנצלים את חוסר הקורלציה בין דגמי האות המותמר ליישום שיטות קידוד יעילות הידועות לאותות חסרי קורלציה. לקבוצה זו שיכים מקודדי התמרה Transform Coders ומקודדים בפסי תדר נפרדים Subband-Coders הנבדלים ברזולוציה ובטכניקות החלוקה של ספקטרום הדיבור.

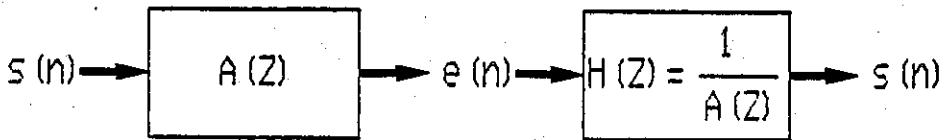
התכונות שמאפיינות את משפחת מקודדי צורת גל הם: קצב שידור גבוה 16-64 Kbps הנדרש להשגת איכות גבוהה בלתי תלויה ברעש רקע או ריבוי דוברים, ונפילה מהירה באיכות הדיבור כשיורדים בקצב.

2.7 אות שארית מחיזוי לינארי

אות שארית - $e(n)$ הוא העירור למסנן ה-LPC - $H(Z)$, שמוביל לשיחזור מדויק של אות המקור - $s(n)$, ומתקבל באמצעות המסנן ההפוך $A(Z)$, כפי שמתואר בציור 2.3.

כאשר:

$$(2.8) \quad \begin{aligned} H(Z) &= 1/A(Z) \\ A(Z) &= 1 + \sum_{i=1}^p a_i Z^{-i} \end{aligned}$$



ציור 2.3 - קבלת אות שארית ושחזור אות הדיבור.
 Fig. 2.3 - Generating residual signal and reconstructing speech signal.

במקודדי שארית מנצלים את פוטנציאל הדחיסה של מקדמי מסנן החיזוי ושל אות השארית כדי להשיג איכות דיבור טובה ("איכות תיקשורת") בתחום הקצבים 8-16 Kbps. בפרקים הבאים נתאר בהרחבה מקודדים השיכים למחלקה זו ומספר ניסיונות לשפר את ביצועיהם.

פרק 3 - מקודדי שארית

3.1 מבוא

בפרק זה נתאר מספר מקודדים (MPE, CELP, APC) בהם משודר מידע על אודות השארית, בנוסף למקדמי ה-LPC, כדי לקבל איכות טובה בקצבים 8-16 Kbps. כן נציג תוצאות סימולציה לשניים מהם (CELP, MPE) שבהם נתרכז בהמשך העבודה.

תחילה נתאר בקצרה את מקודד ה-APC (Adaptive Predictive Coder) בו כל דגם באות השארית עובר קוונטיזציה בנפרד. המקודד תוכנן לתחום 9.6-16 Kbps ומשיג ביצועים טובים למדי. הבעיה היא שבקצבים מתחת ל 10 Kbps הקצאת הסיביות לדגם קטנה, הקוונטיזציה גסה ואיכות הדיבור המשוחרר נפגעת.

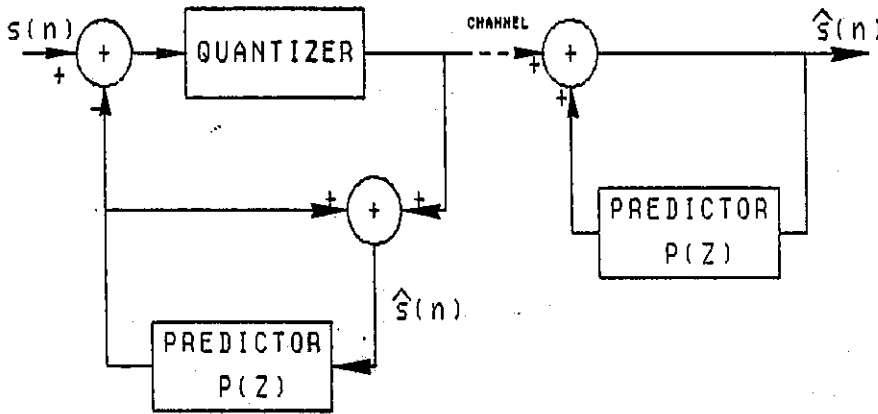
ברור אם כך שדרושות שיטות שתיצגנה את אות השארית בפחות מ-1 bit לדגם. הטכניקות שנמאר מתגברות על הבעיה על ידי כך שהקידוד מבוצע למסגרות זמן ולא לדגמים בודדים:

MPE - Multi-Pulse Excitation - בכל מסגרת זמן מיוצג אות השארית על ידי מספר קטן של פולסים חשובים. הפוטנציאל הטמון בצורת קידוד זו נעוץ בעובדה שמספיקים כ-4 פולסים במסגרת של 5msec (כלומר 800 פולסים לשניה במקודד עם קצב תמסורת של כ- 10 Kbps) כדי להשיג איכות טובה.

CELP - Code Excited Linear Prediction - בכל מסגרת זמן נבחר העירור האופטימלי מתוך מילון סדרות אקראיות. העירור נבחר כך שיתקבל מינימום שגיאה ריבועית בין אות המקור והאות המשוחרר ומשדרים את אינדקס וקטור העירור שנבחר מהמילון. החיסרון של האלגוריתם הוא סיבוכיותו הגבוהה הנובעת מתהליך החיפוש. מצד שני, ניתן להשיג עמו איכות טובה בתחום 4.8-9.6 Kbps.

[10][9] Adaptive Predictive Coders - APC 3.2

הסכמה הבסיסית של מקודד APC מוצגת בצירור 3.1.



צירור 3.1 - סכמת בלוקים של APC.
Fig. 3.1 - Block diagram of APC.

ההפרש בין דגם אות הדיבור לחיזויו עובר קוונטיזציה ומשודר למקלט. המקלט משחזר את אות המקור על ידי סיכום אות ההפרש שנקלט עם האות החזוי. בערוץ משודרת אינפורמציה על אות השארית ועל הפרמטרים המשתנים של החזאי והקוונטיזציה. שגיאת השחזור במקודד זה היא שגיאת הקוונטיזציה. שונות שגיאת הקוונטיזציה יחסית לשונות האות בכניסה לקוונטיזציה ולכן רצוי שהחזאי ימצה את את הקורלציה באות הדיבור ויביא למינימום את שונות שגיאת החיזוי. החזאי P(Z) מורכב משני חזאים ניפרדים:

$$(3.1) \quad P(Z) = (1 - P_s(Z))(1 - P_p(Z))$$

$P_s(Z)$ - חזאי ה-LPC שהוצג בפרק קודם המבוסס על הקורלציה בין דגמים סמוכים.

$$(3.2) \quad P_s(Z) = \sum_{i=1}^M a_i Z^{-i}$$

$P_p(Z)$ - חזאי Pitch בעל 1-3 מקדמים (taps) שמנצל את הקורלציה בין דגמים במחזורי Pitch עוקבים.

$$(3.4) \quad P_p(Z) = \begin{cases} \beta_1 Z^{-M} & 1 \text{ tap} \\ \beta_1 Z^{-(M-1)} + \beta_2 Z^{-M} & 2 \text{ taps} \\ \beta_1 Z^{-(M-1)} + \beta_2 Z^{-M} + \beta_3 Z^{-(M+1)} & 3 \text{ taps} \end{cases}$$

M - מחזור ה-Pitch המתקבל בתהליך נפרד כדוגמת זה שהוצג בסעיף 2.4, ומקדמי החזאי $P_p(z)$ מחושבים כך שיתקבל מינימום אנרגיה של אות השארית החדש המתקבל על ידי העברת האות $e(n)$ מצירור 2.2 (אות השארית מחיזוי לינארי עם חזאי מסדר P) דרך המסנן $1/(1-P_p(Z))$:

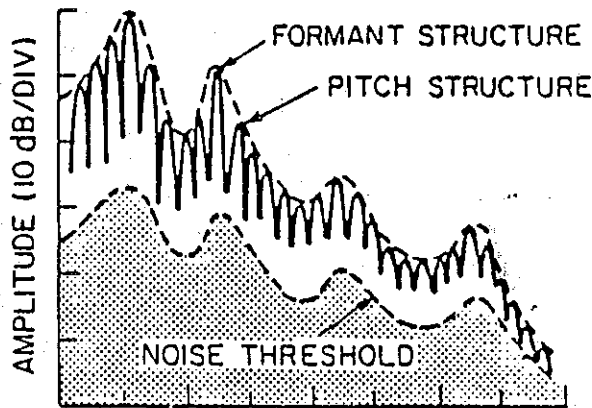
$$(3.5) \quad E = \sum_n (e(n) - \sum_i \beta_i e(n-M-1+i))^2$$

בסיום מוודאים את יציבות המסנן [11].

APC 3.3 עם עצוב ספקטרום שגיאת הקונטיזציה

כפי שצוין בסעיף הקודם, שגיאת השחזור נובעת משגיאת הקונטיזציה של אות השארית והיא בקרוב לבנה (ככל שהקונטיזציה גס יותר קרוב זה טוב פחות). ניתן לנצל את פונקצית הרגישות הספקטרלית של האוזן [12] כדי לעצב לפיה את ספקטרום השגיאה ולהשיג איכות סוביקטיבית טובה יותר.

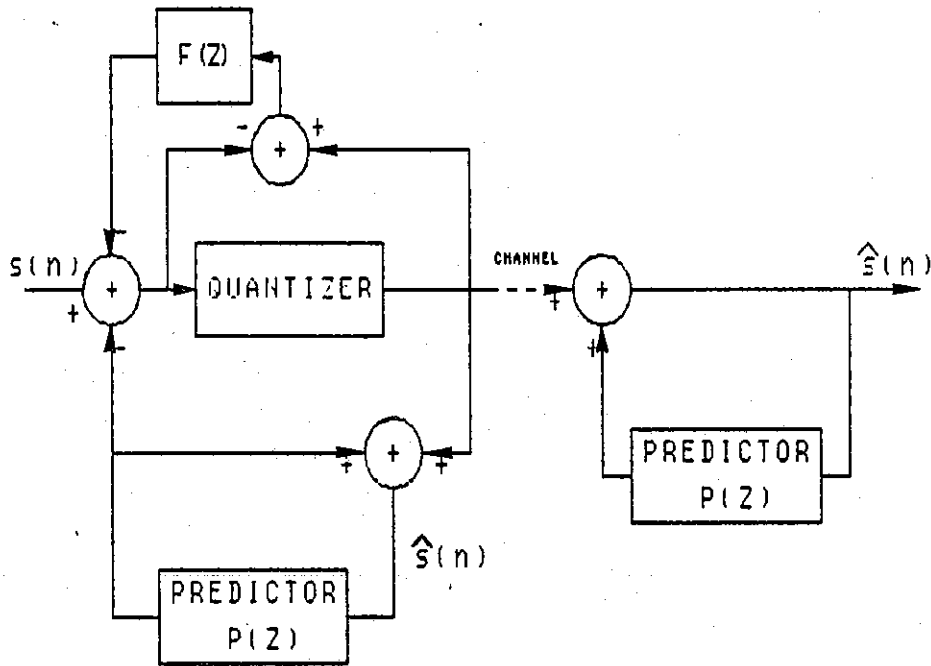
באזורי הפורמנטים מרוכזת אנרגיה הממסכת רעש ולכן נוכל להרשות לרכיבי תדר באות השגיאה להיות בעלי אנרגיה גדולה יותר יחסית לרכיבי תדר של השגיאה באזורים שבין הפורמנטים. כמו כן בצלילים קוליים מרוכזת מרבית האנרגיה בתדרים נמוכים והיא ממסכת את הרעש בתדרים הגבוהים. עובדה זו מאפשרת ליחסי אות לרעש בתדרים הגבוהים להיות קטנים מאלו שבתדרים הנמוכים. ספקטרום הספק אופיני לדיבור קולי ותחום עוצמות הרעש שממוסך על ידי האות מודגם בציור 3.2.



ציור 3.2 - ספקטרום הספק אופיני לדיבור קולי ותחום עוצמות הרעש שממוסך על ידי האות.

Fig. 3.2 - Typical power spectrum of voiced speech signals and the masked noise threshold.

בציור 3.3 מוצגת סכמה של מקודד APC בה יש שליטה על מעטפת ספקטרום השגיאה בעזרת המסנן $F(Z)$.



צילור 3.3 - APC עם שליטה על מעטפת ספקטרום שגיאת הקונטיזציה.
 Fig. 3.3 - APC with noise spectrum control.

לסיכום נצין כי כדי לרדת במערכת זו לקצבים נמוכים מ-10 Kbps נדרש לקודד את אות השארית בפחות מסיבית אחת לדגם. קונטיזציה גסה זו גורמת לירידה חזקה באיכות הדיבור המשוחזר.

Multi Pulse Excitation - MPE 3.4

מבוא

בשיטה זו ארת העירור המשודר למקלט מורכב ממספר קטן של פולסים חשובים הנבחרים בכל מסגרת זמן כך שתקבל אנרגיה שגיאה משוקללת מינימלית. האינפורמציה המשודרת בנוסף לפרמטרי ה-LPC היא מיקום הפולסים והאמפליטודות שלהם.

3.4.1 בסעיף נגדיר את הבעיה הכללית של קביעת המיקומים והאמפליטודות ואת קריטריון השגיאה.

3.4.2 בסעיף נתאר את האלגוריתם לפתרון הבעיה כפי שהוצע ב [12].

3.4.3 בסעיף נתאר מספר שיפורים שהוצעו בספרות לאלגוריתם הבסיסי, ונסכם בהצגת תוצאות סימולציות.

3.4.1 מודל ה Multi Pulse לעירור

במודל זה מיוצג העירור - $u(n)$ על ידי מספר קבוע של פולסים - K בכל מסגרת זמן:

$$(3.6) \quad u(n) = \sum_{i=1}^K b_i \delta(n-m_i)$$

b_i, m_i הם המיקומים והאמפליטודות של הפולסים בהתאמה. האות המשוחזר $s(n)$ מתקבל על ידי:

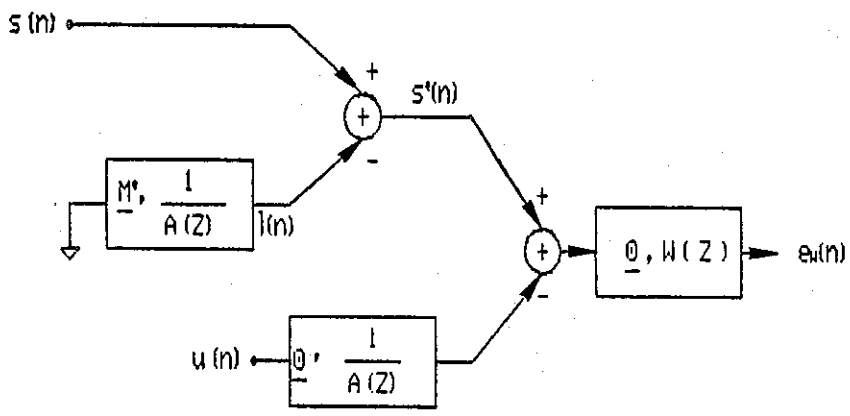
$$(3.7) \quad s(n) = \sum_{i=1}^K b_i h(n-m_i) + l(n)$$

כאשר $h(n)$ - תגובה להלם של מסנן הסינטזה:

$$(3.8) \quad \begin{aligned} h(n) &= Z^{-1}\{1/A(Z)\} \\ A(Z) &= 1 + \sum_{i=1}^p a_i Z^{-i} \end{aligned}$$

$l(n)$ - תגובת מסנן ה-LPC לתנאי התחלה M שהתקבלו בסיום הסינטזה של המסגרת הקודמת.

בציור 3.4 מתואר מודל ה-Multi-Pulse לעירור המשלב את קריטריון השגיאה המשוקללת. לכל מסנן מצוינים בתרשים: 1. וקטור הזכרון המתאים לחישוב תגובת הזכרון. 2. פונקציית התמסורת שלו.



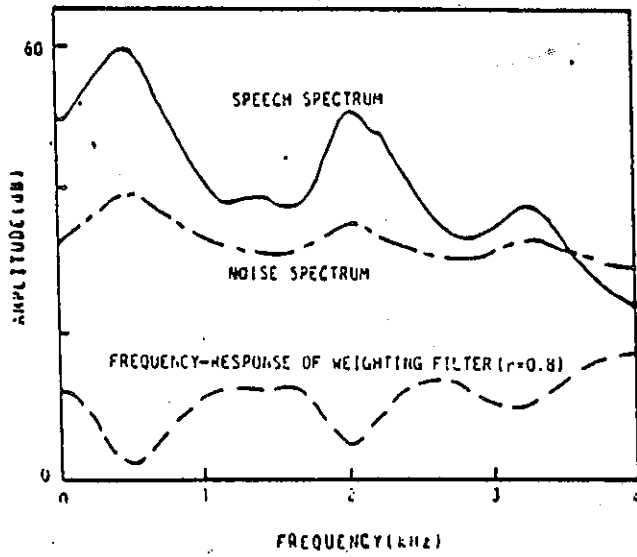
ציור 3.4 - סכמה של מודל ה- Multi-Pulse לעירור.
 Fig. 3.4 - Schematic model of Multi Pulse excitation.

$W(Z)$ - פונקציית תמסורת של מסנן (עם זכרון מאופס) שמשקלל את ספקטרום השגיאה לפי פונקציית הרגישות של האוזן והאינדקס w מציין כי האות הועבר דרך מסנן זה.

פונקציית התמסורת $W(Z)$ נתונה על ידי:

$$(3.9) \quad W(Z) = \frac{A(Z)}{A(Z/\Gamma)}$$

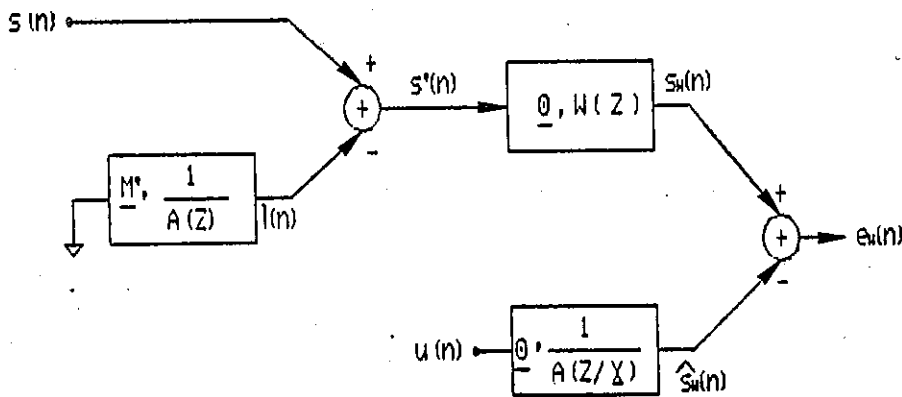
והמקדם Γ , $0 \leq \Gamma \leq 1$, שולט על מידת ההדגשה של השגיאה באזורי הפורמנטים. במבחני האזנה נמצא שערך אופיני צריך להיות 0.7-0.9. בציור 3.5 מובאת דוגמא לתגובת התדר של ספקטרום ההספק לאות דיבור לעומת תגובת התדר של המסנן המשקלל עבור $\Gamma=0.8$ וספקטרום ההספק של ארת השגיאה.



ציור 3.5 - דוגמא לספקטרום ההספק של אות דיבור, אות השגיאה ותגובת התדר של המסנן המשקלל.

Fig. 3.5 - An example of speech power spectrum, frequency response of the weighting filter, and noise power spectrum.

עתה, לאחר שהוגדר $W(Z)$ ניתן להעבירו לפני המסכס (ציור 3.4) ולקבל את המסנן השקול בענף התחתון על ידי הכפלת פונקציות התמסורת כמתואר בציור 3.6.



ציור 3.6 - העברת המסנן המשקלל מציור 3.4 לפני המסכס.

Fig. 3.6 - Moving weighting filter $W(Z)$ (fig. 3.4) before the adder.

נסמן $h_w(n)$ - התגובה להלם של מסנן ה-LPC המשוקלל:

$$(3.10) \quad h_w(n) = Z^{-1}\{1/A(Z/\Gamma)\}$$

$$(3.11) \quad s_w(n) = \sum_{i=1}^K b_i h_w(n-m_i) \quad \text{מצירור 3.6 נקבל:}$$

$$(3.12) \quad E = \sum_n e_w(n)^2 = \sum_n (s_w(n) - \sum_{i=1}^K b_i h_w(n-m_i))^2$$

ואנרגיית השגיאה המשוקללת:

קביעת המיקומים האופטימליים אינה בעיה לינארית ופתרון מדויק דורש מעבר על כל הקומבינציות האפשריות של K מיקומים מתוך N . גישה זו אינה מעשית כיוון שמספר קומבינציות טיפוסי הוא לדוגמה $10^7 \approx (64!/(59!5!))$ - בחירת 5 פולסים מתוך אוסף של 64. בסעיף הבא נתאר את האלגוריתם האיטרטיבי שהוצע ב-[13] שעל אף היותו פתרון תת-אופטימלי משיג ביצועים טובים למדי בתחום קצבי הבינים - 9.6-16 Kbps.

3.4.2 אלגוריתם ה-MPE

בתחילה, ללא פולסי עירור, האות המשוחזר נוצר מזכרון מסנן הסינטזה. מפחיתים את תגובת הזכרון מהאות המקורי ומעבירים את האות דרך המסנן המשקלל $W(Z)$ (עם זכרון מאופס).

$$(3.13) \quad s_w(n) = (s(n) - l(n)) * w(n)$$

כאשר

* - קונבולוציה

$$w(n) - \text{תגובת ההלם של המסנן המשקלל} - \{w(n)\} = Z^{-1}\{W(Z)\}$$

מבצעים מינימיזציה של E ב-(3.12) עבור מקרה של פולס בודד:

$$(3.14) \quad E = \sum_{n=0}^{N-1} e_w(n)^2 = \sum_{n=0}^{N-1} (s_w(n) - b_1 h_w(n-m_1))^2$$

כעת מחשבים שגיאה חדשה על ידי הפחתת התרומה של הפולס לשחזור, והתהליך חוזר על עצמו K פעמים. נגדיר וקטור קרוסקורלציה $\underline{\alpha}$:

$$(3.15) \quad \underline{\alpha} = [\alpha(1), \dots, \alpha(N)]$$

$$\alpha(i) = \sum_{n=0}^{N-1} s_w(n) h_w(n-i)$$

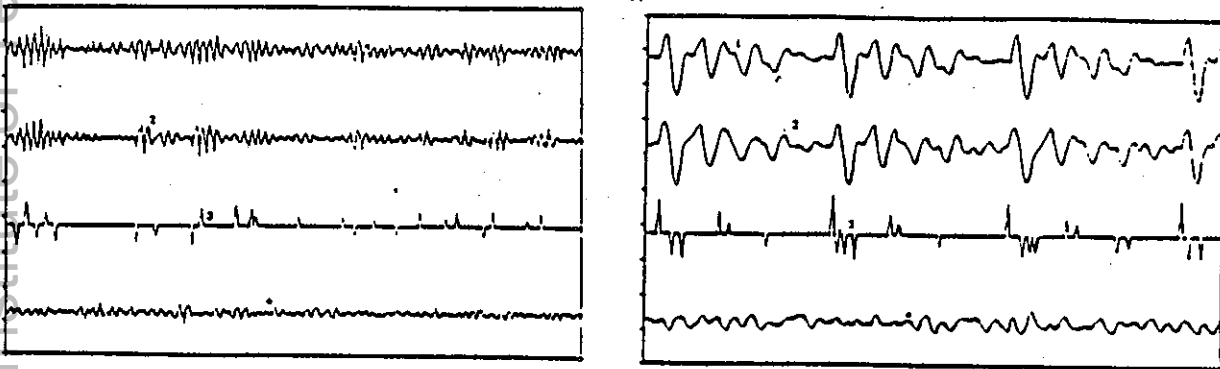
ואת פונקציית האוטוקורלציה של המסנן:

$$(3.16) \quad \Phi(i, j) = \sum_{n=0}^{N-1} h_w(n-i) h_w(n-j)$$

האלגוריתם המתקבל:

0. אתחול $k=0$, $\alpha_0 = \underline{a}$, $e_{w0}(n) = s_w(n) = (s(n) - l(n)) * w(n)$
1. מצא איבר מקסימלי בסדרה: $\{ \alpha_k(i) / \phi(i, i) \} i=0, \dots, N-1$ אינדקסו m .
2. חשב את האמפליטודה $b_k = \alpha_k(m) / \phi(m, m)$
3. אם נמצאו פולסים, סיים.
4. עדכן את השגיאה $e_{wk+1}(n) = e_{wk}(n) - b_k h_w(n-m)$
- למעשה עדכן את α_k $\alpha_{k+1}(i) = \alpha_k(i) - b_k \phi(i, m)$
5. חזור ל 1.

בציור 3.7 מוצגת התוצאה של הפעלת האלגוריתם עבור קטע קולי וקטע אל-קולי. ניתן לראות שבקטע הקולי העירור מקבל מבנה מחזורי במחזור Pitch ובקטע האל-קולי העירור מקבל צורה אקראית דמוית רעש.



קטע אל-קולי

קטע קולי

ציור 3.7 - האותות המתקבלים באלגוריתם האיטרטיבי עבור קטע קולי וקטע אל-קולי: 1. אות המקור 2. האות המשוחזר 3. אות העירור 4. אות השגיאה.

Fig. 3.7 - Signals obtained by the iterative algorithm in Voiced and Un-Voiced frames: 1. original signal 2. reconstructed signal 3. excitation signal 4. error signal.

1. אנליזת ה-MPE בשיטת האוטוקורלציה [13]

ניתן להקטין את סיבוכיות האלגוריתם שהוצג על ידי כך שניקבע זמן תגובה סופי וקבוע לתגובת מסנן הסינטזה המשוקלל (שקול להכפלת התגובה בחלון מלבני). הצדקה להנחה זו מתקבלת מתוך הביטוי לתגובה:

$$(3.17) \quad h_w(n) = Z^{-1} \{1/A(Z/\Gamma)\} = h(n)\Gamma^n$$

כאשר $h(n)$ - תגובת מסנן הסינטזה.

$h(n)$ היא תגובת מסנן יציב ולכן דועכת בזמן, $\Gamma < 1$ ולכן $h_w(n)$ דועך מהר יותר. נניח אם כך שהחל מנקודה מסוימת - L , התגובה להלם מתאפסת. נוסף לכך נקבע את תחום המסגרת לחישוב השגיאה להיות $0 \leq (N-1+L)$ (תחום החיפוש למיקום הפולסים נשאר $0 \leq (N-1)$). נקבל כי רכיבי האוטוקורלציה החדשים של המסנן

$$(3.18) \quad \Phi(i, j) = \begin{cases} \sum_{n=0}^{N+L-1} h_w(n-i)h_w(n-j) & |i-j| \leq L-1 \\ 0 & \text{אחרת} \end{cases}$$

צורה זו של אנליזה מבטלת את הסיבוכיות הכרוכה בחישוב הסדרה $\{\Phi(i, i)\}_{i=1}^N$ בשלב 1 באלגוריתם - כיוון שאיברי סדרה זו קבועים, ולכן חיפוש מיקום האיבר המקסימלי מתבצע על הסדרה $\{\alpha_k(i)\}$ ולא על $\{\alpha_k(i)/\Phi(i, i)\}$.

ביצועי האלגוריתם נפגעים רק במעט (בהשוואה לאנליזה בשיטת הקווריאנס) אם עבור $\Gamma = 0.8$ משתמשים ב- $L = 10-20$ דגמים.

2. רה-אופטימיזציה לאמפליטודות [14]

שיפור בבביצועי האלגוריתם מתקבל אם מבצעים רה-אופטימיזציה לאמפליטודות לאחר כל איטרציה באלגוריתם, כלומר פותרים את המערכת:

$$(3.19) \quad \alpha(j) = \sum_{i=1}^k b_i \Phi(i, j) \quad j=1, \dots, k$$

בצורה זו נשמרת השגיאה אורתוגונלית לעירור בכל שלבי האלגוריתם ולכן נמנע מצב בו עלול להבחר באחת האיטרציות פולס במיקום שנבחר קודם.

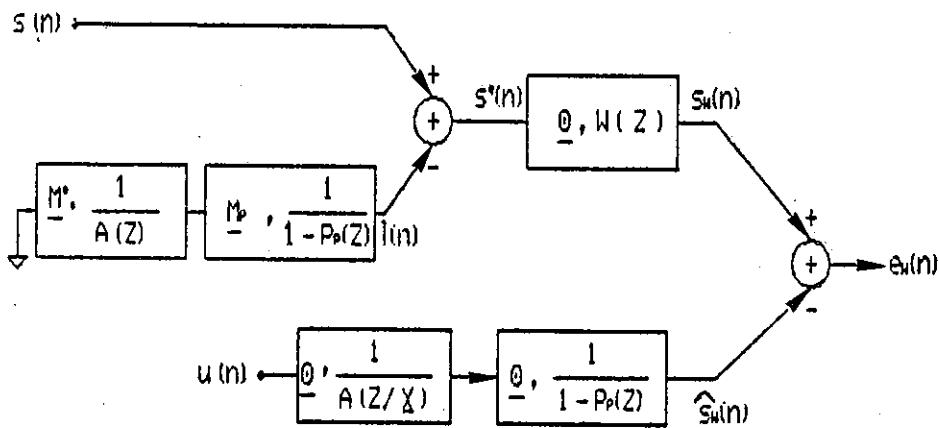
3. הוספת חזאי Pitch [14]

כפי שצויין בסיום סעיף 3.4.2, אות העירור מתקבל במבנה מחזורי המתאים למחזור ה-Pitch. ניתן להקטין את מספר הפולסים ולשמור על אותה איכות על ידי הוספת חזאי Pitch - $P_p(z)$ למסנן הסינטזה.

כמו כן, מסתבר, שבאלגוריתם המקורי איכות הדיבור המשוחרר תלויה במספר הפולסים הממוצע במחזור Pitch, כלומר, ככל שתדר ה-Pitch יהיה גדול יותר הקצב שבו נצטרך לשרי יגדל כדי לשמור על איכות קבועה. הוספת החזאי $P_p(z)$ מפחיתה את רגישות המערכת לגודל זה. הסכמה החדשה מתוארת בציור 3.8, והשינוי באלגוריתם הוא שתגובת ההלם של מסננת הסינטזה החדשה ניתנת על ידי:

$$(3.20) \quad h_w(n) = Z^{-1} \{W(Z) / (A(Z)(1 - P_p(Z))\}$$

ותגובת הזכרון כוללת את תרומת התגובה לזכרון M_p של המסנן $1/(1 - P_p(z))$.



ציור 3.8 - שילוב חזאי Pitch בסכמת ה-MPE.

Fig. 3.8 - Pitch predictor filter used in MPE analysis.

אם משך מסגרת האנליזה הוא קצר מזמן מחזור Pitch מינימלי (כ-20 דגמים), ניתן לותר על תרומת מסנן ה-Pitch לתגובת ההלם של מסננת הסינטזה ולחשבה לפי:

$$(3.21) \quad h_w(n) = Z^{-1} \{W(Z) / A(Z)\} = Z^{-1} \{1 / A(Z/\Gamma)\}$$

כיוון שמסנן ה-Pitch משכפל את תגובת ההלם במחזור Pitch (עם שקלול שנקבע לפי מקדמי החזאי) והיא מתקבלת מחוץ למסגרת האנליזה.

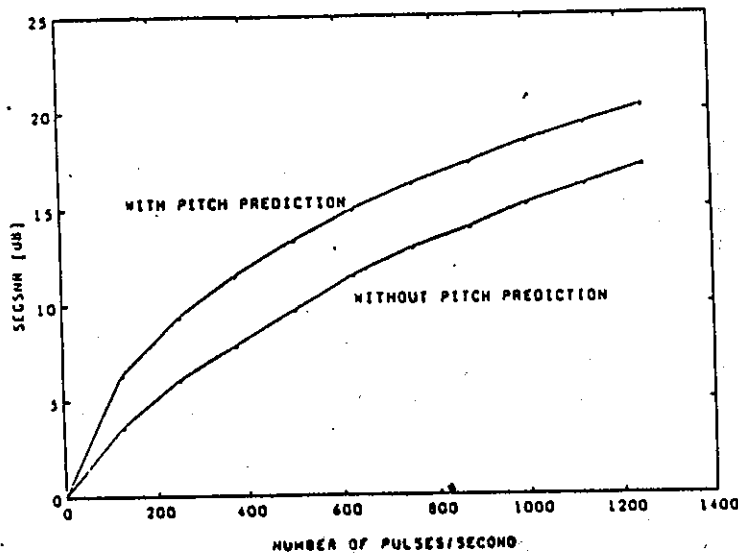
3.4.4 תוצאות סימולציות

הסימולציות שיתוארו בסעיף זה ובהמשך העבודה נערכו על 3 משפטים באנגלית שנאמרו כל אחד על ידי גבר ואשה (סה"כ 6 משפטים). אות המקור האנלוגי הוגבל לתחום התדרים 200-3200 Hz, נדגם בתדר $f_s=8$ KHz ורוחב מילת הדגימה 12 bit. כמו כן נקבעו הפרמטרים הבאים כקבועים לכל הסימולציות אלא אם צוין אחרת:

סדר מודל ה-LPC נלקח $P=10$, המקדמים מעודכנים בכל $M=192$ דגמים (24 msec) ושיטת אנליזת ה-LPC היא שיטת האוטוקורלציה. אורך הבלוק עליו מחושבים מקדמי האוטוקורלציה באנליזת ה-LPC הוא $M=256$ דגמים (30 msec) והוא מוכפל בחלון HANNING - $0.5(1-\cos(2\pi n/M))$.

חזאי ה-Pitch הוא בעל 3 מקדמים המעודכנים בכל 64 דגמים (8 msec). אורך הבלוק עליו מחושבים מקדמי הקורלציה לקביעת המקדמים הוא 128 (16 msec.) ובדיקת יציבות המסנן נעשית לפי [11].

בציור 3.9 מתואר ה-SEGNSNR (SEGMENTAL SNR) - ממוצע ב-dB של יחסי אות לרעש בבלוקים של 256 דגמים) כפונקציה של מספר הפולסים בעירור, עבור אנליזה MPE בשיטת הקוריאנט, עם ובלי חזאי Pitch. בשני המקרים בוצעה רה-אופטימיזציה לאמפליטודות בסיום כל איטרציה. גודל מסגרת האנליזה לאלגוריתם ה-MPE הוא $N=64$ דגמים (8 msec) ו- $\Gamma=0.8$.



ציור 3.9 - SEGNSNR המתקבל באנליזת קוריאנט עם ובלי חזאי Pitch.
Fig. 3.9 - SEGNSNR in covariance analysis with and without Pitch predictor.

התוצאות אומנם מצביעות על יתרון של אנליזה עם חזאי Pitch אבל יש לבחון את המחיר בסיבוכיות ובקצב הדרושים לשידור מקדמי החזאי Pitch ולהשוותם לקידוד פולסים נוספים באנליזה ללא חזאי כדי לקבוע איזו משתי הגישות עדיפה.

[17],[16] Code Excited Linear Prediction - CELP 3.5

במקודד ה-CELP, המכונה גם Stochastic Coder, העירור נבחר מתוך מילון של סדרות אקראיות לבנות מפולגות גאוסית. הסיזרה נבחרת כך שיתקבל מינימום שגיאת שיחזור ריבועית משוקללת, והפרמטר שמשודר עבור כל מסגרת זמן הוא אינדקס הסיזרה במילון וקבוע הגבר. החיסרון הבולט של האלגוריתם הוא סיבוכיותו הגבוהה הכרוכה בתהליך החיפוש של הסדרה הטובה ביותר.

מקודד זה התפתח ממקודד APC על ידי כך שיושמו טכניקות קידוד וקטוריות (Block Code או Vector Quantization) לאות העירור במקום הקוונטיזציה הסקלרית, אך תוך שימוש בגישה של אנליזה על ידי סינטזה כמו ב-MPE.

טכניקת קידוד וקטורי תואר בפרק 6 אך נציין שניתן להתקרב בעזרתה לפונקציית קצב עיוות של המקור - $R(D)$, על ידי הגדלת אורך הוקטור N (מסגרת זמן האנליזה), ולכן ניתן להשיג איכות גבוהה גם בקצבים נמוכים במיוחד.

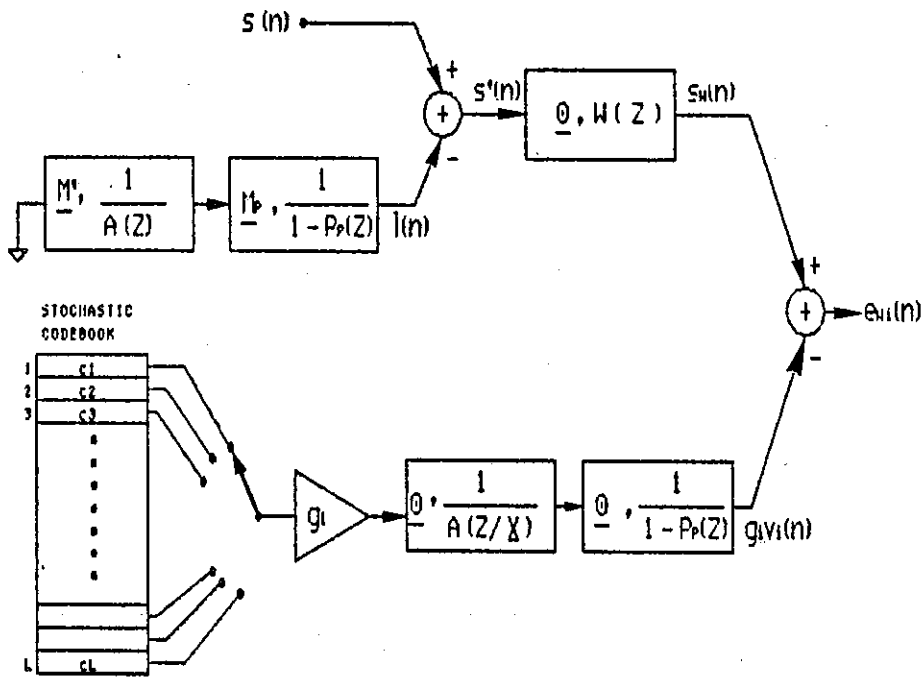
בעוד שבמקודד ה-APC ההקצאה המינימלית היא סיבית אחת לדגם הרי שכאן ההקצאה נקבעת לפי מספר הוקטורים במילון - L :
 $R = (1/N) \log_2(L)$ (כאשר N - אורך הוקטור). ולכן ניתן לשלוט בקצב בקלות על ידי שינוי מספר הוקטורים במילון.

אכלוס המילון נעשה כך שהוקטורים ייצגו את מרחב ההסתברות ה- N מימדי של וקטורי עירור אפשריים. ב-[17] נמצא שפילוג גאוסית הוא הקרוב ביותר לפילוג אות העירור (לאחר שהוצא מימנו מבנה ה-Pitch על ידי המסנן ההפוך $1/(1-Pp(Z))$ וששימוש בוקטורים חסרי קורלציה בין הדגמים (בעלי ספקטרום לבן) אינה גורעת בהרבה מהאיכות, בהשוואה למילון מאומן (מילון שנבנה מתוך סיזרת אימון Training Sequence כדוגמת אלגוריתם ה-LBG שיוצג בפרק 6) כמו כן, נמצא שעבור מסגרות זמן של 5 msec (40 דגמים) מספיק מילון המכיל 1024 סדרות עירור ליצוג טוב של מרחב סדרות העירור. החיפוש של הסדרה האופטימלית במילון גורם לעליה ניכרת בעומס החישוב.

3.5.1 בחירת סיזרת עירור אופטימלית

בציר 3.10 מתוארת סכמה המדגימה את תהליך החיפוש של סדרת העירור האופטימלית. שקלול השגיאה מבוצע בעזרת המסנן $W(Z)$ שהוצג קודם ומטעמי נוחות שולב קודם למחסר (כפי שנעשה בציר 3.6). מסנן הסינטזה במקודד CELP - $P(Z)$, מורכב ממסנן ה-LPC וממסנן המכיל חזאי Pitch, כיוון שתחת תנאים אלו אות העירור אכן מתאים למודל של רעש לבן גאוסית.

$$(3.22) \quad P(Z) = 1/[A(Z)(1-Pp(Z))]$$



ציור 3.10 - סכמה המדגימה את תהליך החיפוש של סדרת העירור האופטימלית.

Fig. 3.10 - Block diagram illustrating the procedure for selecting the optimum innovation sequence.

בדומה למקודד ה-MPE מופחתת תחילה תגובת הזכרון של מסנן הסינטזה מאות המקור ומחושב האות המשוקלל $s_w(n)$.

כל סידרה במילון $\{c_i(n)\}_{i=1}^L$ מוכפלת בגורם הגבר g_i ומועברת דרך מסנן הסינטזה המשוקלל - $P_w(Z)$:

$$(3.23) \quad P_w(Z) = W(Z) / [A(Z)(1 - P_p(Z))] = 1 / [A(Z/\Gamma)(1 - P_p(Z))]$$

נסמן:

$$h_w(n) = Z^{-1} \{P_w(Z)\}$$

$$v_i(n) = c_i(n) * h_w(n)$$

סך השגיאה הריבועית המשוקללת:

$$(3.24) \quad E_e = \sum_n (s_w(n) - g_i v_i(n))^2$$

סדרת העירור האופטימלית היא זו שעבורה גודל זה מינימלי.

קבוע ההגבר - g_i מחושב לכל סידרה בנפרד כך שיתקבל מינימום ל- E_i .
 גזירה של ביטוי 3.24, השואה לאפס וחילוץ ההגבר האופטימלי
 נותנת:

$$(3.25) \quad g_i = \frac{\sum_n s_w(n)v_i(n)}{\sum_n v_i(n)^2}$$

והשגיאה המינימלית מתקבלת על ידי הצבתו ב 3.24:

$$(3.26) \quad E_i = \sum_n s_w(n)^2 - \frac{(\sum_n s_w(n)v_i(n))^2}{\sum_n v_i(n)^2}$$

סיכום הפעולות שיש לבצע בכל מסגרת זמן :

1. מחשבים את הסידרה $\{R_i\} \quad i=1, \dots, L$

$$(3.27) \quad R_i = \frac{(\sum_n s_w(n)v_i(n))^2}{\sum_n v_i(n)^2}$$

2. מוצאים איבר מקסימלי שהאינדקס שלו הוא m .
3. מחשבים את גורם ההגבר המתאים - g_m מ 3.25.
4. משדרים את האינדקס m ואת גורם ההגבר לאחר שעבר קוונטיזציה.

סיבוכיות אלגוריתם החיפוש

הסיבוכיות הכרוכה בחיפוש הסידרה האופטימלית בדרך שהוצגה
 קודם היא $L(N^2+2N)$ פעולות כפל למסגרת. רב הפעולות - LN^2 ,
 דרושות לחישוב $\{v_i(n)\}_{i=1}^L$. ב-[18] הוצע לבצע את החיפוש
 בביטויים מתאימים מתחום התדר ולחסוך בסדר גודל של $N/\log_2(N)$
 במספר הכפלים.
 נסמן - $S_w(k), V_i(k), H_w(k), C_i(k)$ התמרות פוריה דיסקרטיות (DFT)
 של $s_w(n), v_i(n), h_w(n), c_i(n)$ בהתאמה, אזי הסידרה $\{R_i\}$ ניתנת
 לחישוב לפי:

$$(3.28) \quad R_i = \frac{(\sum_k S_w(k)V_i^*(k))^2}{\sum_k |V_i(k)|^2}$$

$$V_i(k) = H_w(k)C_i(k)$$

והסיבוכיות המתקבלת בגישה זו היא $2N \log_2(N) + 4LN$ פעולות כפל
 למסגרת. (כמפורט בנספח ב-f).

לדוגמא - עבור $L=1024, N=32$ סיבוכיות החיפוש בתחום הזמן היא
 $1.11 \cdot 10^6$ פעולות כפל למסגרת שהם $34.8 \cdot 10^3$ פעולות לדגם בעוד
 שלחיפוש בתחום התדר דרוש לבצע $131 \cdot 10^3$ פעולות כפל למסגרת שהם
 4106 פעולות לדגם, כלומר חיסכון ביחס 8:1.

תוצאות הסימולציות

בטבלא 3.1 מוצגות התוצאות שהתקבלו בהפעלת אלגוריתם ה-CELP על אותם המשפטים שהוזכרו קודם, עבור אורכים שונים של מסגרות אנליזה ומילון בעל 1024 סדרות קוד גאוסיות לבנות.

גודל מסגרת האנליזה	SEGSNR [dB]
16	17.7
32	12.6
64	8.1

טבלא 3.1 - SEGSNR כפונקציה של גודל מסגרת האנליזה
Table 3.1 - SEGSNR obtained for different frame sizes

פרק 4 - יישום רשת נוירונים לפתרון בעית ה-Multipulse

4.1 מבוא

בפרק הקודם הראינו שחיפוש מיקומים אופטימליים לפולסים במקודד MPE אינו מעשי כיוון שמספר הקומבינציות האפשרי גדול מאוד, ולכן השתמשנו באלגוריתם תת-אופטימלי - שהוצג ב-[12] ובפרק הקודם.

בדקנו לפיכך אוסף קומבינציות מוגבל בסביבת המיקומים שנבחרו על ידי האלגוריתם האיטרטיבי (עבור כל קומבינציה חושבה אנרגיית השגיאה). התוצאות הצביעו על כך שככל שגדל מספר הפולסים שהוקצו למסגרת, גדל מספר המקרים שנמצא סט מיקומים טוב יותר (גם הוא לא בהכרח אופטימלי כיוון שלא נבדקו כל הקומבינציות האפשריות). קימת אם כך אפשרות לשפר את הביצועים עם אותו מספר פולסים אם נדע לבחור אותם בצורה נכונה.

ב-[19] ו-[20] הוצעה שיטה לפתרון בעיות אופטימיזציה רבות משתנים בעזרת מחשב אנלוגי. מימוש המחשב נעשה עם רשת נוירונים.

רשת נוירונים היא מערכת שמקובל להציגה כמוזל חשמלי לצורה שבה פועל המוח האנושי בפתרון בעיות זיהוי וסיווג [21], אך ניתן ליישמה במגוון רחב של בעיות כדוגמת זו העומדת בפנינו.

בסעיף 4.2 נציג את הרשת ואת משוואות המצב שלה. בסעיף 4.3 נתאים את הבעיה של קביעת מיקומי הפולסים והאמפליטודות לדפוס שמכתיבה הרשת. בסעיף 4.4 נתאר את הדינמיקה של הרשת ונסכם בניתוח תוצאות הסימולציה.

4.2 יישום רשת נוירונים לפתרון בעיות אופטימיזציה

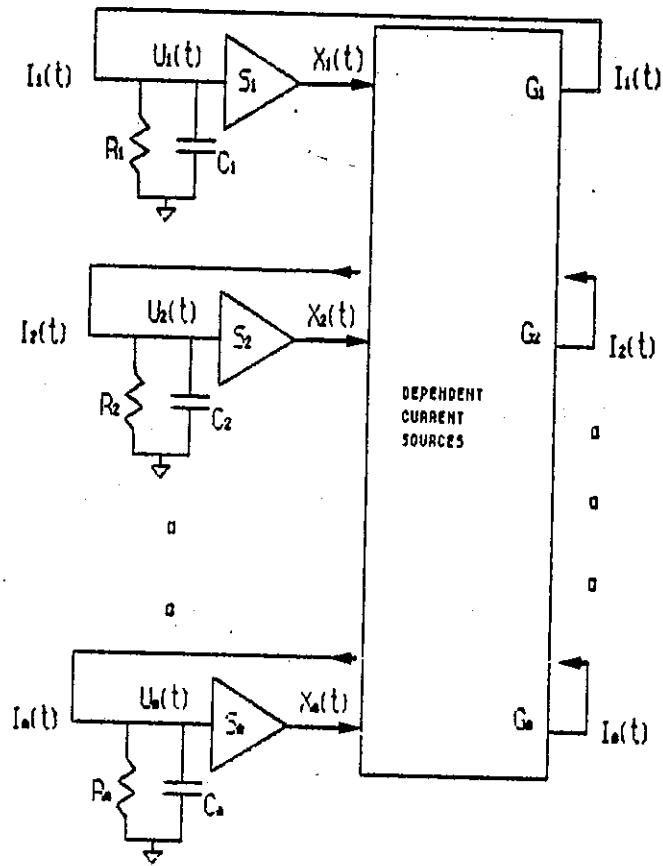
רשת נוירונים היא מערכת עיבוד אנלוגית המורכבת מיחידות עיבוד קטנות - הנוירונים, כמתואר בציור 4.1, עבור רשת של m נוירונים. לכל נוירון מתאים זרם כניסה - $I_i(t)$, מתח כניסה - $U_i(t)$, מתח יציאה - $X_i(t)$, נגד וקבל R_i, C_i ופונקציית תמסורת S_i :

$$(4.1) \quad X_i(t) = S_i(U_i(t))$$

את זרמי הכניסה מספקים מקורות זרם התלויים בוקטור מצב המוצאים \underline{X} -

$$(4.2) \quad \underline{X} = [X_1(t), \dots, X_m(t)]$$

בציור 4.1 מתוארת רשת של m נוירונים.



צירוף 4.1 - רשת נוירונים.
Fig. 4.1 - Neural Net.

משפט: וקטור המוצאים ברשת \underline{X} מתכנס מכל מצב התחלתי למינימום מקומי של פונקצית אנרגיה $E(\underline{X})$ מהצורה:

$$(4.3.1) \quad E(\underline{X}) = -F(\underline{X}) + \sum_{i=1}^m f_i(X_i)$$

$$(4.3.2) \quad f_i(X_i) = 1/R_i \int_0^{X_i} \bar{S}_i(u) du$$

$$(4.3.3) \quad \bar{S}_i(X_i) = \bar{S}_i^{-1}(X_i) = U_i$$

בתנאי שמתקיים:

1. E - גזירה וחסומה מלמטה.
2. \bar{S}_i עולה ממש, גזירה ובעלת פונקציה הפוכה \bar{S}_i^{-1} .
3. זרמי הכניסה לנוירונים הם הנגזרות החלקיות של $F(\underline{X})$:

$$(4.4) \quad \begin{aligned} I_i(t) &= G_i(t) \\ \underline{G} &= [G_1(t), \dots, G_m(t)] = [dF/dX_1, \dots, dF/dX_m] \end{aligned}$$

. (\underline{G} - הוא הגראדינט של $F(\underline{X})$)

הוכחה:

טענה : מספיק להוכיח כי נגזרת פונקציית האנרגיה לפי הזמן שלילית לכל X , כיוון שהמשמעות היא שוקטור המצב משתנה (בזמן) במרחב m מימדי בכוון בו התוספת לפונקציה היא שלילית, מתנאי 1 ברור שהמערכת תתכנס למצב מינימום מקומי של פונקציית האנרגיה.

$$\begin{aligned} dE/dt &= -e < 0, \quad e \geq 0 \\ \implies dE &= -e dt \end{aligned}$$

הוכחת הטענה :

מערכת המשואות הדיפרנציאליות שמיצגות את הרשת מצויר 4.1:

$$(4.5) \quad C_i dU_i/dt = I_i - U_i/R_i, \quad i=1, \dots, m$$

נגזור את E (ביטוי 4.3):

(4.6)

$$\begin{aligned} \frac{dE}{dt} &= \sum_{i=1}^N \frac{\partial E}{\partial X_i} \frac{dX_i}{dt} = - \sum_{i=1}^N \left[G_i - \frac{U_i}{R_i} \right] \frac{dX_i}{dt} = - \sum_{i=1}^N \left[I_i - \frac{U_i}{R_i} \right] \frac{dX_i}{dt} = \\ &= - \sum_{i=1}^N C_i \frac{dU_i}{dt} \frac{dX_i}{dt} = - \sum_{i=1}^N C_i \left[\frac{d}{dX_i} \bar{S}_i(X_i) \right] \left[\frac{dX_i}{dt} \right]^2 \leq 0 \end{aligned}$$

אי השוויון האחרון נכון על סמך תנאי 2.

מ.ש.ל.

נסכם את הסעיף ברישום הפעולות שיש לבצע כדי לפתור בעיה אופטימיזציה בעזרת הרשת:

1. רישום הבעיה לפי הדפוס שמכתיבה הרשת בביטוי 4.3 ומציאת משוואות זרמי הכניסה מ 4.4.
2. לקבוע וקטור מצב התחלתי X .
3. להפעיל פיתרון נומרי למערכת המשואות הדיפרנציאליות המוצגת בביטוי 4.5, עם תנאי עצירה כשמזהים התכנסות. התכנסות תיבדק לפי השינוי היחסי בפונקציית האנרגיה.

הערה : הנגדים והקבלים קובעים את קצב ההתכנסות למצב יציב במימוש אמיתי אך בפיתרון בעזרת סימולציה הגדלים האמיתיים חסרי חשיבות ונבחרו (כך ש- $T=C_i R_i=1 \text{ sec}$):

$$R_i = 1 \text{ ohm} \quad C_i = 1 \text{ Farad} \quad i=1, \dots, m$$

4.3 פתרון בעית עירור ה-Multi-Pulse בעזרת רשת נוירונים

בפרק הקודם הגדרנו את בעית עירור ה-Multi-Pulse כמצייאת K פולסים כך שאנרגיית השגיאה המשוקללת תהיה מינימלית:

$$(4.7) \quad E_w = \sum_{n=0}^{N-1} (s_w(n) - \sum_{i=1}^K b_i h_w(n-m_i))^2 =$$

$$= \sum_{n=0}^{N-1} s_w(n)^2 - 2 \sum_{i=1}^K b_i \alpha(m_i) + \sum_{i=1}^K \sum_{j=1}^K b_i b_j \phi(m_i, m_j)$$

כאשר:
 $s_w(n)$ - אות הדיבור המשוקלל לאחר שהופחתה תגובת הזכרון.
 $h_w(n)$ - התגובה להלם של מסנן ה-LPC המשוקלל.
 b_i, m_i - מיקומים ואמפליטודות של פולסי העירור, בהתאמה.

$$\alpha(i) = \sum_{n=0}^{N-1} s_w(n) h_w(n-i)$$

$$\phi(i, j) = \sum_{n=0}^{N-1} h_w(n-i) h_w(n-j)$$

נגדיר: $\underline{X} = [b_1, \dots, b_K, m_1, \dots, m_K]$ וקטור מצב:
 $E = \sum_{i=1}^K \sum_{j=1}^K b_i b_j \phi(m_i, m_j) - 2 \sum_{i=1}^K b_i \alpha(m_i)$ ופונקציית אנרגיה חדשה:

המקימת את אותם מצבי מינימום אנרגיה של E_w . נחאים את E לדפוס של הרשת (משוואה 4.3) על ידי הוספת והחסרת ביטוי:

$$(4.8) \quad E = \sum_{i=1}^K \sum_{j=1}^K b_i b_j \phi(m_i, m_j) - 2 \sum_{i=1}^K b_i \alpha(m_i) + \sum_{i=1}^K (m_i^2 + b_i^2) - \sum_{i=1}^K (m_i^2 + b_i^2)$$

נתאים את הביטויים:

$$(4.9) \quad F(\underline{X}) = - \sum_{i=1}^K \sum_{j=1}^K b_i b_j \phi(m_i, m_j) - 2 \sum_{i=1}^K b_i \alpha(m_i) + \sum_{i=1}^K (m_i^2 + b_i^2) \quad .1$$

$$(4.10) \quad f_i(x_i) = x_i^2 \quad .2$$

לפי 4.3.2 מתקבל: $\bar{S}_i(u) = 2u$ ולפי 4.3.3 מתקבל:

$$(4.11) \quad x_i = \bar{S}_i(u_i) = 0.5u_i$$

מתחי הכניסה לנוירונים משתנים לפי (מתוך 4.11):

$$(4.12) \quad dU_i/dt = dU_i/dX_i \cdot dX_i/dt = 2dX_i/dt$$

נציב את הביטויים במערכת המשוואות שמיצגת את הרשת (ביטוי 4.5) ונקבל:

$$(4.13) \quad dX_i/dt = 0.5[I_i - 2X_i] \quad i=1, \dots, 2K$$

את הזרמים נקבל על ידי גזירת $F(\underline{X})$ לפי המשתנים המתאימים.

הזרמים הקשורים באמפליטודות:

$$(4.14) \quad I_j = -2 \sum_{i=1}^K b_i \phi(m_i, m_j) + 2\alpha(m_j) + 2b_j, \quad j=1, \dots, K$$

הזרמים הקשורים למיקומים:

$$(4.15)$$

$$I_j = -2 \sum_{i=1}^K b_i b_j \frac{\partial \phi(x, m_i)}{\partial x} \Big|_{x=m_j} + 2b_j \frac{\partial \alpha(x)}{\partial x} \Big|_{x=m_j}, \quad j=1 \dots K$$

נרשום מפורשות את מערכת המשוואות הדיפרנציאליות שיש לפתור (מתקבל על ידי הצבת הביטויים לזרמים ב-4.13):

$$(4.15)$$

$$db_j/dt = - \sum_{i=1}^K b_i \phi(m_i, m_j) + \alpha(m_j) \quad j=1, \dots, K$$

$$dm_j/dt = -2 \sum_{i=1}^K b_i b_j \frac{\partial \phi(x, m_i)}{\partial x} \Big|_{x=m_j} + 2b_j \frac{\partial \alpha(x)}{\partial x} \Big|_{x=m_j}, \quad j=1, \dots, K$$

בכדי לפתור את המערכת יש להתמודד עם שתי בעיות :

1. הפונקציות $\alpha(i), \phi(i, j)$ הן דגימה של פונקציות אוטוקורלציה וקרוסקורלציה בהתאמה והוגדרו עבור זמנים שהם כפולות שלמות של זמן הדגימה - T . בכדי לפתור את מערכת המשוואות נדרש להגדיר את ערכי הפונקציות ונגזרותם לכל t בתחום $0 \leq t \leq (N-1)T$.

2. המצב היציב אליו תתכנס הרשת יתקבל עם מיקומים שאינם בהכרח כפולה שלמה של זמן מחזור דגימה ולכן יש להגדיר מיפוי שיתאים ערכים שלמים למיקומים לכל ערך רציף שיתקבל מהרשת.

כדי להתגבר על הבעיה הראשונה בוצעה אנטרפולציה פולינומאלית מסדרים שונים. לפי גישה זו מתאימים פולינום מסדר M - $y=f(x)$ כך שיעבור דרך סט נתון של נקודות $(x_1, y_1), \dots, (x_M, y_M)$ ומקבלים את הערך המתאים ל- x כלשהוא על ידי הצבתו בפולינום [22]. הסדרים שנבדקו: $M=2, 3, 4$.

לפתרון הבעיה השניה נוסו שתי גישות. הראשונה - העגלה למספר השלם הקרוב ביותר, והשניה העגלה לכיוון מנוגד לגראדינט. כלומר חושב הגראדינט של פונקציית האנרגיה וההעגלה נעשתה לפי:

$$(4.16) \quad m_i = \begin{cases} \lfloor m_i \rfloor & \text{if } dE/dX_i \geq 0 \\ \lceil m_i \rceil & \text{if } dE/dX_i < 0 \end{cases}$$

1- נגזרת מסמנים העגלה לשלם הקרוב: הקטן או הגדול, בהתאמה.

המניע לפעולה זו הוא שהגרדינט מכוון לכוון שבו הפונקציה עולה עם שיפוע מקסימלי והנחה שהתקדמות בכיוון מנוגד לכוון זה תוביל לערך קטן יותר.

פיתרון נומרי למערכת המשוואות הדיפרנציאליות נעשה לפי אלגוריתם Runga Kuta עם תיקון שגיאה [22]. תנאי ההתחלה שסופקו למערכת היו המיקומים והאמפליטודות שהתקבלו על ידי אלגוריתם ה-MPE, בתקוה שזהו מצב קרוב למינימום והמערכת תתכנס במהירות.

בכל פרק זמן נבדק השינוי היחסי בפונקציה האנרגיה כדי לזהות מצב יציב. לאחר זיהוי ההתכנסות והעגלת המיקומים, בוצעה רה-אופטימיזציה לאמפליטודות וחושבה אנרגיית השגיאה החדשה כדי להשוותה לאנרגיית השגיאה ההתחלתית שהתקבלה עבור הפיתרון התת-אופטימלי של אלגוריתם ה-MPE.

4.4 הסבר לדינמיקה של הרשת

בסעיף זה נראה שמערכת המשוואות שהתקבלה מיצגת מימוש אנלוגי לטכניקה נומרית ידועה למציאת אקסטרמום לפונקציות והיא שיטת הגרדינט (Steepest Descent או Gradient Method) [23].

זוהי שיטה המוגדרת על ידי אלגוריתם איטרטיבי המתכנס למינימום (או מקסימום) של פונקציה גזירה וחסומה $E(\underline{X})$. הרעיון העומד מאחוריה הוא שבכל שלב מתקדמים למצב חדש שבו הפונקציה E מקבלת ערך נמוך יותר. כיוון ההתקדמות נעשה בכיוון מנוגד לגרדינט וגודל הצעד נקבע לפי קבוע פסיעה. את קבוע הפסיעה α מקטינים לפי הקרבה למינימום.

שלבי האלגוריתם:

0. קובעים \underline{X}_0 - וקטור מצב התחלתי, $k=0$.

1. מוצאים את וקטור המצב הבא לפי: $\underline{X}_{k+1} = \underline{X}_k - \alpha_k \underline{G}_k$

כאשר: \underline{G}_k הוא גרדינט הפונקציה במצב \underline{X}_k - $\underline{G}_k = [dE/dx_{1k}, \dots, dE/dx_{mk}]$ הוא קבוע הפסיעה בצעד ה-k.

2. אם $(E(\underline{X}_k) - E(\underline{X}_{k+1})) / E(\underline{X}_k) \geq \epsilon$ חוזר ל 1.

3. סוף.

ניתן לראות בקלות שמערכת המשוואות הדיפרנציאליות שיש לפתור (ביטוי 4.15) יכולה להרשם באופן הבא:

$$(4.17) \quad dx_i/dt = -0.5dE/dx_i, \quad i=1, \dots, 2K$$

ופונקציית האנרגיה E - מביטוי 4.8. כלומר, צעד הפסיעה האקוילנטי הוא: $\alpha_{\text{eff}} = 0.5dt$. לפי סף דיוק שנקבע על ידי המשתמש באלגוריתם Runga Kuta עם תיקון שגיאה [22].

4.5 תוצאות הסימולציה

השיטה שהוצגה לא הצליחה להשיג שיפור לפתרון המתקבל על ידי אלגוריתם ה-MPE. למרות שתוך כדי התהליך התקבלו פתרונות עבורם אנרגיית השגיאה היתה קטנה מהאנרגיה ההתחלתית, הרי שאחרי הפעלת ההעגלה (שתי הטכניקות שהוצגו) והאופטימיזציה לאמפליטודות, התקבל פתרון עבורו אנרגיית השגיאה עלתה בדרך כלל על האנרגיה ההתחלתית.

הרשת הופעלה גם במקרה של פולס בודד בכל מסגרת זמן. יחודו של מקרה זה בכך שפתרון אלגוריתם ה-MPE במקרה זה הוא אופטימלי. גם בנסיון זה נכשלה הרשת במרבית המקרים. התכנסות מוצא הנוירון שהוגדר למיקום היתה לערך קרוב למיקום האופטימלי אך ההעגלה היתה שגויה. הגדלת מספר הפולסים רק החמירה את בעיה זו.

מתוך התוצאות הללו מסתבר שבעית חיפוש הפולסים האופטימליים אינה ניתנת לפיתרון נומרי בגישה המסוימת הזאת.

פרק 5 - עדכון תנאי התחלה למסנן הסינטזה

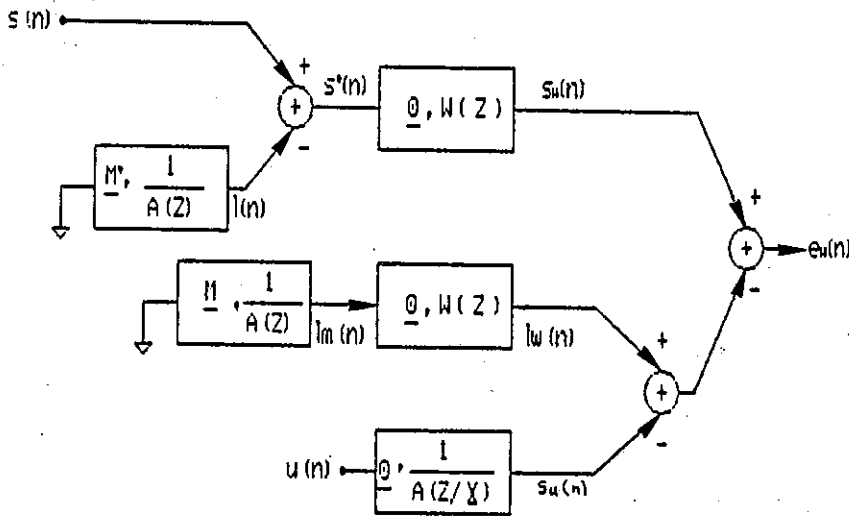
5.1 מברא

האות המשוחזר במקודד ה-MPE מורכב מהתגובה לעירור- $u(n)$ (Zero State Response) והתגובה לתנאי התחלה- M' (Zero Input Response) שנוצרו בסיום האנליזה של המסגרת הקודמת.

בפרק זה נבחן את הפוטנציאל הטמון בקביעת וקטור תנאי התחלה M - למסנן הסינטזה, כאמצעי נוסף לשחזור אות המקור כפי שמודגם בצירוף 5.1. את אות המקור המשוקלל (לאחר הפחתת תגובת הזכרון של מסנן הסינטזה) ננסה לשחזר כסכום של:

1. תגובה לעירור $u(n)$ שנשמנה ב- $s_u(n)$.
2. תגובה לתנאי ההתחלה M שנשמנה ב- $l_m(n)$.

שני מרכיבים אלו יחושבו ויקודדו בכל מסגרת זמן, ובערוץ ישודר מידע על העירור, על וקטור הזכרון ועל מקדמי החזאי.



צירוף 5.1 - עדכון תנאי התחלה למסנן הסינטזה.

Fig. 5.1 - Synthesis Filter Initial Condition updating.

בסעיף 5.2 נציג את משוואות עדכון הזכרון כך שתגובת המסננת לתנאי התחלה אלו תהיה הקרובה ביותר לאות הדיבור תחת קריטריון שגיאה ריבועית משוקללת.

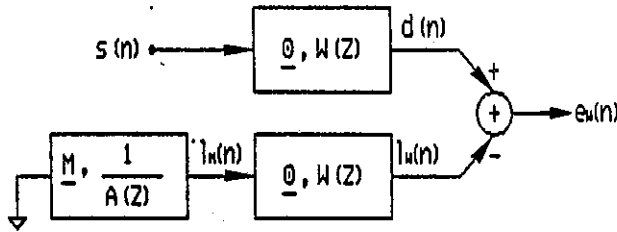
בסעיף 5.3 נשלב את סכמת העדכון בסכמת ה-MPE ונציג שתי אפשרויות לשילוב, הנובעות מהפיתרון האיטרטיבי של בעיית העירור.

5.2 פיתוח סט המשואות לתנאי התחלה אופטימליים

בציור 5.2 מתואר מודל סכמתי לבעיה של חיפוש תנאי התחלה אופטימליים: $M=[M(-1), \dots, M(-P)]$, כך שסך השגיאה הריבועית המשוקללת בין אות המקור $s(n)$ ותגובת הזכרון $lw(n)$ תהיה מינימלית. שקלול השגיאה מבוצע בעזרת המסנן $w(z)$ עם זכרון מאופס. מטעמי נוחות שולב המסנן המשקלל קודם למסכם.

נסמן:

$lw(n)$ - תגובת הזכרון המשוקללת.
 $d(n)$ - האות שאותו מבקשים לקדם.



ציור 5.2 - מודל סכמתי לבעיה עדכון תנאי התחלה.

Fig. 5.2 - Block diagram of update initial condition problem.

דרוש מינימום על סך השגיאה הריבועית המשוקללת:

$$(5.1) \quad E = \sum_n (lw(n) - d(n))^2, \quad n=0, \dots, N-1$$

מטרת תת-הטעיף הבא היא להציג את תגובת הזכרון המשוקללת בצורה הבאה:

$$(5.2) \quad lw(n) = \sum_{i=1}^P fn(i)M(-i) \quad n=0, \dots, N-1$$

כלומר להציג כל דגם בתגובה כקומבינציה לינארית של דגמי הזכרון כיוון שאז ניתן לגזור מפורשות ולהשוות לאפס. סט המשואות שנצטרך אז לפתור כדי לקבל את הזכרון האופטימלי הוא:

$$(5.3) \quad \sum_n d(n)fn(j) = \sum_{i=1}^P M(-i) \sum_n fn(i)fn(j) \quad j=1, \dots, P$$

5.2.1 קבלת תגובת הזכרון כפונקציה של דגמי הזכרון

a. תגובת הזכרון מתקבלת ממשוואות ההפרש :

$$(5.4) \quad \sum_{i=1}^p a_i l_m(n-i) = 0, \quad a_0 = 1$$

$$l_m(-1) = M(-1); \dots; l_m(-P) = M(-P)$$

נבצע התמרת Z למשוואת ההפרש ונחלץ את $L_m(Z)$, $L_m(Z) = Z\{l_m(n)\}$

$$(5.5) \quad L_m(Z) = \frac{X(Z)}{A(Z)}$$

$$X(Z) = \sum_{j=0}^{p-1} Z^{-j} \left[\sum_{t=-1}^{p-j} M(-t) a_{j+t} \right]$$

$$A(Z) = 1 + \sum_{i=1}^p a_i Z^{-i}$$

$L_w(Z)$ מתקבל על ידי הכפלה בפונקציה התמסורת של המסנן המשקלל:

$$(5.6) \quad L_w(Z) = \frac{X(Z)}{A(Z/\Gamma)}$$

b. את $L_w(Z)$ נציג כמוצא מסנן $R(Z)$ (עם זכרון מאופס) לכניסה $X(Z)$:

$$(5.7) \quad R(Z) = 1/A(Z/\Gamma)$$

c. התמרה הפוכה ל $X(Z)$:

$$(5.8) \quad x(n) = \begin{cases} \sum_{t=-1}^{p-n} M(-t) a_{n+t} & n=0, \dots, P-1 \\ 0 & P \leq n \leq N-1 \end{cases}$$

נגדיר: $\{v_+(n)\}$ - מקדמי היצוג של דגמי הזכרון בדגם אות הכניסה $x(n)$.

$$(5.9) \quad v_+(n) = \begin{cases} a_{n+t} & 0 \leq n \leq P-1, 1 \leq t \leq P-n \\ 0 & 0 \leq n \leq P-1, P-n+1 \leq t \leq P \\ 0 & P \leq n \leq N-1, 1 \leq t \leq P \end{cases}$$

ואז משוואה 5.8 ניתנת לרישום שקול על ידי:

$$(5.10) \quad x(n) = \sum_{t=-1}^p M(-t) v_+(n) \quad n=0, \dots, N-1$$

d. נתאר את $R(Z)$ בעזרת מערכת משוואות מצב:

$$(5.11) \quad \begin{aligned} q(n+1) &= Aq(n) + Bx(n) \\ l_w(n) &= Cq(n) + dx(n) \end{aligned}$$

כאשר :

$$A = \begin{bmatrix} 0 & 1 & 0 & \dots & \dots & 0 \\ 0 & 0 & 1 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & \dots & \dots & 0 & 1 \\ -a_1 \Gamma^1 & \dots & \dots & \dots & -a_p \Gamma^p & 0 \end{bmatrix}_{p \times p}$$

$$B^T = [0 \dots 1]_{p \times 1}$$

$$C = [a_1 \Gamma^1 \dots a_p \Gamma^p]_{1 \times p}$$

$$d = 1$$

ו- $q(n)$ וקטור מצב ברגע ה- n במימוש קנוני של המסנן. ניתן לקבל את הפיתרון הכללי כסכום של שני אברים

$$(5.12) \quad l w(n) = CA^n q(0) + \left[\sum_{k=0}^{n-1} CA^{n-1-k} B x(k) + x(n) \right]$$

האבר הראשון מיצג את התרומה של התגובה לתנאי ההתחלה $q(0)$ והאבר השני מיצג את התגובה לכניסה $x(n)$. כפי שצויין ב- b . זכרון המסנן $R(Z)$ מאופס ולכן האבר הראשון מתאפס.

נציב את $x(n)$ מנוסחא 5.10 בביטוי 5.12 ולאחר סידור אברים מתקבל :

$$(5.13) \quad l w(n) = \sum_{i=1}^p M(-i) [v_i(n) + \sum_{k=0}^{n-1} v_k(n) CA^{n-1-k} B]$$

ולסיכום נחלץ את מקדמי היצוג $f_n(i)$ (ראה (5.2)) :

$$(5.14) \quad f_n(i) = v_i(n) + \sum_{k=0}^{n-1} v_k(n) CA^{n-1-k} B$$

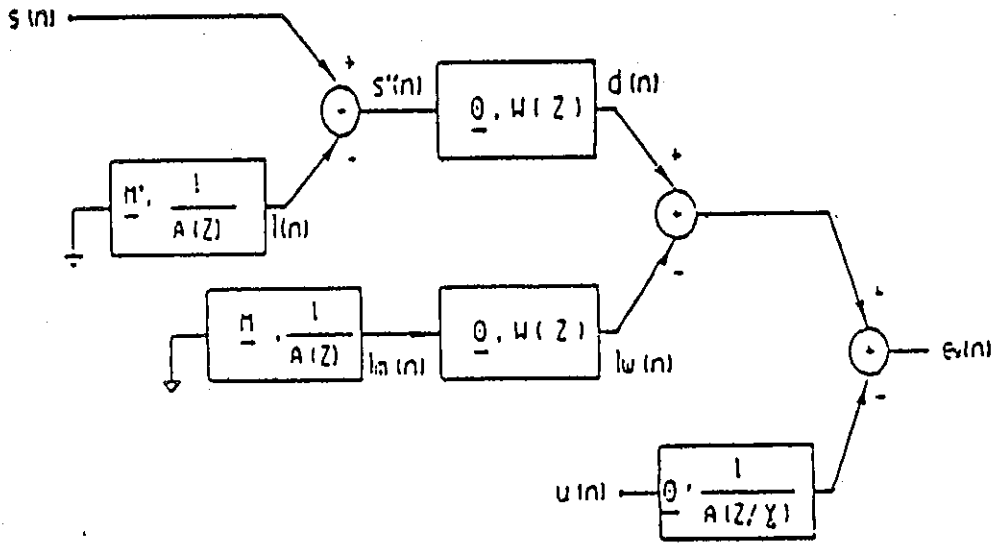
5.3 שילוב סכמת עדכון הזכרון בסכמת עירור Multi-Pulse

נגדיר את הבעיה הכללית המשלבת חיפוש של עירור (א-פולסים) ותנאי התחלה אופטימליים תחת קריטריון שגיאה ריבועית משוקללת:

$$(5.15) \quad E = \sum_n (s_w(n) - \sum_{i=1}^K b_i h_w(n-m_i) - \sum_{j=1}^p f_n(j) M(-j))^2$$

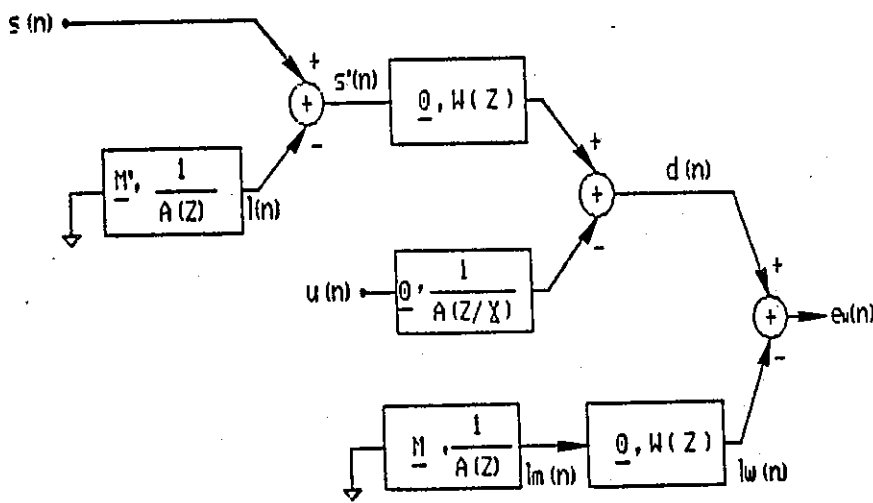
פיתרון גלובלי אינו אפשרי כיוון שהבעיה אינה לינארית אם אין יודעים את המיקומים. אולם, ניתן לשלב את העדכון באחת משתי צורות, הנותנות תוצאות שונות, ושמנצלות את האלגוריתם האיטרטיבי למציאת פולסים שהוצג בפרק 3. המשותף לשתי הסכמות הוא שקביעת הערור נעשה בנפרד לעדכון :

1. עדכון תנאי ההתחלה לפני מציאת העירור כפי שמודגם בציור 5.3.



ציור 5.3 - עדכון תנאי ההתחלה קודם למציאת העירור.
 Fig. 5.3 - Update initial condition before locating the pulses.

2. עדכון תנאי ההתחלה אחרי מציאת העירור כפי שמודגם בציור 5.4.



ציור 5.4 - עדכון תנאי ההתחלה אחרי מציאת העירור.
 Fig. 5.4 - Update initial condition after locating the pulses.

האלגוריתם המתאים לסכמה בציר 5.3:

1. מחשבים את אות המקור המשוקלל, לאחר שהופחתה ממנו תגובת הזכרון שנוצר בסיום הסינטזה של מסגרת הזמן הקודמת:

$$s_w(n) = (s(n) - l(n)) * w(n)$$

2. מחשבים את תנאי ההתחלה האופטימליים אשר יביאו למינימום את אנרגיית הפרש בין התגובה - $lw(n)$ והאות - $s_w(n)$ על ידי פתרון מערכת משוואות לינאריות 5.3.
3. בעזרת האלגוריתם האיטרטיבי מפרק 3 מוצאים K פולסים בעירור $u(n)$ שיביאו למינימום את אנרגיית הפרש בין התגובה $lw(n)$ ו $s_w(n)$.

האלגוריתם המתאים לסכמה בציר 5.4:

1. מחשבים את אות המקור המשוקלל, לאחר שהופחתה ממנו תגובת הזכרון שנוצר בסיום הסינטזה של מסגרת הזמן הקודמת:

$$s_w(n) = (s(n) - l(n)) * w(n)$$

2. בעזרת האלגוריתם האיטרטיבי מפרק 3 מוצאים K פולסים בעירור $u(n)$ שיביאו למינימום את אנרגיית הפרש בין התגובה לעירור ו $s_w(n)$.
3. מחשבים את תנאי ההתחלה האופטימליים אשר יביאו למינימום את אנרגיית הפרש בין התגובה - $lw(n)$ ואות השגיאה המתקבל בסיום השלב השני.

בשני המקרים האות המשוחזר מתקבל על ידי העברת אות העירור דרך מסנן הסינטזה עם זכרון המתקבל מסיכום M ו M' .

במקרה הראשון נראה לכאורה כאילו ניתן לותר על התרומה של התגובה לתנאי ההתחלה שקימים במקלט - M' (כלומר לאפס אותם) ולמצא את וקטור הזכרון האופטימלי עבור אות הדיבור המקורי המשוקלל. התוצאות תהינה זהות לחלוטין לאלו שיתקבלו ללא האיפוס, כיוון שוקטור הזכרון האופטימלי יתקן חוסר ההתאמה בין התגובה $l(n)$ לאות. בצורה דומה ניתן היה לחשוב על איפוס וקטור הזכרון גם בסכמה השנייה.

אולם תגובת הזכרון אשר ידועה ממילא למקלט, מנוצלת במקרה זה להקטנת האנרגיה של וקטור הזכרון האופטימלי ושל הדיבור שיש לקודד (בדומה להפחתת תגובת הזכרון מאות הדיבור המקורי באלגוריתם ה-MPE). כלומר הזכרון M' הוא משערך אחורי לוקטור הזכרון האופטימלי (או לחילופין $l(n)$ הוא משערך אחורי לדיבור המקורי) ולכן מופחת ממנו. משערך אחורי יוסבר בפרק 6 אך נצין כי זהו משערך המבוטס על מידע שקיים במשדר ובמקלט ולכן אין צורך בהקצאת סיביות לשידורו.

בטרם נציג את תוצאות הסימולציות נטפל בשתי בעיות:

1. מהו גודל המסגרת המתאים לאנליזה עם עדכון תנאי התחלה.
2. מי משתי הסכמות עדיפה.

גודל מסגרת האנליזה קובע את קצב העדכון. משיקולים של הקצאת סיביות לתנאי ההתחלה האופטימליים, רצוי היה שמסגרת האנליזה תהיה גדולה ככל האפשר (גודל המסגרת מוגבל מלמעלה על ידי תחום הזמן שבו אות הדיבור סטאציונרי - כ. 20 msec). מצד שני ככל שנקטין את מסגרת האנליזה תגדל היכולת של תגובת תנאי ההתחלה האופטימליים לשחזר את אות המקור.

ניתן לקבוע גודל שיקח בחשבון את תחום ההשפעה של התגובה לפי השיקול הבא: אורך תגובת ההלם $h_w(n)$ הוא כ 20 דגמים (סעיף 3.4.3), התגובה לתנאי התחלה שקולה לתגובה לכניסה $x(n)$ $n=1, \dots, P=10$ (נוסחא 5.5) ולכן אורך תגובת המסננת לתנאי ההתחלה הוא כ 30 דגמים (- סכום האורכים).
בחרנו לכן את גודל המסגרת להיות $N=32$.

ראינו בסעיף הקודם שהתגובה לתנאי ההתחלה שקולה לתגובה לכניסה $x(n)$ באורך P . היות ותנאי ההתחלה נקבעים להיות אופטימליים, הרי שהבעיה השקולה היא למצוא את האמפליטודות האופטימליות של P פולסים שממוקמים ב P המקומות הראשונים במסגרת (השקילות נובעת מכך שהפתרון האופטימלי לבעיה זו הוא יחיד). כלומר לפתור את המערכת:

$$\alpha(m_j) = \sum_{i=1}^P b_i \phi(m_j, m_i) \quad j=1, \dots, P$$

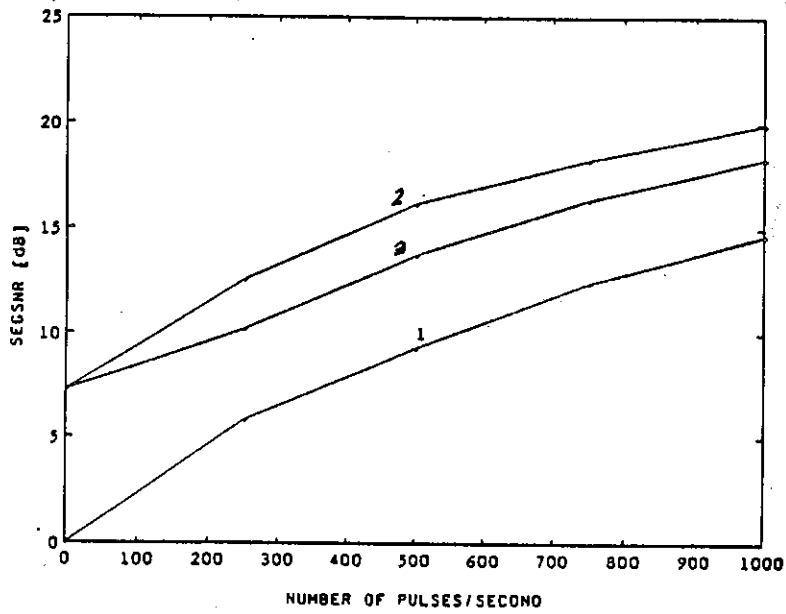
(5.16)

עם אילוץ: $m_j = j \quad j=0, \dots, P-1$

המסקנה היא שאם מבצעים את עדכון הזכרון אחרי שכבר נקבעו הפולסים, בעזרת האלגוריתם האיטרטיבי, יתכן מצב שמספר פולסים נבחרו בתחום P המיקומים הראשונים ואז תגובת הזכרון האופטימלית תהיה שקולה לתגובה לכניסה שבה הדגמים באותם מקומות יהיו אפס.

אם לעומת זאת מבצעים את העדכון לפני שנקבעו הפולסים, אזי הפולסים ימוקמו מחוץ לתחום P המיקומים הראשונים (עבורם הערוך השקול - $x(n)$ אופטימלי) ואז ניתן להשיג תוצאות טובות יותר.

בציור 5.5 מתואר ה- SEGSR כפונקציה של מספר הפולסים בעירור עבור אנליזת MPE בשיטת הקוריאנס עם וברי עדכון זכרון. בכל המקרים בוצעה רה-אופטימיזציה לאמפליטודות בסיום כל איטרציה. גודל מסגרת האנליזה הוא $N=32$ ו $\Gamma=0.8$.



ציור 5.5 - SEGSNR המתקבל עם ובלי עדכון זכרון. 1. ללא עדכון עם עדכון לפני שנמצאים הפולסים 3. עם עדכון אחרי שנמצאים הפולסים.

Fig. 5.5 - SEGSNR obtained with and without vector memory update 1. without update 2. update before locating pulses 3. update after locating pulses.

מתקבל שיפור ניכר, בשתי צורות העדכון בהשוואה לאלגוריתם MPE המקורי, וכצפוי, הביצועים של אלגוריתם העדכון קודם לקביעת הפולסים עולים על האלגוריתם השני.

5.3.1 רה-אופטימיזציה לתנאי ההתחלה ולאמפליטודות

ניתן להוסיף שלב רביעי בשני האלגוריתמים שהוצגו ובו תבוצע רה-אופטימיזציה לאמפליטודות ולתנאי ההתחלה לאחר שנמצאו מיקומי הפולסים. סט המשואות הלינאריות שיש לפתור מתקבל על ידי גזירת E מביטוי 5.15 לפי האמפליטודות ודגמי הזכרון והשוואה לאפס:

(5.17)

$$\alpha(m_j) = \sum_{l=1}^K b_l \phi(m_l, m_j) + \sum_{m=1}^P M(-m) \sum_n f_n(m) h_w(n-m_j) \quad j=1, \dots, K$$

$$\sum_n s_w(n) f_n(l) = \sum_{l=1}^K b_l \sum_n h_w(n-m_l) f_n(l) + \sum_{m=1}^P M(-m) \sum_n f_n(m) f_n(l) \quad l=1, \dots, P$$

במקרה שעדכון וקטור הזכרון מבוצע אחרי חיפוש הפולסים אין צורך לבצע את השלב השלישי (מציאת וקטור זכרון) אלא ניתן לפתור ישירות את מערכת המשוואות הנ"ל.

בבדיקה שערכנו התקבל כי השיפור ב-SNR היה מזערי ולכן לא נעשה בהמשך שימוש בפעולה זו, מה עוד שהסיבוכיות הכרוכה בהיפוך מטריצה מסדר $P+K$ גבוהה במיוחד - $O((P+K)^3)$.

סיכום

שאלת הכדאיות של עדכון וקטור הזכרון תיבחן על ידי השואת הקצאת הסיביות לוקטור הזכרון עם זו הנחוצה לקידוד פולסים. בפרק הבא נתאר את התוצאות שהתקבלו במערכת המשלבת עדכון וקידוד וקטורי של וקטור תנאי ההתחלה.

בטרם נמשיך נציין כי רעיון דומה נבדק ב-[24]. במאמר זה הוצגו תוצאות ראשוניות של ביצועי סכמת ה-MPE עם עדכון תנאי ההתחלה למסנן הסינטזה וגורם השקלול - $\Gamma=0$, כלומר לא מבוצעת צביעה של אות השגיאה.

ניתן להראות כי פיתרון בעית ה-MPE המקורית (ללא עדכון תנאי ההתחלה), עם בחירה זו של גורם השקלול הוא, שהפולסים האופטימליים הם השיאים באות השארית [25] ואין צורך להפעיל את האלגוריתם האיטרטיבי.

במאמר נבדקו ביצועי אלגוריתם עדכון וקטור הזכרון (ללא קידוד) על ידי פתרון מערכת המשוואות 5.17 כאשר מיקומי הפולסים נקבעו על סמך השיאים באות השארית. בדיקות אלו מתאימות לאלגוריתם עדכון וקטור הזכרון אחרי מציאת הפולסים, שהוצג בסעיף הקודם, עם $\Gamma=0$ וזה המקרה הגרוע יותר כי עדיף לבצע עדכון זכרון תחילה.

כיוון שביצועי מערכת ה-MPE עם $\Gamma=0$ הינם גרועים, לא מצאנו טעם לבצע בדיקות מקבילות ובכל התוצאות שיוצגו בהמשך נקבע $\Gamma=0.8$.

פרק 6 - קוונטיזציה וקטורית לוקטור תנאי ההתחלה

6.1 מבוא

בפרק זה נציג את התוצאות שהתקבלו כששולבה טכניקת קידוד וקטורית לתנאי ההתחלה שהוצגו בפרק 5. בכל מסגרת זמן של 4 msec (32 דגמים) מקודדים תנאי התחלה אופטימליים למסנן הסינטזה ואחר כך קובעים K פולסים נוספים לעירור המסנן.

קידוד סקלרי לא נבחן כיוון שמטרת העבודה היא לשפר את ביצועי המקודד בקצב בינים - 9.6 Kbps . עובדה זו מחייבת הקצאת סיביות מינימלית לקידוד תנאי ההתחלה (לכל היותר סיבית לדגם זכרון), בתחום זה קידוד סקלרי אינו יעיל.

בסעיף הבא נתאר את הקוונטיזר הוקטורי ואת אלגוריתם LBG המשמש לתכנון מילון מתוך סדרת לימוד. בהמשך נגדיר את הבעיה הכללית של תכנון מילון עם קריטריון הטיב שלפיו נבחנום ביצועיו של המקודד - SNR, ונראה כי לבעיה זו אין פיתרון פשוט. במקום המילון האופטימלי נציע מילונים אחרים המקיימים שבגבול, כאשר $L \rightarrow \infty$ (מילונים אינסופיים) הם מתלכדים עם הקוונטיזר האופטימלי.

נבדוק אם אכן ניתן להשיג שיפור באיכות ללא תוספת בקצב ומצד שני שהתוספת בסיבוכיות לא תהיה גבוהה. השיפור באיכות ובקצב יעשה על ידי השואת ביצועי המערכת הכוללת קידוד וקטור הזכרון לאלו המתקבלים במערכת עם פולסים בלבד ובאותו הקצב.

בסיום הפרק נציג את המגבלות של טכניקת קידוד תנאי התחלה ונסיק מסקנות להמשך העבודה.

6.2 קוונטיזציה וקטורית

קוונטיזציה היא פעולה שמתרגמת משתנה רציף באמפליטודה למספר קטן של רמות. רמות אלו שמורות במילונים המצויים במשדר ובמקלט ובערוץ משודר אינדקס הרמה שנבחרה. הקוונטיזציה יכולה להיות וקטורית כשהיא פועלת על מספר דגמים בו זמנית (וקטור במימד N) או סקלרית ואז כל דגם מיוצג בנפרד. המניע לשימוש בקוונטיזר וקטורי הוא העובדה שניתן להתקרב בעזרתו לפונקציית קצב-עיוות $R(D)$ של המקור [26]. פונקציה זו מהווה חסם תחתון לקצב בהינתן העיוות. במשפט קידוד מקורות מתורת האנפומציה מוכח כי על ידי הגדלת אורך הוקטור N ניתן להתקרב לחסם (מלמעלה) כרצוננו. ביצועי הקוונטיזר הוקטורי אמנם עולים על הסקלרי אך המחיר הוא בגידול ניכר בסיבוכיות מערכת הקידוד.

6.2.1 הגדרת קוונטיזר וקטורי (VQ)

קוונטיזר וקטורי מוגדר על ידי המיפוי:

$$Q: X \rightarrow Y, \quad X \in \mathbb{R}^M$$

(6.1)

$$Q(\underline{x}) = \underline{y}_i \quad \text{if } \underline{x} \in S_i$$

כאשר $\underline{x}, \underline{y}_i$ הם וקטורי הכניסה והיציאה בהתאמה, $\{S_i\}_{i=1}^L$ היא חלוקה של המרחב בו מוגדרים רכיבי וקטור הכניסה ו $Y = \{\underline{y}_i\}$ היא קבוצה של L וקטורים במימד N הקרויה מילון או ספר-קוד.

נגדיר מדד עיוות בין הכניסה והיציאה של הקוונטיזר ע"י:
 $d(\underline{x}, Q(\underline{x}))$. המקרה המקובל, בו אנו משתמשים בעבודה זו, הוא העיוות הריבועי:

$$(6.2) \quad d(\underline{x}, Q(\underline{x})) = \sum_{i=1}^N (x_i - y_i)^2$$

מקובל למדוד את הביצועים של קוונטיזר לפי תוחלת העיוות:

$$(6.3) \quad D = E(d(\underline{x}, Q(\underline{x})))$$

קוונטיזר אופטימלי יהיה זה שיתן עיוות ממוצע מינימלי על פני כל כל וקטורי הייצוג האפשריים וכל החלוקות האפשריות למרחב בו מוגדר X . לבעיה זו אין פיתרון אנליטי אבל ניתן לקבל שני תנאים הכרחיים לאופטימליות:

1. תנאי השכן הקרוב - בהינתן מילון אזי אזורי ההחלטה חייבים לקיים:

$$(6.4) \quad S_i = \{\underline{x} | d(\underline{x}, \underline{y}_i) < d(\underline{x}, \underline{y}_j) \text{ for all } j \neq i\}$$

2. תנאי הצנטרואיד - בהינתן קבוצת אזורי החלטה אזי וקטורי הייצוג חייבים לקיים (עבור פונקציית עיוות ריבועית):

$$(6.5) \quad \underline{y}_i = E\{\underline{x} | \underline{x} \in S_i\} \quad i=1, \dots, L$$

על סמך שני התנאים ההכרחיים הוצע ב-[27] אלגוריתם איטרטיבי לתכנון קוונטיזר וקטורי המבטיח התכנסות למינימום מקומי של פונקציית העיוות וידוע בשם אלגוריתם LBG. האלגוריתם מסוגל לטפל בשני מקרים:

1. סטטיסטיקת המקור ידועה.
2. סטטיסטיקת המקור אינה ידועה אך נתונה סדרת אימון (Training Sequence).

נתאר את האלגוריתם עבור המקרה השני בו אנו משתמשים בעבודה :

- נתון: L - מספר הוקטורים במילון.
 $\epsilon > 0$ - סף עיוות המשמש כתנאי עצירה.
 Y_0 - מילון התחלתי בגודל L .
 $\{T_j\}_{j=1, \dots, R}$ - סדרת אימון ובה R וקטורים.

0. אתחול: $m=0$ - צעד האיטרציה.
 $D(-1) = \infty$ - עיוות התחלתי.

1. בהנתן מילון Y_m מצא חלוקה של סדרת הלימוד לקבוצות $\{S_{i,m}\}_{i=1}^{L-1}$ המקימות את תנאי השכן הקרוב, כלומר אם $T_j \in S_{i,m}$: T_k

$$d(T_i, Y_{i,m}) < d(T_j, Y_{i,m}) \text{ for all } i \neq j$$

חשב את העיוות הממוצע לאיטרציה זו:

$$D(m) = (1/R) \sum_{i=1}^R \min_{Y_j} \{d(T_i, Y_j)\}$$

2. אם $D(m-1) - D(m) / D(m) < \epsilon$: אז Y_m הוא המילון הסופי. עצור.
 3. לקבלת מילון אופטימלי עבור האיטרציה הבאה - Y_{m+1} , החלף כל איבר ב- Y_m בצנטרואיד של הקבוצות $S_{i,m}$.
 4. חזור ל 1.

כפי שצויין קודם, האלגוריתם מבטיח התכנסות למינימום מקומי ולכן בהנתן סדרת אימון טיב המילון תלוי אך ורק במילון ההתחלתי. בספרות נבחנו גישות שונות לבחירת המילון ההתחלתי [28]. השיטה בה אנו נקטנו היא שיטת הפיצול (Splitting) [29]. נתאר את השיטה:

0. אתחול: Y_{01} הוא הצנטרואיד של כל סדרת הלימוד (זהו מילון עם וקטור יצוג בודד).
 $G=1$ - גודל המילון.

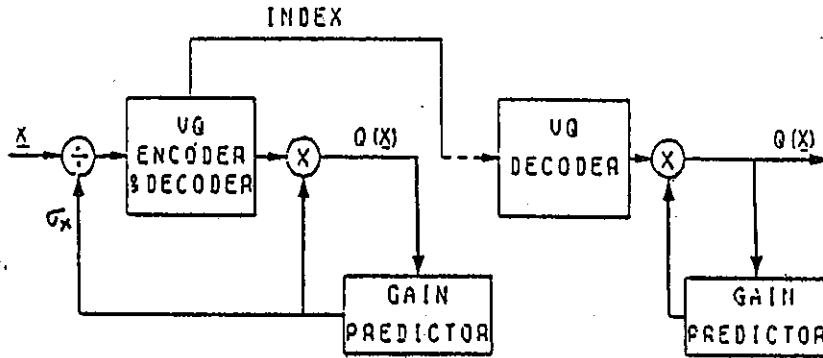
1. בהינתן מילון $Y_{0G} = \{y_{i1}\}$ פצל כל מילת קוד לשתי מילים $y_{i1} \pm \delta$ כאשר δ הוא וקטור פרטובציה קבוע. בסיום פעולה זו מתקבל מילון בגודל $2G$ - Y_{02G} . עדכן את המשתנה לגודל המילון - $G \rightarrow 2G$.

2. אם $G=L$ אז המילון ההתחלתי הוא $Y_0 = Y_{0G}$, אחרת $Y_{0G} = LBG\{Y_{0G}\}$ וחזור ל 1.

בעיה המתעוררת במימוש אלגוריתם ה-LBG עם שיטת הפיצול היא בעית התא ריק. תא ריק הינה קבוצת החלטה s_i שאיננה מכילה אף וקטור לימוד. תופעה זו מתקבלת כאשר מפצלים תא שמכיל וקטור לימוד בודד, במקרה זה ייווצר תא אחד שיכיל את תוכן התא המקורי ותא שני ריק. בכדי להתגבר על בעיה זו לא פוצל התא הבעייתי ובמקום זאת פוצל התא שמכפלת עיוות אכלוס (מספר הוקטורים בתא) מכסימלית [29].

6.2.2 קוונטיזציה וקטורית עם בקרת הגבר (Gain Adaptive VQ)

כמו בקוונטיזציה סקלרית, גם הקוונטיזציה הווקטורית, כשהוא משמש לקידוד אותות לא סטאציונריים, צריך להיות מלווה בסכמת אדפטציה שמתאימה אותו לאות הכניסה [30]. לכל וקטור משוערכת שונות σ_x לפי שיטה אחורית או קדמית והוקטור מנורמל לפיה. הקוונטיזציה מבוצעת על הוקטורים המנורמלים ובמקלט משוחזר האות על ידי הכפלת הוקטור בגורם הנרמול. מערכת עם בקרת הגבר אחורית מוצגת בציר 6.1.



ציר 6.1 - מערכת קוונטיזציה וקטורית עם בקרת הגבר אחורית.

Fig. 6.1 - Basic structure of backward gain-adaptive VQ.

עבור צורת קידוד זו תכנון המילון מתבצע בצורה שונה במקצת מהדרך שתוארה קודם כיוון שהפעם העיוות מוגדר בין הכניסה למקודד לפני נרמול ויציאתו לאחר הכפלה בשורש השונות (ולא בין הכניסה לקוונטיזציה ויציאתו).

$$(6.6) \quad d(\underline{x}, Q(\underline{x})) = \sigma_x^2 \sum_{i=1}^N (x_i / \sigma_x - y_i)^2$$

תכנון המילון מתבצע לפי אלגוריתם ה-LBG אלא שעבור מקרה זה שני התנאים ההכרחיים:

1. תנאי השכן הקרוב: בהינתן מילון אזי אזורי החלטה חייבים לקיים:

$$(6.7) \quad S_j = \{ \underline{x} \mid d(\underline{x}, \underline{y}_j) < d(\underline{x}, \underline{y}_i) \text{ for all } j \neq i \}$$

2. תנאי הצנטרואידי: בהינתן קבוצת אזורי החלטה אזי וקטורי היצוג חייבים לקיים:

$$(6.8) \quad \underline{y}_j = \frac{E\{\sigma_x \underline{x} \mid \frac{\underline{x}}{\sigma_x} \in S_j\}}{E\{\sigma_x^2 \mid \frac{\underline{x}}{\sigma_x} \in S_j\}}$$

בסעיפים הבאים נתאר מספר נסיונות שנעשו עם קוונטיזציה וקטורי שהופעל לקידוד תנאי ההתחלה האופטימליים ואות השארית.

בכל הבדיקות שיתוארו, סדרת הלימוד לפיה תוכננו הקוונטיזציות הווקטוריות הכילה 12,000 וקטורי לימוד (של 32 דגמים). הוקטורים נלקחו משהה משפטים באנגלית שנאמרו על ידי ששה דוברים שונים (שלושה גברים ושלוש נשים).

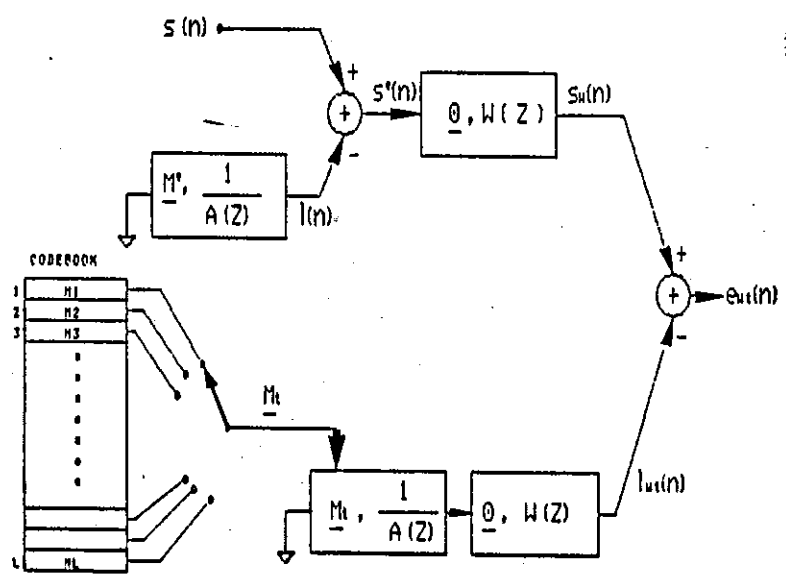
6.3 תכנון מילון אופטימלי לוקטור תנאי-התחלה

בתכנון מערכת עם קידוד וקטורי לוקטור הזכרון נפריד בין שלושה קריטריוני עיוות - טיב:

1. פונקצית העיוות לפיה מתוכנן המילון באלגוריתם ה-LBG.
2. פונקצית העיוות במקודד לפיה יבחר הוקטור במילון.
3. קריטריון הטיב לפיו נבחרים ביצועי המקודד.

רצוי ששלושת הקריטריונים יתלכדו, כלומר שהן תכנון המילון והן הוקטורים שיבחרו בתהליך הקידוד יתרמו לשיפור ביצועי המקודד. כפי שנראה לא ניתן לתכנן את המילון האופטימלי עם התיחסות לקריטריון הטיב של המקודד ושחיפוש הוקטור במילון לפי קריטריון הטיב של המקודד גורר עליה ניכרת בסיבוכיות.

קריטריון הטיב במקודדי אות שארית הוא סך השגיאה הריבועית המשוקללת בין אות המקור והאות המשוחזר. קריטריון זה מגדיר את הדרך שבה יש לבצע את החיפוש במילון כפי שמודגם בציור 6.2.



ציור 6.2 - קידוד וקטורי לתנאי ההתחלה.
 Fig. 6.2 - Vector quantization of memory vector.

מהציוור ניתן לראות כי :

1. העיוות בין וקטור הזכרון האופטימלי לבין וקטור כלשהוא במילון מוגדר לפי :

$$(6.9) \quad d(\underline{M}, \underline{M}_j) = \sum_n (s_w(n) - l_{w_j}(n))^2 \quad n=0, \dots, N-1$$

כלומר זו השגיאה הריבועית בין התגובה הנוצרת מהצבת כל אחד מוקטורי היצוג בזכרון מסנן הסינטזה ושקלולה, לבין אות המקור המשוקלל.

2. בתהליך הקידוד אין צורך למצא את וקטור הזכרון האופטימלי המתקבל מפיתרון מערכת המשוואות מסדר $P \times P$ (מביטוי-5.3) אלא מספיק למצוא את הוקטור במילון שהתגובה $l_{w_j}(n)$ תהיה הקרובה ביותר לאות המקור המשוקלל $s_w(n)$.

מ-1 ניתן להגדיר את פונקציית העיוות, בין סדרת האימון לבין המילון שלפיה יש לתכנן את המילון האופטימלי :

בהינתן :

1. סדרת לימוד של וקטורי זכרון אופטימליים $\{ \underline{TM}_j \} j=1, \dots, R$.
2. האות המשוקלל $s_{w_j}(n)$ המתאים לכל וקטור זכרון בסדרת הלימוד.
3. מקדמי היצוג של דגמי הזכרון בדגם תגובת הזכרון $f_{n_j}(i)$ (הם מתקבלים מתוך מקדמי המסנן $\{ a_{j,p} \}_{p=1}^{\Gamma-1} j=1, \dots, M$ לפי (5.14)).

אזי, פונקציית העיוות D , בין סדרת הלימוד $\{ \underline{TM}_j \} j=1, \dots, M$ לבין המילון $\{ \underline{M}_k \} k=1, \dots, L$:

$$(6.10) \quad \begin{aligned} D &= \sum_{j=1}^M d(\underline{TM}_j, \underline{M}_k) \\ \underline{M}_k &= \min_{t=1, \dots, L} \{ d(\underline{TM}_j, \underline{M}_t) \} \\ d(\underline{TM}_j, \underline{M}_k) &= \sum_n (s_{w_j}(n) - l_{w_k}(n))^2 \\ l_{w_k}(n) &= \sum_i f_{n_j}(i) M_k(-i) \end{aligned}$$

ודרוש מינימום עיוות על פני כל המילונים האפשריים.

תכנון מילון אופטימלי לפונקציית עיוות זו היא בעיה פרמטרית מורכבת שקשה להגדירה בצורה שתתאים לדפוס של אלגוריתם LBG. מסיבה זו הוחלט לקבל את המילון על ידי הפעלת האלגוריתם לסדרת לימוד של וקטורי זכרון עם קריטריון עיוות ריבועי (כפי שיפורט בהמשך). וקטור הזכרון מתקבל על ידי טרנספורמציה חד-ערכית המוגדרת ב 5.3, ולכן עבור מילון אינסופי יתלכדו ביצועיו עם הקוונטיזר האופטימלי שהוגדר קודם. בהמשך נתאר נסיונות בהם קודדו וקטורים נוספים המגדירים חד-ערכית את וקטור הזכרון.

6.4 קידוד וקטורי של וקטור תנאי התחלה

וקטור תנאי ההתחלה האופטימליים, מיצג דגמי הדיבור (על ידי הטונספורמציה המוגדרת בביטוי 5.2). בדומה לאות הדיבור, גם וקטור הזכרון אינו סטאציונרי ולכן הופעל אלגוריתם ה LBG עם בקרת הגבר כפי שתואר בפרק הקודם. את הנרמול ביצענו לפי שונות וקטור הזכרון מהמסגרת הקודמת - g, מידע אשר קיים במשדר ובמקלט (משערך אחורי):

$$(6.11) \quad g = (1/P) \sqrt{\sum_{i=1}^P M(-i)^2}$$

קריטריון השגיאה לפיו נבחר וקטור הייצוג במילון היה שגיאה ריבועית מינימלית בינו לבין וקטור הזכרון המנורמל:

$$(6.12) \quad \underline{M}_t = \min\{E_+, t=1, \dots, L\}$$

$$E_+ = d(\underline{M}, \underline{M}_+) = \sum_{i=1}^P ((M(-i)/g) - M_+(-i))^2$$

טבלא 6.1 נותנת את התוצאות שהתקבלו במערכת עם עדכון וקטור הזכרון לפני קביעת הפולסים וקידוד עם קריטריון שגיאה ריבועית על וקטורי הזכרון. (גודל מסגרת האנליזה 32 דגמים - 4 msec).

2 פולסים ללא עדכון זכרון	2 פולסים עם עדכון ללא קידוד	2 פולסים עם עדכון וקידוד L=512 ומילון
9.3 dB	16.17 dB	0.7 dB

טבלא 6.1 - ביצועי מערכת ה-MPE עם עדכון וקידוד וקטור הזכרון קודם לקביעת הפולסים וקריטריון שגיאה כפי שמוגדר ב-6.12.

Table 6.1 - Performance of MPE algorithm with memory vector update and quantization before locating pulses, and a distortion criteria as defined in 6.12.

התוצאות שהתקבלו גרועות מאלו שהיו מתקבלות ללא עדכון. הסיבה לכך היא שקריטריון השגיאה לחיפוש במילון אינו מתאים לקריטריון הטיב לפיו נמדדים ביצועי המערכת.

כדי להתאים את הקריטריונים בוצע החיפוש בדרך הבאה:

1. לכל וקטור במילון חושבה התגובה המשוקללת - $lw_+(n) \quad t=1, \dots, L$
2. לכל תגובה חושבה אנרגיית השגיאה בינה לבין אות המקור המשוקלל:

$$(6.13) \quad E_+ = d(\underline{M}, \underline{M}_+) = \sum_n (s_w(n) - lw_+(n))^2 \quad t=1, \dots, L$$

3. נבחר הוקטור עבורו אנרגיית השגיאה מינימלית:

$$(6.14) \quad \underline{M}_t = \min\{E_+, t=1, \dots, L\}$$

טבלא 6.2 נותנת את התוצאות שהתקבלו במערכת עם עדכון וקטור הזכרון לפני קביעת הפולסים וקידוד עם קריטריון שגיאה ריבועית בין תגובות הזכרון והאות המשוקלל.

2 פולסים ללא עדכון זכרון	2 פולסים עם עדכון ללא קידוד	2 פולסים עם עדכון וקידוד ומילון L=512
9.3 dB	16.17 dB	10.6 dB

טבלא 6.2 - ביצועי מערכת ה MPE עם עדכון וקידוד וקטור הזכרון קודם לקביעת הפולסים וקריטריון שגיאה כפי שמוגדר ב 6.14.

Table 6.2 - Performance of MPE algorithm with memory vector update and quantization before locating pulses, and a distortion criteria as defined in 6.14.

התוצאות אומנם מצביעות על שיפור ניכר בהשוואה לאלו שהתקבלו עם הקריטריון הקודם אך עדיין אין השיטה מוצלחת, מהטעמים הבאים:

1. התוספת לקצב בכל מסגרת זמן היא של 9 סיביות (ליצוג מילון עם 512 וקטורים) השווה לתוספת הנחוצה לקידוד פולס נוסף (5 סיביות למיקום ו- 4 סיביות לאמפליטודה) אך השיפור באיכות אינו מתקרב לזה המתקבל עם 3 פולסים - [dB] 12.43.
2. הסיבוכיות הכרוכה בחיפוש הוקטור האופטימלי במילון עולה בהרבה על זו הנחוצה לחישוב פולס נוסף.

בשלב זה הוחלט לאמץ גישה שננקטה במקודד CELP והיא שילוב של גורם הגבר g_+ באופן הבא :

1. לכל וקטור במילון מחושב גורם הגבר כך שהשגיאה הריבועית בין התגובה לזכרון ואות המקור - מינימלית.

$$(6.15) \quad E_+ = \sum_n (s_w(n) - g_+ l_w(n))^2$$

למציאת גורם ההגבר האופטימלי גוזרים ומשוים לאפס:

$$(6.16) \quad g_+ = \frac{\sum_n s_w(n) l_w(n)}{\sum_n l_w(n)^2}$$

הצבתו ב-6.15 נותנת את אנרגיית השגיאה המינימלית:

$$(6.17) \quad E_+ = \sum_n s_w(n)^2 - \frac{(\sum_n s_w(n) l_w(n))^2}{\sum_n l_w(n)^2}$$

2. הוקטור שנבחר הוא זה שעבורו השגיאה היא הקטנה מכולם. (בערוץ יש לשדר את אינדקס הוקטור ואת גורם ההגבר המתאים לאחר שעבר קוונטיזציה):

$$(6.18) \quad M_L = \min\{E_t, t=1, \dots, L\}$$

שיטה זו מזכירה את הגישה של תכנון מילון עם אדפטציה להגבר ומשערך קדמי שתוארה בסעיף 6.2.2, אלא שכאן מתאימים את גורם ההגבר לקריטריון הטיב של המקודד.

טבלא 6.3 נותנת את התוצאות שהתקבלו במערכת עם עדכון וקטור הזכרון לפני קביעת הפולסים. הקידוד נעשה עם קריטריון שגיאה כפי שתואר קודם (ללא קוונטיזציה לגורם ההגבר).

גודל המילון L	2 פולסים עם עדכון וקידוד
0	9.3 dB
256	12.3 dB
∞	16.17 dB

טבלא 6.3 - ביצועי מערכת ה MPE עם עדכון וקידוד וקטור הזכרון קודם לקביעת הפולסים וקריטריון שגיאה כפי שמוגדר ב-6.18.

Table 6.3 - Performance of MPE algorithm with memory vector update and quantization before locating pulses, and a distortion criteria as defined in 6.18.

השיפור בביצועי המקודד עדיין אינו מספק. הסיבוכיות הכרוכה בחיפוש הוקטור האופטימלי והקצאת הסיביות לאינדקס ולגורם ההגבר אינם עומדים במבחן ההשוואה לקביעה ולקידוד פולס נוסף. הסיבה לתופעה זו היא שהמילון אינו מותאם לקריטריון הטיב של המקודד ואינו גדול מספיק כדי להתלכד עם המילון האופטימלי. השאלה היא האם קיים מילון עם מספר קטן של וקטורים שביצועיו יעלו על המילונים שתוכננו על ידנו.

כדי לבדוק שאלה זו הוחלט לבנות מילון לוקטורים אחרים המגדירים חד-ערכית את וקטור הזכרון ולכן מקיימים גם הם את התכונה שעבור מילונים אינסופיים הם מתלכדים עם המילון האופטימלי על פי קריטריון הטיב המוחלט של המקודד.

בציור 6.3 מתוארות שלוש סכמות שהמשותף להן הוא העובדה שאות היציאה זהה בכולן אך הן נבדלות בדרך שבא הוא נוצר.

בציור 6.3-a אות היציאה- $w(n)$ נקבע לפי תנאי ההתחלה של מסנן הסינטזה.

בציור 6.3-b אות היציאה ניתן לחישוב, לכל $n \geq 0$, אם ידועים P הדגמים הראשונים באות היציאה (ולא ידועים תנאי ההתחלה של מסנן הסינטזה):

$$(6.19) \quad \underline{w} = [w(0), \dots, w(P-1)]$$

כלומר, בהינתן הוקטור \underline{w} אזי ניתן למצא את תנאי ההתחלה מהקשר:

$$(6.20) \quad w(n) = \sum_{i=1}^P f_n(i)M(-i)$$

ולקבל את אות היציאה לכל $n \geq 0$ באופן דומה לציור 6.2-a.

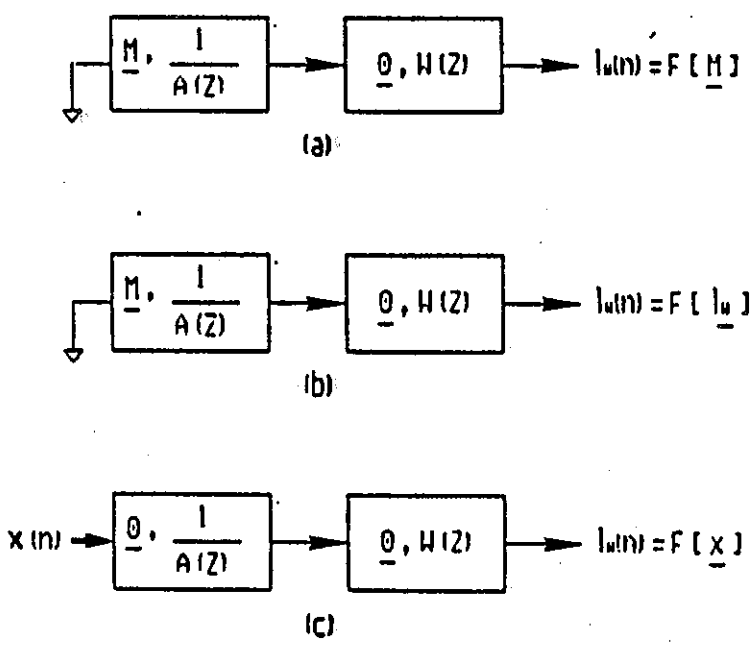
בציור 6.3-c אות היציאה נקבע לפי אות כניסה $x(n)$ שהוגדר בסעיף 5.2.1:

$$(6.21) \quad \underline{x} = [x(0), \dots, x(-(P-1))]$$

$$x(n) = \sum_{i=1}^P v_i(n)M(-i)$$

1- $v_i(n)$ הוגדר בביטוי 5.9.

בסעיפים הבאים נתאר את הבדיקות שביצענו עם קידוד וקטורי של הוקטורים $\underline{x}, \underline{w}$.



ציור 6.3 - מערכות שקולות (אות יציאה זהה).
 Fig. 6.3 - System with the same output signals.

6.5 קידוד P דגמים ראשונים בתגובת הזכרון האופטימלית המשוקללת

כאן, סדרת הלימוד לתוכנית ה LBG הכילה וקטורים של P דגמים ראשונים בתגובת הזכרון האופטימלית המשוקללת. המילון תוכנן עם בקרת הגבר והנרמול נעשה לפי שונות הוקטור מהמסגרת הקודמת.

שיטת החיפוש של הוקטור האופטימלי נעשתה באופן הבא :

1. לכל וקטור במילון חושב וקטור הזכרון המתאים מתוך מערכת המשוואות 6.20.
2. לכל וקטור זכרון חושב גורם הגבר g_+ כך שאנרגיית השגיאה בין אות המקור והתגובה לזכרון - המשוקללים, תהיה מינימלית:

$$(6.22) \quad E_+ = \sum_{n=0}^{N-1} (s_w(n) - g_+ l w_+(n))^2$$

3. הוקטור שנבחר הוא זה שעבורו אנרגיית השגיאה היא הקטנה מכולם:

$$(6.23) \quad \underline{l w}_+ = \text{MIN}\{E_+ \quad t=1, \dots, L\}$$

טבלא 6.4 נותנת את התוצאות שהתקבלו במערכת עם עדכון וקטור הזכרון לפני קביעת הפולסים (ללא קוונטיזציה לגורם ההגבר).

גודל המילון L	2 פולסים עם עדכון וקידוד
0	9.3 dB
256	12.1 dB
∞	16.17 dB

טבלא 6.4 - ביצועי מערכת ה MPE עם עדכון וקידוד וקטור הזכרון קודם לקביעת הפולסים וקריטריון שגיאה כפי שמוגדר ב- 6.23.

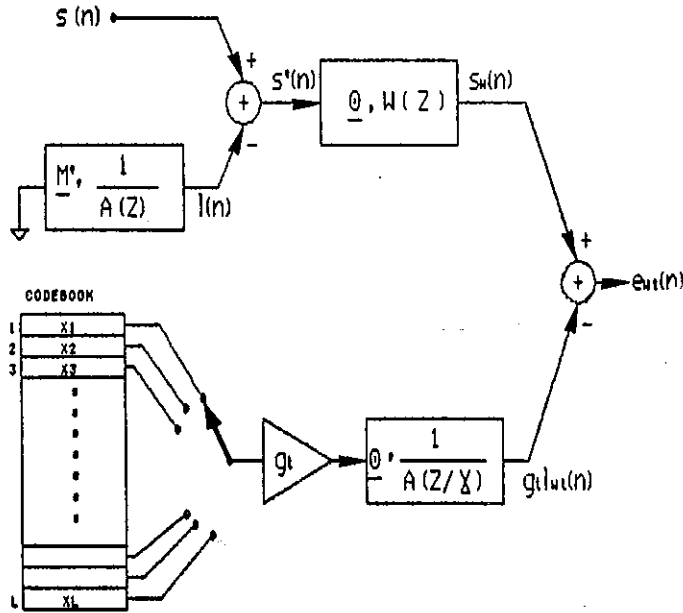
Table 6.4 - Performance of MPE algorithm with memory vector update and quantization before locating pulses, and a distortion criteria as defined in 6.23.

התוצאות דומות לאלו שהתקבלו עם מילון של וקטורי הזכרון. מאותן סיבות שתוארו בסעיף הקודם אין שיטת קידוד זו מומלצת.

6.6 קידוד וקטור הכניסה השקול א

כאן, סדרת הלימוד לתוכנית ה LBG הכילה וקטורים שחושבו מתוך וקטורי הזכרון האופטימלים לפי 6.21. המילון תוכנן עם בקרת הגבר והנרמול נעשה לפי שונות הוקטור מהמסגרת הקודמת.

שיטת החיפוש של הוקטור האופטימלי מודגמת בציור 6.4. (כיוון ששתי המסננות עם זכרון מאופס ניתן לקבל את פונקציית התמסורת השקולה - $(1/A(Z/\Gamma))$ על ידי הכפלת פונקציות התמסורת).



ציור 6.4 -- חיפוש וקטור כניסה אופטימלי.
Fig 6.4 - search of optimal input vector.

האלגוריתם :

1. לכל וקטור במילון מחושבת תגובת מסנן הסינטזה המשוקלל.
2. לכל תגובה מחושב גורם הגבר g_+ כך שאנרגיית השגיאה E_+ תהיה מינימלית:

$$(6.24) \quad E_+ = \sum_n (s_w(n) - g_+ l_w(n))^2$$

למצאת גורם ההגבר האופטימלי גוזרים ומשוים לאפס:

$$(6.25) \quad g_+ = \frac{\sum_n s_w(n) l_w(n)}{\sum_n l_w(n)^2}$$

והצבתו בביטוי 6.24 נותן את אנרגיית השגיאה המינימלית:

$$(6.26) \quad E_+ = \sum_n s_w(n)^2 - \frac{(\sum_n s_w(n) l_w(n))^2}{\sum_n l_w(n)^2}$$

3. הוקטור שנבחר הוא זה שעבורו אנרגיית השגיאה היא הקטנה
מכולם:

$$x_k = \text{MIN}\{E_k, t=1, \dots, L\} \quad (6.27)$$

טבלא 6.5 נותנת את התוצאות שהתקבלו במערכת עם עדכון וקטור הזכרון לפני קביעת הפולסים (ללא קוונטיזציה לגורם ההגבר).

גודל המילון L	2 פולסים עם עדכון וקידוד
0	9.3 dB
16	11.7 dB
32	12.2 dB
64	12.7 dB
128	13.1 dB
256	13.5 dB
∞	16.1 dB

טבלא 6.5 - ביצועי מערכת ה MPE עם עדכון וקידוד וקטור הזכרון קודם לקביעת הפולסים וקריטריון שגיאה כפי שמוגדר ב-6.27.

Table 6.5 - Performance of MPE algorithm with memory vector update and quantization before locating pulses, and a distortion criteria as defined in 6.27.

כדי שניתן יהיה להשוות לביצועי מקודד ה Multi-Pulse נציג בטבלא 6.6 את התוצאות שהתקבלו בקידוד פולסים בלבד. גודל מסגרת האנליזה N=32 דגמים (4 msec).

מספר הפולסים במסגרת	SEGSNR dB
1	5.8
2	9.3
3	12.4
4	14.7

טבלא 6.6 - ביצועי אלגוריתם ה-MPE המקורי. (N=32).
Table 6.6 - Performance of MPE algorithm. (N=32).

במילון של 64 וקטורים משיגים איכות דומה לזו המתקבלת עם קידוד פולס נוסף. הקצאת הסיביות לאינדקס הוקטור במילון ולהגבר אף היא קרובה לזו הנחוצה לקידוד פולס נוסף. ההבדל המשמעותי הוא בסיבוכיות הכרוכה בחיפוש במילון אשר עולה בהרבה על זו הכרוכה בקידוד פולס נוסף.

נבדקו נקודות עבודה שונות כלומר חיפוש במילון של וקטורי עירור ולאחריו קביעת מספר שונה של פולסים. בכולן התקבלו תוצאות דומות לנ"ל. (כלומר מילון של 64 נתן תוצאות קרובות לאלו המתקבלות עם קידוד פולס נוסף).

האלגוריתם האחרון וציוור 6.4 דומים לגישה שננקטה בקידוד אות השארית במקודד CELP- (שתוארה בסעיף 3.5) עם מספר הבדלים:

1. ב-CELP אורך הסדרות במילון הוא כאורך מסגרת האנליזה ואילו אצלנו אורך סדרת העירור הוא - P.
2. ב-CELP אות השארית שמקודד הוא ללא מבנה מחזורי במחזור Pitch.

נסביר את המשמעות של הבדלים אלו בין שתי הגישות:

1. אורך הסדרה קובע חסם העליון לטיב שניתן להשיג במקודד. במקודד CELP ניתן להשיג איכות טובה כרצוננו על ידי הגדלת מספר הוקטורים במילון בעוד שהגדלת מספר הווקטורים במילון עם אורך מוגבל ל-P דגמים מאפשר להשיג לכל היותר את התוצאות המופיעות בציוור 3-5.5.

2. הוצאת המחזוריות מאות העירור איפשרה שימוש במילון מאוכלס באקראי (במשתנים גאוסים חסרי קורלציה) כיוון שזהו מודל טוב לאות העירור [15].

על סמך שיקולים אלו החלטנו לבצע מספר שינויים בגישה לאופן הקידוד:

1. הוחלט להשתמש במילונים לאות עירור כשהסדרות הם בגודל מסגרת האנליזה (כמו ב-CELP).
2. אכלוס המילון יהיה אקראי (אף הוא בדומה למקודד ה-CELP).
3. יבדק אלגוריתם ה-Multi-Pulse עם חזאי Pitch להוצאת המבנה המחזורי של אות העירור.
4. יבדק ניצול אלגוריתם ה-Multi-Pulse (ללא חזאי Pitch), להוצאת המבנה המחזורי של אות העירור במקום חזאי Pitch על ידי שימוש בפולסים לפני שילוב הקוד.

הרעיונות והבדיקות שבוצעו מתוארים בפרק הבא.

7.1 מבוא

בפרק הקודם הראינו שקידוד וקטורי לתנאי ההתחלה של מסנן הסינטזה נתן את הביצועים הטובים ביותר כאשר הוקטור שקודד היה אות כניסה למסנן - באורך P דגמים, אך התוצאות שהתקבלו לא סיפקו. בפרק זה נציג מספר אלגוריתמים חדשים המשלבים עירור פולסים וספר קוד שבו הסדרות הן גאוסיות חסרות קורלציה באורך N (גודל מסגרת האנליזה).

זהו למעשה מודל חדש לאות השארית המשלב את המודלים של ה-Multipulse וה-CELP (Code Excited ...). בהבדל עקרוני שהמילונים בהם השתמשנו מכילים כ-32 מילות קוד בלבד, זאת כדי לשמור על סיבוכיות נמוכה.

שתי הבעיות שהתעוררו בבואנו ליישם מודל זה הינן:

1. כיצד לאפשר שימוש בספר קוד מאוכלס באקראי - במקום שהמילון יתקבל מתוך סדרת אימון - בעזרת אלגוריתם ה-LBG. (צורך זה יובהר בסעיף הבא).

2. כיצד לשלב בעירור את ספר הקוד והפולסים.

בסעיפים הבאים נציג שתי שיטות שונות לשילוב של מילון מאוכלס באקראי ליצוג אות העירור:

1. השיטה הראשונה מנצלת תכונה של אלגוריתם ה-MPE לפיה ממוקמים הפולסים המרכזיים בצורה מחזורית ב-Pitch, ולפיכך אות השארית המתקבל לאחר קביעת מספר פולסים הוא בקרוב לבו - כפי שמתואר בציור 3.7. כלומר לפי שיטה זו מוציאים את המבנה המחזורי מאות השארית בעזרת מספר קטן של פולסים הנמצאים בעזרת אלגוריתם ה-MPE, ולא את השארית הנותר מתאימים את המודל המשלב פולסים וספר קוד.

2. השיטה השנייה מנצלת טכניקה המקובלת במקודדים ה-CELP ו-MPE והיא שימוש במסנן סינטזה עם חזאי Pitch.

7.2 שילוב ספר קוד אחרי קביעת הפולסים

תפקיד הפולסים הנקבעים בעזרת אלגוריתם ה-MPE דומה בפועל לתפקיד חזאי Pitch כיוון ששניהם גורמים להלבנה של אות השארית עם הוצאת הקורלציה במחזור Pitch כפי שמוזגם בציור 3.7. תכונה זו מתקבלת היות ופולסי העירור מתקבלים מתואמים למחזור אות השארית כך שאחרי שקובעים מספר פולסים ומפחיתים את השפעתם, אות השארית הופך ללבן (בקרוב).

כדי לאשר הנחה זו, החלפנו את חזאי ה-Pitch במקודד CELP ב-K פולסים אשר נקבעו על ידי אלגוריתם ה-MPE. כלומר מסנן הסינטזה הוחלף ל- $1/A(Z)$ במקום $1/[A(Z)(1-Pp(Z))]$ והאלגוריתם שמומש הוא:

0. אתחול: מצא אות מקור משוקלל לאחר הפחתה של תגובת הזכרון $s_w(n)$ -

1. מצא K פולסים בעזרת אלגוריתם ה-MPE והפחת את תרומתם מהאות שאותו מקרבים:

$$s_w''(n) = s_w(n) - \sum_{i=1}^K b_i h_w(n-m_i)$$

2. מצא סדרת קוד וגורם הגבר אופטימליים במילון עם $L=1024$ סדרות קוד.

התוצאות שהתקבלו מוצגות בטבלא 7.1.

גודל מסגרת N	SEGSNR-1 (Pitch pred.)	מספר פולסים K	SEGSNR-2 (Multipulse)
32	12.6	1	12.2
64	8.4	1	7.5
64	8.4	2	9.2
64	8.4	3	10.8

טבלא 7.1 - השוואת ביצועי מקודד CELP 1. עם חזאי Pitch 2. קביעת K פולסים ואחרהם מציאת סדרת קוד אופטימלית כשמסנן הסינטזה אינו כולל חזאי Pitch.

Table 7.1 - Comparing CELP performance 1. With Pitch predictor 2. Locating K pulses and then searching for the best code word in the codebook, while synthesis filter does not include a Pitch predictor.

לשם השוואה נבחר את התוצאה שהתקבלה במקרה $N=64, K=3$. קביעה זו מתאימה למקרה הגבולי של מחזור Pitch השווה לכ- 20 דגמים (2.5 msec), במקרה זה מתקבלים ביצועים טובים ב- [dB] 2.4 כאשר מומש החזאי בעזרת פולסים.

ב-[9] נמצא כי הקצאת הסיביות לקידוד פרמטרי חזאי Pitch, מחזור ה-Pitch ו-3 מקדמים, היא 2000 bits/sec. בנספח א-b נמצא כי לקידוד פולס בודד דרושים 1625 bits/sec לקידוד שני פולסים דרושים 2875 bits/sec ולקידוד שלושה פולסים 4125 bits/sec. סיבוכיות חיפוש 1-3 פולסים באלגוריתם MPE דומה לסיבוכיות אלגוריתמים לקביעה של מקדמי חזאי ה-Pitch כמפורט בנספח ב.

מכאן הוסקו המסקנות הבאות:

1. שתי המערכות שקולות בקרוב בביצועהן באותו קצב.
 2. ניתן להחליף את החזאי בפולסים, ומודל רעש גאוסי לבן לעירור אכן מתאים לאות השארית המתקבל אחרי הפחתת התגובה לפולסים.
- לאחר שהתוצאות אישרו את ההנחה שניתן להלבין את אות השארית בעזרת פולסים, ניסינו לשלב את החיפוש של סדרת קוד במילון קטן יותר, אחרי קביעה של מספר גדול יותר של פולסים.
- התוצאות שהתקבלו עבור מסגרת של 8 msec (N=64) מוצגות בטבלא 7.2.

גודל המילון L	K=3 [dB]	K=4 [dB]	K=5 [dB]
0	7.3	9.6	11.4
16	9.3	11.1	12.2
32	9.6	11.4	12.6
64	10.0	11.6	12.9
128	10.2	11.9	13.1

טבלא 7.2 - ביצועי אלגוריתם המשלב חיפוש סדרת קוד אחרי קביעת K פולסים.

Table 7.2 - Performance of algorithm in which the search for the best code word is done after locating K pulses.

התוצאות מצביעות על כך שביצועי מערכת המשלבת מילון של כ-32 סדרות ולאחריו K פולסים, שקולה לביצועים המתקבלים עם מקודד MPE עם K+1 פולסים. נשווה את שני המאפיינים הנוספים של המערכות והם הקצב והסיבוכיות.

הקצאת הסיביות לקידוד פולס היא 10 bit (6 bit למיקום ו-4 bit לאמפליטודה) והיא זהה לזו הנחוצה לקידוד סדרת קוד - 5bit לאינדקס (למילון עם L=32 מילות קוד) ו-5 bit לגורם ההגבר.

הסיבוכיות הכרוכה בחיפוש פולס קטנה בהרבה מזו הנחוצה לבחירת סדרת קוד אופטימלית, כפי שמפורט בנספח ב. המסקנה היא ששילוב סדרת קוד (לאחר K פולסים) במקום פולס אינה כדאית.

כפי שניתן לראות, במילונים קטנים קיימת עדיפות ברורה לבחירה של מקרים בהם מרבית הפולסים נקבעים קודם לכוון שהמילון אינו עשיר דיו. עם הגדלת המילון, מספר הפעמים שנבחרים מקרים בהם משלבים את הקוד בשלבים הראשונים - CP^* , PCP^* - קטן מהאחרים כיוון שמודל רעש לבן אינו מתאים לאות (למעט קטעי קול אל-קוליים).

כמו כן ניתן ללמוד מהתוצאות שביצועי מערכת המשלבת מילון של 32 סדרות קוד עם $K=6$, עולה במקצת על ביצועי מקודד-MPE עם שבעה פולסים -14 [dB], וזהו למעשה הרווח באיכות בהשוואה למקודד ה-MPE, באותו קצב.

הביטוי לסיבוכיות אלגוריתם המיתוג (פעולות כפל לדגם) ניתן בנוסחא הבאה (הפרוט נמצא בנספח ב-g):

$$7.1 \quad T \approx P(4+2K)L + L(2+K) + (K+1) \log_2(N) + 0.5K(N+1)$$

לדוגמא: עבור $N=64, L=32, K=6$ מתקבל $T \approx 650$ פעולות כפל לדגם. כפי שניתן לראות, הגורמים הדומיננטים בקביעת הסיבוכיות הם: גודל המילון - L , ומספר המקרים - $K+1$. על סמך נתוני ההיסטוגרמה והסיבוכיות הוחלט לבדוק רק חלק מהמקרים, ובכך להקטין את הסיבוכיות.

כדי שלא לפגוע באיכות, לא נבדקו המקרים בהם משולב הקוד בשלבים הראשונים (כיוון שכפי שמתקבל מטבלא 7.3, מקרים אלו נבחרו מספר פעמים הקטן ביותר). משמעות החלטה זו הוא שתחילה נקבעים מספר קבוע של פולסים ורק אחרי הפחתת תרומתם מבצעים את המיתוג בין מקרים שונים לשילוב קוד ופולסים.

למעשה משמשים הפולסים שנקבעים בשלב הראשון להלבנת אות השארית ולהתאמתו למילון האקראי המשמש לקידודו (כפי שנעשה בתחילת הסעיף).

נגדיר K_1 - מספר הפולסים שנקבעים בשלב הראשון, אזי שיטת הקידוד החדשה מתוארת באלגוריתם הבא:

0. אתחול: מצא את מקור משוקלל לאחר הפחתה של תגובת הזכרון $s_w(n)$.

1. מצא K_1 פולסים בעזרת אלגוריתם ה-MPE (משמשים להלבנת אות השארית) והפחת את תרומתם מהאות שאותו מקרבים:

$$s_w''(n) = s_w(n) - \sum_{i=1}^{K_1} b_i h_w(n-m_i)$$

2. חשב $K-K_1+1$ אפשרויות לשילוב סדרת קוד בפולסים.

3. מתג את השידור למקרה שבו השגיאה הריבועית המשוקללת היא מינימלית (בנוסף לשידור סיבית לצינון האם במקרה שנבחר משולב קוד או לא).

ברור כי האלגוריתם הקודם הוא מקרה פרטי של האלגוריתם החדש עבור $K_1=0$. הביטוי לסיבוכיות הפעם הוא (מפורט בנספח ב-g):

$$7.2 \quad T_{SPACE} \approx P(4+2(K-K_1)) + L(2+K) + (K+1) \log_2(N) + 0.5(K-K_1)(N+1)$$

כלומר הסיבוכיות קטנה בקרוב לפי היחס $(K-K_1)$: K. מערכת זו, הממתגת בין אפשרויות שונות לשילוב בין פולסים וקוד נקראה:

SPACE - Switched Pulse And Code Excitation

בדקנו את ביצועי המערכת עם הפרמטרים הבאים:

- $N=64$ - גודל מסגרת האנליזה (8 msec)
- $K_1=3$ - שלושה פולסים שנקבעים בשלב הראשון ומוציאים את המבנה המחזורי באות השארית.
- $K=6$ - מספר הפולסים המקסימלי במסגרת זמן (עבור המקרה שלא משולב קוד - P^k)
- L - מספר הוקטורים במילון.

התוצאות שהתקבלו מוצגות בטבלא 7.4.

גודל המילון L	SEGSNR [dB]	היסטוגרמה			
		P^3CP^2	P^4CP	P^5C	P^6
0	12.8	0	0	0	0
16	13.7	221	235	263	435
32	14.0	318	263	273	300
64	14.3	391	270	226	267

טבלא 7.4 - ביצועי מערכת SPACE עם: $K_1=3, K=6, N=64$.
 Table 7.4 - Performance of SPACE coder with:
 $K_1=3, K=6, N=64$.

מתקבל כי במילון עם 32 מילות קוד הביצועים זהים למקודד MPE עם 7 פולסים אלא שכאן הקצב מתאים למקודד MPE עם 6 פולסים למסגרת. התשלום הוא בסיבוכיות שגדלה ביחס $1:3 \approx 124:380$ (הערכים נלקחו מנספח ב-d, g).

7.3 שילוב קוד ופולטים לעירור מסנן סינטזה עם Pitch Predictor

ראינו בסעיף הקודם (ובפרק 3), שכדי להתאים את המודל של סדרת רעש גאוסי לבן לאות השארית, יש צורך להוציא ממנו את המבנה המחזורי במחזור ה-Pitch. הטכניקה המקובלת, שיושמה במקודד CELP, היא שימוש במסנן סינטזה הכולל חזאי Pitch - $1/[(A(Z)(1-Pp(z)))]$. ניתן אם כך להפעיל את אלגוריתם ה-SPACE שהוצע בסעיף הקודם לשילוב סדרת קוד בפולטים ומיתוג בין המקרים השונים, באנליזה שתכלול חזאי Pitch.

בטבלא 7.5 מרוכזות התוצאות שהתקבלו עם מילון של 64 מילות קוד עבור K משתנה, והמיתוג נעשה בין כל האפשרויות לשילוב סדרת קוד - כלומר $K_1=0$. כמו כן נתונות התוצאות שהתקבלו במקודד MPE עם Pitch Predictor ו K פולטים.

K	MPE [dB]	SPACE [dB]
1	6.4	7.6
2	9.3	10.8
3	11.4	13.1
4	13.3	15.0
5	14.9	16.5
6	16.2	17.8

טבלא 7.5 - ביצועי מערכת SPACE עם Pitch Predictor, מילון קבוע - $L=64$ ומיתוג בין $K+1$ האפשרויות לשילוב סדרת קוד עם פולטים ו - $K_1=0$.

Table 7.5 - Performance of SPACE coder with Pitch predictor, codebook with $L=64$ codes and switching throw $K+1$ option to include code in pulses & $K_1=0$.

התוצאות מצביעות על כך שבעזרת מילון עם $L=64$ ו - $K_1=0$, ניתן להשיג איכות קרובה לזו המתקבלת במקודד MPE עם $K+1$ פולטים.

התוצאות, בחתך אחר, עבור - $K=5$ ומילון בגדלים שונים, מוצגות בטבלא 7.6.

גודל המילון L	SEGSNR [dB]	היסטוגרמה					
		CP ⁴	PCP ³	P ² CP ²	P ³ CP	P ⁴ C	P ⁵
0	14.9	0	0	0	0	0	1154
16	16.0	193	152	193	154	138	324
32	16.2	203	166	188	203	166	228
64	16.5	230	186	178	183	189	188

טבלא 7.6 - ביצועי מערכת SPACE עם חזאי Pitch, ומיתוג בין 6 אפשרויות לשילוב קוד בפולסים עם $K_1=0, K=5$.

Table 7.6 - Performance of SPACE coder with Pitch Predictor, and switching throw 6 option to include code in pulses with $K=0, K_1=0$.

כפי שניתן לראות השיפור באיכות עם הגדלת המילון אינה גדולה. במקביל הופכת ההיסטוגרמה לאחידה אחרי שבמילונים קטנים קיימת עדיפות ברורה למקרה P^5 .

סיבוכיות האנליזה של מקודד SPACE עם ובלוי חזאי Pitch אינה שונה בהרבה וניתנת בביטוי 7.2. ניסיונות להקטינה על ידי הגדלת K_1 , כלומר בדיקה של מספר מצומצם של מקרים, גררו פגיעה באיכות באופן שאבד הרווח (באיכות) של פולס נוסף (כמו במקודד MPE עם 6 פולסים) ולכן בהשוואה למקודד SPACE שתואר בסעיף הקודם, שילוב חזאי Pitch אינו מומלץ.

בפרק הבא נציג טבלת השוואה מסכמת בין המקודדים : CELP, MPE, SPACE ונסכם את העבודה.

פרק 8 - סיכום ומסקנות

בטבלא 8.1 מסוכמות תוצאות שהתקבלו בשלושת המקודדים שהוצגו בעבודה זו: MPE, CELP, SPACE בקצבים שהם סביב 9.6 Kbps.

הערות לטבלא

1. הסיבוכיות המצוינת היא של אלגוריתם החיפוש לעירור בלבד כמפורט בנספח ב.
2. עבור מקודד CELP, יחס אות לרעש המצויין הוא ללא קוונטיזציה של המרכיבים השונים והקצב משוער על סמך הצעות שונות בספרות [9], [32]. בכל שאר המערכות התוצאות הם עם קוונטיזציה מלאה והפרוט ניתן בנספח א.
3. גודל המסגרת במקודדים MPE, SPACE הוא 8 msec (N=64).

	SEGSNR [dB]	סיבוכיות [כפלים לדגם]	קצב [Kbps]
MPE			
6 pulses	11.6	118	9.208
7 pulses	12.7	124	10.333
SPACE			
$K_1=3, K=6, L=32$	12.8	380	9.448
CELP, L=1024			
גודל מסגרת			
4 msec	12.6	4160	6.5-7.5
2 msec	17.7	4160	10.2-11.2

טבלא 8.1 - השוואת המקודדים: MPE, CELP, SPACE.
Table 8.1 - Comparison of the coders: MPE, CELP, SPACE.

כדוגמא נפרט את חישוב הקצב במקודד SPACE:

קידוד מקדמי ה-LPC - 1708 bps
 קידוד 5 פולסים - 6375 bps
 סיבית המציינת אם שולב קוד או לא - 125 bps
 קידוד פולס או קוד עם 10 bit למסגרת - 1250 bps
 סה"כ - 9448 bps

הערה: לקידוד פולס נחוצים 9 bit בלבד, וכדי להתאים ל 10 bit ישודר המיקום המדויק של הפולס ולא הפרש בין מיקומי פולסים עוקבים. כלומר ישודרו: 6 bit למיקום (במקום 5 bit) ו- 4 bit לאמפליטודה המנורמלת.

כפי שניתן ללמוד מהטבלא, במקודד SPACE עם מילון של 32 סדרות, הביצועים (SEGSNR) עולים על מקודד MPE בקצב דומה (-) עם 6 פולסים) ודומים לביצועים עם 7 פולסים (בקצב גבוהה יותר). התשלום הוא בסיבוכיות אלגוריתם החיפוש אשר נשארת נמוכה בהרבה מזה של מקודד CELP.

מבחינת איכות הדיבור המתקבלת (בדיקה סובייקטיבית) אזי ב-Space האיכות טובה מזו של CELP עם מסגרת באורך 4 msec. כמו כן עולים הביצועים של SPACE על אלו של MPE עם 6 פולסים עבור דוברים (גברים) ואילו עבור דוברות (נשים) השיפור באיכות כמעט ואינו מורגש.

לסיכום:

בעבודה זו הצגנו מספר גישות לשיפור הביצועים של מקודדי אות השארית: MPE, CELP.

ניסינו לפתור את הבעיה של מציאת הפולסים והאמפליטודות האופטימליים, במקודד MPE, בעזרת רשת נוירונים בתקווה להשיג איכות טובה יותר עם אותו מספר פולסים או לחילופין להקטין את קצב השידור עבור איכות נתונה. הניסיון לא עלה יפה והפיתרון שהתקבל בסיום תהליך החיפוש היה בדרך כלל גרוע מהפתרון שהתקבל בהפעלת האלגוריתם התת-אופטימלי.

בדקנו כיצד ניתן לעדכן את תנאי ההתחלה של מסנן הסינטזה וכיצד לשלב את העדכון באלגוריתם ה-MPE. הראינו שסכמה שבה עדכון תנאי ההתחלה בוצע אחרי שקובעים פולסים עדיפה על פני סכמה שבה העדכון בוצע קודם שקובעים פולסים.

הראינו שתכנון מילון וקטורי אופטימלי לתנאי ההתחלה דורש הגדרת פונקציית עיוות פרמטרית מורכבת שקשה להתאימה לדפוס של אלגוריתם ה-LBG.

הצגנו תוצאות שהתקבלו עם מילונים אשר בגבול (מילונים אינסופיים) מתלכדים עם הקוונטיזר האופטימלי. נסיונות אלו הובילו למודל חדש המשלב באות העירור פולסים וקוד.

מקודד ה-Space המבוסס על מודל זה השיג ביצועים טובים מביצועי מקודד ה-MPE ב-9.6 Kbps ובסיבוכיות נמוכה הזו של מקודד CELP באותו הקצב.

כדי לשפר את האיכות ניתן להתמקד במספר כיוונים:

הראשון, לתכנן מילון אשר יתחשב בקריטריון הטיב של המקודד (במקום מילון שמאוכלס באקראי). תכנון כזה עשוי לידרוש הפעלה של קידוד וקטורי למקדמי חזאי ה-LPC [27] וזאת כדי שניתן יהיה להגדיר את הבעיה בצורה פשוטה יותר.

כיוון שני יכול להיות שילוב ומיתוג של מרכיבים נוספים באות השארית, מלבד פולסים וקוד, שיובילו לאלגוריתמים פשוטים מחד ובעלי איכות גבוהה מאידך.

נספח א

בנספח זה נתאר את הקצאת הסיביות לכל מרכיב במערכת שבה מבוצע קידוד פולסים וקוד. התאור הוא לכל מרכיב בנפרד כך שניתן בקלות לחשב את הקצב הכולל של מערכות עם פרמטרים שונים המשלבים קוד ופולסים.

a. הקצאת סיביות למקדמי ה-LPC

N=192 - אינטרוול עדכון מקדמי ה-LPC.
 R=41 - סיביות לקידוד סט מקדמי ה-LPC לפי הסכמה של LPC-10 [6].

סה"כ - $R_{lpc} = 41 * 8000 / 192 = 1708$ סיביות לשניה.

b. הקצאת סיביות בקידוד פולסים

N - מסגרת זמן אנליזה.
 K - מספר פולסים במסגרת.

מבצעים נרמול לאמפליטודות לפי הגדול ביותר (בערך מוחלט) ומקדדים לפי ההקצאה הבאה [31]:

1. 6 סיביות לגורם הנרמול.
2. 4 סיביות לכל פולס נוסף.
1. 1 סיבית לסימן הפולס לפיו מבצעים נרמול.

נסמן:

Rm - סך הסיביות המוקצות למיקומים במסגרת זמן באורך N דגמים.
 Rb - סך הסיביות המוקצות לאמפליטודות במסגרת זמן באורך N דגמים.

עבור מסגרת זמן אנליזה של 8 msec (N=64):

פולסים לשניה	פולסים למסגרת	Rm [bits]	Rb [bits]	סיביות לשניה
125	1	6	7	1625
250	2	12	11	2875
375	3	18	15	4125
500	4	23	19	5250
625	5	28	23	6375
750	6	33	27	7500
875	7	38	31	8625
1000	8	43	35	9750

טבלא א-1 - הקצאת סיביות בקידוד פולסים.
 Table A-1 - Bit allocation for pulses coding.

הערה - עבור K=24 מבוצע קידוד המיקומים על ידי שידור הפרשים בין מיקומי הפולסים במסגרת ועל ידי כך לחסוך בסיביות.

c. הקצאת סיביות לקידוד קוד [32]

N - מסגרת זמן האנליזה.
 L - מספר הוקטורים במילון.

אינדקס הוקטור הנבחר במילון - $\log_2(L)$.
 גורם הגבר - 5 סיביות.

סה"כ - $R_{code} = 8000(\log_2(L) + 5) / N$ סיביות לשניה.

L	Rcode [bits/sec]		
	N=64	N=32	N=20
2	750	1500	2400
4	875	1750	2800
8	1000	2000	3200
16	1125	2250	3600
32	1250	2500	4000
64	1375	2750	4400
128	1500	3000	4800
256	1625	3250	5200
512	1750	3500	5600
1024	1875	3750	6000

טבלא א-2 - הקצאת סיביות בקידוד קוד.
 Table A-2 - Bit allocation for code coding.

נספח ב

בנספח זה מתוארת הסיבוכיות (בכפלים לדגם) של אלגוריתמים שתוארו בפרקים 1-7. הסיבוכיות הכרוכה בחישוב מקדמי ה-LPC וה-Pitch חושבה בנפרד בסעיפים a, b.

a. סיבוכיות חישוב מקדמי LPC

- m - אורך בלוק לחישוב מקדמי אוטוקורלציה באנליזת LPC.
- M - אנטרופל עדכון מקדמי LPC.
- P - סדר המודל ה-LPC.

- m - הכפלה בחלון
- m(P+1) - חישוב מקדמי אוטוקורלציה
- P² - פתרון המשוואות הנורמליות (בעזרת LEVINSON DURBIN)

סה"כ פעולות למסגרת: $O(m+P^2+m(P+1))$

סה"כ פעולות לדגם: $Tlpc=O((m+P^2+m(P+1))/M)$

אצלנו - $m=256, M=192, P=10$ ולכן $Tlpc=O(16.5)$

b. סיבוכיות הקביעה של פרמטרי חזאי Pitch כפי שמומש אצלנו

- Rmin+Rmax - תחום ערכי מחזור Pitch.
- N - אנטרופל עדכון ה-Pitch.
- M - אורך הבלוק לחישוב מקדמי קורלציה (מנורמלים באנרגיה) לחישוב מקדמי ה-Pitch (q taps).
- P - סדר המודל ה-LPC.

נגדיר: $D=Rmax-Rmin$

- NP - חישוב אות השארית
- 2DM - חישוב מקדמי קורלציה מנורמלים
- q³ - חישוב מקדמי ה-Pitch

סה"כ פעולות למסגרת - $O(NP+2DM+q^3)$

סה"כ פעולות לדגם - $Tpitch=O((NP+2DM+q^3)/N)$

אצלנו - $Rmax=120, Rmin=24, D=96, P=10, M=128, N=64, q=3$

ולכן $Tpitch=O(394)$

c. סיבוכיות הקביעה של פרמטרי חזאי Pitch כפי שמוצע ב-[5]

פעולות לדגם :

10 -
64*2 -

סינון
עדכון מקדמי קורלציה בכל דגם

פעולות בכל 10 msec. (80 דגמים)

49 -
q³=27 -

הכפלה בחלון
חישוב מקדמי ה-Pitch

סה"כ פעולות בכל 10 msec (עדכון בכל דגם):

$$80(10+128)+49+27=11116$$

סה"כ פעולות לדגם: $T_{pitch}=O(140)$

אם מבצעים עדכון מקדמי הקורלציה כל דגם רביעי אז סה"כ פעולות
לדגם:

$$T_{pitch}=O(43)$$

d. סיבוכיות אנליזת Multi Pulse בשיטת קווריאנט ללא Pitch Predictor

- N - גודל מסגרת האנליזה.
- K - מספר פולסים במסגרת הזמן.
- P - סדר מודל ה-LPC.

חישובים הכרחיים (ללא תלות במספר הפולסים במסגרת)

- P - חישוב מקדמי המסנן המשקלל
 - NP - חישוב תגובת הזכרון
 - NP - חישוב אות מקור משוקלל (ללא תגובת זכרון)
 - NP - חישוב $h_w(n)$
 - $0.5(N+1)N$ - חישוב $\phi(i,i)$ (אוטוקורלציה ל $h_w(n)$)
 - $0.5(N+1)N$ - חישוב α (קרוס קורלציה דיבור-מסנן משוקללים)
- השתמשנו ב: $N+(N-1)+\dots+1=0.5(N+1)N$

סיבוכיות חישוב פולסים

- KN - עדכון α (מקרה גרוע ביותר)
 - $O(0.25K^2(1+K)^2)$ - זה-אופטימיזציה לאמפליטודות
- (השתמשנו ב: $1^3+2^3+\dots+K^3=0.25K^2(1+K)^2$)

- NP - חישוב אות משוחזר

סה"כ פעולות למסגרת: $O(P+4NP+N(N+1)+KN+0.25K^2(1+K)^2)$

סה"כ פעולות לדגם: $T_{mpe}=O((P+4NP+N(N+1)+KN+0.25K^2(1+K)^2)/N)$

אצלנו $N=64, P=10$ ולכן:

פולסים לשניה	פולסים למסגרת	פעולות לדגם
125	1	106
250	2	107
375	3	108
500	4	110
625	5	113
750	6	118
875	7	124
1000	8	133
1125	9	145
1250	10	162

טבלא ב-1 - סיבוכיות אלגוריתם MPE - אנליזת קווריאנט.
Table B-1 - Complexity of MPE algorithm, covariance analysis.

3 לחישוב סיבוכיות אנליזה עם Pitch Predictor, בעל מקדמים, יש לבצע את השינויים הבאים :

- N(P+3) - חישוב $h_w(n)$
- N(P+3) - חישוב אות משוחזר
- N(P+3) - חישוב תגובת הזכרון

שהם תוספת של 9 פעולות כפל לדגם בהשוואה לתוצאות של אנליזה בלי חזאי Pitch.

e. סיבוכיות אנליזת Multi Pulse בשיטת אוטוקורלציה ללא Pitch Predictor

- N - גודל מסגרת האנליזה.
- K - מספר פולסים במסגרת הזמן.
- P - סדר מודל ה-LPC.
- L - מספר הדגמים השונים מאפס בתגובה המשוקללת $h_w(n)$.

חישובים הכרחיים (ללא תלות במספר הפולסים במסגרת)

- P - חישוב מקדמי המסנן המשקלל
- NP - חישוב תגובת הזכרון
- NP - חישוב אות מקור משוקלל (ללא תגובת זכרון)
- LP - חישוב $h_w(n)$
- $0.5(L+1)L$ - חישוב $\phi(i,i)$ (אוטוקורלציה ל- $h_w(n)$)
- NL - חישוב α (קרוס קורלציה דיבור-מסנן משוקללים)

השתמשנו ב : $L+(L-1)+\dots+1=0.5(L+1)L$

סיבוכיות חישוב פולסים

- KL - עדכון α (מקרה גרוע ביותר)
 - $O(0.25K^2(1+K)^2)$ - רה-אופטימיזציה לאמפליטודות
- (השתמשנו ב : $1^3+2^3+\dots+K^3=0.25K^2(1+K)^2$)

- NP - חישוב אות משוחזר
- סה"כ פעולות למסגרת:

$O(P+3NP+0.5L(L+1)+L(P+K+N)+0.25K^2(1+K)^2)$

סה"כ פעולות לדגם:

$T_{mpe}=O(P+3NP+0.5L(L+1)+L(P+K+N)+0.25K^2(1+K)^2)/N$

עבור $N=64, P=10, L=10$

פולסים לשניה	פולסים למסגרת	פעולות לדגם
125	1	42
250	2	43
375	3	44
500	4	45
625	5	48
750	6	51
875	7	56
1000	8	65
1125	9	76
1250	10	91

טבלא ב-2 - סיבוכיות אלגוריתם MPE - אנליזת אוטוקורלציה.
Table B-2 - Complexity of MPE algorithm, autocorrelation analysis.

f. סיבוכיות אנליזת CELP עם Pitch Prediction וחיפוש בתחום התדר

- L - מספר הוקטורים במילון.
- N - גודל מסגרת האנליזה.
- P+3 - מספר המקדמים במסנן החיזוי (LPC+Pitch).
- P - חישוב מקדמי המסנן המשקלל
- N(P+3) - חישוב תגובת הזכרון
- N(P+3) - חישוב אות מקור משוקלל $s_w(n)$ (ללא תגובת זכרון)
- N(P+3) - חישוב $h_w(n)$
- $2N \log_2(N)$ - FFT ל- $s_w(n), h_w(n)$
- LN - חישוב תגובת המסנן לכל הסדרות
- LN - חישוב אנרגיית התגובות
- 2LN - חישוב מכפלה פנימית בין דיבור משוקלל לתגובות
- N(P+3) - חישוב אות משוחזר

סה"כ פעולות למסגרת: $2 \log_2(N) + 4LN + 4N(P+3) + P$

סה"כ פעולות לדגם:

$T_{celp} = O((2 \log_2(N) + 4LN + 4N(P+3) + P) / N)$

אצלנו $P+3=13$ ולכן הסיבוכיות עבור מסגרות של 64, 32 דגמים תהיה:

L	פעולות לדגם	
	N=32	N=64
2	70	72
4	78	80
8	94	96
16	126	128
32	190	192
64	318	320
128	574	576
256	1086	1088
512	2110	2112
1024	4158	4160

טבלא ב-3 - סיבוכיות אלגוריתם CELP.
Table B-3 - Complexity of CELP algorithm.

.g סיבוכיות אנליזה הממתגת בין עירור פולסים וקוד ללא חזאי

.Pitch

- L - מספר הוקטורים במילון.
- N - גודל מסגרת האנליזה.
- K - מספר פולסים מקסימלי במסגרת הזמן (עבור המקרה P^k).
- K_1 - מספר הפולסים שנקבעים קודם לשילוב הקוד.
- P - סדר מודל ה-LPC.

חישובים הכרחיים (ללא תלות בקוד ובמספר הפולסים במסגרת)

- P - חישוב מקדמי המסנן המשקלל
- NP - חישוב תגובת הזכרון
- NP - חישוב אות מקור משוקלל (ללא תגובת זכרון)
- NP - חישוב $h_w(n)$
- $N \log_2(N)$ - FFT ל $h_w(n)$
- LN - חישוב תגובת המסנן לכל הסדרות
- LN - חישוב אנרגית התגובות

פעם אחת מבצעים חיפוש K פולסים ללא ערוך קוד

- $0.5(N+1)N$ - חישוב $\phi(i,i)$ (אוטוקורלציה ל- $h_w(n)$)
- $0.5(N+1)N$ - חישוב α (קרוס קורלציה דיבור-מסנן משוקללים)
- KN - עדכון α (מקרה גרוע ביותר)
- $O(0.25K^2(1+K)^2)$ - רה-אופטימיזציה לאמפליטודות

במצב זה מוכנים α, ϕ , מיקומים ואמפליטודות לכל הפולסים שקודמים לקוד.

$(K-K_1)$ פעמים מבצעים (לא כולל את המיקרה שהקוד קודם לכל הפולסים)

- NP - חישוב התגובה לפולסים שנמצאו והחסרתם מהאות שמקרבים
- $(K-K_1)$ פעמים מבצעים

- $N \log_2(N)$ - FFT לאות שמקרבים
- $2LN$ - חישוב מכפלה פנימית בין האות שמקרבים לתגובות
- NP - חישוב האות שמקרבים (ללא תגובה לקוד ולפולסים)

$(K-K_1-2)$ פעמים מבצעים חיפוש של עד $(K-K_1-1)$ פולסים

- $0.5(N+1)N$ - חישוב α (קרוס קורלציה דיבור-מסנן משוקללים)
- $(K-K_1-1)N$ - עדכון α (מקרה גרוע ביותר)
- $O(0.25(K-1)^2K^2)$ - רה-אופטימיזציה לאמפליטודות

סיום

- $O(K^2)$ - $K-K_1+1$ פעמים חישוב אנרגית שגיאה לכל מקרה
- NP - חישוב אות משוחזר

בגלל ריבוי הביטויים המרכיבים את סך פעולות הכפל למסגרת נציין רק את הביטויים הדומיננטים לסיבוכיות האלגוריתם לחישוב דגם מוצא:

$$T_{SPACE} \approx P(4+2(K-K_1)) + L(2+K-K_1) + (1+K-K_1)\log_2(N) + 0.5(N+1)(K-K_1)$$

סיבוכיות אנליזה שבה בודקים את כל המקרים לשלוב סדרת קוד ב-K פולטים (מתאים ל- $K_1=0$):

$$T_{K_1=0} \approx P(4+2K) + L(2+K) + (K+1)\log_2(N) + 0.5K(N+1)$$

עבור: $T_{SPACE}=650$ $\langle=== N=64, L=32, K=6, K_1=0$ פעולות כפל לדגם.
 $T_{SPACE}=380$ $\langle=== N=64, L=32, K=6, K_1=3$ פעולות כפל לדגם.

לחישוב סיבוכיות אנליזה עם חזאי Pitch, בעל 3 מקדמים, יש להחליף בביטויים ל $T_{K_1=0}$, את P (סדר חזאי ה-LPC) ב- $P+3$:

$$T_{SPACE} \approx (P+3)(4+2(K-K_1)) + L(2+K-K_1) + (1+K-K_1)\log_2(N) + 0.5(N+1)(K-K_1)$$

$$T_{K_1=0} \approx (P+3)(4+2K) + L(2+K) + (K+1)\log_2(N) + 0.5K(N+1)$$

עבור: $T_{SPACE}=700$ $\langle=== N=64, L=32, K=6, K_1=0$ פעולות כפל לדגם.
 $T_{SPACE}=410$ $\langle=== N=64, L=32, K=6, K_1=3$ פעולות כפל לדגם.

h. סיבוכיות אנליזת CELP עם פולטים במקום Pitch Predictor
 (הפולטים מחושבים לפני קוד).

- L - מספר הוקטורים במילון.
- N - גודל מסגרת האנליזה.
- K - מספר פולטים במסגרת הזמן.
- P - סדר מודל ה-LPC.

סיבוכיות חישוב פולטים

- P - חישוב מקדמי המסנן המשקלל
- NP - חישוב תגובת הזכרון
- NP - חישוב אות מקור משוקלל (ללא תגובת זכרון)
- NP - חישוב $h_w(n)$
- $0.5(N+1)N$ - חישוב $\phi(i, i)$ (אוטוקורלציה ל $h_w(n)$)
- $0.5(N+1)N$ - חישוב α (קרוס קורלציה דיבור-מסנן משוקללים)
- KN - עדכון α (מקרה גרוע ביותר)
- $O(0.25K^2(1+K)^2)$ - ה-אופטימיזציה לאמפליטודות

סיבוכיות מציאת סדרת הערוך האופטימלית

- NP - חישוב תגובת לפולטים
- $2N \log_2(N)$ - FFT ל $h_w(n), s_w(n)$
- LN - חישוב תגובת המסנן לכל הסדרות
- LN - חישוב אנרגית התגובות
- $2LN$ - חישוב מכפלה פנימית בין דיבור משוקלל לתגובות
- NP - חישוב אות משוחזר

סה"כ פעולות למסגרת:

$$O(P+5NP+N(N+1)+KN+0.25K^2(1+K)^2+4LN+2N \log_2(N))$$

סה"כ פעולות לדגם:

$$T_{celp} = O((P+5NP+N(N+1)+KN+0.25K^2(1+K)^2+4LN+2N \log_2(N))/N)$$

עבור $N=64, K=3, L=1024$ $T_{celp}=4226$ כפלים לדגם.

REFERENCES

- [1] J. Makhoul, "Linear Prediction: A Tutorial Review", Proc. of the IEEE, vol. 63, no. 4, pp. 561-580, April 1975.
- [2] J.D. Markel and A.H. Gray Jr., "A Linear Prediction Vocoder Simulation Based upon the Autocorrelation Method", IEEE Trans. Acoust. Speech and Signal Processing, vol. ASSP-24 pp. 459-473, Dec. 1976.
- [3] J.D. Markel and A.H. Gray Jr., "Linear Prediction of Speech", Springer-Verlag, Berlin-New York, 1976.
- [4] L.R. Rabiner and R.W. Schafer, "Digital Processing of Speech Signals, Prentice-Hall 1978.
- [5] R.V. Cox and R.E. Crochiere, "A Single chip speech periodicity detector", Proc. IEEE Int. Conf. Acoust. Speech Signal Processing, Paris, France, May 1982.
- [6] T.E. Tremain, "The Government Standard Linear Prediction Coding Algorithm: LPC-10", Speech Technology, pp. 40-49, April 1982.
- [7] D.Y. Wong, "On Understanding the Quality Problems of LPC Speech", Proc. IEEE Int. Conf. ASSP 1980, pp. 725-728.
- [8] N.S. Jayant and P. Noll, "Digital Coding of Waveform", Prentice-Hall 1984.
- [9] B.S. Atal, "Predictive Coding of Speech at Low Bit Rates", IEEE Trans. Commun. vol. COM-30, pp. 600-614, April 1982.
- [10] B.S. Atal and M.R. Schroeder, "Predictive Coding of Speech Signals and Subjective Error Criteria", IEEE Trans. Acoust. Speech and Signal Processing, vol. ASSP-27, No. 3, June 1979.
- [11] R.P. Ramachandran and P. Kabal, "Stability and Performance Analysis of Pitch Filters in Speech Coders", IEEE Trans. Acoust. Speech and Signal Processing, vol. ASSP-35, No. 7, July 1987.
- [12] B.S. Atal and J.R. Remde, "A New Model of LPC Excitation for Producing Natural Sounding Speech at Low Bits Rates", Proc. IEEE Int. Conf. ASSP 1982, pp. 614-617.

- [13] M. Berouti et. al., "Efficient Computation and Encoding of the Multipulse Excitation for LPC", Proc. IEEE Int. Conf. ASSP 1984, pp. 10.1.1-10.1.4.
- [14] S. Singhal and B.S. Atal, "Improving Performance of Multi-Pulse LPC Coders At Low Bit Rates", Proc. IEEE Int. Conf. ASSP 1984, pp. 1.3-1.4.
- [15] B.S. Atal and M.R. Schroeder, "Stochastic Coding of Speech Signals at Very Low Bit Rates", Proc. IEEE Int. Conf. Commun. ICC84, pp. 1610-1613, 1984.
- [16] B.S. Atal, "High Quality Speech at Low Bit Rates: Multipulse and Stochastically Excited Linear Prediction Coder", Proc. IEEE Int. Conf., pp. 1681-1684, 1986.
- [17] M.R. Schroeder and B.S. Atal, "Code Excited Linear Prediction (CELP): High Quality Speech at Very Low Bit Rates", Proc. IEEE Int. Conf., ASSP, pp. 937-939, 1985.
- [18] I.M. Transcoso and B.S. Atal, "Efficient Procedures for Finding the Optimum Innovation in Stochastic Coders", Proc. IEEE Int. Conf., ASSP, pp. 2375-2378, 1986.
- [19] M. Fleisher, "Fast Processing of Autoregressive Signals By a Neural Network", The 15th Conf. of Electrical & Electronics Engineerings in Israel, April 1987.
- [20] W. Jeffrey and R. Rosner, "Neural Network Processing As a Tool For Function Optimization", AIP Conf. Proc. 151, New York 1986, pp. 241-246.
- [21] R.P. Lippman, "An Introduction to Computing with Neural Nets", IEEE ASSP Magazine April 1987, pp. 241-246.
- [22] R.E. Scraton, "Basic Numerical Methods", London, Arnold, 1985.
- [23] D.G. Luenberger, "Linear and Nonlinear Programming", 2nd ed. Addison-Wesley, 1984.
- [24] H.J. Trusseli and M.R. Clivanlar, "Optimal Initial Conditions and Pulses Values For Multipulse Speech Coding", Proc. Int. Conf., ASSP, pp. 7.9.1-7.9.4, 1985.
- [25] S.T. Alexander, "A Simple Noniterative Speech Excitation Algorithm Using the LPC Residual", IEEE Trans. on Acoust. Speech and Signal Processing, vol. ASSP-33, No. 2, April 1985.

- [26] R.J. McEliece, "The Theory of Information and Coding", Addison-Wesley Publishing Company, 1977.
- [27] Y. Linde A. Buzo and R.M. Gray, "An Algorithm for Vector Quantizer Design", IEEE Trans. Comm. vol. COM-28, pp. 84-95, July 1980.
- [28] R.M. Gray, "Vector Quantization", IEEE ASSP Magazine, pp. 4-29, Apr. 1984.
- [29] R. Arad, "Scalar and Vector Quantization in Subband Coding of Speech", Research Thesis, Technion I.I.T., Oct. 1986 (in Hebrew).
- [30] J.H. Chen and A. Gersho, "Gain Adaptive Vector Quantization with Application to Speech Coding", IEEE Tran. Comm. vol. COM-35, NO. 9, Sep. 1987
- [31] R. Sharama, "Architecture Design of a High Quality Speech Synthesizer Based on the Multi-Pulse LPC Technique", IEEE Journal on Selected Areas in Commun., vol. SAC-3, No. 2, March 1985, pp. 377-383.
- [32] P. Kroon and B.S. Atal, "Quantization Procedures for the Excitation in CELP Coders", Proc. IEEE Int. Conf., ASSP, pp. 38.8.1-38.8.4 1987.

EXAMINATION OF APPROACHES FOR IMPROVING THE
PERFORMANCE OF RESIDUAL SPEECH CODERS

RESEARCH THESIS

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF MASTER OF SCIENCE
IN ELECTRICAL ENGINEERING

BY
ITZIK ALFY

SUBMITTED TO THE SENATE OF THE TECHNION
ISRAEL INSTITUTE OF TECHNOLOGY

SIVAN 5748

HAIFA

JUNE 1988

This research was carried out in the Signal Processing Laboratory of the Faculty of Electrical Engineering under the supervision of Prof. David Malah.

I wish to express my sincere gratitude to Prof. David Malah for his dedicated supervision throughout this work.

I am also grateful to Mrs. Ziva Avni, Mrs. Orit Midezinsky, Mr. Moshe Namer, Mr. Yoram Or-Chen, Mr. Nimrod Peleg, Mr. Dan Tal and the Communication and Signal Processing laboratories staff who helped me complete this work.

CONTENT

	<u>page</u>
ABSTRACT	1
SYMBOLS LIST	3
CHAPTER 1 : INTRODUCTION	5
CHAPTER 2 : SPEECH SIGNALS CODING	
2.1 INTRODUCTION	7
2.2 SOURCE CODERS - VOCODERS	7
2.3 EXTRACTION OF LPC PARAMETERS	8
2.4 PITCH DETECTION	9
2.5 SUMMARY	9
2.6 WAVEFORM CODERS	10
2.7 THE LPC RESIDUAL SIGNAL	10
CHAPTER 3 : RESIDUAL CODERS	
3.1 INTRODUCTION	11
3.2 APC - ADAPTIVE PREDICTIVE CODER	12
3.3 APC WITH NOISE SPECTRUM CONTROL	13
3.4 MPE - MULTI-PULSE EXCITATION	15
3.4.1 MPE MODEL	15
3.4.2 MPE ALGORITHM	18
3.4.3 IMPROVEMENTS OF THE BASIC ALGORITHM	20
3.4.4 SIMULATION RESULT	22

CONTENT (CONTINUE)

3.5 CELP - CODE EXCITED LINEAR PREDICTION .. 23

 3.5.1 SELECTING OPTIMAL CODE-WORD 23

 3.5.2 COMPLEXITY OF THE SEARCHING

 ALGORITHM 25

 3.5.3 SIMULATION RESULTS 26

CHAPTER 4 : MPE PROBLEM SOLUTION USING A NEURAL NETWORK APPROACH

4.1 INTRODUCTION 27

4.2 SOLVING OPTIMIZATION PROBLEMS USING NEURAL NETWORKS 27

4.3 SOLUTION OF THE MPE PROBLEM UTILIZING A NEURAL NETWORK 30

4.4 DISCRIPTION OF THE NETWORK DYNAMICS ... 32

4.5 SIMULATION RESULTS 33

CHAPTER 5 : UPDATING INITIAL CONDITIONS OF THE SYNTHESIS FILTER

5.1 INTRODUCTION 34

5.2 DETERMINING THE EQUATIONS FOR OPTIMAL INITIAL CONDITIONS 35

 5.2.1 MEMORY RESPONSE AS A FUNCTION OF MEMORY SAMPLES 36

CONTENT (CONTINUE)

5.3 COMBINING MEMORY UPDATING SCHEME WITH
MPE SCHEME 37

5.3.1 RE-OPTIMIZATION OF AMPLITUDES AND
INITIAL CONDITIONS 41

CHAPTER 6 : VECTOR QUANTIZATION OF INITIAL CONDITION

6.1 INTRODUCTION 43

6.2 VQ 43

6.2.1 VQ DEFINITION 44

6.2.2 GAIN ADAPTIVE VQ 46

6.3 DESIGN OF OPTIMAL CODEBOOK FOR MEMORY
VECTOR 47

6.4 VECTOR QUANTIZATION OF MEMORY VECTOR ... 49

6.5 CODING OF FIRST P SAMPLES OF OPTIMAL
MEMORY RESPONSE 53

6.6 CODING OF EQUIVALENT INPUT VECTOR \underline{x} ... 54

CHAPTER 7 : COMBINING CODEBOOK AND PULSES FOR RESIDUAL
SIGNAL CODING

7.1 INTRODUCTION 57

7.2 USING CODEBOOK AFTER LOCATING PULSES ... 58

7.3 COMBINING CODE AND PULSES FOR EXCITING
SYNTHESIS FILTER WITH PITCH PREDICTION.. 63

CONTENT (CONTINUE)

CHAPTER 8 : SUMMARY AND CONCLUSIONS 65

APPENDIX A : BIT ALLOCATION IN THE CODERS CONSIDERED ... 67

APPENDIX B : COMPLEXITY CONSIDERATIONS 69

REFERENCES : 78

ABSTRACT

In this work we examine approaches for improving the performance of residual speech coders in the range of 9.6-16 Kbps.

Existing digital speech coding techniques require binary data rates varying from 64 Kbps down to about 2.4 Kbps. At the upper end of the scale stand waveform coders which provide high speech quality and are robust to acoustic and transmission noise. The performance of the waveform coders typically falls rapidly when the data rate is reduced below 16 Kbps. At the lower end of the data rate scale stand LPC vocoders which are based on a model of the human speech production system. LPC vocoders have a synthetic quality and do not perform well under poor acoustic conditions or when a number of speakers are speaking together. Increasing the data rate of the LPC vocoder above the typical rate of 2.4 Kbps does not significantly improve the quality or acoustic robustness of the coder.

It is evident therefore that it is difficult to achieve good performance in the data rate range of 4.8-16 Kbps by using either waveform coders or LPC vocoders. A class of coders designed to operate at these rates are residual coders. These coders have in common the feature of coding both the LPC residual and the LPC filter coefficients.

At the top, both in quality and bit rate, are the full-band coders in which each residual sample is quantized individually. The APC (Adaptive Predictive Coding) system is an example to such coders. However, in order to achieve data rates of 9.6 Kbps and below, the residual coders cannot afford quantizing each residual sample individually.

One approach is MPE (Multi Pulse Excitation) in which the residual is represented by only few significant samples, which are found by an iterative sub-optimal procedure in a closed loop. CELP (Code Excited Linear Prediction) coders use blocks of white Gaussian noise to represent the residual signal.

In this work we examine approaches for improving the performance of the MPE coder:

We tried to solve the problem of determining the optimal pulses utilizing a Neural Network, hoping to improve the quality of reconstructed speech using the same number of pulses, or to reduce the bit rate for a given quality.

However, this approach did not prove itself useful, since the network provided a solution which was worse than the solution obtained from the sub-optimal algorithm.

We next examined how to use the fact that the response of the synthesis filter is composed of the initial conditions response and the excitation signal response. The idea was to determine optimal initial conditions, such that the composed responses would give better speech quality.

Two ways to include initial conditions updating with the MPE algorithm are presented. The results showed significant improvement in speech quality with comparison to the original algorithm (before coding the initial conditions).

We then applied a vector quantization technique to encode the optimal initial conditions. Determining the optimal code-book is a very complicated problem. Hence, we used sub-optimal code-books which only in the limit (infinite codebooks) are optimal.

Based on the above results, a new model is developed. This model combines pulses and code to excite the synthesis filter, and results in the SPACE (Switched Pulses and Code Excitation) algorithm which performs better than the MPE coder at 9.6 Kbps while its complexity is lower than the CELP coder, at the same bit rate.